Multidimensional Complex Stationary Centered Gaussian Autoregressive Time Series Machine Learning in Poincaré and Siegel Disks: Application for Audio and Radar Clutter Classification

Yann Cabanes

THÈSE PRÉSENTÉE

POUR OBTENIR LE GRADE DE

DOCTEUR
DE L'UNIVERSITÉ DE BORDEAUX

Ecole doctorale de Mathématiques et Informatique de Bordeaux

Spécialité **Mathématiques Appliquées**

Par **Yann CABANES**

Apprentissage dans les disques de Poincaré et de Siegel de séries temporelles multidimensionnelles complexes suivant un modèle autorégressif gaussien stationnaire centré : application à la classification de données audio et de fouillis radar

Sous la direction de **Marc ARNAUDON**,
Frédéric BARBARESCO et **Jérémie BIGOT**

Soutenue le 31 mars 2022 devant le jury composé de :

M. Jesus ANGULO	Directeur de Recherche	MINES ParisTech	Président du jury
M. Salem SAID	Chargé de recherche	CNRS, Laboratoire LJK	Rapporteur
M. Fabrice GAMBOA	Professeur des universités	Université Toulouse 3	Rapporteur
M. Pierrick LEGRAND	Maître de conférences	Université de Bordeaux	Examinateur
Mme. Alice LE BRIGANT	Maîtresse de conférences	Université Paris 1 Panthéon-Sorbonne	Examinatrice
M. Frédéric BARBARESCO	Ingénieur	Thales Land & Air Systems	Examinateur
M. Marc ARNAUDON	Professeur des universités	Université de Bordeaux	Directeur de thèse
M. Jérémie BIGOT	Professeur des universités	Université de Bordeaux	Co-directeur de thèse

Thèse CIFRE Défense Thales-Université de Bordeaux

Apprentissage dans les disques de Poincaré et de Siegel de séries temporelles multidimensionnelles complexes suivant un modèle autorégressif gaussien stationnaire centré : application à la classification de données audio et de fouillis radar

Résumé : L'objectif de cette thèse est la classification de séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré. Nous étudions le cas des séries temporelles unidimensionnelles ainsi que le cas plus général des séries temporelles multidimensionnelles. L'apport de cette thèse est à la fois méthodologique et technique. La méthodologie présentée permet de représenter les lois des séries temporelles observées dans une variété riemannienne dans laquelle la classification sera effectuée. Les étapes majeures de notre méthode sont : la définition de l'espace des coefficients du modèle paramétrique permettant de représenter les séries temporelles considérées, l'estimation des coefficients du modèle paramétrique à partir de séries temporelles observées, munir l'espace des coefficients du modèle paramétrique d'une métrique riemannienne inspirée de la géométrie de l'information et enfin l'adaptation d'algorithmes de machine learning classiques aux variétés riemanniennes obtenues. Dans le cas des séries temporelles multidimensionnelles, nous travaillerons dans un espace produit qui fait intervenir le disque de Siegel (ensemble des matrices complexes de valeurs singulières strictement inférieures à 1) muni d'une métrique riemannienne produit. En plus de l'apport méthodologique évoqué précédemment, nous apportons des outils théoriques nouveaux pour classifier des données dans la variété de Siegel : nous donnons les formules explicites du logarithme riemannien, de l'exponentielle riemannienne et de la courbure sectionnelle de la variété obtenue sur l'espace de Siegel. Notre modèle de représentation des séries temporelles complexes suivant un modèle autorégressif gaussien stationnaire centré sera appliqué à la classification de séries temporelles simulées, au clustering de fouillis radar et à la classification de séries temporelles audio stéréo stationnaires.
Mots-clés : séries temporelles suivant un modèle autorégressif gaussien stationnaire à valeurs complexes, séries temporelles multidimensionnelles, machine learning, géométrie de l'information, variétés riemanniennes, matrices Toeplitz par blocs, disque de Siegel, fouillis radar, séries temporelles audio stationnaires, corrélation spatio-temporelle.

Multidimensional complex stationary centered Gaussian autoregressive time series machine learning in Poincaré and Siegel disks: application for audio and radar clutter classification

Abstract: The objective of this thesis is the classification of complex valued stationary centered Gaussian autoregressive time series. We study the case of one-dimensional time series as well as the more general case of multidimensional time series. The contribution of this thesis is both methodological and technical. The methodology presented can be used to represent the probability distributions of the observed time series in a Riemannian manifold in which the classification will be performed. The major steps of our method are: the definition of the space of the coefficients of the parametric model used to represent the considered time series, the estimation of the coefficients of the parametric model from observed time series, to endow the space of the coefficients of the parametric model with a Riemannian metric inspired by information geometry and finally the adaptation of classical machine learning algorithms to the Riemannian manifolds obtained. In the case of multidimensional time series, we will work in a product manifold which involves the Siegel disk (set of complex matrices with singular values strictly lower than 1) endowed with a Riemannian metric. In addition to the methodological contribution mentioned previously, we bring new theoretical tools to classify data in the Siegel manifold: we give the explicit formulas of the Riemannian logarithm map, of the Riemannian exponential map and of the Siegel manifold sectional curvature. Our representation model for complex stationary centered Gaussian autoregressive time series will be applied to simulated time series classification, to radar clutter clustering and to stationary stereo audio time series classification.
Keywords: Complex stationary centered Gaussian autoregressive time series, multidimensional time series, machine learning, Information geometry, Riemannian manifold, Block-Toeplitz matrices, Siegel disk, radar clutter, stationary audio time series, spatio-temporal correlation.

Unité de recherche
Institut de Mathématiques de Bordeaux (UMR 5251)
Université de Bordeaux
351, cours de la Libération
33405 Talence, France.

Remerciements

Je remercie mon directeur de thèse Marc Arnaudon pour l'encadrement exceptionnel dont j'ai bénéficié. Marc, merci pour ta gentillesse, ta délicatesse, ta patiente, ta disponibilité, merci pour les heures que nous avons passées à discuter de mathématiques au tableau. Tu as été un vrai soutien pour moi tout au long de ma thèse. Il était très confortable pour moi de sentir que tu ferais toujours ton possible pour me soutenir.

Je remercie mon responsable chez Thales Frédéric Barbaresco grâce à qui j'ai pu découvrir la géométrie de l'information et les noms des grands mathématiciens qui y ont contribué. Merci Frédéric de m'avoir fait bénéficier de ton très large réseau de connaissances grâce auquel j'ai pu rencontrer de nombreuses personnes très enrichissantes et commencer plusieurs collaborations. Merci également de m'avoir encouragé à participer à de nombreuses conférences.

Je remercie également mon co-directeur de thèse Jérémie Bigot pour ses excellents conseils qui m'ont permis de préparer au mieux mon oral de soutenance.

Je remercie les membres de mon jury pour les discussions très enrichissantes que nous avons eues. Je remercie notamment Salem Said pour sa gentillesse et nos discussions sur les variétés quotient qui ont été pour moi très instructives.

Je remercie toutes les personnes qui ont assisté à ma soutenance de thèse. Merci à ceux qui étaient présents dans la salle le jour de ma soutenance : ma femme Camille, mon frère Léo, mes amis doctorants Paul et Yiye, mes amis Benjamin et Audrey, merci aux membres de mon jury Pierrick Legrand, Alice Le Brigant et Marc Arnaudon et enfin merci aux deux chercheurs de l'IMB qui sont venus assister à mon exposé. Merci à ma mère Sophie qui aurait souhaité être présente et à ma soeur Maïwenn d'être venue partager le pot thèse. Je remercie également toutes les personnes qui ont assisté à ma soutenance en visioconférence ou même en rediffusion. J'ai rarement l'occasion de partager mes connaissances de thèse avec ma famille et mes proches, cela me fait vraiment plaisir qu'ils aient pu, pour la plupart, assister à ma soutenance.

Je remercie Nina Miolane, Nicolas Guigui, Alice Le Brigant et tous les membres qui travaillent activement sur la librairie Python geomstats de m'avoir appris à partager des codes. Les trois hackathons auxquels j'ai participé ont été très formateurs. J'ai utilisé le formalisme rigoureux nécessaire au partage de connaissance appris durant ces hackathons pour rédiger mes codes personnels tout au long de ma thèse. Je suis convaincu que la mise en ligne de mes codes sur une librairie opensource est une excellente façon de partager des connaissances et de valoriser mon travail de thèse. Sans cet effort de mise en commun des connaissances, les codes que j'ai rédigés durant ma thèse et auxquels j'ai consacré des centaines d'heures de travail n'auraient probablement jamais été réutilisés par la suite.

Je remercie Pierre-Yves Lagrave pour les discussions scientifiques très constructives que nous avons eues. Deux articles sont le fruit de nos travaux communs. J'espère que nous aurons l'occasion de collaborer de nouveau à l'avenir.

Je remercie Frank Nielsen avec qui j'ai eu la chance de publier un article et d'échanger sur le Japon.

Je remercie Claude Adnet et Narayan Bernardin de m'avoir transmis les données radar sur lesquelles j'ai travaillé tout au long de ma thèse.

Je remercie Ben Jeuris et Raf Vandebril pour l'article qu'ils ont rédigé ensemble et sur lequel je me suis beaucoup appuyé tout au long de ma thèse. Les contributions théoriques majeures de ma thèse s'appuient directement sur leurs travaux. Je les remercie également d'avoir répondu à mes questions par mail et de m'avoir envoyé leurs codes MATLAB.

Je remercie Le Yang et Alice Le Brigant, dont les thèses ont toutes deux été encadrées par Marc Arnaudon et Frédéric Barbaresco, d'avoir rédigé avec soin des thèses sur lesquelles j'ai pu m'appuyer pour mes propres travaux.

Je remercie la DGA et l'AID d'avoir contribué au financement de cette thèse CIFRE Défense (convention CIFRE AID N°2017.0008 & ANRT N°2017.60.0062).

Je remercie mon collègue chez Thales et ami Daniel Brooks pour tous les bons moments passés ensemble. Nous avons joué au tennis, skié avec le CE de Thales et participé à de nombreuses conférences ensemble. Ce fut toujours un plaisir.

Je remercie mon ami et collègue doctorant à l'IMB Baptiste Huguet, de m'avoir accueilli dans son bureau au début de ma thèse et de m'avoir assisté dans les diverses démarches administratives à mon arrivée. Baptiste, je te remercie également d'avoir partagé avec moi les cours que tu avais préparé avec soin pour les L1 bio, ce fut pour moi un gain de temps significatif.

Je remercie Paul Freulon, pour les excellents moments que nous avons passés à jouer au tennis et à discuter. Merci également de m'avoir aidé à répéter mon oral de soutenance.

Je remercie ma mère, Sophie, d'avoir fait tant d'efforts pour me soulager de toute tâche logistique la semaine précédent ma soutenance (merci notamment pour le délicieux pot de thèse que tu as préparé avec Camille).

Je remercie mille fois ma femme Camille de m'avoir soutenu tout au long de ces quatre années de thèse. Camille, merci notamment pour cette dernière année au cours de laquelle tu t'es beaucoup occupée de notre fille Lila pour que je puisse terminer plus rapidement la rédaction de mon manuscrit. Je remercie ma fille Lila qui a fêté son premier anniversaire le jour de ma soutenance de thèse d'être si merveilleuse, un vrai rayon de soleil dans ma vie.

Résumé général

Le contenu de ce chapitre introductif est également rédigé en anglais dans le chapitre 1. Dans cette thèse, nous présentons une méthode pour classifier des séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré. Notre motivation initiale vient du traitement du signal radar et en particulier de la classification du fouillis radar, que nous détaillons dans la section I. Ce problème a déjà été abordé dans de précédents travaux, notamment par Frédéric Barbaresco [9, 12, 34, 37, 38], Le Yang [6, 82], Alice Le Brigant [15] et Alexis Decurninge [3, 26]. Nous présentons les différents articles et manuscrits de thèse qui ont servi de support au présent manuscrit dans la section II. Nous résumons chaque chapitre de cette thèse dans la section III. Dans la section IV, nous nous concentrons sur les principales contributions de cette thèse. Nous proposons quelques perspectives des possibles futurs développements de cette thèse dans la section V.

I Motivations radar

La motivation initiale de l'étude des séries temporelles à valeurs complexes qui suivent un modèle autorégressif gaussien stationnaire centré vient du traitement du signal radar, en particulier de l'étude du fouillis radar. Dans le langage radar, nous distinguons les objets mobiles d'intérêt premier que nous appelons les cibles et l'information liée à l'environnement du radar que nous appelons le fouillis. Le fouillis radar est donc l'information enregistrée par un radar liée aux mers, aux forêts, aux champs, aux villes et autres éléments environnementaux qui entourent le radar. Afin de mieux distinguer les cibles du fouillis, il peut être intéressant de développer des algorithmes de machine learning permettant de reconnaître différents types de fouillis. La connaissance du fouillis radar est utilisée dans l'appendice F pour obtenir un estimateur de détection à taux de fausses alarmes constant (CFAR).

Pour étudier les caractéristiques du fouillis radar, il est courant de supposer que les séries temporelles à valeurs complexes liées au fouillis suivent un modèle autorégressif gaussien stationnaire centré [13]. L'hypothèse de stationnarité des séries temporelles dans le cadre de l'étude du fouillis radar est justifiée par des temps d'observation extrêmement courts d'une même zone de l'environnement. Les lois de probabilités de ces séries temporelles sont représentées dans des variétés riemanniennes dans les travaux de Le Yang [6, 82] et d'Alice Le Brigant [15]. Nous reprenons ce modèle de représentation et l'appliquons au clustering de fouillis radar dans le chapitre 8.

Afin d'affiner l'étude des caractéristiques du fouillis radar, nous avons voulu ajouter à l'information temporelle contenue dans chaque série temporelle une information spatiale en étudiant la corrélation entre des séries temporelles enregistrées dans des zones spatialement proches. Pour cela, nous avons développé un modèle spatio-temporel qui est étudié dans l'appendice G. Afin de munir l'espace des coefficients de ce modèle spatio-temporel d'une métrique, nous étudions le cas plus général des séries temporelles stationnaires multidimentionnelles qui deviennent ainsi l'objet d'étude principal de cette thèse.

II Travaux associés

L'étude des séries temporelles radar stationnaires unidimensionnelles a été réalisée par Frédéric Barbaresco dans [8–12, 39]. Dans ces travaux, les séries temporelles radar stationnaires sont représentées dans l'espace produit $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ où $\mathcal{D}$ représente le disque unité complexe. Cet espace est muni d'une métrique riemannienne inspiré de la géométrie de l'information. L'espace $\mathbb{R}_+^*$ permet de représenter la puissance quadratique moyenne de la série temporelle étudiée. L'espace produit $\mathcal{D}^{n-1}$ représente les coefficients du modèle autorégressif, il représente donc l'information Doppler contenue dans la série temporelle.

L'algorithme de Burg est utilisé pour estimer les coefficients du modèle autorégressif à partir d'une série temporelle enregistrée. Cet algorithme est présenté dans les travaux de Frédéric Barbaresco et Alexis Decurninge [3, 26].

La métrique riemannienne construite sur l'espace $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ est présentée par Frédéric Barbaresco dans [37], [41], [35], [34], [33] et les travaux connexes [23], [53], [29]. Cette métrique est également détaillée dans les travaux de thèse de Le Yang [6, 82] et dans les travaux de thèse d'Alice Le Brigant [15]. Nous référons au livre de Shun-ichi Amari [73] pour une présentation complète des outils de géométrie de l'information utilisés pour construire cette métrique. Nous notons $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ la variété riemannienne présentée dans ces travaux : la variété $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ correspond à l'espace $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$

muni d'une métrique riemannienne inspirée de la géométrie de l'information. Cette métrique Riemannienne est présentée dans la section 4.1.3. Le calcul de la moyenne et de la médiane dans cette variété est utilisé pour détecter des cibles radar dans les travaux de Le Yang [6, 82]. L'étude des courbes de la variété $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ est appliquée à la reconnaissance de cibles radar dans le travail d'Alice Le Brigant [15]. Dans le travail de thèse présenté ici, la variété $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ sera utilisée pour le clustering de fouillis radar et plus généralement pour la classification de séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré.

L'un des objectifs de cette thèse est l'étude des séries temporelles multidimensionnelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré. Comme dans le cas des séries temporelles unidimensionnelles, il est possible de représenter les séries temporelles multidimensionnelles par les coefficients du modèle autorégressif. Dans le cas de séries temporelles multidimensionnelles, ces coefficients autorégressifs sont des matrices carrées. Dans l'article écrit par Ben Jeuris et Raf Vandebril [45], les coefficients matriciels du modèle autorégressif sont légèrement modifiés pour appartenir au disque de Siegel $\mathcal{SD}_N$ (ensemble des matrices complexes $N \times N$ de valeurs singulières strictement inférieures à 1). Les séries temporelles stationnaires multidimensionnelles seront alors représentées dans l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$, où $\mathcal{H}_N^+$ est l'espace des matrices hermitiennes définies positives. Cet espace sera muni d'une métrique riemannienne produit dont la construction est détaillée dans l'article de Ben Jeuris et Raf Vandebril [45]. La métrique produit sur l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ induit une métrique riemannienne sur les espaces $\mathcal{H}_N^+$ et $\mathcal{SD}_N$. La métrique du disque de Siegel $\mathcal{SD}_N$ a été étudiée par Frédéric Barbaresco dans [38], [36] et dans l'article connexe [46]. A notre connaissance, l'application logarithme riemannien, l'application exponentielle riemannienne et la courbure sectionnelle de la variété riemannienne définie sur l'espace de Siegel $\mathcal{SD}_N$ n'étaient pas connues, ce sont des apports de cette thèse qui ont été résumés dans l'un de nos précédents articles [21]. Ces outils géométriques sont indispensables à l'utilisation de certains algorithmes de machine learning, notamment les algorithmes impliquant un calcul de la moyenne comme l'algorithme des k-means.

III Principaux enjeux de la thèse

Cette thèse a pour objectif de mettre à disposition du lecteur la grande majorité des outils méthodologiques et techniques que nous utiliserons pour la classification des séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré. Parmi les outils présentés dans cette thèse, certains étaient déjà connus et certains sont nouveaux, nous détaillons les éléments nouveaux dans la section IV. La structure de cette thèse est liée à la méthodologie utilisée. Dans le chapitre 2 nous présentons plusieurs espaces de représentation des séries temporelles étudiées. Dans le chapitre 3, nous munissons ces espaces d'une métrique riemannienne inspirée de la géométrie de l'information. Dans le chapitre 4, nous détaillons la métrique des variétés riemanniennes obtenues. Dans le chapitre 5, nous présentons des algorithmes de machine learning adaptés à la structure de variété riemannienne. Dans les chapitres 6, 7, 8 et 9 nous utilisons certains algorithmes de machine learning présentés dans le chapitre 5 sur les variétés détaillées dans le chapitre 4 pour classifier des séries temporelles stationnaires. Nous détaillons maintenant le contenu de chaque chapitre.

III.1 Résumé du chapitre 2 : Séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré

Dans le chapitre 2, nous présentons différents modèles paramétriques associés aux processus stochastiques discrets à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré.

On note $n-1$ l'ordre des processus stochastiques autorégressifs présentés dans cette section.

Dans le cas des séries temporelles unidimensionnelles, cela s'écrit :

$$u(k) + \sum_{i=1}^{n-1} a_i^{n-1}\, u(k-i) = w^{n-1}(k) \tag{1}$$

où a_i^{n-1} sont les coefficients de prédiction et les erreurs de prédiction $w^{n-1}(k)$ sont des processus gaussiens complexes centrés indépendants et identiquement distribués de variance σ^2. Une telle série temporelle est donc entièrement paramétrée par les coefficients du modèle autorégressif $\left(a_1^{n-1}, \ldots, a_{n-1}^{n-1}\right)$ et par le coefficient σ^2. Nous définissons les coefficients d'autocorrélation $r_i := \mathbb{E}\left[u\left(k+i\right) u\left(k\right)^*\right]$. De façon équivalente, une série temporelle peut être représentée par sa variance r_0 et les coefficients du modèle autorégressif $\left(a_1^1, \ldots, a_{n-1}^{n-1}\right)$. Les coefficients a_i^i sont appelés coefficients de réflexion et seront notés μ_i, ils appartiennent au disque unité complexe $\mathcal{D}$. En utilisant cette représentation, le processus stochastique discret étudié est donc représenté par les coordonnées $(r_0, \mu_1, \ldots, \mu_{n-1})$ dans l'espace $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$. Il est également possible de représenter une série temporelle par ses coefficients d'autocorrélation $(r_0, r_1, \ldots, r_{n-1})$. Ces coefficients déterminent entièrement la matrice d'autocorrélation $\mathbf{R}$ du processus stochastique discret qui est une matrice Toeplitz hermitienne définie positive :

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1^* & r_2^* & \dots & r_{n-1}^* \\ r_1 & r_0 & r_1^* & \dots & r_{n-2}^* \\ r_2 & r_1 & r_0 & \dots & r_{n-3}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & r_{n-3} & \dots & r_0 \end{bmatrix}. \tag{2}$$

Nous notons Z un vecteur qui suit une loi normale complexe. La loi de Z est caractérisée par sa moyenne $\mu := \mathbb{E}[Z]$, sa matrice de covariance $\Gamma := \mathbb{E}\left[ZZ^H\right]$ et sa matrice de relation $C := \mathbb{E}\left[ZZ^T\right]$. Nous notons $Z \sim \mathcal{CN}(\mu, \Gamma, C)$ la loi gaussienne complexe de moyenne μ, de matrice de covariance Γ et de matrice de relation C. Nous appelons loi normale circulaire symétrique centrée une loi normale complexe de moyenne nulle $(\mu = 0)$ et de matrice de relation nulle $(C = 0)$. Une loi normale circulaire symétrique centrée est donc caractérisée par sa matrice de covariance Γ qui est hermitienne définie positive. Lorsque la série temporelle à valeurs complexes est de longueur n, il est équivalent de supposer que la série temporelle est une réalisation d'un processus stochastique autorégressif gaussien stationnaire centré et de supposer que le vecteur $\mathbf{u} = [u(0), u(1), ..., u(n-1)]^T$ est la réalisation d'une loi normale circulaire symétrique centrée dont la matrice de covariance $\mathbf{R}$ est Toeplitz hermitienne définie positive.

L'algorithme de Burg 17 permet d'estimer les paramètres $(r_0, \mu_1, \dots, \mu_{n-1})$ à partir des séries temporelles observées. Le principe de l'agorithme de Burg est détaillé dans l'appendice C.

Dans le cas des séries temporelles multidimensionnelles, le modèle autorégressif est caractérisé par l'équation suivante :

$$U(k) + \sum_{j=1}^{n-1} A_i^{n-1} U(k-j) = W(k) \tag{3}$$

où W est le vecteur d'erreur de prédiction de taille N de matrice de covariance Σ. La matrice de covariance Σ et les coefficients de prédiction A_j^{n-1} sont des matrices carrées de taille $N \times N$.

Ce modèle peut donc être paramétré par la matrice de covariance Σ et les coefficients du modèle autorégressif $\left(A_1^{N-1}, \dots, A_{N-1}^{N-1}\right)$. Nous définissons les coefficients d'autocorrélation $R_i := \mathbb{E}\left[U(k+i)\, U(k)^H\right]$, où $\cdot^H$ désigne l'opérateur de transconjugaison d'une matrice complexe. La série temporelle étudiée peut également être représentée par le coefficient d'autocorrélation R_0 et les coefficients de réflexion $\left(A_1^1, \dots, A_{n-1}^{n-1}\right)$. Pour deux matrices complexes A et B, nous notons $A > B$ lorsque la matrice $A - B$ est hermitienne définie positive. Nous montrons que les coefficients de réflexion A_i^i appartiennent à l'espace $\mathcal{D}_N := \left\{M \in \mathbb{C}^{N\times N} \mid I_N - M\overline{M} > 0\right\}$, où $\overline{M} := JM^*J$ avec J la matrice anti-diagonale [45]. Il est également possible de transformer les coefficients de réflexion $A_i^i \in \mathcal{D}$ en coefficients $M_i \in \mathcal{SD}_N$ où $\mathcal{SD}_N := \left\{M \in \mathbb{C}^{N\times N} \mid I_N - MM^H > 0\right\}$ est appelé le disque de Siegel. Nous pouvons donc représenter la série temporelle multidimensionnelle étudiée par le coefficient d'autocorrélation R_0 et les coefficients $(M_1, \dots, M_{n-1})$ appartenant au disque de Siegel. En utilisant cette représentation, la série temporelle étudiée est donc représentée par les coordonnées $(R_0, M_1, \dots, M_{n-1})$ dans l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. Il est également possible de représenter la série temporelle étudiée par les coefficients d'autocorrélation $(R_0, \dots, R_{n-1})$. Ces coefficients caractérisent la matrice de covariance $\mathbf{R}$ des données vectorisées $\mathbf{U}_n = \left[U(0)^T, \dots, U(n-1)^T\right]^T$. La matrice de covariance de $\mathbf{U}_n$ est la matrice Toeplitz par blocs hermitienne définie positive $\mathbf{R}$ définie par :

$$\mathbf{R} = \begin{bmatrix} R_0 & R_1^H & R_2^H & \dots & R_{n-1}^H \\ R_1 & R_0 & R_1^H & \dots & R_{n-2}^H \\ R_2 & R_1 & R_0 & \dots & R_{n-3}^H \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{n-1} & R_{n-2} & R_{n-3} & \dots & R_0 \end{bmatrix}. \tag{4}$$

Lorsque la série temporelle multidimensionnelle à valeurs complexes est de longueur n, il est équivalent de supposer que la série temporelle est la réalisation d'un processus stochastique suivant une modèle autorégressif gaussien stationnaire centré et de supposer que le vecteur $\mathbf{U}_n$ est la réalisation d'une loi normale complexe circulaire symétrique centrée dont la matrice de covariance $\mathbf{R}$ est Toeplitz par blocs hermitienne définie positive. L'algorithme 5 présenté dans la section 2.2.9 permet d'estimer les coefficients $(R_0, M_1, \dots, M_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ à partir de séries temporelles multidimensionnelles observées.

III.2 Résumé du chapitre 3 : Théorie de la géométrie de l'information

Dans le chapitre 3, nous munissons les espaces de représentation des séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré d'une métrique riemannienne inspirée de la géométrie de l'information. La théorie de la géométrie de l'information donne un cadre théorique qui permet de munir des espaces de lois paramétriques

d'une métrique riemannienne. Comme vu dans le chapitre 2, les séries temporelles étudiées peuvent être considérées comme des réalisations de lois gaussiennes complexes circulaires symétriques centrées : $\mathbf{u} \sim \mathcal{CN}(0, \mathbf{R}, O)$. Dans le cas unidimensionnel, la matrice $\mathbf{R}$ est Toeplitz hermitienne définie positive. Dans le cas multidimensionnel, la matrice $\mathbf{R}$ est Toeplitz par blocs hermitienne définie positive. Dans le cas plus général où la matrice de covariance $\mathbf{R}$ est simplement hermitienne définie positive, nous pouvons utiliser les outils présentés dans le livre de Shun-Ichi Amari [73] pour définir la métrique de la géométrie de l'information sur les lois gaussiennes circulaires symétriques centrées [15, 82]. Cette métrique sur les matrices hermitiennes définies positives $\mathcal{H}_N^+$ est donnée par l'élément : $ds^2 = \text{trace}\left(P^{-1}dPP^{-1}dP\right)$. D'après l'étude des différents espaces de représentation des séries temporelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré effectué au chapitre 2, la matrice de covariance Toeplitz $\mathbf{R}$ d'une série temporelle unidimensionnelle stationnaire peut être représentée par les coefficients $(p_0, \mu_1, \ldots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$. En nous inspirant de la construction de la métrique de la géométrie de l'information sur l'espace $\mathcal{H}_N^+$, nous construisons une métrique riemannienne sur l'espace $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$. Nous notons $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ la variété riemannienne obtenue. La métrique construite est une métrique produit définie par :

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n\frac{dp_0^2}{p_0^2} + \sum_{k=1}^{n-1}(n-k)\frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2}. \tag{5}$$

Pour les séries temporelles multidimensionnelles, la matrice de covariance Toeplitz par blocs $\mathbf{R}$ peut être représentée par les coefficients $(R_0, M_1, \ldots, M_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. Nous construisons une métrique sur l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ en nous inspirant de la construction de la métrique de la géométrie de l'information sur $\mathcal{H}_N^+$. Nous notons $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ la variété riemannienne obtenue. La métrique construite est une métrique produit définie par :

$$\begin{aligned} ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} =& n\,\text{trace}\left(P_0^{-1}\,dP_0\,P_0^{-1}\,dP_0\right) \\ &+ \sum_{l=1}^{n-1}(n-l)\,\text{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1}d\Omega_l\left(I-\Omega_l^H\Omega_l\right)^{-1}d\Omega_l^H\right). \end{aligned} \tag{6}$$

III.3 Résumé du chapitre 4 : Variétés riemanniennes associées aux séries temporelles stationnaires à valeurs complexes

Dans le chapitre 4, nous présentons certains outils géométriques utiles des variétés riemanniennes $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ et $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ obtenues dans le chapitre 3. Pour ces deux variétés, les outils géométriques présentés sont : l'expression de l'élément ds^2, l'expression de la distance entre deux points, le produit scalaire, la norme, la donnée d'une isométrie, l'application logarithme riemannien, l'application exponentielle riemannienne, l'équation des géodésiques et la courbure sectionnelle. Pour étudier ces outils géométriques, nous remarquons que ces variétés sont des variétés produits. En effet, la métrique de la variété $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ peut s'écrire :

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = nds^2_{\mathbb{R}^{++}} + \sum_{k=1}^{n-1}(n-k)\,ds^2_{\mathbb{D}} \tag{7}$$

avec $ds^2_{\mathbb{R}^{++}} = \frac{dp_0^2}{p_0^2}$ et $ds^2_{\mathbb{D}} = \frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2}$.

Nous notons $\mathbb{R}^{++}$ la variété riemannienne définie sur l'espace $\mathbb{R}_+^*$ par la métrique $ds^2_{\mathbb{R}^{++}}$. Nous appelons disque de Poincaré et notons $\mathbb{D}$ la variété définie sur le disque unité complexe $\mathcal{D}$ par la métrique $ds^2_{\mathbb{D}}$. Nous commençons par étudier en détails les variétés $\mathbb{R}^{++}$ et $\mathbb{D}$, puis nous utilisons la relation définie par l'équation 7 pour en déduire les outils géométriques de la variété $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$.

La métrique de la variété riemannienne $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ peut également s'écrire comme une métrique produit :

$$ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} = nds^2_{\mathbb{H}_N^{++}} + \sum_{l=1}^{n-1}(n-l)\,ds^2_{\mathbb{SD}_N} \tag{8}$$

avec

$$\begin{aligned} ds^2_{\mathbb{H}_N^{++}} &= \text{trace}\left(P_0^{-1}\,dP_0\,P_0^{-1}\,dP_0\right) \\ ds^2_{\mathbb{SD}_N} &= \text{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1}d\Omega_l\left(I-\Omega_l^H\Omega_l\right)^{-1}d\Omega_l^H\right). \end{aligned}$$

Nous notons $\mathbb{H}_N^{++}$ la variété définie sur l'espace des matrices hermitiennes définies positives $\mathcal{H}_N^+$ par la métrique $ds^2_{\mathbb{H}_N^{++}}$. Nous notons $\mathbb{SD}_N$ la variété définie sur l'espace de Siegel $\mathcal{SD}_N$ par la métrique $ds^2_{\mathbb{SD}_N}$. Nous étudions d'abord

les variétés $\mathbb{H}_N^{++}$ et $\mathbb{SD}_N$, puis nous utilisons la relation donnée dans l'équation 8 pour en déduire les outils géométriques de la variété riemannienne $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. L'élément de métrique $ds^2_{\mathbb{SD}_N}$, la distance et le produit scalaire de la variété $\mathbb{SD}_N$ sont donnés dans l'article de Ben Jeuris et Raf Vandebril [45]. A la connaissance des auteurs, l'expression explicite de l'application logarithme riemannien, de l'application exponentielle riemannienne et de la courbure sectionnelle de l'espace de Siegel n'étaient pas connues, la donnée de leurs expressions est donc une contribution majeure de cette thèse.

III.4 Résumé du chapitre 5 : Machine Learning sur des variétés riemanniennes

Dans le chapitre 5, nous détaillons des algorithmes de machine learning adaptés à la structure de variété riemannienne. La plupart des algorithmes de machine learning classiques considèrent implicitement que les données à classifier sont des données euclidiennes. Dans un espace euclidien, il est possible d'additionner deux éléments, de multiplier un élément par un scalaire et de calculer très facilement la moyenne x_{mean} des éléments $(x_1, \ldots, x_N)$: $x_{mean} = \frac{1}{N}\sum_{i=1}^{N} x_i$. La distance entre deux points $a = (a_1, \ldots, a_n)$ et $b = (b_1, \ldots, b_n)$ est définie par : $d(a,b) = \left(\sum_{i=1}^{n} (b_i - a_i)^2\right)^{1/2}$. Cependant, ces opérations (entre autres) ne sont pas définies dans des variétés riemanniennes en général. Pour qu'un algorithme de machine learning puisse fonctionner dans une variété riemannienne en respectant sa structure, il faut donc qu'il repose sur des opérations qui soient définies dans des variétés riemanniennes. Pour les variétés $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ et $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$, les opérations qui reposent sur les outils géométriques présentés dans le chapitre 4 seront bien définies. Certains algorithmes de machine learning euclidiens classiques utilisent comme seul outil géométrique la notion de distance pour classifier des données. Pour adapter ces algorithmes à des données appartenant à une variété riemannienne non euclidienne, il suffit de remplacer la distance euclidienne par l'expression de la distance dans la variété riemannienne. Parmi les algorithmes de classification supervisée utilisant uniquement la notion de distance, nous présentons l'algorithme des k plus proches voisins dans la section 5.3.1 et l'algorithme Kernel Density Estimation Classifier dans la section 5.3.2. Parmi les algorithmes de clustering utilisant uniquement la notion de distance, nous présentons l'algorithme de la classification ascendante hiérarchique dans la section 5.4.1.

D'autres algorithmes de machine learning euclidiens classiques utilisent la notion de moyenne. Pour pouvoir utiliser ces algorithmes dans une variété riemannienne non euclidienne, nous devons définir la notion de moyenne dans une variété riemannienne. Dans la section 5.1.2, nous définissons la *moyenne* des points $(x_1, \ldots, x_N)$ comme l'ensemble des minima locaux de la fonction :

$$H_2(x) = \sum_{i=1}^{N} d^2(x, x_i). \tag{9}$$

Une variété de *Hadamard* est une variété riemannienne complète, simplement connexe et de courbure sectionnelle négative ou nulle. Lorsque la variété étudiée est une variété de Hadamard, la fonction $H_2(x)$ possède un unique minimum local (et donc global) [82], [6]. Nous montrons dans le chapitre 4 que les variétés $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ et $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ sont des variétés de Hadamard. Pour approximer la valeur de la moyenne dans une variété de Hadamard, nous pouvons faire une descente de gradient sur la fonction H_2. Le gradient de cette fonction a pour expression :

$$G_2(x) = -2\sum_{i=1}^{N} \log_x(x_i). \tag{10}$$

Le calcul du gradient fait donc intervenir l'application logarithme riemannien. Pour nous déplacer sur la variété riemannienne dans la direction de l'opposé du gradient, nous utilisons ensuite l'application exponentielle riemannienne. L'algorithme d'approximation de la moyenne utilisant la descente de gradient est détaillé dans l'algorithme 8. Il existe également une méthode stochastique permettant d'approximer la moyenne [6], cette méthode est présentée dans l'algorithme 9. Dans le chapitre 5, nous présentons deux algorithmes utilisant un calcul de moyenne pour classifier des données : l'algorithme supervisé nommé Nearest Centroid Classifier présenté dans la section 5.3.3 et l'algorithme de clustering des k-means présenté dans la section 5.4.3. Ces deux algorithmes utilisent trois outils géométriques : la distance, l'application logarithme riemannien et l'application exponentielle riemannienne. Pour chaque algorithme présenté dans le chapitre 5, nous préciserons les outils géométriques sur lesquels il repose et la complexité de l'algorithme par rapport au nombre de données d'entrée.

III.5 Résumé du chapitre 6 : Application à la classification de séries temporelles unidimensionnelles simulées suivant un modèle autorégressif gaussien stationnaire centré

Dans le chapitre 6, nous appliquons certains algorithmes de machine learning présentés dans le chapitre 5 à la classification de séries temporelles unidimensionnelles simulées. Les séries temporelles simulées sont des séries temporelles unidimensionnelles à valeurs complexes qui suivent un modèle autorégressif gaussien stationnaire centré. Comme vu dans la section 2.1 du chapitre 2, ces séries temporelles peuvent être considérées comme des réalisations de lois circulaires symétriques centrées $\mathcal{CN}(0, \mathbf{R}, 0)$ dont la matrice de covariance $\mathbf{R}$ est Toeplitz. Nous pouvons donc simuler une

série temporelle Z de matrice de covariance $\mathbf{R}$ grâce à l'équation $Z = \mathbf{R}^{1/2}X$ où X est un vecteur aléatoire gaussien complexe standard. Nous utilisons cette propriété pour simuler des séries temporelles unidimensionnelles dans le chapitre 6. Nous simulons 200 séries temporelles de matrice de covariance Toeplitz $\mathbf{R}_0$ et 200 séries temporelles de matrice de covariance Toeplitz $\mathbf{R}_1$. Puis nous séparerons chacun de ses deux jeux de données en deux : 100 séries temporelles sont utilisées comme jeu de données d'entraînement et 100 séries temporelles sont utilisées comme jeu de données de test. Nous présentons plusieurs espaces de représentation des séries temporelles unidimensionnelles à valeurs complexes suivant un modèle autorégressif gaussien stationnaire centré. Dans chacun de ces espaces de représentation, nous utilisons d'abord l'algorithme de la Tangent PCA 10 pour visualiser les jeux de données simulés. Ensuite, nous utilisons l'algorithme supervisé du nearest centroid classifier présenté dans la section 5.3.3 pour classifier les séries temporelles simulées. Enfin, nous visualisons la Tangente PCA du résultat de la classification obtenue et la matrice de confusion correspondante.

III.6 Résumé du chapitre 7 : Application à la classification de séries temporelles multidimensionnelles simulées suivant un modèle autorégressif gaussien stationnaire centré

Dans le chapitre 7, nous appliquons certains algorithmes de machine learning présentés dans le chapitre 5 à la classification de séries temporelles multidimensionnelles simulées. Les séries temporelles simulées sont des séries temporelles multidimensionnelles à valeurs complexes qui suivent un modèle autorégressif gaussien stationnaire centré. Comme vu dans la section 2.2 du chapitre 2, ces séries temporelles peuvent être considérées comme des réalisations de lois circulaires symétriques centrées $\mathcal{CN}(0, \mathbf{R}, 0)$ dont la matrice de covariance $\mathbf{R}$ est Toeplitz par blocs. Nous pouvons donc simuler une série temporelle Z de matrice de covariance $\mathbf{R}$ grâce à l'équation $Z = \mathbf{R}^{1/2}X$ où X est un vecteur aléatoire gaussien complexe standard. Nous utilisons cette propriété ainsi que l'expression du modèle autorégressif associé pour simuler des séries temporelles multidimensionnelles à valeurs complexes. Nous simulons 200 séries temporelles de matrice de covariance Toeplitz par blocs $\mathbf{R}_0$ et 200 séries temporelles de matrice de covariance Toeplitz par blocs $\mathbf{R}_1$. Puis nous séparons chacun de ces deux jeux de données en deux : 100 séries temporelles sont utilisées comme jeu de données d'entraînement et 100 séries temporelles sont utilisées comme jeu de données de test. Nous présentons plusieurs espaces de représentation des séries temporelles multidimensionnelles à valeurs complexes qui suivent un modèle autorégressif gaussien stationnaire centré. Dans chacun de ces espaces de représentation, nous commençons par utiliser l'algorithme de la Tangent PCA 10 pour visualiser les jeux de données simulés. Ensuite, nous utilisons l'algorithme supervisé du nearest centroid classifier présenté dans la section 5.3.3 pour classifier ces données. Enfin, nous visualisons la Tangente PCA du résultat de la classification obtenue et la matrice de confusion correspondante.

III.7 Résumé du chapitre 8 : Application au clustering de fouillis radar

Dans le chapitre 8, nous présentons une application de la méthode de classification des séries temporelles suivant un modèle autorégressif gaussien stationnaire centré présentée dans cette thèse au clustering de séries temporelles radar. Les données radar étudiées viennent d'un radar Thales situé à Saint-Mandrier sur la côte méditerranéenne française et orienté vers la mer. Pour obtenir des informations sur son environnement, le radar étudié envoie des ondes électromagnétiques qui vont se réfléchir sur l'environnement puis être enregistrées par le radar. Il est possible de distinguer différentes directions de provenance des échos, nous nous concentrons ici sur le faisceau d'élévation nulle. Dans un même secteur angulaire de quelques degrés, le radar Thales étudié envoie une rafale, c'est à dire une série d'une dizaine d'impulsions identiques. Entre chaque impulsion il y a une phase d'écoute pendant laquelle les réflexions des ondes émises sont enregistrées. Lors de la phase d'écoute, le début du signal enregistré correspond à la réflexion du signal émis sur les éléments les plus proches du radar et la fin du signal enregistré correspond aux éléments les plus éloignés du radar. Chaque phase d'écoute est ensuite subdivisée temporellement et nous appelons case distance la zone spatiale observée par le radar lors d'une subdivision. A chaque case distance est associé un nombre complexe. Le module de ce nombre complexe représente le module du signal enregistré dans la subdivision temporelle associée à la case distance étudiée. La phase de ce nombre complexe représente la différence de phase entre le signal émis et le signal enregistré. Ce nombre complexe est obtenu grâce à une opération de convolution entre le signal émis et le signal enregistré, cette opération est nommée compression d'impulsion. Lors de chaque rafale, nous représentons les informations associées au secteur angulaire observé sous la forme d'une matrice U. Chaque ligne de la matrice U représente une même case distance et chaque colonne représente

une même impulsion. Nous noterons :

$$U = \begin{bmatrix} u_{0,0} & u_{0,1} & \dots & u_{0,n-1} \\ u_{1,0} & u_{1,1} & \dots & u_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-1,0} & u_{N-1,1} & \dots & u_{N-1,n-1} \end{bmatrix} \tag{11}$$

où N représente le nombre de cases distance (environ 900) et n représente le nombre d'impulsions (environ 10). Concernant le radar Thales étudié, les cases distances les plus proches du radar sont situées à environ 5 kilomètres du radar, les cases distances les plus éloignées sont situées à environ 60 kilomètres, chaque case distance ayant une longueur radiale de 60 mètres. Lors de l'étude du secteur angulaire suivant (le radar tourne dans le sens horaire), le radar envoie une nouvelle rafale dans la direction correspondante. L'information radar associée à ce nouveau secteur angulaire sera représentée par une nouvelle matrice. Le nombre d'impulsions d'une rafale et la durée d'écoute entre deux impulsions peut varier d'une rafale à une autre, deux rafales différentes seront donc associées à des matrices de tailles différentes en général. Nous cherchons maintenant à effectuer le clustering des cases distances autour du radar en fonction de la nature de l'environnement associé. Dans le langage radar, nous appelons cibles les objets d'intérêt premier tels les avions, les drônes, les missiles, les navires... Nous appelons fouillis radars les signaux liés à l'environnement tels les vagues, les champs, les forêts, les zones urbaines, la pluie, la grêle... Nous supposons dans notre étude que la grande majorité des cases distances à classifier correspondent à du fouillis radar (et non à des cibles). Nous supposons que les séries temporelles à valeurs complexes associées au fouillis radar suivent un modèle autorégressif gaussien stationnaire centré. Nous avons vu dans le chapitre 2 que ces séries temporelles peuvent être représentées par les coefficients $(p_0, \mu_1, \dots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathbb{D}^{n-1}$. La valeur de ces coefficients est alors associée à la nature du fouillis : le coefficient p_0 représente l'intensité de la série temporelle enregistrée et les coefficients $(\mu_1, \dots, \mu_{n-1})$ représentent la structure Doppler de la série temporelle. Nous choisissons d'effectuer le clustering des cases distances uniquement à partir des coefficients $(\mu_1, \dots, \mu_{n-1})$ représentant l'information Doppler. Pour chaque case distance, nous estimons ces coefficients grâce à l'algorithme de Burg présenté en détails dans l'appendice C. Nous utiliserons ensuite l'algorithme du k-means 12 pour effectuer le clustering des cases distances étudiées, chaque case distance étant représentée par les coefficients $(\mu_1, \dots, \mu_{n-1})$ appartenant à l'espace $\mathbb{D}^{n-1}$ muni de la métrique décrit dans la section 4.1.3 en omettant le terme de puissance p_0. Pour cette étude, nous choisissons $n = 8$, ce qui correspond au nombre d'impulsions de la rafale la plus courte. Nous allons visualiser le résultat du clustering dans le produit de disques unité complexes $\mathcal{D}^{n-1}$ et nous allons visualiser le résultat spatial du clustering sur une carte de terrain pour interpréter les résultats obtenus.

III.8 Résumé du chapitre 9 : Application à la classification de données audio stéréo

Enfin, dans le chapitre 9 nous utilisons plusieurs algorithmes de machine learning présentés dans le chapitre 5 pour visualiser et effectuer la classification supervisée de données audio stéréo. Les données classifiées correspondent à des bandes sonores de sons stationnaires téléchargées sur YouTube. Nous avons téléchargés trois types de sons stationnaires : des bruits de feux de bois, des bruits de cascades et des bruits de vents. Parmi les autres bruits stationnaires, il y a la mer lorsque le bruit des vagues est constant, la pluie intense, les douches, les ventilations, les sèches cheveux, les sèches linges, les climatisations, les bruits de moteurs en régime continu, le gaz qui s'échappe d'une gazinière... Chacun des trois jeux de données téléchargés (feux, cascades, vents) est ensuite divisé en un jeu de données d'entraînement et un jeu de données de test. Notre objectif est maintenant de classifier ces données. Une bande son stéréo est caractérisée par un scalaire qui représente la fréquence d'échantillonnage (48000 Hz ici) et une matrice réelle dont le nombre de lignes est égale au nombre de secondes d'enregistrement multiplié par la fréquence d'échantillonnage. Cette matrice a deux colonnes ce qui correspond au nombre de canaux d'enregistrement. Toutes les bandes sons téléchargées doivent avoir une même fréquence d'échantillonnage, nous vérifions également que les deux canaux d'enregistrement sont différents (sinon il s'agit d'un enregistrement mono). Nous supposons que les séries temporelles audio bidimensionnelles à valeurs réelles à classifier suivent un modèle autorégressif gaussien stationnaire centré. Nous nous plaçons ainsi dans le cadre des séries temporelles multidimensionnelles suivant un modèle autorégressif gaussien stationnaire centré présenté dans la section 2.2. Pour chaque enregistrement audio stéréo, nous estimons les coefficients $(R_0, M_1, \dots, M_{n-1})$ appartenant à l'espace $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ grâce à l'algorithme 5. Ici la dimension de la série temporelle est $N = 2$, nous choisissons $n = 4$ pour le nombre de coefficients ce qui correspond à un modèle autorégressif d'ordre trois. Nous munissons l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ de la métrique décrite dans la section 4.2.3. Nous remarquons que lorsque les séries temporelles d'entrée sont à valeurs réelles, les coefficients $(R_0, M_1, \dots, M_{n-1})$ estimés par l'algorithme 5 sont également des réels. Nous pouvons remarquer que tout les outils géométriques de la section 4.2.3 ne font intervenir que des éléments réels lorsque les matrices en entrée sont réelles : la sous-variété de $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ constituée de produits de matrices réelles est donc

une sous-variété totalement géodésique de $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. Les outils géométriques présentés dans la section 4.2.3 sont donc parfaitement adaptés au cas particulier des produits de matrices réelles de la variété $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. Les résultats de classification de notre méthode sur le jeu de données audio constitué à partir de vidéos YouTube sont présentés dans la section 9.2 en utilisant l'algorithme du nearest centroid classifier présenté dans la section 5.3.

IV Principales contributions

Nous résumons dans cette section les contributions techniques apportées par cette thèse.

IV.1 Coefficients de réflexion théoriques des processus stochastiques continus stationnaires ayant un spectre de forme gaussienne

Dans l'appendice B, la fonction d'autocorrélation R_f d'un processus stochastique continu f à valeurs complexes est définie par :

$$R_f(\tau) = \int_{-\infty}^{+\infty} f(t+\tau) f^*(t) dt. \tag{12}$$

Cette définition est liée à la définition des coefficients d'autocorrélation d'un processus stochastique discret stationnaire $\mathbf{u}$ donnée dans la section 2.1.2 :

$$r_k = r(k) = \mathbb{E}\left[\mathbf{u}(n)\mathbf{u}^*(n-k)\right] = \mathbb{E}\left[\mathbf{u}(n+k)\mathbf{u}^*(n)\right] \quad \forall k \in \mathbb{Z}. \tag{13}$$

Ces deux définitions sont liées par la relation :

$$r_k = R_f(kT). \tag{14}$$

Pour simplifier les notations, nous normalisons l'échelle temporelle et considérons la période d'échantillonnage $T = 1$ dans cette thèse.

La fonction d'autocorrélation R_f du processus stochastique continu stationnaire f a une transformée de Fourier appelée densité spectrale de puissance :

$$S_f(\xi) = \int_{-\infty}^{+\infty} R_f(\tau) e^{-i2\pi\xi\tau} d\tau. \tag{15}$$

La fonction d'autocorrélation $R_f(\tau)$ est donc la transformée de Fourier inverse de la densité spectrale de puissance $S_f(\xi)$:

$$R_f(\tau) = \int_{-\infty}^{+\infty} S_f(\xi) e^{+i2\pi\xi\tau} d\xi. \tag{16}$$

Nous prouvons dans l'appendice B que le processus stochastique continu à valeurs complexes stationnaire f dont la densité spectrale de puissance S_f a la forme d'une distribution gaussienne de moyenne m et de variance σ^2 avec un coefficient de puissance P, *i.e.*:

$$S_f(\xi) = P \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\xi-m)^2}{2\sigma^2}} \tag{17}$$

a les coefficients d'autocorrélation théoriques suivants :

$$r_k = R_f(k) = P e^{i2\pi mk} e^{-2\pi^2\sigma^2 k^2} \quad \forall k \geqslant 1. \tag{18}$$

Ces coefficients peuvent être obtenus en utilisant la transformée de Fourier inverse donnée dans l'équation (16). Grâce à l'algorithme de Levinson 13, nous pouvons alors calculer explicitement l'expression des coefficients de réflexion $(\mu_1, \ldots, \mu_n)$ à partir des coefficients d'autocorrélation $(r_0, r_1, \ldots, r_n)$. Nous obtenons :

$$\mu_k = (-1)^k e^{i2\pi mk} e^{-2\pi^2\sigma^2 k} \quad \forall k \geqslant 1. \tag{19}$$

Remarquons que dans ce cas particulier, nous avons $\mu_k = \mu_1^k$. Nous remarquons également que l'argument du coefficient μ_k dépend seulement de la moyenne m et que son module dépend seulement de la variance σ^2.

IV.2 Les outils géométriques de la variété de Siegel

Nous montrons dans la section 2.2.8 que les séries temporelles multidimensionnelles à valeurs complexes qui suivent un modèle autorégressif gaussien stationnaire centré peuvent être représentées dans l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ où $\mathcal{H}_N^+$ représente l'ensemble des matrices hermitiennes définies positives et $\mathcal{SD}_N$ représente le disque de Siegel défini par l'ensemble des matrices complexes dont les valeurs singulières sont strictement inférieures à 1 :

$$\mathcal{SD}_N = \left\{ M \in \mathbb{C}^{N \times N} \mid I - MM^H > 0 \right\}. \tag{20}$$

Dans la section 3.2.3 du chapitre 3, nous munissons l'espace $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ d'une métrique produit inspirée de la géométrie de l'information. Nous notons $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ la variété riemannienne produit obtenue et $ds^2_{\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}}$ sa métrique qui a pour expression :

$$\begin{aligned} ds^2_{\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}} =& n \operatorname{trace}\left(P_0^{-1}\, dP_0\, P_0^{-1}\, dP_0\right) \\ &+ \sum_{l=1}^{n-1} (n-l) \operatorname{trace}\left(\left(I - \Omega_l \Omega_l^H\right)^{-1} d\Omega_l \left(I - \Omega_l^H \Omega_l\right)^{-1} d\Omega_l^H\right). \end{aligned} \tag{21}$$

Nous notons $\mathbb{H}_N^{++}$ la variété riemannienne construite en munissant l'espace $\mathcal{H}_N^+$ de la métrique :

$$ds^2_{\mathbb{H}_N^{++}} = \operatorname{trace}\left(P_0^{-1}\, dP_0\, P_0^{-1}\, dP_0\right). \tag{22}$$

Nous notons $\mathbb{SD}_N$ la variété obtenue en munissant l'espace $\mathcal{SD}_N$ de la métrique :

$$ds^2_{\mathbb{SD}_N} = \operatorname{trace}\left(\left(I - \Omega_l \Omega_l^H\right)^{-1} d\Omega_l \left(I - \Omega_l^H \Omega_l\right)^{-1} d\Omega_l^H\right). \tag{23}$$

Nous avons alors :

$$ds^2_{\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}} = n ds^2_{\mathbb{H}_N^{++}} + \sum_{l=1}^{n-1} (n-l)\, ds^2_{\mathbb{SD}_N}. \tag{24}$$

Pour étudier les outils géométriques de la variété riemannienne produit $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$, nous commençons donc par étudier les variétés $\mathbb{H}_N^{++}$ et $\mathbb{SD}_N$. La variété riemannienne $\mathbb{H}_N^{++}$ est une variété déjà connue dont nous rappelons certains outils géométriques dans la section 4.2.1. Cependant, les outils géométriques de la variété de Siegel $\mathbb{SD}_N$ sont moins bien connus. Dans l'article de Ben Jeuris et Raf Vandebril [45], l'application distance et l'expression du produit scalaire sont donnés. Le carré de la distance entre deux points $\Omega, \Psi \in \mathcal{SD}_N$ a pour valeur :

$$\begin{aligned} d^2_{\mathbb{SD}_N}(\Omega, \Psi) =& \frac{1}{4} \operatorname{trace}\left(\log^2\left(\frac{I + C^{1/2}}{I - C^{1/2}}\right)\right) && (25) \\ =& \operatorname{trace}\left(\operatorname{arctanh}^2\left(C^{1/2}\right)\right) && (26) \end{aligned}$$

avec $C = (\Psi - \Omega)\left(I - \Omega^H \Psi\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Omega \Psi^H\right)^{-1}$.
L'expression du produit scalaire est, $\forall \Omega \in \mathcal{SD}_N, \forall v, w \in \mathbb{C}^{N \times N}$:

$$\begin{aligned} \langle v, w \rangle_\Omega =& \frac{1}{2} \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} v \left(I - \Omega^H \Omega\right)^{-1} w^H\right) \\ &+ \frac{1}{2} \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} w \left(I - \Omega^H \Omega\right)^{-1} v^H\right). \end{aligned} \tag{27}$$

D'après l'article de Ben Jeuris et Raf Vandebril [45], l'application suivante :

$$\Phi_\Omega(\Psi) = \left(I - \Omega\Omega^H\right)^{-1/2} (\Psi - \Omega)\left(I - \Omega^H \Psi\right)^{-1}\left(I - \Omega^H \Omega\right)^{1/2} \tag{28}$$

est dit être une isométrie sur la variété de Siegel. Nous prouvons cette propriété dans l'appendice E. Nous prouvons également que la différentielle de l'isométrie Φ a l'expression suivante :

$$D\Phi_\Omega(\Psi)[h] = \left(I - \Omega\Omega^H\right)^{1/2}\left(I - \Psi\Omega^H\right)^{-1} h \left(I - \Omega^H \Psi\right)^{-1}\left(I - \Omega^H \Omega\right)^{1/2}. \tag{29}$$

Nous prouvons que l'inverse de la fonction Φ décrite dans l'équation (28) a la propriété suivante :

$$\Phi_\Omega^{-1}(\Psi) = \Phi_{-\Omega}(\Psi). \tag{30}$$

Cette propriété de l'isométrie Φ est simplement mentionnée (sans preuve) dans l'article de Ben Jeuris et Raf Vandebril [45].

La contribution théorique majeure de cette thèse est l'expression explicite de l'application logarithme riemannien, de l'application exponentielle riemannienne et de la courbure sectionnelle de la variété de Siegel.

L'application logarithme riemannien a pour expression :

$$\log_{\Omega}(\Psi) = \left(I - \Omega\Omega^H\right)^{1/2} V' \left(I - \Omega^H\Omega\right)^{1/2} \tag{31}$$
$$\left[\begin{array}{l} V' = \operatorname{arctanh}(X)\, X^{-1}\Psi' \quad \text{où } X = \left(\Psi'\Psi'^{\,H}\right)^{1/2} \\ \Psi' = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \end{array}\right.$$

L'application exponentielle riemannienne a pour expression :

$$\exp_{\Omega}(V) = \left(I - \Omega\Omega^H\right)^{-1/2}\left(\Psi' + \Omega\right)\left(I + \Omega^H\Psi'\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{32}$$
$$\left[\begin{array}{l} \Psi' = \tanh(Y)\, Y^{-1}V' \quad \text{où } Y = \left(V'V'^{\,H}\right)^{1/2} \\ V' = \left(I - \Omega\Omega^H\right)^{-1/2} V \left(I - \Omega^H\Omega\right)^{-1/2} \end{array}\right.$$

La courbure sectionnelle en zéro du plan σ défini par les matrices orthonormées E_1 et E_2 a l'expression suivante :

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1E_2^H - E_2E_1^H\right\|^2 + \left\|E_1^HE_2 - E_2^HE_1\right\|^2\right). \tag{33}$$

La valeur de la courbure sectionnelle en un point quelconque peut ensuite être obtenue en utilisant l'isométrie Φ décrite dans l'équation (28).

Pour plus de détails, les outils géométriques de l'espace de Siegel présentés dans cette thèse sont résumés dans la section 4.2.2. L'appendice E est consacré à l'étude du disque de Siegel et à la démonstration des outils mathématiques nouveaux apportés dans cette thèse.

La connaissance des applications logarithmes et exponentielles riemanniens de la variété $\mathbb{SD}_N$ nous permet de construire les applications logarithmes et exponentielles riemanniens de la variété riemannienne produit $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$, ce que nous faisons dans la section 4.2.3. La connaissance de ces applications nous permet en autres d'utiliser les algorithmes du chapitre 5 faisant intervenir un calcul de moyenne lors de la classification de séries temporelles multidimensionnelles stationnaires représentées dans $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$.

IV.3 Triangles rectangles infinitésimaux et courbure sectionnelle dans les variétés riemanniennes

Dans la section D.4, nous prouvons la relation suivante entre la longueur de l'hypoténuse d'un triangle rectangle infinitésimal et sa courbure sectionnelle:

Theorem 1. *Soit U un ouvert convexe relativement compact d'une variété Riemannienne complète M. Il existe une constante $C_U > 0$ telle que pour tout triangle OAB de U rectangle en 0, nous ayons l'inégalité suivante:*

$$\left|d^2(A,B) - d^2(O,A) - d^2(O,B) + \frac{1}{3}K(\sigma)\, d^2(O,A)\, d^2(O,B)\right| \leqslant C_U\left(d(O,A) + d(O,B)\right)^5. \tag{34}$$

Dans les travaux de Andrei A. Agrachev, Davide Barilari et Luca Rizzi [4], ce théorème est énoncé dans le cas particulier où le point O est fixé et les directions des points A et B en partant de O sont fixées. Le résultat énoncé dans leurs travaux est dit être dû à Loeper et Villani. Le théorème 1 étudié ici est énoncé et prouvé indépendamment de ces précédents travaux, dans un cadre plus général.

IV.4 Classification de séries temporelles audio stationnaires dans $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

Nous appliquons le modèle multidimensionnel à la classification de séries temporelles audio stéréo stationnaires dans le chapitre 9. A notre connaissance, c'est la première fois que la variété $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ est utilisée pour classifier des séries temporelles audio stéréo stationnaires.

IV.5 Le modèle spatio-temporel radar

Dans l'appendice G et dans l'article [83], nous proposons un modèle spatio-temporel pour représenter le fouillis radar. Dans ce modèle, nous supposons que les séries temporelles enregistrées par le radar pendant une rafale sont des séries temporelles stationnaires au sens large sur l'axe temporel comme sur l'axe spatial, l'axe spatial considéré étant l'axe radial.

En effet, nous supposons que la matrice d'observation Z du faisceau d'élévation nulle enregistrée pendant une rafale pour un groupe de cases distance consécutives a la corrélation spatio-temporelle suivante:

$$Z = p_0^{1/2} R_s^{1/2} N R_t^{T\,1/2} \tag{35}$$

avec :

- Z: la matrice des observations radar de dimension $(p,\ n)$, où p correspond au nombre de cases distance et n correspond au nombre d'impulsions émises pendant la rafale.
- p_0: un nombre réel strictement positif qui correspond à l'espérance de la puissance quadratique, i.e. $p_0 = \mathbb{E}\left[|z_{i,j}|^2\right]$ pour tout $(i, j) \in [\![1, p]\!] \times [\![1, n]\!]$.
- R_s: la matrice de corrélation spatiale normalisée de dimension (p, p). C'est une matrice Toeplitz hermitienne définie positive puisque le signal est supposé être stationnaire au sens large sur l'axe spatial. Elle est dite *normalisée* dans le sens où ses coefficients diagonaux sont égaux à 1.
- R_t: la matrice de corrélation temporelle normalisée de dimension $(n,\ n)$. C'est une matrice Toeplitz hermitienne définie positive puisque le signal est supposé être stationnaire au sens large sur l'axe temporel.
- N: matrice de dimension $(p,\ n)$ dont les coefficients sont des variables aléatoires Gaussiennes complexes centrées réduites indépendantes.

Ce modèle ressemble au modèle présenté par Romain Couillet et al. [24], sauf pour la structure de la matrice de covariance spatiale R_s qui est seulement supposée Toeplitz ici.

Nous présentons également un modèle vectoriel équivalent :

$$\widetilde{Z} = p_0^{1/2} R_{st}^{1/2} \widetilde{N} \tag{36}$$

où $\widetilde{Z}$ est la vectorisation de la matrice Z et $\widetilde{N}$ est la vectorisation de la matrice N. La matrice de corrélation spatio-temporelle normalisée R_{st} est alors le produit de Kronecker de la matrice de corrélation spatiale normalisée R_s et de la matrice de corrélation temporelle normalisée R_t :

$$R_{st} = R_s \otimes R_t. \tag{37}$$

En utilisant la structure spécifique de la matrice de covariance $p_0 R_{st}$, nous construisons ensuite une variété riemannienne inspirée de la géométrie de l'information pour représenter les informations contenues dans un groupe de cases distance spatialement proches.

IV.6 Codes open source Python sur geomstats

Certains des codes utilisés dans cette thèse sont mis en ligne dans le package public geomstats [52]. Geomstats est un package Python de machine learning dans les variétés riemanniennes. Ce package met à disposition des variétés riemanniennes et des algorithmes de machine learning adaptés à des variétés riemanniennes. Cette librairie donne un cadre pour que chacun des algorithmes de machine learning mis à disposition soit compatible avec chacune des variétés riemanniennes proposées. Un utilisateur travaillant sur une variété riemannienne différente de celles disponibles sur geomstats pourra tout de même utiliser tous les algorithmes de machine learning présents sur geomstats à condition de définir la variété utilisée en respectant les conventions de geomstats. Dans cette thèse, nous avons utilisé les algorithmes de machine learning disponibles sur geomstats pour classifier des séries temporelles stationnaires unidimensionnelles dans la variété $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ et des séries temporelles stationnaires multidimensionnelles dans la variété $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. Le produit de disques de Poincaré $\mathbb{D}^{n-1}$ utilisé dans le chapitre 8 au clustering de fouillis radar a été mis en ligne sur la librairie geomstats pendant ce travail de thèse. L'algorithme de clustering de la classification ascendante hiérarchique présenté dans la section 5.4.1, l'algorithme de classification supervisée des k plus proches voisins présenté dans la section 5.3.1 et l'algorithme de classification supervisée nommé Kernel Density Estimation Classifier ainsi que plusieurs fonctions noyaux usuelles présentés dans la section 5.3.2 ont également été mis en ligne sur geomstats pendant ce travail de thèse. A terme, nous souhaiterions que la plupart des codes Python implémentés pour ce travail de thèse soient mis en ligne sur geomstats.

V Perspectives

Dans cette section, nous suggérons deux possibilités de futurs développements de cette thèse. La première possibilité concerne la classification supervisée de fouillis radar dans la variété riemannienne $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ dont les outils géométriques sont présentés dans la section 4.1.3. La seconde, plus théorique, concerne la possibilité de retrouver les formules de la variété de Siegel présentées dans la section 4.2.2 et dans l'appendice E en considérant la variété de Siegel comme une variété riemannienne homogène.

V.1 Classification supervisée de fouillis radar

Les données radar utilisées dans cette thèse ont été fournies sans labels. Dans le chapitre 8, nous effectuons la classification non-supervisée de fouillis radar. Afin d'effectuer la classification supervisée de données radar, nous les avons labellisées en utilisant la base de données Corine Land Cover. Il s'agit d'une base de données publique du gouvernement français qui partitionne le territoire français en polygones, chaque polygone étant associé à un label de terrain. Nous avons utilisé les labels suivants : surfaces artificielles, zones agricoles, zones forestières et semi-naturelles, marécages et enfin plans d'eau. Chacun de ces labels est divisé en sous-catégories que nous n'avons pas utilisées. Nous avons retrouvé le label de terrain de chaque case distance étudiée à partir de sa position GPS. Cependant, les performances de classification obtenues en utilisant ces labels n'étaient pas satisfaisantes. La méthode de classification proposée dans cette thèse classifie les cases distances à partir de l'information Doppler, or il semble que l'information Doppler ne permette pas de retrouver avec précision les labels de terrain. Cela est mis en évidence par les résultats du clustering de fouillis radar présentés dans la section 8.4 où l'on voit que les clusters ne sont pas liés aux labels de terrain : ils sont liés aux coefficients de réflexion et donc à l'information Doppler. L'information Doppler dépend beaucoup de la position du radar par rapport à la case distance étudiée. Pour la classification supervisée de fouillis radar, nous suggérons donc d'utiliser des labels se référant à l'information Doppler plutôt qu'au type de terrain.

V.2 L'étude de la variété de Siegel $\mathbb{SD}_N$ comme variété Riemannienne homogène

Une variété riemannienne $\mathcal{M}$ est dite *homogène* s'il existe un groupe de Lie G qui agit transitivement et de manière lisse par isométries sur $\mathcal{M}$. Une variété homogène $\mathcal{M}$ peut alors être vue comme le quotient d'un groupe de Lie G d'isométries par un sous-groupe K où K est le sous-groupe des isométries de G qui fixe un certain point $m \in \mathcal{M}$.

La contribution théorique majeure de cette thèse concerne l'étude des outils géométriques de la variété de Siegel $\mathbb{SD}_N$ présentée dans la section 4.2.2 et dans l'appendice E. D'après les travaux de Salem Said et al. [70,71], la variété de Siegel possède une structure de variété homogène qui n'a pas été exploitée ici. Nous pourrions donc essayer de retrouver les formules géométriques de la variété de Siegel présentées dans cette thèse en exploitant sa structure de variété homogène.

En dimension $N = 1$, la variété de Siegel $\mathbb{SD}_N$ correspond au disque de Poincaré $\mathbb{D}$ qui peut être considéré comme le quotient suivant [62] :

$$\mathbb{D} = \frac{SU(1,1)}{U(1)}$$

avec :

$$SU(1,1) = \left\{ g_{\alpha,\beta} = \begin{pmatrix} \alpha & \beta \\ \bar{\beta} & \bar{\alpha} \end{pmatrix}, |\alpha|^2 - |\beta|^2 = 1 \right\}$$
$$U(1) = \left\{ \begin{pmatrix} e^{i\theta} & 0 \\ 0 & e^{-i\theta} \end{pmatrix}, \theta \in \mathbb{R} \right\}$$

En effet, le groupe de Lie $SU(1,1)$ agit transitivement sur $\mathbb{D}$ par isométries :

$$SU(1,1) \times \mathbb{D} \to \mathbb{D}$$
$$\left(\begin{pmatrix} \alpha & \beta \\ \bar{\beta} & \bar{\alpha} \end{pmatrix}, z \right) \mapsto \frac{\alpha z + \beta}{\bar{\beta} z + \bar{\alpha}}$$

et les éléments de $U(1)$ fixent le point 0.

La structure quotient du disque de Poincaré a notamment été exploitée dans les travaux de Pierre-Yves Lagrave pour construire des réseaux de neurones équivariants [61–63].

Le disque de Poincaré $\mathbb{D}$ peut également être considéré comme le quotient suivant :

$$\mathbb{D} = \frac{U(1,1)}{U(1) \times U(1)}$$

avec

$$U(1,1) = \left\{ g \in M_2(\mathbb{C}), gPg^H = P \text{ où } P = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\}.$$

Ici, $.^H$ désigne l'opérateur de transconjugaison d'une matrice complexe, l'opérateur $.^*$ désigne la conjugaison complexe et $.^T$ désigne la transposition matricielle.

Plus généralement, la variété de Siegel $\mathbb{SD}_N$ de dimension N quelconque est également une variété homogène. La variété de Siegel peut en effet être considérée comme le quotient suivant :

$$\mathbb{SD}_N = \frac{U(N,N)}{U(N) \times U(N)}.$$

où $U(N)$ est le groupe unitaire de dimension N :

$$U(N) = \left\{ U \in M_N(\mathbb{C}), UU^H = I_N \right\}.$$

Plus précisément, si l'on pose P et S les matrices suivantes :

$$P = \begin{pmatrix} I_N & 0_N \\ 0_N & -I_N \end{pmatrix} \quad S = \begin{pmatrix} 0_N & I_N \\ -I_N & 0_N \end{pmatrix},$$

nous avons alors [71] :

$$U(N,N) = \left\{ g = \begin{pmatrix} A & B \\ C & D \end{pmatrix}, gPg^H = P \right\}$$

et

$$U(N) \times U(N) = \left\{ k = \begin{pmatrix} U & 0 \\ 0 & V \end{pmatrix}, (U,V) \in U(N)^2 \right\}.$$

Les groupes $U(N,N)$ et $U(N) \times U(N)$ agissent sur la variété de Siegel $\mathbb{SD}_N$ par transformations fractionnaires matricielles :

$$\begin{aligned} U(N,N) \times \mathbb{SD}_N &\to \mathbb{SD}_N \\ \left(\begin{pmatrix} A & B \\ C & D \end{pmatrix}, \Omega \right) &\mapsto (A\Omega + B)(C\Omega + D)^{-1}. \end{aligned}$$

Par ailleurs, notons que nous distinguons ici la variété de Siegel $\mathbb{SD}_N$ de la variété de Siegel *symétrique* (restreinte aux matrices symétriques, voir section E.6). La variété de Siegel symétrique peut être considérée comme le quotient $\frac{Sp(2N,\mathbb{R})}{U(N)}$ où $Sp(2N,\mathbb{R})$ est le groupe symplectique réel [70]. Plus précisément, nous avons [71] :

$$Sp(2N,\mathbb{R}) = \left\{ g = \begin{pmatrix} A & B \\ C & D \end{pmatrix}, gPg^H = P \text{ et } gSg^T = S \right\}$$

et

$$U(N) = \left\{ k = \begin{pmatrix} U & 0 \\ 0 & U^* \end{pmatrix}, UU^H = I_N \right\}.$$

Les algèbres de Lie des groupes de Lie mentionnés dans cette section sont décrites dans [71].

Abbreviations

AR	Autoregressive
BT	Block-Toeplitz
CLE	Continuous Lyapunov Equation
FFT	Fast Fourier Transform
HPD	Hermitian Positive Definite
TBBT	Toeplitz-Block Block-Toeplitz

List of Symbols

Manifolds

$\mathbb{H}_N^{++}$	HPD manifold , 67, 68
$\mathbb{R}^{++}$	Strictly positive real manifold vii, xx, 5, 55, 56, 58–60
$\mathbb{D}$	The Poincaré disk vii, xv, xx, 5, 12, 55, 57–60, 66, 202
$\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$	HPD and Siegel disks product manifold vii, viii, x–xiv, xix, xxi, 1, 4, 5, 7–11, 66–68, 77, 108, 109, 117, 118, 122, 126–128
$\mathbb{R}^{++} \times \mathbb{D}^{n-1}$	Strictly positive reals and Poincaré disks product manifold iv, v, vii, viii, xiv, xv, xx, xxi, 2, 4, 5, 11, 22, 55, 58–60, 77, 110, 111, 117, 118, 123, 124, 128, 220
$\mathbb{SD}_N$	Siegel manifold vii, viii, xii, xiii, xv, xvi, xix, 1, 5, 9, 10, 12, 13, 67, 68
$\mathcal{M}$	A general smooth Riemannian manifold

Mathematical notations

I, I_n	The identity matrix of unspecified size and of size n
J, J_n	The counter-identity matrix of unspecified size and of size n (ones on the anti-diagonal)

Matrix operations

X^*	The elementwise conjugate of X
X^H	The conjugate transpose of X
X^T	The transpose of X
$\bar{X}$	$:= JX^*J$
$\det$	The matrix determinant
$\mathfrak{L}$	The Continuous Lyapunov Equation Operator $X = \mathfrak{L}(A, Q) \iff AX + XA^H = Q$
$\|.\|_2$	The Frobenius norm
$\otimes$	The Kronecker product
trace	The matrix trace

Scalar and matrix spaces

$\mathbb{R}, \mathbb{C}$	The set of real or complex numbers
$\mathbb{R}^{m\times n}, \mathbb{C}^{m\times n}$	The set of $m \times n$ real or complex matrices
$\mathbb{R}^n, \mathbb{C}^n$	The set of real or complex vectors of length n
$\mathbb{R}_+^*$	The set of strictly positive real numbers
$\mathcal{B}_{n,N}$	The set of Block-Toeplitz matrices, consisting of n by n blocks, each block an $N \times N$ matrix

$\mathcal{B}^+_{n,N}$ The set of Block-Toeplitz Hermitian Positive Definite matrices, consisting of n by n blocks, each block an $N \times N$ matrix , xx, 55, 66, 101–103, 108, 122

$\mathcal{D}$ The complex unit disk , 22, 23, 34

$\mathcal{D}_N$ $:= \left\{X \in \mathbb{C}^{N\times N} | I_N - X\bar{X} > 0\right\}$

$\mathcal{H}_N$ The set of $N \times N$ Hermitian matrices , 60

$\mathcal{H}^+_N$ The set of $N \times N$ Hermitian Positive Definite matrices vii, xii, 4, 5, 9, 60, 66, 67

$\mathcal{SD}_N$ The Siegel disk
$:= \left\{X \in \mathbb{C}^{N\times N} | I_N - XX^H > 0\right\}$ v, vii, xii, 2, 5, 9, 62, 63, 66, 67, 171, 172, 175

$\mathcal{T}_{n,N}$ The set of Toeplitz-Block Block-Toeplitz matrices, consisting of n by n blocks, each block an $N \times N$ matrix

$\mathcal{T}^+_{n,N}$ The set of Toeplitz-Block Block-Toeplitz Hermitian Positive Definite matrices, consisting of n by n blocks, each block an $N \times N$ matrix

$\mathcal{T}_n$ The set of $n \times n$ Toeplitz matrices

$\mathcal{T}^+_n$ The set of $n \times n$ Toeplitz Hermitian Positive Definite matrices , 22, 58

Contents

Chapter 1

Introduction

The content of this introduction chapter have been previously written in French in the "Résumé général". In this thesis, we present a method to classify complex valued stationary centered Gaussian autoregressive time series. Our initial motivation comes from radar signal processing and especially radar clutter classification, which we detail in Section 1.1. This issue has already been addressed in previous works, in particular by Frédéric Barbaresco [9, 12, 34, 37, 38], Le Yang [6, 82], Alice Le Brigant [15] and Alexis Decurninge [3, 26]. We present the various articles and thesis manuscripts that were used as support for this dissertation in Section 1.2. We summarize each chapter of this dissertation in Section 1.3. In Section 1.4, we focus on the main contributions of this thesis. We propose a few perspectives for future developments of this thesis in Section 1.5.

Contents

1.1 Radar motivations

The initial motivation for studying complex valued stationary centered Gaussian autoregressive time series comes from radar signal processing, in particular the study of radar clutter. In radar semantics, we distinguish between moving objects of primary interest which we call targets and information related to the radar environment which we call clutter. Radar clutter is therefore the information recorded by a radar related to seas, forests, fields, cities and other environmental elements that surround the radar. In order to better distinguish targets from clutter, it may be interesting to develop

machine learning algorithms to recognize different types of clutter. Knowledge of radar clutter is used in Appendix F to obtain a constant false alarm rate (CFAR) detection estimator.

To study the characteristics of the complex valued time series associated with radar clutter, it is common to assume that they are stationary centered autoregressive Gaussian time series [13]. The assumption of stationarity of the time series is here justified by extremely short observation times of the same zone of the environment. The laws of these time series are represented in Riemannian manifolds in the works of Le Yang [6, 82] and Alice Le Brigant [15]. We use this representation model and apply it to radar clutter clustering in Chapter 8.

In order to refine the study of the characteristics of radar clutter, we wanted to add spatial information to the temporal information contained in each time series by studying the correlation between time series recorded in spatially close cells. For this, we have developed a spatio-temporal model which is studied in Appendix G. In order to provide the space of the coefficients of this spatio-temporal model with a Riemannian metric, we study the more general case of multidimensional stationary time series which thus becomes the main object of study of this thesis.

1.2 Related work

The study of one-dimensional stationary radar time series was carried out by Frédéric Barbaresco in [8–12, 39]. In these works, the stationary radar time series are represented in the product space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ where $\mathcal{D}$ represents the complex unit disk. This space is endowed with a Riemannian metric inspired by information geometry. The space $\mathbb{R}_+^*$ is used to represent the mean quadratic power of the studied time series. The product space $\mathcal{D}^{n-1}$ represents the coefficients of the autoregressive model, it therefore represents the Doppler information contained in the time series.

The Burg algorithm is used to estimate the coefficients of the autoregressive model from a recorded time series. This algorithm is presented in the works of Frédéric Barbaresco and Alexis Decurninge [3, 26].

The Riemannian metric constructed on the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ is presented by Frédéric Barbaresco in [37], [41], [35], [34], [33] and related works [23], [53], [29]. This metric is also detailed in the thesis work of Le Yang [6, 82] and in the thesis work of Alice Le Brigant [15]. We refer to Shun-ichi Amari's book [73] for a full presentation of the information geometry tools used to construct this metric. We denote $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ the Riemannian manifold presented in these works: the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ corresponds to the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with a Riemannian metric inspired by information geometry. This Riemannian metric is presented in Section 4.1.3. The computation of the mean and median in this manifold is used to detect radar targets in the work of Le Yang [6, 82]. The study of the curves of the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is applied to the recognition of radar targets in the work of Alice Le Brigant [15]. In the thesis work presented here, the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ will be used for radar clutter clustering and more generally to the classification of complex stationary centered Gaussian autoregressive time series.

One of the motivation of this thesis is the study of complex multidimensional stationary autoregressive centered Gaussian time series. As in the case of one-dimensional time series, it is possible to represent multidimensional time series by the coefficients of the autoregressive model. In the case of multidimensional time series, these autoregressive coefficients are square matrices. In the article written by Ben Jeuris and Raf Vandebril [45], the matrix coefficients of the autoregressive model are slightly modified to belong to the Siegel disk $\mathcal{SD}_N$ (set of complex matrices $N \times N$ of singular values strictly less than 1). The multidimensional stationary time series will then be represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$, where $\mathcal{H}_N^+$ is the space of Hermitian Positive Definite (HPD) matrices. This space will be endowed with a Riemannian product metric, the construction of which is detailed in the article written by Ben Jeuris and Raf Vandebril [45]. The product metric on the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ induces a Riemannian metric on the spaces $\mathcal{H}_N^+$ and $\mathcal{SD}_N$. The metric of the Siegel disk $\mathcal{SD}_N$ have been studied by Frédéric Barbaresco in [38], [36] and the related work [46]. To our knowledge, the Riemannian logarithm map, the Riemannian exponential map and the sectional curvature of the Riemannian manifold defined on the Siegel space $\mathcal{SD}_N$ were not known, these are contributions of this thesis which have been summarized in one of our previous articles [21]. These geometric tools are essential for the use of certain machine learning algorithms, in particular algorithms involving a computation of the mean as the k-means algorithm.

1.3 Main dissertation issues

This thesis aims to provide the reader with the vast majority of methodological and technical tools that we will use for the classification of stationary centered complex autoregressive Gaussian time series. Among the tools presented in this thesis, some were already known and some are new, we detail the new elements in Section 1.4. The structure of this thesis is linked to the methodology used. In Chapter 2 we present several representation spaces of the studied time series. In Chapter 3, we endow these spaces with a Riemannian metric inspired by information geometry. In Chapter 4, we detail the metric of the Riemannian manifolds obtained. In Chapter 5, we present machine learning algorithms adapted to the Riemannian manifold structure. In Chapters 6, 7, 8 et 9 we use some machine learning algorithms presented in Chapter 5 on the Riemannian manifolds presented in detail in Chapter 4 to classify stationary time series. We now detail the content of each chapter.

1.3.1 Summary of Chapter 2: Stationary centered complex Gaussian autoregressive time series

In Chapter 2, we present different parametric models associated with the complex stationary centered Gaussian autoregressive discrete-time stochastic processes.

We denote $n-1$ the order of the autoregressive stochastic processes presented in this section.

In the case of one-dimensional time series, this is written:

$$u(k) + \sum_{i=1}^{n-1} a_i^{n-1}\, u(k-i) = w^{n-1}(k) \tag{1.1}$$

where a_i^{n-1} are the prediction coefficients and the prediction errors $w^{n-1}(k)$ are independent identically distributed centered complex Gaussian processes of variance σ^2. Such a time series is therefore entirely parameterized by the coefficients of the autoregressive model $\left(a_1^{n-1}, \ldots, a_{n-1}^{n-1}\right)$ and by the coefficient σ^2. We define the autocorrelation coefficients $r_i := \mathbb{E}\left[u\,(k+i)\,u\,(k)^*\right]$. Equivalently, a time series can be represented by its variance r_0 and the coefficients of the autoregressive model $\left(a_1^1, \ldots, a_{n-1}^{n-1}\right)$. The coefficients a_i^i are called reflection coefficients and are denoted μ_i, they belong to the complex unit disk $\mathcal{D}$. Using this representation, the discrete-time stochastic process studied is therefore represented by the coordinates $(r_0, \mu_1, \ldots, \mu_{n-1})$ in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$. It is also possible to represent the time series by the autocorrelation coefficients $(r_0, r_1, \ldots, r_{n-1})$. These coefficients fully determine the autocorrelation matrix $\mathbf{R}$ of the discrete-time stochastic process which is a Toeplitz Hermitian Positive Definite (THPD) matrix:

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1^* & r_2^* & \ldots & r_{n-1}^* \\ r_1 & r_0 & r_1^* & \ldots & r_{n-2}^* \\ r_2 & r_1 & r_0 & \ldots & r_{n-3}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & r_{n-3} & \ldots & r_0 \end{bmatrix}. \tag{1.2}$$

We denote by Z a vector which follows a complex normal distribution. The distribution of Z is characterized by its mean $\mu := \mathbb{E}[Z]$, its covariance matrix $\Gamma := \mathbb{E}\left[ZZ^H\right]$ and its relation matrix $C := \mathbb{E}\left[ZZ^T\right]$. We then denote by $Z \sim \mathcal{CN}(\mu, \Gamma, C)$ the complex Gaussian distribution with mean μ, covariance matrix Γ and relation matrix C. We call circularly-symmetric central complex normal distribution a complex normal distribution with zero mean $(\mu = 0)$ and zero relation matrix $(C = 0)$. A circularly-symmetric central complex normal distribution is therefore characterized by its covariance matrix Γ which is Hermitian Positive Definite. When the complex time series is of length n, it is equivalent to assume that the time series is a realization of stationary centered Gaussian autoregressive discrete-time stochastic process and that the vector $\mathbf{u} = [u(0), u(1), ..., u(n-1)]^T$ is the realization of a circularly-symmetric central complex normal distribution whose covariance matrix $\mathbf{R}$ is Toeplitz Hermitian Positive Definite.

The Burg algorithm 17 can be used to estimate the parameters $(r_0, \mu_1, \ldots, \mu_{n-1})$ from observed time series. The principle of the Burg agorithm is detailed in Appendix C.

In the case of multidimensional time series, the autoregressive model is characterized by the following equation:

$$U(k) + \sum_{j=1}^{n-1} A_i^{n-1}\, U(k-j) = W(k) \tag{1.3}$$

where W is the prediction error vector of size N of covariance matrix Σ. The covariance matrix Σ and the prediction coefficients A_j^{n-1} are square matrices of size $N \times N$. This model can therefore be parameterized by the covariance matrix Σ and the coefficients of the autoregressive model $\left(A_1^{N-1}, \ldots, A_{N-1}^{N-1}\right)$. We define the autocorrelation coefficients $R_i := \mathbb{E}\left[U\,(k+i)\,U\,(k)^H\right]$, where $\cdot^H$ denotes the complex matrix conjugate transpose. The time series studied can also be represented by the autocorrelation coefficient R_0 and the reflection coefficients $\left(A_1^1, \ldots, A_{n-1}^{n-1}\right)$. For two complex matrices A and B, we denote by $A > B$ when the matrix $A - B$ is Hermitian Positive Definite. We show that the reflection coefficients A_i^i belong to the space $\mathcal{D}_N := \left\{M \in \mathbb{C}^{N\times N} \mid I_N - M\overline{M} > 0\right\}$, where $\overline{M} := JM^*J$ where J denotes the anti-diagonal matrix [45]. It is also possible to transform the reflection coefficients $A_i^i \in \mathcal{D}$ into coefficients $M_i \in \mathcal{SD}_N$ where $\mathcal{SD}_N := \left\{M \in \mathbb{C}^{N\times N} \mid I_N - MM^H > 0\right\}$ is called the Siegel disk. We can therefore represent the multidimensional time series studied by the autocorrelation coefficient R_0 and the coefficients $(M_1, \ldots, M_{n-1})$ which belong to the Siegel disk. Using this representation, the time series studied is therefore represented by the coordinates $(R_0, M_1, \ldots, M_{n-1})$ in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. It is also possible to represent the time series studied by the autocorrelation coefficients $(R_0, \ldots, R_{n-1})$. These coefficients caracterize the covariance matrix $\mathbf{R}$ of the vectorized data $\mathbf{U}_n = \left[U(0)^T, \ldots, U(n-1)^T\right]^T$. The covariance matrix of $\mathbf{U}_n$ is the Block-Toeplitz Hermitian Positive Definite matrix $\mathbf{R}$ defined by:

$$\mathbf{R} = \begin{bmatrix} R_0 & R_1^H & R_2^H & \dots & R_{n-1}^H \\ R_1 & R_0 & R_1^H & \dots & R_{n-2}^H \\ R_2 & R_1 & R_0 & \dots & R_{n-3}^H \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{n-1} & R_{n-2} & R_{n-3} & \dots & R_0 \end{bmatrix}. \tag{1.4}$$

When the complex multidimensional time series is of length n, it is equivalent to assume that the time series is a realization of a stationary centered Gaussian autoregressive stochastic process and to assume that the vector $\mathbf{U}_n$ is the realization of a complex centered circularly-symmetric central complex normal distribution whose covariance matrix $\mathbf{R}$ is Block-Toeplitz Hermitian Positive Definite. In Section 2.2.9 we present Algorithm 5 which can be used to estimate the coefficients $(R_0, M_1, \dots, M_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ from observed multidimensional time series.

1.3.2 Summary of Chapter 3: Information Geometry Theory

In Chapter 3, we endow the representation spaces of the complex stationary centered Gaussian autoregressive time series with a Riemannian metric inspired by information geometry. The information geometry theory provides a theoretical framework to endow spaces of parametric probability distributions with a Riemannian metric. As seen in Chapter 2, the studied time series can be considered as realizations of complex circularly-symmetric central complex normal distributions: $\mathbf{u} \sim \mathcal{CN}(0, \mathbf{R}, O)$. In the one-dimensional case, the matrix $\mathbf{R}$ is Toeplitz Hermitian Positive Definite (THPD). In the multidimensional case, the matrix $\mathbf{R}$ is Block-Toeplitz Hermitian Positive Definite (BTHPD). In the more general case where the covariance matrix $\mathbf{R}$ is simply Hermitian Positive Definite, we can use the tools presented in the book of Shun-Ichi Amari [73] to define the information geometry metric on circularly-symmetric central complex normal distributions [15, 82]. This metric on Hermitian Positive Definite matrices $\mathcal{H}_N^+$ is given by the element: $ds^2 = \text{trace}\left(P^{-1}dPP^{-1}dP\right)$. According to the study of the different representation spaces of complex stationary centered Gaussian autoregressive time series carried out in Chapter 2, the Toeplitz covariance matrix $\mathbf{R}$ of a unidimensional stationary time series can be represented by the coefficients $(p_0, \mu_1, \dots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$. Taking inspiration from the construction of the information geometry metric on the space $\mathcal{H}_N^+$, we construct a Riemannian metric on the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$. We denote by $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ the Riemannian manifold obtained. The constructed metric is a product metric defined by:

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n\frac{dp_0^2}{p_0^2} + \sum_{k=1}^{n-1}(n-k)\frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2}. \tag{1.5}$$

For multidimensional time series, the Block-Toeplitz covariance matrix $\mathbf{R}$ can be represented by the coefficients $(R_0, M_1, \dots, M_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. We construct a metric on the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ taking inspiration from the construction of the information geometry metric on $\mathcal{H}_N^+$. We denote by $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ the Riemannian manifold obtained. The constructed metric is a product metric defined by:

$$\begin{aligned} ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} =& n\,\text{trace}\left(P_0^{-1}\,dP_0\,P_0^{-1}\,dP_0\right) \\ &+ \sum_{l=1}^{n-1}(n-l)\,\text{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1}d\Omega_l\left(I-\Omega_l^H\Omega_l\right)^{-1}d\Omega_l^H\right). \end{aligned} \tag{1.6}$$

1.3.3 Summary of Chapter 4: Riemannian manifolds associated with complex valued stationary time series

In Chapter 4, we present some useful geometric tools of the Riemannian manifolds $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ and $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ obtained in Chapter 3. For these two manifolds, the geometric tools presented are: the expression of the element ds^2, the expression of the distance between two points, the scalar product, the norm, an isometry, the Riemannian logarithm map, the Riemannian exponential map, the equation of the geodesics and the sectional curvature. To study these geometric tools, we notice that these manifolds are product manifolds. Indeed, the metric of the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ can be written:

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = nds^2_{\mathbb{R}^{++}} + \sum_{k=1}^{n-1}(n-k)\,ds^2_{\mathbb{D}} \tag{1.7}$$

with $ds^2_{\mathbb{R}^{++}} = \frac{dp_0^2}{p_0^2}$ and $ds^2_{\mathbb{D}} = \frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2}$.

We denote $\mathbb{R}^{++}$ the Riemannian manifold defined on the space $\mathbb{R}_+^*$ by the metric $ds^2_{\mathbb{R}^{++}}$. We call Poincaré disk and denote $\mathbb{D}$ the manifold defined on the complex unit disk $\mathcal{D}$ by the metric $ds^2_{\mathbb{D}}$. We start by studying in detail the manifolds $\mathbb{R}^{++}$ and $\mathbb{D}$, then we use the relation defined by Equation (1.7) to deduce the geometric tools of the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$.

The metric of the Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ can also be written as a product metric:

$$ds^2_{\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}} = n ds^2_{\mathbb{H}_N^{++}} + \sum_{l=1}^{n-1} (n-l)\, ds^2_{\mathbb{SD}_N} \tag{1.8}$$

with

$$\begin{aligned} ds^2_{\mathbb{H}_N^{++}} &= \text{trace}\left(P_0^{-1}\, dP_0\, P_0^{-1}\, dP_0\right) \\ ds^2_{\mathbb{SD}_N} &= \text{trace}\left(\left(I - \Omega_l \Omega_l^H\right)^{-1} d\Omega_l \left(I - \Omega_l^H \Omega_l\right)^{-1} d\Omega_l^H\right). \end{aligned}$$

We denote by $\mathbb{H}_N^{++}$ the manifold defined on the space of Hermitian positive definite matrices $\mathcal{H}_N^+$ by the metric $ds^2_{\mathbb{H}_N^{++}}$. We denote by $\mathbb{SD}_N$ the manifold defined on the Siegel space $\mathcal{SD}_N$ by the metric $ds^2_{\mathbb{SD}_N}$. We first study the manifolds $\mathbb{H}_N^{++}$ and $\mathbb{SD}_N$, then we use the relation given by Equation (1.8) to deduce the geometric tools of the Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. The metric element $ds^2_{\mathbb{SD}_N}$, the distance and the scalar product of the manifold $\mathbb{SD}_N$ are given by Ben Jeuris and Raf Vandebril [45]. To the authors' knowledge, the explicit expression of the Riemannian logarithm map, of the Riemannian exponential map and of the sectional curvature of the Siegel space were not known, their explicit expressions is therefore a major contribution of this thesis.

1.3.4 Summary of Chapter 5: Machine Learning on Riemannian manifolds

In Chapter 5, we detail machine learning algorithms adapted to the structure of Riemannian manifolds. Most classic machine learning algorithms implicitly consider the data to be classified to be Euclidean data. In a Euclidean space, it is possible to add two elements, multiply an element by a scalar and to compute easily the mean x_{mean} of the elements $(x_1, \ldots, x_N)$: $x_{mean} = \frac{1}{N}\sum_{i=1}^N x_i$. The distance between two points $a = (a_1, \ldots, a_n)$ and $b = (b_1, \ldots, b_n)$ is defined by: $d(a,b) = \left(\sum_{i=1}^n (b_i - a_i)^2\right)^{1/2}$. However, these operations (among others) are not defined in Riemannian manifolds in general. For a machine learning algorithm to operate in a Riemannian manifold while respecting its structure, it must therefore be based on operations which are defined in Riemannian manifolds. For the manifolds $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ and $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$, the operations which rely on the geometric tools presented in Chapter 4 will be well defined. Some classical Euclidean machine learning algorithms use the notion of distance as their only geometric tool to classify data. To adapt these algorithms to data belonging to a non-Euclidean Riemannian manifold, it suffices to replace the Euclidean distance by the expression of the distance in the Riemannian manifold. Among the supervised classification algorithms using only the notion of distance, we present the k-nearest neighbors (k-NN) algorithm in Section 5.3.1 and the Kernel Density Estimation Classifier algorithm in Section 5.3.2. Among the clustering algorithms using only the notion of distance, we present the Hierarchical Agglomerative Clustering (HAC) algorithm in Section 5.4.1.

Other classical Euclidean machine learning algorithms use the notion of mean. To use these algorithms in a non-Euclidean Riemannian manifold, we must define the notion of mean in a general Riemannian manifold. In Section 5.1.2, we define the *mean* of the set of points $(x_1, \ldots, x_N)$ as the set of local minima of the function:

$$H_2(x) = \sum_{i=1}^N d^2(x, x_i). \tag{1.9}$$

A *Hadamard* manifold is a Riemannian manifold that is complete, simply connected and has everywhere non-positive sectional curvature. When the studied manifold is a Hadamard manifold, the function $H_2(x)$ has a unique local minimum which is also the global minimum [82], [6]. We show in Chapter 4 that the manifolds $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ and $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ are Hadamard manifolds. To approximate the value of the mean in a Hadamard manifold, we can perform a gradient descent on the function H_2. The gradient of this function has for expression:

$$G_2(x) = -2 \sum_{i=1}^N \log_x(x_i). \tag{1.10}$$

The computation of the gradient therefore involves the Riemannian logarithm map. To move on the Riemannian manifold in the direction of the opposite of the gradient, we then use the Riemannian exponential map. A mean approximation algorithm using a gradient descent is detailed in Algorithm 8. There is also a stochastic method to approximate the mean [6], this method is presented in Algorithm 9. In Chapter 5, we present two algorithms using a computation of

the mean to classify data: the supervised algorithm named Nearest Centroid Classifier presented in Section 5.3.3 and the k-means clustering algorithm presented in Section 5.4.3. These two algorithms use three geometric tools: the distance, the Riemannian logarithm map and the Riemannian exponential map. For each algorithm presented in Chapter 5, we will specify the geometric tools on which it is based and the complexity of the algorithm with respect to the number of input data.

1.3.5 Summary of Chapter 6: Application to simulated unidimensional stationary centered Gaussian autoregressive time series classification

In Chapter 6, we use some machine learning algorithms presented in Chapter 5 to the classification of simulated one-dimensional time series. The simulated time series are complex one-dimensional stationary centered Gaussian autoregressive time series. As seen in Section 2.1 of Chapter 2, these time series can be considered realizations of circularly-symmetric central complex normal distributions $\mathcal{CN}(0, \mathbf{R}, 0)$ whose covariance matrix $\mathbf{R}$ is Toeplitz. We can therefore simulate a time series Z of covariance matrix $\mathbf{R}$ using the equation $Z = \mathbf{R}^{1/2}X$ where X is a standard complex normal random vector. We use this property to simulate one-dimensional time series in Chapter 6. We simulate 200 time series using the Toeplitz Hermitian Positive Definite (THPD) matrix $\mathbf{R}_0$ and 200 time series using the THPD matrix $\mathbf{R}_1$. Then we split each of its two datasets into two: 100 time series are used as the training dataset and 100 time series are used as the testing dataset. We present several representation spaces of complex one-dimensional stationary centered Gaussian autoregressive time series. In each of these representation spaces, we first use the Tangent PCA algorithm 10 to visualize the simulated datasets. Then, we use the supervised algorithm of the nearest centroid classifier presented in Section 5.3.3 to classify the dataset. We finally visualize the Tangent PCA of the classification result obtained and the corresponding confusion matrix.

1.3.6 Summary of Chapter 7: Application to simulated multidimensional stationary centered Gaussian autoregressive time series classification

In Chapter 7, we apply some machine learning algorithms presented in Chapter 5 to simulated multidimensional time series classification. The simulated time series are multidimensional complex stationary centered Gaussian autoregressive time series. As shown in Section 2.2 of Chapter 2, these time series can be considered as realizations of circularly-symmetric central complex normal distributions $\mathcal{CN}(0, \mathbf{R}, 0)$ whose covariance matrix $\mathbf{R}$ is Block-Toeplitz. We can therefore simulate a time series Z of covariance matrix $\mathbf{R}$ using the equation $Z = \mathbf{R}^{1/2}X$ where X is a standard complex Gaussian random variable. We use this property as well as the expression of the associated autoregressive model to simulate complex multidimensional time series. We simulate 200 time series using the same Block-Toeplitz covariance matrix $\mathbf{R}_0$ and 200 time series using the Block-Toeplitz covariance matrix $\mathbf{R}_1$. Then we split each of these two datasets into two: 100 time series are used as the training dataset and 100 time series are used as the testing dataset. We present several representation spaces of complex multidimensional stationary centered Gaussian autoregressive time series. In each of these representation spaces, we first use the Tangent PCA algorithm 10 to visualize the simulated datasets. We then use the supervised nearest centroid classifier algorithm presented in Section 5.3.3 to classify this dataset. We finally visualize the Tangent PCA of the classification result obtained and the corresponding confusion matrix.

1.3.7 Summary of Chapter 8: Application to radar clutter clustering

In Chapter 8, we present an application of the classification method of complex stationary centered Gaussian autoregressive time series presented in this thesis to the clustering of radar time series. The radar data studied come from a Thales radar located in Saint-Mandrier on the French Mediterranean coast and oriented towards the sea. To obtain information on its environment, the studied radar sends electromagnetic waves which will be reflected on the environment and then be recorded by the radar. It is possible to distinguish different directions of origin of the echoes, we focus here on the beam of zero elevation. In the same angular sector of a few degrees, the Thales radar studied sends a burst which is a series of about ten identical pulses. Between each pulse their is a listening phase during which the reflections of the emitted pulses are recorded. During the listening phase, the beginning of the recorded signal corresponds to the reflection of the signal emitted on elements close to the radar and the end of the recorded signal corresponds to elements far from the radar. Each listening phase is then subdivided temporally and we call distance cell the spatial zone observed by the radar during a subdivision. Each distance cell is associated with a complex number. The modulus of this complex number represents the modulus of the signal recorded in the temporal subdivision corresponding to cell studied. The phase of this complex number represents the difference of phase between the emitted signal and the recorded signal. This complex number is obtained by making a convolution between the emitted signal and the recorded signal, this operation is called pulse compression. During each burst, we represent the information associated with the observed angular sector in the form of a matrix U. Each row of the matrix U represents the same distance cell and each column represents the same pulse. We

will note:

$$U = \begin{bmatrix} u_{0,0} & u_{0,1} & \dots & u_{0,n-1} \\ u_{1,0} & u_{1,1} & \dots & u_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-1,0} & u_{N-1,1} & \dots & u_{N-1,n-1} \end{bmatrix} \tag{1.11}$$

where N represents the number of distance cells (approximately 900) and n represents the number of pulses (about 10). Concerning the Thales radar studied, the closest distance cells to the radar are located about 5 kilometers from the radar, the furthest distance cells are located about 60 kilometers, each distance cell having a radial length of 60 meters. When studying the next angular sector (the radar turns clockwise), the radar sends a new burst in the corresponding direction. The radar information associated with this new angular sector will be represented by a new matrix. The number of pulses of a burst and the listening time between two pulses can vary from one burst to another, two different bursts will therefore be associated with matrices of different sizes in general. We now seek to perform the clustering of the distance cells around the radar according to the nature of the associated environment. We call targets the objects of primary interest such as airplanes, drones, missiles, ships... We call radar clutter the signals related to the environment such as waves, fields, forests, urban areas, rain, hail... We assume in our study that the vast majority of the distance cells to be classified represent radar clutter (and not targets). We assume that the complex valued time series associated with radar clutter are stationary centered Gaussian autoregressive time series. We show in Chapter 2 that these time series can be represented by the coefficients $(p_0, \mu_1, \dots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathbb{D}^{n-1}$. The value of these coefficients is then associated with the nature of the clutter: the coefficient p_0 represents the intensity of the recorded time series and the coefficients $(\mu_1, \dots, \mu_{n-1})$ represent the Doppler structure of the time series. We choose to perform the clustering of the distance cells using only the coefficients $(\mu_1, \dots, \mu_{n-1})$ representing the Doppler information. For each distance cell, we estimate these coefficients thanks to the Burg algorithm presented in detail in Appendix C. We will then use the k-means algorithm 12 to perform the clustering of the studied distance cells, each distance cell being represented by the coefficients $(\mu_1, \dots, \mu_{n-1})$ belonging to the space $\mathbb{D}^{n-1}$ endowed with the metric described in Section 4.1.3 when we omit the power term p_0. For this study, we choose $n = 8$, which corresponds to the number of pulses of the shortest burst. We will visualize the result of the clustering in the product of complex unit disks $\mathcal{D}^{n-1}$ and we will visualize the spatial result of the clustering on a ground map to interpret the results obtained.

1.3.8 Summary of Chapter 9: Application to stereo audio classification

Finally, in Chapter 9 we use several machine learning algorithms presented in Chapter 5 to visualize and perform the supervised classification of stereo audio data. The classified data corresponds to soundtracks of stationary sounds downloaded from YouTube. We downloaded three types of stationary sounds: sounds of wood fires, sounds of waterfalls and sounds of winds. Other stationary noises include the sea when the sound of the waves is constant, heavy rain, showers, ventilations, hair dryers, clothes dryers, air conditioning, continuous engine noise, gas coming out of a gas stove... Each of the three downloaded datasets (fires, waterfalls, winds) is then divided into a training dataset and a testing dataset. Our goal now is to classify this dataset. A stereo soundtrack is characterized by a scalar which represents the sampling frequency (48000 Hz here) and a real matrix whose number of lines is equal to the number of seconds of recording multiplied by the sampling frequency. This matrix has two columns which corresponds to the number of recording channels. All downloaded soundtracks must have the same sample rate, we also check that the two recording channels are different (otherwise it is a mono recording). We assume that the two-dimensional real valued audio time series to be classified are stationary centered Gaussian autoregressive time series. We thus place ourselves within the framework of the multidimensional stationary centered Gaussian autoregressive time series presented in Section 2.2. For each stereo audio recording, we estimate the coefficients $(R_0, M_1, \dots, M_{n-1})$ belonging to the space $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ using Algorithm 5. Here the dimension of the time series is $N = 2$, we choose $n = 4$ for the number of coefficients which corresponds to an autoregressive model of order three. We endow the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ with the metric described in Section 4.2.3. We note that when the input time series are real valued, the coefficients $(R_0, M_1, \dots, M_{n-1})$ estimated by Algorithm 5 are also real. We notice that all the geometric tools presented in Section 4.2.3 only involve real elements when the input matrices are real: the submanifold of $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ made up of products of real matrices is therefore a totally geodesic submanifold of $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. The geometric tools presented in Section 4.2.3 are therefore perfectly suited to the particular case of products of real matrices of the manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. The classification results of our method on the audio dataset made from YouTube videos are presented in Section 9.2 using the nearest centroid classifier algorithm presented in Section 5.3.3.

1.4 Main contributions

We summarize in this section the technical contributions made by this thesis.

1.4.1 Theoretical reflection coefficients of stationary continuous stochastic processes with Gaussian spectrum shape

In appendix B, the autocorrelation function R_f of a complex valued continuous stochastic process f is defined by:

$$R_f(\tau) = \int_{-\infty}^{+\infty} f(t+\tau) f^*(t) dt. \tag{1.12}$$

This definition is linked to the definition of the autocorrelation coefficients of a discrete-time stationary stochastic process $\mathbf{u}$ given in Section 2.1.2 by:

$$r_k = r(k) = \mathbb{E}\left[\mathbf{u}(n)\mathbf{u}^*(n-k)\right] = \mathbb{E}\left[\mathbf{u}(n+k)\mathbf{u}^*(n)\right] \quad \forall k \in \mathbb{Z}. \tag{1.13}$$

These two definitions are linked by the relation:

$$r_k = R_f(kT). \tag{1.14}$$

For simplicity we normalize the time scale and consider the sampling period $T = 1$ in this thesis.
The autocorrelation function R_f has a Fourier transform called the power spectral density:

$$S_f(\xi) = \int_{-\infty}^{+\infty} R_f(\tau) e^{-i2\pi\xi\tau} d\tau. \tag{1.15}$$

The autocorrelation function $R_f(\tau)$ is therefore the inverse Fourier transform of the power spectral density $S_f(\xi)$:

$$R_f(\tau) = \int_{-\infty}^{+\infty} S_f(\xi) e^{+i2\pi\xi\tau} d\xi. \tag{1.16}$$

We prove in Appendix B that the stationary continuous stochastic process f which power spectral density S_f has the shape of Gaussian distribution of mean m and variance σ^2 with a power coefficient P, *i.e.*:

$$S_f(\xi) = P \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\xi-m)^2}{2\sigma^2}} \tag{1.17}$$

has the following theoretical autocorrelation coefficients:

$$r_k = R_f(k) = P e^{i2\pi mk} e^{-2\pi^2\sigma^2 k^2} \quad \forall k \geqslant 1. \tag{1.18}$$

These coefficients can be obtained using the inverse Fourier transform given in Equation (1.16). Thanks to the Levinson algorithm 13, we can then compute explicitly the expression of the reflection coefficients $(\mu_1, \ldots, \mu_n)$ from the autocorrelation coefficients $(r_0, r_1, \ldots, r_n)$. We obtain:

$$\mu_k = (-1)^k e^{i2\pi mk} e^{-2\pi^2\sigma^2 k} \quad \forall k \geqslant 1. \tag{1.19}$$

Note that in this particular case, we have $\mu_k = \mu_1^k$. We also note also that the argument of the coefficient μ_k depends only on the mean m and its modulus depends only on the variance σ^2.

1.4.2 The Siegel manifold geometrical tools

We show in Section 2.2.8 that the multidimensional complex stationary centered Gaussian autoregressive time series can be represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ where $\mathcal{H}_N^+$ represents the set of Hermitian Positive Definite matrices and $\mathcal{SD}_N$ represents the Siegel disk defined by the set of complex matrices whose singular values are strictly lower than 1:

$$\mathcal{SD}_N = \left\{ M \in \mathbb{C}^{N\times N} \mid I - MM^H > 0 \right\}. \tag{1.20}$$

In Section 3.2.3 of Chapter 3, we endow the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ with a product metric inspired by information geometry. We denote $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ the Riemannian product manifold obtained and we denote $ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}}$ its metric which has the following expression:

$$ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} = n\,\mathrm{trace}\left(P_0^{-1}\,dP_0\,P_0^{-1}\,dP_0\right) + \sum_{l=1}^{n-1}(n-l)\,\mathrm{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1}d\Omega_l\left(I-\Omega_l^H\Omega_l\right)^{-1}d\Omega_l^H\right). \tag{1.21}$$

We denote $\mathbb{H}_N^{++}$ the Riemannian manifold obtained when we endow the space $\mathcal{H}_N^+$ with the metric:

$$ds^2_{\mathbb{H}_N^{++}} = \mathrm{trace}\left(P_0^{-1}\,dP_0\,P_0^{-1}\,dP_0\right). \tag{1.22}$$

We denote $\mathbb{SD}_N$ the manifold obtained when we endow the space $\mathcal{SD}_N$ with the metric:

$$ds^2_{\mathbb{SD}_N} = \mathrm{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1}d\Omega_l\left(I-\Omega_l^H\Omega_l\right)^{-1}d\Omega_l^H\right). \tag{1.23}$$

We then have:

$$ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} = n ds^2_{\mathbb{H}_N^{++}} + \sum_{l=1}^{n-1}(n-l)\,ds^2_{\mathbb{SD}_N}. \tag{1.24}$$

To study the geometric tools of the product Riemannian manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$, we therefore start by studying the manifolds $\mathbb{H}_N^{++}$ and $\mathbb{SD}_N$. The Riemannian manifold $\mathbb{H}_N^{++}$ is an already known manifold, we recall certain of its geometric tools in Section 4.2.1. However, the geometric tools of the Siegel manifold $\mathbb{SD}_N$ are less well known. In the article written by Ben Jeuris and Raf Vandebril [45], the distance map and the expression of the scalar product are given. The square of the distance between two points $\Omega,\Psi\in\mathcal{SD}_N$ has the value:

$$d^2_{\mathbb{SD}_N}(\Omega,\Psi) = \frac{1}{4}\,\mathrm{trace}\left(\log^2\left(\frac{I+C^{1/2}}{I-C^{1/2}}\right)\right) \tag{1.25}$$

$$= \mathrm{trace}\left(\mathrm{arctanh}^2\left(C^{1/2}\right)\right) \tag{1.26}$$

with $C=(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Psi^H\right)^{-1}$.
The expression of the scalar product is, $\forall\Omega\in\mathcal{SD}_N,\forall v,w\in\mathbb{C}^{N\times N}$:

$$\langle v,w\rangle_\Omega = \frac{1}{2}\,\mathrm{trace}\left(\left(I-\Omega\Omega^H\right)^{-1}v\left(I-\Omega^H\Omega\right)^{-1}w^H\right) + \frac{1}{2}\,\mathrm{trace}\left(\left(I-\Omega\Omega^H\right)^{-1}w\left(I-\Omega^H\Omega\right)^{-1}v^H\right). \tag{1.27}$$

The following application:

$$\Phi_\Omega(\Psi) = \left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \tag{1.28}$$

is said to be an isometry on the Siegel manifold in [45]. We prove this property in Appendix E. We also prove that the differential of the isometry Φ has the following expression:

$$D\Phi_\Omega(\Psi)[h] = \left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}h\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}. \tag{1.29}$$

We prove that the inverse of the function Φ described in Equation (1.28) has the following property:

$$\Phi_\Omega^{-1}(\Psi) = \Phi_{-\Omega}(\Psi). \tag{1.30}$$

This property of isometry Φ is simply mentioned (without proof) in the article of Ben Jeuris and Raf Vandebril [45].
The major theoretical contribution of this thesis is the explicit expression of the Riemannian logarithm map, of the Riemannian exponential map and of the sectional curvature of the Siegel manifold.
The Riemannian logarithm map has the following expression:

$$\log_\Omega(\Psi) = \left(I-\Omega\Omega^H\right)^{1/2}V'\left(I-\Omega^H\Omega\right)^{1/2} \tag{1.31}$$

$$\left[\begin{array}{l} V' = \mathrm{arctanh}(X)\,X^{-1}\Psi' \quad \text{where } X=\left(\Psi'\Psi'^H\right)^{1/2} \\ \Psi' = \left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \end{array}\right.$$

The Riemannian exponential map has the expression:

$$\exp_{\Omega}(V) = \left(I - \Omega\Omega^H\right)^{-1/2}\left(\Psi' + \Omega\right)\left(I + \Omega^H\Psi'\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{1.32}$$
$$\left[\begin{array}{l} \Psi' = \tanh(Y)\, Y^{-1} V' \quad \text{where } Y = \left(V' V'^{\,H}\right)^{1/2} \\ V' = \left(I - \Omega\Omega^H\right)^{-1/2} V \left(I - \Omega^H\Omega\right)^{-1/2} \end{array}\right.$$

The sectional curvature at zero of the plan σ defined by the orthonormal matrices E_1 and E_2 has the following expression:

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right). \tag{1.33}$$

The value of the sectional curvature at any point can then be obtained using the isometry Φ described in Equation (1.28).

For more details, the geometrical tools of the Siegel space presented in this thesis are summarized in Section 4.2.2. We dedicate Appendix E to the study of the Siegel disk and to the proofs of the new mathematical tools brought in this thesis.

The knowledge of the logarithm and exponential maps of the Riemannian manifold $\mathbb{SD}_N$ allows us to construct the logarithm and exponential maps of the product Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$, which we do in Section 4.2.3. The knowledge of these applications allows us, among other things, to use the machine learning algorithms presented in Chapter 5 involving a computation of the mean when classifying multidimensional stationary time series represented in the Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$.

1.4.3 Infinitesimal right triangles and sectional curvature in Riemannian manifolds

In Section D.4, we prove the following relation between the length of the hypotenuse of an infinitesimal right triangle and the sectional curvature:

Theorem 2. *Let U be a relatively compact convex open set of a complete Riemannian manifold M. There exists a constant $C_U > 0$ such that for any triangle OAB of U rectangle in 0, we have the following inequality:*

$$\left|d^2(A,B) - d^2(O,A) - d^2(O,B) + \frac{1}{3}K(\sigma)\, d^2(O,A)\, d^2(O,B)\right| \leqslant C_U\left(d(O,A) + d(O,B)\right)^5 \tag{1.34}$$

In the works of Andrei A. Agrachev, Davide Barilari and Luca Rizzi [4], this theorem is stated in the particular case where point O is fixed and the directions of points A and B starting from O are fixed. The result stated in their works is said to be due to Loeper and Villani. The theorem 2 studied here is stated and proved independently of these previous works, in a more general context.

1.4.4 Stationary audio time series classification in $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

We apply the multidimensional model to the classification of stationary stereo audio time series in Chapter 9. To our knowledge, this is the first time that the Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ is used to classify stationary stereo audio time series.

1.4.5 The radar spatio-temporal model

In Appendix G and in the article [83], we propose a spatio-temporal model to represent the radar clutter. In this model, we assume that the times series recorded by the radar during a burst are wide-sense stationary on both the temporal axis and the spatial axis, the spatial axis considered being the radial axis.

Indeed, we assume that the observation matrix Z of the zero elevation beam recorded during a burst for a group of consecutive cells has the following spatio-temporal correlation:

$$Z = p_0^{1/2} R_s^{1/2} N R_t^{T\,1/2} \tag{1.35}$$

with:

- Z: radar observation matrix of size $(p,\, n)$, where p corresponds to the number of cells and n corresponds to the number of pulses emitted during the burst.

- p_0: a positive real number which corresponds to the expectation of the quadratic power, i.e. $p_0 = \mathbb{E}\left[|z_{i,j}|^2\right]$ for all $(i,j) \in [\![1,p]\!] \times [\![1,n]\!]$.
- R_s: the scaled spatial autocorrelation matrix of size (p, p). It is a Toeplitz HPD matrix since the signal is assumed to be wide-sense stationary on the spatial axis. Its diagonal coefficients are equal to 1.
- R_t: the scaled temporal autocorrelation matrix of size (n, n). It is a Toeplitz HPD matrix since the signal is assumed to be wide-sense stationary on the temporal axis. Its diagonal coefficients are equal to 1.
- N: matrix of size (p, n) whose coefficients are independent standard complex Gaussian random variables.

This model looks like to model presented by Romain Couillet et al. [24], except for the structure of the spatial covariance matrix R_s which is only assumed to be Toeplitz here.

We also present an equivalent vectorized model:

$$\widetilde{Z} = p_0^{1/2} R_{st}^{1/2} \widetilde{N} \tag{1.36}$$

where $\widetilde{Z}$ is the vectorization of matrix Z and $\widetilde{N}$ is the vectorization of matrix N. The scaled spatio-temporal correlation matrix R_{st} is then the Kronecker product of the scaled spatial correlation matrix R_s and the scaled temporal correlation matrix R_t:

$$R_{st} = R_s \otimes R_t. \tag{1.37}$$

Using the specific structure of the covariance matrix $p_0 R_{st}$, we then construct a Riemannian manifold inspired by information geometry to represent the information contained in a group of spatially close cells.

1.4.6 Open source Python codes on geomstats

Some of the codes used in this thesis are put online on the public package geomstats [52]. Geomstats is a public Python package about machine learning in Riemannian manifolds. This package provides Riemannian manifolds and machine learning algorithms adapted to Riemannian manifolds. This library provides a framework so that each of the available machine learning algorithms is compatible with each of the available Riemannian manifolds. A user working on a Riemannian manifold different from those available on geomstats will still be able to use all the machine learning algorithms present on geomstats provided that the manifold used is defined using the conventions of geomstats. In this thesis, we have used the machine learning algorithms available on geomstats to classify complex one-dimensional stationary time series in the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ and complex multidimensional stationary time series in the Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. The product of Poincaré disks $\mathbb{D}^{n-1}$ used in Chapter 8 for radar clutter clustering was uploaded to the geomstats library during this thesis work. The Kernel Density Estimation Classifier classification algorithm as well as several usual kernel functions presented in Section 5.3.2 were also uploaded to geomstats during this thesis work. Eventually, we would like most of the Python codes implemented for this thesis work to be shared on geomstats.

1.5 Perspectives

In this section, we suggest two possibilities for future developments of this thesis. The first possibility concerns the supervised classification of radar clutter in the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ whose geometric tools are presented in Section 4.1.3. The second, more theoretical, concerns the possibility of finding the formulas of the Siegel manifold presented in Section 4.2.2 and in Appendix E by considering the Siegel manifold as a homogeneous Riemannian manifold.

1.5.1 Supervised radar clutter classification

The radar dataset used in this thesis has been provided without labels. In Chapter 8, we perform unsupervised classification of radar clutter. In order to perform the supervised classification of radar data, we labeled them using the Corine Land Cover database. It is a public database of the French government which partitions the French territory into polygons, each polygon being associated with a ground label. We used the following labels: artificial surfaces, agricultural areas, forest and semi natural areas, wetlands and finally water bodies. Each of these labels is divided into subcategories which we have not used. We found the ground label of each distance cell studied from its GPS position. However, the classification performance obtained using these labels was not satisfactory. The classification method proposed in this thesis classifies the distance cells from the Doppler information, but it seems that the Doppler information cannot be used to find the ground labels with precision. This is highlighted by the radar clutter clustering results presented in Section 8.4 where we can see that the clusters are not related to the ground labels: they are related to the reflection coefficients and therefore to the Doppler spectra. The Doppler information highly depends on the position of the radar with respect to the distance cell studied. For the supervised classification of radar clutter, we therefore suggest using labels referring to the Doppler information rather than to the ground type.

1.5.2 The study of the Siegel manifold $\mathbb{SD}_N$ as an homogeneous Riemannian manifold

A Riemannian manifold $\mathcal{M}$ is said to be *homogeneous* if there exists a Lie group G which acts transitively and smoothly by isometries on $\mathcal{M}$. A homogeneous manifold $\mathcal{M}$ can then be considered as the quotient of a Lie group G of isometries by a subgroup K where K is the subgroup of isometries of G which fixes a certain point $m \in \mathcal{M}$.

The major theoretical contribution of this thesis concerns the study of the geometric tools of the Siegel manifold $\mathbb{SD}_N$ presented in Section 4.2.2 and in Appendix E. According to the work of Salem Said et al. [70,71], the Siegel manifold has a homogeneous manifold structure which has not been used here. We could therefore try to find the geometric formulas of the Siegel manifold presented in this thesis by exploiting its structure as a homogeneous manifold.

In dimension $N = 1$, the Siegel manifold $\mathbb{SD}_N$ corresponds to the Poincaré disk $\mathbb{D}$ which can be considered as the following quotient [62]:

$$\mathbb{D} = \frac{SU(1,1)}{U(1)}$$

with:

$$SU(1,1) = \left\{ g_{\alpha,\beta} = \begin{pmatrix} \alpha & \beta \\ \bar{\beta} & \bar{\alpha} \end{pmatrix}, |\alpha|^2 - |\beta|^2 = 1 \right\}$$
$$U(1) = \left\{ \begin{pmatrix} e^{i\theta} & 0 \\ 0 & e^{-i\theta} \end{pmatrix}, \theta \in \mathbb{R} \right\}$$

Indeed, the Lie group $SU(1,1)$ acts transitively on $\mathbb{D}$ by isometries:

$$SU(1,1) \times \mathbb{D} \to \mathbb{D}$$
$$\left(\begin{pmatrix} \alpha & \beta \\ \bar{\beta} & \bar{\alpha} \end{pmatrix}, z \right) \mapsto \frac{\alpha z + \beta}{\bar{\beta} z + \bar{\alpha}}$$

and the elements of $U(1)$ fix the point 0.

The quotient structure of the Poincaré disc has been used in particular in the work of Pierre-Yves Lagrave to build equivariant neural networks [61–63].

The Poincaré disk $\mathbb{D}$ can also be considered as the following quotient:

$$\mathbb{D} = \frac{U(1,1)}{U(1) \times U(1)}$$

with

$$U(1,1) = \left\{ g \in M_2(\mathbb{C}), gPg^H = P \text{ where } P = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\}.$$

Here, $.^H$ denotes the transconjugation operator of a complex matrix, the operator .* denotes the complex conjugation and $.^T$ denotes the matrix transposition.

More generally, the Siegel manifold $\mathbb{SD}_N$ of any dimension N is also a homogeneous manifold. The Siegel manifold can indeed be considered as the following quotient:

$$\mathbb{SD}_N = \frac{U(N,N)}{U(N) \times U(N)}.$$

where $U(N)$ is the unitary group of dimension N:

$$U(N) = \left\{ U \in M_N(\mathbb{C}), UU^H = I_N \right\}.$$

More precisely, if we set P and S the following matrices:

$$P = \begin{pmatrix} I_N & 0_N \\ 0_N & -I_N \end{pmatrix} \quad S = \begin{pmatrix} 0_N & I_N \\ -I_N & 0_N \end{pmatrix},$$

we then have [71]:

$$U(N,N) = \left\{ g = \begin{pmatrix} A & B \\ C & D \end{pmatrix}, gPg^H = P \right\}$$

and

$$U(N)\times U(N)=\left\{k=\begin{pmatrix}U&0\\0&V\end{pmatrix},(U,V)\in U(N)^2\right\}.$$

The groups $U(N,N)$ and $U(N)\times U(N)$ act on the Siegel manifold $\mathbb{SD}_N$ by matrix fractional transformations:

$$\begin{aligned}&U(N,N)\times\mathbb{SD}_N\to\mathbb{SD}_N\\&\left(\begin{pmatrix}A&B\\C&D\end{pmatrix},\Omega\right)\mapsto(A\Omega+B)(C\Omega+D)^{-1}.\end{aligned}$$

Moreover, note that we distinguish here the Siegel manifold $\mathbb{SD}_N$ from the *symmetric* Siegel manifold (restricted to symmetric matrices, see section E.6). The symmetric Siegel manifold can be considered of as the quotient $\frac{Sp(2N,\mathbb{R})}{U(N)}$ where $Sp(2N,\mathbb{R})$ is the real symplectic group [70]. Specifically, we have [71]:

$$Sp(2N,\mathbb{R})=\left\{g=\begin{pmatrix}A&B\\C&D\end{pmatrix},gPg^H=P\text{ and }gSg^T=S\right\}$$

and

$$U(N)=\left\{k=\begin{pmatrix}U&0\\0&U^*\end{pmatrix},UU^H=I_N\right\}.$$

The Lie algebras of the Lie groups mentioned in this section are described in [71].

Chapter 2

Stationary centered complex Gaussian autoregressive time series

In this chapter, we study complex stationary centered Gaussian autoregressive time series. We will present the case of one-dimensional time series in Section 2.1 and we present the more general case of multidimensional time series in Section 2.2. For one-dimensional time series as for multidimensional time series, we mathematically define the properties of the time series studied. We then describe several representation spaces for these time series and we will present different relationships between these spaces.

Contents

2.1 Unidimensional stationary centered complex Gaussian autoregressive time series

In this section, we mathematically define one-dimensional complex stationary centered Gaussian autoregressive time series. Throughout this section, we refer extensively to the book writen by Simon Haykin called "Adaptive Filter Theory" [44].

2.1.1 Stochastic process

According to the first chapter of Simon Haykin's book [44], the term *stochastic process* is used to describe the temporal evolution of a random phenomenon subject to probabilistic laws. The stochastic processes that interest us here are those defined for regularly spaced discrete times. This restriction appears naturally in many applications such as in the study of radar time series presented in Chapter 8 and in the study of audio time series presented in Chapter 9. A stochastic process is not the realization of a single function of time: it is in theory an infinite number of different realizations of the process. A particular realization of a discrete-time stochastic process is called a *time series*. To simplify the notations, we normalize the time with respect to the sampling period. Hence, the sequence $u(n-M), \ldots, u(n-1), u(n)$ represents the time series of length $M+1$ made up of the present observation $u(n)$ and of the M previous observations done at times $n-1, \ldots, n-M$. We will use to bold letter $\mathbf{u}(n)$ to denote the stochastic process and we will denote by $u(n)$ a realization of this process, i.e. a time series. Implicitly, we call stochastic process $\mathbf{u}(n)$ the sequence of infinite length $\{\mathbf{u}(n)\}_{n\in\mathbb{Z}}$. Likewise, the time series $u(n)$ implicitly denotes the sequence of infinite length $\{u(n)\}_{n\in\mathbb{Z}}$.

2.1.2 Stationarity

We say that a stochastic process is strictly stationary if its statistical properties are invariant by temporal translation. Formally, let $\mathbf{u}_t$ be a continuous-time stochastic process and let f be its joint probability density at times $t_1+\tau, \ldots, t_n+\tau$. The stochastic process $\mathbf{u}_t$ is said to be strictly stationary when:

$$f_{\mathbf{u}}\left(u_{t_1+\tau}, \ldots, u_{t_n+\tau}\right) = f_{\mathbf{u}}\left(u_{t_1}, \ldots, u_{t_n}\right) \quad \forall \tau \in \mathbb{R}, \forall n \in \mathbb{N}, t_1, \ldots, t_n \in \mathbb{R}. \tag{2.1}$$

In practice, it is generally impossible to determine the joint probability distribution of a stochastic process from discrete observations of this process. In general, we will therefore content ourselves with a partial characterization of this process from its moments of order one and two.

We define the *mean* value of the discrete-time stochastic process $\mathbf{u}(n)$ by:

$$\mu(n) = \mathbb{E}\left[\mathbf{u}(n)\right] \tag{2.2}$$

where $\mathbb{E}$ denotes the statistical expectation operator.

We define the *autocorrelation function* of the process $\mathbf{u}(n)$ by:

$$r\left(n, n-k\right) = \mathbb{E}\left[\mathbf{u}(n)\mathbf{u}^*(n-k)\right] \quad \forall k \in \mathbb{Z} \tag{2.3}$$

where the asterisk denotes the complex conjugation.

We define the *autocovariance function* of the process $\mathbf{u}(n)$ by:

$$c\left(n, n-k\right) = \mathbb{E}\left[\left(\mathbf{u}(n) - \mu(n)\right)\left(\mathbf{u}(n-k) - \mu(n-k)\right)^*\right] \quad \forall k \in \mathbb{Z}. \tag{2.4}$$

According to Equations (2.2), (2.3) and (2.4), the mean value $\mu(n)$, the autocorrelation function $r\left(n, n-k\right)$ and the autocovariance function $c\left(n, n-k\right)$ are related by the following equation:

$$c\left(n, n-k\right) = r\left(n, n-k\right) - \mu(n)\mu^*(n-k). \tag{2.5}$$

For a partial characterization of a stochastic process by its moments of order one and two, we just need to determine the mean $\mu(n)$ and the autocorrelation function $r\left(n, n-k\right)$ or to determine the mean $\mu(n)$ and the autocovariance function $c\left(n, n-k\right)$. Note that for stochastic processes with zero mean ($\mu(n) = 0$ for all $n \in \mathbb{Z}$), the autocorrelation function $r\left(n, n-k\right)$ is equal to the autocovariance function $c\left(n, n-k\right)$.

In the case of discrete strictly stationary signals, the mean value $\mu(n)$, the autocorrelation function $r\left(n, n-k\right)$ and the autocovariance function $c\left(n, n-k\right)$ have simpler expressions. The mean value function is indeed constant:

$$\mu(n) = \mu \quad \forall n \in \mathbb{Z}. \tag{2.6}$$

The autocorrelation and autocovariance functions only depend on the difference k between the time of observation n and the time of observation $n-k$, that is:

$$r\left(n, n-k\right) = r(k) \tag{2.7}$$

and

$$c\left(n, n-k\right) = c(k). \tag{2.8}$$

In the following, we will also denote by $r_k := r(k)$ the autocorrelation coefficients of a wide-sense stochastic process and by $c_k := c(k)$ its autocovariance coefficients. The equations (2.6), (2.7) and (2.8) are not sufficient to guarantee that a stochastic process is stationary in the strict sense. We say that a stochastic process is *weakly stationary* or *wide-sense stationary* when it satisfies Equations (2.6), (2.7) and (2.8). The strict stationarity implies the wide-sense stationarity.

However, there are wide-sense stationary processes which are not strictly stationary. For example, a a sequence of independent random variables $\mathbf{U} = \{\mathbf{u}(n)\}_{n\in\mathbb{Z}}$ with the same mean and the same variance is always wide-sense stationary, but $\mathbf{U}$ is stationary in the strict sense only if the terms $\mathbf{u}(n)$ have the same law [7].

In the following sections, we will focus more specifically on wide-sense stationary processes of zero mean: $\mu = 0$. The partial second-order characterization of these processes is then entirely determined by the value of the autocorrelation coefficients $\{r_k\}_{k\in\mathbb{Z}}$. We will now define the autocorrelation matrix of a wide-sense stationary stochastic process from the autocorrelation coefficients, then we will study the properties of the autocorrelation matrix.

2.1.3 Correlation matrix

Let $\mathbf{u}(n)$ denote a zero-mean wide-sense stationary stochastic process. We denote by $\mathbf{u}_M(n)$ the stochastic process of dimension M defined by:

$$\mathbf{u}_M(n) = \left[\mathbf{u}(n-M+1), \ldots, \mathbf{u}(n-1), \mathbf{u}(n)\right]^T, \tag{2.9}$$

where the superscript T denotes the matrix transposition. We define the correlation matrix of size M of the stochastic process $\mathbf{u}(n)$ by:

$$\mathbf{R}_M = \mathbb{E}\left[\mathbf{u}_M(n)\mathbf{u}_M^H(n)\right], \tag{2.10}$$

where the superscript H denotes the Hermitian transposition which is a standard transposition combined with a complex conjugation. The autocorrelation matrix $\mathbf{R}_M$ will simply be denoted $\mathbf{R}$ when the dimension of the matrix is not ambiguous.

By replacing $\mathbf{u}_M(n)$ in Equation (2.10) by its expression in Equation (2.9), we find using the wide-sense stationarity of $\mathbf{u}(n)$:

$$\mathbf{R} = \begin{bmatrix} r_0 & r_{-1} & \cdots & r_{-M+1} \\ r_1 & r_0 & \cdots & r_{-M+2} \\ \vdots & \vdots & \ddots & \vdots \\ r_{M-1} & r_{M-2} & \cdots & r_0 \end{bmatrix}. \tag{2.11}$$

The coefficient r_0 on the main diagonal is always a positive real. For a complex stochastic process $\mathbf{u}(n)$, the other coefficients r_i for $i \neq 0$ are generally complex.

The correlation matrix $\mathbf{R}$ plays a fundamental role in the study of wide-sense stationary stochastic processes. We now study different properties of the matrix $\mathbf{R}$.

Property 1. *The correlation matrix of a wide-sense stationary discrete-time stochastic process is Hermitian, i.e.* $\mathbf{R} = \mathbf{R}^H$.

To prove this property, it suffices to notice that $r_{-k} = r_k^*$.

The correlation matrix $\mathbf{R}$ is therefore entirely determined by the autocorrelation coefficients r_k for $k = 0, 1, \ldots, M-1$. Indeed, we have:

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1^* & \cdots & r_{M-1}^* \\ r_1 & r_0 & \cdots & r_{M-2}^* \\ \vdots & \vdots & \ddots & \vdots \\ r_{M-1} & r_{M-2} & \cdots & r_0 \end{bmatrix}. \tag{2.12}$$

We say that a square matrix is *Toeplitz* when all the elements of its main diagonal are equal and when all the elements of a same diagonal parallel to the main diagonal are also equal.

Property 2. *The correlation matrix of a wide-sense stationary discrete-time stochastic process is Toeplitz.*

From the structure of the correlation matrix $\mathbf{R}$ described in Equation (2.12), we note that all the elements of the main diagonal are equal to r_0, that all the elements of the first diagonal above the main diagonal are equal to r_1^*, that all the elements of the first diagonal below the main diagonal are equal to r_1 and so on for the other diagonals. The correlation matrix $\mathbf{R}$ is therefore Toeplitz. Note that when $\mathbf{u}(n)$ represents a zero-mean stochastic process, its correlation matrix $\mathbf{R}$ is Toeplitz if and only if the stochastic process $\mathbf{u}(n)$ is wide-sense stationary.

Let $\mathbf{a}$ be a complex non-zero vector of size M. We define y as the scalar product of the vector $\mathbf{a}$ and the stochastic process $\mathbf{u}_M(n)$:

$$y = \mathbf{a}^H \mathbf{u}_M(n). \tag{2.13}$$

Hence, we have:

$$\begin{aligned}\mathbb{E}\left[|y|^2\right] =& \mathbb{E}\left[yy^*\right] \\ =& \mathbb{E}\left[\mathbf{a}^H\mathbf{u}_M(n)\mathbf{u}_M^H(n)\mathbf{a}\right] \\ =& \mathbf{a}^H\mathbb{E}\left[\mathbf{u}_M(n)\mathbf{u}_M^H(n)\right]\mathbf{a} \\ =& \mathbf{a}^H\mathbf{R}\mathbf{a}.\end{aligned} \tag{2.14}$$

Since $\mathbb{E}\left[|y|^2\right] \geqslant 0$, we have:

$$\mathbf{a}^H\mathbf{R}\mathbf{a} \geqslant 0. \tag{2.15}$$

A Hermitian matrix $\mathbf{R}$ which satisfies the property described in the previous equation for any complex vector $\mathbf{a}$ is said to be *nonnegative definite* or *positive semidefinite*. A Hermitian matrix $\mathbf{R}$ which satisfies $\mathbf{a}^H\mathbf{R}\mathbf{a} > 0$ for all non-zero complex vector $\mathbf{a}$ is said to be *positive definite*.

Property 3. *The correlation matrix of a discrete-time stochastic process is always positive semidefinite.*

In most practical cases, the correlation matrix of a discrete-time stochastic process is positive definite.

Property 4. *The correlation matrix of a wide-sense stationary process is always positive definite when and additive noise is present which is almost always the case in practice.*

This property has an important consequence: when the matrix $\mathbf{R}$ is positive definite, its inverse matrix $\mathbf{R}^{-1}$ exists. The presence of this additive noise is in particular present in the autoregressive models studied in next section. The correlation matrix $\mathbf{R}$ of the wide-sense stationary stochastic processes presented in next section will therefore belong to the set of Toeplitz Hermitian positive definite matrices of dimension M which we denote by $\mathcal{T}_M^+$.

2.1.4 Autoregressive models

Let us start by giving the definition of a *white noise* [7].

Definition 1. *We call white noise a stochastic process $\mathbf{w}(n)$ whose:*

- *mean is zero:* $\mathbb{E}\left[\mathbf{w}(n)\right] = 0 \quad \forall n \in \mathbb{Z}$,
- *variance is constant:* $\mathbb{E}\left[|\mathbf{w}(n)|^2\right] = \sigma^2 \quad \forall n \in \mathbb{Z}$,
- *autocorrelation is zero:* $\mathbb{E}\left[\mathbf{w}(n)\mathbf{w}(k)^*\right] = 0 \quad \forall k \neq n$.

A white noise is therefore a wide-sense stationary process by construction.
We now define the autoregressive processes of order M [7, 44].

Definition 2. *We say that a stochastic process $\mathbf{u}(n)$ is an autoregressive process (AR) of order M when it satisfies the following equation:*

$$\mathbf{u}(n) + a_1\mathbf{u}(n-1) + \ldots + a_M\mathbf{u}(n-M) = \mathbf{w}(n) \quad \forall n \in \mathbb{Z}, \tag{2.16}$$

where $a_1, a_2, \ldots, a_M$ are complex constants called the AR parameters and $\mathbf{w}(n)$ is a white noise. We say that a time series $u(n)$ is an autoregressive time series when it is a realisation of an autoregressive stochastic process.

We call *Gaussian autoregressive process* an autoregressive process whose white noise $\mathbf{w}(n)$ is Gaussian. We will detail the properties of the particular case of Gaussian autoregressive processes in Section 2.1.10.

The term "*autoregressive*" comes from the fact that the present value $\mathbf{u}(n)$ of the stochastic process is equal to a linear combination of the M previous values $\mathbf{u}(n-1), \ldots, \mathbf{u}(n-M)$ plus an error term $\mathbf{w}(n)$, as shown by the following equation:

$$\mathbf{u}(n) = -\sum_{k=1}^{M} a_k\mathbf{u}(n-k) + \mathbf{w}(n). \tag{2.17}$$

The left terms of Equation (2.16) can be considered as the convolution of the sequence $\{\mathbf{u}(n)\}$ with the sequence $\{a_n\}$. We can indeed rewrite Equation (2.16) as follows:

$$\sum_{k=0}^{M} a_k\mathbf{u}(n-k) = \mathbf{w}(n), \tag{2.18}$$

by setting $a_0 = 1$.

By taking the *z-transform* on both sides of Equation (2.18), we transform the convolution operation on the left side of the equation into a multiplication of the z-transforms of the sequences $\{\mathbf{u}(n)\}$ and $\{a_n\}$.

We note:

$$H_A(z) = \sum_{n=0}^{M} a_n z^{-n} \tag{2.19}$$

the z-transform of the sequence $\{a_n\}$, and we note:

$$U(z) = \sum_{n=0}^{\infty} \mathbf{u}(n) z^{-n} \tag{2.20}$$

the z-transform of the sequence $\{\mathbf{u}(n)\}$, where z is a complex variable.

We can then rewrite Equation (2.18) into the form:

$$H_A(z)U(z) = W(z), \tag{2.21}$$

where

$$W(z) = \sum_{n=0}^{\infty} \mathbf{w}(n) z^{-n}. \tag{2.22}$$

With the white noise $\mathbf{w}(n)$ acting as input, we may use Equation (2.17) to produce the AR process $\mathbf{u}(n)$ as output. In Haykin [44], this transformation is seen as a filter called *process generator*. The transfer function of this transformation equals:

$$H_G(z) = \frac{U(z)}{W(z)} = \frac{1}{H_A(z)} = \frac{1}{\sum_{n=0}^{M} a_n z^{-n}}. \tag{2.23}$$

The AR process generator transforming $W(z)$ into $U(z)$ is called an *all-pole filter* as its transfer function $H_G(z)$ is entirely determined by the position of its *poles*, as shown by:

$$H_G(z) = \frac{1}{\prod_{i=1}^{M} (1 - p_i z^{-1})}. \tag{2.24}$$

The parameters $p_1, p_2, \ldots, p_M$ are the poles of $H_G(z)$, they correspond to the roots of the *characteristic equation*:

$$1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_M z^{-M} = 0. \tag{2.25}$$

The AR process given in Equation (2.16) is wide-sense stationary if and only if the roots of the characteristic Equation (2.25) all lie inside the unit circle in the z-plane [44]:

$$|p_i| < 1 \quad \forall 1 \leqslant i \leqslant M. \tag{2.26}$$

When the process $\mathbf{u}(n)$ is wide-sense stationary, the coefficient a_M is therefore of modulus strictly less than 1:

$$|a_M| < 1 \tag{2.27}$$

since $a_M = \prod_{i=1}^{M} p_i$ and $|p_i| < 1$ for all $i \in [\![1, n]\!]$.

We denote by μ_M and call *reflection coefficient* the coefficient a_M of the autoregressive model of order M. These coefficients will play an important role in this thesis.

Note that the coefficients of the autoregressive model $a_1, \ldots, a_M$ are the coefficients which minimize the expectation of the square of the prediction error $\mathbf{w}(n)$ obtained by estimating the present value of the autoregressive process $\mathbf{u}(n)$ as a linear combination of the M previous values $\mathbf{u}(n-1), \ldots, \mathbf{u}(n-M)$. We now use this property to define the autoregressive coefficients a_i^j of a wide-sense stationary stochastic process which is not necessarily an AR process.

Definition 3. *Let $\mathbf{u}(n)$ be a wide-sense stationary stochastic process. We call autoregressive coefficients of order M and denote by $a_1^M, \ldots, a_M^M$ the coefficients which minimize the mean square prediction error, the prediction error associated with the coefficients $a_1^M, \ldots, a_M^M$ being defined as the term $\mathbf{w}^M(n)$ obtained by the following convolution operation:*

$$\sum_{k=0}^{M} a_k^M \mathbf{u}(n-k) = \mathbf{w}^M(n), \tag{2.28}$$

by setting $a_0^M = 1$.

This definition makes it possible to define the autoregressive coefficients of a wide-sense stationary process for any order M.

Since the coefficients of the autoregressive model $a_1^M, \ldots, a_M^M$ are those which minimize the quadratic mean of the prediction error $\mathbf{w}^M(n)$, it is possible to show that $\mathbf{w}^M(n)$ is uncorrelated with the past values $\mathbf{u}(n-k)$ for all $k \geqslant 1$ of the process $\mathbf{u}(n) = \{\mathbf{u}(n)\}_{n\in\mathbb{Z}}$.

2.1.5 Yule-Walker equation

A zero-mean autoregressive process of order M described by Equation (2.16) is entirely determined by the coefficients $a_1^M, \ldots, a_M^M$ of the autoregressive model and by the variance $\sigma^2_{\mathbf{w}^M}$ of the white noise $\mathbf{w}^M(n)$.

In this section, we show that these coefficients are in bijection with the autocorrelation coefficients $r_0, r_1, \ldots, r_M$:

$$\left(a_1^M, \ldots, a_M^M, \sigma^2_{\mathbf{w}^M}\right) \Leftrightarrow (r_0, r_1, \ldots, r_M)\,. \tag{2.29}$$

According to Equation (2.18), we have:

$$\sum_{k=0}^{M} a_k \mathbf{u}(n-k) = \mathbf{w}(n) \quad \forall n \in \mathbb{Z}. \tag{2.30}$$

By multiplying both sides of Equation (2.30) by $\mathbf{u}^*(n-l)$ and then taking the expectation, we obtain:

$$\mathbb{E}\left[\sum_{k=0}^{M} a_k \mathbf{u}(n-k)\mathbf{u}^*(n-l)\right] = \mathbb{E}\left[\mathbf{w}(n)\mathbf{u}^*(n-l)\right] \quad \forall n \in \mathbb{Z}. \tag{2.31}$$

The left-hand side of Equation (2.31) can be expressed using the autocorrelation coefficients:

$$\mathbb{E}\left[\sum_{k=0}^{M} a_k \mathbf{u}(n-k)\mathbf{u}^*(n-l)\right] = \sum_{k=0}^{M} a_k \mathbb{E}\left[\mathbf{u}(n-k)\mathbf{u}^*(n-l)\right] = \sum_{k=0}^{M} a_k r_{l-k}. \tag{2.32}$$

We simplify the right-hand side of Equation (2.31) by observing that

$$\mathbb{E}\left[\mathbf{w}(n)\mathbf{u}^*(n-l)\right] = 0 \quad \forall l \geqslant 1 \tag{2.33}$$

since $\mathbf{u}(n-l)$ involves only samples of white noise up to time $n-l$ which are uncorrelated with the white noise sample $\mathbf{w}(n)$ [44].

Since $a_0 = 1$, we finally obtain from Equations (2.31), (2.32) and (2.33):

$$r_l + \sum_{k=1}^{M} a_k^M r_{l-k} = 0 \tag{2.34}$$

and therefore

$$r_l = -\sum_{k=1}^{M} a_k^M r_{l-k}. \tag{2.35}$$

If we do this operation for all $l \in [\![1, M]\!]$, we obtain M equations which can be written in the matrix form:

$$\begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_M \end{bmatrix} = - \begin{bmatrix} r_0 & r_1^* & \ldots & r_{M-1}^* \\ r_1 & r_0 & \ldots & r_{M-2}^* \\ \vdots & \vdots & \ddots & \vdots \\ r_{M-1} & r_{M-2} & \ldots & r_0 \end{bmatrix} \begin{bmatrix} a_1^M \\ a_2^M \\ \vdots \\ a_M^M \end{bmatrix}. \tag{2.36}$$

This equation is called the Yule-Walker equation [82] [15], it can also be written in the form:

$$\begin{aligned} R_M \widetilde{a}_M &= -\widetilde{r}_M \\ \widetilde{a}_M &= [a_1^M, \ldots, a_M^M]^T, \\ \widetilde{r}_M &= [r_1, \ldots, r_M]^T \end{aligned} \tag{2.37}$$

where R_M corresponds to the matrix $\mathbf{R}$ defined in Equation (2.12).

When the matrix R_M is invertible, we can express the coefficients $\left(a_1^M, \ldots, a_M^M\right)$ as a function of the coefficients $(r_0, r_1, \ldots, r_M)$:

$$\widetilde{a}_M = -R_M^{-1} \widetilde{r}_M. \tag{2.38}$$

Note that this equation is equivalent to the following equation:

$$\widetilde{a}_M = -\left(\frac{1}{r_0}R_M\right)^{-1}\left(\frac{1}{r_0}\widetilde{r}_M\right). \tag{2.39}$$

We denote $\rho_i = \frac{r_i}{r_0}$ for $i \in [\![0, M]\!]$ the normalized correlation coefficients. The equation (2.39) defines a bijection between the normalized autocorrelation coefficients $\rho_1, \rho_2, \ldots, \rho_M$ and the coefficients $a_1, a_2, \ldots, a_M$ of the autoregressive model:

$$(\rho_1, \rho_2, \ldots, \rho_M) \Leftrightarrow \left(a_1^M, a_2^M, \ldots, a_M^M\right). \tag{2.40}$$

We can use Equation (2.39) to compute the autoregressive coefficients $\left(a_1^M, a_2^M, \ldots, a_M^M\right)$ from the normalized autocorrelation coefficients $(p_1, p_2, \ldots, p_M)$. We will see in Section 2.1.6 how to compute the normalized autocorrelation coefficients $(p_1, p_2, \ldots, p_M)$ from the autoregressive coefficients $\left(a_1^M, a_2^M, \ldots, a_M^M\right)$.

We now express the variance of the zero-mean white noise $\sigma^2_{\mathbf{w}^M}$ as a function of the autocorrelation coefficients $r_0, r_1, \ldots, r_M$ and the autoregressive coefficients $a_1^M, a_2^M, \ldots, a_M^M$. By using the definition of an autoregressive process of order M described by the convolution equation (2.28) and by recalling that the zero-mean white noise $\mathbf{w}^M(n)$ is independent of $\mathbf{u}(n-k)$ for all $k \geqslant 1$, we have:

$$\begin{aligned}
\sigma^2_{\mathbf{w}^M} =& \mathbb{E}\left[\mathbf{w}^M(n)\mathbf{w}^{M*}(n)\right] && (2.41)\\
=& \mathbb{E}\left[\left(\sum_{k=0}^{M} a_k^M \mathbf{u}(n-k)\right)\mathbf{w}^{M*}(n)\right] \quad \text{with } a_0^M = 1 && (2.42)\\
=& \mathbb{E}\left[\mathbf{u}(n)\mathbf{w}^{M*}(n)\right] && (2.43)\\
=& \mathbb{E}\left[\mathbf{u}(n)\left(\sum_{k=0}^{M} a_k^M \mathbf{u}(n-k)\right)^*\right] && (2.44)\\
=& \sum_{k=0}^{M} a_k^{M*}\mathbb{E}\left[\mathbf{u}(n)\mathbf{u}^*(n-k)\right] && (2.45)\\
=& \sum_{k=0}^{M} a_k^{M*} r_k. && (2.46)
\end{aligned}$$

We can thus compute the variance of white noise $\sigma^2_{\mathbf{w}^M}$ from the autocorrelation coefficients $r_0, r_1, \ldots, r_M$, the autoregressive coefficients $a_1^M, a_2^M, \ldots, a_M^M$ being of function of the coefficients $r_0, r_1, \ldots, r_M$ according to Equation (2.38). Explicitly, we have:

$$\begin{aligned}
\sigma^2_{\mathbf{w}^M} =& \sum_{k=0}^{M} a_k^{M*} r_k && (2.47)\\
=& r_0 + \sum_{k=1}^{M} a_k^{M*} r_k \quad \text{as } a_0^M = 1 && (2.48)\\
=& r_0 + \widetilde{a}_M^H \widetilde{r}_M && (2.49)\\
=& r_0 + \left(-R_M^{-1}\widetilde{r}_M\right)^H \widetilde{r}_M \quad \text{according to Equation (2.38)} && (2.50)\\
=& r_0 - \widetilde{r}_M^H R_M^{-1} \widetilde{r}_M. && (2.51)
\end{aligned}$$

If we want to compute the autocorrelation coefficients $r_0, r_1, \ldots, r_M$ from the autoregressive coefficients $a_1^M, a_2^M, \ldots, a_M^M$ and the variance of white noise $\sigma^2_{\mathbf{w}^M}$, we can start by computing the normalized autocorrelation coefficients $\rho_1, \rho_2, \ldots, \rho_M$ from the autoregressive coefficients $a_1^M, a_2^M, \ldots, a_M^M$. We present a method to do this in Section 2.1.6. We can then compute the mean quadratic power coefficient r_0 by noticing that Equation (2.46) gives us:

$$\sigma^2_{\mathbf{w}^M} = \sum_{k=0}^{M} a_k^{M*} r_k = \left(\sum_{k=0}^{M} a_k^{M*}\rho_k\right) r_0, \tag{2.52}$$

and therefore:

$$r_0 = \frac{\sigma^2_{\mathbf{w}^M}}{\sum_{k=0}^{M} a_k^{M*}\rho_k}. \tag{2.53}$$

In the proof of Property 26 of Appendix A, we also prove the following relation:

$$\sigma^2_{\mathbf{w}^M} = \left(\prod_{k=1}^{M} \left(1 - |\mu_k|^2 \right) \right) r_0. \tag{2.54}$$

This relation can be used to compute the coefficient r_0 when the variance of the white noise $\sigma^2_{\mathbf{w}^M}$ and the reflection coefficients $\mu_1, \mu_2, \ldots, \mu_M$ are known.

Once the value of the mean quadratic power r_0 is obtained, we just have to notice that $r_i = \rho_i r_0$ to be able to compute the autocorrelation coefficients $r_1, \ldots, r_M$.

2.1.6 Levinson algorithm

In Definition 3, we define the autoregressive coefficients of a wide-sense stationary stochastic process. In general, wide-sense stationary stochastic processes are not autoregressive processes of order M for $M \in \mathbb{N}$. If we want to model a wide-sense stationary process $\mathbf{u}(n)$ as an autoregressive process, then it is useful to estimate the coefficients of the autoregressive model of order M for several values of M, then to choose the order of the model which best corresponds to the studied process $\mathbf{u}(n)$.

When the autocorrelation coefficients $r_0, r_1, \ldots, r_M$ of the studied signal are known, it is possible to compute the coefficients $a_1^n, a_2^n, \ldots, a_n^n$ of the autoregressive model of order $n \in [\![1, M]\!]$ using the Yule-Walker equation. The Yule-Walker equation is indeed equivalent to Equation (2.38) that we give again here:

$$\widetilde{a}_n = -R_n^{-1} \widetilde{r}_n \tag{2.55}$$

with

$$\widetilde{a}_n = [a_1^n, \ldots, a_n^n]^T,$$
$$\widetilde{r}_n = [r_1, \ldots, r_n]^T$$

and R_n the autocorrelation matrix of size $n \times n$.

However, when we want to compute the coefficients $\widetilde{a}_n$ for all $n \in [\![1, M]\!]$, to use Equation (2.55) for all $n \in [\![1, M]\!]$ is not optimal in terms of computation time. It is indeed possible to use the Toeplitz structure of the matrix R_M to iteratively compute the vector coefficients $\widetilde{a}_1, \widetilde{a}_2, \ldots, \widetilde{a}_M$ using the value of the vector $\widetilde{a}_n$ to compute $\widetilde{a}_{n+1}$. It is the principle of the Levinson algorithm 14 detailed in Appendix A. We show in particular that the following relation links the autoregressive coefficients of successive order:

$$a_k^n = a_k^{n-1} + \mu_n a_{n-k}^{n-1*} \quad \forall k \in [\![1, n]\!]. \tag{2.56}$$

Using Equation (2.56), we can compute the autoregressive coefficients a_i^j for all $1 \leqslant i \leqslant j \leqslant n$ from the reflection coefficients $\mu_1, \mu_2, \ldots, \mu_n$, where $\mu_i = a_i^i$. The coefficients a_i^j are then computed iteratively on the order: the coefficients of order j are used for the computation of the coefficients of order $j+1$. The reflection coefficients $\mu_1, \mu_2, \ldots, \mu_n$ therefore entirely determine the autoregressive coefficients a_i^j for all $1 \leqslant i \leqslant j \leqslant n$. In Appendix A, we show that the autoregressive coefficients a_i^j for all $1 \leqslant i \leqslant j \leqslant n$ are also entirely determined by the coefficients $a_1^n, a_2^n, \ldots, a_n^n$ of the autoregressive model of order n. Equation (2.56) is indeed equivalent to the following equation:

$$a_k^{n-1} = \frac{a_k^n - \mu_n a_{n-k}^{n*}}{1 - |\mu_n|^2}. \tag{2.57}$$

This last equation can be used to compute the coefficients of the autoregressive model of order $n-1$ from the coefficients of the autoregressive model of order n, the computation of the autoregressive coefficients is therefore done by descending recurrence on the order of the autoregressive model. We prove Equation (2.57) in the proof of Property 29 of Appendix A.

Finally, the algorithms presented establish the following equivalences:

$$(\mu_1, \mu_2, \ldots, \mu_n) \Leftrightarrow \left(a_i^j \right)_{1 \leqslant i \leqslant j \leqslant n} \Leftrightarrow (a_1^n, a_2^n, \ldots, a_n^n). \tag{2.58}$$

We recall that we have presented in Equation (2.40) of Section 2.1.5 the equivalence between the normalized autocorrelation coefficients $\rho_1, \rho_2, \ldots, \rho_n$ (with $\rho_i = \frac{r_i}{r_0}$) and the AR coefficients of order n:

$$(\rho_1, \rho_2, \ldots, \rho_n) \Leftrightarrow (a_1^n, a_2^n, \ldots, a_n^n). \tag{2.59}$$

We have seen in Equation (2.39) of Section 2.1.5 that we can compute the autoregressive coefficients $a_1^n, a_2^n, \ldots, a_n^n$ from the normalized autocorrelation coefficients $\rho_1, \rho_2, \ldots, \rho_n$ expressing the Yule-Walker equation as follows:

$$\widetilde{a}_n = -\left(\frac{1}{r_0} R_n \right)^{-1} \left(\frac{1}{r_0} \widetilde{r}_n \right) \tag{2.60}$$

with $\widetilde{a}_n = [a_1^n, \dots, a_n^n]^T$.

We now study the inverse transformation, i.e. how to compute the normalized autocorrelation coefficients $\rho_1, \rho_2, \dots, \rho_n$ from the AR coefficients $a_1^n, a_2^n, \dots, a_n^n$. We recall that the coefficients $a_1^n, a_2^n, \dots, a_n^n$ entirely determine the AR coefficients a_i^j for $1 \leqslant i \leqslant j \leqslant n$, the explicit computation of the coefficients a_i^j for $1 \leqslant i \leqslant j \leqslant n-1$ can be obtained by a descending recurrence on the order of the coefficients using Equation (2.57). In Appendix A, we present Algorithm 15 which performs the inverse operation than the Levinson algorithm: this algorithm can be used to compute the autocorrelation coefficients $r_0, r_1, \dots, r_n$ from the mean quadratic power coefficient $p_0 = r_0$ and the AR coefficients a_i^j with $1 \leqslant i \leqslant j \leqslant n$. If we use Algorithm 15 by setting $p_0 = 1$, we can compute the normalized autocorrelation coefficients $\rho_1, \rho_2, \dots, \rho_n$ from the autoregressive coefficients a_i^j with $1 \leqslant i \leqslant j \leqslant n$.

Note that the Toeplitz Hermitian Positive Definite matrix R_n is entirely determined by the autocorrelation coefficients $r_0, r_1, \dots, r_{n-1}$. We show in Appendix A that the reflection coefficients $\mu_1, \mu_2, \dots, \mu_{n-1}$ where $\mu_k = a_i^i$ entirely determine the AR coefficients a_i^j for all $1 \leqslant i \leqslant j \leqslant n-1$. From Equation (2.27), we have $|\mu_i| < 1$ for all $i \in \mathbb{N}^*$. This inequality is also proven in the proof of Property 27 of Appendix A. The Levinson algorithm therefore defines a bijection between the following spaces:

$$\begin{aligned} \mathcal{T}_n^+ &\to \mathbb{R}_+^* \times \mathcal{D}^{n-1} \\ R_n &\mapsto (p_0, \mu_1, \mu_2, \dots, \mu_{n-1}) \end{aligned} \tag{2.61}$$

where $\mathcal{T}_n^+$ represents the set of Toeplitz Hermitian positive definite matrices and $\mathcal{D}$ represents the open complex unit disk. This bijection will be used in Chapter 3 to build a Riemannian metric on the product space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ inspired by a metric defined on the space $\mathcal{H}_n^+$ coming from information geometry, where $\mathcal{H}_n^+$ represents the space of Hermitian positive definite matrices (note that $\mathcal{T}_n^+ \subset \mathcal{H}_n^+$). The Riemannian manifold obtained, denoted $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$, will then be used to represent unidimensional complex stationary centered Gaussian autoregressive time series.

As seen in Equation (2.61), there is a one-to-one correspondence between a Toeplitz HPD matrix $R_n \in \mathcal{T}_n^+$ and the coefficients $(p_0, \mu_1, \mu_2, \dots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$. In next section, we will see that a Toeplitz HPD matrix and its corresponding reflection coefficients are closely related to a probability measure on the complex unit circle.

2.1.7 Orthogonal polynomials on the unit circle

As explained in the article of Ben Jeuris and Raf Vandebril [45], a one-to-one correspondence exists between a Toeplitz HPD matrix and a probability measure on the complex unit circle, where the coefficients of the Toeplitz matrix are found as the moments (or Fourier coefficients) of the corresponding probability measure [27, 49, 55]. The concept of orthogonality for polynomials on the unit circle is linked to the probability measure on the complex unit circle considered, and therefore indirectly to the related Toeplitz matrix. Finally, the computation of an orthonormal basis of polynomials on the unit circle can be performed using the Szegő's recursion [76, 77], in which the Verblunsky coefficients arise. It turns out that these coefficients are equal to the reflection coefficients μ_l appearing in Equation (2.61).

In section 2.1.6, we refered to the Levinson algorithm 13 to efficiently compute the AR coefficients a_i^j for $1 \leqslant i \leqslant j \leqslant M$ from the autocorrelation coefficients $r_0, r_1, \dots, r_M$. In Section 2.1.8, we give a method to estimate the AR coefficients of a stochastic process from an observed time series $u(0), u(1), \dots, u(n)$. In Section 2.1.9, we give two methods to determine the order of the AR model which best corresponds (in a sense to be defined) to the time series $u(0), u(1), \dots, u(n)$ observed.

2.1.8 Autoregressive coefficients estimation

In section 2.1.6, we presented the Levinson algorithm which can be used to compute the autoregressive coefficients a_i^j for $1 \leqslant i \leqslant j \leqslant n$ from the autocorrelation coefficients $r_0, r_1, \dots, r_n$. In practice, the autocorrelation coefficients $r_0, r_1, \dots, r_n$ of an observed time series $u(0), u(1), \dots, u(M-1)$ of length M are generally not known. In this case, it is possible to estimate the autoregressive coefficients a_i^j pour $1 \leqslant i \leqslant j \leqslant n$ by starting by estimating the autocorrelation coefficients $r_0, r_1, \dots, r_n$, then using the Levinson algorithm 14.

The autocorrelation coefficients are generally estimated by empirical mean:

$$\widehat{r}_i = \frac{1}{M-i} \sum_{k=0}^{M-i-1} u(k+i)u^*(k). \tag{2.62}$$

However, this method is not very precise when the integer $(M-i)$ is small. Moreover, this method does not guarantee that the Toeplitz Hermitian matrix $\widehat{R}_{n+1}$ constructed from the autocorrelation coefficients $\widehat{r}_0, \widehat{r}_1, \dots, \widehat{r}_n$ is positive definite. In the case where the estimated matrix $\widehat{R}_{n+1}$ is not positive definite, at least one of the reflection coefficients $\widehat{\mu}_1, \widehat{\mu}_2, \dots, \widehat{\mu}_n$ obtained using the Levinson algorithm does not belong to the complex unit disk $\mathcal{D}$, the AR coefficients estimated are therefore not those of a stationary stochastic process.

To estimate the AR coefficients of a stochastic process $\mathbf{u}(n)$ from a realization of a time series $u(0), u(1), \ldots, u(n-1)$, we propose to use another method: the Burg algorithm. We devote Appendix C to the description of the Burg algorithm 16. The Burg algorithm estimates the AR coefficients iteratively on the order by taking advantage of the relation between the AR coefficients of successive order:

$$a_k^n = a_k^{n-1} + \mu_n a_{n-k}^{n-1*} \quad \forall k \in [\![1, n]\!]. \tag{2.63}$$

We show in Property 38 of Appendix C that the reflection coefficients μ_i estimated by the Burg algorithm belong to the complex unit disk $\mathcal{D}$: we have $|\mu_i| < 1$ for all $i \in \mathbb{N}^*$. We also present a Burg algorithm with a regularization parameter penalizing irregular spectra: the regularized Burg algorithm 17.

2.1.9 Selecting model order

An autoregressive process can be used to generate data or to analyze existing data [44]. When we generate the data, we choose the order M of the process to be generated. When we use an autoregressive model to analyze existing data, we generally need to estimate the order of the autoregressive model M that best matches the data to be analyzed.

In this section, we present two methods to estimate the order of an autoregressive model. The first method is called *AIC* and was developed by Akaike (1973) [2]. The second method is called *MDL* and was developed by Rissanen (1978) [65] and Schwartz (1978). These two methods use arguments coming from information geometry.

Let $u_1, u_2, \ldots, u_N$ be the data observed during N independent realizations of a discrete-time stationary stochastic process $\mathbf{u}(n)$ and denote by $g(u_i)$ the probability density of u_i. We denote by $f_{\mathbf{u}}\left(u_i \mid \widehat{\theta}_m\right)$ the conditional probability density of u_i with respect to the parameter $\widehat{\theta}_m$, where $\widehat{\theta}_m$ represents the vector of estimated parameters which model the studied stochastic process. The integer m represents the order of the model. We can therefore write:

$$\widehat{\theta}_m = \left[\widehat{\theta}_1^m, \widehat{\theta}_2^m, \ldots, \widehat{\theta}_m^m\right]^T. \tag{2.64}$$

In this thesis, we could choose $\widehat{\theta}_m = [\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{m-1}]^T$. We could also choose $\widehat{\theta}_m = [\widehat{r}_0, \widehat{r}_1, \ldots, \widehat{r}_{m-1}]^T$ or any equivalent set of parameters presented in the previous sections.

The studied stochastic process can be represented by several models of different orders. To choose the order m of the model best suited to the stochastic process $\mathbf{u}(n)$, the *information-theoretic criterion* (AIC) proposed by Akaike is the following: we choose to represent the stochastic process $\mathbf{u}(n)$ by the model of parameter $\widehat{\theta}_m$ for which the function

$$\text{AIC}(m) = -2L\left(\widehat{\theta}_m\right) + 2m \tag{2.65}$$

reaches its minimum, where the function $L\left(\widehat{\theta}_m\right)$ is the logarithm of the pseudo-maximum likelihood estimator [64] of the model parameters:

$$L\left(\widehat{\theta}_m\right) = \max_{\widehat{\theta}_m} \sum_{i=1}^{N} \ln\left(f_{\mathbf{u}}\left(u_i \mid \widehat{\theta}_m\right)\right). \tag{2.66}$$

The term $L\left(\widehat{\theta}_m\right)$ described in Equation (2.66) is obtained by minimizing the Kullback-Leibler divergence between the true density of unknown probability $g(u)$ and conditional probability density $f_{\mathbf{u}}\left(u_i \mid \widehat{\theta}_m\right)$. We recall the expression of the Kullback-Leibler divergence between these two functions:

$$D_{g\|f}\left(\widehat{\theta}_m\right) = \int_{-\infty}^{\infty} g(u) \ln\left(g\left(u\right)\right) du - \int_{-\infty}^{\infty} g(u) \ln\left(f_{\mathbf{u}}\left(u \mid \widehat{\theta}_m\right)\right) du. \tag{2.67}$$

In this thesis, we refer to the Burg algorithm [16] described in Appendix C to estimate the model parameter $\widehat{\theta}_m = [\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{m-1}]^T$. However, the Burg algorithm is not a maximum likelihood estimator of the autoregressive coefficients, an exact forward-backward maximum likelihood autoregressive parameter estimation method is given in [5] for Gaussian autoregressive time series.

According to Equation (2.65), the function $AIC(m)$ is a sum of two terms. The first term, $-2L\left(\widehat{\theta}_m\right)$ tends to decrease rapidly as the order m of the model increases. The second term, $2m$, represents a *model complexity penalty* which makes the function $AIC(m)$ an estimator of the Kullback-Leibler divergence. This second term grows linearly with respect to the order m of the model. When we plot the graph of the function $AIC(m)$ as a function of the model order m we generally observe a unique minimum for a value m which will be designated as *optimum order* of the model and noted MAIC for *Minimum AIC* [50].

We now present a second criterion that can be used to choose the optimal order of the model of a stochastic process $\mathbf{u}(n)$. This second criterion proposed by Rissanen in 1978 [65], 1986 [66] and 1989 [67] is based on the idea of exactly

describing observed data by coding them with a minimum of binary digits. Indeed, a model can be seen as a tool to capture the characteristics and constraints of the observed data and therefore to reduce the number of digits necessary to describe the data. Rissanen then proposes to use the number of digits necessary to code the observed data by exploiting the constraints induced by the model as well as the number of digits necessary to describe the model as a criterion to determine the quality of the model.

We can state Rissanen's *minimum description length (MDL) criterion* as follows [44]: given a data set of interest and a family of competing statistical models, the best model is the one that provides the shortest description length for the data.

Mathematically, to determine the optimal number of parameters of the model, the MDL criterion proposed by Rissanen is the following: we choose the integer m for which the following function of $\widehat{\theta}_m$:

$$\text{MDL}(m) = -L\left(\widehat{\theta}_m\right) + \frac{1}{2} m \ln(N) \tag{2.68}$$

reaches its minimum.

2.1.10 Complex Gaussian processes

When we study a wide-sense stationary autoregressive stochastic process of order M:

$$\mathbf{u}(n) + a_1\mathbf{u}(n-1) + \ldots + a_M\mathbf{u}(n-M) = \mathbf{w}(n),$$

the white noise $\mathbf{w}(n)$ is generally assumed to be Gaussian. The autoregressive stochastic process $\mathbf{u}(n)$ is then called a *Gaussian* process. In this section, we start by recalling the definition of Gaussian stochastic processes with complex values, then we study their properties.

We first recall the definition of a multivariate complex distribution.

Definition 4. *A complex random vector Z is said to be Gaussian when the real random vector $(Re(Z), Im(Z))^T$ is Gaussian.*

Definition 5. *A time continuous complex stochastic process $\{X_t\}_{t\in T}$ is said to be Gaussian if for all finite number $k \in \mathbb{N}^*$ of time indices $t_1, \ldots, t_k \in T$, the random variable:*

$$X_{t_1,\ldots,t_k} = (X_{t_1}, \ldots, X_{t_k})$$

is a complex multivariate Gaussian random variable. This is equivalent to say that every linear combination of $(X_{t_1}, \ldots, X_{t_k})$ has a univariate complex Gaussian distribution.

A multivariate Gaussian complex distribution is characterized by its moments of order one and two.

Property 5. *A multivariate complex Gaussian distribution Z is characterized by:*

- *its mean vector $\mu = \mathbb{E}[Z]$,*
- *its covariance matrix $\Gamma = \mathbb{E}\left[(Z-\mu)(Z-\mu)^H\right]$,*
- *its relation matrix $C = \mathbb{E}\left[(Z-\mu)(Z-\mu)^T\right]$.*

Its probability density function is then defined by [60]:

$$f(z) = \frac{1}{\pi^n \sqrt{\det(\Gamma)\det\left(\bar{\Gamma} - C^H\Gamma^{-1}C\right)}} \exp\left(-\frac{1}{2}\left((\bar{z}-\bar{\mu})^T (z-\mu)^T\right)\begin{pmatrix}\Gamma & C\\ \bar{C} & \bar{\Gamma}\end{pmatrix}^{-1}\begin{pmatrix}z-\mu\\ \bar{z}-\bar{\mu}\end{pmatrix}\right) \tag{2.69}$$

We denote by $Z \sim \mathcal{CN}(\mu, \Gamma, C)$ a complex Gaussian vector of mean μ, of covariance matrix Γ and relation matrix C.

We now come back to the time continuous complex Gaussian stochastic processes described in Definition 5 and give a first property of these processes.

Property 6. *A wide-sense stationary Gaussian process is strict-sense stationary [7, 44].*

The wide-sense stationarity being equivalent to the strict-sense stationarity for Gaussian processes, we might omit to specify the kind of stationarity (wide-sense or strict-sense) for Gaussian processes and simply call them stationary Gaussian processes.

We now detail a particular case of complex Gaussian processes: circularly-symmetric central complex Gaussian distributions.

Definition 6. *A complex random vector Z is called circularly-symmetric if for every deterministic $\phi \in]-\pi, \pi]$, the distribution of $e^{i\phi}Z$ equals the distribution of Z.*

We now give an important property of the mean μ and of the relation matrix C of circularly-symmetric central complex Gaussian distributions.

Property 7. *The circularly-symmetric central complex Gaussian distributions are the multivariate complex Gaussian distributions of zero mean ($\mu = 0$) and zero relation matrix ($C = 0$). The circularly-symmetric central complex Gaussian distributions are therefore entirely determined by their covariance matrices Γ.*

We can now define the circularly-symmetric centered complex Gaussian stochastic processes.

Definition 7. *We call circularly-symmetric centered complex Gaussian stochastic processes the processes $\{X_t\}_{t\in T}$ for which:*

$$(X_{t_1}, \ldots, X_{t_k}) \sim \mathcal{CN}(0, \Gamma, 0) \quad \forall k \in \mathbb{N}^*, \forall t_1, \ldots, t_k \in T. \tag{2.70}$$

We now study the circularly-symmetric centered complex Gaussian stochastic processes which are stationary. Let $\mathbf{u}(n)$ be such a process of length N. The process $\mathbf{u}(n)$ therefore has the following characteristics:

- its mean is zero: $\mu = \mathbb{E}[\mathbf{u}(n)] = 0$,
- since the process $\mathbf{u}(n)$ is wide-sense stationary, its autocorrelation function is of the form:

$$r_k = \mathbb{E}[\mathbf{u}(n)\mathbf{u}^*(n-k)] \quad \forall k \in [\![0, N-1]\!],$$

- its relation function is zero: $\mathbb{E}[\mathbf{u}(n)\mathbf{u}(n-k)] = 0 \quad \forall k \in [\![0, N-1]\!]$.

The autocorrelation coefficients $r_0, r_1, \ldots, r_{N-1}$ determine the correlation matrix $\mathbf{R}$ of the Gaussian process $\mathbf{u}(n)$. We recall that the covariance matrix Γ coincide with the autocorrelation matrix $\mathbf{R}$ in the case of zero-mean stochastic processes according to the definition of the autocorrelation and autocovariance coefficients given in Equations (2.3) and (2.4). We denote in general $\mathcal{N}(0, \mathbf{R})$ the Gaussian vector of zero mean, of correlation matrix $\mathbf{R}$ and of zero relation matrix. The joint probability density of the N samples of the process $\mathbf{u}(n)$ is then defined by:

$$f_{\mathbf{u}}(U) = \frac{1}{\pi^N \det(\mathbf{R})} \exp\left(-U^H \mathbf{R}^{-1} U\right), \tag{2.71}$$

where

$$U = [u(1), u(2), \ldots, u(N)]^T$$

is an N-by-1 data vector. The matrix $\mathbf{R}$ is then the N-by-N Toeplitz Hermitian Positive Definite matrix of the second order moments defined in terms of $\mathbf{u}$ by $\mathbf{R} = \mathbb{E}\left[UU^H\right]$.

We now give an important property about the moments of circularly-symmetric centered stationary Gaussian processes [44].

Property 8. *We denote by $u_n = u(n), n \in [\![1, N]\!]$ the observations of a circularly-symmetric centered stationary Gaussian process. According to Reed (1962):*

- *If $k \neq l$, then:*

$$\mathbb{E}\left[\mathbf{u}^*_{s_1}\mathbf{u}^*_{s_2}\ldots\mathbf{u}^*_{s_k}\mathbf{u}_{t_1}\mathbf{u}_{t_2}\ldots\mathbf{u}_{t_l}\right] = 0, \tag{2.72}$$

  where s_i and t_j are integers selected in the set $\{1, 2, \ldots, N\}$.

- *If $k = l$, then*

$$\mathbb{E}\left[\mathbf{u}^*_{s_1}\mathbf{u}^*_{s_2}\ldots\mathbf{u}^*_{s_l}\mathbf{u}_{t_1}\mathbf{u}_{t_2}\ldots\mathbf{u}_{t_l}\right] = \sum_{\sigma\in S_l}\prod_{i=1}^{l}\mathbb{E}\left[\mathbf{u}^*_{s_{\sigma(i)}}\mathbf{u}_{t_i}\right] \tag{2.73}$$

  where S_l represents the set of permutations of the set $[\![1, l]\!]$. The term on the right side of Equation (2.73) is therefore a sum of $l!$ terms which are products of moments of order 2. Equation (2.73) is called the Gaussian moment-factoring theorem.

2.1.11 Power spectral density

The autocorrelation function of a wide-sense stationary centered discrete-time stochastic process $\mathbf{u}(n)$ is defined in Section 2.1.2 by:

$$r(k) = \mathbb{E}\left[\mathbf{u}(n+k)\mathbf{u}^*(n)\right]. \tag{2.74}$$

This function describes the moments of order 2 of the stochastic process $\mathbf{u}(n)$ in the time domain. The statistical parameter associated with the autocorrelation function in the frequency domain is the *power spectral density*, which is also called *power spectrum* or also *spectrum* [44]. The *power spectral density* is a widely used tool to describe time series. We will recall its definition and its fundamental properties in this section.

Consider a wide-sense stationary centered discrete-time stochastic process $\mathbf{u}(n)$ and denote by $r(l)$ its autocorrelation function defined for all $l \in \mathbb{Z}$. In the following, we denote by $u(n)$ with $n \in \mathbb{Z}$ the time series of infinite length representing a unique realization of the studied stochastic process. We start by focusing on a *windowed* portion of this time series by defining the auxiliary time series:

$$u_N(n) = \begin{cases} u(n), & n \in [\![-N, N]\!] \\ 0, & |n| > N \end{cases} \tag{2.75}$$

The length $2N+1$ of the window will then tend towards infinity. By definition, the *discrete-time Fourier transform* of the time series $U_N(n)$ is given by:

$$U_N(\omega) = \sum_{n=-N}^{N} u_N(n) e^{-j\omega n} \tag{2.76}$$

where $\omega \in]-\pi, \pi]$ is the *angular frequency*. We denote by $\mathbf{U}_N(\omega)$ the random variable associated with the stochastic process $\mathbf{u}(n)$ whose realization $U_N(\omega)$ corresponds to the time series $u(n)$. In general, the term $U_N(\omega)$ is complex-valued. Its complex conjugate has the following expression:

$$U_N^*(\omega) = \sum_{k=-N}^{N} u_N^*(k) e^{j\omega k}. \tag{2.77}$$

By multiplying Equations (2.76) and (2.77), we obtain the expression of the square modulus of $U_N(\omega)$:

$$|U_N(\omega)|^2 = \sum_{n=-N}^{N} \sum_{k=-N}^{N} u_N(n) u_N^*(k) e^{-j\omega(n-k)}. \tag{2.78}$$

Equation (2.78) is obtained for each realization $u(n)$ of the studied stochastic process. By taking the expectation on each side of Equation (2.78), we obtain:

$$\mathbb{E}\left[|\mathbf{U}_N(\omega)|^2\right] = \sum_{n=-N}^{N} \sum_{k=-N}^{N} \mathbb{E}\left[\mathbf{u}_N(n)\mathbf{u}_N^*(k)\right] e^{-j\omega(n-k)}. \tag{2.79}$$

where the symbol $\mathbb{E}$ denotes the expectation with respect to an infinity of independent realizations of the stochastic process studied.

Note that we have:

$$\mathbb{E}\left[\mathbf{u}_N(n)\mathbf{u}_N^*(k)\right] = \begin{cases} \mathbb{E}\left[\mathbf{u}(n)\mathbf{u}^*(k)\right] = r(n-k) & for -N \leqslant (n,k) \leqslant N \\ 0 & otherwise \end{cases} \tag{2.80}$$

We can therefore rewrite Equation (2.79) in the form:

$$\mathbb{E}\left[|\mathbf{U}_N(\omega)|^2\right] = \sum_{n=-N}^{N} \sum_{k=-N}^{N} r\left(n-k\right) e^{-j\omega(n-k)}. \tag{2.81}$$

By setting $l = n - k$, we obtain:

$$\frac{1}{N}\mathbb{E}\left[|\mathbf{U}_N(\omega)|^2\right] = \sum_{l=-N}^{N} \left(1 - \frac{|l|}{N}\right) r(l) e^{-j\omega l}. \tag{2.82}$$

When the following equation holds

$$\lim_{N\to\infty} \frac{1}{2N+1} \sum_{l=-N+1}^{N-1} |l|\, r(l) e^{-j\omega l} = 0, \tag{2.83}$$

then we have [44]

$$\lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[|\mathbf{U}_N(\omega)|^2\right] = \sum_{l=-\infty}^{\infty} r(l)e^{-j\omega l}. \tag{2.84}$$

The term on the right side of this last equation can be seen as the Fourier transform of the autocorrelation coefficients, as we will see in Property 9. We can then define the function:

$$S(\omega) = \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[|\mathbf{U}_N(\omega)|^2\right]. \tag{2.85}$$

The term $|U_N(\omega)|^2/N$ is called *periodogram* of the windowed time series $u_N(n)$. Equation (2.85) means that the periodogram converges towards $S(\omega)$ in mean, and not in square mean or any other form of convergence in general.

As mentioned in [44], when the limit presented in Equation (2.85) exists, the quantity $S(\omega)$ has the following interpretation (Priestley, 1981): the term $S(\omega)d\omega$ is the average of the contribution to the total power from components of a wide-sense stationary stochastic process with angular frequencies located between ω and $\omega + d\omega$, the average is taken over all possible realizations of the process.

Therefore, the quantity $S(\omega)$ is the "spectral density of expected power" of the process, which is abbreviated as *power spectral density*.

Using the equality described in Equation (2.84), we can rewrite the definition of the power spectral density given by Equation (2.85) as

$$S(\omega) = \sum_{l=-\infty}^{\infty} r(l)e^{-j\omega l}. \tag{2.86}$$

To summarize, the power spectral density is defined directly from a wide-sense stationary stochastic process in Equation (2.85). The power spectral density can also be defined as the Fourier transform of the autocorrelation coefficients as shown in Equation (2.86). These two points of view will now be used to present the fundamental properties of the power spectral density.

We start by giving a first property directly related to Equation (2.86).

Property 9. *The autocorrelation function and the power spectral density of a wide-sense stationary stochastic process form a Fourier transform pair:*

$$S(\omega) = \sum_{l=-\infty}^{\infty} r(l)e^{-j\omega l}, \quad -\pi < \omega \leqslant \pi \tag{2.87}$$

and

$$r(l) = \frac{1}{2\pi}\int_{-\pi}^{\pi} S(\omega)e^{j\omega l}d\omega, \quad l \in \mathbb{Z}. \tag{2.88}$$

Equation (2.87) means that the power spectral density is the discrete Fourier transform of the autocorrelation function. Equation (2.88) means that the autocorrelation function is the inverse of the discrete Fourier transform of the power spectral density. This equation comes from the formula for inverting discrete Fourier transforms. This pair of equations is called the *Einstein-Wiener-Khintchine* relation.

Property 10. *The frequency support of the power spectral density $S(\omega)$ is the Nyquist interval $-\pi < \omega \leqslant \pi$.*

Outside this interval, the power spectral density is 2π-periodic:

$$S(\omega + 2k\pi) = S(\omega) \quad \forall k \in \mathbb{Z}. \tag{2.89}$$

Property 11. *The power spectral density of a stationary discrete-time stochastic process is real.*

We can indeed arrange the sum defining the power spectral density in Equation (2.86) to obtain [44]:

$$S(\omega) = r(0) + 2\sum_{k=1}^{\infty} \mathrm{Re}\left[r(k)e^{-j\omega k}\right] \tag{2.90}$$

where Re represents the real part operator. Using the definition of the power spectral density given in Equation (2.85):

$$S(\omega) = \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[|\mathbf{U}_N(\omega)|^2\right], \tag{2.91}$$

we can also notice that the power spectral density is the limit of a series of real numbers and is therefore real.

Property 12. *The power spectral density of a real-valued stationary discrete-time stochastic process is even (i.e. $S(\omega) = S(-\omega)$). If the process is complex valued, its power spectral density is not necessarily even.*

Property 13. *The mean-square value $r(0)$ of a stationary discrete-time stochastic process equals the mean value of the power spectral density $S(\omega)$ over the interval $]-\pi,\pi]$:*

$$r(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} S(\omega)d\omega \tag{2.92}$$

This property is obtained directly by evaluating Equation (2.86) at $\omega = 0$.

Property 14. *The power spectral density of a stationary discrete-time stochastic process is nonnegative:*

$$S(\omega) \geqslant 0 \quad \forall\omega. \tag{2.93}$$

This property follows directly from the definition of the power spectral density $S(\omega)$ given in Equation (2.85):

$$S(\omega) = \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left[\left|\mathbf{U}_N(\omega)\right|^2\right]. \tag{2.94}$$

We now define other spectra associated with the AR coefficients.

2.1.12 Autoregressive spectrum

We define the autoregressive spectrum of the complex centered wide-sense stationary process $\mathbf{u}(n)$ of autoregressive coefficients $a_1^{n-1},\ldots,a_{n-1}^{n-1}$ as in [15]:

$$S_{AR}^{(n-1)}(f) = \frac{P_{n-1}}{\left|\sum_{k=0}^{(n-1)} a_k^{(n-1)} e^{-i2\pi kf}\right|^2} \tag{2.95}$$

where $P_{n-1} \in \mathbb{R}_+^*$ is the variance of the prediction error $w^{n-1}(k)$ of the linear autoregressive model presented in Equation (2.16).

2.1.13 Capon spectrum

The Capon spectrum is defined as the harmonic mean of the autoregressive spectra of orders $k = 1,\ldots,n-1$:

$$S_{Capon}^{(n-1)}(f)^{-1} = \frac{1}{n-1}\sum_{k=1}^{n-1} S_{AR}^{(k)}(f)^{-1}. \tag{2.96}$$

2.1.14 Example of stationary centered complex Gaussian autoregressive continuous stochastic processes with Gaussian spectrum shape

We will prove in Appendix B that the complex autoregressive stationary continuous stochastic processes which power spectral density S_f has the shape of Gaussian distribution of mean m and variance σ^2 with a power coefficient P, *i.e.*:

$$S_f(\xi) = P\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\xi-m)^2}{2\sigma^2}} \tag{2.97}$$

has the following theoretical autocorrelation coefficients :

$$r(k) = Pe^{i2\pi mk} e^{-2\pi^2\sigma^2k^2}. \tag{2.98}$$

This coefficient is obtained computing the continuous-time Fourier transform of the power spectral density $S_f(\xi)$. Thanks to the Levinson algorithm, we also prove in Appendix B that the reflection coefficients μ_k for $k \geqslant 1$ of the corresponding signal have the following expression:

$$\mu_k = (-1)^k e^{i2\pi mk} e^{-2\pi^2\sigma^2 k} \quad \forall k \geqslant 1 \tag{2.99}$$

Note that in this particular case, we have $\mu_k = \mu_1^k$. Note also that the argument of the coefficient μ_k depends only on the mean m and its modulus depends only on the variance σ^2.

2.2 Multidimensional stationary centered complex Gaussian autoregressive time series

In this section, we define multidimensional complex stationary centered Gaussian autoregressive time series. We will generalize the several notions presented for one-dimensional time series in Section 2.1 to multidimensional time series. This section is mainly inspired by the work of Ben Jeuris and Raf Vandebril [45].

2.2.1 Stochastic process

The definition of stochastic processes given in Section 2.1.1 is also available for multidimensional stochastic processes, we therefore refer to this section for a more complete description of stochastic processes. We recall that a stochastic process is not the realization of a single function of time: it is in theory an infinite number of different realizations of the process. A particular realization of a discrete-time stochastic process is called a *time series*. In this section, we will use the capital letters $U(n-M), \ldots, U(n-1), U(n)$ to represent the multidimensional time series of length $M+1$ made up of the present observation $U(n)$ and of the M previous observations done at times $n-1, \ldots, n-M$. Each observation $U(i)$ is a complex vector of dimension N by 1. We will use to bold letter $\mathbf{U}(n) = \{\mathbf{U}(n)\}_{n\in\mathbb{Z}}$ to denote the stochastic process and we will denote by $U(n)$ a realization of this process, i.e. a time series.

2.2.2 Stationarity

The definition of the stationarity given for multidimensional stochastic processes in this section is the same as the definition given in Section 2.1.2 for unidimensional stochastic processes by replacing the lowercase letters u, r and c by the capital letters U, R and C. The conjugate operator $*$ is replaced by the conjugate transpose operator H.

As explained in Section 2.1.2, a stochastic process is strictly stationary if its statistical properties are invariant by temporal translation. Formally, let $\mathbf{U}_t$ be a continuous-time stochastic process of dimension N and let $f_{\mathbf{U}}$ be its joint probability density at times $t_1+\tau, \ldots, t_n+\tau$. The stochastic process $\mathbf{U}_t$ is said to be strictly stationary when:

$$f_{\mathbf{U}}\left(U_{t_1+\tau}, \ldots, U_{t_n+\tau}\right) = f_{\mathbf{U}}\left(U_{t_1}, \ldots, U_{t_n}\right) \quad \forall\tau, \forall n \in \mathbb{N}, t_1, \ldots, t_n \in \mathbb{R}. \tag{2.100}$$

In practice, it is generally impossible to determine the joint probability distribution of a stochastic process from discrete observations of this process. In general, we will therefore content ourselves with a partial characterization of this process from its moments of order one and two.

We define the *mean* value of the process $\mathbf{U}(n)$ by:

$$\mu(n) = \mathbb{E}\left[\mathbf{U}(n)\right] \tag{2.101}$$

where $\mathbb{E}$ denotes the statistical expectation operator.

We define the *autocorrelation function* of the process $\mathbf{U}(n)$ by:

$$R\left(n, n-k\right) = \mathbb{E}\left[\mathbf{U}(n)\mathbf{U}^H(n-k)\right] \quad \forall k \in \mathbb{Z}. \tag{2.102}$$

We define the *autocovariance function* of the process $\mathbf{U}(n)$ by:

$$C\left(n, n-k\right) = \mathbb{E}\left[\left(\mathbf{U}(n) - \mu(n)\right)\left(\mathbf{U}(n-k) - \mu(n-k)\right)^H\right] \quad \forall k \in \mathbb{Z}. \tag{2.103}$$

According to Equations (2.101), (2.102) and (2.103), the mean value $\mu(n)$, the autocorrelation function $R\left(n, n-k\right)$ and the autocovariance function $C\left(n, n-k\right)$ are related by the following equation:

$$C\left(n, n-k\right) = R\left(n, n-k\right) - \mu(n)\mu^H(n-k). \tag{2.104}$$

For a partial characterization of a stochastic process by its moments of order one and two, we just need to determine the mean $\mu(n)$ and the autocorrelation function $R\left(n, n-k\right)$ or to determine the mean $\mu(n)$ and the autocovariance function $C\left(n, n-k\right)$. Note that for stochastic processes with zero mean ($\mu(n) = 0$ for all $n \in \mathbb{Z}$), the autocorrelation function $R\left(n, n-k\right)$ is equal to the autocovariance function $C\left(n, n-k\right)$.

In the case of wide-sense stationary time series, the mean value $\mu(n)$, the autocorrelation function $R\left(n, n-k\right)$ and the autocovariance function $C\left(n, n-k\right)$ have simpler expressions. The mean value function is indeed constant:

$$\mu(n) = \mu \quad \forall n \in \mathbb{Z}. \tag{2.105}$$

The autocorrelation and autocovariance functions only depend on the difference k between the time of observation n and the time of observation $n-k$, that is:

$$R\left(n, n-k\right) = R(k) \tag{2.106}$$

and

$$C\left(n, n-k\right) = C(k). \tag{2.107}$$

The equations (2.105), (2.106) and (2.107) are not sufficient to guarantee that a stochastic process is stationary in the strict sense. We say that a stochastic process is *weakly stationary* or *wide-sense stationary* when it satisfies Equations (2.105), (2.106) and (2.107).

In the following sections, we will focus more specifically on wide-sense stationary processes of zero mean: $\mu = 0$. The partial second-order characterization of these processes is then entirely determined by the value of the autocorrelation coefficients $R(k)$. We will now define the autocorrelation matrix of a multidimensional wide-sense stationary stochastic process from the autocorrelation coefficients, then we will study the properties of this matrix.

2.2.3 Correlation matrix

Let $\mathbf{U}(n) = \{\mathbf{U}(n)\}_{n\in\mathbb{Z}}$ denote a zero-mean wide-sense stationary stochastic process of dimension N, each term $\mathbf{U}(i)$ being a complex vector of dimension N by 1. We denote by $\mathbf{U}_M(n)$ the random vector defined by the vector of dimension $M \times N$ by 1:

$$\mathbf{U}_M(n) = \left[\mathbf{U}(n-M+1)^T, \ldots, \mathbf{U}(n-1)^T, \mathbf{U}(n)^T\right]^T, \tag{2.108}$$

where the superscript T denotes the matrix transposition. We define the correlation matrix of size M of the stochastic process $\mathbf{U}(n)$ by:

$$\mathbf{R}_M = \mathbb{E}\left[\mathbf{U}_M(n)\mathbf{U}_M^H(n)\right]. \tag{2.109}$$

The autocorrelation matrix $\mathbf{R}_M$ will simply be denoted $\mathbf{R}$ when the dimension of the matrix is not ambiguous.

By replacing $\mathbf{U}_M(n)$ in Equation (2.109) by its expression in Equation (2.108), we find using the wide-sense stationarity of $\mathbf{U}(n)$:

$$\mathbf{R} = \begin{bmatrix} R_0 & R_{-1} & \ldots & R_{-M+1} \\ R_1 & R_0 & \ldots & R_{-M+2} \\ \vdots & \vdots & \ddots & \vdots \\ R_{M-1} & R_{M-2} & \ldots & R_0 \end{bmatrix}. \tag{2.110}$$

The coefficient R_0 on the main diagonal is always a Hermitian positive semidefinite matrix.

The correlation matrix $\mathbf{R}$ plays a fundamental role in the study of multidimensional wide-sense stationary stochastic processes. We now study different properties of the matrix $\mathbf{R}$.

Property 15. *The correlation matrix of a wide-sense stationary discrete-time stochastic process is Hermitian, i.e.* $\mathbf{R} = \mathbf{R}^H$.

To prove this property, it suffices to notice that $R_{-k} = R_k^H$.

The correlation matrix $\mathbf{R}$ is therefore entirely determined by the autocorrelation coefficients R_k for $k = 0, 1, \ldots, M-1$. Indeed, we have:

$$\mathbf{R} = \begin{bmatrix} R_0 & R_1^H & \ldots & R_{M-1}^H \\ R_1 & R_0 & \ldots & R_{M-2}^H \\ \vdots & \vdots & \ddots & \vdots \\ R_{M-1} & R_{M-2} & \ldots & R_0 \end{bmatrix}. \tag{2.111}$$

We say that a square matrix is Block-Toeplitz when it is Toeplitz by blocks.

Property 16. *The correlation matrix of a wide-sense stationary discrete-time stochastic process is Block-Toeplitz with blocks of dimension* N *by* N.

From the structure of the correlation matrix $\mathbf{R}$ described in Equation (2.111), we note that all the blocks of the main diagonal are equal to R_0, that all the blocks of the first diagonal above the main diagonal are equal to R_1^H, that all the blocks of the first diagonal below the main diagonal are equal to R_1 and so on for the other diagonals. The correlation matrix $\mathbf{R}$ is therefore Block-Toeplitz. Note that when $\mathbf{U}(n)$ represents a multidimensional zero-mean stochastic process, its correlation matrix $\mathbf{R}$ is Block-Toeplitz if and only if the multidimensional stochastic process $\mathbf{U}(n)$ is wide-sense stationary.

Property 17. *The correlation matrix of a discrete-time stochastic process is always positive semidefinite.*

The proof of this property is the same than the proof of Property 3 by taking $\mathbf{a}$ be a complex non-zero vector of size $M \times N$ and $y = \mathbf{a}^H\mathbf{U}_M(n)$.

In most practical cases, the correlation matrix of a discrete-time stochastic process is positive definite.

Property 18. *The correlation matrix of a wide-sense stationary process is always positive definite when and additive noise is present which is almost always the case in practice.*

This property has an important consequence: when the matrix $\mathbf{R}$ is positive definite, its inverse matrix $\mathbf{R}^{-1}$ exists. The presence of this additive noise is in particular present in the autoregressive models studied in next section. The correlation matrix $\mathbf{R}$ of the multidimensional wide-sense stationary stochastic processes presented in next section will therefore belong to the set of Block-Toeplitz Hermitian positive definite matrices which we denote by $\mathcal{B}_{M,N}^+$.

2.2.4 Autoregressive models

We first recall the definition of a multidimensional white noise.

Definition 8. *We call multidimensional white noise a stochastic process* $\mathbf{W}(n)$ *whose:*

- *mean is zero:* $\mathbb{E}\left[\mathbf{W}(n)\right] = 0 \quad \forall n \in \mathbb{Z}$,
- *variance is constant:* $\mathbb{E}\left[\mathbf{W}(n)\mathbf{W}(n)^H\right] = \Sigma \quad \forall n \in \mathbb{Z}$,
- *autocorrelation is zero:* $\mathbb{E}\left[\mathbf{W}(n)\mathbf{W}(k)^H\right] = 0 \quad \forall k \neq n$.

A multidimensional white noise is therefore a wide-sense stationary process by construction.
We now define multidimensional autoregressive processes of order M.

Definition 9. *We say that a stochastic process* $\mathbf{U}(n)$ *of dimension* N *by* 1 *is a multidimensional autoregressive process (AR) of order* M *when it satisfies the following equation:*

$$\mathbf{U}(n) + A_1\mathbf{U}(n-1) + \ldots + A_M\mathbf{U}(n-M) = \mathbf{W}(n) \quad \forall n \in \mathbb{Z}, \tag{2.112}$$

where $A_1, A_2, \ldots, A_M$ *are square matrices called the AR parameters and* $\mathbf{W}(n)$ *is a white noise of dimension* N *by* 1. *We say that a time series* $U(n)$ *is an multidimensional autoregressive time series when it is a realisation of a multidimensional autoregressive stochastic process.*

We call multidimensional Gaussian autoregressive process an autoregressive process whose white noise $\mathbf{W}(n)$ is a multidimensional Gaussian random variable. We will detail the properties of this particular case in Section 2.2.10.

The term "*autoregressive*" comes from the fact that the present value $\mathbf{U}(n)$ of the stochastic process is equal to a linear combination of the M previous values $\mathbf{U}(n-1), \ldots, \mathbf{U}(n-M)$ plus an error term $\mathbf{W}(n)$, as shown by the following equation:

$$\mathbf{U}(n) = -\sum_{k=1}^{M} A_k\mathbf{U}(n-k) + \mathbf{W}(n). \tag{2.113}$$

The left terms of Equation (2.112) can be considered as the convolution of the sequence $\{\mathbf{U}(n)\}$ with the sequence $\{A_n\}$. We can indeed rewrite Equation (2.112) as follows:

$$\sum_{k=0}^{M} A_k^M\mathbf{U}(n-k) = \mathbf{W}(n), \tag{2.114}$$

by setting $A_0 = I_N$.
Note that the coefficients of the autoregressive model $A_1, \ldots, A_M$ are the coefficients which minimize the expectation of the square Frobenius norm of the prediction error $\mathbf{W}(n)$ obtained by estimating the present value of the autoregressive process $\mathbf{U}(n)$ as a linear combination of the M previous values $\mathbf{U}(n-1), \ldots, \mathbf{U}(n-M)$. We now use this property to define the autoregressive coefficients A_i^j of a wide-sense stationary stochastic process which is not necessarily an AR process.

Definition 10. *Let* $\mathbf{U}(n)$ *be a wide-sense stationary stochastic process. We call autoregressive coefficients of order* M *and denote by* $A_1^M, \ldots, A_M^M$ *the coefficients which minimize the expectation of the square Frobenius norm of the prediction error, the prediction error associated with the coefficients* $A_1^M, \ldots, A_M^M$ *being defined as the term* $\mathbf{W}^M(n)$ *obtained by the following convolution operation:*

$$\sum_{k=0}^{M} A_k^M\mathbf{U}(n-k) = \mathbf{W}^M(n), \tag{2.115}$$

by setting $A_0 = I_N$.

This definition makes it possible to define the autoregressive coefficients of a wide-sense stationary process for any order M.

Since the coefficients of the autoregressive model $A_1^M, \ldots, A_M^M$ are those which minimize the expectation of the square norm of the prediction error $\mathbf{W}^M(n)$, it is possible to show that $\mathbf{W}^N(n)$ is uncorrelated with the passed values $\mathbf{U}(n-k)$ for all $k \geqslant 1$ of the process $\mathbf{U}(n) = \{\mathbf{U}(n)\}_{n\in\mathbb{Z}}$.

2.2.5 Multidimensional Yule-Walker equation

A zero-mean autoregressive process of order M described by Equation (2.112) is entirely determined by the coefficients $A_1^M, \ldots, A_M^M$ of the autoregressive model and by the variance $\Sigma_{\mathbf{W}^M}$ of the white noise $\mathbf{W}^M(n)$.

In this section, we show that these coefficients are in bijection with the autocorrelation coefficients $R_0, R_1, \ldots, R_M$:

$$\left(A_1^M, \ldots, A_M^M, \Sigma_{\mathbf{W}^M}\right) \Leftrightarrow (R_0, R_1, \ldots, R_M)\,. \tag{2.116}$$

According to Equation (2.114), we have:

$$\sum_{k=0}^{M} A_k^M \mathbf{U}(n-k) = \mathbf{W}(n) \quad \forall n \in \mathbb{Z}. \tag{2.117}$$

By multiplying both sides of Equation (2.117) by $\mathbf{U}(n-l)^H$ and then taking the expectation, we obtain:

$$\mathbb{E}\left[\sum_{k=0}^{M} A_k^M \mathbf{U}(n-k)\mathbf{U}(n-l)^H\right] = \mathbb{E}\left[\mathbf{W}(n)\mathbf{U}(n-l)^H\right] \quad \forall n \in \mathbb{Z}. \tag{2.118}$$

The left-hand side of Equation (2.118) can be expressed using the autocorrelation coefficients:

$$\mathbb{E}\left[\sum_{k=0}^{M} A_k^M \mathbf{U}(n-k)\mathbf{U}(n-l)^H\right] = \sum_{k=0}^{M} A_k^M \mathbb{E}\left[\mathbf{U}(n-k)\mathbf{U}(n-l)^H\right] = \sum_{k=0}^{M} A_k^M R_{l-k}. \tag{2.119}$$

We simplify the right-hand side of Equation (2.118) by observing that

$$\mathbb{E}\left[\mathbf{W}(n)\mathbf{U}(n-l)^H\right] = 0 \quad \forall l \geqslant 1 \tag{2.120}$$

since $\mathbf{U}(n-l)$ involves only samples of white noise up to time $n-l$ which are uncorrelated with the white noise sample $\mathbf{W}(n)$.

Since $A_0 = I_N$, we finally obtain from Equations (2.118), (2.119) and (2.120):

$$R_l + \sum_{k=1}^{M} A_k^M R_{l-k} = 0 \tag{2.121}$$

and therefore

$$R_l = -\sum_{k=1}^{M} A_k^M R_{l-k}. \tag{2.122}$$

$$\widetilde{A}_M\, \widetilde{R}_M = -\widetilde{V}_M \tag{2.123}$$

with:

$$\begin{array}{rcl} \widetilde{A}_M & = & [A_1^M, \ldots, A_M^M] \\ \widetilde{V}_M & = & [R_1, \ldots, R_M] \\ \widetilde{R}_M & = & \begin{bmatrix} R_0 & R_1 & R_2 & \ldots & R_{M-1} \\ R_1^H & R_0 & R_1 & \ldots & R_{M-2} \\ R_2^H & R_1^H & R_0 & \ldots & R_{M-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{M-1}^H & R_{M-2}^H & R_{M-3}^H & \ldots & R_0 \end{bmatrix} \end{array} \tag{2.124}$$

Note that $\widetilde{R}_M$ looks like the correlation matrix $\mathbf{R}_M$ of the multidimensional stochastic process $\mathbf{U}(n)$ given in Equation (2.111):

$$\mathbf{R}_M = \begin{bmatrix} R_0 & R_1^H & R_2^H & \ldots & R_{M-1}^H \\ R_1 & R_0 & R_1^H & \ldots & R_{M-2}^H \\ R_2 & R_1 & R_0 & \ldots & R_{M-3}^H \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{M-1} & R_{M-2} & R_{M-3} & \ldots & R_0 \end{bmatrix}. \tag{2.125}$$

When the matrix $\widetilde{R}_M$ is invertible, we can express the coefficients $\left(A_1^M, \ldots, A_M^M\right)$ as a function of the coefficients $(R_0, R_1, \ldots, R_M)$:

$$\widetilde{A}_M = -\widetilde{V}_M \widetilde{R}_M^{-1}. \tag{2.126}$$

We now express the variance of the zero-mean white noise $\Sigma_{\mathbf{W}^M}$ as a function of the autocorrelation coefficients $R_0, R_1, \ldots, R_M$ and the autoregressive coefficients $A_1^M, A_2^M, \ldots, a_M^M$. By using the definition of an autoregressive process of order M described by the convolution Equation (2.115) and by recalling that the zero-mean white noise $\mathbf{W}^M(n)$ is independent of $\mathbf{U}(n-k)$ for all $k \geqslant 1$, we have:

$$\begin{aligned}
\Sigma_{\mathbf{W}^M} =& \mathbb{E}\left[\mathbf{W}^M(n)\left(\mathbf{W}^M(n)\right)^H\right] && (2.127)\\
=& \mathbb{E}\left[\left(\sum_{k=0}^{M} A_k^M \mathbf{U}(n-k)\right)\left(\mathbf{W}^M(n)\right)^H\right] \quad \text{with } A_0^M = I_N && (2.128)\\
=& \mathbb{E}\left[\mathbf{U}(n)\left(\mathbf{W}^M(n)\right)^H\right] && (2.129)\\
=& \mathbb{E}\left[\mathbf{U}(n)\left(\sum_{k=0}^{M} A_k^M \mathbf{U}(n-k)\right)^H\right] && (2.130)\\
=& \sum_{k=0}^{M} \mathbb{E}\left[\mathbf{U}(n)\mathbf{U}^H(n-k)\right] {A_k^M}^H && (2.131)\\
=& \sum_{k=0}^{M} R_k {A_k^M}^H. && (2.132)
\end{aligned}$$

We can thus compute the variance of white noise $\Sigma_{\mathbf{W}^M}$ from the autocorrelation coefficients $R_0, R_1, \ldots, R_M$, the autoregressive coefficients $A_1^M, A_2^M, \ldots, A_M^M$ being of function of the coefficients $R_0, R_1, \ldots, R_M$ according to Equation (2.126). Explicitly, we have:

$$\begin{aligned}
\Sigma_{\mathbf{W}^M} =& \sum_{k=0}^{M} R_k {A_k^M}^H && (2.133)\\
=& R_0 + \sum_{k=1}^{M} R_k {A_k^M}^H \quad \text{as } A_0^M = I_N && (2.134)\\
=& R_0 + \widetilde{V}_M \widetilde{A}_M^H && (2.135)\\
=& R_0 + \widetilde{V}_M \left(-\widetilde{V}_M \widetilde{R}_M^{-1}\right)^H \quad \text{according to Equation (2.126)} && (2.136)\\
=& R_0 - \widetilde{V}_M \widetilde{R}_M^{-1} \widetilde{V}_M^H. && (2.137)
\end{aligned}$$

2.2.6 Multidimensional Levinson algorithm

In Definition 10, we define the autoregressive coefficients of a multidimensional wide-sense stationary stochastic process. In general, wide-sense stationary stochastic processes are not autoregressive processes of order M for $M \in \mathbb{N}$. If we want to approximate a wide-sense stationary process $\mathbf{U}(n)$ as an autoregressive process, it is then useful to estimate the coefficients of the autoregressive model of order M for several values of M, then to choose the order of the model which best corresponds to the studied process $\mathbf{U}(n)$.

When the autocorrelation coefficients $R_0, R_1, \ldots, R_M$ of the studied signal are known, it is possible to compute the coefficients $A_1^n, A_2^n, \ldots, A_n^n$ of the autoregressive model of order $n \in [\![1, M]\!]$ using the multidimensional Yule-Walker equation. The Yule-Walker equation is indeed equivalent to Equation (2.126).

In Section 2.1.6, we refer to the Levinson algorithm 13 to compute efficiently the autoregressive coefficients a_i^j for all $1 \leqslant i \leqslant j \leqslant M$ from the autocorrelation coefficients $r_0, r_1, \ldots, r_M$ of a unidimensional complex wide-sense stationary centered time series. In the case of multidimensional time series, there exists a multidimensional version of the Levinson algorithm which is only valid in the particular case where the autocorrelation coefficients $R_0, R_1, \ldots, R_M$ of the multidimensional complex wide-sense stationary centered time series are Toeplitz. A Block-Toeplitz HPD matrix whose blocks are Toeplitz is called a Toeplitz-Block Block-Toeplitz HPD matrix. This set is denoted $\mathcal{T}_{M,N}^+$ [45]. We present in Algorithm 1 an algorithm which we call multidimensional Levinson algorithm to compute efficiently the autoregressive coefficients A_i^j for all $1 \leqslant i \leqslant j \leqslant M$ of a multidimensional complex wide-sense stationary centered time series from its Toeplitz-Block Block-Toeplitz HPD autocorrelation matrix $\mathbf{R}_M$ [45]. To perform the multidimensional Levinson algorithm, we define $\overline{M} := JM^*J$, where J denotes the anti-diagonal matrix:

$$J = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 1 & 0 \\ \vdots & \vdots & \iddots & \vdots & \vdots \\ 0 & 1 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix}.$$

Algorithm 1 Multidimensional Levinson algorithm

Input: A Toeplitz-Block Block-Toeplitz HPD correlation matrix $\mathbf{R}_n$ or its autocorrelation coefficients $(R_0, R_1, \dots, R_n)$.

Initialization:

$$P_0 = R_0 \tag{2.138}$$

$$A_1^1 = -R_1 {R_0}^{-1} \tag{2.139}$$

for $l = 2, ..., n-1$: **do**

$$A_l^l = -\Delta_l {P_{l-1}}^{-1} \tag{2.140}$$

$$\Delta_l = R_l + \sum_{j=1}^{l-1} A_j^{l-1} R_{l-j} \tag{2.141}$$

$$P_{l-1} = R_0 + \sum_{j=1}^{l-1} J {A_j^{l-1}}^* J R_j = R_0 + \sum_{j=1}^{l-1} \overline{A_j^{l-1}} R_j \tag{2.142}$$

$$\widetilde{A}_l = \left[\widetilde{A}_{l-1}, 0\right] + A_l^l \left[\overline{A_{l-1}^{l-1}}, \dots, \overline{A_1^{l-1}}, I\right] \tag{2.143}$$

$$\tag{2.144}$$

end for

return $(P_0, A_1^1, ..., A_{n-1}^{n-1})$

We now consider a multidimensional complex wide-sense stationary centered time series whose correlation matrix is a Toeplitz-Block Block-Toeplitz HPD matrix. Using the recursive relations induced by Algorithm 1, it it proved in the work of Ben Jeuris and Raf Vandebril [45] that the coefficients A_n^n of a such a time series belong the space:

$$\mathcal{D}_N = \left\{M \in \mathbb{C}^{N \times N} \mid I - M\overline{M} > 0\right\} \tag{2.145}$$

where the partial ordering between complex matrices $A > B$ means that the difference $(A - B)$ is a Hermitian positive definite matrix.

Note that in dimension one, this space corresponds to the complex unit disk $\mathcal{D}$. Unfortunately, the multidimensional Levinson algorithm 1 is not surjective from the space of Toeplitz-Block Block-Toeplitz HPD matrices on the space $\mathcal{D}_N$. Indeed, we can see using Equation (2.139) that the coefficients:

$$R_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \in \mathcal{T}_2^+ \quad \text{and} \quad A_1^1 = \begin{bmatrix} 10 & 0 \\ 0 & 0 \end{bmatrix} \in \mathcal{D}_2 \tag{2.146}$$

correspond to the autocorrelation coefficient

$$R_1 = \begin{bmatrix} -10 & 0 \\ 0 & 0 \end{bmatrix} \tag{2.147}$$

which is not Toeplitz. Moreover, the coefficients R_0 and A_1^1 correspond to the Block-Toeplitz matrix:

$$\mathbf{R}_1 = \begin{bmatrix} R_0 & R_1^H \\ R_1 & R_0 \end{bmatrix} = \left[\begin{array}{cc|cc} 1 & 0 & -10 & 0 \\ 0 & 1 & 0 & 0 \\ \hline -10 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \tag{2.148}$$

whose determinant is negative: $\det(\mathbf{R}_1) = -99 < 0$. The matrix $\mathbf{R}_1$ is not HPD, it is therefore not the correlation matrix of a multidimensional wide-stationary time series.

It is proven in [45] that the coefficients $\left(A_1^1, A_2^2, \ldots, A_{M-1,M-1}\right) \in \mathcal{D}_N^{M-1}$ computed by the multidimensional Levinson algorithm 1 from a Toeplitz-Block Block-Toeplitz HPD matrix can be sligtly modified to belong to the Siegel disk defined by:

$$\mathcal{SD}_N = \left\{M \in \mathbb{C}^{N\times N} \mid I - MM^H > 0\right\}. \tag{2.149}$$

Note that there also exists a more restrictive definition of the Siegel disk with an additional symmetry condition: $M = M^T$ [54, 75].

In next section, we justify the representation of a Block-Toeplitz HPD matrix by coefficients belonging to the Siegel disk $\mathcal{SD}_N$ called the *Verblunsky coefficients*.

2.2.7 Matrix orthogonal polynomials on the unit circle

As explained in the article of Ben Jeuris and Raf Vandebril [45], a Block-Toeplitz HPD matrix can be represented by its *Verblunsky coefficients* [78, 79]. By generalizing the scalar probability measure on the complex unit circle presented in Section 2.1.7 to a nonnegative matrix measure, the Block-Toeplitz matrix defined by its moments is a Block-Toeplitz HPD matrix [28, 55, 56]. On the other hand, constructing orthogonal matrix polynomials on the unit circle with respect to the matrix measure results in a generalization of the Szegő recursion, with corresponding generalized Verblunsky coefficients [1, 25, 28].

We use the proposed generalization of the Verblunsky coefficients [28] to define a new transformation of a Block-Toeplitz HPD matrix in next section. We also detail in next section an algorithm based on the structure of the multidimensional Levinson algorithm 1 that can be used to transform a Toeplitz-Block Block-Toeplitz HPD matrix $\mathbf{R}_M \in \mathcal{T}_{M,N}^+$ into a HPD matrix $P_0 = R_0$ and a product of $(M-1)$ Siegel spaces $(\Omega_1, \Omega_2, \ldots, \Omega_{M-1}) \in \mathcal{SD}_N^{M-1}$.

2.2.8 Parameterization of multidimensional autoregressive time series in Siegel disks

In the work of Ben Jeuris and Raf Vandebril [45], the coefficients $\left(A_1^1, A_2^2, \ldots, A_{n-1}^{n-1}\right)$ computed from a Toeplitz-Block Block-Toeplitz HPD matrix using Algorithm 1 are slightly modified to belong to the Siegel disk $\mathcal{SD}_N$ (2.149) which metric has been studied in [45, 54] and will be detailed in Section 4.2.2. It is indeed proven in [45] that the following coefficients:

$$\Omega_l = \overline{P_{l-1}^{-1/2}} A_l^l P_{l-1}^{1/2} \tag{2.150}$$

belong to the Siegel disk $\mathcal{SD}_N$ (2.149), where the coefficients P_{l-1} and A_l^l are those computed by the multidimensional Levinson algorithm 1. We recall that $\overline{M} := JM^*J$, where J denotes the anti-diagonal matrix. Since

$$A_l^l = -\Delta_l {P_{l-1}}^{-1} \tag{2.151}$$

according to Equation (2.140) of the multidimensional Levinson algorithm 1, we therefore have:

$$\Omega_l = -\overline{{P_{l-1}}^{-1/2}} \Delta_l {P_{l-1}}^{-1/2}. \tag{2.152}$$

We present Algorithm 2 which transforms a Toeplitz-Block Block-Toeplitz HPD matrix $\mathbf{R}_M$ into coefficients $(P_0, \Omega_1, ..., \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. This algorithm is based on the multidimensional Levinson algorithm 1 and on Equation (2.152).

In general, the correlation matrix of a multidimensional complexe wide-sense stationary centered time series belongs to set of Block-Toeplitz HPD matrices $\mathcal{B}_{n,N}^+$ and does not belong to the subset of Toeplitz-Block Block-Toeplitz HPD matrices $\mathcal{T}_{n,N}^+ \subset \mathcal{B}_{n,N}^+$. We now present Algorithm 3 which performs the following transformation [45]:

$$\begin{aligned} \mathcal{B}_{n,N}^+ &\rightarrow \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1} \\ \widetilde{R}_n &\mapsto (P_0, \Omega_1, ..., \Omega_{n-1}). \end{aligned} \tag{2.162}$$

It is indeed proven in the works of Dette and Wagener [28] and Fritzsche and Kirstein [32] that coefficients $(\Omega_1, ..., \Omega_{n-1})$ computed by Algorithm 3 from a general Block-Toeplitz HPD matrix belong to the Siegel disk $\mathcal{SD}_N$. Note that the matrix $\widetilde{R}_{l-1}$ used in Equations (2.165), (2.166) and (2.167) of Algorithm 3 looks like the correlation matrix $\mathbf{R}_{l-1}$, as shown by Equations (2.124) and (2.125).

The coefficients $(P_0, \Omega_1, ..., \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ obtained using Algorithm 3 when the correlation matrix $\mathbf{R}_M$ belongs to $\mathcal{T}_{M,N}^+ \subset \mathcal{B}_{n,N}^+$ correspond to the coefficients obtained using Algorithm 2. In the particular case where $\mathbf{R}_n \in \mathcal{T}_{n,N}^+$, it is then better to use Algorithm 2 than Algorithm 3 to compute the coefficients $(P_0, \Omega_1, ..., \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ for complexity reasons as Algorithm 2 takes advantage of the Toeplitz-Block Block-Toeplitz structure of the matrix $\mathbf{R}_n$ to perform a recursive computation.

The reciprocal of the transformation given by Algorithm 3 is detailed in Algorithm 4.

Algorithm 2 Compute HPD and Siegel coefficients from a Toeplitz-Block Block-Toeplitz HPD matrix

Input: A Toeplitz-Block Block-Toeplitz HPD matrix $\mathbf{R}_n$ or its autocorrelation coefficients $(R_0, R_1, \ldots, R_n)$.
Initialization:

$$P_0 = R_0 \tag{2.153}$$

$$A_1^1 = -R_1 {R_0}^{-1} \tag{2.154}$$

$$\Omega_1 = -{R_0}^{-1/2} R_1 {R_0}^{-1/2} \tag{2.155}$$

for $l = 2, ..., n-1$: **do**

$$A_l^l = -\Delta_l {P_{l-1}}^{-1} \tag{2.156}$$

$$\Delta_l = R_l + \sum_{j=1}^{l-1} A_j^{l-1} R_{l-j} \tag{2.157}$$

$$P_{l-1} = R_0 + \sum_{j=1}^{l-1} J {A_j^{l-1}}^* J R_j = R_0 + \sum_{j=1}^{l-1} \overline{A_j^{l-1}} R_j \tag{2.158}$$

$$\widetilde{A}_l = \left[\widetilde{A}_{l-1}, 0\right] + A_l^l \left[\overline{A_{l-1}^{l-1}}, \ldots, \overline{A_1^{l-1}}, I\right] \tag{2.159}$$

$$\Omega_l = -\overline{{P_{l-1}}^{-1/2}} \Delta_l {P_{l-1}}^{-1/2} \tag{2.160}$$

$$\tag{2.161}$$

end for
return $(P_0, \Omega_1, ..., \Omega_{n-1})$

Algorithm 3 Compute HPD and Siegel coefficients from Block-Toeplitz HPD matrix

Input: A Block-Toeplitz HPD matrix $\mathbf{R}_n$ or its autocorrelation coefficients $(R_0, R_1, \ldots, R_n)$.
Initialization:

$$P_0 = R_0 \tag{2.163}$$

for $l = 1, ..., n-1$: **do**

$$\Omega_l := -L_{l-1}^{-1/2} \left(R_l - M_{l-1}\right) K_{l-1}^{-1/2} \tag{2.164}$$

$$L_{l-1} = R_0 - [R_1, ..., R_{l-1}]\, \widetilde{R}_{l-1}^{-1} [R_1, ..., R_{l-1}]^H \tag{2.165}$$

$$K_{l-1} = R_0 - [R_{l-1}^H, ..., R_1^H]\, \widetilde{R}_{l-1}^{-1} [R_{l-1}^H, ..., R_1^H]^H \tag{2.166}$$

$$M_{l-1} = [R_1, ..., R_{l-1}]\, \widetilde{R}_{l-1}^{-1} [R_{l-1}^H, ..., R_1^H]^H \tag{2.167}$$

end for
return $(P_0, \Omega_1, ..., \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$

Algorithm 4 Compute Block-Toeplitz HPD matrix from HPD and Siegel coefficients

Input Coefficients $(P_0, \Omega_1, \dots, \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$.
Initialization:

$$R_0 = P_0 \tag{2.168}$$

for $l = 1, ..., n-1$: **do**

$$R_l = M_{l-1} - L_{l-1}^{1/2} \Omega_l K_{l-1}^{1/2} \tag{2.169}$$
$$L_{l-1} = R_0 - [R_1, ..., R_{l-1}]\, \widetilde{R}_{l-1}^{-1} [R_1, ..., R_{l-1}]^H \tag{2.170}$$
$$K_{l-1} = R_0 - [R_{l-1}^H, ..., R_1^H]\, \widetilde{R}_{l-1}^{-1} [R_{l-1}^H, ..., R_1^H]^H \tag{2.171}$$
$$M_{l-1} = [R_1, ..., R_{l-1}]\, \widetilde{R}_{l-1}^{-1} [R_{l-1}^H, ..., R_1^H]^H \tag{2.172}$$

end for
return $(R_0, R_1, ..., R_{n-1})$

In next section, we present an algorithm to estimate the coefficients $(P_0, \Omega_1, ..., \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ of a multi-dimensional complex wide-sense stationary centered autoregressive time series $\{\mathbf{U}(n)\}_{n\in\mathbb{Z}}$ from a finite number of observations $U(0), U(1), \dots, U(n-1)$ of this time series.

2.2.9 Siegel coefficients estimation

In Section 2.2.5, we present in Equation (2.126) a formula to compute the autoregressive coefficients A_i^n for $1 \leqslant i \leqslant n$ from the autocorrelation coefficients $R_0, R_1, \dots, R_n$. In Section 2.2.8, we present Algorithm 3 which computes the coefficients $(P_0, \Omega_1, ..., \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ from the autocorrelation coefficients $R_0, R_1, \dots, R_n$. In practice, the autocorrelation coefficients $R_0, R_1, \dots, R_n$ of an observed time series $U(0), U(1), \dots, U(M-1)$ of length M are generally not known.

The autocorrelation coefficients can be estimated by computing the empirical mean:

$$\widehat{R}_i = \frac{1}{M-i} \sum_{k=0}^{n-i-1} U(k+i) U^H(k). \tag{2.173}$$

However, this method is not very precise when the integer $(M-i)$ is small. Moreover, this method does not guarantee that the Block-Toeplitz Hermitian matrix $\widehat{\mathbf{R}}_n$ constructed from the autocorrelation coefficients $\widehat{R}_0, \widehat{R}_1, \dots, \widehat{R}_{n-1}$ is positive definite.

To estimate the coefficients $(P_0, \Omega_1, \dots, \Omega_{n-1})$ of a stochastic process $\mathbf{U}(n)$ from a finite number of observations $U(0), U(1), \dots, U(M-1)$ of single time series $\{U(n)\}_{n\in\mathbb{Z}}$, we present Algorithm 5 which generalize the Burg algorithm 16 to multidimensional time series.

For any positive-definite matrix R, we denote $R^{\frac{1}{2}}$ the lower triangular matrix satisfying $R = R^{\frac{1}{2}} \left(R^{\frac{1}{2}}\right)^H$ (Cholesky decomposition of R). $R^{\frac{1}{2}}$ can be made unique by requiring the diagonal elements to be positive. We also denote $R^{-\frac{1}{2}} := \left(R^{\frac{1}{2}}\right)^{-1}$.

Theorem 3. *The matrices M_i of Algorithm 5 satisfies, $\forall i \in [\![1, n-1]\!]$:*

$$I - M_i\, {M_i}^H \geqslant 0 \tag{2.181}$$

i.e. the matrix M_i has singular values of magnitude less or equal to one [51].

Proof. Let $i \in [\![1, n-1]\!]$, we first define the matrices E_f and E_b containing respectively the forward and backward prediction errors:

$$E_f := [F_{i-1,i}, \dots, F_{i-1,p-1}] \tag{2.182}$$
$$E_b := [B_{i-1,i-1}, \dots, B_{i-1,p-2}] \tag{2.183}$$

and

$$\widetilde{E}_f := \left(E_f\, E_f^H\right)^{-\frac{1}{2}} E_f, \quad \left(\widetilde{E}_f \widetilde{E}_f^H = I\right) \tag{2.184}$$
$$\widetilde{E}_b := \left(E_b\, E_b^H\right)^{-\frac{1}{2}} E_b, \quad \left(\widetilde{E}_b \widetilde{E}_b^H = I\right) \tag{2.185}$$

Algorithm 5 Estimate multidimensional reflection coefficients

Input A vector sequence $U_0, \ldots, U_{p-1}$.

Initialization:

$$\boldsymbol{F}_{0,k} = \boldsymbol{B}_{0,k} = U_k \quad k = 0, ..., p-1 \tag{2.174}$$

$$P_0 = \frac{1}{p} \sum_{k=0}^{p-1} U_k \, U_k^H \tag{2.175}$$

for $i = 1, ..., n-1$: **do**

$$R_{i-1}^F = \sum_{k=i}^{p-1} \boldsymbol{F}_{i-1,k} {\boldsymbol{F}_{i-1,k}}^H \tag{2.176}$$

$$R_{i-1}^B = \sum_{k=i}^{p-1} \boldsymbol{B}_{i-1,k-1} {\boldsymbol{B}_{i-1,k-1}}^H \tag{2.177}$$

$$R_{i-1}^{FB} = \sum_{k=i}^{p-1} \boldsymbol{F}_{i-1,k} {\boldsymbol{B}_{i-1,k-1}}^H \tag{2.178}$$

$$M_i = -{R_{i-1}^F}^{-\frac{1}{2}} \, R_{i-1}^{FB} \left({R_{i-1}^B}^{-\frac{1}{2}} \right)^H \tag{2.179}$$

$$\left\{ \begin{array}{rcl} \boldsymbol{F}_{i,k} & = & \boldsymbol{F}_{i-1,k} + M_i \boldsymbol{B}_{i-1,k-1} \quad k = i, ..., p-1 \\ \boldsymbol{B}_{i,k} & = & \boldsymbol{B}_{i-1,k-1} + {M_i}^H \boldsymbol{F}_{i-1,k} \quad k = i, ..., p-1 \end{array} \right. \tag{2.180}$$

end for

return $(P_0, M_1, ..., M_{n-1})$

With these definitions, we can write:

$$M_i = \widetilde{E}_f \, \widetilde{E}_b^H \tag{2.186}$$

since

$$R_{i-1}^F = E_f E_f^H \quad R_{i-1}^B = E_b E_b^H \quad R_{i-1}^{FB} = E_f E_b^H . \tag{2.187}$$

We now consider:

$$0 \leqslant \left[\begin{array}{c} \widetilde{E}_f \\ \widetilde{E}_b \end{array} \right] \left[\widetilde{E}_f^H \quad \widetilde{E}_b^H \right] \tag{2.188}$$

$$= \left[\begin{array}{cc} I & M_i \\ M_i^H & I \end{array} \right] \tag{2.189}$$

$$= \left[\begin{array}{cc} I & M_i \\ 0 & I \end{array} \right] \left[\begin{array}{cc} I - M_i M_i^H & 0 \\ 0 & I \end{array} \right] \left[\begin{array}{cc} I & 0 \\ M_i^H & I \end{array} \right] \tag{2.190}$$

$$\iff I - M_i M_i^H \geqslant 0 \tag{2.191}$$

□

Note that if the number of linearly independent columns of the matrix $\left[\begin{array}{c} \widetilde{E}_f \\ \widetilde{E}_b \end{array} \right]$ is greater or equal to its number of rows, then we have:

$$I - M_i M_i^H > 0. \tag{2.192}$$

Under our model hypotheses, it occurs almost surely when: $i \leqslant p - 2 * N$, where i is the order of the model, p is the number of observation vectors U_k and N is the dimension of each vector U_k.

Once the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \ldots, \widehat{\Omega}_{n-1} \right) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ have been estimated using Algorithm 5, we can can transform these coefficients into the autocorrelation coefficients $\left(\widehat{R}_0, \widehat{R}_1, \ldots, \widehat{R}_{n-1} \right)$ using Algorithm 4. We can then use Equation (2.126) to estimate the coefficients $\left(A_1^{n-1}, A_2^{n-1}, \ldots, A_{n-1}^{n-1} \right)$ from the autocorrelation coefficients $\left(\widehat{R}_0, \widehat{R}_1, \ldots, \widehat{R}_{n-1} \right)$.

2.2.10 Complex Gaussian processes

When we study a multidimensional wide-sense stationary autoregressive stochastic process of order M:

$$\mathbf{U}(n) + A_1\mathbf{U}(n-1) + \ldots + A_M\mathbf{U}(n-M) = \mathbf{W}(n),$$

the white noise $\mathbf{W}(n)$ is generally assumed to be Gaussian. The stochastic process $\mathbf{U}(n)$ is then called a *Gaussian* process.

The definition of a multivariate complex Gaussian distribution and its probability density function are given in Section 2.1.10. Gaussian stochastic processes with complex values have also been studied in Section 2.1.10, in particular circularly-symmetric centered complex Gaussian stochastic processes (see Definition 7).

We now generalize the definition of circularly-symmetric centered complex Gaussian stochastic processes given in Definition 7 to multidimensional stochastic processes.

Definition 11. *We call multidimensional circularly-symmetric centered complex Gaussian stochastic processes the processes $\{X_t\}_{t\in T}$ for which:*

$$\left(X_{t_1}^T, \ldots, X_{t_k}^T\right)^T \sim \mathcal{CN}(0, \Gamma, 0) \quad \forall k \in \mathbb{N}^*, \forall t_1, \ldots, t_k \in T. \tag{2.193}$$

We now study the circularly-symmetric centered complex Gaussian stochastic processes which are stationary.

Let $\mathbf{U}(n)$ be such a process of length M and dimension N. The process $\mathbf{U}(n)$ therefore has the following characteristics:

- its mean is zero: $\mu = \mathbb{E}\left[\mathbf{U}(n)\right] = 0$,
- since the process $\mathbf{U}(n)$ is wide-sense stationary, its autocorrelation function is of the form: $R_k = \mathbb{E}\left[\mathbf{U}(n)\mathbf{U}(n-k)^H\right] \quad \forall k \in [\![0, M-1]\!]$.
- its relation function is zero: $\mathbb{E}\left[\mathbf{U}(n)\mathbf{U}(n-k)^T\right] = 0 \quad \forall k \in [\![0, M-1]\!]$.

The autocorrelation coefficients $R_0, R_1, \ldots, R_{M-1}$ entirely determine the correlation matrix $\mathbf{R}$ of the multidimensional Gaussian process $\mathbf{U}(n)$. Indeed, due to the stationarity of the process, the correlation matrix $\mathbf{R}$ is Block-Toeplitz as seen in Equation (2.125). We recall that the covariance matrix Γ coincide with the autocorrelation matrix $\mathbf{R}$ in the case of zero-mean stochastic processes according to the definition of the autocorrelation and autocovariance coefficients given in Equations (2.3) and (2.4). We denote in general $\mathcal{N}(0, \mathbf{R})$ the Gaussian process of zero mean, of correlation matrix $\mathbf{R}$ and of zero relation matrix. The joint probability density of the M samples of the process $\mathbf{U}(n)$ is then defined by:

$$f_{\mathbf{U}}(U) = \frac{1}{\pi^N \det(\mathbf{R})} \exp\left(-U^H\mathbf{R}^{-1}U\right), \tag{2.194}$$

where

$$U = \left[U(1)^T, U(2)^T, \ldots, U(M)^T\right]^T$$

is an $M \times N$-by-1 data vector. The matrix $\mathbf{R}$ is then the Block-Toeplitz Hermitian Positive Definite matrix of shape $(M \times N, M \times N)$ of the second order moments defined in terms of $\mathbf{U}$ by $\mathbf{R} = \mathbb{E}\left[UU^H\right]$.

In this chapter, we presented several representation spaces of complex stationary centered Gaussian autoregressive time series. In Section 2.1, we focused on **unidimensional** time series. Amoung others, we showed that unidimensional complex stationary centered Gaussian autoregressive time series can be represented by their Toeplitz HPD correlation matrix $\mathbf{R} \in \mathcal{T}_n^+$ or by the coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ in the product space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$. In Section 2.2, we studied the more general case of **multidimensional** time series. Amoung others, we showed that multidimensional complex stationary centered Gaussian autoregressive time series can be represented by their Block-Toeplitz HPD correlation matrix $\mathbf{R} \in \mathcal{B}_{n,N}^+$ or by the coefficients $(P_0, \Omega_1, \ldots, \Omega_{n-1})$ in the product space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. In the next chapter, we will endow these representation spaces with a Riemannian metric inspired by the information geometry theory.

Chapter 3

Information Geometry Theory

The information geometry theory can be used to endow the spaces of parametric probability distributions with a Riemannian metric [73]. In this chapter, we endow the spaces $\mathbb{R}_+^* \times \mathbb{D}^{n-1}$ and $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ used to represent complex stationary centered Gaussian autoregressive time series in Chapter 2 with a Riemannian metric inspired by the information geometry theory. The Riemannian metric constructed on the spaces $\mathbb{R}_+^* \times \mathbb{D}^{n-1}$ and $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ in this chapter is not the Fisher metric (often referred to as the information geometry metric). Instead, the Riemannian metric constructed have a simpler expression than the Fisher metric as it will be a product Riemannian metric.

The metric constructed in this chapter has also been studied using the Lie groups theory by Frédéric Barbaresco in [39, 40] based on the works of Jean-Marie Souriau, Jean-Louis Koszul and Ernest Vinberg [72]. This Lie groups point of view has also been used by Pierre-Yves Lagrave for radar clutter classification using Neuronal Networks in [61–63]. However, in this chapter we will only adopt the Riemannian manifolds point of view obtained using the information geometry theory presented by Shun-Ichi Amari in [73] to define a Riemannian metric on the representation spaces of complex stationary centered Gaussian autoregressive time series presented in Chapter 2.

We start by presenting the information geometry metric in Section 3.1. The representation space of the unidimensional model described previously in Section 2.1 will be endowed with a Riemannian metric in Section 3.2.2. The representation space of the multidimensional model described in Section 2.2 will be endowed with a Riemannian metric in Section 3.2.3.

Contents

3.1 Introduction to Information Geometry Theory

In this section we present the information geometry theory. For a more complete presentation of information geometry tools, we refer to the book by Shun-ichi Amari called "Information Geometry and Its Applications" [73] from which this section is inspired.

We start by introducing the notion of divergence.

3.1.1 Divergence definition

Let P and Q be two points of a Riemannian manifold M with respective coordinates ξ_P and ξ_Q. The divergence is a differentiable function of P and Q. The divergence between the points P and Q will be denoted by $D[P:Q]$ or $D[\xi_P : \xi_Q]$.

The following definition is given in [73]:

Definition 12. *The function $D[P:Q]$ is called a divergence when it satisfies the following criteria:*

1) $D[P:Q] \geqslant 0 \quad \forall (P,Q) \in M^2$.

2) $D[P:Q] = 0$ *when and only when* $P = Q$.

3) *When* P *and* Q *are sufficiently close, by denoting their coordinates by* ξ_P *and* $\xi_Q = \xi_P + d\xi$*, the Taylor expansion of* D *is written as:*

$$D[\xi_P : \xi_P + d\xi] = \frac{1}{2}\sum_{i,j} g_{i,j}(\xi_P)\, d\xi^i d\xi^j + O\left(|d\xi|^3\right) \tag{3.1}$$

where the matrix $G = (g_{i,j})$ *is positive-definite, depending on* ξ_P.

The divergence $D[P:Q]$ represents the degree of separation between the points P and Q. The divergence is not necessarily a symmetric map, therefore we have $D[P:Q] \neq D[Q:P]$ in general. The square root of a divergence is therefore not a distance in general. We call $D[P:Q]$ the divergence from P to Q.

Several examples of divergences are given in the book by Shun-ichi Amari [73]. We now present an example of divergence on probability distributions.

3.1.2 Kullback-Leibler Divergence

The Kullback-Leibler divergence is defined by S. Kullback and R. A. Leibler in the general case of probability measures over a set X [47]. Let P and Q be two probability measures over a set X such that P is absolutely continuous with respect to Q, then the Kullback-Leibler divergence from Q to P is defined as

$$D_{KL}(P:Q) = \int_X \log\left(\frac{dP}{dQ}\right) dP \tag{3.2}$$

where $\frac{dP}{dQ}$ is the Radon-Nikodym derivative of P with respect to Q, and provided the expression o the right-hand side exists.

If μ is a measure on X such that P and Q are absolutely continuous with respect to μ, then the expression of the Kullback-Leibler divergence is:

$$D_{KL}[p:q] = \int_X p(x) \log\left(\frac{p(x)}{q(x)}\right) d\mu(x) \tag{3.3}$$

where p and q are the densities defined by $p = \frac{dP}{d\mu}$ and $q = \frac{dQ}{d\mu}$.
In the following, we will often consider the set $X = \mathbb{R}^n$ and the measure $d\mu = dx$ (the Lebesgue measure).
When the random variables associated with q and q have discrete values, the integral is replaced by a sum.
We will now give a method to construct a divergence on a manifold from a convex function defined on this manifold.

3.1.3 Convex function and Bregman Divergence

Let X be a convex subset of a real vector space and let $f : X \to \mathbb{R}$ be a function.
A function ψ is said to be convex when the inequality

$$\lambda\psi(a) + (1-\lambda)\psi(b) \geqslant \psi(\lambda a + (1-\lambda)b) \tag{3.4}$$

holds for all $(a, b) \in X^2$ and for all scalar $0 \leqslant \lambda \leqslant 1$.
A function ψ is said to be *strictly* convex when the inequality

$$\lambda\psi(a) + (1-\lambda)\psi(b) < \psi(\lambda a + (1-\lambda)b) \tag{3.5}$$

holds for all $(a, b) \in X^2$ such that $a \neq b$ and for all scalar $0 < \lambda < 1$.
When the function ψ of coordinates $\xi = (\xi_1, \xi_2, \ldots, \xi_n)$ is of class C^2 on an open convex subset X of a real vector space, then the function ψ is strictly convex if and only if its Hessian matrix

$$H(\xi) = \left(\frac{\partial^2}{\partial\xi_i \partial\xi_j}\psi(\xi)\right) \tag{3.6}$$

is positive-definite for all $\xi \in X$. Note that a function ψ convex for a coordinate system θ will not necessarily be convex expressed in another coordinate system ξ. The convexity is preserved affine change of coordinates [73].

We now give an example of a convex function on exponential families of probability distributions. An exponential family of probability distributions is described by:

$$p(x,\theta) = \exp\left(\sum_i \theta_i x^i + k(x) - \psi(\theta)\right) \tag{3.7}$$

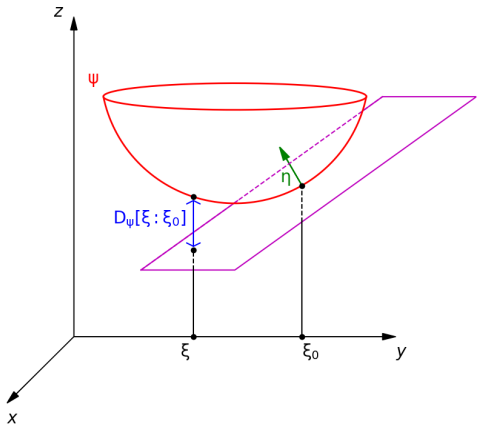

Figure 3.1: The convex function $z = \psi(\xi)$, its supporting hyperplane with normal vector $\eta = \nabla\psi(\xi_0)$ and divergence $D_\psi[\xi : \xi_0]$

where $p(x, \theta)$ is the probability density of the distribution p of parameter θ at the point x. The term $k(x)$ is a function of x. The term $\exp(-\psi(\theta))$ is the normalization factor by which:

$$\int p(x, \theta)\, dx = 1. \tag{3.8}$$

The expression of $\psi(\theta)$ is therefore given by:

$$\psi(\theta) = \log\left(\int \exp\left(\sum_i \theta_i x^i + k(x)\right) dx\right). \tag{3.9}$$

The set of exponential families of probability distributions $M = \{p(x, \theta)\}$ is considered as a manifold of which θ is a coordinate system.

By differentiating twice the application ψ given in Equation (3.9), it is proved in [73] that its Hessian matrix is positive definite. We have indeed:

$$\nabla\nabla\psi(\theta) = \mathrm{Var}_\theta[x]. \tag{3.10}$$

The function ψ is therefore a strictly convex function for the coordinate system θ. This function is known as the cumulant generating function in statistics and the free energy in statistical physics.

We now define the Bregman divergence associated with the convex function ψ.

Definition 13. *The Bregman divergence associated with the convex function ψ is defined by the function:*

$$D_\psi[\xi : \xi_0] = \psi(\xi) - \psi(\xi_0) - \nabla\psi(\xi_0) \cdot (\xi - \xi_0). \tag{3.11}$$

The Bregman divergence represents the difference between the convex function ψ evaluated at the point ξ and its tangent plane at ξ_0 evaluated at the point ξ. The figure 3.1 is given in [73] to illustrate the Bregman divergence from the graph of the convex function $\psi(\xi)$.

Consider the example of an exponential family of probability distributions. The Bregman divergence from θ to θ' associated with the function ψ defined in Equation (3.9) of an exponential family is computed using:

$$D_\psi[\theta : \theta'] = \psi(\theta) - \psi(\theta') - \nabla\psi(\theta') \cdot (\theta - \theta') \tag{3.12}$$

From Shun-ichi Amari's book [73], we have:

$$D_\psi[\theta : \theta'] = D_{KL}[p(x, \theta') : p(x, \theta)] \tag{3.13}$$

where $D_{KL}[p(x, \theta') : p(x, \theta)]$ is the Kullback-Leibler divergence defined in Section 3.1.2. This gives us here:

$$D_{KL}[p(x, \theta') : p(x, \theta)] = \int p(x, \theta') \log\left(\frac{p(x, \theta')}{p(x, \theta)}\right) dx. \tag{3.14}$$

3.1.4 Legendre transformation

We refer to the book "Convex Analysis" of R. Tyrrell Rockafellar [68] for a comprehensive study of the Legendre transform of convex functions. This book studies the class of convex functions for which the Legendre transform is well defined (see *essentially strictly convex* functions). Here, we will apply the Legendre transform to the function $\psi(\theta)$ associated with an exponential family of probability distribution (3.9) which is a strictly convex function as seen in Equation (3.10).

We denote by ξ^* the gradient of the strictly convex function ψ at the point ξ:

$$\xi^* := \nabla\psi(\xi). \tag{3.15}$$

The gradient ξ^* is equal to the normal vector ν of the hyperplane tangent to the map ψ at the point ξ. Since the map ψ is convex, two different points of the manifold M will be associated with two different normal vectors. The map which associates a point ξ to the corresponding normal vector ν is bijective and differentiable. We can therefore use the set of normal vectors ν cas a second coordinate system of M related with the coordinate system ξ by Equation (3.15). This transformation is called the Legendre transformation. The Legendre transformation has a dual structure concerning the coordinate systems ξ and ξ^*. To show it, we define the following function of ξ^*:

$$\psi^*(\xi^*) = \xi \cdot \xi^* - \psi(\xi) \tag{3.16}$$

where

$$\xi \cdot \xi^* = \sum_i \xi_i \xi_i^* \tag{3.17}$$

and ξ is a function of ξ^* defined by the inverse of the function $\xi^* = \nabla\psi(\xi)$.

By differentiating Equation (3.16) with respect to ξ^*, we obtain

$$\nabla\psi^*(\xi^*) = \xi + \frac{\partial\xi}{\partial\xi^*}\xi^* - \nabla\psi(\xi)\frac{\partial\xi}{\partial\xi^*}. \tag{3.18}$$

Since the last two terms of Equation (3.18) cancel each other out according to Equation (3.15), we have two dual coordinate systems in the following sense:

$$\xi^* = \nabla\psi(\xi), \quad \xi = \nabla\psi^*(\xi^*). \tag{3.19}$$

The function ψ^* is called the Legendre dual of ψ.

The function ψ^* satisfies the following property:

$$\psi^*(\xi^*) = \max_{\xi'} \left\{\xi' \cdot \xi^* - \psi\left(\xi'\right)\right\}. \tag{3.20}$$

When the function ψ is a convex function (not necessarily stricly convex) the "max" in Equation (3.20) should be replaced by a "sup". However, the supremum is reached in the case of strictly convex functions [14], which justify the expression given in Equation (3.20).

This property is frequently used to define the function ψ^*.

We will now show that the function ψ^* is convex for the coordinate system ξ^*. The Hessian matrix of $\psi^*(\xi^*)$ is defined by:

$$G^*(\xi^*) = \nabla\nabla\psi^*(\xi^*) = \frac{\partial\xi}{\partial\xi^*}, \tag{3.21}$$

which is the Jacobian matrix of the transformation from ξ^* to ξ.

The Hessian matrix of $\psi(\xi)$ has the expression:

$$G(\xi) = \nabla\nabla\psi(\xi) = \frac{\partial\xi^*}{\partial\xi}, \tag{3.22}$$

this is the Jacobian matrix of the transformation from ξ to ξ^*.

Hence, we have

$$G^* = G^{-1}. \tag{3.23}$$

The matrix G^* is therefore positive definite since G is positive definite. This proves that the function ψ^* is convex for the coordinate system ξ^*.

We can define a new Bregman divergence from the dual convex function ψ^* :

$$D_{\psi^*}\left[\xi^* : \xi^{*\prime}\right] = \psi^*(\xi^*) - \psi^*\left(\xi^{*\prime}\right) - \nabla\psi^*\left(\xi^{*\prime}\right)\cdot\left(\xi^* - \xi^{*\prime}\right), \tag{3.24}$$

which we call dual divergence.
According to Amari [73], we can prove that:

$$D_{\psi*}\left[\xi^* : \xi^{*\prime}\right] = D_{\psi}\left[\xi' : \xi\right]. \tag{3.25}$$

The divergence defined from the convex function $\psi(\xi)$ is equal to the divergence defined from the convex function $\psi^*(\xi^*)$ if the order of the two input points is exchanged.
We now present the dual structure when the convex function ψ is the free energy $\psi(\theta)$ defined in Equation (3.9).
The dual coordinate system given by the Legendre transformation is given by:

$$\theta^* = \nabla\psi(\theta) = \mathbb{E}_{\theta}[x], \tag{3.26}$$

where $\mathbb{E}_{\theta}$ is the expectation with respect to $p(x,\theta)$.
Because of Equation (3.26), the dual coordinate system $\theta*$ is called the expectation parameter in statistics.
According to Equation (3.16), the convex function $\psi^*(\theta^*)$ has the following expression:

$$\psi^*(\theta^*) = \theta^* \cdot \theta - \psi(\theta), \tag{3.27}$$

where θ is a function of $\theta*$ given by the reciprocal of the function $\theta^* = \nabla\psi(\theta)$.
According to [73], this proves that ψ^* is the negative entropy:

$$\psi^*(\theta^*) = \int p(x,\theta)\log\left(p(x,\theta)\right)dx. \tag{3.28}$$

The dual divergence obtained from the convex function ψ^* is the Kullback-Leibler divergence defined in Section 3.1.2:

$$D_{\Psi*}\left[\theta^* : \theta^{*\prime}\right] = D_{KL}\left[p(x,\theta) : p\left(x,\theta'\right)\right], \tag{3.29}$$

where $\theta = \nabla\psi^*(\theta^*)$ and $\theta' = \nabla\psi^*\left(\theta^{*\prime}\right)$.
We will now see how the Bregman divergence can be used to define a Riemannian metric.

3.1.5 Bregman Divergence and Riemannian Metric

Let ψ be a convex function on a manifold M for a coordinate system θ. When $d\theta$ represents an infinitesimal line, the square of its length ds is given by:

$$ds^2 = 2D_{\psi}[\theta : \theta + d\theta] = \sum_{i,j} g_{i,j}d\theta^i d\theta^j \tag{3.30}$$

according to the third condition of the definition of the divergence (3.1).
We then notice that the Riemannian metric $g_{i,j}$ is given by the Hessian matrix of the convex function ψ:

$$g_{i,j}(\theta) = \frac{\partial^2}{\partial\theta^i\partial\theta^j}\psi(\theta) \tag{3.31}$$

according to the definition of the Bregman divergence 13. Equation (3.31) can also be written as follows: $g_{i,j}(\theta) = \partial_i\partial_j\psi(\theta)$.
We can also construct a Riemannian metric from the convex function ψ^* on the coordinate system θ^* defined thanks to the Legendre transformation 3.1.4. For convenience, we will denote by η the coordinate system θ^* and by φ the convex function ψ^*.
The metric associated with the dual structure is therefore:

$$ds^{*2} = 2D_{\varphi}[\eta : \eta + d\eta] = \sum_{i,j} g^{i,j}d\eta_i d\eta_j \tag{3.32}$$

where the matrix $G^* = \left(g^{i,j}\right)$ is the inverse of the matrix $G = (g_{i,j})$ according to Equation (3.23).
The dual Riemannian metric $g^{i,j}$ is given by the Hessian matrix of the convex function φ on the dual coordinate system η:

$$g^{i,j}(\eta) = \frac{\partial^2}{\partial\eta_i\partial\eta_j}\varphi(\eta). \tag{3.33}$$

This equation can also be written: $g^{i,j}(\eta) = \partial^i\partial^j\varphi(\eta)$.
According to Equations (3.21) and (3.22), lthe coordinate systems θ and η are related by the following equations:

$$d\eta = Gd\theta, \quad d\theta = G^{-1}d\eta. \tag{3.34}$$

According to Equation (3.25), we have

$$D_{\varphi}\left[\eta : \eta'\right] = D_{\psi}\left[\theta' : \theta\right]. \tag{3.35}$$

According to [74], the metric ds^2 and its dual metric ds^{*2} are equal:

$$ds^2 = ds^{*2}. \tag{3.36}$$

According to Equation (3.34), we have indeed:

$$ds^2 = d\theta^T G d\theta = \left(G^{-1}d\eta\right)^T G \left(G^{-1}d\eta\right) = d\eta^T G^{-1} d\eta = ds^{*2}. \tag{3.37}$$

These two dual metrics therefore define the same Riemannian manifold structure.

For an exponential family of probability distributions (3.7) whose convex function $\psi(\theta)$ is given by Equation (3.9), the matrix of the metric associated with the Bregman divergence coincides with the Fisher matrix, well known in statistics.

Theorem 4. *The matrix G of the Riemannian metric associated with an exponential family is the Fisher matrix defined by*

$$g_{i,j}(\theta) = \mathbb{E}_{\theta}\left[\partial_i \log\left(p(x,\theta)\right) \partial_j \log\left(p(x,\theta)\right)\right], \tag{3.38}$$

which can also be written:

$$g_{i,j}(\theta) = -\mathbb{E}_{\theta}\left[\partial_i \partial_j \log\left(p(x,\theta)\right)\right]. \tag{3.39}$$

A proof of this theorem is given in Amari's book [73]. The use of the Fisher metric on varieties of probability ditributions is justified by the invariance properties of this metric by change of parameterization. These invariance properties are studied in Amari's book [73].

We will use the dual structure presented in this section to define a metric on the manifold of circularly-symmetric central complex normal distributions in the following sections. We will set $g^{i,j}(\eta) = \partial^i \partial^j \varphi(\eta)$ where φ is the negentropy function defined in Equation (3.28).

3.2 Information theory of circularly-symmetric central complex normal distributions

In this section, we use the information geometry theory presented in Section 3.1 to endow the space of HPD matrices with a Riemannian metric in Section 3.2.1. We will then use the results obtained in Section 3.2.1 to endow the particular case of Toeplitz HPD matrices with a Riemannian metric in Section 3.2.2; we will also to endow the particular case of Toeplitz-Block Block-Toeplitz HPD matrices with a Riemannian metric in Section 3.2.3.

3.2.1 Information theory and Hermitian Positive Definite matrices

As seen in Section 2.1.10, a complex multivariate Gaussian distribution Z is defined by its mean vector $\mu := \mathbb{E}[Z]$, its covariance matrix $\Gamma := \mathbb{E}\left[(Z-\mu)(Z-\mu)^H\right]$ and its relation matrix $C := \mathbb{E}\left[(Z-\mu)(Z-\mu)^T\right]$. The time series studied in Section 2.1 can be considered as realizations of complex multivariate normal distributions of zero mean ($\mu = 0$), of zero relation matrix ($C = 0$) and Toeplitz covariance matrix Γ. We start by recalling the definition of circularly-symmetric central complex normal distributions given in Section 2.1.10.

Definition 14. *The circularly-symmetric central complex normal distribution corresponds to the case of zero mean ($\mu = 0$) and zero relation matrix ($C = 0$).*

The probability density function of a circularly-symmetric central complex normal distribution of covariance matrix Γ has the following expression :

$$p(z,\Gamma) = \frac{1}{\pi^n \det(\Gamma)} \exp\left(-z^H \Gamma^{-1} z\right). \tag{3.40}$$

This density also can be written:

$$p(z,\Gamma) = \exp\left(-z^H \Gamma^{-1} z - \ln\left(\det(\Gamma)\right) - n\ln(\pi)\right), \tag{3.41}$$

or:

$$p(z,\Gamma) = \exp\left(-z^H \Gamma^{-1} z + \ln\left(\det\left(\Gamma^{-1}\right)\right) - n\ln(\pi)\right). \tag{3.42}$$

We set $\theta := \text{Vec}\left(\Gamma^{-1}\right)$ and $T(z) = -\text{Vec}\left(zz^H\right)$, where Vec is the application which transforms a matrix $M \in \mathbb{C}^{n\times n}$ into a vector of $\mathbb{C}^{n^2}$ by superimposing the columns of M one under others.

We can therefore express the density the following way:

$$p(z,\Gamma) = \exp\left(\theta \cdot T(z) - \psi(\theta)\right), \tag{3.43}$$

where

$$\psi(\theta) = -\ln\left(\det\left(\Gamma^{-1}\right)\right) + n\ln(\pi) \tag{3.44}$$

with $\theta = \text{Vec}\left(\Gamma^{-1}\right)$.

We recognize the structure of a probability distribution belonging to the exponential family (3.7).

The function ψ defined in Equation (3.44) is therefore a convex function for the coordinate system θ according to Equation (3.10).

We can therefore endow the space of Hermitian positive definite matrices $\mathcal{H}_n^+$ with a Riemannian manifold structure thanks to the metric resulting from the Bregman divergence (3.31):

$$g_{i,j}(\theta) = \frac{\partial^2}{\partial\bar{\theta^i}\partial\theta^j}\psi(\theta). \tag{3.45}$$

According to Theorem 4, this metric coincides with the Fisher metric:

$$g_{i,j}(\theta) = -\mathbb{E}_\theta\left[\frac{\partial^2}{\partial\bar{\theta^i}\partial\theta^j}\log\left(p(x,\theta)\right)\right]. \tag{3.46}$$

According to Section 3.1.4, the metric $g_{i,j}(\theta)$ has a dual structure.

The dual coordinate system is defined by $\eta := \nabla\psi(\theta)$. From Equation (3.26), we have:

$$\eta = \mathbb{E}_\theta\left[T(z)\right] = \text{Vec}(\Gamma). \tag{3.47}$$

The neg-entropy function ϕ defined in Equation (3.28) is convex for the coordinate system η, it is the dual function of the function ψ. For circularly-symmetric central complex normal distributions, the neg-entropy function has the following expression:

$$\begin{aligned}
\varphi(\eta) :=& \mathbb{E}_\Gamma\left[\ln\left(p(Z,\Gamma)\right)\right] && (3.48)\\
=& \mathbb{E}_\Gamma\left[\ln\left(\frac{1}{\pi^n\det(\Gamma)}\exp\left(-z^H\Gamma^{-1}z\right)\right)\right] \quad \text{according to Equation (3.40)} && (3.49)\\
=& -\ln\left(\det(\Gamma)\right) - n\ln(\pi e) - \mathbb{E}_\Gamma\left[z^H\Gamma^{-1}z\right] && (3.50)\\
=& -\ln\left(\det(\Gamma)\right) - n\ln(\pi) - n && (3.51)\\
=& -\ln\left(\det(\Gamma)\right) - n\ln(\pi e) && (3.52)
\end{aligned}$$

with $\eta = \text{Vec}(\Gamma)$. As mentioned in [82], the function $\varphi(\eta)$ is a Kähler potential. The definition and properties of a *Kähler potential* are detailed in [81].

We can therefore also define a Riemannian metric on the space of Hermitian positive definite matrices $\mathcal{H}_n^+$ as follows:

$$g^{i,j}(\eta) = \frac{\partial^2}{\partial\bar{\eta_i}\partial\eta_j}\varphi(\eta). \tag{3.53}$$

Since $ds^2 = ds^{*2}$ (3.36), the Riemannian metrics defined by Equations (3.31) and (3.53) are equivalents.

This metric endows the space $\mathcal{H}_n^+$ with a Riemannian manifold structure. The geometric tools of this manifold will be presented in Section 4.2.1.

In Section 3.2.2, we will construct a Riemannian metric on the submanifold of circularly-symmetric central complex normal distributions whose covariance matrix Γ is Toeplitz. In Section 3.2.3, we will construct a Riemannian metric on the submanifold of circularly-symmetric central complex normal distributions whose covariance matrix Γ is Block-Toeplitz. For these two submanifolds, the Riemannian metric will be constructed using Equation (3.53).

3.2.2 Information theory and Toeplitz Hermitian Positive Definite matrices

We now use the information geometry metric presented in previous Section 3.1 to endow representation spaces of unidimensional complex stationary centered Gaussian autoregressive time series presented in Section 2.1 with a Riemannian metric. This section is inspired by the thesis work of Alice Le Brigant [15] and the work of Le Yang [6], [82].

We have seen in Section 2.1 that complex stationary centered Gaussian autoregressive time series can be represented by their covariance matrices which is Toeplitz Hermitian positive definite.

The time series studied in Section (2.1) are complex stationary centered Gaussian autoregressive time series of order k:

$$u(l) + \sum_{i=1}^{k} a_i^k \, u(l-i) = w^k(l) \tag{3.54}$$

where a_i^k are the prediction coefficients and the prediction errors $w^k(l)$ are independent identically distributed centered complex Gaussian processes. Note that the coefficients $w^k(l)$ are independent of the coefficients $(u\,(l-i))_{i\geqslant 1}$.

We have seen previously that the covariance matrix of this time series is Toeplitz Hermitian positive definite (2.12):

$$\mathbf{R}_{k+1} = \begin{bmatrix} r_0 & r_1^* & r_2^* & \dots & r_k^* \\ r_1 & r_0 & r_1^* & \dots & r_{k-1}^* \\ r_2 & r_1 & r_0 & \dots & r_{k-2}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_k & r_{k-1} & r_{k-2} & \dots & r_0 \end{bmatrix}. \tag{3.55}$$

The Levinson algorithm 13 presented in detail in Appendix A can be used to compute the coefficients a_i^j of the autoregressive model from the autocorrelation coefficients $(r_0, r_1, \dots, r_{n-1})$.

The neg-entropy function is expressed for circularly-symmetric central complex normal distributions in Equation (3.48). We would like to express this function in the particular case of circularly-symmetric central complex normal distributions whose covariance matrix is Toeplitz with respect to the mean quadratic power coefficient $p_0 = r_0 \in \mathbb{R}$ and the reflection coefficients $(\mu_1, \dots, \mu_{n-1}) \in \mathcal{D}^{n-1}$ defined by $\mu_i = a_i^i$.

We denote by p_k the expectation of the square mean of the estimation error of the autoregressive model when the coefficients a_i^k are optimal:

$$p_k := \mathbb{E}\left[\left|w^k(l)\right|^2\right], \tag{3.56}$$

where $w_k(l)$ is the error term defined in Equation (3.54).

We then have:

$$\begin{bmatrix} r_0 & r_1^* & r_2^* & \dots & r_k^* \\ r_1 & r_0 & r_1^* & \dots & r_{k-1}^* \\ r_2 & r_1 & r_0 & \dots & r_{k-2}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_k & r_{k-1} & r_{k-2} & \dots & r_0 \end{bmatrix} \begin{bmatrix} 1 \\ a_1^k \\ a_2^k \\ \vdots \\ a_k^k \end{bmatrix} = \begin{bmatrix} p_k \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{3.57}$$

Equation (3.57) gives us $(k+1)$ equations whose line i is obtained by performing the operation $\mathbb{E}\left[(\cdot)\, u\,(l-i)^*\right]$ on the right and left sides of Equation (3.54).

The first line gives us in particular:

$$p_k = r_0 + \sum_{j=1}^{k} r_j^* \, a_j^k. \tag{3.58}$$

In Appendix A, we define

$$p_k = r_0 + \sum_{j=1}^{k} r_j \, a_j^{k*}. \tag{3.59}$$

According to Proprerty 26 of Appendix A, we have:

$$p_{l+1} = p_l \left(1 - |\mu_{l+1}|^2\right) \quad \forall l \in \mathbb{N}. \tag{3.60}$$

The definitions of p_k given by Equations (3.58) and (3.59) coincide since p_k is real according to Equation (3.60) and is therefore equal to its conjugate.

The coefficient p_k also has the following property:

Property 19. *The coefficient p_k can be expressed as the quotient of the determinant of Γ_{k+1} by the determinant of Γ_k [43]:*

$$p_k = \frac{\det\left(\Gamma_{k+1}\right)}{\det\left(\Gamma_k\right)} \tag{3.61}$$

Proof. Indeed, when we compute the determinant of the matrix $\mathbf{R}_{k+1}$ defined in Equation (3.55), by using the elements of its last column and by using the associated minors, we obtain [15]:

$$\det\left(\Gamma_{k+1}\right) = \sum_{i=1}^{k+1} (-1)^{k+1+i}\, r^*_{k+1-i} \det\left(A_{i,k+1}\right) \tag{3.62}$$

where the matrix $A_{i,j}$ is a matrix of size k obtained from the matrix Γ_{k+1} by removing the row i and the column j.
Note that $A_{k+1,k+1} = \Gamma_k$, therefore $\det\left(A_{k+1,k+1}\right) = \det\left(\Gamma_k\right)$.
For $i \in [\![1, k]\!]$, we have:

$$A_i^{k+1} = \begin{bmatrix} r_0 & r_1^* & r_2^* & \dots & r_{k-1}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{i-2} & r_{i-3} & r_{i-4} & \dots & r_{k-i+1}^* \\ r_i & r_{i-1} & r_{i-2} & \dots & r_{k-i-1}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_k & r_{k-1} & r_{k-2} & \dots & r_1 \end{bmatrix} \begin{matrix} \widetilde{L}_1 \\ \vdots \\ \widetilde{L}_{i-1} \\ \widetilde{L}_i \\ \vdots \\ \widetilde{L}_k \end{matrix} . \tag{3.63}$$

To transform Equation (3.62), we would like to express the determinant of the matrix A_i^{k+1} as a function of the determinant of the matrix Γ_k. We denote by $\widetilde{L}_j$ the j^{th} row of the matrix A_i^{k+1} and by L_j the j^{th} row of the matrix Γ_k. Note that

$$\forall j \in [\![1, i-1]\!], \widetilde{L}_j = L_j, \quad \forall j \in [\![i, k-1]\!], \widetilde{L}_j = L_{j+1}. \tag{3.64}$$

We now modify the line $\widetilde{L}_k$ of the matrix A_i^{k+1} by an operation which does not change its determinant:

$$\widetilde{L}_k \leftarrow \widetilde{L}_k + \sum_{j=1}^{i-1} a_{k+1-j}^k \widetilde{L}_j + \sum_{j=i}^{k-1} a_{k-j}^k \widetilde{L}_j = L_{k+1} + \sum_{\substack{j=1 \\ j\neq i}}^{k-1} a_{k+1-j}^k L_j. \tag{3.65}$$

From the equations defined by the matrix equation (3.57), we then have $\widetilde{L}_k = -a_{k+1-i}^k L_i$.
To obtain the matrix Γ_k (except for a multiplication of its i^{th} row) from the matrix A_i^{k+1}, it suffices to perform a cyclic permutation of the last rows of the matrix A_i^{k+1}:

$$\widetilde{L}_i \leftarrow \widetilde{L}_k \quad \text{and} \quad \forall j \in [\![i, k-1]\!], \quad \widetilde{L}_{i+1} \leftarrow \widetilde{L}_i. \tag{3.66}$$

This operation modifies the determinant of the matrix A_i^{k+1} by a factor $(-1)^{k-i}$.
Therefore, we have:

$$\det\left(A_{i,k+1}\right) = (-1)^{k-i}\left(-a_{k+1-i}^k\right)\det\left(\Gamma_k\right) = (-1)^{k-i+1}\, a_{k+1-i}^k \det\left(\Gamma_k\right). \tag{3.67}$$

We finally have, coming back to Equation (3.62) :

$$\begin{aligned} \det\left(\Gamma_{k+1}\right) &= \sum_{i=1}^{k+1} (-1)^{k+1+i}\, r^*_{k+1-i} \det\left(A_{i,k+1}\right) && (3.68)\\ &= \sum_{i=1}^{k+1} (-1)^{k+1+i}\, r^*_{k+1-i}\, (-1)^{k-i+1}\, a_{k+1-i}^k \det\left(\Gamma_k\right) && (3.69)\\ &= \sum_{i=1}^{k+1} (-1)^{2(k+1)}\, r^*_{k+1-i} a_{k+1-i}^k \det\left(\Gamma_k\right) && (3.70)\\ &= \left(\sum_{i=0}^{k} r_i^* a_i^k\right) \det\left(\Gamma_k\right) && (3.71) \end{aligned}$$

by setting $a_0^k = 1$.
We also have $p_k = \sum_{i=0}^k r_i^* a_i^k$ according to Equation (3.58). So we finally have:

$$\det\left(\Gamma_{k+1}\right) = p_k \det\left(\Gamma_k\right), \tag{3.72}$$

which concludes the proof.

□

Since $p_0 = r_0$, we can prove by recurrence that:

$$\det\left(\Gamma_k\right) = \prod_{i=0}^{k-1} p_i. \tag{3.73}$$

According to Equation (3.60), we therefore have:

$$\det\left(\Gamma_n\right) = p_0^n \prod_{k=1}^{n-1} \left(1 - |\mu_k|^2\right)^{n-k}. \tag{3.74}$$

The neg-entropy function expressed for circularly-symmetric central complex normal distributions in Equation (3.48):

$$\varphi\left(\eta\right) = -\ln\left(\det\left(\Gamma_k\right)\right) - n\ln\left(\pi e\right), \tag{3.75}$$

also has the following expression:

$$\varphi\left(\eta\right) := \mathbb{E}_\Gamma\left[\ln\left(p\left(Z,\Gamma\right)\right)\right] = -n\ln\left(p_0\right) - \sum_{k=1}^{n-1}\left(n-k\right)\ln\left(1-|\mu_k|^2\right) - n\ln\left(\pi e\right). \tag{3.76}$$

The metric induced by the Hessian matrix of the neg-entropy function is therefore given by:

$$ds^2 = -\frac{\partial^2}{\partial\bar{\beta}_i\partial\beta_j}\left(n\ln\left(p_0\right) + \sum_{k=1}^{n-1}\left(n-k\right)\ln\left(1-|\mu_k|^2\right)\right)d\bar{\beta}_i d\beta_j \tag{3.77}$$

where $\beta_0 = p_0$ and $\beta_i = \mu_i$ for all $i \geqslant 1$.
We obtain:

$$ds^2 = n\left(\frac{dp_0}{p_0}\right)^2 + \sum_{k=1}^{n-1}\left(n-k\right)\frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2}. \tag{3.78}$$

We therefore have a product metric:

$$ds^2 = \sum_{k=0}^{n-1} ds_k^2. \tag{3.79}$$

We recognize a metric on $\mathbb{R}_+^*$:

$$ds_0^2 = n\left(\frac{dx}{x}\right)^2 \tag{3.80}$$

and a metric on the complex unit disk which is equal to the metric of the Poincaré disk except for a multiplicative constant:

$$ds_k^2 = \left(n-k\right)\frac{|dz|^2}{\left(1-|z|^2\right)^2}, \quad \forall 1 \leqslant k \leqslant n-1. \tag{3.81}$$

The metric defined by Equation (3.78) endows the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ with a Riemannian metric. This manifold is denoted $\mathbb{R}^{++} \times \mathbb{D}^{n-1} := \left(\mathbb{R}_+^*, ds_0^2\right) \times \left(\mathcal{D}, ds_1^2\right) \times \ldots \times \left(\mathcal{D}, ds_{n-1}^2\right)$ and called *Poincaré polydisk* in [15].
Note that this manifold is not endowed with the Fisher metric whose matrix G is defined in Equation (3.38) by:

$$g_{i,j}\left(\theta\right) = \mathbb{E}_\theta\left[\partial_i \log\left(p\left(x,\theta\right)\right)\partial_j \log\left(p\left(x,\theta\right)\right)\right] = -\mathbb{E}_\theta\left[\partial_i\partial_j \log\left(p\left(x,\theta\right)\right)\right]. \tag{3.82}$$

Indeed, in the particular case of AR process of order one ($n = 2$), the infinitesimal square length element given in Equation (3.78) becomes:

$$ds^2 = 2\left(\frac{dp_0}{p_0}\right)^2 + \frac{|d\mu_1|^2}{\left(1-|\mu_1|^2\right)^2} \tag{3.83}$$

which corresponds to the scalar product matrix:

$$G = \begin{pmatrix} \frac{2}{p_0^2} & 0 \\ 0 & \frac{1}{\left(1-|\mu_1|^2\right)^2} \end{pmatrix} \tag{3.84}$$

However, the Fisher matrix of on AR process of order one is:

$$G_{Fisher} = \begin{pmatrix} \frac{2}{p_0^2} & \frac{-\mu_1}{p_0\left(1-|\mu_1|^2\right)} \\ \frac{-\bar{\mu}_1}{p_0\left(1-|\mu_1|^2\right)} & \frac{1}{\left(1-|\mu_1|^2\right)^2} \end{pmatrix} \tag{3.85}$$

The computation of the coefficients $g_{0,0}$ and $g_{1,0}$ has been done explicitly in the thesis of Le Yang [82]. The matrix G_{Fisher} is not a diagonal matrix, therefore the associated metric is not a product metric. As a consequence, the description of the geometric tools (distance, Riemannian logarithm map, Riemannian exponential map, geodesics, sectional curvature...) of the manifold obtained by endowing the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ with the Fisher metric would be very difficult.

As seen in Section 3.1 and explained in the book written by Shun-ichi Amari [73], if we consider an exponential family whose probability density function (3.7) is given by:

$$p\left(x,\theta\right) = \exp\left(\sum_i \theta_i x^i + k(x) - \psi\left(\theta\right)\right) = \exp\left(\theta \cdot x + k(x) - \psi\left(\theta\right)\right), \tag{3.86}$$

then the scalar product matrix obtained by taking the Hessian of the convex function $\psi\left(\theta\right)$ (related to the Bregman divergence) is equal to the Fisher matrix (Theorem 4). As seen in Equation (3.36), this metric defines the same manifold structure (same distance, geodesics...) than the dual metric defined by the Hessian of the negative entropy function on the set of dual parameters obtained by the Legendre transformation. However, the manifold constructed in this section on the set $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Hessian of the negative entropy function does not correspond to the Fisher matrix. Indeed, the set of dual parameters $\eta = \text{Vec}\left(\Gamma\right)$ given in Equation (3.47) as been transformed into another set of parameters $(p_0, \mu_1, \ldots, \mu_{n-1})$. However, the set of parameters $(p_0, \mu_1, \ldots, \mu_{n-1})$ might be the Legendre transform of a set of parameters θ which cannot be expressed in the form of Equation (3.86). In this case, the conditions of Theorem 4 are not satisfied and the manifold obtained by considering the Hessian of the neg-entropy function on the set of dual parameters does not necessarily coincide with manifold obtained using the Fisher metric.

We will present the geometric tools of the product manifold constructed here in Section 4.1 of Chapter 4. Now that we have endowed with a Riemannian metric the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ which can be used to represent complex *unidimensional* stationary centered Gaussian autoregressive time series, we endow with a Riemannian metric the space $H_N^+ \times \mathcal{SD}_N^{n-1}$ presented in Section 2.2.8 which can be used to represent complex *multidimensional* stationary centered Gaussian autoregressive time series.

3.2.3 Information theory and Block-Toeplitz Hermitian Positive Definite matrices

In this section use the information geometry metric presented in Section 3.1 to endow representation spaces of multidimensional complex stationary centered Gaussian autoregressive time series presented in Section 2.2 with a Riemannian metric. The objective is to construct a Riemannian metric on one of the representation spaces of multidimensional complex stationary centered Gaussian autoregressive time series. This section is inspired by the work of Ben Jeuris and Raf Vandebril [45].

We recall the equation of the multidimensional complex stationary centered Gaussian autoregressive time series studied in Section 2.2:

$$U(k) + \sum_{j=1}^{n-1} A_j^{n-1} U(k-j) = W(k) \tag{3.87}$$

where W is the prediction error vector and the autoregressive coefficients A_j^{n-1} are square matrices. The prediction errors $W(k)$ are assumed to have independent and indentically distributed circularly-symmetric centered Gaussian distributions. The coefficient $W(k)$ is independent of the vectors $U\left(k-j\right)$ for all $j \geqslant 1$.

The time series described by Equation (3.87) have a Block-Toeplitz Hermitian positive definite covariance matrix according to Equation (2.125):

$$\mathbf{R}_n = \begin{bmatrix} R_0 & R_1^H & R_2^H & \ldots & R_{n-1}^H \\ R_1 & R_0 & R_1^H & \ldots & R_{n-2}^H \\ R_2 & R_1 & R_0 & \ldots & R_{n-3}^H \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{n-1} & R_{n-2} & R_{n-3} & \ldots & R_0 \end{bmatrix} \begin{matrix} L_1 \\ L_2 \\ L_3 \\ \vdots \\ L_n \end{matrix}. \tag{3.88}$$

We recall the formula of the neg-entropy function expressed for circularly-symmetric central complex normal distributions given in Equation (3.48):

$$\varphi\left(\mathbf{R}_n\right) = -\ln\left(\det\left(\mathbf{R}_n\right)\right) - n\ln\left(\pi e\right). \tag{3.89}$$

We have seen in Section 2.2.6 that the matrix coefficients A_i^i of the autoregressive model belong to the set $\mathcal{D}_N := \left\{\Gamma \in \mathbb{C}^{N\times N} \mid I - \Gamma\overline{\Gamma} > 0\right\}$. We now try to express the function φ with respect to R_0 and to the autoregressive coefficients $\left(A_1^1, A_2^2, \ldots, A_{n-1}^{n-1}\right)$. We can then obtain a Riemannian metric on the space $\mathcal{H}_N^+ \times \mathcal{D}_N^{n-1}$ by computing the Hessian matrix of φ with respect to R_0 and to the coefficients A_i^i.

The autocorrelation coefficients R_i and the coefficients of the autoregressive model are related by the Yule-Walker equations (2.123):

$$\widetilde{A}_n \, \widetilde{R}_n = -\widetilde{V}_n \tag{3.90}$$

with:

$$\begin{array}{rcl} \widetilde{A}_n & = & [A_1^n, \ldots, A_n^n] \\ \widetilde{V}_n & = & [R_1, \ldots, R_n] \\ \widetilde{R}_n & = & \begin{bmatrix} R_0 & R_1 & R_2 & \ldots & R_{n-1} \\ R_1^H & R_0 & R_1 & \ldots & R_{n-2} \\ R_2^H & R_1^H & R_0 & \ldots & R_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{n-1}^H & R_{n-2}^H & R_{n-3}^H & \ldots & R_0 \end{bmatrix}. \end{array} \tag{3.91}$$

We denote by L_1 the N first rows of the matrix $\mathbf{R}_n$, L_2 the next N rows and so on where N is the size of the blocks of the matrix $\mathbf{R}_n$. By permuting the rows of the matrix $\mathbf{R}_n$ by the operation $L_i \leftarrow L_{n-i}$ then the columns by $C_i \leftarrow C_{n-i}$, we can transform the matrix $\mathbf{R}_n$ into the matrix $\widetilde{R}_n$. Hence we have, since there is an even number of real rows (of size 1) to permute:

$$\det\left(\mathbf{R}_n\right) = \det\left(\widetilde{R}_n\right). \tag{3.92}$$

Recall the block matrix inversion formula based on the notion of "Schur complement" [45]:

$$\widetilde{R}_{l+1}^{-1} = \begin{bmatrix} \alpha_l & -\alpha_l U_l \widetilde{R}_l^{-1} \\ -\widetilde{R}_l^{-1} U_l^H \alpha_l & \widetilde{R}_l^{-1} U_l^H \alpha_l U_l \widetilde{R}_l^{-1} \end{bmatrix} \tag{3.93}$$

with $\alpha_l = \left(R_0 - U_l \widetilde{R}_l^{-1} U_l^H\right)^{-1}$. Note that the matrix α_l is Hermitian positive definite as the main submatrix of the Hermitian positive definite matrix $\widetilde{R}_{l+1}^{-1}$.

We recall the notation $\overline{M} := JM^*J$, where J is the antidiagonal matrix. The auxiliary matrix P_l involved in the multidimensional Levinson algorithm 1 can be expressed as a function of the coefficient α_l:

$$\bar{P}_l = R_0 + \widetilde{A}_l U_l^H = R_0 + U_l \widetilde{R}_l^{-1} U_l^H = \alpha_l^{-1}. \tag{3.94}$$

Since the matrix α_l is a Hermitian positive definite matrix, the matrix $\bar{P}_l$ and therefore the matrix P_l are also HPD matrices.

Using the multidimensional Levinson algorithm 1, we have the following recurrence formula [45]:

$$\bar{P}_l = \overline{P_{l-1}} - \Delta_l P_{l-1}^{-1} \overline{\Delta_l} = \left(I - A_l^l \overline{A_l^l}\right) \overline{P_{l-1}} \tag{3.95}$$

with $\overline{P_0} = R_0$.

By recurrence, we therefore have:

$$\bar{P}_l = R_0 \prod_{i=0}^{l} \left(I - A_i^i \overline{A_i^i}\right). \tag{3.96}$$

The determinant of $\widetilde{R}_n$ also has the following recurrence property [45]:

$$\begin{aligned} \det\left(\widetilde{R}_n\right) &= \det\left(\widetilde{R}_{n-1}\right) \det\left(R_0 - U_{n-1} \widetilde{R}_{n-1}^{-1} U_{n-1}^H\right) && (3.97) \\ &= \det\left(\widetilde{R}_{n-1}\right) \det\left(\alpha_{n-1}^{-1}\right) && (3.98) \\ &= \det\left(\widetilde{R}_{n-1}\right) \det\left(\overline{P_{n-1}}\right). && (3.99) \end{aligned}$$

We therefore obtain by recurrence:

$$\det\left(\widetilde{R}_n\right) = \prod_{i=0}^{n-1} \det\left(\bar{P}_i\right) \tag{3.100}$$

and therefore, according to Equation (3.96) :

$$\det\left(\widetilde{R}_n\right) = \left(\det\left(R_0\right)\right)^n \prod_{i=1}^{n-1} \left(\det\left(I - A_i^i\right)\right)^{n-i}. \tag{3.101}$$

The neg-entropy function for circularly-symmetric central complex normal distributions recalled in Equation (3.89) can therefore be expressed as a function of P_0 and of the coefficients A_i^i the following way:

$$\varphi\left(\mathbf{R}_n\right) = -n\log\left(\det\left(P_0\right)\right) - \sum_{l=1}^{n-1}(n-l)\log\left(\det\left(I - \Gamma_l\bar{\Gamma}_l\right)\right) - n\log\left(\pi e\right). \tag{3.102}$$

We obtain the infinitesimal squared length ds^2 by computing the Hessian matrix of the function φ (we use complex differentiation):

$$\begin{aligned} ds^2 =& n\,\mathrm{trace}\left(P_0^{-1}dP_0P_0^{-1}dP_0\right) \\ &+ \sum_{l=1}^{n-1}(n-l)\,\mathrm{trace}\left(\left(I-\Gamma_l\bar{\Gamma}_l\right)^{-1}d\Gamma_l\left(I-\bar{\Gamma}_l\Gamma_l\right)^{-1}d\bar{\Gamma}_l\right). \end{aligned} \tag{3.103}$$

We recognize a product metric. Up to a multiplicative constant, the metric obtained on $\mathcal{H}_N^+$ is:

$$ds^2 = \mathrm{trace}\left(P_0^{-1}dP_0P_0^{-1}dP_0\right). \tag{3.104}$$

This metric is well known and is discussed in Section 4.2.1 of Chapter 4.
However, the metric obtained on the spaces $\mathcal{D}_N$ is up to a multiplicative constant:

$$ds^2 = \mathrm{trace}\left(\left(I-\Gamma_l\bar{\Gamma}_l\right)^{-1}d\Gamma_l\left(I-\bar{\Gamma}_l\Gamma_l\right)^{-1}d\bar{\Gamma}_l\right). \tag{3.105}$$

This metric on the space $\mathcal{D}_N$ does not correspond to the metric of a already known manifold and the explicit expression of the distance on this manifold does not seem obvious [45]. However the "metric" defined by Equation (3.105) is not a metric as it do not respect the indentity of indiscernables ($d(x,y) = 0 \Leftrightarrow x = y$). Indeed, if we consider:

$$\Gamma_l = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad d\Gamma_l = \begin{pmatrix} dz_1 & dz_2 \\ dz_3 & dz_4 \end{pmatrix}, \tag{3.106}$$

then we obtain:

$$\begin{aligned} ds^2 &= \mathrm{trace}\left(\left(I-\Gamma_l\bar{\Gamma}_l\right)^{-1}d\Gamma_l\left(I-\bar{\Gamma}_l\Gamma_l\right)^{-1}d\bar{\Gamma}_l\right) && (3.107)\\ &= \mathrm{trace}\left(d\Gamma_l d\bar{\Gamma}_l\right) && (3.108)\\ &= \mathrm{trace}\left(d\Gamma_l J d\Gamma_l^* J\right) \quad \text{where } J = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} && (3.109)\\ &= \mathrm{trace}\left(\begin{pmatrix} dz_1 & dz_2 \\ dz_3 & dz_4 \end{pmatrix}\begin{pmatrix} dz_4^* & dz_3^* \\ dz_2^* & dz_1^* \end{pmatrix}\right) && (3.110)\\ &= 2\,\mathrm{Re}\left(dz_1dz_4^*\right) + \left|dz_2\right|^2 + \left|dz_3\right|^2. && (3.111) \end{aligned}$$

If we denote $z_j = dx_j + i dy_j$ for $j \in [\![1,4]\!]$, then we obtain:

$$ds^2 = 2dx_1dx_4 + 2dy_1dy_4 + dx_2^2 + dy_2^2 + dx_3^2 + dy_3^2 \tag{3.112}$$

which is not the infinitesimal element ds^2 of a metric: if $dz_1 = dz_2 = dz_3 = 0$ and $|dz_4| > 0$, then $ds^2 = 0$.
However, the set $\mathcal{D}_N := \left\{\Gamma \in \mathbb{C}^{N\times N} \mid I - \Gamma\bar{\Gamma} > 0\right\}$ looks like the space:

$$\mathcal{SD}_N := \left\{\Omega \in \mathbb{C}^{N\times N} \mid I - \Omega\Omega^H > 0\right\}. \tag{3.113}$$

This space is called the Siegel space. Note that the Siegel space defined here has no symmetry condition; there is another definition of Siegel space including this symmetry condition: $\Omega = \Omega^T$.

The Siegel space is commonly associated with a Riemannian metric which will be presented in Section 4.2.2.

Hence, we will transform the coefficients $A_i^i \in \mathcal{D}_N$ so that they belong to the Siegel space $\mathcal{SD}_N$. We will then express the metric ds^2 given in Equation (3.105) with respect to the newly obtained Siegel coefficients, hoping the recognize the usual Riemannian metric of the Siegel disk.

To transform the coefficients $A_i^i \in \mathcal{D}_N$ so that they belong to the Siegel space $\mathcal{SD}_N$, we can use the auxiliary matrix P_{l-1} which is used in Algorithm 1:

$$\Omega_l = -\overline{P_{l-1}^{-1/2}} \Delta_l P_{l-1}^{-1/2}. \tag{3.114}$$

Since we have:

$$A_l^l = -\Delta_l P_{l-1}^{-1}, \tag{3.115}$$

We then have:

$$\Omega_l = \overline{P_{l-1}^{-1/2}} A_l^l P_{l-1}^{1/2}. \tag{3.116}$$

It has been proved by Dette and Wagener [28] and by Fritzsche and Kirstein [32] that this operation transforms a Block-Toeplitz HPD matrix into coefficients which belong to the Siegel disk.

We would now like to express the neg-entropy function φ expressed for the circularly-symmetric central complex normal distributions in Equation (3.89) with respect to the coefficient P_0 and to the coefficients $\Omega_i \in \mathcal{SD}_N$.

In the case where the blocks R_i of the matrix $\mathbf{R}_n$ are themselves Toeplitz, we have [45]:

$$\overline{\Delta_l} = \Delta_l^H. \tag{3.117}$$

In this case:

$$\begin{aligned}
\det\left(I - A_l^l \bar{A}_l^l\right) &= \det\left(I - \Delta_l P_{l-1}^{-1} \overline{\Delta_l} \overline{P_{l-1}^{-1}}\right) && (3.118)\\
&= \det\left(I - \Delta_l P_{l-1}^{-1} \Delta_l^H \overline{P_{l-1}^{-1}}\right) && (3.119)\\
&= \det\left(I - \overline{P_{l-1}^{-1/2}} \Delta_l P_{l-1}^{-1} \Delta_l^H \overline{P_{l-1}^{-1/2}}\right) && (3.120)\\
&= \det\left(I - \Omega_l \Omega_l^H\right). && (3.121)
\end{aligned}$$

According to Equation (3.102), the function φ therefore has the following expression:

$$\varphi\left(\mathbf{R}_n\right) = -n \log\left(\det\left(P_0\right)\right) - \sum_{l=1}^{n-1} (n-l) \log\left(\det\left(I - \Omega_l \Omega_l^H\right)\right) - n \log\left(\pi e\right). \tag{3.122}$$

As before, we obtain the metric ds^2 by computing the Hessian matrix of the function φ with respect to the coefficient P_0 and to the coefficients Ω_i using the complex differentiation:

$$\begin{aligned}
ds^2 =& n \operatorname{trace}\left(P_0^{-1} dP_0 P_0^{-1} dP_0\right) && (3.123)\\
&+ \sum_{l=1}^{n-1} (n-l) \operatorname{trace}\left(\left(I - \Omega_l \Omega_l^H\right)^{-1} d\Omega_l \left(I - \Omega_l^H \Omega_l\right)^{-1} d\Omega_l^H\right).
\end{aligned}$$

We have thus endowed the space $H_N^+ \times \mathcal{SD}_N^{n-1}$ with a product Riemannian metric.

The metric on H_N^+ remains the same as the metric obtained using the first transformation (3.105).

Up to a multiplicative constant, we recognize the usual metric on the Siegel space $\mathcal{SD}_N$:

$$ds^2 = \operatorname{trace}\left(\left(I - \Omega_l \Omega_l^H\right)^{-1} d\Omega_l \left(I - \Omega_l^H \Omega_l\right)^{-1} d\Omega_l^H\right). \tag{3.124}$$

The metric given in Equation (3.124) can also be derived from the metric of the Siegel upper halfplane described by Siegel himself [75], using the following bijection

$$\begin{aligned}
\Omega &= (B - iI)(B + iI)^{-1} && (3.125)\\
B &= i(I + \Omega)(I - \Omega)^{-1} && (3.126)
\end{aligned}$$

where B is an element of the Siegel upper halfplane ($\mathrm{Im}\,(B) > 0$). Note that this link and the Siegel disk itself are classically only defined for symmetric matrices (in order for the positive definiteness of $\mathrm{Im}\,(B)$ to make sense) [45]. However, removing the symmetry restriction only disrupts the link and the definition of the Siegel upper halfplane, while the Siegel disk and its geometry remain well-defined.

For simplicity, we will use the metric defined by Equation (3.124) even when the blocks R_i of the matrix $\mathbf{R}_n$ are not assumed to be Toeplitz. In addition to this simplification, the metric given in Equation (3.123) does not correspond to the Fisher metric for an another reason. Indeed, the set of parameters $(R_0, \Omega_1, \ldots, \Omega_{n-1})$ might be the Legendre transform of a set of parameters θ which cannot be expressed in the form of Equation (3.7). In this case, the conditions of Theorem 4 are not satisfied and the manifold obtained by considering the Hessian of the neg-entropy function on the set of dual parameters does not necessarily coincide with manifold obtained using the Fisher metric, as detailed in Section 3.2.2 in the particular case where the blocks of the Toeplitz-Block Block-Toeplitz HPD matrix $\mathbf{R}_n$ are of size one (the matrix $\mathbf{R}_n$ is then Toeplitz HPD matrix).

The geometric tools of this metric will be studied in detail in Section 4.2.2.

We devote Chapter 4 to the presentation of the geometric tools of the Riemanian manifolds obtained in Section 3.2. The Riemannian manifold on the space $\mathcal{H}_n^+$ obtained in Section 3.2.1 will be studied in Section 4.2.1. The manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ defined in Section (3.2.2) will be studied in Section 4.1.3. Finally, the manifold obtained on the space $H_N^+ \times \mathcal{SD}_N^{n-1}$ will be studied in Section (4.2.3) of Chapter 4.

Chapter 4

Riemannian manifolds associated with complex valued stationary time series

The spaces of representation of complex stationary centered Gaussian autoregressive time series have been studied in Chapter 2. We have shown that the one-dimensional complex stationary centered Gaussian autoregressive time series presented in Chapter 2 in Section 2.1 can be represented by their Toeplitz covariance matrix or by their reflection coefficients which belong to the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$. This product space has been endowed with a product Riemannian metric in Section 3.2.2. The multidimensional complex stationary centered Gaussian autoregressive time series presented in Section 2.2 can be represented in at least three spaces. Indeed, the autocorrelation matrix $\mathbf{R}$ belongs to the space of Block-Toeplitz HPD matrices $\mathcal{B}_{n,N}^+$, the coefficients $\left(R_0, A_1^1, \ldots, A_{n-1}^{n-1}\right)$ belong to $\mathcal{H}_N^+ \times \mathcal{D}_N^{n-1}$ and the coefficients $(R_0, \Omega_1, \ldots, \Omega_{n-1})$ belong to $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. This last space has been endowed with a Riemannian metric in Section 3.2.3. In this chapter, we will present the geometric tools of the product manifolds obtained in Chapter 3.

The table 4.1 synthesizes the spaces and the metrics obtained in Chapter 3 and introduces the notations used in this chapter.

Table 4.1: Manifolds synthesis

Manifold name and symbol	Space	Metric
Strictly positive real manifold $\mathbb{R}^{++}$	Strictly positive reals $\mathbb{R}_+^*$	$ds^2_{\mathbb{R}^{++}} = \left(\frac{dx}{x}\right)^2$
Poincaré disk $\mathbb{D}$	Complex unit disk $\mathcal{D} := \{z \in \mathbb{C}, \lvert z\rvert < 1\}$	$ds^2_{\mathbb{D}} = \frac{\lvert dz\rvert^2}{\left(1-\lvert z\rvert^2\right)^2}$
Strictly positive reals and Poincaré disks product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$	Positive reals ans complex unit disks product space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$	$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n ds^2_{\mathbb{R}^{++}} + \sum_{k=1}^{n-1} (n-k)\, ds^2_{\mathbb{D}}$
HPD manifold $\mathbb{H}_N^{++}$	HPD matrices $\mathcal{H}_N^+ := \left\{M \in \mathbb{C}^{N\times N} \mid x^H M x > 0, \forall x \in \mathbb{C}^N\right\}$	$ds^2_{\mathbb{H}_N^{++}} = \operatorname{trace}\left(P^{-1}\, dP\, P^{-1}\, dP\right)$
Siegel manifold $\mathbb{SD}_N$	Siegel disk $\mathcal{SD}_N := \left\{M \in \mathbb{C}^{N\times N} \mid I - MM^H > 0\right\}$	$ds^2_{\mathbb{SD}_N} = \operatorname{trace}\left(\left(I-\Omega\Omega^H\right)^{-1} d\Omega \left(I-\Omega^H\Omega\right)^{-1} d\Omega^H\right)$
HPD and Siegel disks product manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$	HPD and Siegel disks product space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$	$ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} = n ds^2_{\mathbb{H}_N^{++}} + \sum_{l=1}^{n-1} (n-l)\, ds^2_{\mathbb{SD}_N}$

Contents

4.1 Manifolds associated with the unidimensional autoregressive model

The Riemman manifolds presented in this section come from the representation spaces of unidimensional complex stationary centered Gaussian autoregressive time series presented in Section 2.1 and endowed with a Riemannian metric inspired by information geometry in Section 3.2.2.

4.1.1 The manifold of strictly positive reals $\mathbb{R}^{++}$

We denote $\mathbb{R}^{++}$ the Riemannian manifold defined by the space $\mathbb{R}_+^*$ endowed with the following Riemannian metric:

$$ds^2_{\mathbb{R}^{++}} = \left(\frac{dx}{x}\right)^2. \tag{4.1}$$

We use here the natural coordinate system induced by the space $\mathbb{R}_+^*$. The tangent space of $\mathbb{R}_+^*$ is $\mathbb{R}$.
The distance associated with this space is given by:

$$d_{\mathbb{R}^{++}}(a,b) = \left|\log\left(\frac{b}{a}\right)\right| \quad \forall a,b \in \mathbb{R}_+^*. \tag{4.2}$$

The formula of the scalar product is given by:

$$\langle e,f\rangle_a = \frac{ef}{a^2} \quad \forall a \in \mathbb{R}_+^*, \forall e,f \in \mathbb{R}. \tag{4.3}$$

We deduce the expression of the squared norm:

$$\|v\|_a^2 = \langle v,v\rangle_a = \left(\frac{v}{a}\right)^2 \quad \forall a \in \mathbb{R}_+^*, \forall v \in \mathbb{R}. \tag{4.4}$$

For all $a \in \mathbb{R}_+^*$, the map defined by:

$$\Phi_a(b) = \frac{b}{a} \tag{4.5}$$

is an isometry over the manifold of strictly positive reals $\mathbb{R}^{++}$. We have in particular $\Phi_a(a) = 1$ and $\Phi_a^{-1} = \Phi_{1/a}$.
The Riemannian logarithm map is defined by:

$$\log_a(b) = a\log\left(\frac{b}{a}\right) \quad \forall a,b \in \mathbb{R}_+^*. \tag{4.6}$$

We can check that we have $d^2_{\mathbb{R}^{++}}(a,b) = \|\log_a(b)\|_a^2$.
The Riemannian exponential map is defined by:

$$\exp_a(v) = a\exp\left(\frac{v}{a}\right) \quad \forall a \in \mathbb{R}_+^*, \forall v \in \mathbb{R}. \tag{4.7}$$

We can check that the Riemannian logarithm map (4.6) and the Riemannian exponential map (4.7) are reciprocal of each other:

$$\log_a(\exp_a(v)) = v \quad \forall a \in \mathbb{R}_+^*, \forall v \in \mathbb{R} \tag{4.8}$$
$$\exp_a(\log_a(b)) = b \quad \forall a,b \in \mathbb{R}. \tag{4.9}$$

The geodesics parameterized by t starting from the point a with an initial velocity v have the equation [82]:

$$\zeta(t,a,v) = \exp_a(tv) = a\exp\left(\frac{tv}{a}\right) \quad \forall t \in \mathbb{R}, \forall a \in \mathbb{R}_+^*, \forall v \in \mathbb{R}. \tag{4.10}$$

Since the Riemannian manifold $\mathbb{R}^{++}$ is of dimension 1, its sectional curvature is not defined.

4.1.2 The Poincaré disk $\mathbb{D}$

We denote $\mathbb{D}$ the Riemannian manifold defined by the complex unit disk $\mathcal{D} := \{z \in \mathbb{C} | \, |z| < 1\}$ endowed with the following metric:

$$ds^2_{\mathbb{D}} = \frac{|dz|^2}{\left(1 - |z|^2\right)^2}. \tag{4.11}$$

We use the natural coordinate system induced by the complex unit disk $\mathcal{D}$. Tangent space is $\mathbb{C}$.
On this manifold, the distance between two points z_1 and z_2 has the following formula:

$$d_{\mathbb{D}}(z_1, z_2) = \frac{1}{2}\left|\log\left(\frac{1 + \left|\frac{z_1 - z_2}{1 - z_1 z_2^*}\right|}{1 - \left|\frac{z_1 - z_2}{1 - z_1 z_2^*}\right|}\right)\right| \quad \forall z_1, z_2 \in \mathcal{D} \tag{4.12}$$

or:

$$d_{\mathbb{D}}(z_1, z_2) = \frac{1}{2}\log\left(\frac{1 + \Delta}{1 - \Delta}\right) \tag{4.13}$$

by noting $\Delta := \left|\frac{z_1 - z_2}{1 - z_1 z_2^*}\right|$, which can also be written:

$$d_{\mathbb{D}}(z_1, z_2) = \operatorname{arctanh}(\Delta). \tag{4.14}$$

The scalar product has the following expression:

$$\langle u, v \rangle_z = \frac{1}{2}\frac{uv^* + vu^*}{\left(1 - |z|^2\right)^2} \quad \forall z \in \mathcal{D}, \forall u, v \in \mathbb{C}. \tag{4.15}$$

We deduce from the scalar product the expression of the squared norm:

$$\|u\|_z^2 = \langle u, u \rangle_z = \frac{|u|^2}{\left(1 - |z|^2\right)^2} \quad \forall z \in \mathcal{D}, \forall u \in \mathbb{C} \tag{4.16}$$

For all $\Omega \in \mathcal{D}$, the map defined by:

$$\Phi_\Omega(\Psi) = \frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi} \quad \forall \Omega, \Psi \in \mathcal{D} \tag{4.17}$$

is an isometry on the Poincaré disk $\mathbb{D}$. In particular, this application sends the point Ω in 0 : $\Phi_\Omega(\Omega) = 0$. This map also has the following property: $\Phi_\Omega^{-1} = \Phi_{-\Omega}$.
The Riemannian logarithm map has the following expression, $\forall \Omega, \Psi \in \mathcal{D}$:

$$\log_\Omega(\Psi) = \left(1 - |\Omega|^2\right) \operatorname{arctanh}\left(\left|\frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi}\right|\right) e^{i\theta} \tag{4.18}$$

with

$$\theta = \arg\left(\frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi}\right). \tag{4.19}$$

The Riemannian exponential map has the following expression, $\forall \mu \in \mathcal{D}, \forall v \in \mathbb{C}$:

$$\exp_\mu(v) = \frac{\left(\mu + e^{i\theta}\right) e^{\left(\frac{2|v|}{1-|\mu|^2}\right)} + \left(\mu - e^{i\theta}\right)}{\left(1 + \bar{\mu}e^{i\theta}\right) e^{\left(\frac{2|v|}{1-|\mu|^2}\right)} + \left(1 - \bar{\mu}e^{i\theta}\right)} \tag{4.20}$$

with $\theta = \arg(v)$.
The geodesic parametrized by t starting from μ with velocity v is given by [6]:

$$\zeta(t, \mu, v) = \frac{\left(\mu + e^{i\theta}\right) e^{\left(\frac{2|v|}{1-|\mu|^2}t\right)} + \left(\mu - e^{i\theta}\right)}{\left(1 + \bar{\mu}e^{i\theta}\right) e^{\left(\frac{2|v|}{1-|\mu|^2}t\right)} + \left(1 - \bar{\mu}e^{i\theta}\right)} \tag{4.21}$$

with $\theta = \arg(v)$.

The geodesic parameterized by the arc length starting from the point μ with an initial velocity vector $v/\|v\|_\mu$ has the equation:

$$\zeta(t,\mu,v) = \exp_\mu(tv) = \frac{(\mu + e^{i\theta})e^{2t} + (\mu - e^{i\theta})}{(1+\bar{\mu}e^{i\theta})e^{2t} + (1-\bar{\mu}e^{i\theta})} \tag{4.22}$$

since $\|v\|_\mu = \frac{|v|}{1-|\mu|^2}$.

The sectional curvature of the Poincaré disk $\mathbb{D}$ is constant at all points of the complex unit disk:

$$K = -4. \tag{4.23}$$

The curvature formula will be demonstrated in the more general case of Siegel space in Section E.7.

We will show in Section D.2 that if we multiply the metric ds^2 of a Riemannian manifold by a constant $c > 0$, then the sectional curvature K is divided by c.

So if we denote:

$$\widetilde{ds}^2 = \frac{4|dz|^2}{\left(1-|z|^2\right)^2} = 4ds^2, \tag{4.24}$$

then the curvature $\widetilde{K}$ of the Riemannian manifold obtained would be, according to Property 44 :

$$\widetilde{K} = \frac{K}{4} = -1. \tag{4.25}$$

4.1.3 The Poincaré polydisk $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$

This has previously been studied in the work of Le Yang [6], [82], in the work of Alice Le Brigant [15] and in the work of Ben Jeuris and Raf Vandebril [45].

We define the product manifold of strictly positive reals and of Poincaré disks as the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ whose Riemannian metric at the point of coordinates $(p_0, \mu_1, \ldots, \mu_{n-1})$ is written:

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n ds^2_{\mathbb{R}^{++}} + \sum_{k=1}^{n-1}(n-k)\, ds^2_{\mathbb{D}} \tag{4.26}$$

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n\frac{dp_0^2}{p_0^2} + \sum_{k=1}^{n-1}(n-k)\frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2}. \tag{4.27}$$

This manifold is called Poincaré polydisk in the work of Alice Le Brigant [15]. We will not use this denomination here not to confuse the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ with the manifold $\mathbb{D}^{n-1}$ used in Chapter 8 to cluster radar time series.

We notice that the Riemannian metric $ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}}$ of the Poincaré polydisk is a weighted sum of metrics studied previously. In this linear combination, the metric defined on the space $\mathbb{R}_+^*$ in Equation (4.1) is used once and the metric defined on the space $\mathcal{D}$ in Equation (4.11) in used $(n-1)$ times. This metric is therefore a product metric. The manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is therefore a product manifold of the manifold $\mathbb{R}^{++}$ studied in Section 4.1.1 and of $(n-1)$ times the manifold $\mathbb{D}$ studied in Section 4.1.2, the metric of each space of the product manifold being multiplied by a constant.

When a product space has a product metric, then the squared distance between two points is the sum of the squared distances of the spaces that compose the product space. Therefore, we have here:

$$\begin{aligned} &d^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}}\left((p_{0,1},\mu_{1,1},...,\mu_{n-1,1}),(p_{0,2},\mu_{1,2},...,\mu_{n-1,2})\right) \\ =&n d^2_{\mathbb{R}^{++}}(p_{0,1},p_{0,2}) + \sum_{l=1}^{n-1}(n-l)\, d^2_{\mathbb{D}}(\mu_{j,1},\mu_{j,2}) \qquad (4.28) \\ =&n\log^2\left(\frac{p_{0,2}}{p_{0,1}}\right) + \sum_{l=1}^{n-1}\frac{n-l}{4}\log^2\left(\frac{1+\left|\frac{\mu_{l,1}-\mu_{l,2}}{1-\mu_{l,1}\mu^*_{l,2}}\right|}{1-\left|\frac{\mu_{l,1}-\mu_{l,2}}{1-\mu_{l,1}\mu^*_{l,2}}\right|}\right). \qquad (4.29) \end{aligned}$$

We recall that the space $\mathcal{T}_n^+$ of Toeplitz Hermintian positive definite matrices is in bijection with the space avec l'espace $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ via Algorithms 13 and 14. We can therefore endow the space $\mathcal{T}_n^+$ with a distance by defining:

$$d^2_T(T_1,T_2) := d^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}}\left((p_{0,1},\mu_{1,1},...,\mu_{n-1,1}),(p_{0,2},\mu_{1,2},...,\mu_{n-1,2})\right) \tag{4.30}$$

where the coefficients $(p_{0,i}, \mu_{1,i}, ..., \mu_{n-1,i})$ are obtained from the matrices T_i for $i = 1, 2$ using Algorithm 13 or Algorithm 14.

The scalar product on the space $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is also a weighted sum of the scalar product defined on $\mathbb{R}^{++}$ in Equation (4.3) and of $(n-1)$ times the scalar product defined on $\mathbb{D}$ in Equation (4.15).

Therefore, here we have $\forall e, f \in \mathbb{R}, \forall p_0 \in \mathbb{R}_+^*, \forall u_1, \dots, u_{n-1}, v_1, \dots, v_{n-1} \in \mathbb{C}, \forall \mu_1, \dots, \mu_{n-1} \in \mathcal{D}$:

$$\begin{aligned} &\left\langle (e, u_1, \dots, u_{n-1}), (f, v_1, \dots, v_{n-1}) \right\rangle_{(p_0, \mu_1, \dots, \mu_{n-1})} \\ =&n\frac{ef}{p_0^2} + \sum_{k=1}^{n-1} \frac{n-k}{2} \frac{u_k v_k^* + v_k u_k^*}{\left(1 - |\mu_k|^2\right)^2}. \end{aligned} \tag{4.31}$$

The squared norm of the space $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is also a weighted sum of the norm defined on $\mathbb{R}^{++}$ in Equation (4.4) and of $(n-1)$ times the norm defined on $\mathbb{D}$ in Equation (4.16).

Hence we have, $\forall e \in \mathbb{R}, \forall p_0 \in \mathbb{R}_+^*, \forall u_1, \dots, u_{n-1} \in \mathbb{C}, \forall \mu_1, \dots, \mu_{n-1} \in \mathcal{D}$:

$$\left\| (e, u_1, \dots, u_{n-1}) \right\|^2_{(p_0, \mu_1, \dots, \mu_{n-1})} = n\frac{e^2}{p_0^2} + \sum_{k=1}^{n-1} (n-k) \frac{|u_k|^2}{\left(1 - |\mu_k|^2\right)^2}. \tag{4.32}$$

We can obtain an isometry on the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ by taking the product of the isometry on $\mathbb{R}^{++}$ defined in Equation (4.5) and of $(n-1)$ isometries on the Poincaré disk (4.17).

The Riemannian logarithm map of the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is the product of the logarithm map of the manifold $\mathbb{R}^{++}$ presented in Equation (4.6) and of $(n-1)$ logarithm maps of the manifold $\mathbb{D}$ presented in Equation (4.18). If we denote $\log_{\mathbb{R}^{++}}$ the logarithm map defined on $\mathbb{R}^{++}$ in Equation (4.6):

$$\log_{\mathbb{R}^{++}} (p_{0,1}, p_{0,2}) = p_{0,1} \log \left(\frac{p_{0,2}}{p_{0,1}} \right), \tag{4.33}$$

and if we denote by $\log_{\mathbb{D}}$ the logarithm map defined on $\mathbb{D}$ in Equation (4.18):

$$\log_{\mathbb{D}} (\mu_{k,1}, \mu_{k,2}) = \left(1 - |\mu_{k,1}|^2\right) \operatorname{arctanh} \left(\left| \frac{\mu_{k,2} - \mu_{k,1}}{1 - \overline{\mu_{k,1}} \mu_{k,2}} \right| \right) e^{i\theta_k} \tag{4.34}$$

with $e^{i\theta_k} = \arg \left(\frac{\mu_{k,2} - \mu_{k,1}}{1 - \overline{\mu_{k,1}} \mu_{k,2}} \right)$,

then we have, $\forall p_{0,1}, p_{0,2} \in \mathbb{R}_+^*, \forall \mu_{1,1}, ..., \mu_{n-1,1}, \mu_{1,2}, ..., \mu_{n-1,2} \in \mathcal{D}$:

$$\begin{aligned} &\log_{\mathbb{R}^{++} \times \mathbb{D}^{n-1}} \left((p_{0,1}, \mu_{1,1}, ..., \mu_{n-1,1}), (p_{0,2}, \mu_{1,2}, ..., \mu_{n-1,2}) \right) \\ &= \left(\log_{\mathbb{R}^{++}} (p_{0,1}, p_{0,2}), \log_{\mathbb{D}} (\mu_{1,1}, \mu_{1,2}), \dots, \log_{\mathbb{D}} (\mu_{n-1,1}, \mu_{n-1,2}) \right). \end{aligned} \tag{4.35}$$

The Riemannian exponential map of the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is also the product of the exponential map of the manifold $\mathbb{R}^{++}$ presented in Equation (4.7) and of $(n-1)$ exponential maps of the manifold $\mathbb{D}$ presented in Equation (4.20).

If we denote $\exp_{\mathbb{R}^{++}}$ the exponential map defined on $\mathbb{R}^{++}$ in Equation (4.7):

$$\exp_{\mathbb{R}^{++}} (p_0, v_0) = p_0 \exp \left(\frac{v_0}{p_0} \right) \quad \forall p_0 \in \mathbb{R}_+^*, \forall v_0 \in \mathbb{R}, \tag{4.36}$$

and if we denote by $\exp_{\mathbb{D}}$ the exponential map defined on $\mathbb{D}$ in Equation (4.20):

$$\exp_{\mathbb{D}} (\mu_k, v_k) = \frac{\left(\mu_k + e^{i\theta_k}\right) e^{\left(\frac{2|v_k|}{1-|\mu_k|^2}\right)} + \left(\mu_k - e^{i\theta_k}\right)}{\left(1 + \overline{\mu_k} e^{i\theta_k}\right) e^{\left(\frac{2|v_k|}{1-|\mu_k|^2}\right)} + \left(1 - \overline{\mu_k} e^{i\theta_k}\right)} \tag{4.37}$$

with $\theta_k = \arg (v_k)$,

then we have, $\forall p_0 \in \mathbb{R}_+^*, \forall v_0 \in \mathbb{R}, \forall \mu_1, ..., \mu_{n-1} \in \mathcal{D}, \forall v_1, ..., v_{n-1} \in \mathbb{C}$:

$$\begin{aligned} &\exp_{\mathbb{R}^{++} \times \mathbb{D}^{n-1}} \left((p_0, \mu_1, ..., \mu_{n-1}), (v_0, v_1, ..., v_{n-1}) \right) \\ &= \left(\exp_{\mathbb{R}^{++}} (p_0, v_0), \exp_{\mathbb{D}} (\mu_1, v_1), \dots, \exp_{\mathbb{D}} (\mu_{n-1}, v_{n-1}) \right). \end{aligned} \tag{4.38}$$

The geodesics of the produt manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ are also the product of the geodesics of the manifold $\mathbb{R}^{++}$ presented in Equation (4.10) and of $(n-1)$ geodesics of the manifold $\mathbb{D}$ studied in Equation (4.21):

$$\begin{aligned} &\zeta_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}}\left(t,\left(p_0,\mu_1,...,\mu_{n-1}\right),\left(v_0,v_1,...,v_{n-1}\right)\right) \\ &=\exp_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}}\left(\left(p_0,\mu_1,...,\mu_{n-1}\right),\left(tv_0,tv_1,...,tv_{n-1}\right)\right) \\ &=\left(\exp_{\mathbb{R}^{++}}\left(p_0,tv_0\right),\exp_{\mathbb{D}}\left(\mu_1,tv_1\right),\ldots,\exp_{\mathbb{D}}\left(\mu_{n-1},tv_{n-1}\right)\right). \end{aligned} \tag{4.39}$$

Let us now study the sectional curvature K of the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$. We recall that the sectional curvature of the manifold $\mathbb{R}^{++}$ is not defined since it is a one-dimensional manifold. According to Equation (4.23), the Poincaré disk $\mathbb{D}$ has a constant sectional curvature equal to -4. We also recall that when we multiply the metric ds^2 of a Riemannian manifold by a constant c, its curvature is divided by c. The manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is therefore the product of a manifold of undefined curvature and $(n-1)$ manifolds of respective curvatures $-4/(n-k)$ for k ranging from 1 to $(n-1)$.

We can then obtain bounds of the sectional curvature of the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ using Lemma 1 proved in [82].

Property 20. *The sectional curvature of the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ is bounded between -4 and 0:*

$$-4 \leqslant K \leqslant 0 \tag{4.40}$$

Proof. The property is true according to Lemma 1 since all the manifolds constituting the product manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ either have an undefined sectional curvature or a sectional curvature bounded between -4 and 0. □

4.2 Manifolds associated with the vectorial autoregressive model

The Riemman manifolds presented in this section come from the representation spaces of multidimensional complex stationary centered Gaussian autoregressive time series presented in Section 2.2 and endowed with a Riemannian metric inspired by information geometry 3.2.3. The three manifolds presented in this section generalize the manifolds presented in Section 4.1 in higher dimension.

4.2.1 The HPD manifold

The space of Hermitian Positive Definite matrices $\mathcal{H}_N^+$ is defined as:

$$\mathcal{H}_N^+ := \left\{A \in \mathbb{C}^{n\times n} \mid x^H A x > 0, \forall x \in \mathbb{C}^n \backslash \{0\}\right\}. \tag{4.41}$$

This definition of the Hermitian Positive Definite matrices is equivalent to the fact that the matrix A is Hermitian and has strictly positive eigenvalues. We will also denote by $A > 0$ a matrix $A \in \mathcal{H}_N^+$.

We endow the space $\mathcal{H}_N^+$ with the following metric:

$$ds^2_{\mathbb{H}_N^{++}} = \operatorname{trace}\left(P^{-1}\, dP\, P^{-1}\, dP\right) \quad \forall P \in \mathcal{H}_n^+. \tag{4.42}$$

We use here the natural coordinate system induced by $\mathcal{H}_N^+$. The tangent space of $\mathcal{H}_N^+$ is the space $\mathcal{H}_N$ of Hermitian matrices. The metric $ds^2_{\mathbb{H}_N^{++}}$ endow the space $\mathcal{H}_N^+$ with a Riemannian metric.

On this Riemannian manifold, the distance between two matrices A and B has the following expression:

$$d_{\mathbb{H}_N^{++}}(A,B) = \left\|\log\left(A^{-1/2}BA^{-1/2}\right)\right\|_2 \quad \forall A,B \in \mathcal{H}_N^+. \tag{4.43}$$

The scalar product has the following expression:

$$\langle E,F\rangle_A = \operatorname{trace}\left(A^{-1}EA^{-1}F\right) \quad \forall A \in \mathcal{H}_N^+, \forall E,F \in \mathcal{H}_N. \tag{4.44}$$

The expression of the squared norm of the vector E expressed at the point A is therefore:

$$\|E\|_A^2 = \langle E,E\rangle_A = \operatorname{trace}\left(A^{-1}EA^{-1}E\right) \quad \forall A \in \mathcal{H}_N^+, \forall E \in \mathcal{H}_N. \tag{4.45}$$

The following map is an isometry on the space $\mathcal{H}_N^+$ for the metric described previously:

$$\Phi_A(B) = A^{-1/2}BA^{-1/2} \quad \forall A,B \in \mathcal{H}_N^+. \tag{4.46}$$

We have in particular $\Phi_A(A) = I_N$ and $\Phi_A^{-1} = \Phi_{A^{-1}}$.

The differential of isometry Φ has the following expression:

$$D\Phi_A(B)[h] = A^{-1/2}hA^{-1/2} \quad \forall A,B \in \mathcal{H}_N^+, \forall h \in \mathcal{H}_N. \tag{4.47}$$

The Riemannian logarithm map has the expression:

$$\log_A(B) = A^{1/2} \log\left(A^{-1/2} B A^{-1/2}\right) A^{1/2} \quad \forall A, B \in \mathcal{H}_N^+ \tag{4.48}$$

where the application $\log$ to the right of the equation represents the standard matrix logarithm map.
The Riemannian exponential map has the expression:

$$\exp_A(V) = A^{1/2} \exp\left(A^{-1/2} V A^{-1/2}\right) A^{1/2} \quad \forall A \in \mathcal{H}_N^+, \forall V \in \mathcal{H}_N \tag{4.49}$$

where the map $\exp$ to the right of the equation represents the standard matrix exponential map.
The geodesics parameterized by t starting from the point A with an initial velocity vector V have for equation:

$$\zeta(t, A, V) = \exp_A(tV) = A^{1/2} \exp\left(tA^{-1/2} V A^{-1/2}\right) A^{1/2} \quad \forall t \in \mathbb{R}, \forall A \in \mathcal{H}_N^+, \forall V \in \mathcal{H}_N. \tag{4.50}$$

The sectional curvature of the manifold $\mathbb{H}_N^{++}$ given for the tangent space of the identity I_N has the following expression:

$$K(\sigma) = \frac{1}{4} \operatorname{trace}\left((E_1 E_2 - E_2 E_1)^2\right) \tag{4.51}$$

where E_1 and E_2 form an orthonormal basis of the tangent plane at the identity σ.

Proof. We start by computing the sectional curvature in the identity I_N for all plane σ. Thanks to the isometry Φ (4.46) and its differential (4.47), we can then compute the sectional curvature at any point.

We place ourselves within the framework of Theorem 7.

Let P be a point whose normal coordinates at the identity I_N are $V = x_1 E_1 + x_2 E_2$, the vectors E_1 and E_2 forming an orthonormal basis of the plane σ.

We have $P = \exp_{I_N}(V) = \exp(V)$ where the exponential map in the last member of the equality represents the matrix exponential map.

We have:

$$P = Id + V + \frac{1}{2}V^2 + \frac{1}{6}V^3 + O\left(V^3\right). \tag{4.52}$$

By differentiating, we obtain:

$$dP = dV + \frac{1}{2}(VdV + dVV) + \frac{1}{6}\left(V^2 dV + VdVV + dVV^2\right) + O\left(V^3\right). \tag{4.53}$$

Since $P^{-1} = \exp(-V)$, we have:

$$P^{-1} = Id - V + \frac{1}{2}V^2 - \frac{1}{6}V^3 + O\left(V^3\right). \tag{4.54}$$

Using the previous two equations, we have:

$$P^{-1}dP = dV + \frac{1}{2}VdV + \frac{1}{2}dVV + \frac{1}{6}V^2 dV - \frac{1}{3}VdVV + \frac{1}{6}dVV^2 + O\left(V^3\right) \tag{4.55}$$

Hence, we have:

$$\begin{aligned} P^{-1}dPP^{-1}dP =& dV^2 + \frac{1}{2}dV^2 V - \frac{1}{2}VdV^2 \\ &+ \frac{1}{3}dVV^2 dV - \frac{1}{3}dVVdVV \\ &+ \frac{1}{6}dV^2 V^2 + \frac{1}{6}V^2 dV^2 - \frac{1}{3}VdVVdV \\ &+ \frac{1}{4}(dVV - VdV)^2 + O\left(V^3\right) \end{aligned} \tag{4.56}$$

Using the linearity of the trace and the property of invariance of the trace by permutation, i.e. $\operatorname{trace}(AB) = \operatorname{trace}(BA)$, we obtain:

$$\operatorname{trace}\left(P^{-1}dPP^{-1}dP\right) = \operatorname{trace}\left(dV^2\right) - \frac{1}{12}\operatorname{trace}\left((VdV - dVV)^2\right) \tag{4.57}$$

We recall that $V := x_1 E_1 + x_2 E_2$.

Since the vectors E_1 and E_2 are orthogonal and normalized, we have $\text{trace}\left(dV^2\right) = dx_1^2 + dx_2^2$.
We deduce that the metric $ds^2_{\mathbb{H}_N^{++}}$ at the identity I_N has the expression:

$$ds^2_{\mathbb{H}_N^{++}} = \text{trace}\left(P^{-1}\, dP\, P^{-1}\, dP\right) \tag{4.58}$$

$$= dx_1^2 + dx_2^2 - \frac{1}{12}\,\text{trace}\left(\left(E_1E_2 - E_2E_1\right)^2\right)\left(x_1dx_2 - x_2dx_1\right)^2 + O\left(x^3\right). \tag{4.59}$$

According to Theorem 7, the sectional curvature at the identity is therefore:

$$K\left(\sigma\right) = \frac{1}{4}\,\text{trace}\left(\left(E_1E_2 - E_2E_1\right)^2\right). \tag{4.60}$$

□

We note that:

$$K\left(\sigma\right) = -\frac{1}{4}\left\|E_1E_2 - E_2E_1\right\|_2^2 \tag{4.61}$$

Therefore, we have $K\left(\sigma\right) \leqslant 0$.
According to the triangular inequality, we also have:

$$\left\|E_1E_2 - E_2E_1\right\|_2 \leqslant \left\|E_1E_2\right\|_2 + \left\|E_2E_1\right\|_2 \leqslant \left\|E_1\right\|\left\|E_2\right\|_2 + \left\|E_2\right\|\left\|E_1\right\|_2 = 2, \tag{4.62}$$

which implies that $K\left(\sigma\right) \geqslant -1$.
Finally, we have:

$$-1 \leqslant K\left(\sigma\right) \leqslant 0 \quad \forall\sigma. \tag{4.63}$$

4.2.2 The Siegel disk

The Siegel manifold generalize the Poincaré disk studied in Section 4.1.2 to complex matrices. This section is an overview of Appendix E dedicated to the Siegel disk in which a few additional properties and many proofs are given. The Riemannian logarithm map, the Riemannian exponential map and the sectional curvature of the Siegel manifold are the main contributions of this thesis. These novelties have been summarized in our previous work [21].

4.2.2.1 Definition

Definition 15. *The Siegel disk is defined as the set of complex matrices M of shape $N \times N$ with singular values lower than one, which can also be written:*

$$\mathcal{SD}_N = \left\{M \in \mathbb{C}^{N\times N} \mid I - MM^H > 0\right\} \tag{4.64}$$

or equally:

$$\mathcal{SD}_N = \left\{M \in \mathbb{C}^{N\times N} \mid I - M^HM > 0\right\} \tag{4.65}$$

We use the partial ordering of the set complex matrices: we note $A > B$ when the difference $A - B$ is a positive definite matrix.

Note that another definition of the Siegel disk also exists in other papers [54], imposing an additional symmetry condition on the matrix M: $M = M^T$. We will note require the symmetry condition in our work.

Property 21. *The Siegel disk can also be defined as the set of complex matrices M with a linear operator norm lower than one: $\mathcal{SD}_N = \left\{M \in \mathbb{C}^N, \left|\left|\left|M\right|\right|\right| < 1\right\}$, where $\left|\left|\left|M\right|\right|\right| = \sup_{X\in\mathbb{C}^N, \|X\|=1}\left(\|MX\|\right)$.*

4.2.2.2 The metric

We endow the Siegel space $\mathcal{SD}_N$ with the following metric:

$$ds^2_{\mathbb{SD}_N} = \text{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H\right). \tag{4.66}$$

The expression of the square of the distance is given for all $\Omega, \Psi \in \mathcal{SD}_N$ by:

$$d^2_{\mathbb{SD}_N}(\Omega,\Psi) = \frac{1}{4}\operatorname{trace}\left(\log^2\left(\frac{I+C^{1/2}}{I-C^{1/2}}\right)\right) \tag{4.67}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}^2\left(C^{1/2}\right)\right) \tag{4.68}$$

with $C = (\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Psi^H\right)^{-1}$.
Another distance named the *Kobayashi distance measure* d_K on the Siegel disk $\mathcal{SD}_N$ is also given in [45].
The expression of the scalar product is then, $\forall\Omega\in\mathcal{SD}_N, \forall v,w\in\mathbb{C}^{N\times N}$:

$$\langle v,w\rangle_\Omega = \frac{1}{2}\operatorname{trace}\left(\left(I-\Omega\Omega^H\right)^{-1}v\left(I-\Omega^H\Omega\right)^{-1}w^H\right) + \frac{1}{2}\operatorname{trace}\left(\left(I-\Omega\Omega^H\right)^{-1}w\left(I-\Omega^H\Omega\right)^{-1}v^H\right). \tag{4.69}$$

The norm of a vector belonging to the tangent space is therefore:

$$\|v\|^2_\Omega = \operatorname{trace}\left(\left(I-\Omega\Omega^H\right)^{-1}v\left(I-\Omega^H\Omega\right)^{-1}v^H\right). \tag{4.70}$$

4.2.2.3 The isometry

According to [45], the following function is an isometry of the Siegel space for the distance defined in Equation (4.67) :

$$\Phi_\Omega(\Psi) = \left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}. \tag{4.71}$$

We will prove this result in Theorem 9.
The differential of the isometry Φ has the following expression:

Property 22. *The differential of the isometry Φ has the following expression:*

$$D\Phi_\Omega(\Psi)[h] = \left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}h\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \tag{4.72}$$

We will prove Equation (4.72) in the proof of Property 46.

Property 23. *The inverse of the function Φ described in Equation (4.71) also has the following expression:*

$$\Phi^{-1}_\Omega(\Psi) = \Phi_{-\Omega}(\Psi) \tag{4.73}$$

We will present three proofs of Equation (4.73) in the proof of Property 49.

4.2.2.4 The Riemannian logarithm map

To compute the Riemannian logarithm map at a point Ω the key idea here is to transport the problem at zero, compute a certain logarithm at zero and transport the result back to Ω. This idea is illustrated on Figure 4.1. If we want to compute the logartithm map: $\log_\Omega(\Psi)$, we first transport both Ω and Ψ using the isometry Φ_Ω (4.71). The point Ω is sent to zero, and we denote Ψ' the image of Ψ by Φ_Ω :

$$\Psi' := \Phi_\Omega(\Psi) = \left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}. \tag{4.74}$$

Then we compute the logarithm map at zero $\log_0\left(\Psi'\right)$:

$$V' := \log_0\left(\Psi'\right) = \operatorname{arctanh}(X)\,X^{-1}\Psi' \quad \text{where } X = \left(\Psi'\Psi'^H\right)^{1/2}. \tag{4.75}$$

Finally, we transport back the logarithm to the point Ω using the differential of the isometry Φ given in Equation (E.17):

$$V := \log_\Omega(\Psi) = D\Phi_{-\Omega}(0)\left[V'\right] = \left(I-\Omega\Omega^H\right)^{1/2}V'\left(I-\Omega^H\Omega\right)^{1/2} \tag{4.76}$$

We summarize the logarithm map in the following equation:

$$\log_\Omega(\Psi) = \left(I-\Omega\Omega^H\right)^{1/2}V'\left(I-\Omega^H\Omega\right)^{1/2} \tag{4.77}$$
$$\left[\begin{array}{l} V' = \operatorname{arctanh}(X)\,X^{-1}\Psi' \quad \text{where } X = \left(\Psi'\Psi'^H\right)^{1/2} \\ \Psi' = \left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \end{array}\right.$$

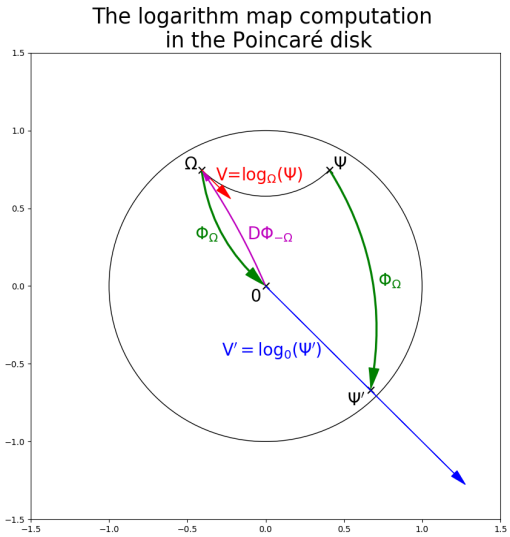

Figure 4.1: The Poincaré disk logarithm map computation

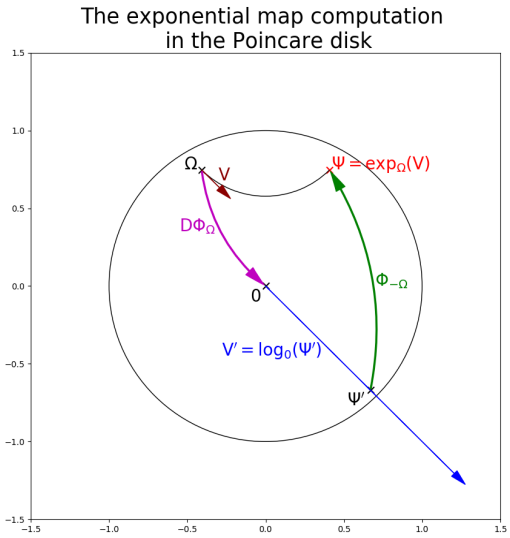

Figure 4.2: The Poincaré disk exponential map computation

4.2.2.5 The Riemannian exponential map

To compute the Riemannian exponential map at a point Ω the key idea here is to transport the problem at zero (as for the logarithm), compute a certain exponential at zero and transport the result back to Ω. This idea is illustrated on Figure 4.2. If we want to compute the exponential map: $\exp_\Omega(V)$, we first transport the vector V at zero using the differential of the isometry Φ given in Equation (E.17):

$$\begin{aligned} V' :=& D\Phi_\Omega(\Omega)[V] && (4.78)\\ =& \left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Omega\Omega^H\right)^{-1} V \left(I-\Omega^H\Omega\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} && (4.79)\\ =& \left(I-\Omega\Omega^H\right)^{-1/2} V \left(I-\Omega^H\Omega\right)^{-1/2}. && (4.80)\end{aligned}$$

Then we compute the exponential map at zero $\exp_0\left(V'\right)$:

$$\Psi' := \exp_0\left(V'\right) = \tanh(Y)\, Y^{-1} V' \quad \text{where } Y = \left(V'V'^{\,H}\right)^{1/2}. \tag{4.81}$$

Finally, we transport back the exponential to the point Ω using the isometry $\Phi_{-\Omega}$ which is the inverse of isometry Φ_Ω (see Property 49) and transport the point 0 back to Ω and the point Ψ' back to $\exp_\Omega(V)$:

$$\begin{aligned} \Psi :=& \exp_\Omega(V) && (4.82)\\ =& \Phi_{-\Omega}\left(\Psi'\right) && (4.83)\\ =& \left(I-\Omega\Omega^H\right)^{-1/2}\left(\Psi'+\Omega\right)\left(I+\Omega^H\Psi'\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} && (4.84)\end{aligned}$$

We summarize the exponential map in the following equation:

$$\exp_\Omega(V) = \left(I-\Omega\Omega^H\right)^{-1/2}\left(\Psi'+\Omega\right)\left(I+\Omega^H\Psi'\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \tag{4.85}$$
$$\left[\begin{array}{l} \Psi' = \tanh(Y)\, Y^{-1} V' \quad \text{where } Y = \left(V'V'^{\,H}\right)^{1/2} \\ V' = \left(I-\Omega\Omega^H\right)^{-1/2} V \left(I-\Omega^H\Omega\right)^{-1/2} \end{array}\right.$$

4.2.2.6 The geodesics

The expression the geodesics can be obtained using the exponential map: the geodesic starting from Ω with velocity V is given by the application :

$$\zeta(t,\Omega,V) = \exp_\Omega(tV). \tag{4.86}$$

4.2.2.7 The symmetric Siegel disk

We defined the Siegel disk in Definition 16 as the set a complex matrices with singular values lower than one: $\mathcal{SD}_N = \left\{M \in \mathbb{C}^N, I - MM^H > 0\right\}$. We recall that another definition of the Siegel disk also exists in other papers [54], imposing an additional symmetry condition on the matrix M: $M = M^T$. We will show in Section E.6 that the symmetric Siegel disk is a totally flat submanifold of the Siegel disk.

Hence the formula of the logarithm map E.4.2 and the exponential map E.5.2 computed in previous sections are still meaningful when working in the submanifold of symmetric matrices.

4.2.2.8 The sectional curvature

We first focus on the sectional curvature at 0. We can then obtain the sectional curvature at any point using the isometry Φ defined in Equation (4.71) and its differential E.17. Indeed, the sectional curvature at the point Ω defined by the tangent vectors E_1 and E_2 is equal to the sectional curvature at the point 0 defined by the vectors $D\Phi_\Omega(\Omega)[E_1]$ and $D\Phi_\Omega(\Omega)[E_2]$.

Let σ be a section defined at the point $\Omega = 0$ by two orthonormal matrices E_1 and E_2.

Theorem 5. *The sectional curvature at zero of the plan σ defined by E_1 and E_2 has the following expression:*

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1E_2^H - E_2E_1^H\right\|^2 + \left\|E_1^HE_2 - E_2^HE_1\right\|^2\right) \tag{4.87}$$

This theorem will be proved in Section E.7.1.

Corollary 1.

$$-4 \leqslant K(\sigma) \leqslant 0 \tag{4.88}$$

We recall that the Siegel space is a generalization of the Poincaré disk $\mathbb{D}$ defined as the complex unit disk endowed with the metric:

$$ds^2_{\mathbb{D}} = \frac{|d\mu_k|^2}{\left(1-|\mu_k|^2\right)^2} \tag{4.89}$$

If we perform Theorem 10 on the orthonormal basis vectors $E_1 = 1$ and $E_2 = i$, we find that the curvature of the Poincaré disk $\mathbb{D}$ is equal to -4.

4.2.3 The HPD Siegel product

The HPD Siegel product is a generalization of the manifold $\mathbb{R}_+^* \times \mathbb{D}^{n-1}$ studied in Section 4.1.3 to multidimensional signals.

We endow the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ with the following metric [11,45]:

$$ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} = n ds^2_{\mathbb{H}_N^{++}} + \sum_{l=1}^{n-1}(n-l)\, ds^2_{\mathbb{SD}_N} \tag{4.90}$$

$$\begin{aligned} ds^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}} =& n \operatorname{trace}\left(P_0^{-1}\, dP_0\, P_0^{-1}\, dP_0\right) \\ &+ \sum_{l=1}^{n-1}(n-l) \operatorname{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1} d\Omega_l \left(I-\Omega_l^H\Omega_l\right)^{-1} d\Omega_l^H\right). \end{aligned} \tag{4.91}$$

The expression of the distance between two points is then:

$$\begin{aligned} & d^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}}\left((P_{0,1},\Omega_{1,1},...,\Omega_{n-1,1}),(P_{0,2},\Omega_{1,2},...,\Omega_{n-1,2})\right) \\ =& n d^2_{\mathbb{H}_N^{++}}(P_{0,1},P_{0,2}) + \sum_{l=1}^{n-1}(n-l)\, d^2_{\mathbb{SD}_N}(\Omega_{l,1},\Omega_{l,2}) \qquad (4.92) \\ =& n \left\|\log\left(P_{0,1}^{-1/2}\, P_{0,2}\, P_{0,1}^{-1/2}\right)\right\|_F^2 + \qquad (4.93) \\ & \sum_{l=1}^{n-1}\frac{n-l}{4} \operatorname{trace}\left(\log^2\left(\frac{1+C_l^{1/2}}{1-C_l^{1/2}}\right)\right) \\ & \left[\begin{array}{c} C_l = (\Omega_{l,2}-\Omega_{l,1})\left(I-\Omega_{l,1}^H\,\Omega_{l,2}\right)^{-1} \\ \left(\Omega_{l,2}^H-\Omega_{l,1}^H\right)\left(I-\Omega_{l,1}\,\Omega_{l,2}^H\right)^{-1} \end{array}\right. \end{aligned}$$

We recall that the space $\mathcal{B}_{n,N}^+$ of Block-Toeplitz Hermitian positive definite matrices is in bijection with the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ via the algorithms 3 and 2. We can therefore endow the space $\mathcal{B}_{n,N}^+$ with a distance by defining:

$$d^2_{BT}\left(\widetilde{T}_1,\widetilde{T}_2\right) := d^2_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}}\left((P_{0,1},\Omega_{1,1},...,\Omega_{n-1,1}),(P_{0,2},\Omega_{1,2},...,\Omega_{n-1,2})\right) \tag{4.94}$$

where the coefficients $(P_{0,i},\Omega_{1,i},...,\Omega_{n-1,i})$ are obtained from the matrices $\widetilde{T}_i$ for $i=1,2$ using Algorithm 3. The coefficients $(P_{0,i},\Omega_{1,i},...,\Omega_{n-1,i})$ can also be computed using Algorithm 2 in the particular case where the matrix $\widetilde{T}_i$ has Toeplitz structured blocks (the matrix $\widetilde{T}_i$ is then a Toeplitz-Block Block-Toeplitz HPD matrix).

The scalar product on the space $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$ is also a weighted sum of the scalar product defined on $\mathcal{H}_N^+$ in Equation (4.44) and of $(n-1)$ times the scalar product defined on $\mathcal{SD}_N$ in Equation (4.69).

Therefore, we have $\forall E,F \in \mathcal{H}_N, \forall P_0 \in \mathcal{H}_N^+, \forall U_1,\ldots,U_{n-1},V_1,\ldots,V_{n-1} \in \mathbb{C}^{N\times N}, \forall \Omega_1,\ldots,\Omega_{n-1} \in \mathcal{SD}_N$:

$$
\begin{aligned}
&\left\langle \left(E, U_1, \ldots, U_{n-1}\right), \left(F, V_1, \ldots, V_{n-1}\right)\right\rangle_{(P_0, \Omega_1, \ldots, \Omega_{n-1})} \\
=&n \operatorname{trace}\left(P_0^{-1} E P_0^{-1} F\right) \\
&+ \sum_{k=1}^{n-1} \frac{n-k}{2} \left(\operatorname{trace}\left(\left(I - \Omega_k \Omega_k^H\right)^{-1} U_k \left(I - \Omega_k^H \Omega_k\right)^{-1} V_k^H\right)\right. \\
&\qquad \left. + \operatorname{trace}\left(\left(I - \Omega_k \Omega_k^H\right)^{-1} V_k \left(I - \Omega_k^H \Omega_k\right)^{-1} U_k^H\right)\right).
\end{aligned}
\tag{4.95}
$$

The squared norm on the space $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ is therefore a weighted sum of the norm defined on $\mathcal{H}_N^+$ in Equation (4.45) and of $(n-1)$ times the norm defined on $\mathcal{SD}_N$ in Equation (4.70).

Hence, we have $\forall E \in \mathcal{H}_N, \forall P_0 \in \mathcal{H}_N^+, \forall U_1, \ldots, U_{n-1} \in \mathbb{C}^{N \times N}, \forall \Omega_1, \ldots, \Omega_{n-1} \in \mathcal{SD}_N$:

$$
\begin{aligned}
&\left\|\left(E, U_1, \ldots, U_{n-1}\right)\right\|^2_{(P_0, \Omega_1, \ldots, \Omega_{n-1})} \\
=&n \operatorname{trace}\left(P_0^{-1} E P_0^{-1} E\right) \\
&+ \sum_{k=1}^{n-1} (n-k) \operatorname{trace}\left(\left(I - \Omega_k \Omega_k^H\right)^{-1} U_k \left(I - \Omega_k^H \Omega_k\right)^{-1} U_k^H\right).
\end{aligned}
\tag{4.96}
$$

We can obtain an isometry on the product manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ by taking the product of the isometry on $\mathbb{H}_N^{++}$ defined in Equation (4.46) and of $(n-1)$ isometries on the Siegel disk (4.71).

The Riemannian logarithm map of the product manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ is the product of the logarithm map of the manifold $\mathbb{H}_N^{++}$ presented in Equation (4.48) and of $(n-1)$ logarithm maps of the manifold $\mathbb{SD}_N$ presented in Equation (4.77). If we denote $\log_{\mathbb{H}_N^{++}}$ the logarithm map defined on $\mathbb{H}_N^{++}$ in Equation (4.48):

$$
\log_{\mathbb{H}_N^{++}}\left(P_{0,1}, P_{0,2}\right) = P_{0,1}^{1/2} \log\left(P_{0,1}^{-1/2} P_{0,2} A^{-1/2}\right) P_{0,1}^{1/2} \quad \forall P_{0,1}, P_{0,2} \in \mathcal{H}_N^+,
\tag{4.97}
$$

and if we denote $\log_{\mathbb{SD}_N}$ the logarithm map defined on $\mathbb{SD}_N$ in Equation (4.77) :

$$
\begin{aligned}
&\log_{\mathbb{SD}_N}\left(\Omega_{k,1}, \Omega_{k,2}\right) \\
=&\left(I - \Omega_{k,1}\Omega_{k,1}^H\right)^{1/2} V_k^{'} \left(I - \Omega_{k,1}^H \Omega_{k,1}\right)^{1/2} \\
&\left[\begin{array}{l}
V_k^{'} = \operatorname{arctanh}\left(X_k\right) X_k^{-1} \Psi_k^{'} \quad \text{where } X_k = \left(\Psi_k^{'} \Psi_k^{'H}\right)^{1/2} \\
\Psi_k^{'} = \left(I - \Omega_{k,1}\Omega_{k,1}^H\right)^{-1/2} \left(\Omega_{k,2} - \Omega_{k,1}\right) \left(I - \Omega_{k,1}^H \Omega_{k,2}\right)^{-1} \left(I - \Omega_{k,1}^H \Omega_{k,1}\right)^{1/2}
\end{array}\right.
\end{aligned}
\tag{4.98}
$$

then we have, $\forall P_{0,1}, P_{0,2} \in \mathcal{H}_N^+, \forall \Omega_{1,1}, ..., \Omega_{n-1,1}, \Omega_{1,2}, ..., \Omega_{n-1,2} \in \mathcal{SD}_N$:

$$
\begin{aligned}
&\log_{\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}}\left(\left(P_{0,1}, \Omega_{1,1}, ..., \Omega_{n-1,1}\right), \left(P_{0,2}, \Omega_{1,2}, ..., \Omega_{n-1,2}\right)\right) \\
=&\left(\log_{\mathbb{H}_N^{++}}\left(P_{0,1}, P_{0,2}\right), \log_{\mathbb{SD}_N}\left(\Omega_{1,1}, \Omega_{1,2}\right), \ldots, \log_{\mathbb{SD}_N}\left(\Omega_{n-1,1}, \Omega_{n-1,2}\right)\right).
\end{aligned}
\tag{4.99}
$$

The Riemannian exponential map of the product manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ is also the product of the exponential map of the manifold $\mathbb{H}_N^{++}$ presented in Equation (4.49) and of $(n-1)$ exponential maps of the manifold $\mathbb{SD}_N$ presented in Equation (4.85).

If we denote by $\exp_{\mathbb{H}_N^{++}}$ the exponential map defined on $\mathcal{H}_N^+$ in Equation (4.49):

$$
\exp_{\mathbb{H}_N^{++}}\left(P_0, V_0\right) = P_0^{1/2} \exp\left(P_0^{-1/2} V_0 P_0^{-1/2}\right) P_0^{1/2} \quad \forall P_0 \in \mathcal{H}_N^+, \forall V_0 \in \mathcal{H}_N,
\tag{4.100}
$$

and if we denote by $\exp_{\mathbb{SD}_N}$ the exponential map defined on $\mathcal{SD}_N$ in Equation (4.85):

$$
\begin{aligned}
&\exp_{\mathbb{SD}_N}\left(\Omega_k, V_k\right) = \left(I - \Omega_k \Omega_k^H\right)^{-1/2} \left(\Psi_k^{'} + \Omega_k\right) \left(I + \Omega_k^H \Psi_k^{'}\right)^{-1} \left(I - \Omega_k^H \Omega_k\right)^{1/2} \\
&\left[\begin{array}{l}
\Psi_k^{'} = \tanh\left(Y_k\right) Y_k^{-1} V_k^{'} \quad \text{where } Y_k = \left(V_k^{'} V_k^{'H}\right)^{1/2} \\
V_k^{'} = \left(I - \Omega_k \Omega_k^H\right)^{-1/2} V_k \left(I - \Omega_k^H \Omega_k\right)^{-1/2}
\end{array}\right.
\end{aligned}
\tag{4.101}
$$

then we have, $\forall P_0 \in \mathcal{H}_N^+, \forall V_0 \in \mathcal{H}_N, \forall \Omega_1, ..., \Omega_{n-1} \in \mathcal{SD}_N, \forall V_1, ..., V_{n-1} \in \mathbb{C}^{N \times N}$:

$$\begin{aligned}
&\exp_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}}\left(\left(P_0,\Omega_1,...,\Omega_{n-1}\right),\left(V_0,V_1,...,V_{n-1}\right)\right)\\
&=\left(\exp_{\mathbb{H}_N^{++}}\left(P_0,V_0\right),\exp_{\mathbb{SD}_N}\left(\Omega_1,V_1\right),\ldots,\exp_{\mathbb{SD}_N}\left(\Omega_{n-1},V_{n-1}\right)\right).
\end{aligned}\tag{4.102}$$

The geodesics of the product manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$ are also the product of the geodesics of the manifold $\mathbb{H}_N^{++}$ presented in Equation (4.50) and of $(n-1)$ geodesics of the manifold $\mathbb{SD}_N$ presented in Equation (4.86):

$$\begin{aligned}
&\zeta_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}}\left(t,\left(P_0,\Omega_1,...,\Omega_{n-1}\right),\left(V_0,V_1,...,V_{n-1}\right)\right)\\
&=\exp_{\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}}\left(\left(P_0,\Omega_1,...,\Omega_{n-1}\right),\left(tV_0,tV_1,...,tV_{n-1}\right)\right)\\
&=\left(\exp_{\mathbb{H}_N^{++}}\left(P_0,tV_0\right),\exp_{\mathbb{SD}_N}\left(\Omega_1,tV_1\right),\ldots,\exp_{\mathbb{SD}_N}\left(\Omega_{n-1},tV_{n-1}\right)\right).
\end{aligned}\tag{4.103}$$

Let us now study the sectional curvature K of the product manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$. We recall that the manifold $\mathbb{H}_N^{++}$ has a sectional curvature bounded between -1 and 0 (4.63). According to Equation (4.88), the Siegel manifold $\mathbb{SD}_N$ has a sectional curvature bounded between -4 and 0. We also recall that when we multiply the metric ds^2 of a manifold by a constant c, its curvature is divided by c. The Riemannian manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$ is therefore the product of a manifold of curvature bounded between -1 and 0 and of $(n-1)$ manifolds whose respective curvature are bounded between $-4/(n-k)$ and 0 for k ranging from 1 to $(n-1)$. We can then have a bounding of the sectional curvature of the product manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$ using Lemma 1 proved in [82].

Property 24. *The sectional curvature of the manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$ is bounded between -4 and 0:*

$$-4\leqslant K\leqslant 0.\tag{4.104}$$

Proof. The property is true according to Lemma 1 since all the manifolds constituting the product manifold $\mathbb{H}_N^{++}\times\mathbb{SD}_N^{n-1}$ have a sectional curvature bounded between -4 and 0. □

We showed in Chapter 2 that complex stationary centered Gaussian autoregressive time series can be represented in the space $\mathbb{R}_+^*\times\mathcal{D}^{n-1}$ for one-dimensional time series and in the space $\mathbb{H}_N^{++}\times\mathbb{SD}_n^{n-1}$ for multidimensional time series. In Chapter 3, nous we endowed these spaces with a product Riemannian metric. In Chapter 4, we have given mathematical tools to perform operations in the Riemannian product manifolds constructed in Chapter 3: we have given the explicit expression of the distance, of the scalar product, of the norm, of an isometry, of the Riemannian logarithm map, of the Riemannian exponential map and finally we gave a bounding of the sectional curvature. Thanks to these tools, we will be able to rewrite some classical Machine Learning algorithms so that they respect the Riemannian manifold structure of the data studied here. This work is the subject of Chapter 5 and of the Python package geomstats [52].

Chapter 5

Machine Learning on Riemannian manifolds

In this chapter, some classical machine learning algorithms are generalized to Riemannian manifolds. Indeed, most machine learning algorithms are designed to work on Euclidean data. We can then use all the operations defined on vector spaces: we can add and subtract two elements, we can multiply and divide an element by a scalar... The squared Euclidean distance between the points $A = (a_1, \ldots, a_n)$ and $B = (b_1, \ldots, b_n)$ is given by: $d^2(A, B) = \sum_{i=1}^{n} (b_i - a_i)^2$. The mean of the points $(x_1, \ldots, x_N)$ is defined by: $x = \frac{1}{N} \sum_{i=1}^{N} x_i$. All these operations are available in a Euclidean space but they are not defined in a Riemannian manifold in general. To classify data on Riemannian manifolds, we will therefore generalize machine learning algorithms designed for Euclidean data so that they can work on any Riemannian manifold. Machine learning algorithms operating in Riemannian manifolds use geometric tools such as the distance, the scalar product, the Riemannian logarithm map, the Riemannian exponential map... Some of the codes of the algorithms presented here are available on the public Python package about machine learning on Riemannian manifolds named geomstats [52].

We start by defining the notion of mean over a Riemannian manifold in Section 5.1. In Section 5.2, we will present an algorithm named Tangent PCA (TPCA) which generalizes the PCA to Riemannian manifolds. This algorithm will be used in the application chapters to visualize the data in two dimensions. In Section 5.3 we present supervised classification algorithms. In Section 5.4 we present unsupervised classification algorithms.

Contents

5.1 Riemannian means

We start by defining the notion of p-mean in Section 5.1.1. The p-mean is defined as the set of minimizers of a certain function. When $p = 2$, the p-mean is called the mean. This definition of the Riemannian mean generalizes the definition of Euclidean mean to Riemannian manifolds. We will study the notion of Riemannian mean in Section 5.1.2.

5.1.1 The Riemannian p-mean

The p-mean of a dataset $(x_i)_{i=1,\ldots,N}$ is defined as the set of minimizers of the function:

$$H_p : x \mapsto \frac{1}{N} \sum_{i=1}^{N} d(x, x_i)^p. \tag{5.1}$$

This function is defined for the more general case of probability distributions on the manifold M in [6].

For $p \in]0,1[$, there are several local minimizers of the function g in general. A clustering algorithm named the p-mean-shift algorithm based an the non-uniqueness of the p-mean for $p < 1$ was presented in [20].

A *Hadamard* manifold is a Riemannian manifold that is complete, simply connected and has everywhere non-positive sectional curvature. For $p \in [1, +\infty[$, the function g has a unique global minimizer e_p in Hadamard manifolds [82], [6]. The existence and uniqueness of the Riemannian p-mean has also been studied by Xavier Pennec in [58]. The expression of the p-mean does not have an explicit formula in general.

In the following, we assume that the manifold studied is a Hadamard manifold. When $p \geqslant 1$, the p-mean can be approximated by performing a gradient descent on the function H_p given in Equation (5.1).

When $p > 1$, the gradient of the function H_p is given by the function G_p defined as follows:

$$G_p(x) = -\frac{p}{N} \sum_{i=1}^{N} d^{p-2}(x, x_i) \log_x(x_i), \quad \forall p > 1. \tag{5.2}$$

Note that the computation of the gradient G_p of the fonction H_p involves the Riemannian logarithm map.

When $p = 1$, the gradient of the function $H_1(x) = \frac{1}{N} \sum_{i=1}^{N} d(x, x_i)$ is not defined at points $(x_1, \ldots, x_N)$.

We then define the function:

$$G_1(x) = -\sum_{1 \leqslant i \leqslant N | x_i \neq x} \frac{\log_x(x_i)}{d(x, x_i)}. \tag{5.3}$$

The vector $G_1(x)$ is a subgradient of H_1 at x [6].

Algorithm 6 The p-mean approximation by gradient descent

Input: $x_1, \ldots, x_N \in M$
$\mu_0 = x_1$
$\mu_1 = \exp_{\mu_0}(-t_1 G_p(\mu_0))$
$k = 1$
while $k \leqslant K_{max}$ and $d(\mu_k, \mu_{k-1}) \geqslant \epsilon$ **do**
  $\mu_{k+1} = \exp_{\mu_k}(-t_{k+1} G_p(\mu_k))$
  $k = k + 1$
end while
return The coefficient μ_n obtained during the last iteration.

A gradient descent used to approximate the p-mean is described in Algorithm 6. We denote t_k the step size used in the gradient descent in Algorithm 6. Under certain conditions, it is proved in [6] that the agorithm 6 converges to the p-mean e_p when $\lim_{k \to \infty} t_k = 0$ and that $\sum_{k=1}^{\infty} t_k = \infty$.

Algorithm 7 The stochastic p-mean approximation

Input: $x_1, \ldots, x_N \in M$
$\mu_0 = x_1$
Randomly choose uniformly $i \in [\![2, N]\!]$
$\mu_1 = \exp_{\mu_0}\left(t_1\, p\, d(\mu_0, x_i)^{p-2}\, \log_{\mu_0}(x_i)\right)$
for $k = 1$ to n **do**
  Randomly choose uniformly $i \in [\![1, N]\!]$
  $\mu_{k+1} = \exp_{\mu_k}\left(t_{k+1}\, p\, d(\mu_k, x_i)^{p-2} \log_{\mu_k}(x_i)\right)$
end for
return The coefficient μ_n obtained during the last iteration.

A stochastic method to approximate the p-mean is presented in Algorithm 7. We note again t_k the size of the steps used in this algorithm. When $p \in [1, 2[$, the value of the product $d(\mu_k, x_i)^{p-2} \log_{\mu_k}(x_i)$ is not defined when $\mu_k = x_i$, we consider this product to be equal to zero by convention. Under certain conditions, it is proved in [6] that the agorithm 7 converges to the p-mean e_p when $\sum_{k=1}^{\infty} t_k = \infty$ and that $\sum_{k=1}^{\infty} t_k^2 < \infty$. We frequently choose $t_k = \frac{1}{k+1}$ for all $k \geqslant 1$.

The p-mean has specific names for $p = 1, 2$: e_1 is called the median and e_2 is called the mean. We will detail the special case of the mean in next section.

5.1.2 The Riemannian mean

We define the Riemannian mean as the p-mean defined in the previous section with $p = 2$. The mean of a dataset $(x_i)_{i=1,\dots,N}$ is therefore the set of minimizers of the function:

$$H_2(x) = \sum_{i=1}^{N} d^2(x, x_i).$$

The gradient of this function has for expression:

$$G_2(x) = -2 \sum_{i=1}^{N} \log_x(x_i). \tag{5.4}$$

In Algorithm 6, the p-mean is approximated using a gradient descent. This algorithm therefore becomes Algorithm 8 when $p = 2$.

Algorithm 8 The mean approximation by gradient descent

Input: $x_1, \dots, x_N \in M$
$\mu_0 = x_1$
$\mu_1 = \exp_{\mu_0}\left(2t_1 \sum_{i=1}^{N} \log_{\mu_0}(x_i)\right)$
$k = 1$
while $k \leqslant K_{max}$ and $d(\mu_k, \mu_{k-1}) \geqslant \epsilon$ **do**
  $\mu_{k+1} = \exp_{\mu_k}\left(2t_{k+1} \sum_{i=1}^{N} \log_{\mu_k}(x_i)\right)$
  $k = k + 1$
end while
return The coefficient μ_n obtained during the last iteration.

Algorithm 7 which estimates the p-mean using a stochastic method becomes Algorithm 9 when $p = 2$.

Algorithm 9 The stochastic mean approximation

Input: $x_1, \dots, x_N \in M$
$\mu_0 = x_1$
Randomly choose uniformly $i \in [\![2, N]\!]$
$\mu_1 = \exp_{\mu_0}\left(t_1 \log_{\mu_0}(x_i)\right)$
for $k = 1$ to n **do**
  Randomly choose uniformly $i \in [\![1, N]\!]$
  $\mu_{k+1} = \exp_{\mu_k}\left(2t_{k+1} \log_{\mu_k}(x_i)\right)$
end for
return The coefficient μ_n obtained during the last iteration.

We are now interested in the particular case of product manifolds endowed with a product metric, such as the manifolds studied in Sections 4.1.3 and 4.2.3. For product manifolds with a product metric, the expression of the squared distance between two points $A = (a_1, \dots, a_n)$ and $B = (b_1, \dots, b_n)$ on the product manifold $M = M_1 \times \dots \times M_n$ is the sum of the squared distances on the manifolds $M_1, \dots, M_n$:

$$d_M^2(A, B) = \sum_{i=1}^{n} d_{M_i}^2(a_i, b_i). \tag{5.5}$$

We denote by x the point of coordinates $(x_1, \dots, x_n)$ and $a_1, \dots, a_N$ the points of the dataset. We denote $(a_{i,1}, \dots, a_{i,n})$ the coordinates of the point a_i. Then, we have:

$$H_{2,M}(x) = \sum_{i=1}^{N} d_M^2(x, a_i) \tag{5.6}$$

$$= \sum_{i=1}^{N} \sum_{j=1}^{n} d_{M_j}^2(x_j, a_{i,j}) \tag{5.7}$$

$$= \sum_{j=1}^{n} \sum_{i=1}^{N} d_{M_j}^2(x_j, a_{i,j}) \tag{5.8}$$

$$= \sum_{j=1}^{n} H_{2,M_j}(x_j). \tag{5.9}$$

For the product manifolds endowed with a product metric, it is therefore possible to estimate independently the mean in each of the spaces M_i to estimate the mean $M = M_1 \times \ldots \times M_n$.

For example we defined in Equation (4.30) the squared distance between two Toeplitz matrices T_1 and T_2 as a linear combination of squared distances between the coordinates $(p_{0,1}, \mu_{1,1}, ..., \mu_{n-1,1})$ and $(p_{0,2}, \mu_{1,2}, ..., \mu_{n-1,2})$ obtained using the Levinson algorithm 13. Hence the coordinates can be averaged independently:

$$\begin{array}{ccccccccc} T_0 & \mapsto & (& \boxed{p_{0,0},} & & \boxed{\mu_{1,0},} & \cdots, & \boxed{\mu_{n-1,0}} &) \\ \vdots & & & \boxed{\vdots} & & \boxed{\vdots} & & \boxed{\vdots} & \\ T_{m-1} & \mapsto & (& \boxed{p_{0,m-1},} & & \boxed{\mu_{1,m-1},} & \cdots, & \boxed{\mu_{n-1,m-1}} &) \\ & & & \downarrow & & \downarrow & & \downarrow & \\ T & \leftarrow & (& p_0, & & \mu_1, & \cdots, & \mu_{n-1} &) \end{array}. \tag{5.10}$$

We now present an algorithm which generalizes the classical PCA to Riemannian manifolds. This algorithm will use the Riemannian mean defined in Section 5.1.2.

5.2 A visualization tool: the TPCA

In this section the classical Euclidean Principal Componant Analysis (PCA) is generalized to Riemannian manifolds: this algorithm is called the Tangent Principal Componant Analysis (TPCA). This method will be used in the application chapters to visualize the data in a two-dimensional space.

The first step of the TPCA is to approximate the mean x of the dataset using Algorithm 8 or Algorithm 9 described in Section 5.1.2.

Then we represent each point x_i of the dataset by the tangent vector at the mean $\overrightarrow{x\,x_i} := \log_x(x_i)$ which we renormalize according to the scalar product at the mean. Hence, we represent each point x_i of the Riemannian manifold by a vector in the tangent space at the mean which is a vector space and therefore suitable for Principal Components Analysis. Finally, we perform a classical PCA on the vectors $\overrightarrow{x\,x_i}$ for $i \in [\![1, N]\!]$ renormalized by the scalar product at the mean. Interested readers may refer to [30] and [59] for more details on the TPCA.

We detail the TPCA in Algorithm 10.

Algorithm 10 The Tangent Principal Component Analysis algorithm

Input: $x_1, \ldots, x_N \in M$

Compute de mean μ of the dataset $x_1, \ldots, x_N$.

Compute the Riemannian logarithm map $\overrightarrow{x\,x_i} := \log_x(x_i) \quad \forall i \in [\![1, N]\!]$

Define the renormalized vectors $\widetilde{\overrightarrow{x\,x_i}}$ renormalizing the vector $\overrightarrow{x\,x_i}$ using the scalar product matrix at the mean P_μ the following way : $\widetilde{\overrightarrow{x\,x_i}} = P_\mu^{1/2} \overrightarrow{x\,x_i} \quad \forall i \in [\![1, N]\!]$.

Define the empirical covariance matrix $\mathbf{S} = \frac{1}{N} \sum_{i=1}^{N} \widetilde{\overrightarrow{x\,x_i}}\, \widetilde{\overrightarrow{x\,x_i}}^H$.

Compute the eigenvalues $\lambda_1, \ldots, \lambda_n$ and the associated eigenvectors $v_1, \ldots, v_n$. The eigenvalues are sorted from the largest to the smallest. As the matrix $\mathbf{S}$ is an HPD matrix in general when the number of points N is largest than the dimension n, the eigenvalues are positive reals and the basis $(v_1, \ldots, v_n)$ is orthonormal.

return Project the vectors $\widetilde{\overrightarrow{x\,x_i}}$ on the basis vectors $(v_1, \ldots, v_k)$ using the standard scalar product, where k denotes the number of principal components desired.

5.3 Supervised algorithms

In this section, we present supervised classification algorithms that work in Riemannian manifolds. For each algorithm, we will specify the necessary geometric tools of the manifold (distance, scalar product, Riemannian logarithm map, Riemannian exponential map...).

5.3.1 The K-Nearest Neighbors

The k-Nearest Neighbors (KNN) classifier principle is quite simple: each point of the target dataset is labeled according to the k nearest points of the training set, i.e. a majority vote is performed on their labels. The number k of neighbors to be taken into account is a parameter of the algorithm chosen by the user. The output of this classification algorithm can also be a probabilistic result: for each point of the testing dataset, we give a vector whose coefficients indicate the probability of each label. The probability of each label will be equal to the number of points in the training dataset of the corresponding label among the k nearest neighbors of the point to be labeled. We just need to have an explicit expression of the distance between two points to be able to use this algorithm on a manifold. For each point to label, we have to evaluate n distances where n represents the number of points in the training dataset, then we have to sort these distances in ascending order which is done in $O(n\log(n))$ operations. The complexity of this algorithm is therefore in $O(n\log(n))$ with respect to the number n of points of the training dataset.

5.3.2 The Kernel Density Estimation Classifier

To label a point x using the Kernel Density Estimation Classifier, we estimate the density distribution of each label of the training dataset at the point x. The point x is then labeled according to the label having the greatest density at the point x. This algorithm is studied in [48].

To estimate the density distribution of each label, we use a kernel function K. There are many possible kernel functions K_s. The parameter s is a scale parameter affecting the size of the kernel. We now give some examples of kernel functions. The following function is named the Gaussian radial kernel:

$$G_s(x) = \exp\left(\frac{-x^2}{2s^2}\right) \tag{5.11}$$

We define the uniform radial kernel function by:

$$U_s(x) = \mathbb{1}_{x<s} \tag{5.12}$$

We define the triangular radial kernel function by:

$$T_s(x) = \left(1-\frac{x}{s}\right)\mathbb{1}_{\frac{x}{s}<1} \tag{5.13}$$

We define the parabolic radial kernel function by:

$$P_s(x) = \left(1-\left(\frac{x}{s}\right)^2\right)\mathbb{1}_{\frac{x}{s}<1} \tag{5.14}$$

We define the cosine radial kernel function by:

$$C_s(x) = \cos\left(\frac{\pi}{2}\left(\frac{x}{s}\right)\right)\mathbb{1}_{\frac{x}{s}<1} \tag{5.15}$$

We define the sigmoid radial kernel function by:

$$S_s(x) = \frac{1}{\exp\left(\frac{x}{s}\right)+\exp\left(-\frac{x}{s}\right)}\mathbb{1}_{\frac{x}{s}<1} \tag{5.16}$$

We define the bump radial kernel function by:

$$B_s(x) = \exp\left(\frac{-1}{1-\left(\frac{x}{s}\right)^2}\right)\mathbb{1}_{\frac{x}{s}<1} \tag{5.17}$$

A few other kernel functions are described on the Python package geomstats [52].

Let x be a point to label. We denote by L_i the label i and $x_{i,1},\dots,x_{i,n_i}$ the n_i points of the training dataset associated with the label i. We start by choosing a kernel function K and a kernel size s. We then estimate the density D_i of each label L_i at the point x:

$$D_i(x) = \sum_{j=1}^{n_i} K_s\left(d\left(x,x_{i,j}\right)\right). \tag{5.18}$$

The point x is then labeled according to the label L_i having the greatest density D_i at the point x. The output of this algorithm can also be a vector whose coefficient c_i represents the probability that the point x is associated with the label L_i. We then have: $c_i = D_i(x) / \sum_j D_j(x)$.

To use this algorithm, it suffices to have an explicit expression of the distance on the manifold M considered.

To label the point x, this algorithm computes the distance between the point x and the n points of the training dataset. The complexity of this algorithm is therefore $O(n)$, where n is the number of points of the training dataset.

5.3.3 The Nearest Centroid Classifier

To use the Nearest Centroid Classifier, we first approximate the mean μ_i of each label L_i of the training dataset. The target point x is then labelled according to the label L_i of the closest mean μ_i. This algorithm can also be called the Minimum Distance to Mean. It is a supervised version of the clustering algorithm named k-means.

To approximate the mean μ_i of each label L_i of the training dataset, we can use algorithms 8 and 9 presented in Section 5.1.2. We recall that these mean approximation algorithms use the Riemannian logarithm map and the Riemannian exponential map. They have a linear complexity with respect to the number of points to be averaged. If we denote by n_i the number of points in the training dataset associated with the label L_i and if we denote by n the total number of points of the training dataset, we have $n = \sum_i n_i$. The estimation of the means μ_i therefore has a linear complexity, i.e. in $O(n)$ with respect to the number n of points of the training dataset.

Once the averaging step has been completed, each point x is classified according to the label corresponding to the nearest mean μ_i. When the number l of different labels is small, this prediction is then very fast: it suffices to evaluate the distances $d(x, \mu_i)$ for $i \in [\![1, l]\!]$. We label the point x according to the label L_i corresponding to the minimum distance $d(x, \mu_i)$ obtained. This step is done in $O(l)$ operations, where l is the number of distinct labels.

To use the Nearest Centroid Classifier algorithm, it is therefore sufficient to have an explicit expression of the distance, the Riemannian logarithm map and the Riemannian exponential map of the studied manifold M. The averaging step is done in $O(n)$ operations, where n is the number of points in the training dataset. Then the labeling of each new data point x is done in $O(l)$ operations, where l is the number of distinct labels. This algorithm is therefore extremely efficient when we have a large number of points to label and when each label in the training dataset is well represented by its mean.

We will now present unsupervised classification algorithms. These algorithms are used to group spatially close points into clusters, they can be used to structure unlabeled data.

5.4 Unsupervised algorithms

In this section, we will present unsupervised classification algorithms operating on Riemannian manifolds. For each algorithm, we will specify the mathematical tools on which it is based. We will also give the complexity of each algorithm with respect to the number of points of the dataset.

We will present the HAC algorithm in Section 5.4.1, we will present the mean-shift algorithm in Section 5.4.2 and the k-means algorithm will be presented in Section 5.4.3. Finally we present in Section 5.4.4 a method called silhouette which allows to measure and visualize the degree of separation between the clusters given by a clustering algorithm. This tool can help to choose the number of clusters of an unsupervised algorithm.

5.4.1 The Hierarchical Agglomerative Clustering

The Hierarchical Agglomerative Clustering (HAC) works as follows: at the initial state, each point corresponds to a cluster. Then, until the desired number of cluster is reached, merge the two closest clusters. To do this, we need to define a measure of dissimilarity between clusters, i.e. a measure of dissimilarity between two sets of data. For example, we can define the dissimilarity between two clusters $c_1 = (c_{1,1}, \dots, c_{1,n_1})$ and $c_2 = (c_{2,1}, \dots, c_{2,n_2})$ as the average distance between a point belonging to c_1 and a point belonging to c_2:

$$d_{average}(c_1, c_2) = \frac{1}{n_1 n_2} \sum_{\substack{1 \leqslant i \leqslant c_1 \\ 1 \leqslant j \leqslant c_2}} d(c_{1,i}, c_{2,j}). \tag{5.19}$$

We can also define the dissimilarity between two clusters as being the minimum distance between a point of the cluster c_1 and a point of the cluster c_2:

$$d_{single}(c_1, c_2) = \min_{\substack{1 \leqslant i \leqslant c_1 \\ 1 \leqslant j \leqslant c_2}} d(c_{1,i}, c_{2,j}). \tag{5.20}$$

Note that the dissimilarity d_{single} is not a mathematical distance on the set of clusters since it is not separated and does not verify the triangular inequality.

It is also possible to define the dissimilarity between two clusters as being the maximum distance between a point of the cluster c_1 and a point of the cluster c_2:

$$d_{complete}\left(c_1, c_2\right) = \max_{\substack{1 \leqslant i \leqslant c_1 \\ 1 \leqslant j \leqslant c_2}} d\left(c_{1,i}, c_{2,j}\right). \tag{5.21}$$

Algorithm 11 The Hierarchical Agglomerative Clustering

Initialization:
Each point represents a cluster.
Choose a distance between clusters, for example the distance $d_{average}$ presented in Equation (5.19), the distance d_{single} presented in Equation (5.20) or the distance $d_{complete}$ presented in Equation (5.21).
while The number of clusters is greater than the desired number of clusters **do**
Merge the two closest clusters according to the distance chosen previously. They now form a new cluster. The number of cluster decrease by one at each loop.
end while
return Each point is labeled according to its final cluster.

The HAC algorithm is described in Algorithm 11. There exist a graphic named dendrogram which can be used to visualize the clusters being merged. A dendrogram looks like a reversed tree which leafs represent the data to cluster. The vertical axis represents the distance at which two clusters are from each other when they are merged. The leaves of the tree represents the initial data, are therefore represented at zero height. Then we see the clusters merging with each other as we rise along the y-axis until we reach the root at the very top: the height of the root represents the distance between the last two clusters to be merged.

Once an clustering is obtained using the HAC algorithm, we can assign a new data point to an already existing cluster: the points will be assigned to the closest cluster with respect to the *average* dissimilarity measure between clusters (5.19), the *single* dissimilarity measure (5.20) or the *complete* dissimilarity measure (5.21), the new data point being considered as a cluster with one element. We usually use the same cluster dissimilarity measure to assign a new data point to an already existing cluster than the dissimilarity measure initialy used to create the clusters in the HAC algorithm.

Note that it suffices to have the explicit expression of the distance between two points of the manifold M to be able to use the HAC algorithm. When we merge the first two clusters in the first iteration, we need to compute $n\left(n-1\right)/2$ distances, where n is the number of points in the dataset. At each iteration of the algorithm, we recompute the distances involving the clusters that have been modified in the previous iteration and we keep in memory the distances that have not been modified. Whether we use the *average*, the *single* or the *complete* dissimilarity measure, we can use the dissimilarities computed previously for the clusters c_i and c_j to compute easily the dissimilarities from the new cluster $c = c_i \cup c_j$ to the other clusters. Finally, the complexity of the HAC algorithm when one of these three dissimilarity measures (average, single or complete) is used is in $O\left(n^2\right)$, where n denotes the number of points in the dataset.

5.4.2 The mean-shift

The mean-shift algorithm works as follows: each point x_i of the dataset $x_1, \ldots, x_n$ is associated with a point p_i initially placed in x_i which moves along a gradient ascent of the estimated density function of the dataset $x_1, \ldots, x_n$.

In the mean-shift algorithm, we use a kernel function K (of size s) to define a function D estimating the density of the dataset $(x_i)_{i=,\ldots,N}$:

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} K_s\left(d\left(x, x_i\right)\right) \tag{5.22}$$

where $d(x, x_i)$ represents the distance between x and x_i. Several examples of kernel functions have been presented in Section 5.3.2. A kernel function taking into account the curvature of the manifold is described in [48].

The gradient of the function D is givent by:

$$\operatorname{grad}_x D\left(\cdot\right) = -\frac{1}{N} \sum_{i=1}^{N} K_s'\left(d\left(x, x_i\right)\right) \frac{\log_x\left(x_i\right)}{d\left(x, x_i\right)} \tag{5.23}$$

as the gradient of the function $d\left(x, y\right)$ is:

$$\operatorname{grad}_x d\left(\cdot, y\right) = -\frac{\log_x\left(y\right)}{d\left(x, y\right)}. \tag{5.24}$$

To move along the gradient defined in Equation (5.23), we set $p_{i,0} = x_i$ and we build by reccurence:

$$p_{i,j+1} = \exp\left(t_{j+1} \operatorname{grad}_{p_{i,j}} D\left(\cdot\right)\right), \tag{5.25}$$

where t_{j+1} represents the step size used in iteration $j+1$.

Each sequence of points $p_{i,j}$ will tend towards a local maximum of the density function estimated by the function D when $j \to \infty$. The points x_i are then grouped according to the local maximum of the density where the points p_i have arrived: all the points x_i whose points p_i have reached the same maximum are grouped together in the same cluster. The larger the scale parameter s of the kernel function K, the more regular the estimated density function will be. The estimated density function will therefore have fewer local maxima, so the mean-shift algorithm will give fewer clusters. The mean-shift algorithm has been adapted to the Siegel metric described in Section 4.2.2.2 in [22], [12].

According to Equations (5.23) and (5.25), it suffice to have an explicit expression of the distance function, of the Riemannian logarithm map and of the Riemannian exponential map of the manifold M to use the mean-shift algorithm.

We now study the complexity of the mean-shift algorithm. For each point x_i of the n points of the dataset, we associate a point p_i which moves along the gradient defined in Equation (5.23). To compute this gradient in iteration j, we compute the distance between the point $p_{i,j}$ and each of the n points of the dataset. The complexity of the mean-shift algorithm is therefore quadratic with respect to the number of points of the dataset.

5.4.3 The k-means

The k-means algorithm is an algorithm based on the mean. We describe the k-means algorithm in Algorithm 12.

Algorithm 12 k-means algorithm for k clusters

Initialization:
Pick randomly k points in the dataset. They now represent the barycenters of each cluster.
for i = 1 to loop number **do**
  Assign each point of the dataset to the closest barycenter.
  Compute the new barycenter of each cluster.
end for
return Each point is labeled according to the closest barycenter.

We denote by k the number of clusters of the k-means algorithm and we denote by $(x_{i,1}, \ldots, x_{i,n_i})$ the n_i points of cluster i, for all $i \in [\![1, k]\!]$. We denote by μ_i the mean of the cluster i. The proof of the convergence of the algorithm is done by noting that the sum:

$$S = \sum_{i=1}^{k} \sum_{j=1}^{n_i} d^2 \left(\mu_i, x_{i,j}\right) \tag{5.26}$$

strictly decreases at each iteration of the algorithm. Depending on the initial values, the k-means algorithm can converge toward several different final states. It is therefore usual to restart the k-means algorithm several times with different initial cluster centers, then to compare the final states obtained. When different realizations of the k-means algorithm give several different final results, the value of the sum S obtained during the last iteration can be a criterion for choosing among the different final states: the smaller the value of the sum S is, the more interesting is the clustering obtained.

Once the k-means algorithm has been performed on a dataset, we can assign a new data point to an already existing cluster for a low computational cost: the new data point will be assigned to the cluster having the closest barycenter. An other version of the k-means called oneline k-means can be used when we expect the data points to be received one by one. The oneline k-means algorithm is available on the Python package geomstats [52].

Let us study the mathematical tools useful to use the k-means algorithm. At each iteration of the k-means algorithm, we use the definition of the distance on the manifold M to assign each point to the nearest barycenter. Then, we approximate the mean of the new barycenter. The algorithms 8 and 9 described in Section 5.1.2 approximate the position of the mean of a dataset using the Riemannian logarithm map and the Riemannian exponential map of the manifold M. To use the k-means algorithm on a Riemannian manifold M, it is therefore sufficient to have the explicit expression of the distance, of the Riemannian logarithm map and of the Riemannian exponential map on the manifold M.

Finally, we study the k-means algorithm complexity. At each iteration of the k-means algorithm, we compute the distance between each point and each barycenter in order to determine if a point should stay in its current cluster or be assigned to another cluster. This is done in $O(kn)$ operations, where k is the number of clusters and n is the number of points of the dataset. Then, we approximate the mean of each of the newly formed clusters. This step is done in $O(n)$ operations. For a fixed number k of clusters and a fixed number of iterations, the complexity of the k-means algorithm is therefore linear with respect to the number of points of the dataset.

5.4.4 Silhouette

One of the challenges of clustering is to choose a meaningful number of clusters k. We now introduce a tool to guide us in the choice of the number of clusters k.

We now define the silhouette $s(x)$ of a point x. Take any point x in the data set, and denote by A the cluster to which it has been assigned. If A contains only one point x, we set $s(x) = 0$ as a neutral choice. When cluster A contains other points apart from x, we define:

$$a(x) = \min_{y \in A \setminus \{x\}} d(x, y) \tag{5.27}$$

Let us now consider a cluster C different from A, and define:

$$c(x) = \min_{y \in C} d(x, y). \tag{5.28}$$

After computing $c(x)$ for all clusters C different from A, we select the smallest of those numbers, and denote it by:

$$b(x) = min_{C \neq A} c(x). \tag{5.29}$$

The value $b(x)$ represents how far x is from the cluster that would have been the second best choice for x.
Hence we have:

$$b(x) = \min_{y \notin A} d(x, y). \tag{5.30}$$

The silhouette number $s(x)$ is then computed as follows:

$$s(x) = \frac{b(x) - a(x)}{\max(a(x), b(x))}. \tag{5.31}$$

The silhouette measures how much x is closer from the cluster A than the cluster that would have been the second best choice for x.

We always have $-1 \leqslant s(x) \leqslant 1$.

The silhouette gives indications about the shape of the clusters given by the classification algorithm. Clusters with small radius far from the other clusters will give silhouette values close to 1.

However, the silhouette does not give any information about the performance of the clustering agorithm. Most of the time, if the clusters overlap, a perfect classification using the true labels (if they are known) will give a worst silhouette than the k-means clustering. Moreover, well separated clusters with non-spherical interlaced shapes can also have a low mean silhouette value.

When the clusters do not overlap, the average silhouette value can by used as a criterion to determine the number of clusters: we classify the data with different number of clusters and keep the classification giving the best mean silhouette value.

The silhouette is used in [17] on simulated radar clutter which was simulated using the model presented in Section 6.1. Interested readers can read [69] for a reference about the silhouette construction, and [22] for another radar application.

The algorithms presented in Chapter 5 will be applied in Chapters 6, 7, 8 and 9 to classify complex stationary centered Gaussian autoregressive time series. Among others, we will represent the unidimensional time series in the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ presented in Section 4.1.3 and the multidimensional time series in the Riemannian manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ presented in Section 4.2.3. To be able to use the classification algorithms presented in Chapter 5, we will therefore use mathematical tools presented in Chapter 4 such as the distance, the scalar product, the Riemannian logarithm map and the Riemannian exponential map of the manifolds $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ and $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$.

Chapter 6

Application to simulated unidimensional stationary centered Gaussian autoregressive time series classification

In this chapter, we will simulate complex one-dimensional stationary centered Gaussian autoregressive time series. The simulated time series therefore correspond to the modeling hypotheses of Section 2.1. We have seen that these time series can be represented by the power spectral density function S, by the autocorrelation coefficients $(r_0, r_1, \ldots, r_{n-1})$ associated with the Toeplitz HPD autocorrelation matrix R, or by the coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ where the coefficient $p_0 = r_0$ is the mean quadratic power coefficient and the coefficients $(\mu_1, \ldots, \mu_{n-1})$ are called coefficients reflection. In the particular case of signals whose power spectral density S has a Gaussian shape of power P, mean m and variance σ^2, that is to say:

$$S_f(\xi) = P\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\xi-m)^2}{2\sigma^2}}, \tag{6.1}$$

the theoretical autocorrelation coefficients are:

$$r_k = R_f(k) = Pe^{i2\pi mk}e^{-2\pi^2\sigma^2k^2} \quad \forall k \geqslant 1 \tag{6.2}$$

and the theoretical reflection coefficients are:

$$\mu_k = (-1)^k e^{i2\pi mk}e^{-2\pi^2\sigma^2k} \quad \forall k \geqslant 1. \tag{6.3}$$

This property presented in Sections 1.4.1 and 2.1 is proven in Appendix B.

In Section 6.1, we present a simulation model of complex one-dimensional stationary centered Gaussian autoregressive time series based on the Toeplitz autocorrelation matrix R. In Section 6.2, we use Equation (6.2) and the simulation model presented in Section 6.1 to simulate two datasets. The first dataset will consist of simulated time series having a theoretical spectrum of Gaussian shape of power P, of mean m_1 and of variance σ^2, the second dataset will consist of simulated time series having a theoretical spectrum of Gaussian shape of power P, of mean $m_2 \neq m_1$ and of variance σ^2. For each simulated time series, we will then use the regularized Burg algorithm 17 detailed in Appendix C to estimate the theoretical coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ from the simulated time series.

Finally, we perform the supervised classification of the simulated time series in Section 6.3. Each of the two datasets is slited into a training dataset and a testing dataset. To perform the classification, we will use the nearest centroid classifier algorithm presented in Section 5.3.3 on different representation spaces.

Contents

6.1 Simulation model

We have seen in Section 2.1 that the complex one-dimensional stationary centered Gaussian autoregressive time series can be considered as realizations of multidimensional circularly-symmetric central complex normal distributions whose covariance matrix R is Toeplitz:

$$\mathbf{R} = \begin{bmatrix} r_0 & r_1^* & r_2^* & \dots & r_{n-1}^* \\ r_1 & r_0 & r_1^* & \dots & r_{n-2}^* \\ r_2 & r_1 & r_0 & \dots & r_{n-3}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & r_{n-3} & \dots & r_0 \end{bmatrix}. \tag{6.4}$$

A stationary time series with a correlation matrix R can therefore be simulated using the equation:

$$Z = R^{1/2}X \tag{6.5}$$

with:

R**:** the autocorrelation matrix (Toeplitz Hermitian Positive Definite).

X**:** a standard complex Gaussian random vector whose dimension is equal to the length of the time series.

We now use the simulation model presented in Equation (6.5) to simulate the data to be classified.

6.2 Simulated dataset

We simulate two datasets of 200 time series each. The first dataset consists of simulated time series having a theoretical spectrum of Gaussian shape of power $P_1 = 1$, of mean $m_1 = 0.1$ and of variance $\sigma_1^2 = 0.05$. The second dataset consists of simulated time series having a theoretical spectrum of Gaussian shape of power $P_2 = 1$, of mean $m_2 = -0.1 \neq m_1$ and of variance $\sigma_2^2 = 0.05$. The length of each simulated time series is 20.

We simulate the two datasets using the simulation model presented in Equation (6.5):

$$Z = R^{1/2}X. \tag{6.6}$$

To simulate the 200 vectors of the first dataset, we use the Toeplitz correlation matrix $\mathbf{R}_1$ whose coefficients are obtained using Equation (6.2):

$$r_k = R_f(k) = Pe^{i2\pi mk}e^{-2\pi^2\sigma^2k^2} \quad \forall k \geqslant 1$$

by noting $P = P_1 = 1$, $m = m_1 = 0.1$ and $\sigma^2 = \sigma_1^2 = 0.05$.

The coefficients of the Toeplitz correlation matrix $\mathbf{R}_2$ used to simulate the 200 vectors of the second dataset are also obtained from Equation (6.2) by noting $P = P_2 = 1 = P_1$, $m = m_2 = -0.1 = -m_1$ and $\sigma^2 = \sigma_2^2 = 0.05 = \sigma_1^2$.

Each of these two datasets is then splited into a training dataset consisting of 100 time series and a testing dataset consisting of 100 time series.

We now present tools to visualize and classify the time series simulated in this section.

6.3 Classification

In this section, we present several methods to visualize and classify the time series simulated in the previous section. Whatever method is used to represent the signals, they are classified using the nearest centroid classifier algorithm presented in Section 5.3.3. For each classification method presented, we start by visualizing the whole simulated dataset as well as the theoretical positions associated with the correlation matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ used to simulate the time series. These theoretical elements are only used to view the dataset, they are not used for classification. The nearest centroid classifier algorithm is then performed on the simulated dataset. We present the Tangent PCA (or the PCA) of the classification result on the testing dataset and we present the corresponding confusion matrix.

We start by presenting three Euclidean methods to represent the simulated time series based on their spectral estimation: we will use the squared modulus of the FFT, the periodogram and the Capon spectra. For each of these three spectral representation methods, we will study the spectra in standard scale and in logarithmic scale (dB).

For each spectral estimator (FFT square modulus, periodograms, Capon spectra) and for each scale (standard and logarithmic), we will present four figures. The first figure represents the estimated spectra for each simulated signal. We also represent the theoretical spectra computed from the coefficients used to simulate the time series. The second figure

represents the PCA on the estimated spectra. The nearest centroid classifier algorithm is then performed on the spectra. The third figure represents the PCA of the classification result obtained. Finally, the fourth figure represents the confusion matrix of the classification obtained.

We then present a classification method based on the estimation of the Toeplitz autocorrelation matrix R of each simulated time series. We compare the classification results of the obtained using the Euclidean metric on the estimated HPD autocorrelation matrices $R \in \mathcal{H}_n^+$ or the information geometry metric presented in Section 4.2.1.

Finally, we present a classification method based on the estimation of the coefficients $(p_0, \mu_1, \ldots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$. We compare the results of the classification obtained using the Euclidean metric on the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ or the metric inspired by information geometry presented in Section 4.1.3.

6.3.1 The FFT square modulus

In this section, we represent the time series simulated in Section 6.2 by the squared modulus of their FFTs. We first present the spectra obtained in standard scale then in logarithmic scale (dB).

We recall that for each time series $\mathbf{u}$, the FFT $\mathbf{f}$ is defined by the equation:

$$\mathbf{f} = \frac{1}{\sqrt{n}} W\mathbf{u}, \quad \text{with:} \tag{6.7}$$

$$\mathbf{f} := [f_0, \ldots, f_{n-1}]^T \tag{6.8}$$

$$\mathbf{u} = [u_0, \ldots, u_{n-1}]^T \tag{6.9}$$

$$W_{k,l} = w^{kl}, \text{where } w = e^{-\frac{i2\pi}{n}}. \tag{6.10}$$

Hence, the squared modulus of the FFT for the reduced frequency $f_k = k/N$ of a time series of length N has the expression:

$$\widehat{S}\left(\frac{k}{N}\right) = \frac{1}{N}\left|\sum_{n=0}^{N-1} u(n)e^{-i2\pi n\frac{k}{N}}\right|^2. \tag{6.11}$$

It is therefore an estimator for the frequencies $f_k = k/N$ of the power spectral density defined by:

$$S(f) = \sum_{m=-\infty}^{+\infty} r_m e^{-i2\pi mf} = \lim_{N\to+\infty} \mathbb{E}\left[\frac{1}{N}\left|\sum_{n=0}^{N-1} u(n)e^{-i2\pi nf}\right|^2\right] \tag{6.12}$$

with $r_m = \mathbb{E}[u(n+m)u(n)^*]$.

We also recall the definition of the periodogram:

$$\widehat{S}_{per}(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} u(n)e^{-i2\pi nf}\right|^2. \tag{6.13}$$

The periodogram therefore coincides with the squared modulus of the FFT for frequencies of the form $f_k = k/N$.

The periodogram is also defined by:

$$\widehat{S}_{per}(f) = \sum_{m=-(N-1)}^{N-1} \widehat{r}(m)e^{-i2\pi mf}. \tag{6.14}$$

with:

$$\widehat{r}(m) = \frac{1}{N}\sum_{k=0}^{N-m-1} u(k+m)u(k)^*. \tag{6.15}$$

We will use this second equation to compute the theoretical spectra associated with the autocorrelation matrices R_0 and R_1 by replacing the coefficients $\widehat{r}(m)$ by the theoretical coefficients of the matrices R_0 and R_1.

We start by presenting the squared modulus of FFTs in standard scale. In Figure 6.1, we plot the squared modulus of the estimated FFTs. The FFTs of the time series simulated with the Toeplitz autocorrelation matrix R_0 are plotted in red, the FFTs of the time series simulated with the autocorrelation matrix R_1 are plotted in blue. The theoretical spectrum associated with the matrix R_0 is plotted with a thick red line, the theoretical spectrum associated with the matrix R_1 s plotted with a thick blue line. Figure 6.2 represents the PCA of the squared modulus of the FFT. Figure 6.3 represents the PCA of the classification result using the nearest centroid classifier algorithm. Figure 6.4 represents the confusion matrix of the classification result using the nearest centroid classifier algorithm.

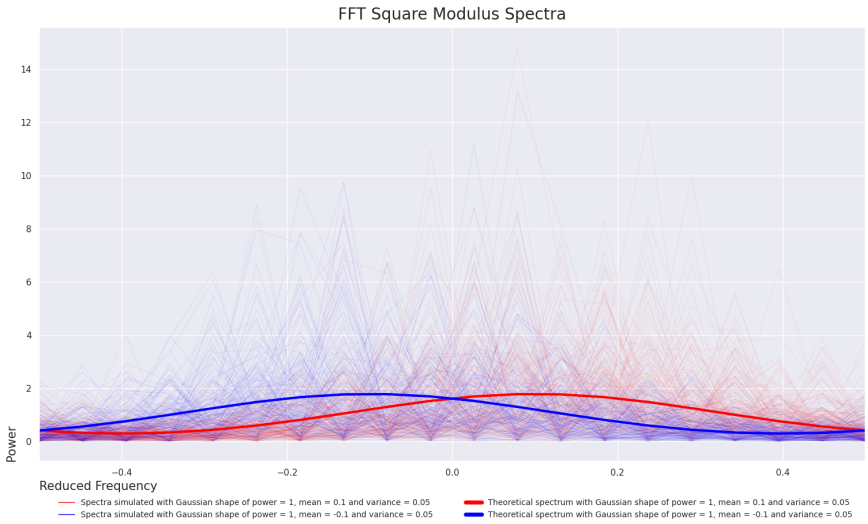

Figure 6.1: FFT square modulus spectra

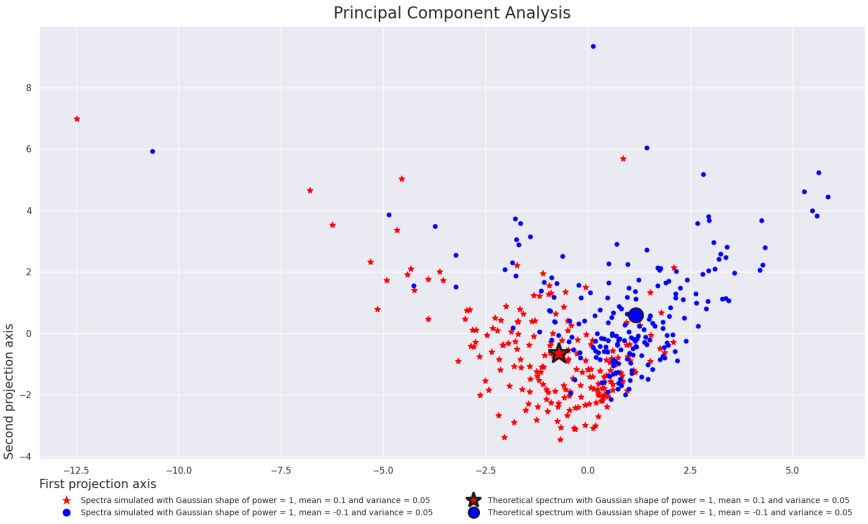

Figure 6.2: PCA on the FFT square modulus spectra

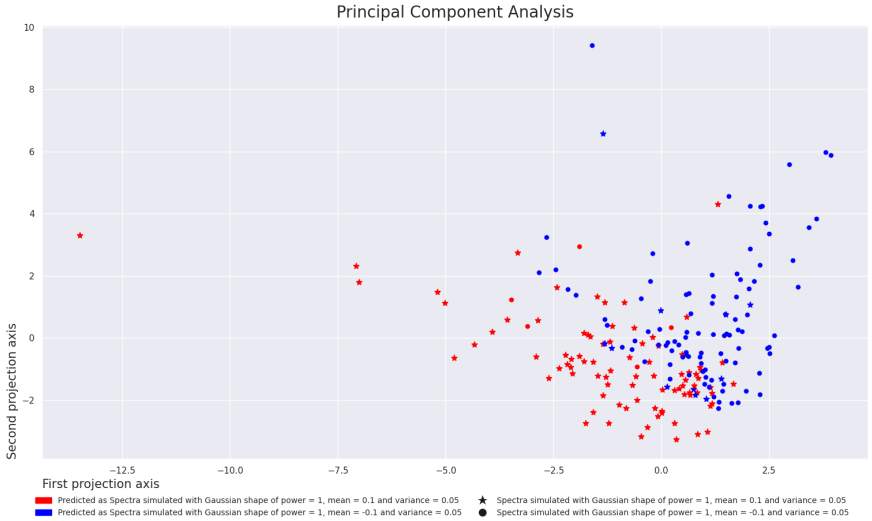

Figure 6.3: PCA of the classification result of the nearest centroid classifier on the FFT square modulus spectra

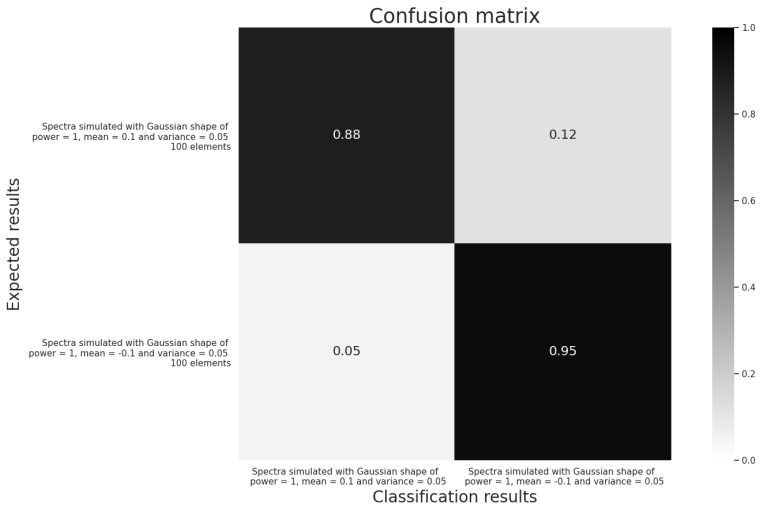

Figure 6.4: Confusion matrix of the classification result of the nearest centroid classifier on the FFT square modulus spectra

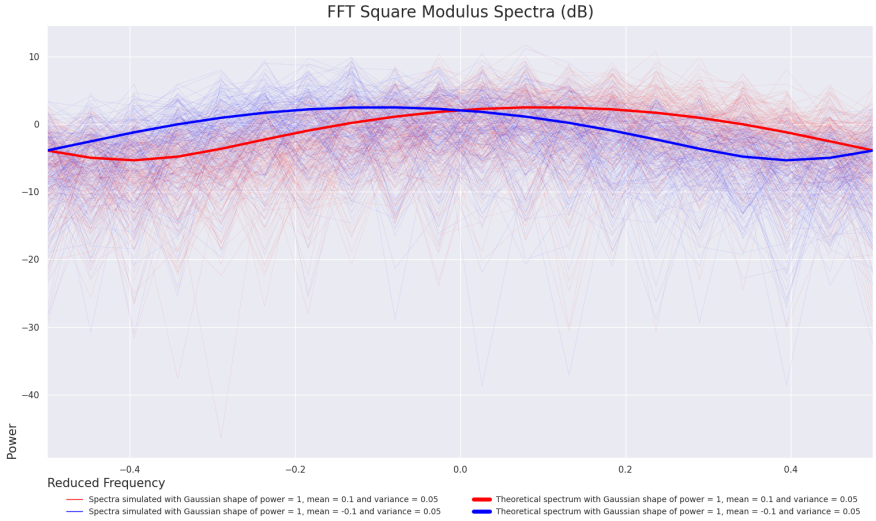

Figure 6.5: FFT square modulus spectra (dB)

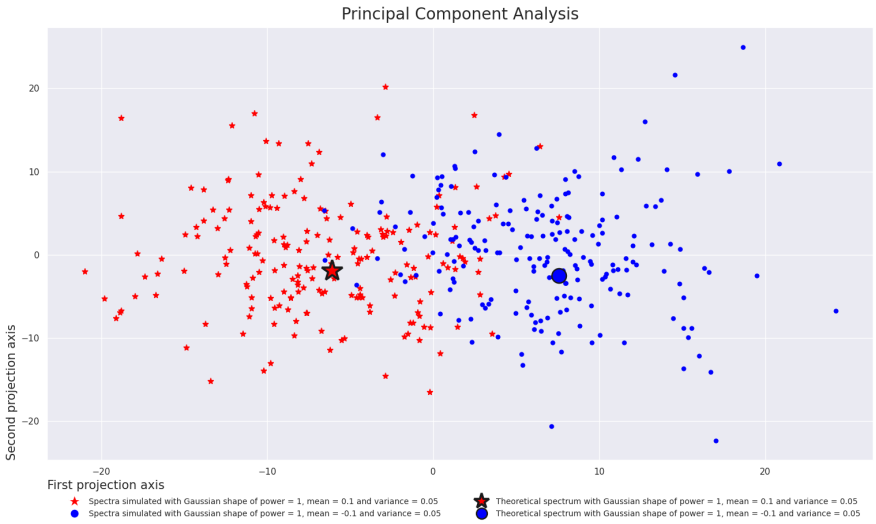

Figure 6.6: PCA on the FFT square modulus spectra (dB)

We now present the classification of the squared modulus of the FFT in logarithmic scale. More precisely, the FFTs are expressed in dB, i.e. the function $x \mapsto 10 \times \log_{10}(x)$ is applied to the squared modulus of the FFT. In Figure 6.5, we plot the squared modulus of the FFT in logarithmic scale. Figure 6.6 represents the PCA of the squared modulus of the FFT expressed in dB. Figure 6.7 represents the PCA of the classification result obtained using the nearest centroid classifier algorithm. Figure 6.8 represents the confusion matrix of the classification result obtained using the nearest centroid classifier algorithm.

6.3.2 Periodograms

In this section, we represent the time series simulated in Section 6.2 by their periodograms. We first present the spectra obtained in standard scale then in logarithmic scale (dB).

We recall the definition of the periodogram:

$$\widehat{S}_{per}(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} u(n)e^{-i2\pi nf}\right|^2. \tag{6.16}$$

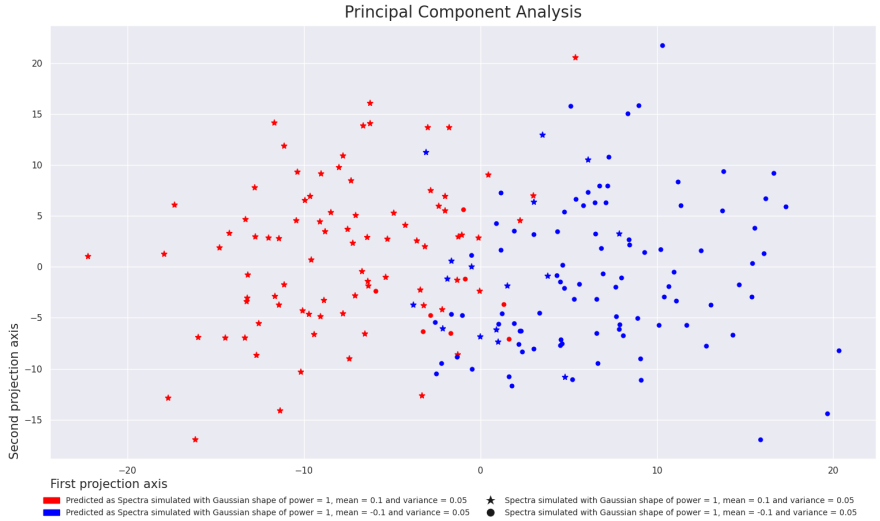

Figure 6.7: PCA of the classification result of the nearest centroid classifier on the FFT square modulus (dB) spectra

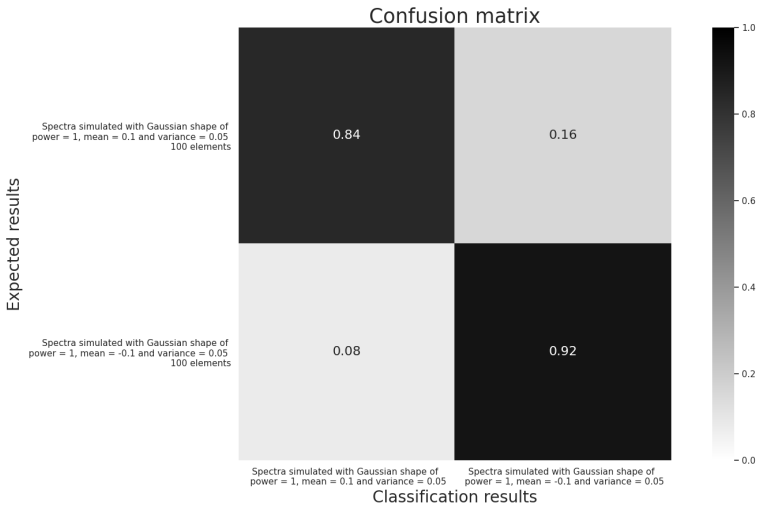

Figure 6.8: Confusion matrix of the classification result of the nearest centroid classifier on the FFT square modulus (dB) spectra

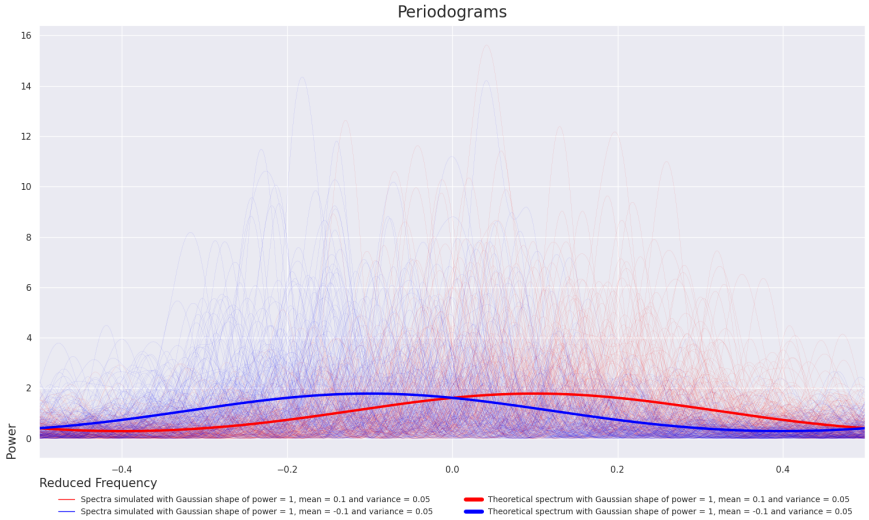

Figure 6.9: Periodograms

The periodogram is also defined by:

$$\widehat{S}_{per}(f) = \sum_{m=-(N-1)}^{N-1} \widehat{r}(m) e^{-i2\pi mf}. \tag{6.17}$$

with:

$$\widehat{r}(m) = \frac{1}{N} \sum_{k=0}^{N-m-1} u(k+m)u(k)^*, \tag{6.18}$$

We will use this second equation to compute the theoretical spectra associated with the autocorrelation matrices R_0 and R_1 by replacing the coefficients $\widehat{r}(m)$ by the theoretical coefficients of the matrices R_0 and R_1.

We first present the periodograms in standard scale. In Figure 6.9, we present the spectra plotted from the periodograms computed for each time series. We use 1001 points to plot the spectra: we compute the values of the function $\widehat{S}_{per}(f)$ for f ranging from -0.5 to 0.5 with a step size of 0.001. To plot the PCA and classify the periodograms, we use 1000 points: we eliminate one of the two extreme values ($f = -0.5$ or $f = 0.5$) since these two values always coincide as they correspond to the same frequency. Figure 6.10 represents the PCA of the periodograms. Figure 6.11 represents the PCA of the classification result obtained using the nearest centroid classifier algorithm. Figure 6.12 represents the confusion matrix of the classification result obtained using the nearest centroid classifier algorithm.

We now present the periodograms in logarithmic scale (in dB). In Figure 6.13, we plot the periodograms in dB. Figure 6.14 represents the PCA of the periodograms expressed in dB. Figure 6.15 represents the PCA of the classification result obtained using the nearest centroid classifier algorithm. Figure 6.16 represents the confusion matrix of the classification result obtained using the nearest centroid classifier algorithm.

6.3.3 Capon spectra

In this section, we represent the time series by their Capon spectra. We first present the spectra obtained in standard scale then in logarithmic scale (dB).

We begin by recalling the definition of Capon spectra.

We first define the autoregressive spectrum of the complex centered stationary Gaussian process $\mathbf{u}$ of autoregressive coefficients $a_1^{n-1}, \ldots, a_{n-1}^{n-1}$:

$$S_{AR}^{(n-1)}(f) = \frac{P_{n-1}}{\left| \sum_{k=0}^{(n-1)} a_k^{(n-1)} e^{-i2\pi kf} \right|^2} \tag{6.19}$$

where $P_{n-1} \in \mathbb{R}_+^*$ is the variance of the prediction error $\nu^{n-1}(k)$ of the linear autoregressive model presented in Equation (2.28).

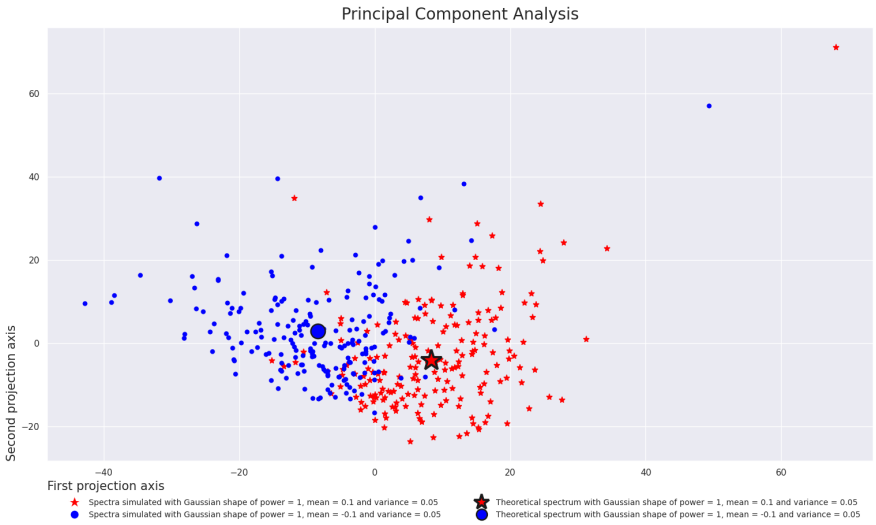

Figure 6.10: PCA on periodograms

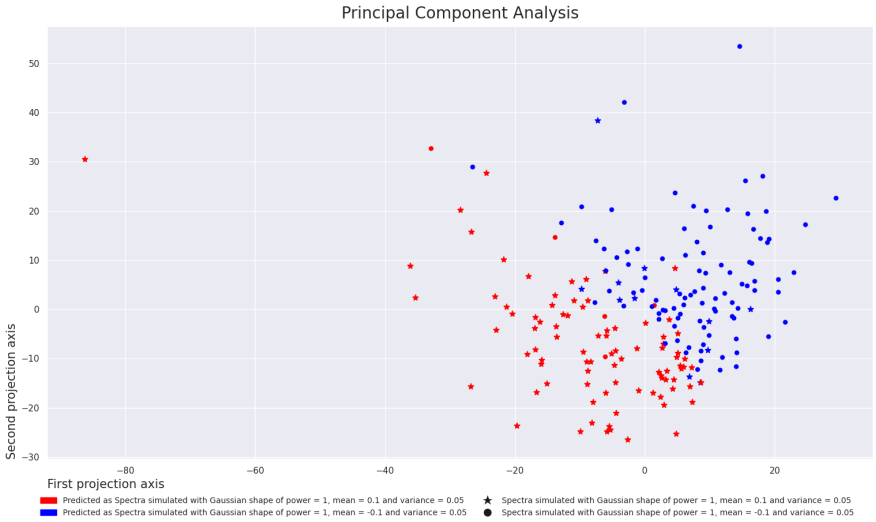

Figure 6.11: PCA of the classification result of the nearest centroid classifier on the periodograms

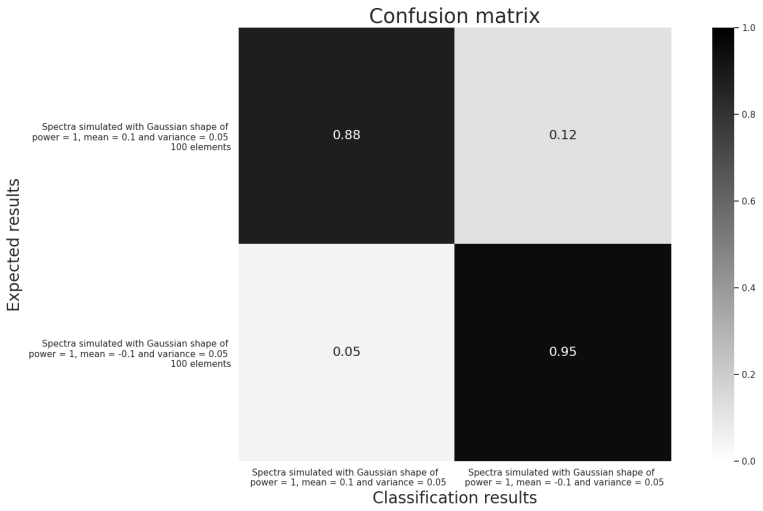

Figure 6.12: Confusion matrix of the classification result of the nearest centroid classifier on the periodograms

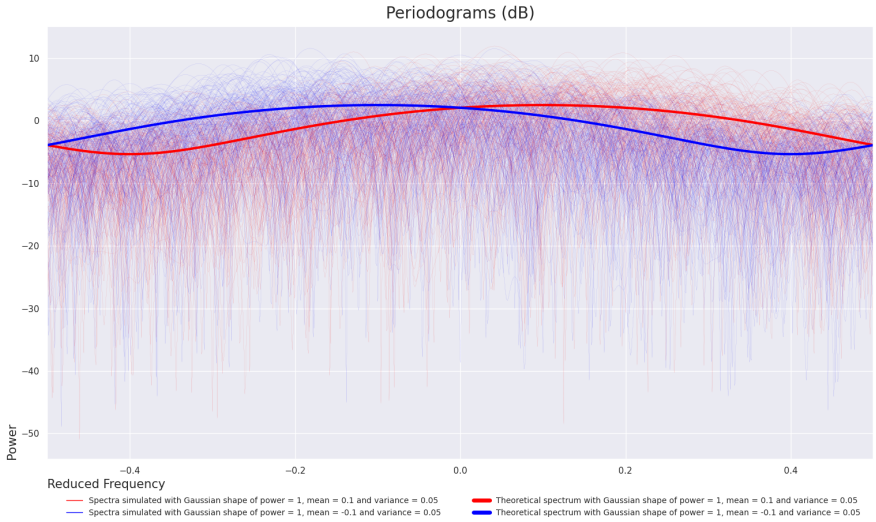

Figure 6.13: Periodograms (dB)

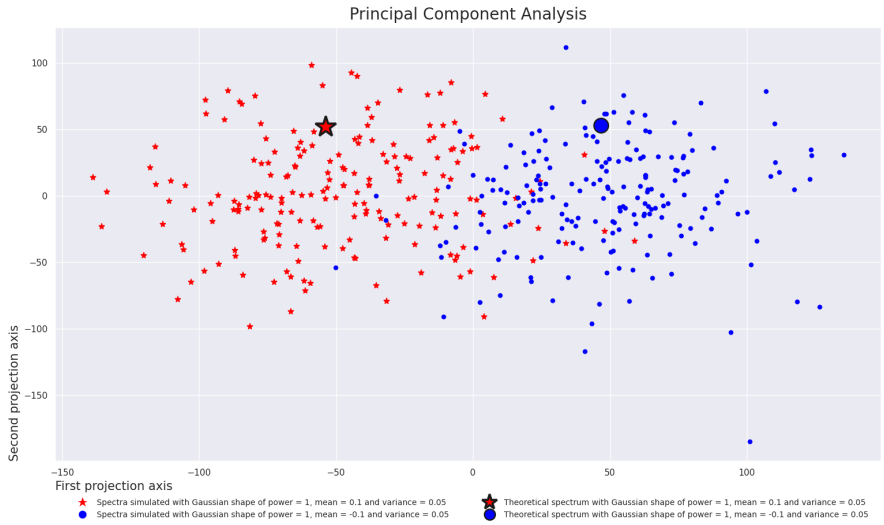

Figure 6.14: PCA on periodograms (dB)

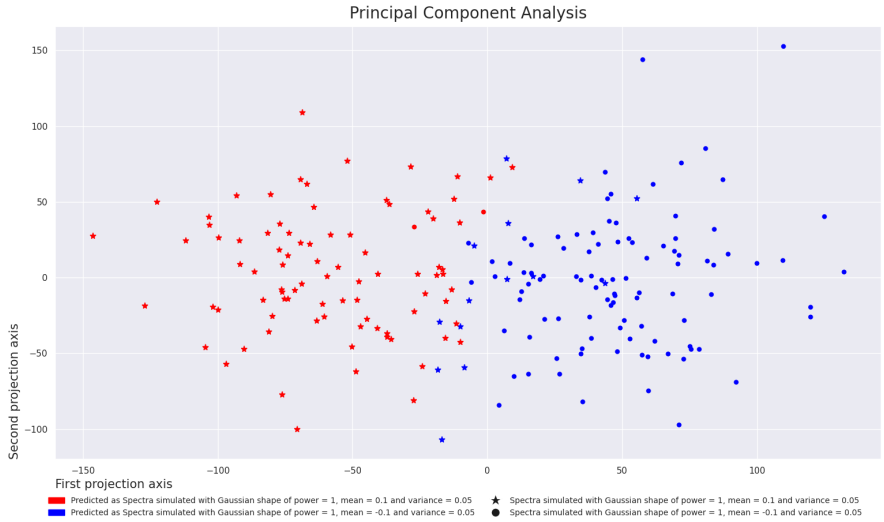

Figure 6.15: PCA of the classification result of the nearest centroid classifier on the periodograms (dB)

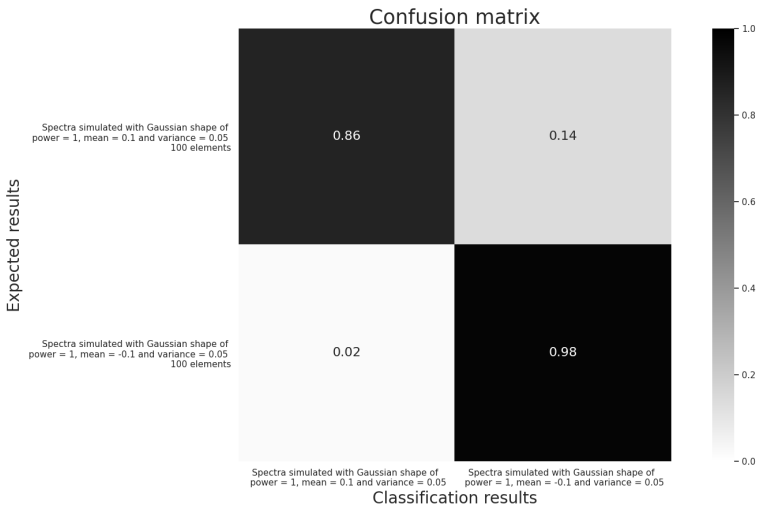

Figure 6.16: Confusion matrix of the classification result of the nearest centroid classifier on the periodograms (dB)

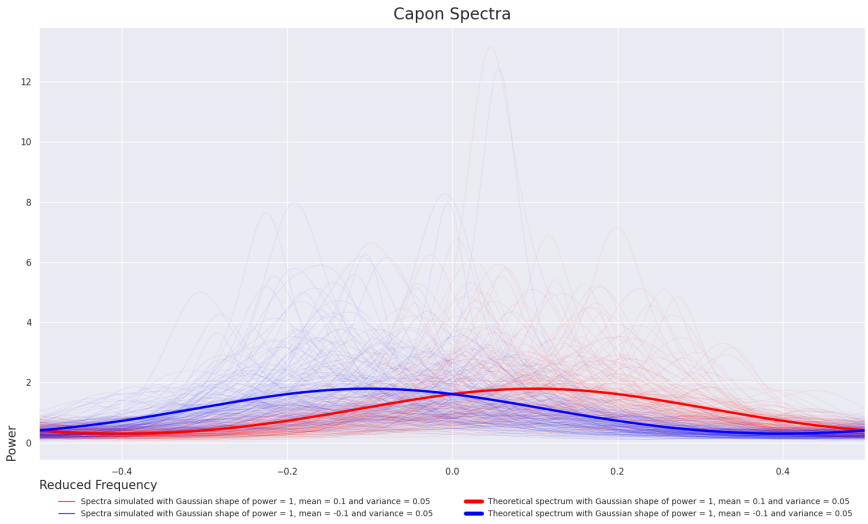

Figure 6.17: Capon spectra

The Capon spectrum is then defined as the harmonic mean of the autoregressive spectra of orders $k = 1, \ldots, n-1$:

$$S_{Capon}^{(n-1)}(f)^{-1} = \frac{1}{n-1} \sum_{k=1}^{n-1} S_{AR}^{(k)}(f)^{-1}. \tag{6.20}$$

We choose $n = 5$ here. To estimate the Capon spectra of each of the simulated time series, we use the Burg algorithm 17 to compute the coefficients a_i^j for $1 \leqslant i \leqslant j \leqslant 4$. To compute the theoretical spectra, we use the Levinson algorithm 13 to compute the coefficients a_i^j for $1 \leqslant i \leqslant j \leqslant 4$ from the theoretical autocorrelation coefficients $(r_0, r_1, r_2, r_3, r_4)$ of the theoretical autocorrelation matrix $\mathbf{R}$ used to simulate the signals.

We first present the Capon spectra in standard scale. In Figure 6.17, we present the computed Capon spectra for each time series. We use 1001 points to plot the spectra: we compute the values of the function $\widehat{S}_{per}(f)$ for f ranging from -0.5 to 0.5 with a step size of 0.001. To plot the PCA and classify the Capon spectra, we use 1000 points: we eliminate one of the two extreme values ($f = -0.5$ or $f = 0.5$) since these two values always coincide as they represent the same frequency. Figure 6.18 represents the PCA of the Capon spectra. The nearest centroid classifier is then performed on the Capon spectra. Figure 6.19 represents the PCA of the classification result obtained. Figure 6.20 represents the confusion matrix of the classification result.

We then present the Capon spectra in logarithmic scale (in dB). In Figure 6.21, we plot the Capon spectra in dB. The

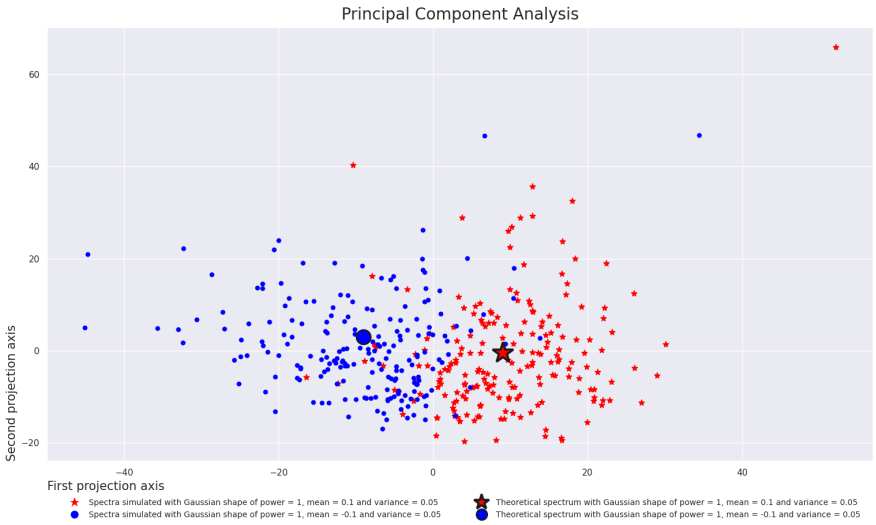

Figure 6.18: PCA on the Capon spectra

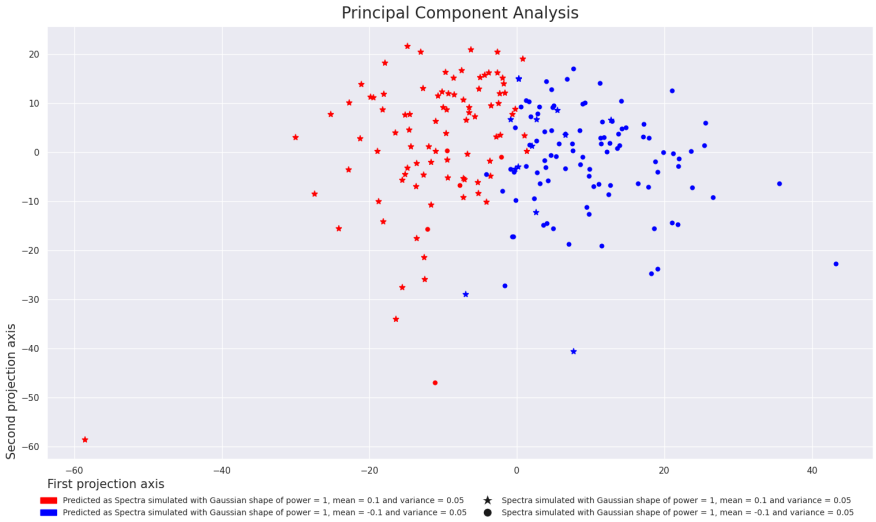

Figure 6.19: PCA of the classification result of the nearest centroid classifier on the Capon spectra

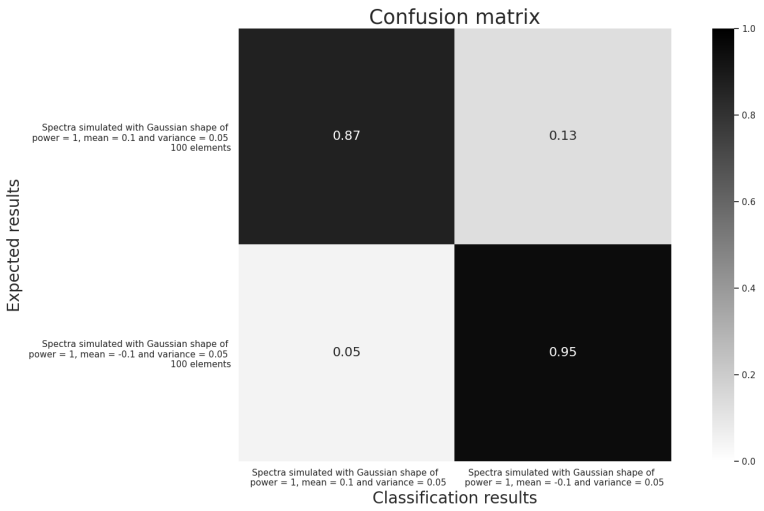

Figure 6.20: Confusion matrix of the classification result of the nearest centroid classifier on the Capon spectra

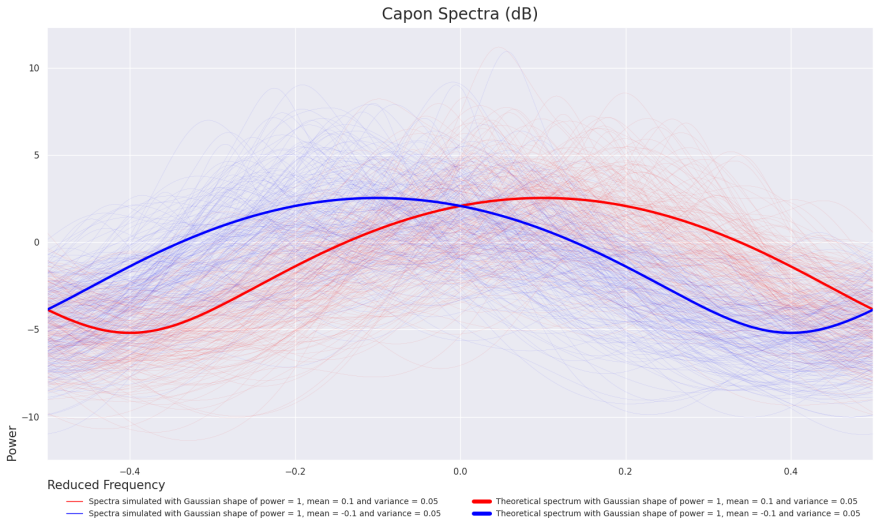

Figure 6.21: Capon spectra (dB)

figure 6.22 represents the PCA of the Capon spectra in dB. The nearest centroid classifier algorithm is then performed on the Capon spectra in dB. Figure 6.23 represents the PCA of the classification result obtained. Figure 6.24 represents the confusion matrix of the classification result.

6.3.4 HPD matrices $\mathcal{H}_n^+$

In this section, we represent the signals by their autocorrelation matrices $\mathbf{R} \in \mathcal{H}_n^+$. We first present the classification results obtained by endowing the space $\mathcal{H}_n^+$ with the Euclidean metric, then by using the information geometry metric presented in Section 4.2.1.

To estimate the autocorrelation matrices for each of the simulated time series, we start by using the Burg algorithm 17 to estimate the mean quadratic power coefficient p_0 and the reflection coefficients $\mu_i = a_i^i$ for $1 \leqslant i \leqslant n-1$. We choose $n = 5$ here. Then, we use the inverse Levinson algorithm 15 to compute the coefficients $(\widehat{r}_0, \widehat{r}_1, \ldots, \widehat{r}_{n-1})$ from the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1}) \in \mathbb{R}_+^* \times \mathbb{D}^{n-1}$ estimated previously. The theoretical correlation matrices are obtained directly: they are the matrices used to simulate the data.

We first present the classification of autocorrelation matrices in the space $\mathcal{H}_n^+$ endowed with the Euclidean metric. Figure 6.25 represents the PCA of the estimated autocorrelation matrices for each time series. The nearest centroid classifier algorithm is then performed on the estimated autocorrelation matrices using the Euclidean metric. Figure 6.26

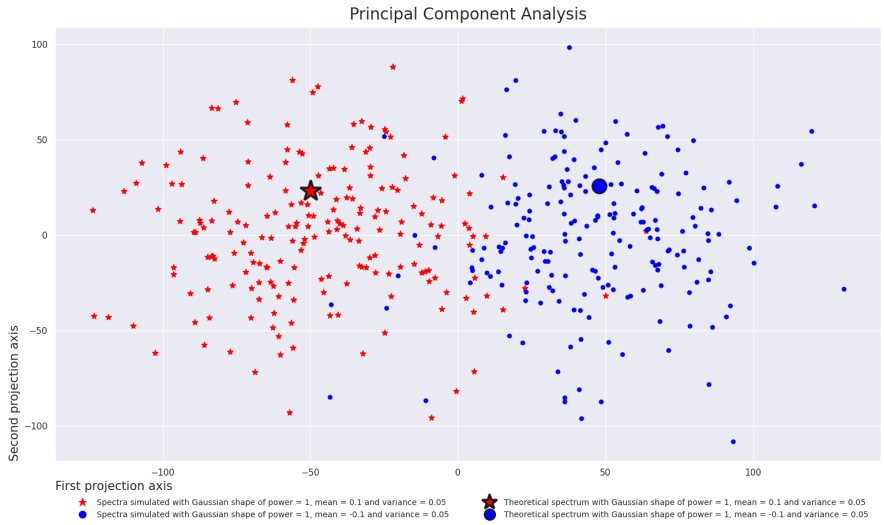

Figure 6.22: PCA on the Capon spectra (dB)

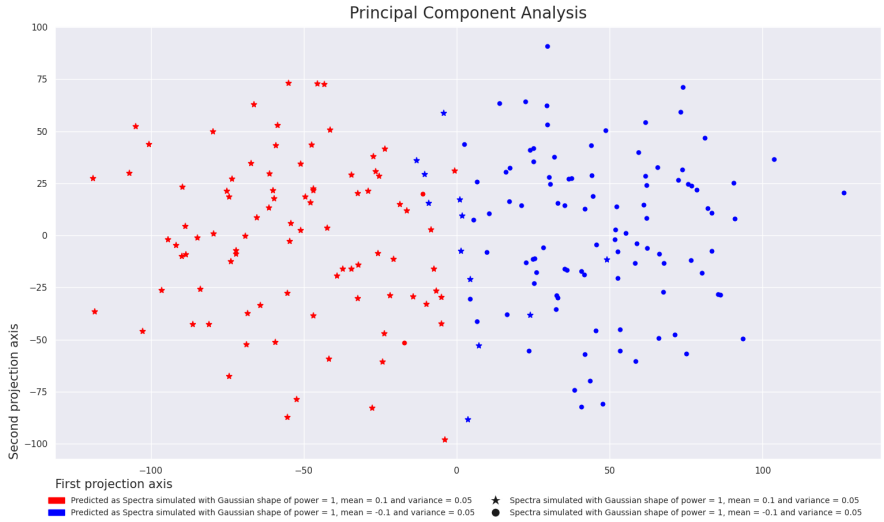

Figure 6.23: PCA of the classification result of the nearest centroid classifier on the Capon spectra (dB)

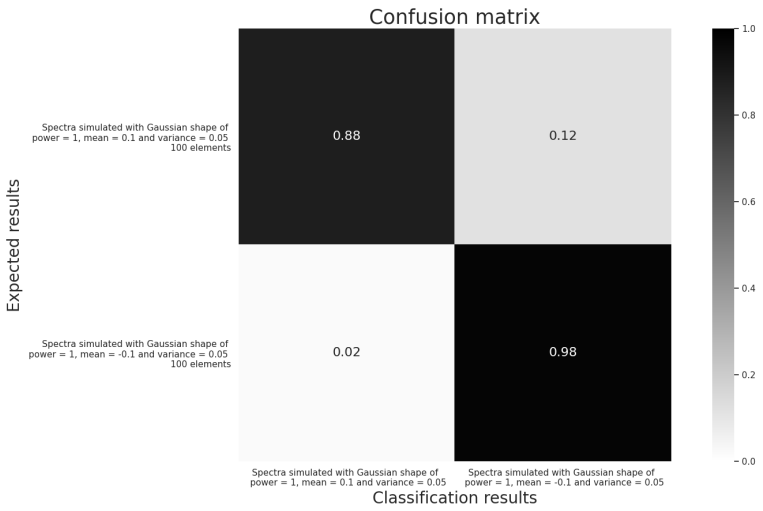

Figure 6.24: Confusion matrix of the classification result of the nearest centroid classifier on the Capon spectra (dB)

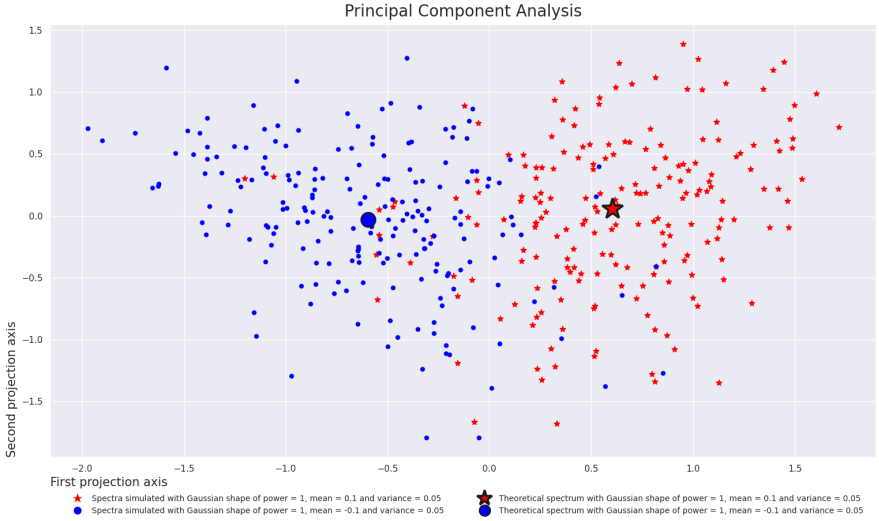

Figure 6.25: PCA on the HPD matrices using the Euclidean metric

represents the PCA of the classification result obtained. Figure 6.27 represents the confusion matrix of the classification result.

We now present the classification of the autocorrelation matrices in the space $\mathcal{H}_n^+$ endowed with the information geometry metric presented in Section 4.2.1. Figure 6.28 represents the Tangent PCA 10 of the autocorrelation matrices estimated for each time series. The nearest centroid classifier algorithm is then performed on the estimated autocorrelation matrices using the information geometry metric. Figure 6.29 represents the Tangent PCA of the classification result obtained. Figure 6.30 represents the confusion matrix of the classification result.

Comparing the confusion matrices 6.27 and 6.30, we note that for this experiment we obtain better classification performance by endowing the space $\mathcal{H}_n^+$ with the information geometry metric than with the Euclidean metric.

6.3.5 Positive real axis and Poincaré disks $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$

Finally, in this section we represent the time series by the coefficients $(p_0, \mu_1, \ldots, \mu_{n-1}) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$. We first present the classification results obtained by endowing the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ with the Euclidean metric, then by using the metric inspired by information geometry presented in Section 4.1.3.

To estimate the coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ of each of the simulated time series, we use the regularized Burg algorithm 17. Nous choose $n = 5$ here. The theoretical coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ can be computed using the

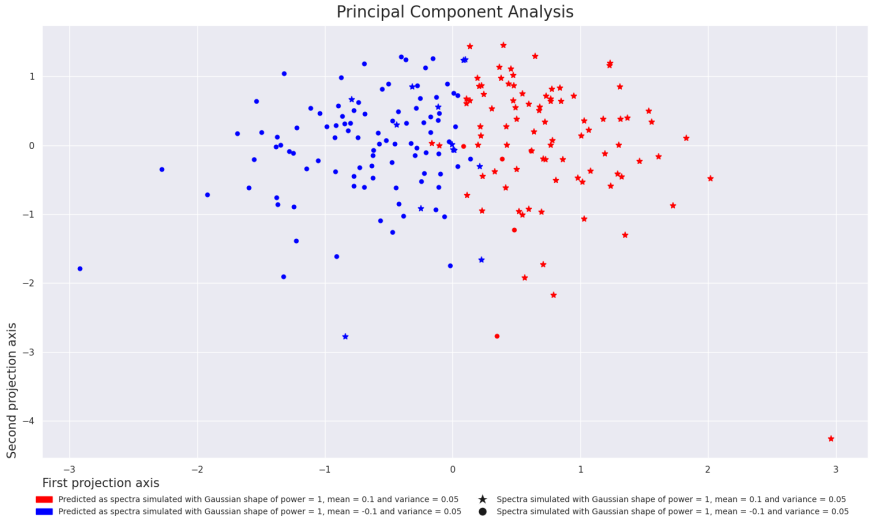

Figure 6.26: PCA of the classification result of the nearest centroid classifier on the HPD matrices using the Euclidean metric

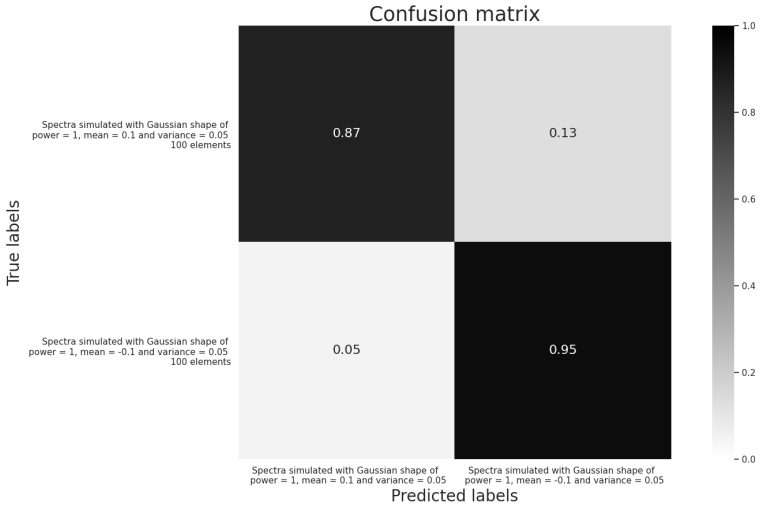

Figure 6.27: Confusion matrix of the classification result of the nearest centroid classifier on the HPD matrices using the Euclidean metric

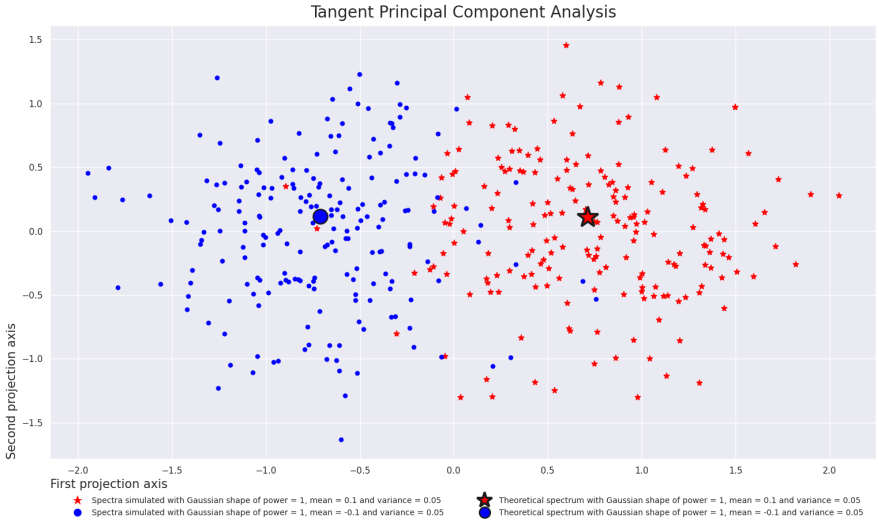

Figure 6.28: TPCA on the HPD matrices using the Information Geometry metric

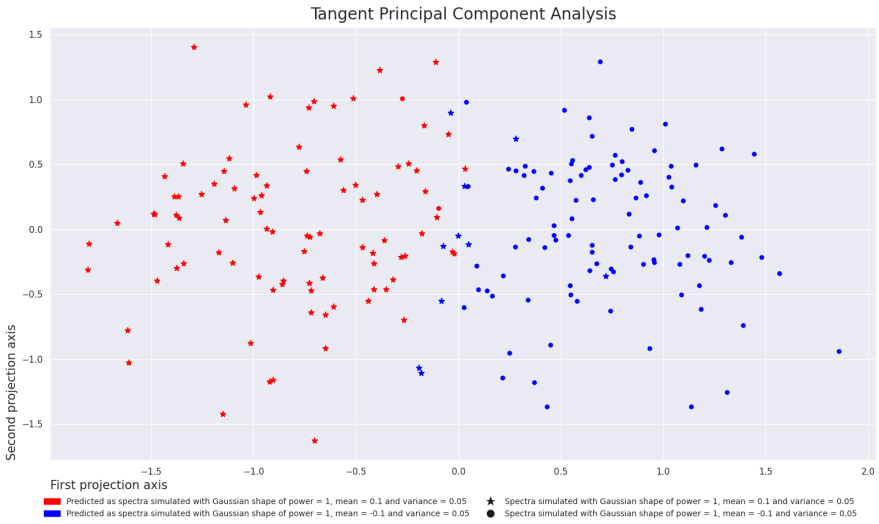

Figure 6.29: TPCA of the classification result of the nearest centroid classifier on the HPD matrices using the Information Geometry metric

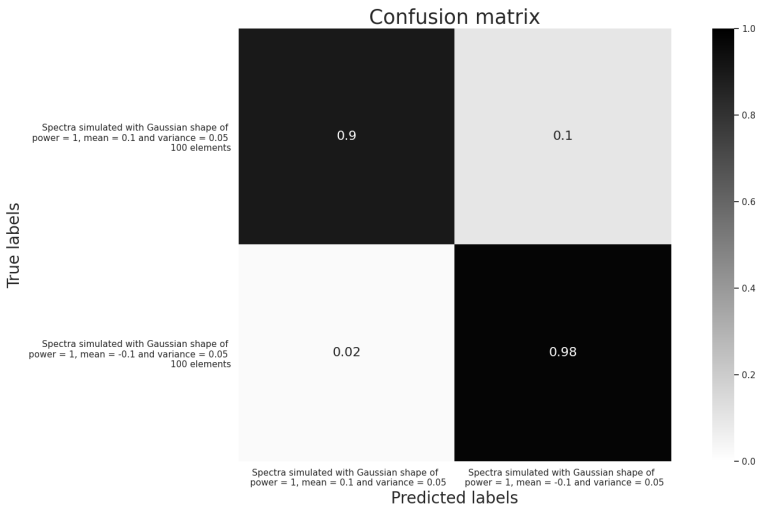

Figure 6.30: Confusion matrix of the classification result of the nearest centroid classifier on the HPD matrices using the Information Geometry metric

Levinson algorithm 14 from the autocorrelation coefficients $(r_0, r_1, \ldots, r_{n-1})$ of the autocorrelation matrix $\mathbf{R}$ used to simulate the data. In Figure 6.31 we visualize the estimated reflection coefficients $(\widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each of time series in the space $\mathcal{D}^{n-1}$. The theoretical coefficients are also represented.

We first present the classification of the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each of the time series in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the Euclidean metric. In Figure 6.32 we represent the PCA of the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each of time series. The nearest centroid classifier algorithm is then performed on the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the Euclidean metric. In Figure 6.33 we represent the PCA of the classification result obtained. In Figure 6.34 we represent the corresponding confusion matrix.

Finally, we present the classification of the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each time series in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the metric inspired by information geometry presented in Section 4.1.3. The figure 6.35 represents the TPCA of the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each time series. The nearest centroid classifier is then performed on the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the metric inspired by information geometry presented in Section 4.1.3. In Figure 6.36 we represent the TPCA of the classification result obtain. Figure 6.37 reprensents the corresponding confusion matrix.

Comparing the confusion matrices 6.34 and 6.37, we note that for this experiment, the classification performance achieved by endowing the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ with the metric inspired by the information geometry are better than performance obtained using the Euclidean metric.

Comparing the confusion matrices 6.30 and 6.37, we note that for this experiment the classification performance is slightly better on the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ than on the space $\mathcal{H}_n^+$ when these two spaces are endowed with the metric coming from information geometry. We therefore prefer to classify time series in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the metric inspired by information geometry presented in Section 4.1.3 since it is a low dimensional space endowed with a product metric, the computation times are therefore relatively short. In addition, the mean of several complex stationary centered Gaussian autoregressive Gaussian time series represented in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ is still complex stationary centered Gaussian autoregressive Gaussian time series.

By comparing all the confusion matrices obtained during this experiment, we notice that the best performance is obtained by representing the simulated time series to be classified in the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the Riemannian metric inspired by information geometry.

We will therefore use the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ endowed with the metric inspired by information geometry to perform the clustering of real radar data in Chapter 8.

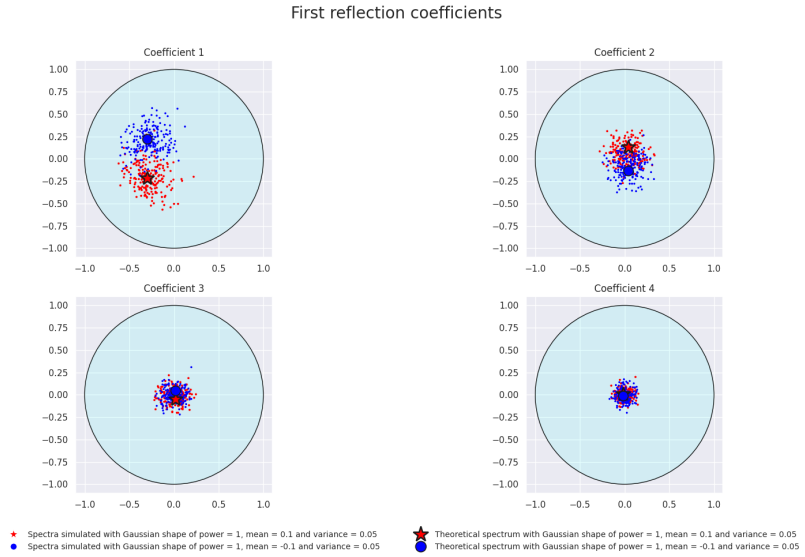

Figure 6.31: The four first reflection coefficients

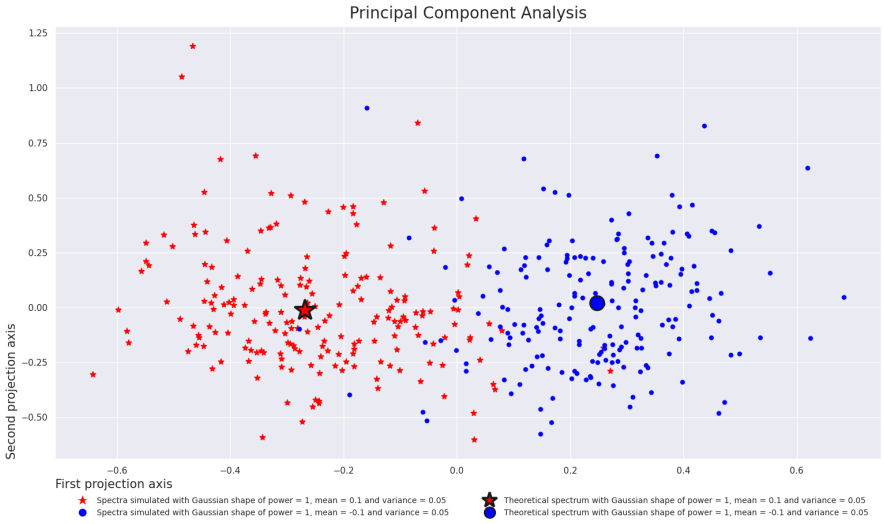

Figure 6.32: PCA on $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Euclidean metric

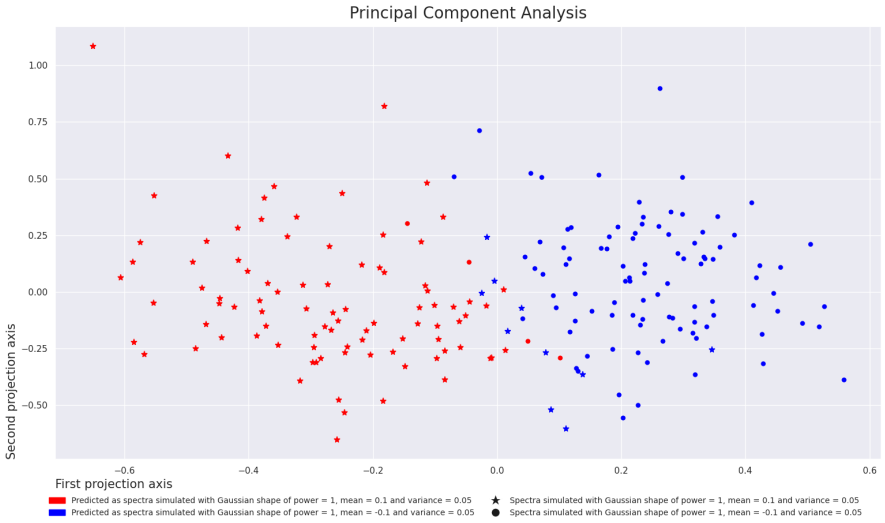

Figure 6.33: PCA of the classification result of the nearest centroid classifier on $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Euclidean metric

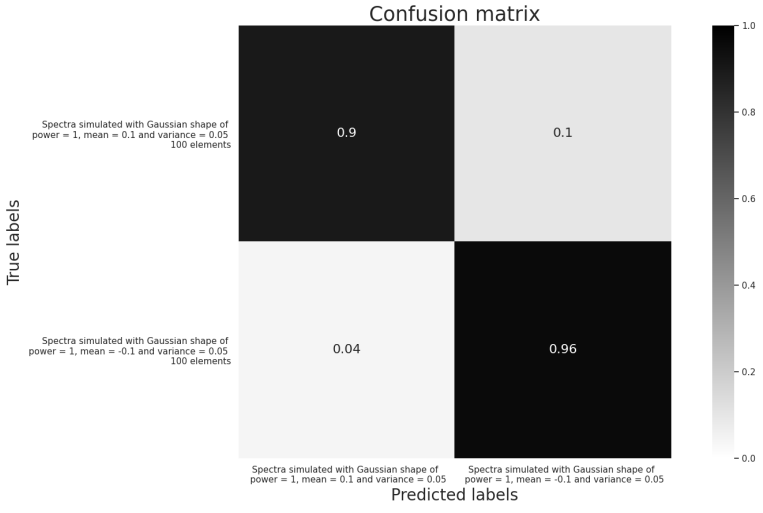

Figure 6.34: Confusion matrix of the classification result of the nearest centroid classifier on $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Euclidean metric

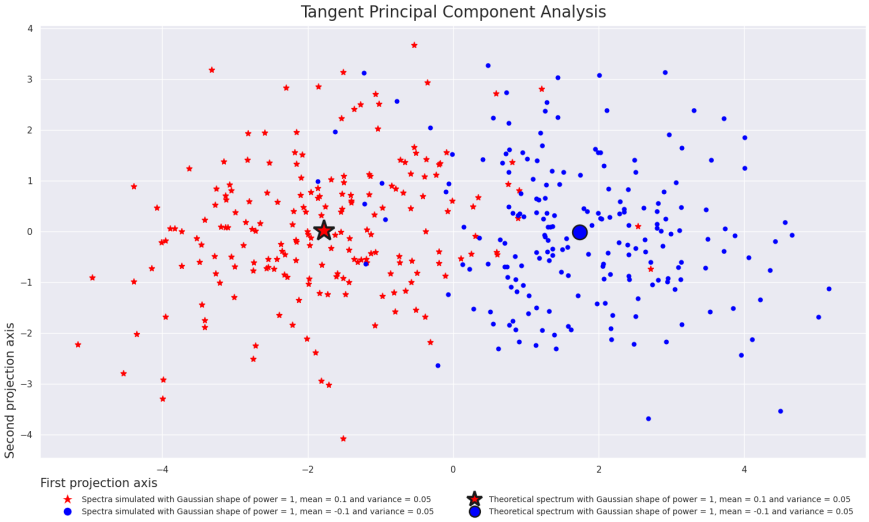

Figure 6.35: TPCA on $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Information Geometry metric

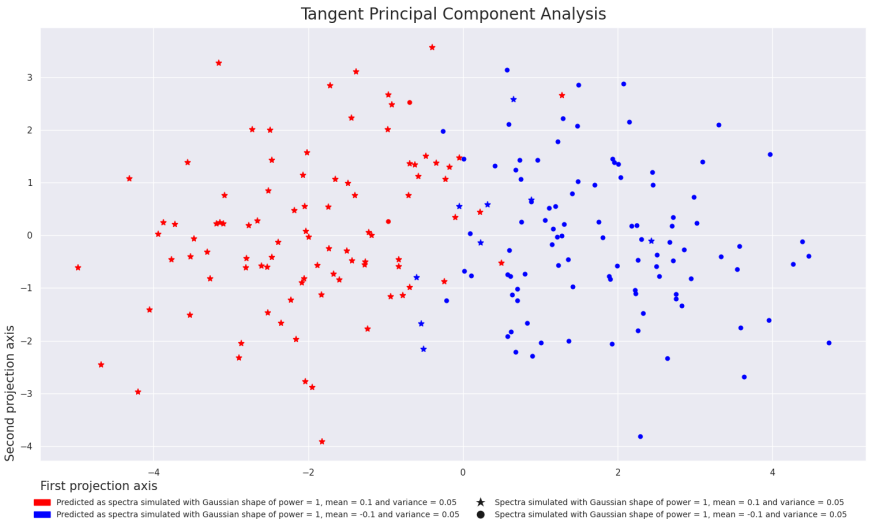

Figure 6.36: TPCA of the classification result of the nearest centroid classifier on $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Information Geometry metric

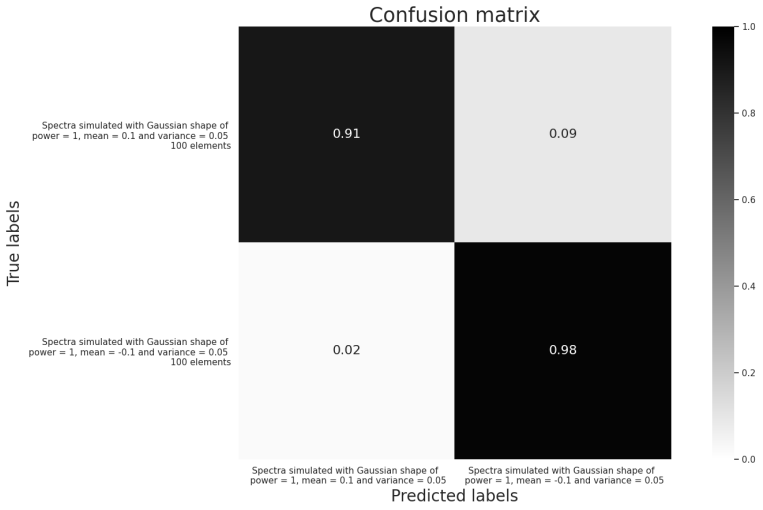

Figure 6.37: Confusion matrix of the classification result of the nearest centroid classifier on $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ using the Information Geometry metric

Chapter 7

Application to simulated multidimensional stationary centered Gaussian autoregressive time series classification

In this chapter, we will simulate complex multidimensional stationary centered autoregressive Gaussian autoregressive time series. We will then classify these time series.

Contents

7.1 Simulation model

A complex multidimensional stationary centered autoregressive Gaussian time series can be simulated using a Block-Toepliz Hermitian Positive Definite correlation matrix R:

$$Z = R^{1/2} X \tag{7.1}$$

with:

R: the autocorrelation matrix (Block-Toeplitz Hermitian Positive Definite).

X: a standard complex Gaussian random vector whose dimension is equal to the dimension of the time series to simulate times the length of the time series.

We can also use the equation of the autoregressive model:

$$Z(k) + \sum_{j=1}^{n-1} A_i^{n-1} \, Z(k-j) = W(k) \tag{7.2}$$

where W is the prediction error vector of size N, the covariance matrix Σ and the prediction coefficients A_j^{n-1} are square matrices of size $N \times N$.

The autocorrelation coefficients A_j^{n-1} can be computed from the autocorrelation coefficients R_i of the autocorrelation matrix $\mathbf{R}$ using Equation (2.123).

7.2 Simulated dataset

We simulate two datasets from two different Block-Toeplitz autocorrelation matrices $\mathbf{R}_0$ and $\mathbf{R}_1$. We recall that a Block-Toeplitz HPD matrix has the following structure:

$$\mathbf{R} = \begin{bmatrix} R_0 & R_1^H & R_2^H & \dots & R_{n-1}^H \\ R_1 & R_0 & R_1^H & \dots & R_{n-2}^H \\ R_2 & R_1 & R_0 & \dots & R_{n-3}^H \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{n-1} & R_{n-2} & R_{n-3} & \dots & R_0 \end{bmatrix}. \tag{7.3}$$

We choose here to simulate time series of length 2 using autocorrelation matrices $\mathbf{R}$ of sizes 6×6 made up of 3 blocks R_0, R_1 and R_2 of sizes 2×2. The Block-Toeplitz autocorrelation matrix $\mathbf{R}_0$ of the first dataset is defined by the autocorrelation coefficients:

$$R_0 = \begin{bmatrix} 10 & 1+i \\ 1-i & 8 \end{bmatrix} \quad R_1 = \begin{bmatrix} 4+2i & 2 \\ -1 & -4-i \end{bmatrix} \quad R_2 = \begin{bmatrix} 3 & -1 \\ 1 & -3+i \end{bmatrix}. \tag{7.4}$$

The Block-Toeplitz autocorrelation matrix $\mathbf{R}_1$ of the second dataset is defined by the autocorrelation coefficients:

$$R_0 = \begin{bmatrix} 10 & 1-i \\ 1+i & 8 \end{bmatrix} \quad R_1 = \begin{bmatrix} 4+2i & -2 \\ 1 & -4-i \end{bmatrix} \quad R_2 = \begin{bmatrix} 3 & 2i \\ -1 & -3+i \end{bmatrix}. \tag{7.5}$$

Before simulating data, we check that the Block-Toeplitz Hermitian matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ constructed from the autocorrelation coefficients described previously are HPD matrices. We simulate time series of length 20 and dimension $N = 2$. For each simulated time series, we use Equation (7.1) to simulate the 3 first temporal elements (since $n = 3$ here), then we use Equation (7.2) to simulate the 17 following elements.

We simulate 200 time series associated with the autocorrelation matrix $\mathbf{R}_0$ and 200 time series associated with the autocorrelation matrix $\mathbf{R}_1$. Each of these two datasets is then divided into a training dataset consisting of 100 time series and a testing dataset consisting of 100 time series.

Note that the diagonal coefficients of the autocorrelation coefficients R_0, R_1 and R_2 used to construct the Block-Toeplitz HPD matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ are equal, the time series simulated from the matrices $\mathbf{R}_0$ and $\mathbf{R}_1$ cannot therefore be dissociated using a one-dimensional method on only one of the two dimensions of the time series. To classify the simulated time series, we will use two representation spaces of multidimensional time series that we present in the next section.

7.3 Classification

In this section, we present the results of the classification of the time series simulated in the previous section. We first represent the simulated time series in the space of Block-Toeplitz HPD matrices $\mathcal{B}_{n,N}^+$ which we first endow with the Euclidean metric, then with the information geometry metric on HPD matrices $\mathcal{H}_{n\times N}^+$ described in Section 4.2.1. The simulated time series will then be represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ which we first endow with the Euclidean metric, then with the metric inspired by information geometry described in Section 4.2.3. In each case, we will use the nearest centroid classifier algorithm described in Section 5.3.3 to classify the data. Note that the dimension of the time series is $N = 2$, we choose the number of coefficients $n = 3$ to classify the data, which corresponds to an autoregressive model of order two.

7.3.1 Classification in $\mathcal{B}_{n,N}^+$

In this section, we classify the data simulated in Section 7.2 in the space $\mathcal{B}_{n,N}^+$.

For each of the simulated time series, we start by estimating their Block-Toeplitz autocorrelation matrices $\widehat{\mathbf{R}} \in \mathcal{B}_{n,N}^+$. To estimate these matrices, we use Algorithm 5 to estimate the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \dots, \widehat{\Omega}_{n-1}\right) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$. Then, we use the relationships between the different representation spaces of the complex multidimensional stationary centered Gaussian autoregressive time series presented in Section 2.2 to compute the autocorrelation matrices $\widehat{R} \in \mathcal{B}_{n,N}^+$. The coefficients $\left(\widehat{R}_0, \widehat{R}_1, \dots, \widehat{R}_{n-1}\right)$ of the matrix $\mathbf{R}$ can indeed be computed from the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \dots, \widehat{\Omega}_{n-1}\right) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ using Algorithm 4.

The data is first classified using the Euclidean metric, then the information geometry metric on HPD matrices $\mathcal{H}_{n\times N}^+$ described in Section 4.2.1.

We now present the classification of the signals simulated in Section 7.2 represented in the space $\mathcal{B}_{n,N}^+$ endowed with the Euclidean metric. Figure 7.1 represents the PCA of the studied dataset represented in the space $\mathcal{B}_{n,N}^+$ endowed with the Euclidean metric. The 200 time series simulated from the autocorrelation matrix $\mathbf{R}_0$ are represented by red stars, the 200 time series simulated from the autocorrelation matrix $\mathbf{R}_1$ are represented by blue circles. The matrix $\mathbf{R}_0$ is represented by a big red star and the matrix $\mathbf{R}_1$ is represented by a big blue circle. Figure 7.2 represents the PCA of the classification

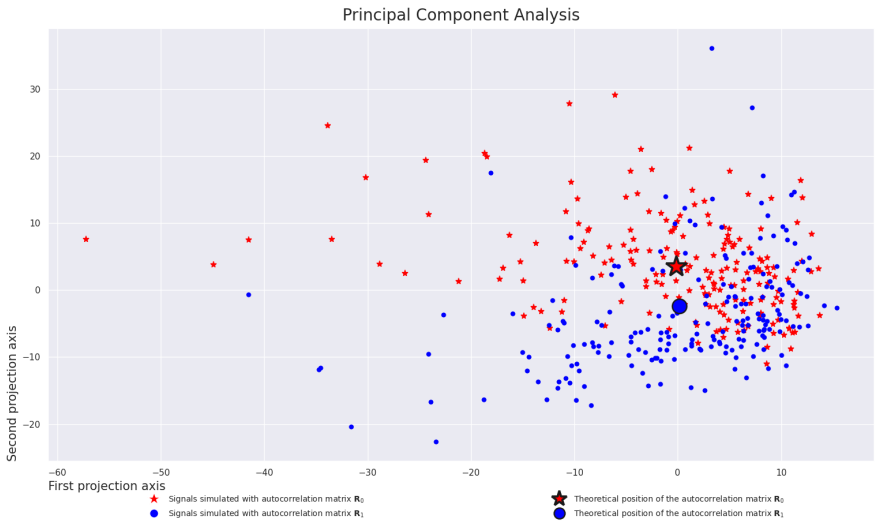

Figure 7.1: PCA on multidimensional covariance matrices using the Euclidean metric

result obtained using the nearest centroid classifier algorithm. True labels are characterized by the shape of the dots, predicted labels are characterized by the color of the dots. Finally, Figure 7.3represents the confusion matrix of the result of the classification obtained using the nearest centroid classifier algorithm.

We now present the classification of the time series simulated in Section 7.2 represented in the space $\mathcal{B}_{n,N}^{+}$ endowed with the information geometry metric on HPD matrices $\mathcal{H}_{n\times N}^{+}$ presented in Section 4.2.1. The figure 7.4 represents the Tangent PCA (see Section 5.2) of the dataset represented in the space $\mathcal{B}_{n,N}^{+}$ endowed with the information geometry metric. The figure 7.5 represents the TPCA of the classification result obtained using the nearest centroid classifier algorithm. Finally, Figure 7.6represents the confusion matrix of the result of the classification obtained using the nearest centroid classifier algorithm.

Comparing the confusion matrices of Figures 7.3 and 7.6, we note that the classification obtained using the information geometry metric gives a better result for this experience than the classification obtained using the Euclidean metric.

We now present the results of the classification when the time series are represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$.

7.3.2 Classification in $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$

In this section, we classify the data simulated in Section 7.2 in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$.

For each of the simulated time series, we start by estimating their coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \ldots, \widehat{\Omega}_{n-1}\right) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ using Algorithm 5. The data is first classified using the Euclidean metric, then the metric inspired by information geometry described in Section 4.2.3.

We now present the classification of the time series simulated in Section 7.2 represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ endowed with the Euclidean metric. The figure 7.7 represents the PCA of the studied dataset represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ endowed with the Euclidean metric. The 200 time series simulated from the autocorrelation matrix $\mathbf{R}_0$ are represented by red stars, the 200 time series simulated from the autocorrelation matrix $\mathbf{R}_1$ are represented by blue circles. The point $(P_0, \Omega_1, \ldots, \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ associated with the matrix $\mathbf{R}_0$ is represented by a big red star, the point associated with the matrix $\mathbf{R}_1$ is represented by a big blue circle. We use Algorithm 3 to compute the theoretical coefficients $(P_0, \Omega_1, \ldots, \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ from the theoretical autocorrelation matrices $\mathbf{R}$. Figure 7.8 represents the PCA of the classification result obtained using the nearest centroid classifier algorithm. Finally, Figure 7.9 represents the confusion matrix of the classification result obtained using the nearest centroid classifier algorithm.

We finally present the classification of the time series simulated in Section 7.2 represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ endowed with the metric inspired by information geometry presented in Section 4.2.3. The figure 7.10 represents the Tangent PCA of the dataset to be classified represented in the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ endowed with the metric presented in Section 4.2.3. The figure 7.11 represents the Tangent PCA of the classification result obtained using the nearest centroid classifier algorithm. Finally, Figure 7.12 represents the confusion matrix of the classification result obtained using the nearest centroid classifier algorithm.

Comparing Figures 7.9 and 7.12, we note that the classification obtained using the metric inspired by information geometry gives a better result for this experience than the classification obtained using the Euclidean metric.

Comparing the confusion matrices of Figures 7.6 and 7.12, we note that the results of the classification is very good on the spaces $\mathcal{B}_{n,N}^{+}$ and $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ when using the metric inspired by information geometry. The performances obtained

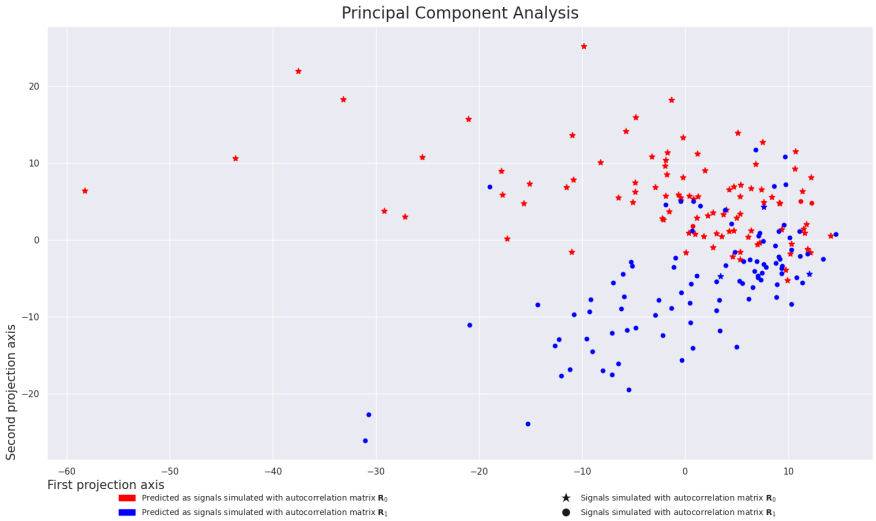

Figure 7.2: PCA of the classification result of the nearest centroid classifier on multidimensional covariance matrices using the Euclidean metric

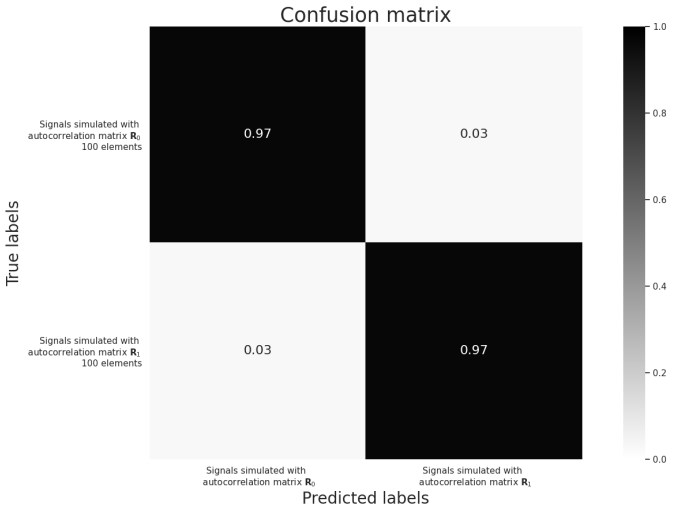

Figure 7.3: Confusion matrix of the classification result of the nearest centroid classifier on multidimensional covariance matrices using the Euclidean metric

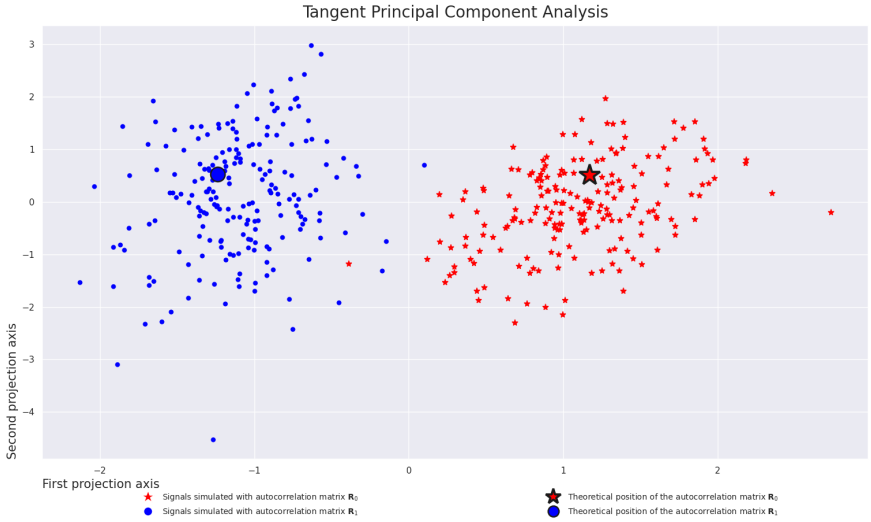

Figure 7.4: TPCA on multidimensional covariance matrices using the Information Geometry metric

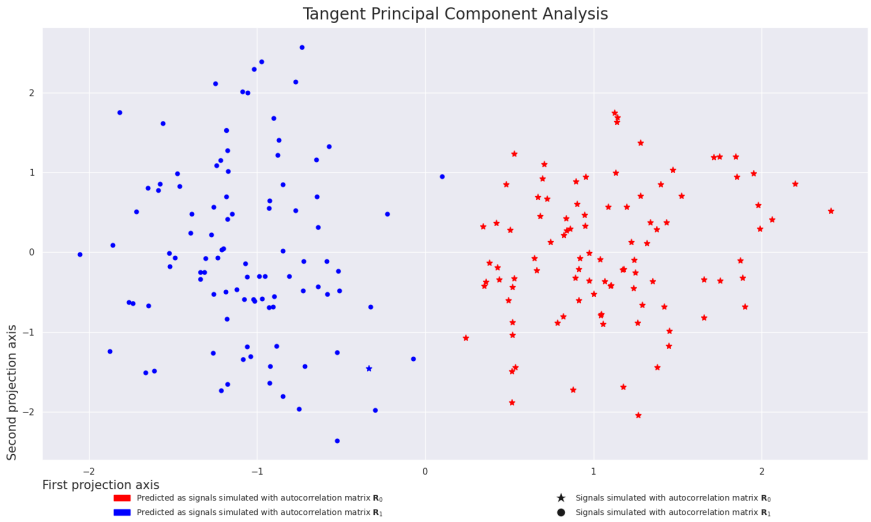

Figure 7.5: TPCA of the classification result of the nearest centroid classifier on multidimensional covariance matrices using the Information Geometry metric

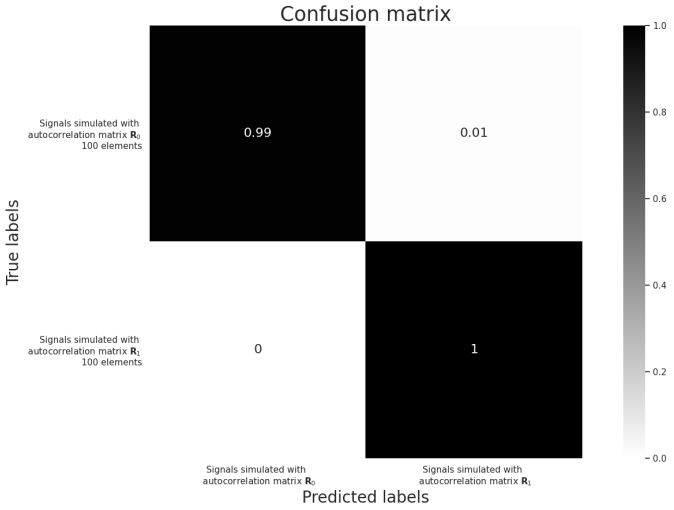

Figure 7.6: Confusion matrix of the classification result of the nearest centroid classifier on multidimensional covariance matrices using the Information Geometry metric

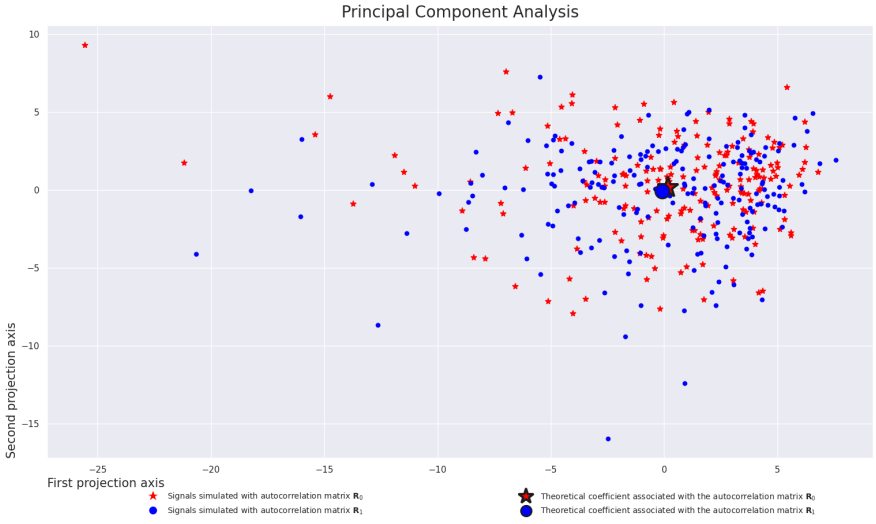

Figure 7.7: PCA on $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ using the Euclidean metric

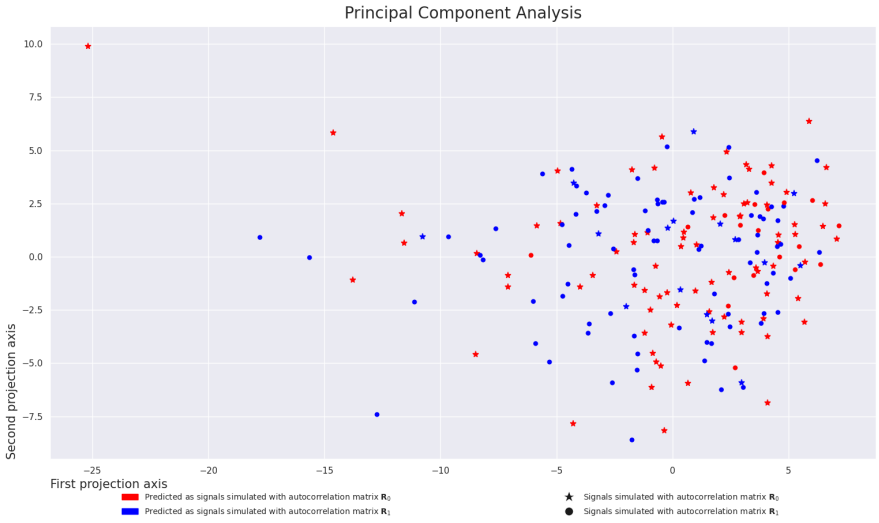

Figure 7.8: PCA of the classification result of the nearest centroid classifier on $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ using the Euclidean metric

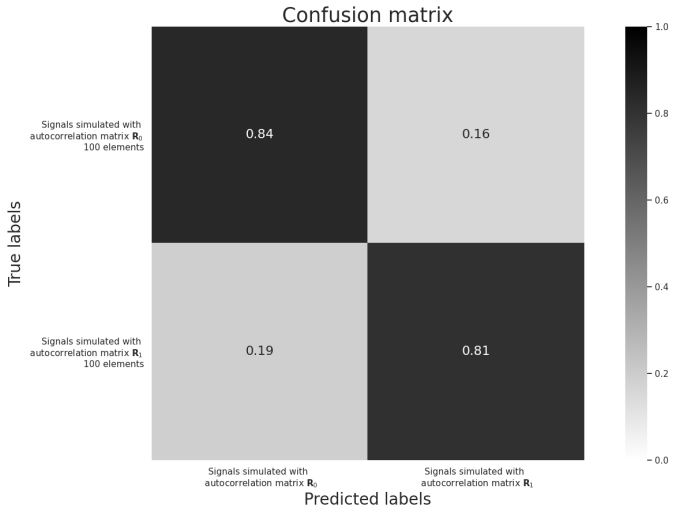

Figure 7.9: Confusion matrix of the classification result of the nearest centroid classifier on $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ using the Euclidean metric

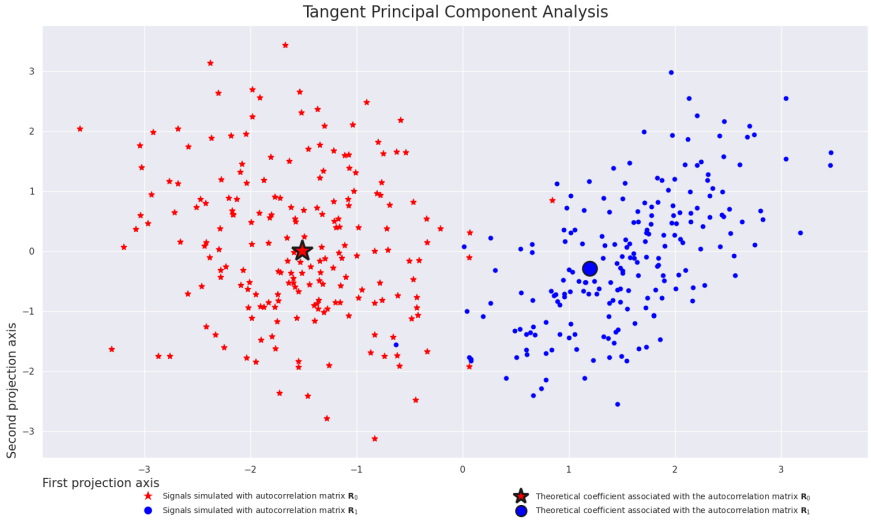

Figure 7.10: TPCA on $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

on the space $\mathcal{B}_{n,N}^+$ are slightly better. However, the space $\mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ has other advantages: it is of smaller dimension, endowed with a product metric, the computation times are therefore potentially shorter and the mean of the coefficients $(P_0, \Omega_1, \ldots, \Omega_{n-1}) \in \mathcal{H}_N^+ \times \mathcal{SD}_N^{n-1}$ of several complex multidimensional stationary centered Gaussian autoregressive time series is equal to the coefficients of a complex multidimensional stationary centered Gaussian autoregressive time series.

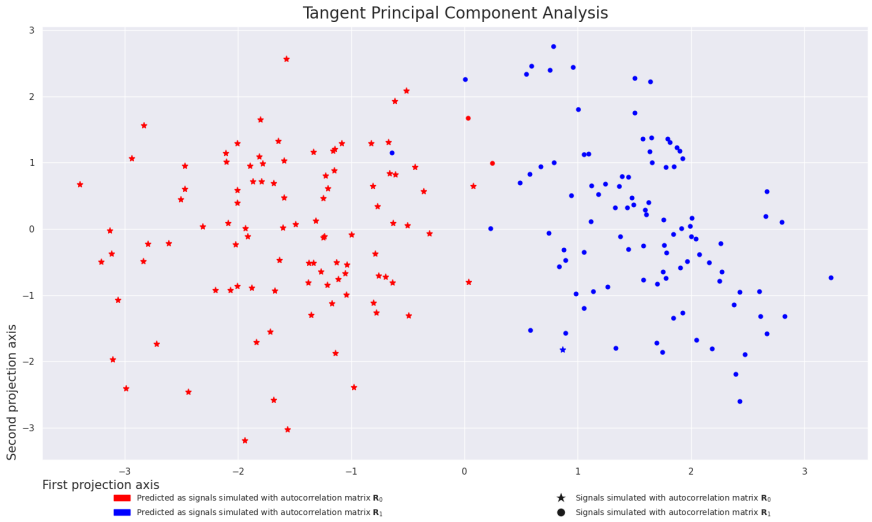

Figure 7.11: TPCA of the classification result of the nearest centroid classifier on $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

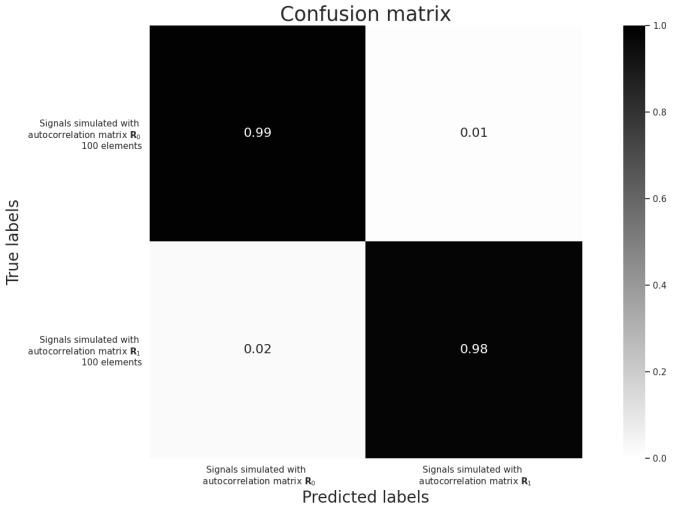

Figure 7.12: Confusion matrix of the classification result of the nearest centroid classifier on $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

Chapter 8

Application to radar clutter clustering

The objective of this chapter is the clustering of radar clutter coming from a recording made in Saint-Mandrier, in France [17–20]. The exact same dataset as the one presented here have been used by Pierre-Yves Lagrave in [61–63] for radar clutter classification using Neuronal Networks and the Lie groups theory. Here we assume that the presence of moving targets is rare in the recorded data, so we assume that the recorded signals are related to radar clutter, the radar clutter being the signals recorded by a radar related to its environment: waves, fields, forests... In Section 8.1, we will present the structure of the recorded data. The data will then be divided into distance cells, each cell representing a small area around the radar in which a one-dimensional time series has been recorded. In Section 8.2, we will assume that the complex one-dimensional time series recorded in each cell is a wide-sense stationary centered autoregressive Gaussian time series. These time series have been presented in detail in Section 2.1. In Section 8.3, we will estimate the coefficients of the autoregressive model $(p_0, \mu_1, \dots, \mu_{n-1})$ using the Burg algorithm presented in Section 2.1.8 and detailed in Appendix C. We recall that the coefficient p_0 corresponds to the mean quadratic power of the time series and the coefficients $(\mu_1, \dots, \mu_{n-1})$ are called reflection coefficients and are coefficients of the autoregressive model $\left(\mu_i := a_i^i\right)$. The reflection coefficients correspond to the Doppler information. In this chapter, we would like to cluster the data using only the Doppler information: we will not use the mean quadratic power coefficient p_0 and we will classify the data on the reflection coefficients $(\mu_1, \dots, \mu_{n-1}) \in \mathcal{D}^{n-1}$. We define the manifold $\mathbb{D}^{n-1}$ as the space $\mathcal{D}^{n-1}$ endowed with the metric of the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ defined in Section 4.1.3 omitting the power coefficient p_0. Finally, we will use the k-means algorithm presented in Section 5.4.3 on the Riemannian manifold $\mathbb{D}^{n-1}$ to perform the radar clutter clustering in Section 8.4.

Contents

8.1 Radar data structure

We first detail the input radar data. The data are studied after pulse compression. In this section, the input data will be taken on a single burst for a single elevation corresponding to the horizontal beam. Therefore, the radar provides us a 2D complex matrix presented in Equation (8.1).

$$U = \begin{bmatrix} u_{0,0} & u_{0,1} & \dots & u_{0,n-1} \\ u_{1,0} & u_{1,1} & \dots & u_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-1,0} & u_{N-1,1} & \dots & u_{N-1,n-1} \end{bmatrix} \tag{8.1}$$

Here the first coordinate corresponds to the spatial axis (index close to zero corresponds to cells close to the radar); the second coordinate represents the temporal axis (pulse index in the burst). The parameter n denotes the number of pulses of the burst and corresponds to the length of the temporal axis. The parameter N represents the total number of cells and corresponds to the lenght of the spatial axis. The data to classify are the cells, each cell being represented by a row of the

matrix U. In Section 8.4, the radar data will be clustered over 150 bursts (only the beam by zero elevation will be used). Therefore, the input data will be 150 matrices $U_1, U_2, \ldots, U_{150}$ with a similar structure than the matrix U presented in Equation (8.1). Each matrix U_i is constituted of N_i cells, each cell being caracterized by a time series of length n_i.

We now present the modeling assumptions made on the radar signal recorded in each of the distance cells.

8.2 Radar signal modeling

For each cell, we assume that the complex observation vector Z can be represented using a SIRV (Spherically Invariant Random Vectors) model [13], [42]:

$$Z = \underbrace{\sqrt{\tau} R^{1/2} x}_{\text{information coming from the environment}} + \underbrace{b_{radar}}_{\text{thermal noise coming from the radar itself}} \tag{8.2}$$

with:

τ**:** clutter texture; it is a positive real random variable independent from x and b_{radar}.

R**:** scaled environment autocorrelation matrix (Toeplitz Hermitian Positive Definite whose diagonal coefficients are equal to one).

x**,** b_{radar}**:** independent standard complex Gaussian random vectors whose dimension is equal to the number of pulses.

The product τR represents the environment autocorrelation matrix.

We assume that the mean power of the signal coming from the environment $\sqrt{\tau} R^{1/2} x$ is sufficiently larger than the radar noise b_{radar} to consider that the autocorrelation matrix of the whole signal Z will be close enough to the environment autocorrelation matrix τR to characterize the clutter.

We therefore place ourselves within the framework of the complex one-dimensional time series which are assumed to be wide-sense stationary centered autoregressive Gaussian processes. These time series have been presented in Section 2.1.

We now present how to estimate the coefficients of the autoregressive model from the observed time series.

8.3 Model parameters estimation

In our clustering problem, the autoregressive coefficients $(p_{i,0}, \mu_{i,1}, \ldots, \mu_{i,n-1})$ of the radar signal will be estimated independently for each cell $\mathbf{u}_i = [u_{i,0}, u_{i,1}, \ldots, u_{i,n-1}]^T$:

$$U = \begin{bmatrix} u_{0,0} & u_{0,1} & \cdots & u_{0,n-1} \\ u_{1,0} & u_{1,1} & \cdots & u_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N-1,0} & u_{N-1,1} & \cdots & u_{N-1,n-1} \end{bmatrix} \begin{matrix} \rightarrow & (\widehat{p}_{0,0}, \widehat{\mu}_{0,1}, \ldots, \widehat{\mu}_{0,n-1}) \\ \rightarrow & (\widehat{p}_{1,0}, \widehat{\mu}_{1,1}, \ldots, \widehat{\mu}_{1,n-1}) \\ & \vdots \\ \rightarrow & (\widehat{p}_{N-1,0}, \widehat{\mu}_{N-1,1}, \ldots, \widehat{\mu}_{N-1,n-1}) \end{matrix} \tag{8.3}$$

To estimate the autoregressive coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ from the column vector $\mathbf{u}$ associated with a row of matrix 8.3, we use the Burg algorithm 17 presented in Section 2.1.8 and decribed in detail in Appendix C.

We now would like to cluster the data using only the Doppler information. Hence, we will not use the mean quadratic power coefficient p_0 and we will classify the data on the reflection coefficients $(\mu_1, \ldots, \mu_{n-1}) \in \mathcal{D}^{n-1}$. We define the manifold $\mathbb{D}^{n-1}$ as the space $\mathcal{D}^{n-1}$ endowed with the metric of the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ defined in Section 4.1.3 omitting the power coefficient p_0. In Section 8.4, we will use the k-means algorithm presented in Section 5.4.3 on the Riemannian manifold $\mathbb{D}^{n-1}$ to perform the radar clutter clustering.

8.4 Radar clutter clustering

We now present the clustering result using the k-means algorithm with four classes on the radar data recorded at Saint-Mandrier. We display on Figure 8.1 a ground map of Saint-Mandrier (France) in which the radar geographical position is represented by a red dot. The dataset contains 150 bursts of different shapes: both the number of pulses n and the number of cells N (see the matrix U defined in Equation (8.1)) vary from one burst to another.

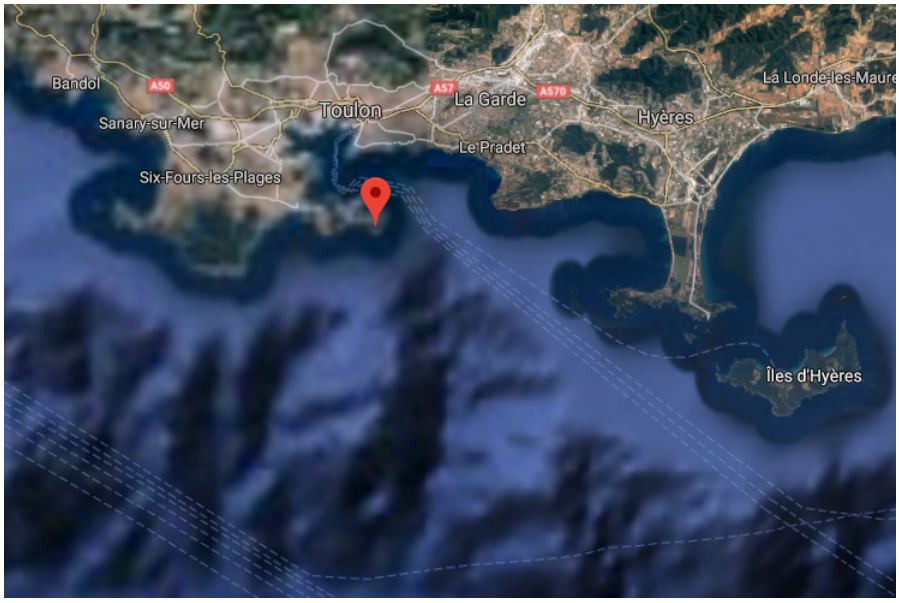

Figure 8.1: Ground map Saint-Mandrier

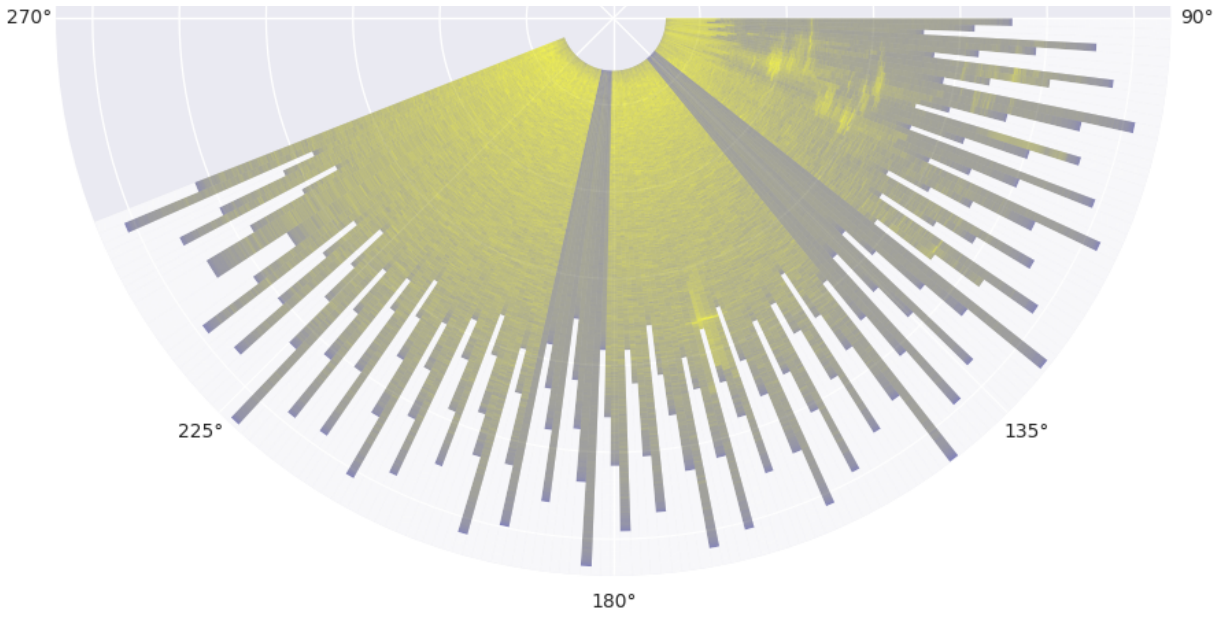

Figure 8.2: Power map Saint-Mandrier

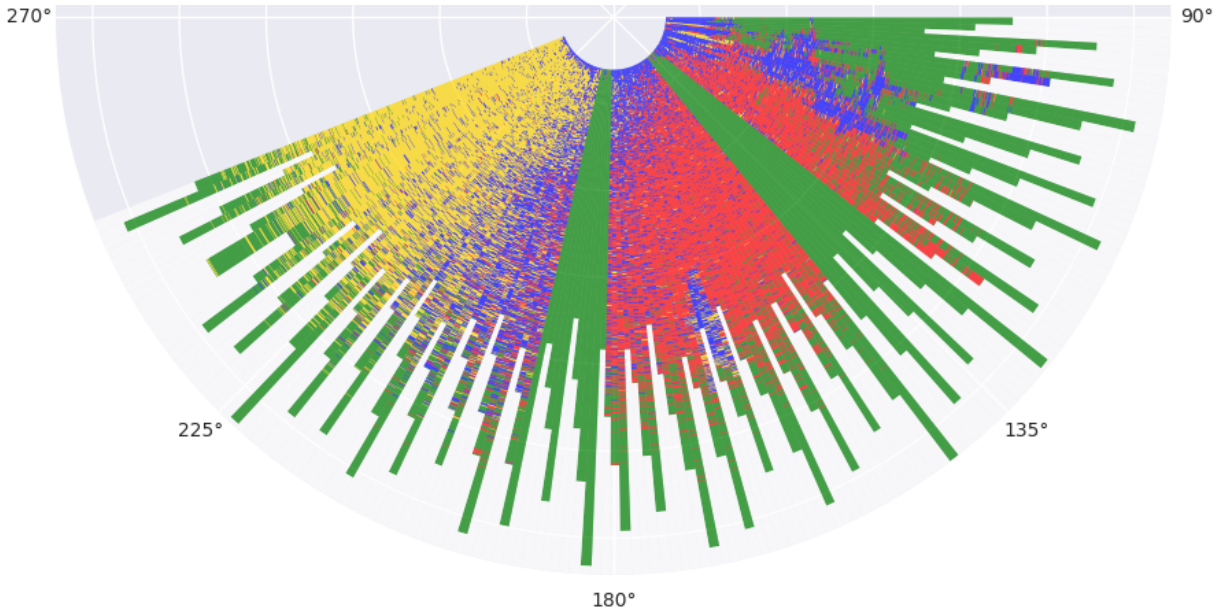

Figure 8.3: Clustering map obtained with the k-means algorithm for 4 classes on the reflection coefficients

We plot on Figure 8.2 a map of the averaged power recorded by the radar cell per cell. The power is represented from blue to yellow, yellow meaning that a lot of power has been recorded. The two well delimited grey zones in the south may correspond to areas where the radar was in listen-only mode and did not emit any pulses.

We would like to cluster the data cell by cell. We decided not to use the power coefficient p_0 to cluster the radar data: we only use the Doppler information which is contained in the reflection coefficients $(\mu_1, \dots, \mu_{n-1})$. As the length of the signals varies from one burst to another, we decided to represent every time series in $\mathbb{D}^{n-1}$ where n denotes the length of shortest time series. As the main Doppler information is contained in the first reflection coefficients, the loss of information is minor.

Figure 8.3 represents the result of the k-means clustering for four classes on the manifold $\mathbb{D}^{n-1}$. In Figure 8.4, we visualize the clustering result on the four first reflection coefficients; the colors correspond to those of Figure 8.3. The first reflection coefficient of the yellow, blue and red clusters are close to the edge of the complex unit disk: they correspond to spectra with a narrow peak. The main difference between these three clusters is the argument of the first reflection coefficient which is related to the average Doppler of the signal as shown in Equation (2.99) for stationary signals with a Gaussian shape. The yellow cluster corresponds to waves moving towards the radar. The red cluster corresponds to waves moving away from the radar. The blue cluster corresponds to elements with a zero radial speed: the waves whose direction is perpendicular to the radar radial axis, the coast and the islands. The green cluster corresponds to spectra with a flatter shape: they correspond to the distance cells far away from the radar, the cells hidden by the coast or the islands and the cells in the two angular sectors in which the radar did not emit any pulses.

We now present a second clustering result in which the input time series have been modified to have a zero average Doppler. To shift the average Doppler of the time series $u(0), u(1), \dots, u(n-1)$ to zero, we first estimate the first reflection coefficient μ_1 using the Burg algorithm. Then, denoting $\mu_1 = r_1 e^{i\theta_1}$, we replace the original coefficient $u(k)$ by the coefficient $\widetilde{u}(k) = u(k)e^{-ik\theta_1}$. We then estimate the reflection coefficients $\widetilde{\mu}_1, \widetilde{\mu}_2, \dots, \widetilde{\mu}_{n-1}$ of this new time series. We finally perform the k-means algorithm with four classes on the reflection coefficients $\widetilde{\mu}_1, \widetilde{\mu}_2, \dots, \widetilde{\mu}_{n-1}$ on the manifold $\mathbb{D}^{n-1}$. We present in Figure 8.5 the result of the clustering on the four first reflection coefficients. Note that the first reflection coefficients $\widetilde{\mu}_1$ observed in the first complex unit disk belongs to the seqment $]-1, 0]$ of the real axis as the time series has been modified to have a zero average Doppler. We also note that the result of the clustering is closely related to the modulus of the first reflection coefficient. We present in Figure 8.6 the map of the clustering result. We note some similarities between this map and the map of the average power presented in Figure 8.2.

We chose the k-means algorithm 5.4.3 here rather than the AHC 5.4.1 or the mean-shift 5.4.2 algorithms for complexity reasons: the complexity of the k-means algorithm is linear with respect to the number of points of the dataset whereas the complexity of the AHC or the mean-shift algorithm is quadratic.

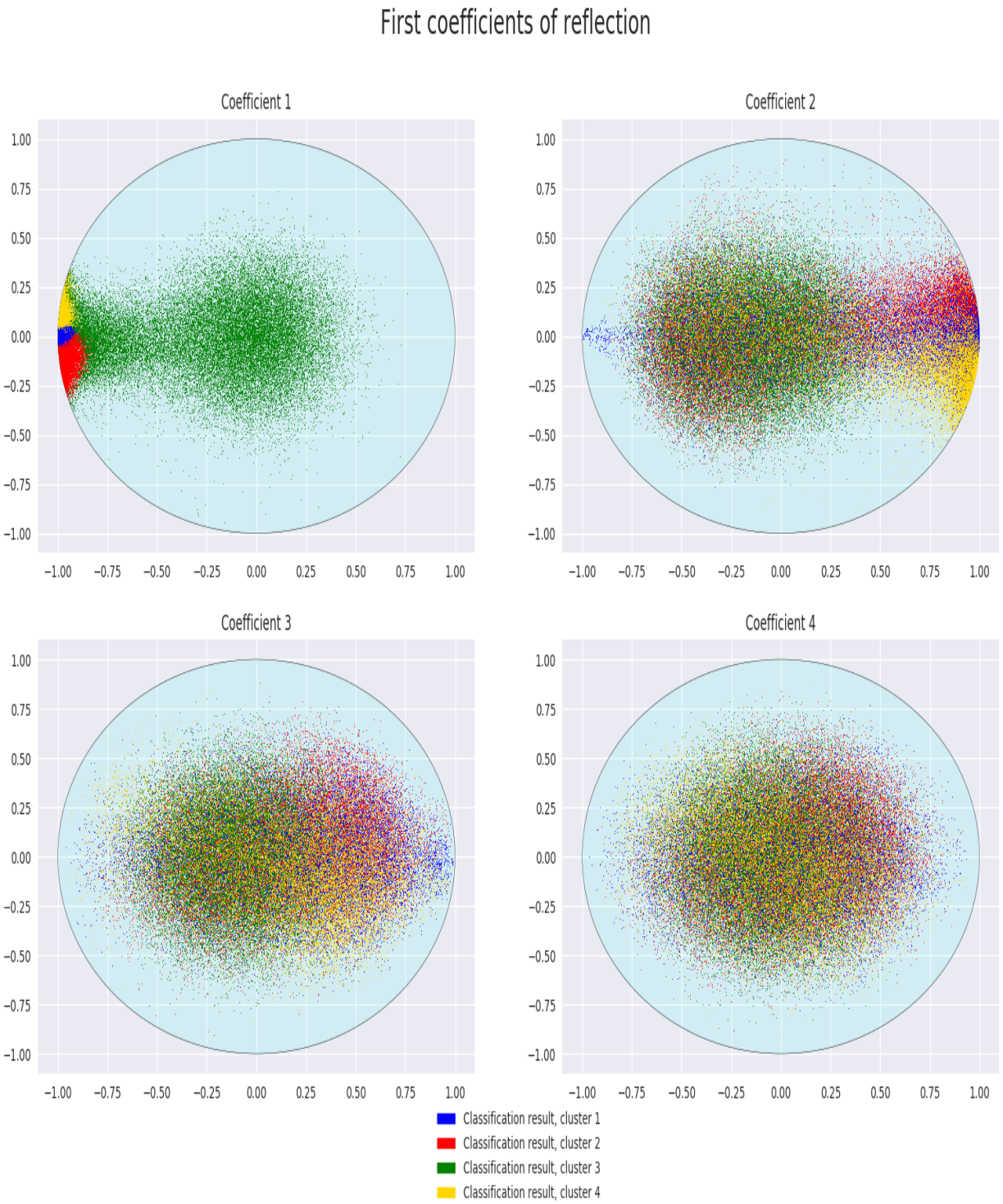

Figure 8.4: Visualization of the clustering result obtained with the k-means algorithm for 4 classes on the four first coefficients of reflection

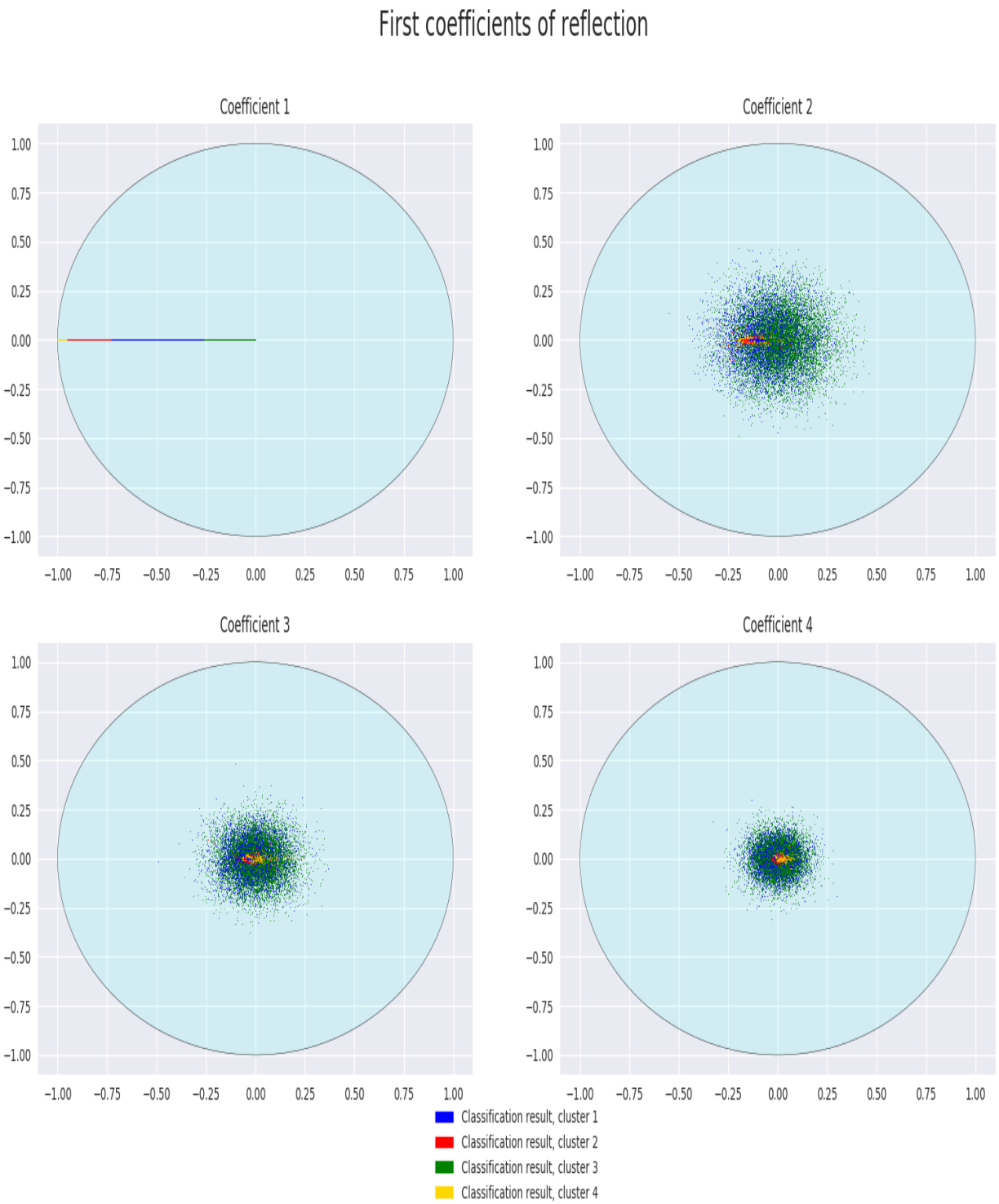

Figure 8.5: Visualization of the clustering result obtained with the k-means algorithm for 4 classes on the four first coefficients of reflection

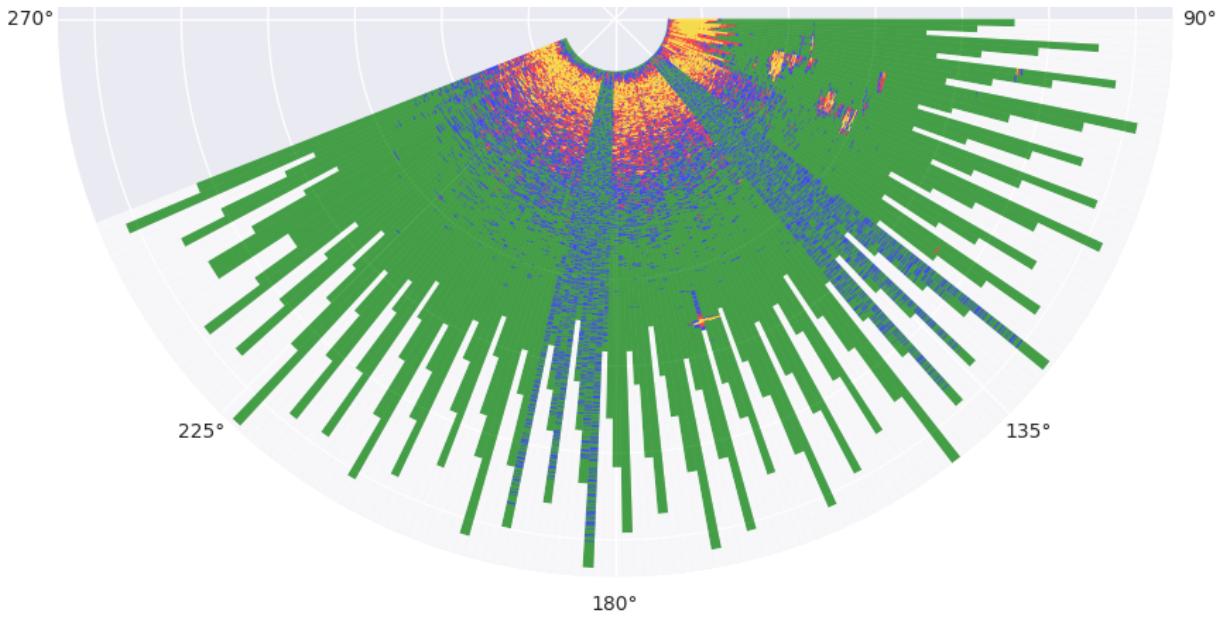

Figure 8.6: Visualization of the clustering result obtained with the k-means algorithm for 4 classes on the four first coefficients of reflection

Chapter 9

Application to stereo audio classification

In this chapter, we classify stereo audio recordings of stationary sounds. For this purpose, the soundtracks of YouTube videos corresponding to stationary sounds have been downloaded. We will have three distinct classes of sounds: wood fire recordings, waterfall recordings, and wind recordings. We will divide the records from each of these three classes into a training dataset and a testing dataset. The audio dataset will be presented in detail in Section 9.1. We will assume that the downloaded audio dataset follows the centered Gaussian autoregressive multidimensional model presented in Section 2.2. We will compare several representation spaces of the studied multidimensional signals. For each of these representation spaces, we will first use the TPCA presented in Algorithm 10 to visualize the downloaded audio dataset. Then we will use the supervised classification algorithm of the nearest centroid classifier presented in Section 5.3.3 to classify the audio dataset.

Contents

9.1 The stereo audio dataset

In this section, we present the audio dataset on which the classification will be performed. The recordings are extracts of 100 seconds from soundtracks of YouTube videos. Each extract comes from a different video.

The downloaded soundtracks correspond to stationary sounds. For this study, we downloaded three types of sounds: wood fires, waterfalls and wind noise. There are other stationary sounds: dense rain, the fire of a gas stove, the noise of a ventilation system, the noise of an air conditioning system, the noise of an engine in general when its regime is stable, the sound of the leaves of a tree continuously agitated by the wind...

To download soundtracks from YouTube videos, we used a free converter site called YouTube to WAV whose url is: `https://loader.to/en14/youtube-wav-converter.html`. All audio files obtained with this converter have a sample rate of $48,000Hz$. An audio recording from the same YouTube video may have a different sample rate when downloaded using another site to extract the soundtrack from a YouTube video, so it is important to download all the soundtracks with converters having the same sample rate to be able to compare the downloaded soundtracks.

Each downloaded video is characterized by its sample rate ($48,000$ Hz here) and a two-dimensional real matrix. The number of rows of this matrix is equal to the sample rate multiplied by the duration of the record in seconds. The number of columns is equal to two, it corresponds to the number of channels of the record. For each record, we check that the two columns of this matrix are distinct. Indeed, the two columns are equal when the original recording is in mono.

Finally, we extract 100 seconds from each downloaded soundtrack. The extract of 100 seconds generally corresponds to the 10 to 110 seconds of the original soundtrack. Indeed, it is preferable not to use the very first seconds of the audio recording because an effect gradually increasing the sound volume over the first seconds is sometimes used on YouTube videos.

For each of the three classes (wood fires, waterfalls and winds), we separate our records into two parts: half of the records will be used as a training dataset, the other half will be used as a testing dataset. We summarize the constitution of the dataset used in this chapter in table 9.1.

We will now visualize and classify this dataset in different representation spaces in Section 9.2.

Table 9.1: Number of records of the stereo audio dataset

	Training set	Testing set
Fires	24	25
Waterfalls	21	22
Winds	32	33

9.2 Classification results

In this section, we classify the audio signals presented in Section 9.1 using the nearest centroid classifier algorithm studied in Section 5.3.3. We use five different spaces to represent the signals to be classified. The first three representation spaces are linked to one-dimensional signal classification methods already presented for simulated signals in Chapter 6: only one of the two recording channels is then used to perform the classification. The following two representation spaces are related to the multidimensional signal classification methods already presented for simulated signals in Chapter 7. We start by representing the audio recordings by their periodograms in dB. Periodograms are calculated from only one of the two recording channels. We then use the Euclidean metric to classify the periodograms expressed in dB. We then represent the signals to be classified by their Toeplitz autocorrelation matrices $\mathbf{R}$. This matrix is estimated for each of the signals using only one of the two recording channels. We classify the Toeplitz autocorrelation matrices $\mathbf{R}$ in the Riemannian manifold $\mathbb{H}_n^{++}$ whose metric is described in Section 4.2.1. Then, we represent the audio signals using only one of the two recording channels in the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ whose metric is described in Section 4.1.3. After that, we represent two-dimensional audio signals by their block-Toeplitz covariance matrices $\mathbf{R}$. These matrices are then classified as elements of the manifold $\mathbb{H}_{n\times N}^{++}$ whose metric is described in Section 4.2.1. Finally, we represent two-dimensional audio signals in the $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$ manifold described in Section 4.2.3. Throughout this section, we have $N = 2$ and we choose $n = 4$. We will compare the classification performances obtained using the nearest centroid classifier algorithm in each of the spaces described above.

9.2.1 Periodograms

We start by representing the audio time series to be classified by their periodograms using only one of the two recording channels.

We recall that periodograms are estimators of the power spectral density S, as seen in Section 2.1.11. For a time series u of length N, the periodogram is defined by:

$$\widehat{S}_{per}(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} u(n)e^{-i2\pi nf}\right|^2. \tag{9.1}$$

In Figure 9.1, we present the spectra plotted from periodograms expressed in dB computed for each audio signal. We use 1001 points to plot each spectrum: we calculate the values of the function $\widehat{S}_{per}(f)$ for f ranging from -0.5 to 0.5 with a 0.001 step size. To plot the PCA and classify the periodograms expressed in dB, we use 1000 points: we eliminate one of the two extreme values ($f = -0.5$ or $f = 0.5$) because these two values correspond to the same frequency. Figure 9.2 represents the PCA of the periodograms. Figure 9.3 represents the PCA of the classification result using the nearest centroid classifier algorithm. Figure 9.4 represents the corresponding confusion matrix.

9.2.2 Unidimensional covariance matrices $\mathbb{H}_n^{++}$

In this section, we represent the audio signals by their autocorrelation matrices $\mathbf{R} \in \mathcal{H}_n^+$ estimated from only one of the two recording channels of the audio data. We then endow the space $\mathcal{H}_n^+$ with the information geometry metric presented in Section 4.2.1: we therefore classify the audio signals in the Riemannian manifold $\mathbb{H}_n^{++}$.

To estimate the autocorrelation matrices for each of the simulated signals, we start by using the regularized Burg algorithm 17 to estimate the mean quadratic power coefficient p_0 and the reflection coefficients $\mu_i = a_i^i$ for $1 \leqslant i \leqslant n-1$. We choose $n = 4$ here. Then, we use the inverse Levinson algorithm 15 to compute the coefficients $(\widehat{r}_0, \widehat{r}_1, \ldots, \widehat{r}_{n-1})$ from the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1}) \in \mathbb{R}_+^* \times \mathbb{D}^{n-1}$ previously estimated.

Figure 9.5 represents the Tangent PCA of the estimated autocorrelation matrices for each signal. Figure 9.6 represents the Tangent PCA of the classification result using the nearest centroid classifier algorithm. Figure 9.7 represents the corresponding confusion matrix.

9.2.3 Classification in $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$

Here we represent the audio signals by the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1}) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$ estimated from only one of the two audio recording channels. We endow the space $\mathbb{R}_+^* \times \mathcal{D}^{n-1}$ with the metric inspired by information geometry presented in Section 4.1.3. The audio dataset is therefore classified in the Riemannian manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$.

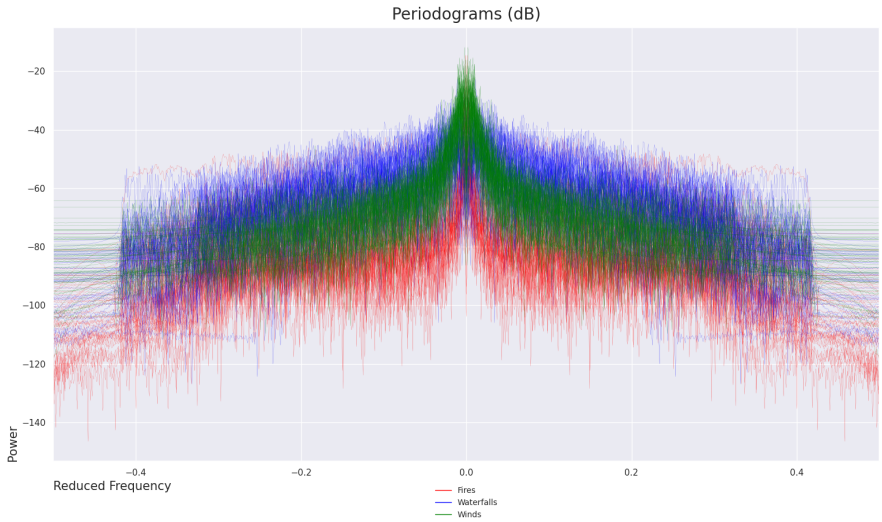

Figure 9.1: Periodograms (dB)

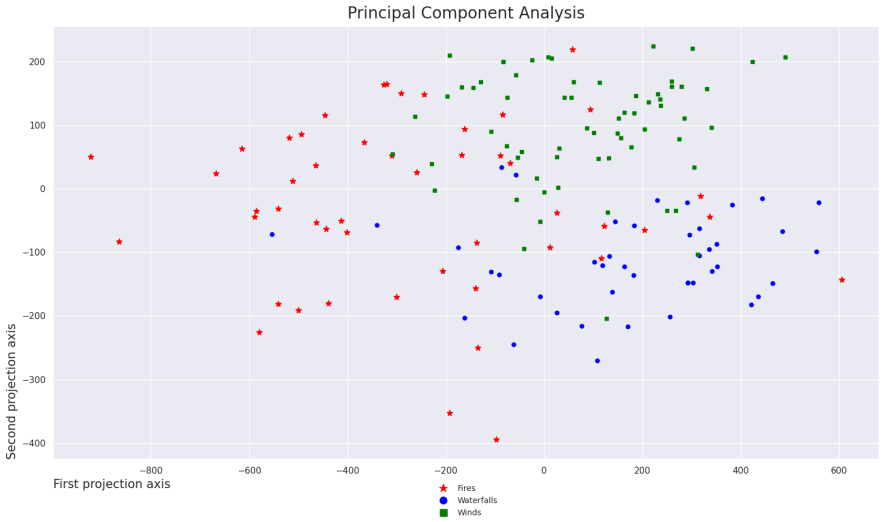

Figure 9.2: PCA on periodograms (dB)

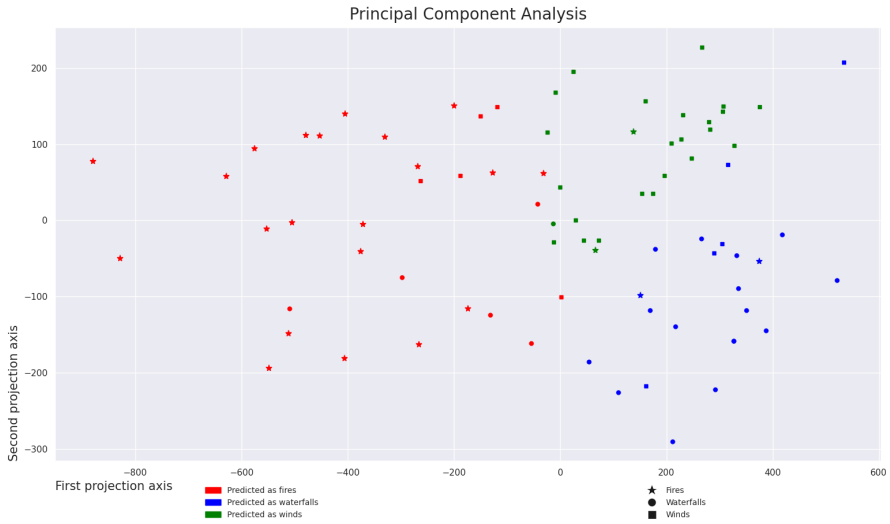

Figure 9.3: PCA of the classification result of the nearest centroid classifier on periodograms (dB)

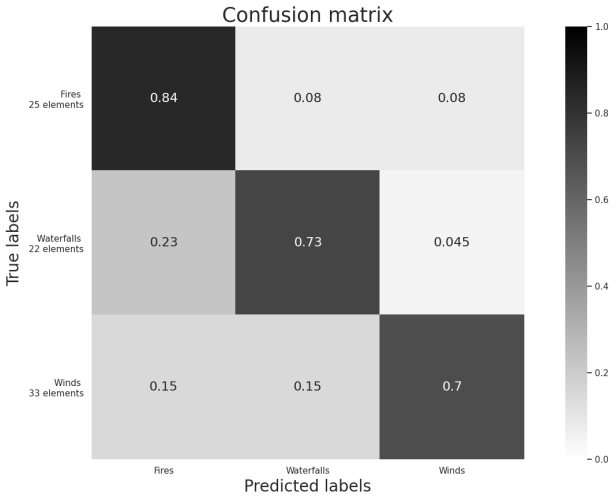

Figure 9.4: Confusion matrix of the classification result of the nearest centroid classifier on periodograms (dB)

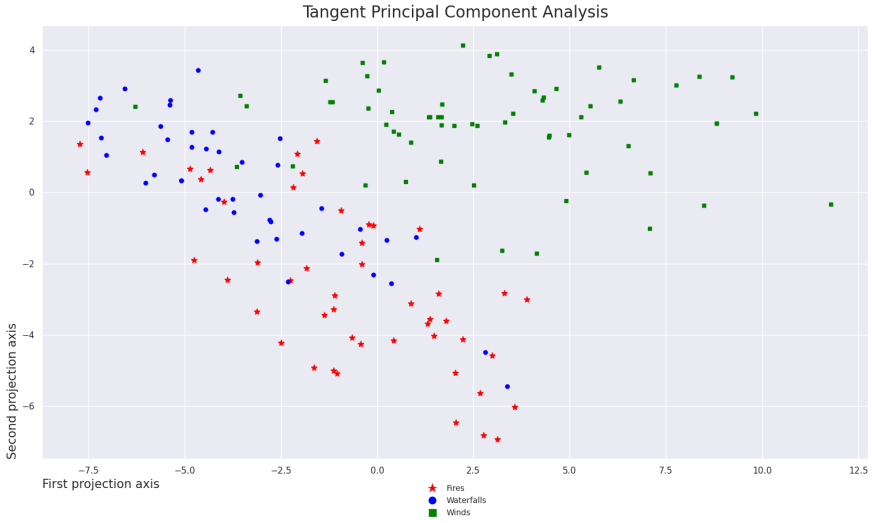

Figure 9.5: TPCA on unidimensional covariance matrices using the Information Geometry metric

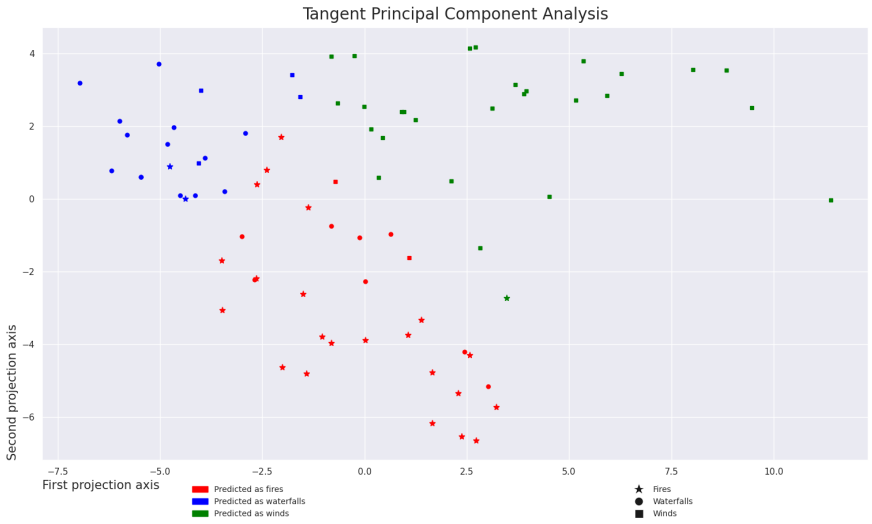

Figure 9.6: TPCA of the classification result of the nearest centroid classifier on unidimensional covariance matrices using the Information Geometry metric

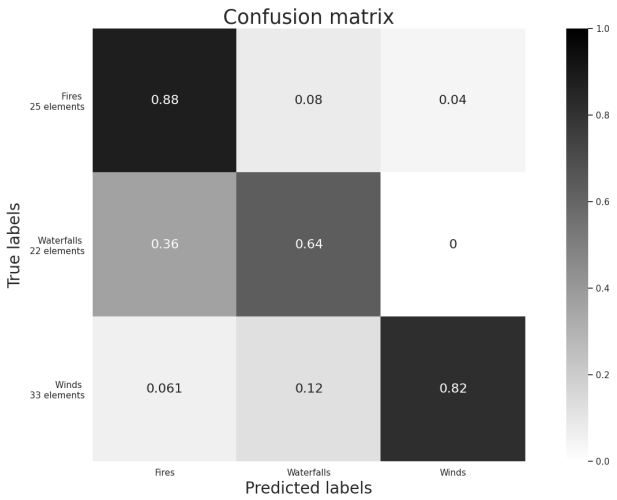

Figure 9.7: Confusion matrix of the classification result of the nearest centroid classifier on unidimensional covariance matrices using the Information Geometry metric

To estimate the coefficients $(p_0, \mu_1, \ldots, \mu_{n-1})$ of each of the simulated signals, we use the regularized Burg algorithm 17 with $n = 4$ here.

Figure 9.8 allows to visualize the coefficients $(\widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each of the signals in the space $\mathcal{D}^{n-1}$. We notice that the estimated coefficients are on the real axis, which is normal since the studied signals are real. Figure 9.9 represents the Tangent PCA of the coefficients $(\widehat{p}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_{n-1})$ estimated for each signal. Figure 9.10 represents the Tangent PCA of the classification result obtained using the nearest centroid classifier algorithm. Figure 9.11 represents the corresponding confusion matrix.

9.2.4 Multidimensional covariance matrices $\mathbb{H}^{++}_{n\times N}$

In this section, we represent each audio recording by its Block-Toeplitz autocorrelation matrix $\widehat{\mathbf{R}} \in \mathcal{B}^{+}_{n,N}$. The data is then classified using the Information geometry metric on the set of HPD matrices $\mathcal{H}^{+}_{n\times N}$ described in Section 4.2.1.

For each of the two-dimensional audio signals, we start by estimating their Block-Toeplitz autocorrelation matrices $\widehat{\mathbf{R}} \in \mathcal{B}^{+}_{n,N}$. To estimate these matrices, we use Algorithm 5 to estimate the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \ldots, \widehat{\Omega}_{n-1}\right) \in \mathcal{H}^{+}_{N} \times \mathcal{SD}^{n-1}_{N}$. Here we have $N = 2$ and we choose $n = 4$. Then, we use the relationships between the different representation spaces of multidimensional complex centered stationary autoregressive Gaussian processes presented in Section 2.2 to compute the autocorrelation matrices $\widehat{\mathbf{R}} \in \mathcal{B}^{+}_{n,N}$. The coefficients $\left(\widehat{R}_0, \widehat{R}_1, \ldots, \widehat{R}_{n-1}\right)$ of matrix $\widehat{\mathbf{R}}$ are indeed computed from the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \ldots, \widehat{\Omega}_{n-1}\right) \in \mathcal{H}^{+}_{N} \times \mathcal{SD}^{n-1}_{N}$ thanks to Algorithm 4.

We now present the classification of two-dimensional audio signals represented by their estimated autocorrelation matrices $\widehat{\mathbf{R}}$ in the manifold $\mathbb{H}^{++}_{n\times N}$. Figure 9.12 represents the TPCA of the audio dataset represented in the manifold $\mathbb{H}^{++}_{n\times N}$. Figure 9.13 represents the TPCA of the classification result obtained using the nearest centroid classifier algorithm. Finally, Figure 9.14 represents the corresponding confusion matrix.

9.2.5 Classification in $\mathbb{H}^{++}_{N} \times \mathbb{SD}^{n-1}_{N}$

In this section, we classify two-dimensional audio signals using the Riemannian manifold $\mathbb{H}^{++}_{N} \times \mathbb{SD}^{n-1}_{N}$ described in Section 4.2.3.

For each audio signal, we use Algorithm 5 to compute the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \ldots, \widehat{\Omega}_{n-1}\right) \in \mathcal{H}^{+}_{N} \times \mathcal{SD}^{n-1}_{N}$.

We now present the classification of audio signals represented by the coefficients $\left(\widehat{P}_0, \widehat{\Omega}_1, \ldots, \widehat{\Omega}_{n-1}\right)$ in the Riemannian manifold $\mathbb{H}^{++}_{N} \times \mathbb{SD}^{n-1}_{N}$. Figure 9.15 represents the TPCA of the audio dataset represented in the manifold $\mathbb{H}^{++}_{N} \times \mathbb{SD}^{n-1}_{N}$. Figure 9.16 represents the TPCA of the classification result obtained using the nearest centroid classifier algorithm. Finally, Figure 9.17 represents the corresponding confusion matrix.

We now compare the classification performances obtained thanks to the nearest centroid classifier algorithm on the different representation spaces of the audio signals. To this end, we compare the confusion matrices presented in Figures 9.4, 9.7, 9.11, 9.14 and 9.17 :we use the average of the diagonal values as an indicator of the performance. The average

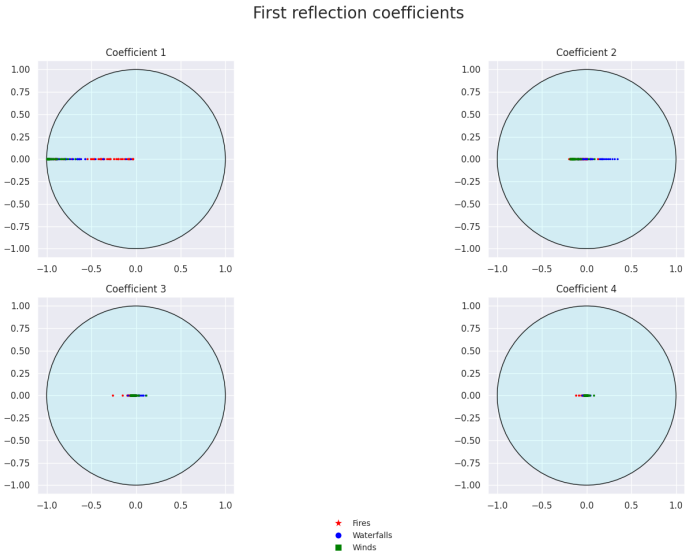

Figure 9.8: First reflection coefficients in Poincaré disks

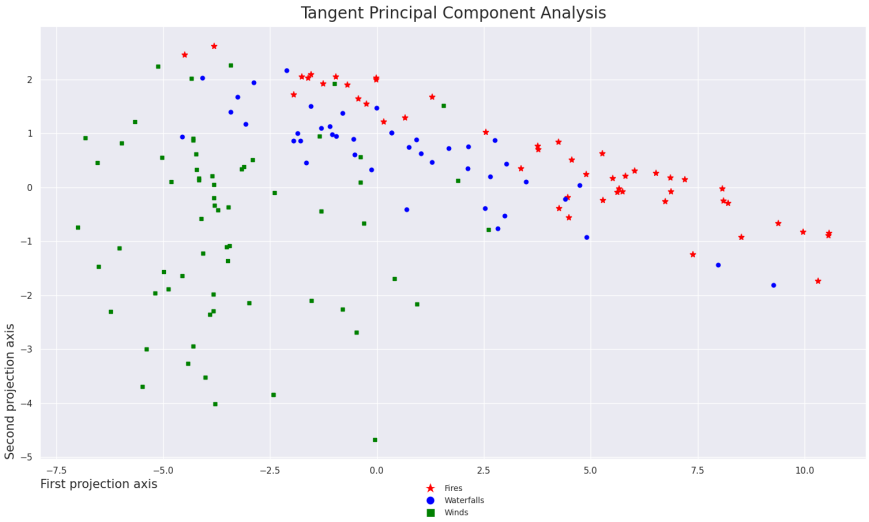

Figure 9.9: TPCA on $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$

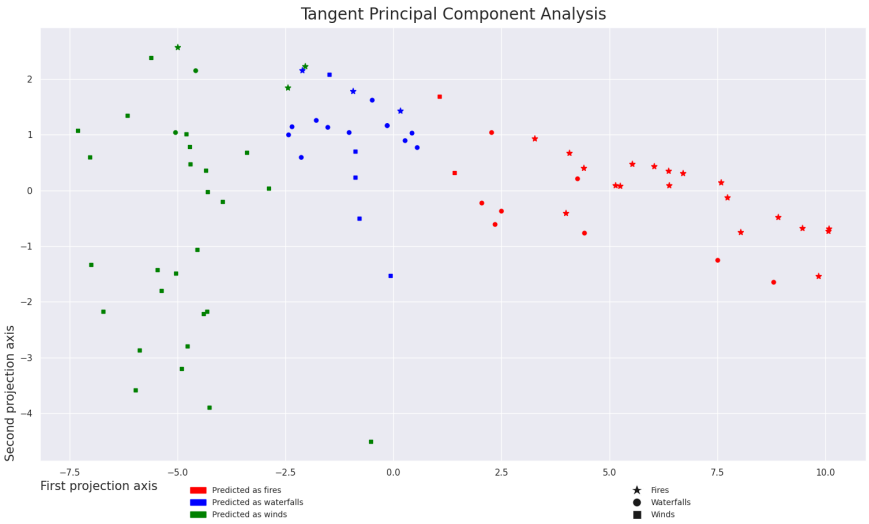

Figure 9.10: TPCA of the classification result of the nearest centroid classifier on $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$

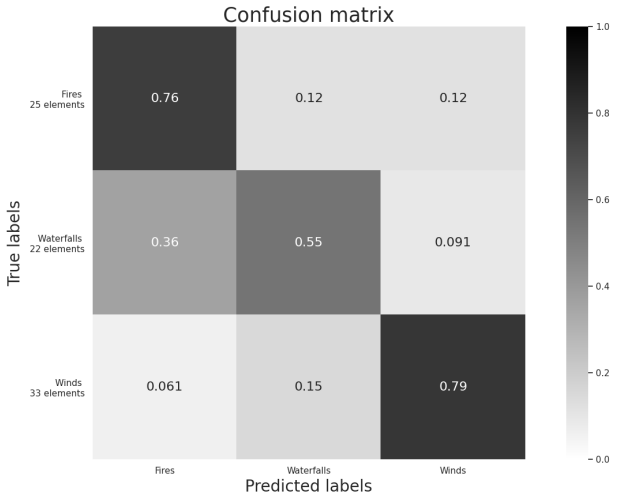

Figure 9.11: Confusion matrix of the classification result of the nearest centroid classifier on $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$

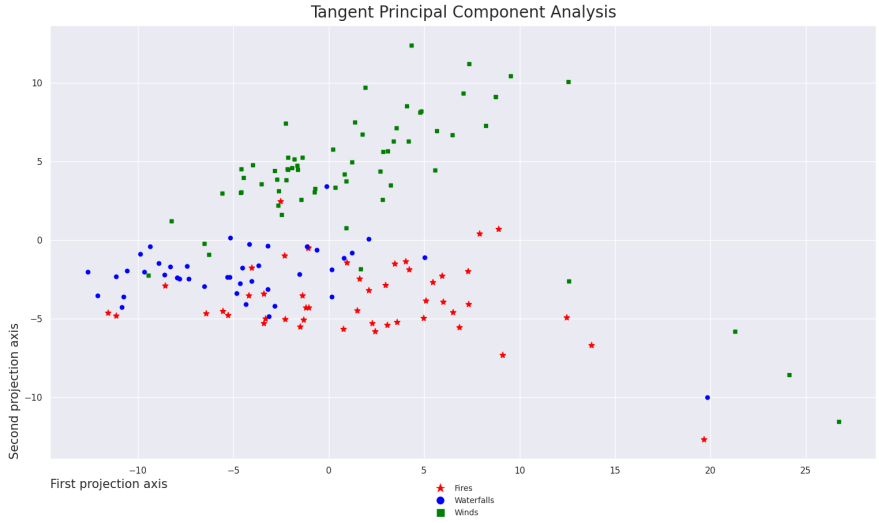

Figure 9.12: TPCA on multidimensional covariance matrices using the Information Geometry metric

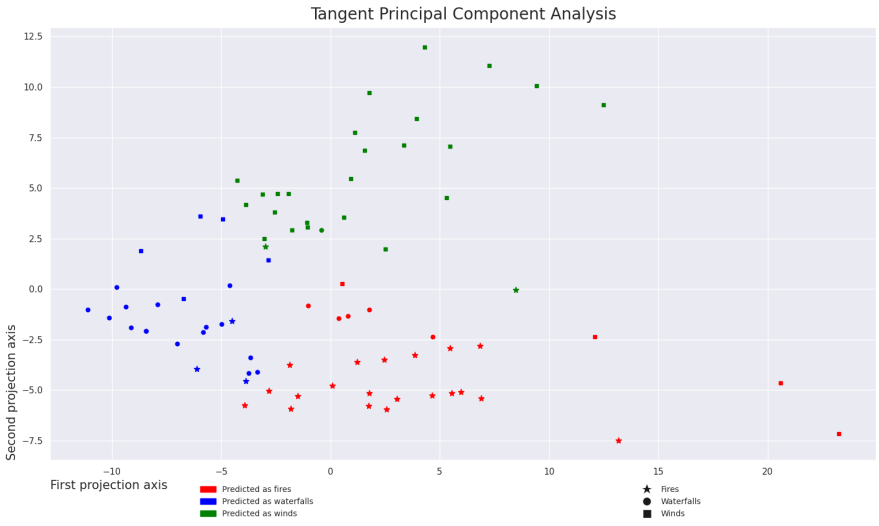

Figure 9.13: TPCA of the classification result of the nearest centroid classifier on multidimensional covariance matrices using the Information Geometry metric

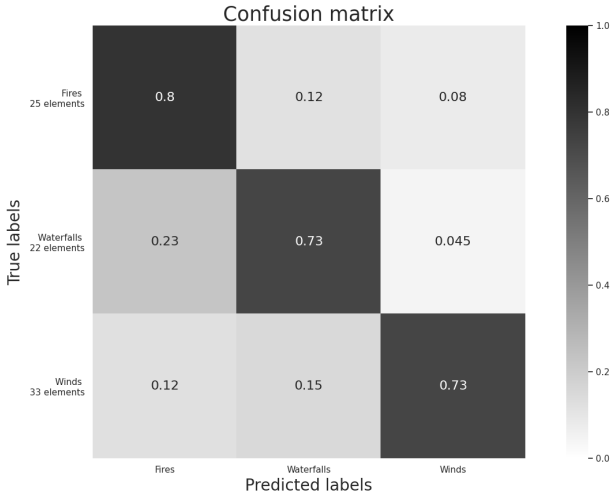

Figure 9.14: Confusion matrix of the classification result of the nearest centroid classifier on multidimensional covariance matrices using the Information Geometry metric

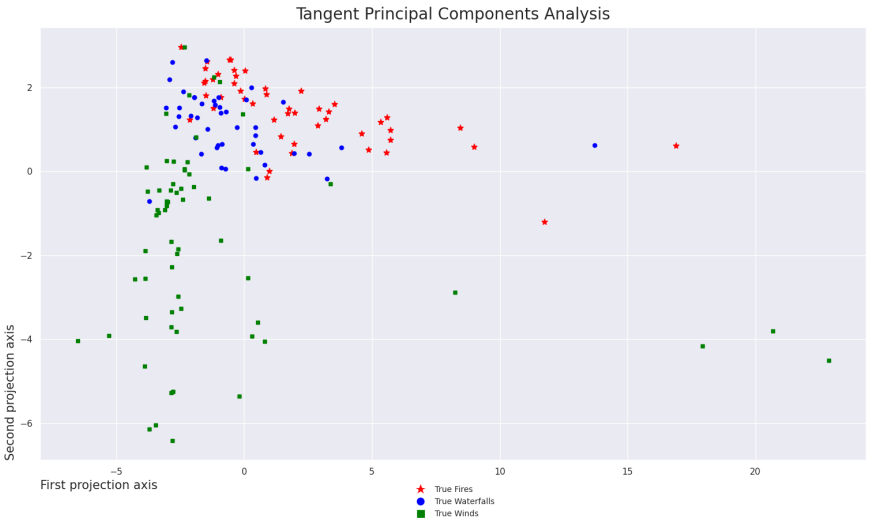

Figure 9.15: TPCA on $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

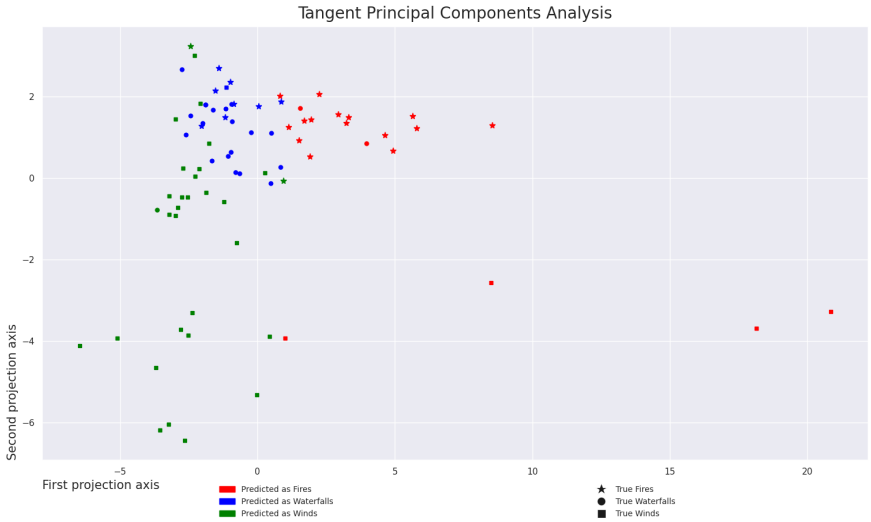

Figure 9.16: TPCA of the classification result of the nearest centroid classifier on $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

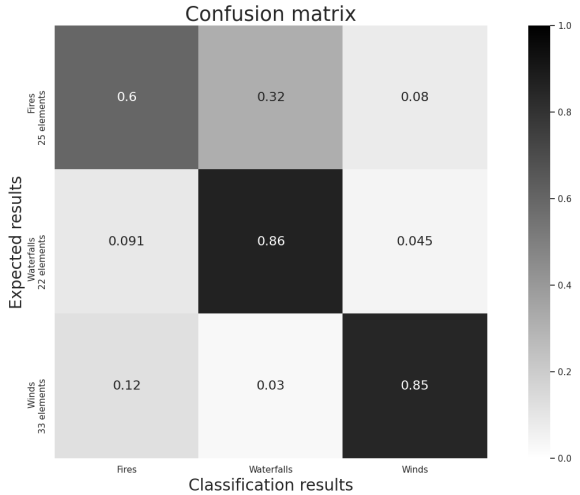

Figure 9.17: Confusion matrix of the classification result of the nearest centroid classifier on $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$

of the diagonal values of the periodogram confusion matrix is 0.76. This mean value is 0.78 for the classification on the manifold $\mathbb{H}_n^{++}$, of 0.7 for the classification on the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$, of 0.75 for the classification on the manifold $\mathbb{H}_{n\times N}^{++}$ and of 0.77 on the manifold $\mathbb{H}_N^{++} \times \mathbb{SD}_N^{n-1}$. In the case of the classification of two-dimensional audio data presented here, multidimensional methods seem to have the same level of performance as unidimensional methods. This result may have the following interpretation: the correlation between the two recording channels might not provide useful information to determine the nature of the sounds (fires, waterfalls or winds).

Appendices

Appendix A

The Levinson algorithm

In Section 2.1, we studied complex stationary centered autoregressive Gaussian processes. We recall the definition of an AR process of order M given in Section 2.1.4:

$$u(k) + \sum_{i=1}^{n-1} a_i^{n-1}\, u(k-i) = w^{n-1}(k) \tag{A.1}$$

where a_i^{n-1} are the prediction coefficients and the process $w^{n-1}(k)$ is a white noise.

In this appendix, we would like to compute efficiently the autoregressive coefficients $\left(a_i^N\right)_{i=1,\dots,N}$, assuming that we know the theoretical autocorrelation coefficients $(r_i)_{i=0,\dots,N}$.

Contents

A.1 The autoregressive and correlation coefficients relation

We recall that due to the hypothesis of stationarity, the autocorrelation matrix of the process is Hermitian (Property 1) and Toeplitz (Property 2) as seen in Section 2.1.3.

By applying the autocorrelation operator $\mathbb{E}[\,\cdot\, u(l)^*]$ for $l \in [\![0, k-1]\!]$ on the elements of the equation of the autoregressive model (A.1), we obtain the following system on the autocorrelation coefficients:

$$\begin{array}{rcl} R_N\, \widetilde{a}_N & = & -\widetilde{r}_N \\ \widetilde{a}_N & = & [a_1^N, \dots, a_N^N]^T \\ \widetilde{r}_N & = & [r_1, \dots, r_N]^T \end{array} \tag{A.2}$$

where R_N corresponds to the autocorrelation matrix $\mathbf{R}$ of the process. Equation (A.2) is known as the Yule-Walker equation and has been studied in Section 2.1.5.

The solution of this problem is then obtained inversing the autocorrelation matrix R_N:

$$\widetilde{a}_N = -R_N^{-1}\, \widetilde{r}_N. \tag{A.3}$$

A.2 The Levinson algorithm

If we want to compute the auregressive coefficients $\widetilde{a}_N$ for N equal 1 to $n-1$, we can invert the matrix R_N for N equal 1 to $n-1$ as in Equation (A.3).

However, we note that the matrix R_N and the vector $\widetilde{r}_N$ have specific structures allowing a recursive computation of the coefficients $\widetilde{a}_N$ for a lower computational cost. It is the principle of the Levinson algorithm [45] detailed in Algorithm 13.

Property 25. *The coefficients a_i^j computed by the Levinson algorithm from the autocorrelation coefficients r_k are the coefficients of the autoregressive model.*

Algorithm 13 The Levinson algorithm

Initialization:

$$\widetilde{a}_1 = a_1^1 = -\frac{r_1}{r_0} \tag{A.4}$$

for $l = 2, ..., n-1$: **do**

$$a_l^l = -\frac{r_l + \sum_{j=1}^{l-1} r_{l-j} a_j^{l-1}}{r_0 + \sum_{j=1}^{l-1} r_j a_j^{l-1}\,*}, \tag{A.5}$$

$$\widetilde{a}_l = \begin{bmatrix} a_1^l \\ \vdots \\ a_{l-1}^l \\ a_l^l \end{bmatrix} = \begin{bmatrix} a_1^{l-1} \\ \vdots \\ a_{l-1}^{l-1} \\ 0 \end{bmatrix} + a_l^l \begin{bmatrix} a_{l-1}^{l-1}\,* \\ \vdots \\ a_1^{l-1}\,* \\ 1 \end{bmatrix} \tag{A.6}$$

end for

Proof. We would like to compute $\widetilde{a}_N$ of Equation (A.3) using the following properties:

$$R_{N-1}\,\widetilde{a}_{N-1} = -\widetilde{r}_{N-1} \tag{A.7}$$

$$R_N = \left[\begin{array}{cccc|c} r_0 & r_1^* & r_2^* & \cdots & r_{N-1}^* \\ r_1 & r_0 & r_1^* & \cdots & r_{N-2}^* \\ r_2 & r_1 & r_0 & \cdots & r_{N-3}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \hline r_{N-1} & r_{N-2} & r_{N-3} & \cdots & r_0 \end{array}\right] = \left[\begin{array}{c|c} R_{N-1} & \widetilde{r}_{N-1}^{B*} \\ \hline \widetilde{r}_{N-1}^{BT} & r_0 \end{array}\right] \tag{A.8}$$

where we denote by $\cdot^T$ the matrix transposition, $\cdot^*$ the complex conjugate and $\cdot^B$ the backward operator that reverse a vector: $\widetilde{r}_{N-1}^B = [r_{N-1}, \ldots, r_1]^T$.

$$\widetilde{r}_N = \left[\begin{array}{c} r_1 \\ \vdots \\ \hline r_{N-1} \\ \hline r_N \end{array}\right] = \left[\begin{array}{c} \widetilde{r}_{N-1} \\ \hline r_N \end{array}\right] = \left[\begin{array}{c} -R_{N-1}\,\widetilde{a}_{N-1} \\ \hline r_N \end{array}\right] \tag{A.9}$$

using Equation (A.7).

$$\widetilde{a}_N = -R_N^{-1}\,\widetilde{r}_N \tag{A.10}$$

$$\widetilde{a}_N = -\left[\begin{array}{c|c} R_{N-1} & \widetilde{r}_{N-1}^{B*} \\ \hline \widetilde{r}_{N-1}^{BT} & r_0 \end{array}\right]^{-1} \left[\begin{array}{c} -R_{N-1}\,\widetilde{a}_{N-1} \\ \hline r_N \end{array}\right] \tag{A.11}$$

We inverse the matrix R_N using the formula given in [80] for inversing Toeplitz matrices:

$$\widetilde{a}_N = -\left[\begin{array}{c|c} R_{N-1}^{-1} + W_{N-1}\alpha_{N-1}^{-1}V_{N-1}^T & W_{N-1}\,\alpha_{N-1}^{-1} \\ \hline \alpha_{N-1}^{-1}\,V_{N-1}^T & \alpha_{N-1}^{-1} \end{array}\right] \left[\begin{array}{c} -R_{N-1}\,\widetilde{a}_{N-1} \\ \hline r_N \end{array}\right] \tag{A.12}$$

where :

$$\alpha_{N-1} = r_0 - \tilde{r}_{N-1}^{BT}\, R_{N-1}^{-1}\, \tilde{r}_{N-1}^{B*} = r_0 + \tilde{r}_{N-1}^{BT}\, \tilde{a}_{N-1}^{B*}$$
$$V_{N-1}^{T} = -\tilde{r}_{N-1}^{BT}\, R_{N-1}^{-1}$$
$$W_{N-1} = -R_{N-1}^{-1}\, \tilde{r}_{N-1}^{B*} = \tilde{a}_{N-1}^{B*}$$

We prove that $R_{N-1}\, \tilde{a}_{N-1}^{B*} = -\tilde{r}_{N-1}^{B*}$ applying $\mathbb{E}[\, x(k)\, (\cdot)^*]$ on each side of Equation (A.1) for different values of k.

$$\tilde{a}_N = -\begin{bmatrix} -\tilde{a}_{N-1} \\ -W_{N-1}\, \alpha_{N-1}^{-1}\, V_{N-1}^{T}\, R_{N-1}\, \tilde{a}_{N-1} \\ +W_{N-1}\, \alpha_{N-1}^{-1}\, r_N \\ \hline -\alpha_{N-1}^{-1}\, V_{N-1}^{T}\, R_{N-1}\, \tilde{a}_{N-1} + \alpha_{N-1}^{-1}\, r_N \end{bmatrix} \tag{A.13}$$

$$\tilde{a}_N = \begin{bmatrix} \tilde{a}_{N-1} + \tilde{a}_{N-1}^{B*}\, a_N^N \\ \hline a_N^N \end{bmatrix} \tag{A.14}$$

where :

$$a_N^N = \alpha_{N-1}^{-1}\left(r_N + \tilde{r}_{N-1}^{BT}\, \tilde{a}_{N-1}\right)$$

□

We now would like to simplifie the computation of the denominator in Equation (A.5). We denote P_{l-1} this denominator. Hence:

$$P_l = r_0 + \sum_{j=1}^{l} r_j\, a_j^{l*} \tag{A.15}$$

Property 26. *We denote $\mu_l = a_l^l$ the reflection coefficients. The coefficients P_l of successive orders are linked by the following equation:*

$$P_{l+1} = P_l\left(1 - |\mu_{l+1}|^2\right) \quad \forall l \in \mathbb{N} \tag{A.16}$$

We deduce by an immediate recurrence that:

$$P_n = \left(\prod_{k=1}^{n}\left(1 - |\mu_k|^2\right)\right) r_0 \tag{A.17}$$

Proof. Let $l \in \mathbb{N}$.

$$P_{l+1} = r_0 + \sum_{j=1}^{l+1} r_j\, a_j^{l+1*} \tag{A.18}$$
$$= r_0 + \sum_{j=1}^{l} r_j\, a_j^{l+1*} + r_{l+1}\, \mu_{l+1}^* \tag{A.19}$$
$$= r_0 + \sum_{j=1}^{l} r_j\, \left(a_j^l + \mu_{l+1}\, a_{l+1-j}^{l*}\right)^* + r_{l+1}\, \mu_{l+1}^* \tag{A.20}$$
$$= r_0 + \sum_{j=1}^{l} r_j\, a_j^{l*} + \mu_{l+1}^*\left(\sum_{j=1}^{l} r_j\, a_{l+1-j}^{l} + r_{l+1}\right) \tag{A.21}$$
$$= P_l + P_l\, \mu_{l+1}^*\, \frac{\sum_{j=1}^{l} r_j\, a_{l+1-j}^{l} + r_{l+1}}{P_l} \tag{A.22}$$
$$= P_l + P_l\, \mu_{l+1}^*\, (-\mu_{l+1}) \tag{A.23}$$
$$= P_l\left(1 - |\mu_{l+1}|^2\right) \tag{A.24}$$

□

Another proof of Property 26 is given at the very end of the thesis of Le Yang [82].

Using Property 26, we can compute faster the denominator appearing in the computation of the reflection coefficient a_l^l in Equation (A.5) of the Levinson algorithm 13. We give a fast version of the Levinson algorithm in Algorithm 14.

Algorithm 14 The Levinson algorithm (fast version)

Initialization:

$$P_0 = r_0 \tag{A.25}$$

for $l = 1, ..., M$: **do**

$$a_l^l = -\frac{r_l + \sum_{j=1}^{l-1} r_{l-j} a_j^{l-1}}{P_{l-1}}, \tag{A.26}$$

$$\widetilde{a}_l = \begin{bmatrix} a_1^l \\ \vdots \\ a_{l-1}^l \\ a_l^l \end{bmatrix} = \begin{bmatrix} a_1^{l-1} \\ \vdots \\ a_{l-1}^{l-1} \\ 0 \end{bmatrix} + a_l^l \begin{bmatrix} a_{l-1}^{l-1\,*} \\ \vdots \\ a_1^{l-1\,*} \\ 1 \end{bmatrix} \tag{A.27}$$

$$P_l = P_{l-1}\left(1 - |\mu_l|^2\right) \tag{A.28}$$

end for

If we want to compute the autocorrelation matrix R_N of the autocorrelation coefficients $(r_i)_{i=0,...,N}$ from the mean quadratic power coefficient P_0 and the reflection coefficients $\left(a_i^i\right)_{i=1,...,N}$, we can inverse the Levinson algorithm. This is done by Algorithm 15.

Algorithm 15 The reversed Levinson algorithm (fast version)

Initialization:

$$r_0 = P_0 \tag{A.29}$$

for $l = 1, ..., M$: **do**

$$r_l = -a_l^l P_{l-1} - \sum_{j=1}^{l-1} r_{l-j} a_j^{l-1}, \tag{A.30}$$

$$\widetilde{a}_l = \begin{bmatrix} a_1^l \\ \vdots \\ a_{l-1}^l \\ a_l^l \end{bmatrix} = \begin{bmatrix} a_1^{l-1} \\ \vdots \\ a_{l-1}^{l-1} \\ 0 \end{bmatrix} + a_l^l \begin{bmatrix} a_{l-1}^{l-1\,*} \\ \vdots \\ a_1^{l-1\,*} \\ 1 \end{bmatrix} \tag{A.31}$$

$$P_l = P_{l-1}\left(1 - |\mu_l|^2\right) \tag{A.32}$$

end for

Property 27. *The coefficients a_l^l lie in the complex unit disk $\mathbb{D}$: $|a_l^l| < 1 \quad \forall l = 1, ..., n-1$.*

Proof. In the PhD thesis of Alice Le Brigant [15], it is shown that $P_l = E\left[|\, w^l \,|^2\right]$. Hence $P_l \in \mathbb{R}_+^*$. Then, we can use Property 26 to prove that $|a_l^l| < 1$. Indeed, the coefficient $P_0 = r_0$ is positive. If we assume that there exist an index l for which $a_l^l \geqslant 1$, then if we denote m the smallest index for which $a_l^l \geqslant 1$, we have $P_m = P_{m-1}\left(1 - |\mu_m|^2\right) \leqslant 0$ which is impossible. □

Property 28. *The modulus of the autoregressive coefficients is bounded the following way:*

$$|a_k^n| \leqslant \binom{n}{k} \quad \forall\, (k, n) \in \mathbb{N}^2, \tag{A.33}$$

with by convention $\binom{n}{0} = 1$ *for all* $n \in \mathbb{N}$ *and* $\binom{n}{k} = 0$ *when* $k > n$.

Proof. We prove Property 28 by recurrence on the indice n.

Initialization: The property is true for $n = 0$ as $a_0^0 = \binom{0}{0} = 1$.

Heredity: We assume that there exists an integer $n \in \mathbb{N}$ such that for all $k \in [\![0, n]\!]$, we have:

$$|a_k^n| \leqslant \binom{n}{k}. \tag{A.34}$$

We now prove that:

$$\left|a_k^{n+1}\right| \leqslant \binom{n+1}{k} \quad \forall k \in [\![0, n+1]\!]. \tag{A.35}$$

If $k = n + 1$, we have:

$$\left|a_{n+1}^{n+1}\right| = |\mu_{n+1}| < 1 = \binom{n+1}{n+1}. \tag{A.36}$$

If $k \in [\![0, n]\!]$, we recall the recurrence relation between the autoregressive coefficients which is given in vector form in Equation (A.6):

$$a_k^j = a_k^{j-1} + \mu_j a_{j-k}^{j-1*} \quad \forall\, (i, j) \in \mathbb{N}^2 \tag{A.37}$$

Therefore, we have:

$$\left|a_k^{n+1}\right| \leqslant |a_k^n| + |\mu_{n+1}| \left|a_{n+1-k}^{n*}\right| \quad \text{according to the triangular inequality} \tag{A.38}$$

$$\Rightarrow \left|a_k^{n+1}\right| \leqslant |a_k^n| + \left|a_{n+1-k}^{n*}\right| \quad \text{since } |\mu_{n+1}| < 1 \tag{A.39}$$

$$\Rightarrow \left|a_k^{n+1}\right| \leqslant \binom{n}{k} + \binom{n}{n+1-k} \quad \text{according to the recurrence hypothesis} \tag{A.40}$$

$$\Leftrightarrow \left|a_k^{n+1}\right| \leqslant \binom{n}{k} + \binom{n}{n-(k-1)} \tag{A.41}$$

$$\Leftrightarrow \left|a_k^{n+1}\right| \leqslant \binom{n}{k} + \binom{n}{k-1} \quad \text{since } \binom{n}{n-k} = \binom{n}{k} \quad \forall\, (n, k) \in \mathbb{N}^2 \tag{A.42}$$

$$\Leftrightarrow \left|a_k^{n+1}\right| \leqslant \binom{n+1}{k} \quad \text{since } \binom{n}{k} + \binom{n}{k+1} = \binom{n+1}{k+1} \quad \forall\, (n, k) \in \mathbb{N}^2 \tag{A.43}$$

Conclusion: We proved that:

$$|a_k^n| \leqslant \binom{n}{k} \quad \forall\, (k, n) \in \mathbb{N}^2. \tag{A.44}$$

□

A.3 Equivalent set of parameters

In this section, we highlight the relationships between different parameters of the autoregressive model.

In Equation (A.6), we give in vector form the following relation which links the AR coefficients of successive order:

$$a_k^n = a_k^{n-1} + \mu_n a_{n-k}^{n-1*} \quad \forall k \in [\![1, n]\!]. \tag{A.45}$$

Using Equation (A.45), we can compute the autoregressive coefficients autorégressifs a_i^j pour tout $1 \leqslant i \leqslant j \leqslant n$ from the reflection coefficients $\mu_1, \mu_2, \ldots, \mu_n$, where $\mu_i = a_i^i$. The coefficients a_i^j are then computed by successive orders: the coefficients of order j are used to compute the coefficients of order $j + 1$. Therefore the reflection coefficients $\mu_1, \mu_2, \ldots, \mu_n$ entirely determine the autoregressive coefficients a_i^j for all $1 \leqslant i \leqslant j \leqslant n$.

We now show that the autoregressive coefficients $\left(a_i^j\right)_{1 \leqslant i \leqslant j \leqslant n}$ are also entirely determined by the coefficients $a_1^n, a_2^n, \ldots, a_n^n$ of the autoregressive model of order n.

Property 29. *Equation* (A.45) *is equivalent to the following equation:*

$$a_k^{n-1} = \frac{a_k^n - \mu_n a_{n-k}^{n*}}{1 - |\mu_n|^2}. \tag{A.46}$$

This last equation can be used to compute the coefficients of the autoregressive model of order $n-1$ from the coefficients of the autoregressive model of order n.

Proof. We use an equivalence proof starting from Equation (A.45).

$$
\begin{aligned}
& a_k^n = a_k^{n-1} + \mu_n a_{n-k}^{n-1*} \quad \forall k \in [\![1, n-1]\!] \\
\Leftrightarrow & \left\{ \begin{array}{rcll} a_k^n & = & a_k^{n-1} + \mu_n a_{n-k}^{n-1*} & \\ a_{n-k}^n & = & a_{n-k}^{n-1} + \mu_n a_k^{n-1*} & \end{array} \right. \\
\Leftrightarrow & \left\{ \begin{array}{rcll} a_k^n & = & a_k^{n-1} + \mu_n a_{n-k}^{n-1*} & \\ a_{n-k}^{n*} & = & \mu_n^* a_k^{n-1} + a_{n-k}^{n-1*} & L_2 \leftarrow L_2^* \end{array} \right. \\
\Leftrightarrow & \left\{ \begin{array}{rcll} a_k^n - \mu_n a_{n-k}^{n*} & = & \left(1 - |\mu_n|^2\right) a_k^{n-1} & L_1 \leftarrow L_1 - \mu_n L_2 \\ a_{n-k}^{n*} - \mu_n^* a_k^n & = & \left(1 - |\mu_n|^2\right) a_{n-k}^{n-1*} & L_2 \leftarrow L_2 - \mu_n^* L_1 \end{array} \right. \\
\Leftrightarrow & \left\{ \begin{array}{rcll} a_k^{n-1} & = & \frac{a_k^n - \mu_n a_{n-k}^{n*}}{1-|\mu_n|^2} & L_1 \leftarrow L_1 / \left(1 - |\mu_n|^2\right) \\ a_{n-k}^{n-1} & = & \frac{a_{n-k}^n - \mu_n a_k^{n*}}{1-|\mu_n|^2} & L_2 \leftarrow L_2^* / \left(1 - |\mu_n|^2\right) \end{array} \right. \\
\Leftrightarrow & \quad a_k^{n-1} = \frac{a_k^n - \mu_n a_{n-k}^{n*}}{1 - |\mu_n|^2} & \quad \forall k \in [\![1, n-1]\!] &
\end{aligned}
$$

□

Finally, we presented two algorithms that can be used to establish the following equivalences:

$$
(\mu_1, \mu_2, \ldots, \mu_n) \Leftrightarrow \left(a_i^j\right)_{1 \leqslant i \leqslant j \leqslant n} \Leftrightarrow (a_1^n, a_2^n, \ldots, a_n^n) \,. \tag{A.47}
$$

Other equivalences between parameter spaces involving autoregressive coefficients are presented in Sections 2.1.5 and 2.1.6.

Appendix B

Theoretical reflection coefficients of continuous stochastic processes with Gaussian spectrum shape

In this appendix, we recall the relations between the spectrum of a continuous stochastic process, its autocorrelation coefficients and its reflection coefficients in Section B.1. We then use the several properties given in Section B.1 to compute the theoretical reflection coefficients of continuous stochastic processes with Gaussian spectrum shape in Section B.2.

Contents

B.1 Spectra, autocorrelation coefficients and reflection coefficients

Let S_f be an integrable positive real function representing a spectrum shape. We denote r_k (respectively $\widetilde{r}_k$) the autocorrelation coefficients corresponding to the spectrum S_f (respectively $\widetilde{S}_f$) and a_k^n (respectively $\widetilde{a}_k^n$) its autoregressive coefficients.

The autocorrelation function R of a continuous complex valued stochastic process f is defined by:

$$R_f(\tau) = \int_{-\infty}^{+\infty} f(t+\tau) f^*(t) dt. \tag{B.1}$$

It has a Fourier transform called the power spectral density:

$$S_f(\xi) = \int_{-\infty}^{+\infty} R_f(\tau) e^{-i2\pi\xi\tau} d\tau. \tag{B.2}$$

$R_f(\tau)$ is therefore the inverse Fourier transform of $S_f(\xi)$:

$$R_f(\tau) = \int_{-\infty}^{+\infty} S_f(\xi) e^{+i2\pi\xi\tau} d\xi. \tag{B.3}$$

B.1.1 Useful properties about the continuous Fourier transform and usual Fourier transforms

We recall some useful properties about the continous Fourier transform and we give the Fourier transforms of three usual functions in the following tabular.

Function	Fourier transform	Remarks
$f(x)$	$\mathcal{F}(f)(\xi) = \widehat{f}(\xi) = \int_{-\infty}^{+\infty} f(x)e^{-i2\pi x\xi}dx$	Definition
$a\,f(x) + b\,g(x)$	$a\,\widehat{f}(\xi) + b\,\widehat{g}(\xi)$	Linearity
$f(x-m)$	$e^{-i2\pi m\xi}\,\widehat{f}(\xi)$	Shift in time domain
$f(ax)$	$\frac{1}{\lvert a\rvert}\widehat{f}(\frac{\xi}{a})$	Scaling in the time domain
$\widehat{f}(x)$	$f(-\xi)$	Duality
$\mathrm{tri}(ax)$	$\frac{1}{\lvert a\rvert}\mathrm{sinc}^2(\frac{\xi}{a})$	$\mathrm{tri}(x) = \max(1-\lvert x\rvert, 0)$
$e^{-a\lvert x\rvert}$	$\frac{2a}{a^2+4\pi^2\xi^2}$	For $\mathrm{Re}(a) > 0$.
e^{-ax^2}	$\sqrt{\frac{\pi}{a}}e^{-\frac{(\pi\xi)^2}{a}}$	For $\mathrm{Re}(a) > 0$.

B.1.2 Spectrum multiplied by a scalar constant

We first study the case where S_f is multiplied by a complex scalar constant: we denote $\widetilde{S}_f = \lambda S_f$, $\lambda \in \mathbb{C}$.

Property 30. *If we denote $\widetilde{S}_f = \lambda S_f$, the autocorrelation coefficients of the corresponding spectra $\widetilde{S}_f$ and S_f are linked by the equation:*

$$\widetilde{r}_k = \lambda r_k \tag{B.4}$$

Proof. We denote $\widetilde{R}_f$ the autocorrelation function associated with the power spectral density $\widetilde{S}_f$. Using these notations, we have:

$$\begin{aligned}
\widetilde{R}_f(\tau) &= \mathcal{F}\left(\widetilde{S}_f\right)(-\tau) && \text{(B.5)}\\
&= \mathcal{F}\left(\lambda S_f\right)(-\tau) && \text{(B.6)}\\
&= \mathcal{F}\left(x \mapsto \lambda S_f(x)\right)(-\tau) && \text{(B.7)}\\
&= \lambda\mathcal{F}\left(S_f\right)(-\tau) && \text{(B.8)}\\
&= \lambda R_f(\tau) && \text{(B.9)}
\end{aligned}$$

Therefore, we have:

$$\widetilde{r}_k = \lambda r_k \tag{B.10}$$

□

Property 31. *If we denote $\widetilde{S}_f = \lambda S_f$, the autoregressive coefficients of the corresponding spectra $\widetilde{S}_f$ and S_f are equal:*

$$\widetilde{a}_k^n = a_k^n \tag{B.11}$$

Proof. Initialization: We have:

$$\widetilde{\mu}_1 = \widetilde{a}_1^1 = -\frac{\widetilde{r}_1}{\widetilde{r}_0} = -\frac{\lambda r_1}{\lambda r_0} = -\frac{r_1}{r_0} = \mu_1 \tag{B.12}$$

The property is initialized for $n = 1$.

Heredity: We assume that there existe $n \in \mathbb{N}$ such that $\forall k \in [\![1,\ n]\!]$:

$$\widetilde{a}_k^n = a_k^n \tag{B.13}$$

We now show that the property is true at the rank $n+1$.

- If $k = n+1$, the Levinson algorithm gives us:

$$\widetilde{\mu}_{n+1} = \widetilde{a}_{n+1}^{n+1} \tag{B.14}$$

$$= -\frac{\widetilde{r}_{n+1} + \sum_{j=1}^{n} \widetilde{r}_{n+1-j}\widetilde{a}_j^n}{\widetilde{r}_0 + \sum_{j=1}^{n} \widetilde{r}_j \widetilde{a}_j^{n*}} \tag{B.15}$$

$$= -\frac{\lambda r_{n+1} + \sum_{j=1}^{n} \lambda r_{n+1-j} a_j^n}{\lambda r_0 + \sum_{j=1}^{n} \lambda r_j a_j^{n*}} \tag{B.16}$$

$$= -\frac{\lambda\left(r_{n+1} + \sum_{j=1}^{n} r_{n+1-j} a_j^n\right)}{\lambda\left(r_0 + \sum_{j=1}^{n} r_j a_j^{n*}\right)} \tag{B.17}$$

$$= \mu_{n+1} \tag{B.18}$$

- If $k \leqslant n$, the Levinson algorithm gives us:

$$\widetilde{a}_k^{n+1} = \widetilde{a}_k^n + \widetilde{a}_{n+1}^{n+1}\widetilde{a}_{n+1-k}^{n*} \tag{B.19}$$

$$= a_k^n + a_{n+1}^{n+1} {a_{n+1-k}^n}^* \tag{B.20}$$

$$= a_k^{n+1} \tag{B.21}$$

Conclusion:

$$\widetilde{a}_k^n = a_k^n \quad \forall 1 \leqslant k \leqslant n \tag{B.22}$$

and in particular:

$$\widetilde{\mu}_n = \mu_n \quad \forall n \geqslant 1 \tag{B.23}$$

□

B.1.3 Translated spectrum

We denote $\widetilde{S}_f$ the translated spectrum: $\widetilde{S}_f(x) = S_f(x-m)$.

Property 32. *If we denote $\widetilde{S}_f(x) = S_f(x-m)$, the autocorrelation coefficients of the corresponding spectra $\widetilde{S}_f$ and S_f are linked by the equation:*

$$\widetilde{r}_k = e^{i2\pi m k} r_k \tag{B.24}$$

Proof.

$$\widetilde{R}_f(\tau) = \mathcal{F}\left(\widetilde{S}_f\right)(-\tau) \tag{B.25}$$

$$= \mathcal{F}\left(x \mapsto \widetilde{S}_f(x)\right)(-\tau) \tag{B.26}$$

$$= \mathcal{F}\left(x \mapsto S_f(x-m)\right)(-\tau) \tag{B.27}$$

$$= e^{-i2\pi m(-\tau)} \mathcal{F}\left(S_f\right)(-\tau) \tag{B.28}$$

$$= e^{i2\pi m\tau} R_f(\tau) \tag{B.29}$$

Therefore, we have:

$$\widetilde{r}_k = e^{i2\pi m k} r_k \tag{B.30}$$

□

Property 33. *If we denote $\widetilde{S}_f(x) = S_f(x-m)$, the autoregressive coefficients of the corresponding spectra $\widetilde{S}_f$ and S_f are linked by the equation:*

$$\widetilde{a}_k^n = e^{i2\pi m k} a_k^n \tag{B.31}$$

Proof. Initialization: We have:

$$\widetilde{\mu}_1 = \widetilde{a}_1^1 = -\frac{\widetilde{r}_1}{\widetilde{r}_0} = -\frac{e^{i2\pi m} r_1}{r_0} = e^{i2\pi m}\left(-\frac{r_1}{r_0}\right) = e^{i2\pi m}\mu_1 \tag{B.32}$$

The property is initialized for $n = 1$.

Heredity: We assume that there existe $n \in \mathbb{N}$ such that $\forall k \in [\![1,\ n]\!]$:

$$\widetilde{a}_k^n = e^{i2\pi mk} a_k^n \tag{B.33}$$

We now show that the property is true at the rank $n+1$.

- If $k = n+1$, the Levinson algorithm gives us:

$$\widetilde{\mu}_{n+1} = \widetilde{a}_{n+1}^{n+1} \tag{B.34}$$

$$= -\frac{\widetilde{r}_{n+1} + \sum_{j=1}^{n} \widetilde{r}_{n+1-j}\widetilde{a}_j^n}{\widetilde{r}_0 + \sum_{j=1}^{n} \widetilde{r}_j \widetilde{a}_j^{n*}} \tag{B.35}$$

$$= -\frac{e^{i2\pi m(n+1)} r_{n+1} + \sum_{j=1}^{n} e^{i2\pi m(n+1-j)} r_{n+1-j} e^{i2\pi mj} a_j^n}{r_0 + \sum_{j=1}^{n} e^{i2\pi mj} r_j e^{-i2\pi mj} a_j^{n*}} \tag{B.36}$$

$$= e^{i2\pi m(n+1)}\left(-\frac{r_{n+1} + \sum_{j=1}^{n} r_{n+1-j} a_j^n}{r_0 + \sum_{j=1}^{n} r_j a_j^{n*}}\right) \tag{B.37}$$

$$= e^{i2\pi m(n+1)}\mu_{n+1} \tag{B.38}$$

- If $k \leqslant n$, the Levinson algorithm gives us:

$$\widetilde{a}_k^{n+1} = \widetilde{a}_k^n + \widetilde{a}_{n+1}^{n+1}\widetilde{a}_{n+1-k}^{n*} \tag{B.39}$$

$$= e^{i2\pi mk} a_k^n + e^{i2\pi m(n+1)} a_{n+1}^{n+1} e^{-i2\pi m(n+1-k)} {a_{n+1-k}^n}^* \tag{B.40}$$

$$= e^{i2\pi mk}\left(a_k^n + a_{n+1}^{n+1} {a_{n+1-k}^n}^*\right) \tag{B.41}$$

$$= e^{i2\pi mk} a_k^{n+1} \tag{B.42}$$

Conclusion:

$$\widetilde{a}_k^n = e^{i2\pi mk} a_k^n \quad \forall 1 \leqslant k \leqslant n \tag{B.43}$$

and in particular:

$$\widetilde{\mu}_n = e^{i2\pi mn}\mu_n \quad \forall n \geqslant 1 \tag{B.44}$$

□

Note that the mean m of a symmetric spectrum influence the argument of the reflection coefficients.

B.1.4 Centered symmetric spectra

Any real wide-sense stationary signal has a spectrum that is symmetrical about the y-axis. Reciprocally, we now assume that $S_f(x)$ is a centered symmetric real spectrum.

Property 34. *If $S_f(x)$ is a symmetric centered real spectrum, then its autocorrelation coefficients are real.*

Proof. Conventional calculations show that $\widehat{S_f}(\xi) = 2\int_0^{+\infty} S_f(x)cos(2\pi x\xi)dx$.

Hence we have $R_f(\xi) = \widehat{S_f}(-\xi) = 2\int_0^{+\infty} S_f(x)cos(2\pi x\xi)dx$, and $R_f(\xi) \in \mathbb{R} \quad \forall \xi \in \mathbb{R}$. □

Property 35. *If $S_f(x)$ is a symmetric centered real spectrum, then its autoregressive coefficients are real.*

Proof. We already proved that the autocorrelation coefficients r_k are real. We can prove easily that the coefficients a_k^n are real too by recurrence on the index n using the Levinson algorithm. □

B.2 Application to the theoretical reflection coefficients computation of continuous stochastic processes with Gaussian spectrum shape

We consider the spectrum S_f which shape corresponds to the density of a Gaussian distribution of mean m and variance σ^2 with a power coefficient P, *i.e.*:

$$S_f(\xi) = P\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\xi-m)^2}{2\sigma^2}}. \tag{B.45}$$

We compute the corresponding autocorrelation function R_f:

$$\begin{aligned} R_f(\tau) &= \widehat{S_f}(-\tau) \quad \left(\text{since } S_f(\tau) = \widehat{R_f}(\tau)\right) \\ &= P\frac{1}{\sqrt{2\pi\sigma^2}}\left(\widehat{x \mapsto e^{-\frac{(x-m)^2}{2\sigma^2}}}\right)(-\tau) \\ &= P\frac{1}{\sqrt{2\pi\sigma^2}}e^{-i2\pi m(-\tau)}\left(\widehat{x \mapsto e^{-\frac{x^2}{2\sigma^2}}}\right)(-\tau) \\ &= P\frac{1}{\sqrt{2\pi\sigma^2}}e^{i2\pi m\tau}\sqrt{\frac{\pi}{\frac{1}{2\sigma^2}}}e^{-2\pi^2\sigma^2(-\tau)^2} \\ &= Pe^{i2\pi m\tau}e^{-2\pi^2\sigma^2\tau^2} \end{aligned} \tag{B.46}$$

Finally, if we want to simulate a signal f which sampling period is T, the autocorrelation matrix R (which is Toeplitz) will be defined by its coefficients:

$$r(k) = R_f(kT) \tag{B.47}$$

For simplicity we normalize the time scale and consider the sampling period $T = 1$. We then denote $r_k = r(k)$. The theoretical autocorrelation coefficients of a signal which power spectral density has the shape of Gaussian distribution of mean m and variance σ^2 with a power coefficient P are therefore:

$$r_k = Pe^{i2\pi mk}e^{-2\pi^2\sigma^2k^2} \tag{B.48}$$

Thanks to the Levinson algorithm, we can compute explicitly the expression of the reflection coefficient μ_k.

Property 36. *The autoregressive coefficients of a continuous stochastic process with a Gaussian spectrum shape of mean m and variance σ^2 are:*

$$a_k^n = (-1)^k e^{i2\pi mk}e^{-2\pi^2\sigma^2k}\frac{\prod_{i=1}^{k}\left(1-e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k}\left(1-e^{-4\pi^2\sigma^2 i}\right)}. \tag{B.49}$$

In particular, its reflection coefficients are given by:

$$\mu_k = (-1)^k e^{i2\pi mk}e^{-2\pi^2\sigma^2k} \quad \forall k \geqslant 1. \tag{B.50}$$

To prove this formula, we will proceed by recurrence on the order of the model using the Levinson algorithm (14).

Proof. Using properties 31 and 33, we can consider the case where the power coefficient $P = 1$ and the mean $m = 0$.
It is therefore sufficient to show the spectra:

$$S_f(\xi) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{\xi^2}{2\sigma^2}} \tag{B.51}$$

of autocorrelation coefficients:

$$r_k = e^{-2\pi^2\sigma^2k^2} \tag{B.52}$$

has the following autoregressive coefficients:

$$a_k^n = (-1)^k e^{-2\pi^2\sigma^2k}\frac{\prod_{i=1}^{k}\left(1-e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k}\left(1-e^{-4\pi^2\sigma^2 i}\right)} \tag{B.53}$$

We now prove this equality by recurrence on the index n using the Levinson algorithm.

Initialization: We have:

$$\mu_1 = a_1^1 = -\frac{r_1}{r_0} = -\frac{e^{-2\pi^2\sigma^2}}{1} = -e^{-2\pi^2\sigma^2} \tag{B.54}$$

The property is initialized for $n = 1$.

Heredity: We assume that there existe $n \in \mathbb{N}$ such that $\forall k \in [\![1,\ n]\!]$:

$$a_k^n = (-1)^k e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.55}$$

We now prove that, $\forall k \in [\![1,\ n+1]\!]$:

$$a_k^{n+1} = (-1)^k e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2(n+2-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.56}$$

If we note $X = e^{-4\pi^2\sigma^2}$, we can see that the quotient $\frac{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)}$ is a polynomial quotient in X. Similarly to the fact that $\binom{n}{k}$ is always an integer, this quotient can always be reduced to a simple polynomial in X, and similarly to the fact that $\binom{n}{k} = \binom{n}{n-k}$, we have the following equality:

$$\frac{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} = \frac{\prod_{i=1}^{n-k} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{n-k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.57}$$

Hence the coefficient a_k^n can also be written, $\forall k \in [\![1,\ n]\!]$:

$$a_k^n = (-1)^k e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=1}^{min(k,\ n-k)} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{min(k,\ n-k)} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.58}$$

We now show that the property is true at the rank $n + 1$.

- If $k = n + 1$, the Levinson algorithm gives us:

$$\mu_{n+1} = a_{n+1}^{n+1} \tag{B.59}$$

$$= -\frac{r_{n+1} + \sum_{j=1}^{n} r_{n+1-j} a_j^n}{r_0 + \sum_{j=1}^{n} r_j a_j^{n*}} \tag{B.60}$$

$$= -\frac{r_{n+1} + \sum_{j=1}^{n} r_j a_{n+1-j}^n}{r_0 + \sum_{j=1}^{n} r_j a_j^{n*}} \tag{B.61}$$

$$= -\frac{\sum_{j=1}^{n+1} r_j a_{n+1-j}^n}{\sum_{j=0}^{n} r_j a_j^{n*}} \tag{B.62}$$

$$= -\frac{\sum_{j=0}^{n} r_{j+1} a_{n-j}^n}{\sum_{j=0}^{n} r_j a_j^{n*}} \tag{B.63}$$

considering that $a_0^n = 1$ for all $n \in \mathbb{N}$, which generalize the formula (B.58) to $k = 0$.
We now show that:

$$-\frac{r_{j+1}a_{n-j}^n}{r_j a_j^{n*}} = (-1)^{(n+1)} e^{-2\pi^2\sigma^2(n+1)} \quad \forall j \in [\![0,\ n]\!] \tag{B.64}$$

which will prove that:

$$\mu_{n+1} = (-1)^{(n+1)} e^{-2\pi^2\sigma^2(n+1)}. \tag{B.65}$$

Let's prove Equation (B.64):

$$-\frac{r_{j+1}a_{n-j}^n}{r_j a_j^{n*}} = (-1)\frac{e^{-2\pi^2\sigma^2(j+1)^2}}{e^{-2\pi^2\sigma^2 j^2}} \tag{B.66}$$

$$(-1)^{(n-j)} e^{-2\pi^2\sigma^2(n-j)} \frac{\prod_{i=1}^{min((n-j),\ n-(n-j))} \left(1-e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{min((n-j),\ n-(n-j))} \left(1-e^{-4\pi^2\sigma^2 i}\right)} \Bigg/ (-1)^j e^{-2\pi^2\sigma^2 j} \frac{\prod_{i=1}^{min(j,\ n-j)} \left(1-e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{min(j,\ n-j)} \left(1-e^{-4\pi^2\sigma^2 i}\right)} \tag{B.67}$$

$$= (-1)^{n+1} e^{2\pi^2\sigma^2\left((j+1)^2 - j^2 + n - j - j\right)} \tag{B.68}$$

$$= (-1)^{n+1} e^{2\pi^2\sigma^2(n+1)}. \tag{B.69}$$

- If $k \leqslant n$, the Levinson algorithm gives us:

$$a_k^{n+1} = a_k^n + \mu_{n+1} {a_{n+1-k}^n}^* \tag{B.70}$$

$$= (-1)^k e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=1}^{min(k,\, n-k)} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{min(k,\, n-k)} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} + \tag{B.71}$$

$$(-1)^{n+1} e^{-2\pi^2\sigma^2(n+1)} \tag{B.72}$$

$$(-1)^{(n+1-k)} e^{-2\pi^2\sigma^2(n+1-k)} \tag{B.73}$$

$$\frac{\prod_{i=1}^{min((n+1-k),\, n-(n+1-k))} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{min((n+1-k),\, n-(n+1-k))} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.74}$$

$$= (-1)^k e^{-2\pi^2\sigma^2 k} \tag{B.75}$$

$$\left(\frac{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} + e^{-4\pi^2\sigma^2(n+1-k)} \frac{\prod_{i=1}^{k-1} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k-1} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \right) \tag{B.76}$$

$$= (-1)^k e^{-2\pi^2\sigma^2 k} \tag{B.77}$$

$$\frac{\prod_{i=1}^{k-1} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.78}$$

$$\left(\left(1 - e^{-4\pi^2\sigma^2(n+1-k)}\right) + \left(1 - e^{-4\pi^2\sigma^2 k}\right) e^{-4\pi^2\sigma^2(n+1-k)} \right) \tag{B.79}$$

$$= (-1)^k e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=0}^{k-1} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.80}$$

$$= (-1)^k e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2(n+2-i)}\right)}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \tag{B.81}$$

Conclusion:
Using properties 31 and 33, we finally prove that the clutter of spectrum:

$$S_f(\xi) = P \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\xi - m)^2}{2\sigma^2}} \tag{B.82}$$

has the following autoregressive coefficients:

$$a_k^n = (-1)^k e^{i2\pi mk} e^{-2\pi^2\sigma^2 k} \frac{\prod_{i=1}^{min(k,\, n-k)} \left(1 - e^{-4\pi^2\sigma^2(n+1-i)}\right)}{\prod_{i=1}^{min(k,\, n-k)} \left(1 - e^{-4\pi^2\sigma^2 i}\right)} \quad \forall 1 \leqslant k \leqslant n \tag{B.83}$$

and in particular:

$$\mu_n = (-1)^n e^{i2\pi mn} e^{-2\pi^2\sigma^2 n} \quad \forall n \geqslant 1. \tag{B.84}$$

□

The k^{th} coefficient of the model of infinite order is therefore:

$$a_k^\infty = \lim_{n \to +\infty} a_k^n = (-1)^k e^{i2\pi mk} e^{-2\pi^2\sigma^2 k} \frac{1}{\prod_{i=1}^{k} \left(1 - e^{-4\pi^2\sigma^2 i}\right)}. \tag{B.85}$$

Appendix C

The Burg algorithms

The work presented in this appendix has been introduced by John Parker Burg in his thesis [16]. The Burg algorithm estimates the coefficients a_i^j of the linear autoregressive model from an observed time series [26]. We first define the forward and backward prediction errors which are functions of the autoregressive coefficients a_i^j. The aim of the Burg algorithm is then to compute the coefficients a_i^j minimizing the prediction errors. The Levinson algorithm induce relations between the forward and backward predictions errors of successive orders. The estimations are performed recursively on the order of the model. The Burg algorithm is not a maximum likelihood estimator of the autoregressive coefficients, an exact forward-backward maximum likelihood autoregressive parameter estimation method is given in [5] for Gaussian autoregressive time series.

Contents

C.1 The forward and backward prediction errors recurrence

We would like to justify the expression of the reflection coefficient μ_i of the regularized Burg algorithm 17.

We denote:

$$\widetilde{a}_l = \begin{bmatrix} a_1^l \\ \vdots \\ a_l^l \end{bmatrix} \tag{C.1}$$

and $\widetilde{a}_l^B$ the reversed version of $\widetilde{a}_l$:

$$\widetilde{a}_l^B = \begin{bmatrix} a_l^l \\ \vdots \\ a_1^l \end{bmatrix} \tag{C.2}$$

With these notations, the relation between the autoregressive coefficients presented in Equation (A.6) can be summarized as follows:

$$\widetilde{a}_l = \begin{bmatrix} \widetilde{a}_{l-1} \\ 0 \end{bmatrix} + a_l^l \begin{bmatrix} \widetilde{a}_{l-1}^{B*} \\ 1 \end{bmatrix} = \begin{bmatrix} \widetilde{a}_{l-1} \\ 0 \end{bmatrix} + \mu_l \begin{bmatrix} \widetilde{a}_{l-1}^{B*} \\ 1 \end{bmatrix} \tag{C.3}$$

Let $\mathbf{u} = [u_0, \ldots, u_{n-1}]$ be a one-dimesional temporal signal.

We define the forward prediction error:

$$f_k^j = u_k + \sum_{l=1}^{j} a_l^j \, u_{k-l}, \quad j \leqslant k \leqslant n-1 \tag{C.4}$$

and the backward prediction error:

$$b_k^j = u_{k-j} + \sum_{l=1}^{j} {a_l^j}^* \, u_{k-j+l}, \quad j \leqslant k \leqslant n-1 \tag{C.5}$$

Theorem 6. *The forward and backward prediction errors of successive orders are linked by the following relations:*

$$\left\{ \begin{array}{l} f_k^{j+1} = f_k^j + \mu_{j+1} \, b_{k-1}^j \\ b_k^{j+1} = b_{k-1}^j + \mu_{j+1}^* \, f_k^j \end{array} \right. \tag{C.6}$$

Proof. The proof is based on the Levinson recurrence.

$$f_k^{j+1} = u_k + \sum_{l=1}^{j+1} a_l^{j+1} u_{k-l} \tag{C.7}$$

$$= u_k + \widetilde{a}_{j+1}^T \begin{bmatrix} u_{k-1} \\ \vdots \\ u_{k-j-1} \end{bmatrix} \tag{C.8}$$

$$= u_k + \left(\begin{bmatrix} \widetilde{a}_j \\ 0 \end{bmatrix} + \mu_{j+1} \begin{bmatrix} \widetilde{a}_j^{B*} \\ 1 \end{bmatrix} \right)^T \begin{bmatrix} u_{k-1} \\ \vdots \\ u_{k-j-1} \end{bmatrix} \tag{C.9}$$

$$= u_k + \begin{bmatrix} \widetilde{a}_j \\ 0 \end{bmatrix}^T \begin{bmatrix} u_{k-1} \\ \vdots \\ u_{k-j-1} \end{bmatrix} + \mu_{j+1} \begin{bmatrix} \widetilde{a}_j^{B*} \\ 1 \end{bmatrix}^T \begin{bmatrix} u_{k-1} \\ \vdots \\ u_{k-j-1} \end{bmatrix} \tag{C.10}$$

$$= f_k^j + \mu_{j+1} \, b_{k-1}^j \tag{C.11}$$

$$b_k^{j+1} = u_{k-(j+1)} + \sum_{l=1}^{j+1} {a_l^{j+1}}^* u_{k-(j+1)+l} \tag{C.12}$$

$$= u_{k-(j+1)} + \widetilde{a}_{j+1}^{*T} \begin{bmatrix} u_{k-j} \\ \vdots \\ u_k \end{bmatrix} \tag{C.13}$$

$$= u_{k-(j+1)} + \left(\begin{bmatrix} \widetilde{a}_j \\ 0 \end{bmatrix} + \mu_{j+1} \begin{bmatrix} \widetilde{a}_j^{B*} \\ 1 \end{bmatrix} \right)^{*T} \begin{bmatrix} u_{k-j} \\ \vdots \\ u_k \end{bmatrix} \tag{C.14}$$

$$= u_{k-(j+1)} + \begin{bmatrix} \widetilde{a}_j \\ 0 \end{bmatrix}^{*T} \begin{bmatrix} u_{k-j} \\ \vdots \\ u_k \end{bmatrix} + {\mu_{j+1}}^* \begin{bmatrix} \widetilde{a}_j^{B*} \\ 1 \end{bmatrix}^{*T} \begin{bmatrix} u_{k-j} \\ \vdots \\ u_k \end{bmatrix} \tag{C.15}$$

$$= b_{k-1}^j + \mu_{j+1}^* \, f_k^j \tag{C.16}$$

□

The Burg algorithms presented in Sections C.2, C.3, C.4 and C.5 are based on the minimization of the forward and backward prediction errors.

C.2 The classical Burg algorithm

We present the classical Burg algorithm in Algorithm 16.

We would like to justifie the expression of the reflection coefficient μ_j in Equation (C.19).

The idea is to minimize the sum of the square modulus of the forward and backward prediction errors of order j, i.e. we would like to minimize the function:

$$E_j = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} \left| f_k^j \right|^2 + \left| b_k^j \right|^2 \tag{C.21}$$

Algorithm 16 The classical Burg algorithm

Initialization:

$$f_k^0 = b_k^0 = u_k \quad k = 0, ..., n-1 \tag{C.17}$$

$$p_0 = \frac{1}{n} \sum_{k=0}^{n-1} |u_k|^2 \tag{C.18}$$

for $j = 1, ..., M$: **do**

$$\mu_j = -\frac{2 \sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*}}{\sum_{k=j}^{n-1} \left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2} \tag{C.19}$$

where:

$$\left\{ \begin{array}{rcll} f_k^j & = & f_k^{j-1} + \mu_j\, b_{k-1}^{j-1} & k = j, ..., n-1 \\ b_k^j & = & b_{k-1}^{j-1} + \mu_j^*\, f_k^{j-1} & k = j, ..., n-1 \end{array} \right. \tag{C.20}$$

end for

return $(p_0, \mu_1, ..., \mu_{n-1})$

Note that we can express E_j as a function of forward and backward prediction errors of order $j-1$ and μ_j:

$$E_j = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} \left|f_k^{j-1} + \mu_j\, b_{k-1}^{j-1}\right|^2 + \left|b_{k-1}^{j-1} + \mu_j^*\, f_k^{j-1}\right|^2 \tag{C.22}$$

Hence if the prediction errors of order $j-1$ are already known, we can compute the optimal value of μ_j to minimize E_j.

Property 37. *The minimal value of E_j is obtained for:*

$$\mu_j = -\frac{2 \sum_{k=j}^{n-1} f_k^{j-1} b_{k-1}^{j-1*}}{\sum_{k=j}^{n-1} |f_k^{j-1}|^2 + |b_{k-1}^{j-1}|^2} \tag{C.23}$$

Proof.

$$E_j = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} \left|f_k^{j-1} + \mu_j\, b_{k-1}^{j-1}\right|^2 + \left|b_{k-1}^{j-1} + \mu_j^*\, f_k^{j-1}\right|^2 \tag{C.24}$$

$$\begin{aligned} = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} & \left|f_k^{j-1}\right|^2 + \\ & \mu_j\, b_{k-1}^{j-1}\, f_k^{j-1*} + \\ & \mu_j^*\, b_{k-1}^{j-1*}\, f_k^{j-1} + \\ & |\mu_j|^2 \left|b_{k-1}^{j-1}\right|^2 + \\ & \left|b_{k-1}^{j-1}\right|^2 + \\ & \mu_j\, f_k^{j-1*}\, b_{k-1}^{j-1} + \\ & \mu_j^*\, f_k^{j-1}\, b_{k-1}^{j-1*} + \\ & |\mu_j|^2 \left|f_k^{j-1}\right|^2 \end{aligned} \tag{C.25}$$

If we denote $\mu_j = x + iy$, then we have:

$$E_j = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} \left| f_k^{j-1} \right|^2 + \tag{C.26}$$
$$(x+iy)\, b_{k-1}^{j-1}\, f_k^{j-1*} +$$
$$(x-iy)\, b_{k-1}^{j-1*}\, f_k^{j-1} +$$
$$(x^2+y^2)\, \left| b_{k-1}^{j-1} \right|^2 +$$
$$\left| b_{k-1}^{j-1} \right|^2 +$$
$$(x+iy)\, f_k^{j-1*}\, b_{k-1}^{j-1} +$$
$$(x-iy)\, f_k^{j-1}\, b_{k-1}^{j-1*} +$$
$$(x^2+y^2) \left| f_k^{j-1} \right|^2$$

Hence we have:

$$\frac{\partial E_j(x+iy)}{\partial x} = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} b_{k-1}^{j-1}\, f_k^{j-1*} + \tag{C.27}$$
$$b_{k-1}^{j-1*}\, f_k^{j-1} +$$
$$2\, x\, \left| b_{k-1}^{j-1} \right|^2 +$$
$$f_k^{j-1*}\, b_{k-1}^{j-1} +$$
$$f_k^{j-1}\, b_{k-1}^{j-1*} +$$
$$2\, x \left| f_k^{j-1} \right|^2$$
$$= \frac{1}{2(n-j)} \sum_{k=j}^{n-1} 4\, Re\left(f_k^{j-1}\, b_{k-1}^{j-1*} \right) + \tag{C.28}$$
$$2\, x\, \left(\left| f_k^{j-1} \right|^2 + \left| b_{k-1}^{j-1} \right|^2 \right)$$

Hence:

$$\frac{\partial E_j(x+iy)}{\partial x} = 0 \tag{C.29}$$
$$\iff x = Re\left(-\frac{2 \sum_{k=j}^{n-1} f_k^{j-1} b_{k-1}^{j-1*}}{\sum_{k=j}^{n-1} |f_k^{j-1}|^2 + |b_{k-1}^{j-1}|^2} \right)$$

We also have:

$$\frac{\partial E_j(x+iy)}{\partial y} = \frac{1}{2(n-j)} \sum_{k=j}^{n-1} i\, b_{k-1}^{j-1}\, f_k^{j-1*} + \tag{C.30}$$
$$- i\, b_{k-1}^{j-1*}\, f_k^{j-1} +$$
$$2\, y\, \left|b_{k-1}^{j-1}\right|^2 +$$
$$i\, f_k^{j-1*}\, b_{k-1}^{j-1} +$$
$$- i\, f_k^{j-1}\, b_{k-1}^{j-1*} +$$
$$2\, y \left|f_k^{j-1}\right|^2$$
$$= \frac{1}{2(n-j)} \sum_{k=j}^{n-1} 4\, Im\left(f_k^{j-1}\, b_{k-1}^{j-1*}\right) + \tag{C.31}$$
$$2\, y\, \left(\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2\right)$$

Hence:

$$\frac{\partial E_j(x+iy)}{\partial y} = 0 \tag{C.32}$$
$$\iff y = Im\left(-\frac{2\sum_{k=j}^{n-1} f_k^{j-1} b_{k-1}^{j-1*}}{\sum_{k=j}^{n-1} |f_k^{j-1}|^2 + |b_{k-1}^{j-1}|^2}\right)$$

Hence the gradient of the function E_j respectively to the coordinate x and y is null at the point:

$$\mu_j = -\frac{2\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*}}{\sum_{k=j}^{n-1} |f_k^{j-1}|^2 + |b_{k-1}^{j-1}|^2} \tag{C.33}$$

As the function E_j is a strictly convex function, the function E_j has a global minimum at μ_j.

□

Property 38. *The modulus of the reflection coefficients estimated by Algorithm 16 are lower than one:*

$$|\mu_j| \leqslant 1, \quad \forall j \in \mathbb{N}^* \tag{C.34}$$

Proof. Note that $\forall(j,k)$ we have:

$$2\left|\, f_k^{j-1}\, b_{k-1}^{j-1*}\right| \leqslant \left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2 \tag{C.35}$$

If we sum the inequalities, we the have:

$$2\sum_{k=j}^{n-1}\left|f_k^{j-1}\, b_{k-1}^{j-1*}\right| \leqslant \sum_{k=j}^{n-1}\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2 \tag{C.36}$$

Yet, using the triangular inequality:

$$2\left|\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*}\right| \leqslant 2\sum_{k=j}^{n-1}\left|f_k^{j-1}\, b_{k-1}^{j-1*}\right| \tag{C.37}$$

Therefore:

$$2\left|\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*}\right| \leqslant \sum_{k=j}^{n-1} |f_k^{j-1}|^2 + |b_{k-1}^{j-1}|^2 \tag{C.38}$$

$$\iff \frac{2\left|\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*}\right|}{\sum_{k=j}^{n-1} \left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2} \leqslant 1 \tag{C.39}$$

$$\iff |\mu_j| \leqslant 1 \tag{C.40}$$

□

Note that $|\mu_j| = 1$ if and only if all inequalities in the proof of Property 38 are equalities. Therefore, we have $|\mu_j| < 1$ almost surely.

C.3 The regularized Burg algorithm

We present the regularized Burg algorithm in Algorithm 17.

Algorithm 17 The regularized Burg algorithm

Initialization:

$$f_k^0 = b_k^0 = u_k \quad k = 0, ..., p-1 \tag{C.41}$$

$$a_k^0 = 1 \quad k = 0, ..., p-1 \tag{C.42}$$

$$p_0 = \frac{1}{p}\sum_{k=0}^{p-1} |u_k|^2 \tag{C.43}$$

for $j = 1, ..., n-1$: **do**

$$\mu_j = -\frac{\frac{2}{p-j}\sum_{k=j}^{p-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\sum_{k=1}^{j-1} \beta_k^j\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{p-j}\sum_{k=j}^{p-1}\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2 + 2\sum_{k=0}^{j-1} \beta_k^j \left|a_k^{j-1}\right|^2} \tag{C.44}$$

where:

$$\beta_k^j = \gamma(2\pi)^2(j-k)^2 \tag{C.45}$$

$$\begin{cases} a_k^j &= a_k^{j-1} + \mu_j\, a_{j-k}^{j-1*} \quad k = 1, ..., j-1 \\ a_j^j &= \mu_j \end{cases} \tag{C.46}$$

$$\begin{cases} f_k^j &= f_k^{j-1} + \mu_j\, b_{k-1}^{j-1} \quad k = j, ..., p-1 \\ b_k^j &= b_{k-1}^{j-1} + \mu_j^*\, f_k^{j-1} \quad k = j, ..., p-1 \end{cases} \tag{C.47}$$

end for

return $(p_0, \mu_1, ..., \mu_{n-1})$

We would like to justify the expression of the reflection coefficient μ_j in Equation (C.44).

The idea is to minimize the sum of the square modulus of the forward and backward prediction errors of order j plus a regularization coefficient, i.e. we would like to minimize the function:

$$E_j = \underbrace{\frac{1}{2(n-j)}\sum_{k=j}^{n-1}\left|f_k^j\right|^2 + \left|b_k^j\right|^2}_{F_j} + \underbrace{\gamma\,(2\pi)^2\sum_{k=1}^{j} k^2\left|a_k^j\right|^2}_{G_j} \tag{C.48}$$

The function F_j has already been studied in the previous section dealing with the classical Burg algorithm C.2. The function G_j is a regularization function of parameter γ. We explain why the function G_j can be considered as regularization function, then we will compute its derivative in order to minimize the function E_j.

We denote $A^m(f)$ the polynomial associated with the autoregressive model of order m:

$$A^m(f) = \sum_{k=0}^{m} a_k^m e^{-j2\pi kf} \tag{C.49}$$

Note that this polynomial corresponds to denominator of the autoregressive spectrum $S_{AR}^m(f)$ presented in Equation (2.95):

$$S_{AR}^m(f) = \frac{P_m}{A^m(f)}. \tag{C.50}$$

Hence the variation of the autoregressive spectrum $S_{AR}^m(f)$, are linked to the variations of the polynomial $A^m(f)$ via the equation:

$$\frac{dS_{AR}^m(f)}{df} = -\frac{P_m \frac{dA^m(f)}{df}}{\left(A^m(f)\right)^2} \tag{C.51}$$

Hence we can obtain a smooth spectrum minimizing the modulus of the first order derivative $\frac{dA^m(f)}{df}$.

The following coefficient as been introduced in [8, 9] to quantify the variations of the polynomial $A^m(f)$:

$$D_1^m = \int_{-1/2}^{1/2} \left|\frac{dA^m(f)}{df}\right|^2 df \tag{C.52}$$

Property 39. *Using the previous notations, we have the following equality:*

$$\int_{-1/2}^{1/2} \left|\frac{dA^m(f)}{df}\right|^2 df = (2\pi)^2 \sum_{k=1}^{m} k^2 \left|a_k^m\right|^2 \tag{C.53}$$

This equality justifies that the function G_j is called a regularization function.

Proof. We recall the expression of the polynomial $A^m(f)$ associated with the autoregressive model of order m:

$$A^m(f) = \sum_{k=0}^{m} a_k^m e^{-j2\pi kf} \tag{C.54}$$

We compute the first order derivative of $A^m(f)$:

$$\frac{dA^m(f)}{df} = -j2\pi \sum_{k=1}^{m} k a_k^m e^{-j2\pi kf} \tag{C.55}$$

Hence its square modulus is:

$$\left|\frac{dA^m(f)}{df}\right|^2 = (2\pi)^2 \left|\sum_{k=1}^{m} k a_k^m e^{-j2\pi kf}\right|^2 \tag{C.56}$$

which gives:

$$\left|\frac{dA^m(f)}{df}\right|^2 = (2\pi)^2 \sum_{1 \leqslant i,l \leqslant m} i\ l\ a_i^m a_l^{m*} e^{-j2\pi(i-l)f} \tag{C.57}$$

Integrating with respect to f, we obtain:

$$\int_{-1/2}^{1/2} \left| \frac{dA^m(f)}{df} \right|^2 df \tag{C.58}$$

$$= (2\pi)^2 \sum_{1 \leqslant i,l \leqslant m} i\ l\ a_i^m a_l^{m*} \int_{-1/2}^{1/2} e^{-j2\pi(i-l)f} df \tag{C.59}$$

$$= (2\pi)^2 \sum_{1 \leqslant i,l \leqslant m} i\ l\ a_i^m a_l^{m*} \mathbb{1}_{\{i=l\}} \tag{C.60}$$

$$= (2\pi)^2 \sum_{i=1}^{m} i^2 \left| a_i^m \right|^2 \tag{C.61}$$

□

Property 40. *The minimal value of E_j is obtained for:*

$$\mu_j = -\frac{\frac{2}{n-j} \sum_{k=j}^{n-1} f_k^{j-1}\ b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1} (j-k)^2\ a_k^{j-1}\ a_{j-k}^{j-1}}{\frac{1}{n-j} \sum_{k=j}^{n-1} \left| f_k^{j-1} \right|^2 + \left| b_{k-1}^{j-1} \right|^2 + 2\gamma(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2\ \left| a_k^{j-1} \right|^2} \tag{C.62}$$

Proof. Let's start by recalling the formula of the regularization function G_j:

$$G_j = \gamma\,(2\pi)^2 \sum_{k=1}^{j} k^2 \left| a_k^j \right|^2. \tag{C.63}$$

Performing a summation index change, we have:

$$G_j = \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left| a_{j-k}^j \right|^2 \tag{C.64}$$

We recall the following recurrence relation of the AR coefficients obtained in Appendix A:

$$a_k^j = a_k^{j-1} + \mu_j a_{j-k}^{j-1*} \quad \forall\,(i,j) \in \mathbb{N}^2 \tag{C.65}$$

with by convention $a_0^j = 1$ for all $j \in \mathbb{N}$ and $a_i^j = 0$ when $i > j$.
Therefore, we have:

$$G_j = \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left| a_{j-k}^{j-1} + \mu_j\ a_k^{j-1\,*} \right|^2 \tag{C.66}$$

$$= \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left(\left| a_{j-k}^{j-1} \right|^2 + \right. \tag{C.67}$$

$$\mu_j a_k^{j-1\,*} a_{j-k}^{j-1\,*} +$$

$$\mu_j^* a_k^{j-1} a_{j-k}^{j-1} +$$

$$\left. |\mu_j|^2 \left| a_k^{j-1} \right|^2 \right)$$

If we denote $\mu_j = x + iy$, then we have:

$$G_j = \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left(\left| a_{j-k}^{j-1} \right|^2 \right. \tag{C.68}$$

$$+ (x+iy)\, a_k^{j-1\,*} a_{j-k}^{j-1\,*}$$

$$+ (x-iy)\, a_k^{j-1} a_{j-k}^{j-1}$$

$$\left. + \left(x^2+y^2\right) \left| a_k^{j-1} \right|^2 \right)$$

Hence we have:

$$\frac{\partial G_j(x+iy)}{\partial x} = \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \Big(a_k^{j-1} * a_{j-k}^{j-1} * + a_k^{j-1} a_{j-k}^{j-1} + 2x \left| a_k^{j-1} \right|^2 \Big) \tag{C.69}$$

$$= \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left(2\,Re\left(a_k^{j-1} a_{j-k}^{j-1} \right) + 2x \left| a_k^{j-1} \right|^2 \right) \tag{C.70}$$

Hence:

$$\frac{\partial E_j(x+iy)}{\partial x} = 0 \tag{C.71}$$

$$\iff \frac{\partial F_j(x+iy)}{\partial x} + \frac{\partial G_j(x+iy)}{\partial x} = 0 \tag{C.72}$$

$$\iff \frac{1}{2(n-j)} \sum_{k=j}^{n-1} 4\,Re\left(f_k^{j-1}\, b_{k-1}^{j-1*} \right) + 2\,x\, \left(\left| f_k^{j-1} \right|^2 + \left| b_{k-1}^{j-1} \right|^2 \right) + \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left(2Re\left(a_k^{j-1} a_{j-k}^{j-1} \right) + 2x \left| a_k^{j-1} \right|^2 \right) = 0 \tag{C.73}$$

$$\iff x = Re\left(- \frac{\frac{2}{n-j} \sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1} (j-k)^2\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{n-j} \sum_{k=j}^{n-1} \left| f_k^{j-1} \right|^2 + \left| b_{k-1}^{j-1} \right|^2 + 2\gamma(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2\, \left| a_k^{j-1} \right|^2} \right) \tag{C.74}$$

We also have:

$$\frac{\partial G_j(x+iy)}{\partial y} = \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \Big(i\, a_k^{j-1} * a_{j-k}^{j-1} * - i\, a_k^{j-1} a_{j-k}^{j-1} + 2y \left| a_k^{j-1} \right|^2 \Big) \tag{C.75}$$

$$= \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left(2\,Im\left(a_k^{j-1} a_{j-k}^{j-1} \right) + 2y \left| a_k^{j-1} \right|^2 \right) \tag{C.76}$$

Hence:

$$\frac{\partial E_j(x+iy)}{\partial y} = 0 \tag{C.77}$$

$$\iff \frac{\partial F_j(x+iy)}{\partial y} + \frac{\partial G_j(x+iy)}{\partial y} = 0 \tag{C.78}$$

$$\iff \frac{1}{2(n-j)} \sum_{k=j}^{n-1} 4\,Im\left(f_k^{j-1}\, b_{k-1}^{j-1*} \right) + 2\,y\, \left(\left| f_k^{j-1} \right|^2 + \left| b_{k-1}^{j-1} \right|^2 \right) + \gamma\,(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2 \left(2\,Im\left(a_k^{j-1} a_{j-k}^{j-1} \right) + 2y \left| a_k^{j-1} \right|^2 \right) = 0 \tag{C.79}$$

$$\iff y = Im\left(- \frac{\frac{2}{n-j} \sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1} (j-k)^2\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{n-j} \sum_{k=j}^{n-1} \left| f_k^{j-1} \right|^2 + \left| b_{k-1}^{j-1} \right|^2 + 2\gamma(2\pi)^2 \sum_{k=0}^{j-1} (j-k)^2\, \left| a_k^{j-1} \right|^2} \right) \tag{C.80}$$

Hence the gradient of the function E_j respectively to the coordinate x and y is null at the point:

$$\mu_j = -\frac{\frac{2}{n-j}\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1}(j-k)^2\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{n-j}\sum_{k=j}^{n-1}\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2 + 2\gamma(2\pi)^2 \sum_{k=0}^{j-1}(j-k)^2\,\left|a_k^{j-1}\right|^2} \tag{C.81}$$

As the function E_j is a strictly convex function, the function E_j has a global minimum at μ_j.

□

Property 41. *The coefficients μ_j computed by the regularized Burg algorithm 17 belong to the complex unit disk for all $1 \leqslant j \leqslant 5$:*

$$|\mu_j| \leqslant 1 \quad \forall 1 \leqslant j \leqslant 5 \tag{C.82}$$

In practice, all the reflection coefficients estimated from a time series using the regularized Burg algorithm are of modulus lower than one for all $j \in \mathbb{N}^*$. Indeed, the modulus of the j^{th} coefficient tends to decrease as the model order j increases. However, we only managed to write the proof for $1 \leqslant i \leqslant 5$ here. The main difficulty of the proof is the coefficient $(j-k)^2$ appearing in the expression of the coefficient μ_j:

$$\mu_j = -\frac{\frac{2}{n-j}\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1}\mathbf{(j-k)^2}\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{n-j}\sum_{k=j}^{n-1}\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2 + 2\gamma(2\pi)^2 \sum_{k=0}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\right|^2}. \tag{C.83}$$

Proof. According to Equation (C.83), it is equivalent to prove that $|\mu_j| \leqslant 1$ and to prove that:

$$\begin{aligned}&\left|\frac{2}{n-j}\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1}\mathbf{(j-k)^2}\, a_k^{j-1}\, a_{j-k}^{j-1}\right| \\ \leqslant &\frac{1}{n-j}\sum_{k=j}^{n-1}\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2 + 2\gamma(2\pi)^2 \sum_{k=0}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\right|^2.\end{aligned} \tag{C.84}$$

According to the triangular inequality, we have:

$$\begin{aligned}&\left|\frac{2}{n-j}\sum_{k=j}^{n-1} f_k^{j-1}\, b_{k-1}^{j-1*} + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1}\mathbf{(j-k)^2}\, a_k^{j-1}\, a_{j-k}^{j-1}\right| \\ \leqslant &\frac{2}{n-j}\sum_{k=j}^{n-1}\left|f_k^{j-1}\, b_{k-1}^{j-1*}\right| + 2\gamma(2\pi)^2 \sum_{k=1}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\, a_{j-k}^{j-1}\right|.\end{aligned} \tag{C.85}$$

In the proof of Property 38, we have already proven in Equation (C.36) that:

$$2\sum_{k=j}^{n-1}\left|f_k^{j-1}\, b_{k-1}^{j-1*}\right| \leqslant \sum_{k=j}^{n-1}\left|f_k^{j-1}\right|^2 + \left|b_{k-1}^{j-1}\right|^2. \tag{C.86}$$

Therefore, it is sufficient to show that:

$$\left|\sum_{k=1}^{j-1}\mathbf{(j-k)^2}\, a_k^{j-1}\, a_{j-k}^{j-1}\right| \leqslant \sum_{k=0}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\right|^2 \tag{C.87}$$

to prove that $|\mu_j| \leqslant 1$.
We recall that $a_0^n = 1$ for all $n \in \mathbb{N}$, Equation (C.87) can therefore be written:

$$\left|\sum_{k=1}^{j-1}\mathbf{(j-k)^2}\, a_k^{j-1}\, a_{j-k}^{j-1}\right| \leqslant j^2 + \sum_{k=1}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\right|^2. \tag{C.88}$$

Note that we cannot use the Cauchy-Schwarz inequality to prove directly this inequality because of the terms $\mathbf{(j-k)^2}$. We now prove that

$$\sum_{k=1}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\right|\left|a_{j-k}^{j-1}\right| \leqslant j^2 + \sum_{k=1}^{j-1}\mathbf{(j-k)^2}\,\left|a_k^{j-1}\right|^2 \tag{C.89}$$

for all $j \in [\![1, 5]\!]$. When Equation (C.89) is true, we can use the triangular inequality to prove that Equation (C.87) is also true and therefore $|\mu_j| \leqslant 1$.

For $j = 1$, Equation (C.89) gives:

$$\sum_{k=1}^{0} (1-k)^2 \ |a_k^0| \ |a_{1-k}^0| \leqslant 1^2 + \sum_{k=1}^{0} (1-k)^2 \ |a_k^0|^2 \tag{C.90}$$

$$\Leftrightarrow 0 \leqslant 1 \tag{C.91}$$

which is always true.

For $j = 2$, Equation (C.89) gives:

$$\sum_{k=1}^{1} (2-k)^2 \ |a_k^1| \ |a_{2-k}^1| \leqslant 2^2 + \sum_{k=1}^{1} (2-k)^2 \ |a_k^1|^2 \tag{C.92}$$

$$\Leftrightarrow |a_1^1|^2 \leqslant 4 + |a_1^1|^2 \tag{C.93}$$

$$\tag{C.94}$$

which is always true.

For $j = 3$, Equation (C.89) gives:

$$\sum_{k=1}^{2} (3-k)^2 \ |a_k^2| \ |a_{3-k}^2| \leqslant 3^2 + \sum_{k=1}^{2} (3-k)^2 \ |a_k^2|^2 \tag{C.95}$$

$$\Leftrightarrow 4 |a_1^2| |a_2^2| + |a_1^2| |a_2^2| \leqslant 9 + 4 |a_1^2|^2 + |a_2^2|^2 \tag{C.96}$$

$$\Leftrightarrow 5 |a_1^2| |a_2^2| \leqslant 9 + 4 |a_1^2|^2 + |a_2^2|^2 \tag{C.97}$$

Note that:

$$0 \leqslant \left(2 |a_1^2| - \frac{5}{4} |a_2^2|\right)^2 \tag{C.98}$$

$$\Leftrightarrow 0 \leqslant 4 |a_1^2|^2 - 5 |a_1^2| |a_2^2| + \frac{25}{16} |a_2^2|^2 \tag{C.99}$$

$$\Leftrightarrow 5 |a_1^2| |a_2^2| \leqslant 4 |a_1^2|^2 + \frac{25}{16} |a_2^2|^2 \tag{C.100}$$

Moreover, we have:

$$|a_2^2|^2 \leqslant 1 \tag{C.101}$$

$$\Leftrightarrow \frac{9}{16} |a_2^2|^2 \leqslant \frac{9}{16} \tag{C.102}$$

$$\Leftrightarrow 0 \leqslant \frac{9}{16} - \frac{9}{16} |a_2^2|^2 \tag{C.103}$$

By adding the inequalities (C.100) and (C.103), we obtain:

$$5 |a_1^2| |a_2^2| \leqslant 4 |a_1^2|^2 + \frac{25}{16} |a_2^2|^2 + \frac{9}{16} - \frac{9}{16} |a_2^2|^2 \tag{C.104}$$

$$\Leftrightarrow 5 |a_1^2| |a_2^2| \leqslant \frac{9}{16} + 4 |a_1^2|^2 + |a_2^2|^2 \tag{C.105}$$

$$\Rightarrow 5 |a_1^2| |a_2^2| \leqslant 9 + 4 |a_1^2|^2 + |a_2^2|^2 \tag{C.106}$$

which proves the inequality for $j = 3$.

For $j = 4$, Equation (C.89) gives:

$$\sum_{k=1}^{3} (4-k)^2 \ |a_k^3| \ |a_{4-k}^3| \leqslant 4^2 + \sum_{k=1}^{3} (4-k)^2 \ |a_k^3|^2 \tag{C.107}$$

$$\Leftrightarrow 9 |a_1^3| |a_3^3| + 4 |a_2^3| |a_2^3| + |a_3^3| |a_1^3| \leqslant 16 + 9 |a_1^3|^2 + 4 |a_2^3|^2 + |a_3^3|^2 \tag{C.108}$$

$$\Leftrightarrow 10 |a_1^3| |a_3^3| \leqslant 16 + 9 |a_1^3|^2 + |a_3^3|^2 \tag{C.109}$$

Note that:

$$0 \leqslant \left(3\left|a_1^3\right| - \frac{5}{3}\left|a_3^3\right|\right)^2 \tag{C.110}$$

$$\Leftrightarrow 0 \leqslant 9\left|a_1^3\right|^2 - 10\left|a_1^3\right|\left|a_3^3\right| + \frac{25}{9}\left|a_3^3\right|^2 \tag{C.111}$$

$$\Leftrightarrow 10\left|a_1^3\right|\left|a_3^3\right| \leqslant 9\left|a_1^3\right|^2 + \frac{25}{9}\left|a_3^3\right|^2 \tag{C.112}$$

Moreover, we have:

$$\left|a_3^3\right|^2 \leqslant 1 \tag{C.113}$$

$$\Leftrightarrow \frac{16}{9}\left|a_3^3\right|^2 \leqslant \frac{16}{9} \tag{C.114}$$

$$\Leftrightarrow 0 \leqslant \frac{16}{9} - \frac{16}{9}\left|a_3^3\right|^2 \tag{C.115}$$

By adding the inequalities (C.112) and (C.115), we obtain:

$$10\left|a_1^3\right|\left|a_3^3\right| \leqslant 9\left|a_1^3\right|^2 + \frac{25}{9}\left|a_3^3\right|^2 + \frac{16}{9} - \frac{16}{9}\left|a_3^3\right|^2 \tag{C.116}$$

$$\Leftrightarrow 10\left|a_1^3\right|\left|a_3^3\right| \leqslant \frac{16}{9} + 9\left|a_1^3\right|^2 + \left|a_3^3\right|^2 \tag{C.117}$$

$$\Rightarrow 10\left|a_1^3\right|\left|a_3^3\right| \leqslant 16 + 9\left|a_1^3\right|^2 + \left|a_3^3\right|^2 \tag{C.118}$$

which proves the inequality for $j = 4$.
For $j = 5$, Equation (C.89) gives:

$$\sum_{k=1}^{4}(5-k)^2\ \left|a_k^4\right|\ \left|a_{5-k}^4\right| \leqslant 5^2 + \sum_{k=1}^{4}(5-k)^2\ \left|a_k^4\right|^2 \tag{C.119}$$

$$\Leftrightarrow 16\left|a_1^4\right|\left|a_4^4\right| + 9\left|a_2^4\right|\left|a_3^4\right| + 4\left|a_3^4\right|\left|a_2^4\right| + \left|a_4^4\right|\left|a_1^4\right| \leqslant 25 + 16\left|a_1^4\right|^2 + 9\left|a_2^4\right|^2 + 4\left|a_3^4\right|^2 + \left|a_4^4\right|^2 \tag{C.120}$$

$$\Leftrightarrow 17\left|a_1^4\right|\left|a_4^4\right| + 13\left|a_2^4\right|\left|a_3^4\right| \leqslant 25 + 16\left|a_1^4\right|^2 + 9\left|a_2^4\right|^2 + 4\left|a_3^4\right|^2 + \left|a_4^4\right|^2 \tag{C.121}$$

Note that:

$$0 \leqslant \left(4\left|a_1^4\right| - \frac{17}{8}\left|a_4^4\right|\right)^2 \tag{C.122}$$

$$\Leftrightarrow 0 \leqslant 16\left|a_1^4\right|^2 - 17\left|a_1^4\right|\left|a_4^4\right| + \frac{17^2}{8^2}\left|a_4^4\right|^2 \tag{C.123}$$

$$\Leftrightarrow 17\left|a_1^4\right|\left|a_4^4\right| \leqslant 16\left|a_1^4\right|^2 + \frac{289}{64}\left|a_4^4\right|^2 \tag{C.124}$$

Moreover, we have:

$$\left|a_4^4\right|^2 \leqslant 1 \tag{C.125}$$

$$\Leftrightarrow \frac{225}{64}\left|a_4^4\right|^2 \leqslant \frac{225}{64} \tag{C.126}$$

$$\Leftrightarrow 0 \leqslant \frac{225}{64} - \frac{225}{64}\left|a_4^4\right|^2 \tag{C.127}$$

We also have:

$$0 \leqslant \left(3\left|a_2^4\right| - \frac{13}{6}\left|a_3^4\right|\right)^2 \tag{C.128}$$

$$\Leftrightarrow 0 \leqslant 9\left|a_2^4\right|^2 - 13\left|a_2^4\right|\left|a_3^4\right| + \frac{13^2}{6^2}\left|a_3^4\right|^2 \tag{C.129}$$

$$\Leftrightarrow 13\left|a_2^4\right|\left|a_3^4\right| \leqslant 9\left|a_2^4\right|^2 + \frac{169}{36}\left|a_3^4\right|^2 \tag{C.130}$$

Besides, we have according to Property 28:

$$\left|a_3^4\right| \leqslant \binom{4}{3} = 4 \tag{C.131}$$

$$\Leftrightarrow \left|a_3^4\right|^2 \leqslant 16 \tag{C.132}$$

$$\Leftrightarrow \frac{25}{36}\left|a_3^4\right|^2 \leqslant \frac{25}{36} \times 16 = \frac{100}{9} \tag{C.133}$$

$$\Leftrightarrow 0 \leqslant \frac{100}{9} - \frac{25}{36}\left|a_3^4\right|^2 \tag{C.134}$$

By adding the inequalities (C.124), (C.127), (C.130)and (C.134), we obtain:

$$17\left|a_1^4\right|\left|a_4^4\right| + 13\left|a_2^4\right|\left|a_3^4\right| \leqslant 16\left|a_1^4\right|^2 + \frac{289}{64}\left|a_4^4\right|^2 + \frac{225}{64} - \frac{225}{64}\left|a_4^4\right|^2 + 9\left|a_2^4\right|^2 + \frac{169}{36}\left|a_3^4\right|^2 + \frac{100}{9} - \frac{25}{36}\left|a_3^4\right|^2 \tag{C.135}$$

$$\Leftrightarrow 17\left|a_1^4\right|\left|a_4^4\right| + 13\left|a_2^4\right|\left|a_3^4\right| \leqslant \frac{8425}{576} + 16\left|a_1^4\right|^2 + 9\left|a_2^4\right|^2 + 4\left|a_3^4\right|^2 + \left|a_4^4\right|^2 \tag{C.136}$$

$$\Rightarrow 17\left|a_1^4\right|\left|a_4^4\right| + 13\left|a_2^4\right|\left|a_3^4\right| \leqslant 25 + 16\left|a_1^4\right|^2 + 9\left|a_2^4\right|^2 + 4\left|a_3^4\right|^2 + \left|a_4^4\right|^2 \tag{C.137}$$

which proves the inequality for $j = 5$.

□

C.4 The classical multisegment Burg algorithm

The aim of the multisegment Burg algorithm presented in 18 is to estimate the coefficients of the autoregressive model of N temporal signals $\left(\mathbf{u}^0, \ldots, \mathbf{u}^{N-1}\right)$ assuming these signals have the same autoregressive coefficients and the same lenght n.

Algorithm 18 The multisegment Burg algorithm

Initialization:

$$f_k^{0,l} = b_k^{0,l} = u_k^l \quad k = 0, ..., n-1;\ l = 0, ..., N-1 \tag{C.138}$$

$$p_0 = \frac{1}{nN}\sum_{l=0}^{N-1}\sum_{k=0}^{n-1}\left|u_k^l\right|^2 \tag{C.139}$$

for $j = 1, ..., M$: **do**

$$\mu_j = -\frac{2\sum_{l=0}^{N-1}\sum_{k=j}^{n-1} f_k^{j-1,l} b_{k-1}^{j-1,l*}}{\sum_{l=0}^{N-1}\sum_{k=j}^{n-1}\left|f_k^{j-1,l}\right|^2 + \left|b_{k-1}^{j-1,l}\right|^2} \tag{C.140}$$

where:

$$\left\{\begin{array}{rcll} f_k^{j,l} & = & f_k^{j-1,l} + \mu_j\, b_{k-1}^{j-1,l} & k = j, ..., n-1;\ l = 0, ..., N-1 \\ b_k^{j,l} & = & b_{k-1}^{j-1,l} + \mu_j^*\, f_k^{j-1,l} & k = j, ..., n-1;\ l = 0, ..., N-1 \end{array}\right. \tag{C.141}$$

end for

return $(p_0, \mu_1, ..., \mu_{n-1})$

Here, the function to minimize is the sum of all forward and backward prediction errors:

$$E_j = \frac{1}{N}\sum_{l=0}^{N-1} E_j^l \tag{C.142}$$

$$= \frac{1}{N}\sum_{l=0}^{N-1}\frac{1}{2(n-j)}\sum_{k=j}^{n-1}\left|f_k^{j,l}\right|^2 + \left|b_k^{j,l}\right|^2$$

Here l denotes the index of the temporal signal, j is the order of the autoregressive model and k is the index of the error ($i \leqslant k \leqslant n-1$).

Property 42. *The total error E_j is minimized for:*

$$\mu_j = -\frac{2\sum_{l=0}^{N-1}\sum_{k=j}^{n-1} f_k^{j-1,l} b_{k-1}^{j-1,l*}}{\sum_{l=0}^{N-1}\sum_{k=j}^{n-1}\left|f_k^{j-1,l}\right|^2 + \left|b_{k-1}^{j-1,l}\right|^2} \tag{C.143}$$

Proof. The ideas of this proof are exactly the same than those of the proof of Property 37, we just add a sum over the number of time series N. □

Another possible method to estimate the reflection coefficients of several time series corresponding to several realizations of the same stochastic process is to use the one-dimensional classical Burg algorithm or regularized Burg algorithm, then to approximate the mean of the reflection coefficients in the Poincare polydisk $\mathbb{D}^{n-1}$ using the Kähler metric.

C.5 The regularized multisegment Burg algorithm

We present the regularized multisegment Burg algorithm in Algorithm 19.

Algorithm 19 The regularized multisegment Burg algorithm

Initialization:

$$f_k^{0,l} = b_k^{0,l} = u_k^l \quad k = 0, ..., n-1;\ l = 0, ..., N-1 \tag{C.144}$$

$$a_k^0 = 1 \quad k = 0, ..., n-1 \tag{C.145}$$

$$p_0 = \frac{1}{nN}\sum_{l=0}^{N-1}\sum_{k=0}^{n-1}\left|u_k^l\right|^2 \tag{C.146}$$

for $j = 1, ..., M$: **do**

$$\mu_j = -\frac{\frac{2}{N(n-j)}\sum_{l=0}^{N}\sum_{k=j}^{n-1} f_k^{j-1,l}\, b_{k-1}^{j-1,l*} + 2\sum_{k=1}^{j-1}\beta_k^j\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{N(n-j)}\sum_{l=0}^{N}\sum_{k=j}^{n-1}\left|f_k^{j-1,l}\right|^2 + \left|b_{k-1}^{j-1,l}\right|^2 + 2\sum_{k=0}^{j-1}\beta_k^j\,\left|a_k^{j-1}\right|^2} \tag{C.147}$$

where:

$$\beta_k^j = \gamma(2\pi)^2(k-j)^2 \tag{C.148}$$

$$\begin{cases} a_k^j &= a_k^{j-1} + \mu_j\, a_{j-k}^{j-1*} \quad k = 1, ..., j-1 \\ a_j^j &= \mu_j \end{cases} \tag{C.149}$$

$$\begin{cases} f_k^{j,l} &= f_k^{j-1,l} + \mu_j\, b_{k-1}^{j-1,l} \quad k = j, ..., n-1; l = 0, ..., N-1 \\ b_k^{j,l} &= b_{k-1}^{j-1,l} + \mu_j^*\, f_k^{j-1,l} \quad k = j, ..., n-1; l = 0, ..., N-1 \end{cases} \tag{C.150}$$

end for

return $(p_0, \mu_1, ..., \mu_{n-1})$

We would like to justify the expression of the reflection coefficient μ_j in Equation (C.147).
Here, the function to minimize is the sum of all forward and backward prediction errors plus a regularization factor:

$$\begin{aligned} E_j &= \frac{1}{N}\sum_{l=0}^{N-1} E_j^l + G_j^l \\ &= \frac{1}{N}\sum_{l=0}^{N-1}\frac{1}{2(n-j)}\sum_{k=j}^{n-1}\left|f_k^{j,l}\right|^2 + \left|b_k^{j,l}\right|^2 + \gamma\,(2\pi)^2\sum_{k=1}^{j} k^2\left|a_k^{j,l}\right|^2 \end{aligned} \tag{C.151}$$

Here l denotes the index of the time series, j is the order of the autoregressive model and k is the index of the error ($i \leqslant k \leqslant n-1$).

Property 43. *The total error E_j is minimized for:*

$$\mu_j = -\frac{\frac{2}{N(n-j)} \sum_{l=0}^{N} \sum_{k=j}^{n-1} f_k^{j-1,l}\, b_{k-1}^{j-1,l*} + 2 \sum_{k=1}^{j-1} \beta_k^j\, a_k^{j-1}\, a_{j-k}^{j-1}}{\frac{1}{N(n-j)} \sum_{l=0}^{N} \sum_{k=j}^{n-1} \left| f_k^{j-1,l} \right|^2 + \left| b_{k-1}^{j-1,l} \right|^2 + 2 \sum_{k=0}^{j-1} \beta_k^j \left| a_k^{j-1} \right|^2} \tag{C.152}$$

Proof. The ideas of this proof are exactly the same than those of the proof of Property 40, we just add a sum over the number of time series N. □

Appendix D

Sectional curvature in Riemannian manifolds

In this appendix, we will recall some useful properties on the sectional curvature of Riemannian manifolds. We will in particular recall the links between sectional curvature and the metric.

Contents

D.1 Sectional curvature of a Riemannian manifold

The following theorem relates the expression of the metric element ds^2 and the sectional curvature. This theorem is detailed in the book of François Rouvière [31].

Let M be a general manifold and m a point of M. We choose an orthonormal basis of T_mM whose two first vectors E_1 and E_2 belongs to the plan σ. For a tangent vector X close to zero, we have $d\left(m, \exp_m\left(X\right)\right) = \|X\|$. We use the coordinate system corresponding to the normal coordinates around m, given by the reciprocal application of $(x_1, \ldots, x_n) \mapsto \exp_m\left(\sum_i x_i E_i\right)$. We define $\exp_m(\sigma)$ as the set of points of the form $\exp_m\left(x_1E_1 + x_2E_2\right)$. On $exp_m(\sigma)$, we have by definition $x_3 = \ldots = x_n = 0$.

Theorem 7. *On $\exp_m(\sigma)$, the expression of the metric in normal coordinates has the following expression:*

$$\begin{aligned} ds^2 =& dx_1^2 + dx_2^2 - \frac{1}{3}K(\sigma)\left(x_1dx_2 - x_2dx_1\right)^2 \\ &+ O\left(\left(|x_1| + |x_2|\right)^3\right)dx_1^2 + O\left(\left(|x_1| + |x_2|\right)^3\right)dx_1dx_2 + O\left(\left(|x_1| + |x_2|\right)^3\right)dx_2^2 \end{aligned} \tag{D.1}$$

where $K(\sigma) = K\left(E_1, E_2\right)$ denotes the sectional curvature of the plan σ defined by E_1 and E_2.

Using the polar coordinate system related to x_1 and x_2 by the equations $x_1 = r\cos\left(\theta\right)$ and $x_2 = r\sin\left(\theta\right)$, we obtain:

$$ds^2 = dr^2 + \left(1 - \frac{1}{3}K(\sigma)r^2\right)r^2d\theta^2 + O\left(r^5\right)dr^2 + O\left(r^5\right)drd\theta + O\left(r^5\right)d\theta^2 \tag{D.2}$$

We can also compute the sectional curvature using the length of the hypotenuse of infinitesimal right triangles.

D.2 Curvature and metric scalar multiplication

In this section, we will show that if we multiply the metric ds^2 of a Riemannian manifold by a constant $c > 0$, then the sectional curvature K is divided by c.

Property 44. *Let E be a space endowed with two Riemannian metrics ds^2 and $\widetilde{ds}^2$ such that $\widetilde{ds}^2 = cds^2$ with $c > 0$. Let K be the sectional curvature associated with the Riemannian manifold $\left(E, ds^2\right)$ and $\widetilde{K}$ the sectional curvature associated with the Riemannian manifold $\left(E, \widetilde{ds}^2\right)$. Then, we have:*

$$\widetilde{K} = \frac{K}{c} \tag{D.3}$$

Proof. We use the coordinate system corresponding to the normal coordinates around m, given by the reciprocal application of $(x_1, \ldots, x_n) \mapsto \exp_m \left(\sum_i x_i E_i\right)$. We define $\exp_m(\sigma)$ as the set of points of the form $\exp_m \left(x_1 E_1 + x_2 E_2\right)$. On $exp_m(\sigma)$, we have by definition $x_3 = \ldots = x_n = 0$.

We recall Theorem 7 given in [31]:

On $exp_m(\sigma)$, the expression of the metric in normal coordinates has the following expression:

$$\begin{aligned} ds^2 =& dx_1^2 + dx_1^2 - \frac{1}{3} K(\sigma) \left(x_1 dx_2 - x_2 dx_1\right)^2 \\ &+ O\left(\left(|x_1| + |x_2|\right)^3\right) dx_1^2 + O\left(\left(|x_1| + |x_2|\right)^3\right) dx_1 dx_2 + O\left(\left(|x_1| + |x_2|\right)^3\right) dx_2^2 \end{aligned} \tag{D.4}$$

where $K(\sigma) = K\left(E_1, E_2\right)$ denotes the sectional curvature of the plan σ defined by E_1 and E_2.

If we denote:

$$\widetilde{x}_1 = \sqrt{c} x_1 \tag{D.5}$$

$$\widetilde{x}_2 = 2 x_2, \tag{D.6}$$

then we have, by multiplying Equation (D.1) by the constant c:

$$\begin{aligned} \widetilde{ds}^2 =& \widetilde{dx}_1^2 + \widetilde{dx}_1^2 - \frac{1}{3} \frac{K(\sigma)}{c} \left(\widetilde{x}_1 \widetilde{dx}_2 - \widetilde{x}_2 \widetilde{dx}_1\right)^2 \\ &+ O\left(\left(|\widetilde{x}_1| + |\widetilde{x}_2|\right)^3\right) \widetilde{dx}_1^2 + O\left(\left(|\widetilde{x}_1| + |\widetilde{x}_2|\right)^3\right) \widetilde{dx}_1 \widetilde{dx}_2 + O\left(\left(|\widetilde{x}_1| + |\widetilde{x}_2|\right)^3\right) \widetilde{dx}_2^2. \end{aligned} \tag{D.7}$$

By identification, we have:

$$\widetilde{K}\left(\sigma\right) = \frac{K\left(\sigma\right)}{c}. \tag{D.8}$$

□

D.3 Sectional curvature and product manifold

Lemma 1. *Let M, N be two Riemannian manifolds such that the sectional curvatures verify*

$$-C \leqslant K_M, K_N \leqslant 0, \tag{D.9}$$

where $C \geqslant 0$ is a constant. Then the sectional curvatures of the product $M \times N$ also verify

$$-C \leqslant K_{M \times N} \leqslant 0. \tag{D.10}$$

Proof. This lemma is proved in the work of Le Yang [82]. □

D.4 Infinitesimal right triangles and sectional curvature in Riemannian manifolds

In this section, we study the relation between the length of the hypotenuse of an infinitesimal right triangle and the sectional curvature.

Theorem 8. *Let U be a relatively compact convex open set of a complete Riemannian manifold M. There exists a constant $C_U > 0$ such that for any triangle OAB of U rectangle in 0, we have the following inequality:*

$$\left| d^2\left(A, B\right) - d^2\left(O, A\right) - d^2\left(O, B\right) + \frac{1}{3} K\left(\sigma\right) d^2\left(O, A\right) d^2\left(O, B\right) \right| \leqslant C_U \left(d\left(O, A\right) + d\left(O, B\right)\right)^5 \tag{D.11}$$

In the works of Andrei A. Agrachev, Davide Barilari and Luca Rizzi [4], this theorem is stated in the particular case where point O is fixed and the directions of points A and B starting from O are fixed. The result stated in their works is said to be due to Loeper and Villani. It is here stated and proved independently of these previous works, in a more general context. In order to prove Theorem 8, we introduce the coordinate system and the notations with which we are going to work.

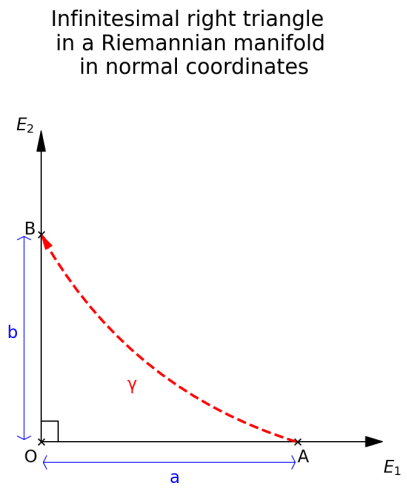

Figure D.1: An infinitesimal right triangle in a Riemannian manifold in normal coordinates

Choice of coordinate system and notations Let U be a relatively compact convex open set of a complete Riemannian manifold M. We denote by $\bar{U}$ the adherence of U in the manifold M. Note that $\bar{U}$ is a compact of M. Let O, A, and B be three points of U forming a right triangle in O.

We choose a normal coordinate system of origin O whose orthonormal basis $(E_1, \ldots, E_n)$ is such that the point A has coordinates $(a, 0, 0, \ldots, 0)$ and the point B has coordinates $(0, b, 0, \ldots, 0)$.

The graphic D.1 represents the right triangle OAB in the orthonormal coordinate system $(E_1, \ldots, E_n)$ centered at the point O.

We denote by σ the plane defined by E_1 and E_2.

We now study the variations of the squared length of the hypotenuse $d^2(A, B)$ when the points O, A and B vary in U by continuing to form a right triangle in O. The orthonormal coordinate system $(E_1, \ldots, E_n)$ will then vary to remain centered at the point O and the vectors E_1 and E_2 will then be determined by the position of the points A and B.

We denote γ_{XY} the geodesic starting at X at $t = 0$ and arriving at Y at $t = 1$. We denote $\gamma := \gamma_{AB}$ the geodesic that starts at point A of coordinates $(a, 0, 0, \ldots, 0)$ at time $t = 0$ and ends at point B of coordinates $(0, b, 0, \ldots, 0)$ at $t = 1$.

We denote $l(c)$ the length of the curve c.

Preliminary 1. *We first show that* $\forall t \in [0, 1]$ *we have* $\|x(t)\| \leqslant |a| + |b|$, *where* $\|x(t)\| := l\left(\gamma_{O\gamma(t)}\right)$.

Since $x(t)$ represents the coordinates of $\gamma(t)$ in a normal coordinate system, we notice that $\|x(t)\| := l\left(\gamma_{O\gamma(t)}\right)$ also corresponds to the Euclidean norm of de $x(t)$ in the coordinate system $(E_1, \ldots, E_n)$.

Proof. The straight lines starting at the origin are geodesics, therefore we have as mentioned previously $l(\gamma_{OA}) = |a|$ and $l(\gamma_{OB}) = |b|$.

According to the triangular inequality, we have:

$$\begin{aligned} l(\gamma) := l(\gamma_{AB}) &\leqslant l(\gamma_{AO}) + l(\gamma_{OB}) && \text{(D.12)} \\ &= l(\gamma_{OA}) + l(\gamma_{OB}) && \text{(D.13)} \\ &= |a| + |b| && \text{(D.14)} \end{aligned}$$

Thus, we can bound the sum of the lengths of the sides of the triangle OAB:

$$l(\gamma_{OA}) + l(\gamma_{AB}) + l(\gamma_{BO}) \leqslant 2(|a| + |b|) \tag{D.15}$$

Let $\gamma(t)$ be a point on the geodesic leading from point A to point B.

We have:

$$l(\gamma_{OA}) + l\left(\gamma_{A\gamma(t)}\right) + l\left(\gamma_{\gamma(t)B}\right) + l(\gamma_{BO}) = l(\gamma_{OA}) + l(\gamma_{AB}) + l(\gamma_{BO}) \leqslant 2(|a| + |b|) \tag{D.16}$$

Therefore, at least one of the following two inequalities is true:

$$l\left(\gamma_{OA}\right)+l\left(\gamma_{A\gamma(t)}\right)\leqslant|a|+|b|\quad\text{or}\quad l\left(\gamma_{\gamma(t)B}\right)+l\left(\gamma_{BO}\right)\leqslant|a|+|b| \tag{D.17}$$

From the triangular inequality, we have:

$$l\left(\gamma_{O\gamma(t)}\right)\leqslant l\left(\gamma_{OA}\right)+l\left(\gamma_{A\gamma(t)}\right)\quad\text{and}\quad l\left(\gamma_{O\gamma(t)}\right)\leqslant l\left(\gamma_{\gamma(t)B}\right)+l\left(\gamma_{BO}\right) \tag{D.18}$$

From equations (D.17) and (D.18), we have that:

$$l\left(\gamma_{O\gamma(t)}\right)\leqslant|a|+|b| \tag{D.19}$$

Hence we have: $\|x(t)\|\leqslant|a|+|b|\quad\forall t\in[0,1]$. □

The following preliminary result will be used to prove preliminaries 3, 4 and 5.

We denote by $O\left(M\right)$ the bundle of orthonormal bases of the Riemannian manifold M. The bundle of orthonormal bases of the compact $\bar{U}$ will be denoted $O\left(\bar{U}\right)$. We denote by u an element of $O\left(M\right)$. Implicitly the element u defines an orthonormal basis centered at the point O.

Preliminary 2. *There exists $\delta>0$ such that the application $(u,x)\mapsto g_{i,j}\left(u,x\right)$ is an application of class C^{∞} on $O\left(M\right)\times B\left(O,\delta\right)$, where $B\left(O,\delta\right)$ is the open ball with center point O and radius δ.*

Proof. Let (M,g) be a Riemannian manifold. Let U be an open subset of M and $\phi:U\to\phi\left(U\right)\subset\mathbb{R}^{n}$ a local map. We set $\psi=\phi^{-1}:V=\phi\left(U\right)\to U$. Let $(i,j)\in[\![1,n]\!]$, for $x\in V$ the coefficient $g_{i,j}(x)$ of the metric is defined by:

$$g_{i,j}(x):=g\left(\psi(x)\right)\left(T_{x}\psi\left(e_{i}\right),T_{x}\psi\left(e_{j}\right)\right) \tag{D.20}$$

where by definition $e_{i}=(0,\ldots,0,1,0,\ldots,0)$ with the number 1 placed in the i^{th} position.
By definition the vector $T_{x}\psi\left(e_{i}\right)\in T_{\psi(x)}M$ is defined by:

$$T_{x}\psi\left(e_{i}\right)=\left.\frac{d}{da}\right|_{a=0}\psi\left(x+ae_{i}\right) \tag{D.21}$$

We now study the particular case of an exponential map centered at the point $O\in M$. There exists $\delta>0$ such that this exponential map is injective on the open ball $B\left(O,\delta\right)$. We set $U=B\left(O,\delta\right)$, we have $V=B\left(O,\delta\right)\subset\mathbb{R}^{n}$ and:

$$\psi(x)=\exp_{O}\left(u(x)\right)\quad\forall x\in V \tag{D.22}$$

where u is the orthonormal basis centered at the point O associated with the choice of the exponential map.
The element $u\in O\left(M\right)$ can also be seen as a linear map:

$$\begin{aligned}u:&\mathbb{R}^{n}\to T_{O}M\\&\left(x^{1},\ldots,x^{d}\right)\mapsto\sum_{i=1}^{d}x^{i}u\left(e_{i}\right).\end{aligned} \tag{D.23}$$

From Equation (D.22) and by linearity of u, we have:

$$\psi\left(x+ae_{i}\right)=\exp_{O}\left(u(x)+au\left(e_{i}\right)\right). \tag{D.24}$$

According to Equation (D.21), this gives:

$$T_{x}\psi\left(e_{i}\right)=T_{u(x)}\exp_{O}\left(\cdot\right)\left(u\left(e_{i}\right)\right) \tag{D.25}$$

where $T_{u(x)}\exp_{O}\left(\cdot\right)\left(u\left(e_{i}\right)\right)$ is the tangent map T at the point $u(x)$ of the function $\exp_{O}\left(\cdot\right)$ in the direction of the vector $u\left(e_{i}\right)$.
Coming back to Equation (D.20), we obtain:

$$g_{i,j}\left(u,x\right)=g\left(exp_{O}\left(u(x)\right)\right)\left(T_{u(x)}\exp_{O}\left(\cdot\right)\left(u\left(e_{i}\right)\right),T_{u(x)}\exp_{O}\left(\cdot\right)\left(u\left(e_{j}\right)\right)\right). \tag{D.26}$$

According to Equation (D.26), the application $(u,x)\mapsto g_{i,j}\left(u,x\right)$ is of class C^{∞} sur $O\left(M\right)\times B\left(O,\delta\right)$.

□

We recall the Hadamard's lemma which will be used in the proofs of the preliminaries 3, 4 and 5.

Lemma 2. *Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be a function of class C^p with $p \geqslant 1$. Then, for all $a = (a_1, \ldots, a_n) \in \mathbb{R}^n$, there exists functions $g_1, \ldots, g_n$, of class C^{p-1} such that for all $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$,*

$$f(x) = f(a) + \sum_{i=1}^{n} (x_i - a_i)\, g_i(x). \tag{D.27}$$

We necessarily have $g_i(a) = \frac{\partial f}{\partial x_i}(a)$. The fonctions g_i are not unique. The following functions are solutions of Equation (D.27):

$$g_i(x) = \int_0^1 \frac{\partial f}{\partial x_i}\left(a + t\,(x-a)\right) dt. \tag{D.28}$$

Preliminary 3. *There exists $\delta > 0$ such that we can write the metric tensor g the following way:*

$$g_{i,j}\,(u,x) = \delta_{i,j} + \sum_{k,l} h_{i,j,k,l}\,(u,x)\, x^k x^l \tag{D.29}$$

where the functions $h_{i,j,k,l}$ are functions of class C^∞ on $O\,(M) \times B\,(O,\delta)$.

Proof. According to the book written by François Rouvière [31], we have in normal coordinates:

$$g_{i,j}(x) = \delta_{i,j} + O\left(\|x\|^2\right). \tag{D.30}$$

We will now use Hadamard's lemma 2 twice.
Let $(i,j) \in [\![1,n]\!]^2$. We set: $f_{i,j} := g_{i,j} - \delta_{i,j}$.
Since the function $(u,x) \mapsto g_{i,j}\,(u,x)$ is of class C^∞ sur $O\,(M) \times B\,(O,\delta)$ according to Preliminary 2, the function $(u,x) \mapsto f_{i,j}\,(u,x)$ is also of class C^∞ on $O\,(M) \times B\,(O,\delta)$.
According to Equation (D.30), we have $f_{i,j}(0) = 0$ and $\frac{\partial f_{i,j}}{\partial x^k}(0) = 0 \;\; \forall k \in [\![1,n]\!]$.
By applying Hadamard's lemma to the function f at the point 0, we can write:

$$f_{i,j}(u,x) = \sum_k h_{i,j,k}(u,x) x^k \tag{D.31}$$

with $h_{i,j,k}(u,x) = \int_0^1 \frac{\partial g_{i,j}}{\partial x^k}\,(u,tx)\,dt$. The functions $(u,x) \mapsto h_{i,j,k}(u,x)$ are functions of class C^∞ on $O\,(M) \times B\,(O,\delta)$.
Moreover, we have according to Hadamard's lemma:

$$h_{i,j,k}(0) = \frac{\partial f_{i,j}}{\partial x^k}(0) = 0 \quad \forall k \in [\![1,n]\!]. \tag{D.32}$$

By applying Hadamard's lemma to the functions $h_{i,j,k}$ at the point 0, we can write:

$$h_{i,j,k}(x) = \sum_l h_{i,j,k,l}(x) x^l \tag{D.33}$$

with $h_{i,j,k,l}(u,x) = \int_0^1 \frac{\partial h_{i,j,k}}{\partial x^l}\,(u,tx)\,dt$. The functions $(u,x) \mapsto h_{i,j,k,l}(u,x)$ are functions of class C^∞ on $O\,(M) \times B\,(O,\delta)$.
According to Equations (D.31) and (D.33), we have:

$$f_{i,j}(x) = \sum_{k,l} h_{i,j,k,l}(x) x^k x^l. \tag{D.34}$$

Since $f_{i,j} := g_{i,j} - \delta_{i,j}$, we have proven that:

$$g_{i,j}\,(u,x) = \delta_{i,j} + \sum_{k,l} h_{i,j,k,l}\,(u,x)\, x^k x^l \tag{D.35}$$

where the functions $h_{i,j,k,l}$ are functions of class C^∞ on $O\,(M) \times B\,(O,\delta)$. □

Preliminary 4. *There exists $\delta > 0$ such that we can write the Christoffel symbols Γ as follows:*

$$\Gamma^k_{i,j}\,(u,x) = \sum_l \tilde{h}^k_{i,j,l}\,(u,x)\, x^l \tag{D.36}$$

where the functions $\tilde{h}^k_{i,j,l}$ are functions of class C^∞ on $O\,(M) \times B\,(O,\delta)$.

Proof. We can express the Christoffel symbols $\Gamma_{i,j}^k$ with respect to the metric tensor g as follows:

$$\Gamma_{i,j}^k = \frac{1}{2}\sum_{l=1}^{n} g^{k,l}\left(\partial_i g_{l,j} + \partial_j g_{i,l} - \partial_l g_{i,j}\right). \tag{D.37}$$

There exists $\delta > 0$ such that the functions $(u,x) \mapsto g_{i,j}(u,x)$ are functions of class C^∞ on $O(M) \times B(O,\delta)$ according to Preliminary 2.

The Christoffel symbols $\Gamma_{i,j}^k$ are therefore also functions of class C^∞ with respect to (u,x) on $O(M) \times B(O,\delta)$.

Then, we apply Hadamard's lemma 2 to the function $\Gamma_{i,j}^k$ at the point 0.

We can write:

$$\Gamma_{i,j}^k(u,x) = \Gamma_{i,j}^k(u,0) + \sum_l \tilde{h}_{i,j,l}^k(u,x)\, x^l \tag{D.38}$$

with $\tilde{h}_{i,j,l}^k(u,x) = \int_0^1 \frac{\partial \Gamma_{i,j}^k}{\partial x^l}(u,tx)\, dt$. The fonctions $(u,x) \mapsto \tilde{h}_{i,j,l}^k(u,x)$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

According to the book by François Rouvière [31], we have $\Gamma_{i,j}^k(u,0) = 0$. We therefore have:

$$\Gamma_{i,j}^k(u,x) = \sum_l \tilde{h}_{i,j,l}^k(u,x)\, x^l \tag{D.39}$$

where the functions $\tilde{h}_{i,j,l}^k$ are functions of class C^∞ on $O(M) \times B(O,\delta)$. □

Preliminary 5. *There exists $\delta > 0$ such that we can write the element ds^2 as follows:*

$$ds^2(u,x) = \sum_i \left(dx^i\right)^2 + \frac{1}{3}\sum_{i,j,k,l} R_{i,j,k,l} x^i x^k dx^j dx^l + \sum_{i,j,k,l,m} \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)\, x^i x^j x^k dx^l dx^m \tag{D.40}$$

where the functions $\tilde{\tilde{h}}_{i,j,k,l,m}$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

Proof. The following theorem is given in François Rouvière's book [31]:

$$ds^2 = \sum_i \left(dx^i\right)^2 + \frac{1}{3}\sum_{i,j,k,l} R_{i,j,k,l} x^i x^k dx^j dx^l + \dots \tag{D.41}$$

where the $\dots$ are terms of order $\geqslant 3$ with respect to the x^i.

We will now show that there exists $\delta > 0$ such that the function $(u,x) \mapsto ds^2(u,x)$ is of class C^∞ on $O(M) \times B(0,\delta)$.

We have the following relation between the infinitesimal square length ds^2 and the metric tensor g:

$$ds^2(u,x) = \sum_{i,j=1}^{n} g_{i,j}(u,x) dx^i dx^j. \tag{D.42}$$

There exists $\delta > 0$ such that the functions $(u,x) \mapsto g_{i,j}(u,x)$ are of class C^∞ on $O(M) \times B(0,\delta)$ according to Preliminary 2. The function $(u,x) \mapsto ds^2(u,x)$ is therefore of class C^∞ on $O(M) \times B(0,\delta)$.

We will now apply Hadamard's lemma three times.

We set $f := ds^2 - \sum_i \left(dx^i\right)^2 + \frac{1}{3}\sum_{i,j,k,l} R_{i,j,k,l} x^i x^k dx^j dx^l$.

According to Equation (D.41), we have $f(0) = 0$, $\frac{\partial f}{\partial x^k}(0) = 0 \ \ \forall k \in [\![1,n]\!]$ and $\frac{\partial^2 f}{\partial x^j x^k}(0) = 0 \ \ \forall (j,k) \in [\![1,n]\!]^2$.

By applying Hadamard's lemma to the function f at the point 0, we can write:

$$f(u,x) = \sum_k \tilde{\tilde{h}}_{k,l,m}(u,x)\, x^k dx^l dx^m \tag{D.43}$$

with $\tilde{\tilde{h}}_{k,l,m}(u,x) = \int_0^1 \frac{\partial f}{\partial x^k}(u,tx)\, dt$. The functions $(u,x) \mapsto \tilde{\tilde{h}}_{k,l,m}(u,x)$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

According to Hadamard's lemma, we have:

$$\tilde{\tilde{h}}_{k,l,m}(u,0) = \frac{\partial f}{\partial x^k}(u,0) = 0. \tag{D.44}$$

Applying Hadamard's lemma to the functions $\tilde{\tilde{h}}_{k,l,m}$ at the point 0, we can write:

$$\tilde{\tilde{h}}_{k,l,m}(u,x) = \sum_j \tilde{\tilde{h}}_{j,k,l,m}(u,x)\, x^j \tag{D.45}$$

with $\tilde{\tilde{h}}_{j,k,l,m}(u,x) = \int_0^1 \frac{\partial \tilde{\tilde{h}}_{k,l,m}}{\partial x^j}(u,tx)\,dt$. The fonctions $(u,x) \mapsto \tilde{\tilde{h}}_{j,k,l,m}(u,x)$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

According to Hadamard's lemma, we have:

$$\tilde{\tilde{h}}_{j,k,l,m}(u,0) = \frac{\partial \tilde{\tilde{h}}_{k,l,m}}{\partial x^j}(u,0). \tag{D.46}$$

According to the definition of $\tilde{\tilde{h}}_{k,l,m}$, we also have:

$$\frac{\partial \tilde{\tilde{h}}_{k,l,m}}{\partial x^j}(u,x) = \int_0^1 \frac{\partial^2 f}{\partial x^j \partial x^k}(u,tx)\,dt \tag{D.47}$$

Hence, we have: $\frac{\partial \tilde{\tilde{h}}_{k,l,m}}{\partial x^j}(u,0) = \frac{\partial^2 f}{\partial x^j \partial x^k}(u,0) = 0$. From Equation (D.46), we therefore have $\tilde{\tilde{h}}_{j,k,l,m}(u,0) = 0$.

We now apply Hadamard's lemma to the function $\tilde{\tilde{h}}_{j,k,l,m}$ at the point 0. We can write:

$$\tilde{\tilde{h}}_{j,k,l,m}(u,x) = \sum_{i,j,k,l,m} \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)\,x^i \tag{D.48}$$

with $\tilde{\tilde{h}}_{i,j,k,l,m}(u,x) = \int_0^1 \frac{\partial \tilde{\tilde{h}}_{j,k,l,m}}{\partial x^i}(u,tx)\,dt$. The fonctions $(u,x) \mapsto \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

Using equations (D.43), (D.45) and (D.48), we obtain:

$$f(u,x) = \sum_{i,j,k,l,m} \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)\,x^i x^j x^k dx^l dx^m. \tag{D.49}$$

Since $f := ds^2 - \sum_i \left(dx^i\right)^2 + \frac{1}{3}\sum_{i,j,k,l} R_{i,j,k,l} x^i x^k dx^j dx^l$, we have:

$$ds^2(u,x) = \sum_i \left(dx^i\right)^2 + \frac{1}{3}\sum_{i,j,k,l} R_{i,j,k,l} x^i x^k dx^j dx^l + \sum_{i,j,k,l,m} \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)\,x^i x^j x^k dx^l dx^m \tag{D.50}$$

where the functions $\tilde{\tilde{h}}_{i,j,k,l,m}$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

□

We now come back to the proof of Theorem 8.

Proof. We denote δ the positive real number considered in Preliminary 2 and used in Preliminaries 3, 4 and 5.

We set $\epsilon = \min\left(R, \frac{1}{2}\delta\right)$ where R is the minimum of the injectivity radii on the compact $\bar{U}$.

We first consider the case where $d(O,A) \geqslant \epsilon$ or $d(O,B) \geqslant \epsilon$.

Then, by choosing

$$C_U = \frac{\sup_{(O,A,B)\in U^3} \left|d^2(A,B) - d^2(O,A) - d^2(O,B) + \frac{1}{3}K(\sigma)\,d^2(O,A)\,d^2(O,B)\right|}{\epsilon^5} \tag{D.51}$$

the inequality (D.11) stated in Theorem 8 is verified.

We now consider that both $d(O,A) < \epsilon$ and $d(O,B) < \epsilon$.

Then, according to Preliminary 1, we have:

$$\|x(t)\| \leqslant |a| + |b| \leqslant \epsilon + \epsilon \leqslant \frac{\delta}{2} + \frac{\delta}{2} = \delta \quad \forall t \in [0,1]. \tag{D.52}$$

We can therefore use the results of Preliminaries 2, 3, 4 and 5.

We denote by $\|x'(t)\|_{x(t)}$ the norm of the velocity vector of the geodesic γ at the point $\gamma(t)$ of coordinates $x(t)$.

We have:

$$\left\|x'(t)\right\|^2_{x(t)} = \sum_{i,j} g_{i,j}(u,x(t))\left(x^i\right)'(t)\left(x^j\right)'(t). \tag{D.53}$$

According to Preliminary 3, we can write:

$$g_{i,j}(u,x) = \delta_{i,j} + \sum_{k,l} h_{i,j,k,l}(u,x)\,x^k x^l \tag{D.54}$$

where the functions $h_{i,j,k,l}$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.

We therefore have:

$$\left\|x'(t)\right\|_{x(t)}^2 = \sum_{i,j}\left(\delta_{i,j} + \sum_{k,l} h_{i,j,k,l}\left(u,x\right)x^k x^l\right)\left(x^i\right)'(t)\left(x^j\right)'(t) \tag{D.55}$$

$$= \sum_{i,j}\delta_{i,j}\left(x^i\right)'(t)\left(x^j\right)'(t) + \sum_{i,j,k,l} h_{i,j,k,l}\left(u,x\right)x^k x^l\left(x^i\right)'(t)\left(x^j\right)'(t) \tag{D.56}$$

$$= \left\|x'(t)\right\|_0^2 + \sum_{i,j,k,l} h_{i,j,k,l}\left(u,x\right)x^k x^l\left(x^i\right)'(t)\left(x^j\right)'(t). \tag{D.57}$$

Hence, we have:

$$\left\|x'(t)\right\|_0^2 = \left\|x'(t)\right\|_{x(t)}^2 - \sum_{i,j,k,l} h_{i,j,k,l}\left(u,x\right)x^k x^l\left(x^i\right)'(t)\left(x^j\right)'(t) \tag{D.58}$$

$$= l^2(\gamma) - \sum_{i,j,k,l} h_{i,j,k,l}\left(u,x\right)x^k x^l\left(x^i\right)'(t)\left(x^j\right)'(t). \tag{D.59}$$

According to Equation (D.12), we have:

$$l(\gamma) \leqslant \left(|a| + |b|\right) \tag{D.60}$$

Therefore, we have:

$$l^2(\gamma) \leqslant \left(|a| + |b|\right)^2. \tag{D.61}$$

We also have:

$$\sum_{i,j,k,l} h_{i,j,k,l}\left(u,x\right)x^k x^l\left(x^i\right)'(t)\left(x^j\right)'(t) \leqslant C_1\left(|a| + |b|\right)^2. \tag{D.62}$$

Hence, according to Equation (D.59):

$$\left\|x'(t)\right\|_0^2 \leqslant C_2\left(|a| + |b|\right)^2 \tag{D.63}$$

with $C_2 = 1 + C_1$.
We therefore have:

$$\left\|x'(t)\right\|_0 \leqslant C_3\left(|a| + |b|\right) \tag{D.64}$$

with $C_3 = \sqrt{C_2}$.
Since $\left|\left(x^i\right)'(t)\right| \leqslant \left\|x'(t)\right\|_0$, we have:

$$\left|\left(x^i\right)'(t)\right| \leqslant C_3\left(|a| + |b|\right) \quad \forall i \in [\![1,n]\!]. \tag{D.65}$$

Moreover, according to Preliminary 4, there exists $\delta > 0$ such that we can write the Christoffel symbols Γ as follows:

$$\Gamma_{i,j}^k\left(u,x\right) = \sum_l \tilde{h}_{i,j,l}^k\left(u,x\right)x^l \tag{D.66}$$

where the functions $\tilde{h}_{i,j,l}^k$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.
We can therefore write:

$$\Gamma_{j,k}^i \leqslant C_4\left(|a| + |b|\right). \tag{D.67}$$

The equation of the geodesics using Christoffel's symbols is recalled in François Rouvière's book [31]:

$$\left(x^i\right)''(t) + \sum_{j,k=1}^n \Gamma_{j,k}^i\left(x(t)\right)\left(x^j\right)'(t)\left(x^k\right)'(t) = 0. \tag{D.68}$$

Therefore we have, according to Equations (D.65) and (D.67):

$$\left|\left(x^i\right)''(t)\right| \leqslant \sum_{j,k=1}^n C_4\left(\left|x^1\right| + \left|x^2\right|\right)C_3\left(\left|x^1\right| + \left|x^2\right|\right)C_3\left(\left|x^1\right| + \left|x^2\right|\right) \tag{D.69}$$

$$\left|\left(x^i\right)''(t)\right| \leqslant C_5\left(\left|x^1\right| + \left|x^2\right|\right)^3 \text{ with } C_5 = n^2 C_4 C_3^2 \quad \forall i \in [\![1,n]\!]. \tag{D.70}$$

We now show that:

$$\left(x^1\right)'(t) = -a + \left(f^1\right)'(t) \text{ with } \left|\left(f^1\right)'(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3. \tag{D.71}$$

Since $x(t)$ is continuous on $[0,1]$ and of class C^1 on $]0,1[$, according to the mean value theorem, there exists a point $t_0 \in]0,1[$, such that:

$$\left(x^1\right)'(t_0) = \frac{\left(x^1\right)(1) - \left(x^1\right)(0)}{1-0} = -a. \tag{D.72}$$

Let $t \in [0,1]$, we have:

$$\left(x^1\right)'(t) = \left(x^1\right)'(t_0) + \int_{t_0}^{t} \left(x^1\right)''(s) ds \tag{D.73}$$

$$\left(x^1\right)'(t) = -a + \int_{t_0}^{t} \left(x^1\right)''(s) ds. \tag{D.74}$$

Therefore, by setting $\left(f^1\right)'(t) := \int_{t_0}^{t} \left(x^1\right)''(s) ds$, we have:

$$\left(x^1\right)'(t) = -a + \left(f^1\right)'(t) \tag{D.75}$$

with

$$\left|\left(f^1\right)'(t)\right| \leqslant \int_{t_0}^{t} \left|\left(x^1\right)''(s)\right| ds \leqslant C_5 \left(|a| + |b|\right)^3. \tag{D.76}$$

We show in a similar way that:

$$\left(x^2\right)'(t) = b + \left(f^2\right)'(t) \text{ with } \left|\left(f^2\right)'(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 \tag{D.77}$$

$$\left|\left(x^i\right)'(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 \quad \forall i \geqslant 3. \tag{D.78}$$

Since $\left(x^i\right)(t) = \left(x^i\right)(0) + \int_0^t \left(x^i\right)'(s) ds$, we obtain from the equations (D.71), (D.77) and (D.78):

$$\left(x^1\right)(t) = (1-t)a + f^1(t) \text{ with } \left|f^1(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 \tag{D.79}$$

$$\left(x^2\right)(t) = bt + f^2(t) \text{ with } \left|f^2(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 \tag{D.80}$$

$$\left|\left(x^i\right)(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 \quad \forall i \geqslant 3. \tag{D.81}$$

According to Preliminary 5, there exists $\delta > 0$ such that we can write the element ds^2 as follows:

$$ds^2(u,x) = \sum_i \left(dx^i\right)^2 + \frac{1}{3} \sum_{i,j,k,l} R_{i,j,k,l} x^i x^k dx^j dx^l + \sum_{i,j,k,l,m} \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)\, x^i x^j x^k dx^l dx^m \tag{D.82}$$

where the functions $\tilde{\tilde{h}}_{i,j,k,l,m}$ are functions of class C^∞ on $O(M) \times B(O,\delta)$.
For the geodesic γ, we therefore have:

$$\begin{aligned} ds^2(u,x) = {} & \left(x^1\right)'^2 dt^2 + \left(x^2\right)'^2 dt^2 + \sum_{i \geqslant 3} \left(x^i\right)'^2 dt^2 \\ & + \frac{1}{3} \sum_{(i,j,k,l) \in \{1,2\}^4} R_{i,j,k,l} x^i x^k \left(x^j\right)' \left(x^l\right)' dt^2 \\ & + \frac{1}{3} \sum_{\substack{i,j,k,l \\ i \text{ or } j \text{ or } k \text{ or } l \geqslant 3}} R_{i,j,k,l} x^i x^k \left(x^j\right)' \left(x^l\right)' dt^2 \\ & + \sum_{i,j,k,l,m} \tilde{\tilde{h}}_{i,j,k,l,m}(u,x)\, x^i x^j x^k \left(x^l\right)' \left(x^m\right)' dt^2. \end{aligned} \tag{D.83}$$

According to the equations (D.71), (D.77), (D.78), (D.79), (D.80) and (D.81), we have:

$$\begin{aligned} ds^2(u,x) = &\left(x^1\right)'^2 dt^2 + \left(x^2\right)'^2 dt^2 \\ &+ \frac{1}{3} \sum_{(i,j,k,l)\in\{1,2\}^4} R_{i,j,k,l} x^i x^k \left(x^j\right)' \left(x^l\right)' dt^2 \\ &+ f(u,x)\, dt^2. \end{aligned} \tag{D.84}$$

with $|f(u,x)| \leqslant C(|a|+|b|)^5$ for a certain constant $C > 0$.

$$ds^2 = \left[\left(x^1\right)'^2 + \left(x^2\right)'^2 + \frac{1}{3} \sum_{(i,j,k,l)\in\{1,2\}^4} R_{i,j,k,l} x^i x^k \left(x^j\right)' \left(x^l\right)' + f(u,x) \right] dt^2 \tag{D.85}$$

$$ds^2 = \left[\left(x^1\right)'^2 + \left(x^2\right)'^2 + \frac{1}{3} R_{1212} \left(x^1 \left(x^2\right)' - x^2 \left(x^1\right)' \right)^2 + f(u,x) \right] dt^2 \tag{D.86}$$

$$ds^2 = \left[\left(x^1\right)'^2 + \left(x^2\right)'^2 - \frac{1}{3} K(\sigma) \left(x^1 \left(x^2\right)' - x^2 \left(x^1\right)' \right)^2 + f(u,x) \right] dt^2 \tag{D.87}$$

Since a geodesic has constant speed, its squared length between points A and B is equal to the integral of its squared speed between times 0 and 1 when $\gamma(0) = A$ and $\gamma(1) = B$. We therefore have:

$$d^2(A,B) = \int_0^1 \left\| x'(t) \right\|^2_{x(t)} dt \tag{D.88}$$

$$= \int_0^1 \sum_{i,j} g_{i,j}(x(t)) \left(x^i\right)' \left(x^j\right)' dt \tag{D.89}$$

$$= \int_0^1 \sum_{i,j} g_{i,j}(x(t))\, dx^i dx^j \frac{1}{dt} \tag{D.90}$$

$$= \int_0^1 ds^2 \frac{1}{dt}. \tag{D.91}$$

We therefore have, using Equation (D.87):
Hence, we have:

$$d^2(A,B) = \int_0^1 \left[\left(x^1\right)'^2 + \left(x^2\right)'^2 - \frac{1}{3} K(\sigma) \left(x^1 \left(x^2\right)' - x^2 \left(x^1\right)' \right)^2 + f(u,x) \right] dt. \tag{D.92}$$

From the linearity of the integral, we have:

$$d^2(A,B) = \int_0^1 \left(x^1\right)'^2 dt + \int_0^1 \left(x^2\right)'^2 dt - \frac{1}{3} K(\sigma) \int_0^1 \left(x^1 \left(x^2\right)' - x^2 \left(x^1\right)' \right)^2 dt + \int_0^1 f(u,x)\, dt \tag{D.93}$$

Note that we have:

$$\left| \int_0^1 f(u,x)\, dt \right| \leqslant \int_0^1 |f(u,x)|\, dt \leqslant \int_0^1 C(|a|+|b|)^5 dt = C(|a|+|b|)^5 \tag{D.94}$$

We denote by f^1 and f^2 the functions such that:

$$\left(x^1\right)'(t) = -a + f^1(t) \tag{D.95}$$

$$\left(x^2\right)'(t) = b + f^2(t). \tag{D.96}$$

We have:

$$\int_0^1 \left(x^1\right)'(t) dt = x^1(1) - x^1(0) = 0 - a = -a. \tag{D.97}$$

Hence, we have:

$$\int_0^1 f^1(t)dt = \int_0^1 \left(\left(x^1\right)'(t) + a\right) dt = \int_0^1 \left(x^1\right)'(t)dt + a = -a + a = 0. \tag{D.98}$$

Therefore, we have:

$$\begin{aligned}
\int_0^1 \left(x^1\right)'(t)^2 dt &= \int_0^1 \left(-a + f^1(t)\right)^2 dt && \text{(D.99)}\\
&= \int_0^1 a^2 - 2af^1(t) + \left(f^1\right)^2(t)dt && \text{(D.100)}\\
&= a^2 - 2a\int_0^1 f^1(t)dt + \int_0^1 \left(f^1\right)^2(t)dt && \text{(D.101)}\\
&= a^2 + \int_0^1 \left(f^1\right)^2(t)dt && \text{(D.102)}
\end{aligned}$$

According to Equation (D.71), we have:

$$\left|f^1(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3. \tag{D.103}$$

Therefore:

$$\left(f^1\right)^2(t) \leqslant C_5^2 \left(|a| + |b|\right)^6. \tag{D.104}$$

And finally:

$$\int_0^1 \left(x^1\right)'^2(t)dt = a^2 + \int_0^1 \left(f^1\right)^2(t)dt \tag{D.105}$$

with

$$\left|\int_0^1 \left(f^1\right)^2(t)dt\right| \leqslant \int_0^1 \left|\left(f^1\right)^2(t)\right| dt \leqslant \int_0^1 C_5^2 \left(|a| + |b|\right)^6 dt = C_5^2 \left(|a| + |b|\right)^6. \tag{D.106}$$

Likewise, we have:

$$\int_0^1 \left(x^2\right)'^2(t)dt = b^2 + \int_0^1 \left(f^2\right)^2(t)dt \tag{D.107}$$

with

$$\left|\int_0^1 \left(f^2\right)^2(t)dt\right| \leqslant \int_0^1 \left|\left(f^2\right)^2(t)\right| dt \leqslant \int_0^1 C_5^2 \left(|a| + |b|\right)^6 dt = C_5^2 \left(|a| + |b|\right)^6. \tag{D.108}$$

We recall that:

$$\begin{aligned}
&x^1(t) = (1-t)a + f^1(t) \text{ with } \left|f^1(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 && \text{(D.109)}\\
&\left(x^1\right)'(t) = -a + \left(f^1\right)'(t) \text{ with } \left|\left(f^1\right)'(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 && \text{(D.110)}\\
&x^2(t) = tb + f^2(t) \text{ with } \left|f^2(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3 && \text{(D.111)}\\
&\left(x^2\right)'(t) = b + \left(f^2\right)'(t) \text{ with } \left|\left(f^2\right)'(t)\right| \leqslant C_5 \left(|a| + |b|\right)^3. && \text{(D.112)}
\end{aligned}$$

Therefore:

$$\begin{aligned}
\left(x^1(t)\left(x^2\right)'(t) - x^2(t)\left(x^1\right)'(t)\right)^2 &= \left((1-t)ab - (-tba)\right)^2 + \tilde{f}(t) && \text{(D.113)}\\
&= a^2b^2 + \tilde{f}(t) && \text{(D.114)}
\end{aligned}$$

with $\left|\tilde{f}(t)\right| \leqslant C\left(|a| + |b|\right)^6$.
Therefore:

$$\int_0^1 \left(x^1(t)\left(x^2\right)'(t) - x^2(t)\left(x^1\right)'(t)\right)^2 dt = a^2b^2 + \int_0^1 \tilde{f}(t)dt \tag{D.115}$$

with

$$\left|\int_0^1 \tilde{f}(t)dt\right| \leqslant \int_0^1 \left|\tilde{f}(t)\right| dt \leqslant \int_0^1 C\left(|a|+|b|\right)^6 dt = C\left(|a|+|b|\right)^6. \tag{D.116}$$

Finally, by taking the expression of $d^2\left(A,B\right)$ in Equation (D.93) and replacing the integrals by the results obtained in Equations (D.105), (D.107), (D.115), we obtain:

$$\begin{aligned} d^2\left(A,B\right) =& a^2 + \int_0^1 \left(f^1\right)^2(t)dt \\ &+ b^2 + \int_0^1 \left(f^2\right)^2(t)dt \\ &- \frac{1}{3}K\left(\sigma\right)\left(a^2b^2 + \int_0^1 \tilde{f}(t)dt\right) \\ &+ \int_0^1 f\left(u,x\right)dt \end{aligned} \tag{D.117}$$

and therefore:

$$\left|d^2\left(A,B\right) - a^2 - b^2 + \frac{1}{3}K\left(\sigma\right)a^2b^2\right| = \left|\int_0^1 \left(f^1\right)^2(t)dt + \int_0^1 \left(f^2\right)^2(t)dt - \frac{1}{3}K\left(\sigma\right)\int_0^1 \tilde{f}(t)dt + \int_0^1 f\left(u,x\right)dt\right| \tag{D.118}$$

$$\leqslant \left|\int_0^1 \left(f^1\right)^2(t)dt\right| + \left|\int_0^1 \left(f^2\right)^2(t)dt\right| + \frac{1}{3}\left|K\left(\sigma\right)\right|\left|\int_0^1 \tilde{f}(t)dt\right| + \left|\int_0^1 f\left(u,x\right)dt\right| \tag{D.119}$$

$$\leqslant C_5^2\left(|a|+|b|\right)^6 + C_5^2\left(|a|+|b|\right)^6 + \frac{1}{3}\left|K\left(\sigma\right)\right|C\left(|a|+|b|\right)^6 + C\left(|a|+|b|\right)^5 \tag{D.120}$$

which concludes the proof.

□

Appendix E

The Siegel disk

This appendix is dedicated to the study of the Siegel disk. A summary of useful properties of the Siegel disk can be found in Section 4.2.2 and in [21]. We give here a more detailed presentation of the Siegel manifold including the mathematical proofs of the properties.

Contents

E.1 Definition

Definition 16. *The Siegel disk is defined as the set of complex matrices M of shape $N \times N$ with singular values lower than one, which can also be written:*

$$\mathcal{SD}_N = \left\{M \in \mathbb{C}^{N\times N} \mid I - MM^H > 0\right\} \tag{E.1}$$

or equally:

$$\mathcal{SD}_N = \left\{M \in \mathbb{C}^{N\times N} \mid I - M^HM > 0\right\} \tag{E.2}$$

We use the partial ordering of the set complex matrices: we note $A > B$ when the difference $A - B$ is a positive definite matrix.

Note that another definition of the Siegel disk also exists in other papers [54], imposing an additional symmetry condition on the matrix M: $M = M^T$. We will note require the symmetry condition in our work.

Property 45. *The Siegel disk can also be defined as the set of complex matrices M with a linear operator norm lower than one: $\mathcal{SD}_N = \left\{M \in \mathbb{C}^N, |||M||| < 1\right\}$, where $|||M||| = \sup_{X\in\mathbb{C}^N, \|X\|=1} (\|MX\|)$.*

E.2 The metric

We endow the Siegel disk $\mathcal{SD}_N$ with the following metric:

$$ds^2_{\mathbb{SD}_N} = \text{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H\right). \tag{E.3}$$

The expression of the square of the distance is given for all $\Omega, \Psi \in \mathcal{SD}_n$ by:

$$d^2_{\mathbb{SD}_N}(\Omega, \Psi) = \frac{1}{4} \operatorname{trace}\left(\log^2\left(\frac{I + C^{1/2}}{I - C^{1/2}}\right)\right) \tag{E.4}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}^2\left(C^{1/2}\right)\right) \tag{E.5}$$

with $C = (\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Omega\Psi^H\right)^{-1}$.
Another distance named the *Kobayashi distance measure* d_K on the Siegel disk $\mathcal{SD}_N$ is also given in [45].
L'expression du produit scalaire est alors, $\forall \Omega \in \mathcal{SD}_N, \forall v, w \in \mathbb{C}^{N\times N}$:

$$\langle v, w\rangle_\Omega = \frac{1}{2} \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} v \left(I - \Omega^H\Omega\right)^{-1} w^H\right) \tag{E.6}$$

$$+ \frac{1}{2} \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} w \left(I - \Omega^H\Omega\right)^{-1} v^H\right). \tag{E.7}$$

The norm of a vector belonging to the tangent space is therefore:

$$\|v\|^2_\Omega = \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} v \left(I - \Omega^H\Omega\right)^{-1} v^H\right). \tag{E.8}$$

E.3 The isometry

The following function is a key function in the Siegel disk:

$$\Phi_\Omega(\Psi) = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.9}$$

In the article of Ben Jeuris and Raf Vandebril [45], this function is said to be an isometry for the Siegel distance described in Equation (E.4). We will prove this result in Theorem 9.

We now study the differential of the isometry Φ.

Preliminary 6. *If* $\|\!|A|\!\| < 1$ *and* $\|\!|B|\!\| < 1$*, then:*

$$(I + AB)^\alpha A = A(I + BA)^\alpha \quad \forall \alpha \in \mathbb{R} \tag{E.10}$$

Proof. Let $\|\!|A|\!\| < 1$ and $\|\!|B|\!\| < 1$ and $\alpha \in \mathbb{R}$.
Then we have $\|\!|AB|\!\| \leqslant \|\!|A|\!\|\,\|\!|B|\!\| < 1$.
Hence, we can express $(I + AB)^\alpha$ as a power series:

$$(I + AB)^\alpha = \sum_{n=0}^{+\infty} \binom{\alpha}{n} (AB)^n \tag{E.11}$$

where $\binom{\alpha}{0} = 1$ and, $\forall n \in \mathbb{N}^*$, $\binom{\alpha}{n} = \frac{\alpha(\alpha-1)\ldots(\alpha-n+1)}{n!} = \frac{\prod_{i=0}^{n-1}(\alpha-i)}{n!}$.
Therefore, we have:

$$(I + AB)^\alpha A = \left(\sum_{n=0}^{+\infty} \binom{\alpha}{n} (AB)^n\right) A \tag{E.12}$$

$$= \sum_{n=0}^{+\infty} \binom{\alpha}{n} (AB)^n A \tag{E.13}$$

$$= \sum_{n=0}^{+\infty} \binom{\alpha}{n} A(BA)^n \tag{E.14}$$

$$= A\left(\sum_{n=0}^{+\infty} \binom{\alpha}{n} (BA)^n\right) \tag{E.15}$$

$$= A(I + BA)^\alpha \tag{E.16}$$

□

This preliminary will be very useful to prove next property.

Property 46. *The differential of the isometry Φ has the following expression:*

$$D\Phi_{\Omega}(\Psi)[h] = \left(I - \Omega\Omega^H\right)^{1/2} \left(I - \Psi\Omega^H\right)^{-1} h \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} \tag{E.17}$$

Proof.

$$\begin{aligned}
& D\Phi_{\Omega}(\Psi)[h] && \text{(E.18)}\\
=& \left(I - \Omega\Omega^H\right)^{-1/2} \left[h \left(I - \Omega^H\Psi\right)^{-1} + (\Psi - \Omega) \left(I - \Omega^H\Psi\right)^{-1} \left(\Omega^H h\right) \left(I - \Omega^H\Psi\right)^{-1}\right] \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.19)}\\
=& \left(I - \Omega\Omega^H\right)^{-1/2} \left[I + (\Psi - \Omega) \left(I - \Omega^H\Psi\right)^{-1} \Omega^H\right] h \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.20)}\\
=& \left(I - \Omega\Omega^H\right)^{-1/2} \left[I + \left(\Psi\Omega^H - \Omega\Omega^H\right) \left(I - \Psi\Omega^H\right)^{-1}\right] h \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.21)}\\
=& \left(I - \Omega\Omega^H\right)^{-1/2} \left[\left(I - \Psi\Omega^H\right) + \left(\Psi\Omega^H - \Omega\Omega^H\right)\right] \left(I - \Psi\Omega^H\right)^{-1} h \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.22)}\\
=& \left(I - \Omega\Omega^H\right)^{-1/2} \left(I - \Omega\Omega^H\right) \left(I - \Psi\Omega^H\right)^{-1} h \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.23)}\\
=& \left(I - \Omega\Omega^H\right)^{1/2} \left(I - \Psi\Omega^H\right)^{-1} h \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.24)}
\end{aligned}$$

□

Property 47. *The differential of the isometry Φ has the following property with respect to the conjugate transpose operator:*

$$D\Phi_{\Omega}(\Psi)[h]^H = D\Phi_{\Omega^H}\left(\Psi^H\right)\left[h^H\right]. \tag{E.25}$$

The proof is immediate using Property 46.

Property 48. *The inverse of the function Φ described in Equation (E.9) is:*

$$\Phi_{\Omega}^{-1}(\Psi) = \left(I - \Omega\Omega^H\right)^{1/2} \left(I + \Psi\Omega^H\right)^{-1} (\Psi + \Omega) \left(I - \Omega^H\Omega\right)^{-1/2} \tag{E.26}$$

Proof.

$$\begin{aligned}
& \Psi' = \Phi_{\Omega}(\Psi) && \text{(E.27)}\\
\Leftrightarrow & \Psi' = \left(I - \Omega\Omega^H\right)^{-1/2} (\Psi - \Omega) \left(I - \Omega^H\Psi\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.28)}\\
\Leftrightarrow & \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} \left(I - \Omega^H\Psi\right) = (\Psi - \Omega) && \text{(E.29)}\\
\Leftrightarrow & \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} + \Omega = \Psi + \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} \Omega^H\Psi && \text{(E.30)}\\
\Leftrightarrow & \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} + \Omega = \left(I + \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} \Omega^H\right) \Psi && \text{(E.31)}
\end{aligned}$$

Using Preliminary 6, we can give another expression of the term Ω on the left side of Equation (E.31):

$$\Omega = \left(I - \Omega\Omega^H\right)^{1/2} \Omega \left(I - \Omega^H\Omega\right)^{-1/2} \tag{E.32}$$

We now give another expression for the first factor on the right side of Equation (E.31):

$$\begin{aligned}
& I + \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} \Omega^H\\
=& I + \left(I - \Omega\Omega^H\right)^{1/2} \Psi'\Omega^H \left(I - \Omega\Omega^H\right)^{-1/2} \quad \text{using Preliminary 6} && \text{(E.33)}\\
=& \left(I - \Omega\Omega^H\right)^{1/2} \left(I + \Psi'\Omega^H\right) \left(I - \Omega\Omega^H\right)^{-1/2} && \text{(E.34)}
\end{aligned}$$

Coming back to Equation (E.31) and using Equations (E.32) and (E.34), we obtain:

$$\begin{aligned}
& \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} + \Omega = \left(I + \left(I - \Omega\Omega^H\right)^{1/2} \Psi' \left(I - \Omega^H\Omega\right)^{-1/2} \Omega^H\right) \Psi && \text{(E.35)}\\
\Leftrightarrow & \left(I - \Omega\Omega^H\right)^{1/2} \left(\Psi' + \Omega\right) \left(I - \Omega^H\Omega\right)^{-1/2} = \left(I - \Omega\Omega^H\right)^{1/2} \left(I + \Psi'\Omega^H\right) \left(I - \Omega\Omega^H\right)^{-1/2} \Psi && \text{(E.36)}\\
\Leftrightarrow & \left(\Psi' + \Omega\right) \left(I - \Omega^H\Omega\right)^{-1/2} = \left(I + \Psi'\Omega^H\right) \left(I - \Omega\Omega^H\right)^{-1/2} \Psi && \text{(E.37)}\\
\Leftrightarrow & \left(I - \Omega\Omega^H\right)^{1/2} \left(I + \Psi'\Omega^H\right)^{-1} \left(\Psi' + \Omega\right) \left(I - \Omega^H\Omega\right)^{-1/2} = \Psi. && \text{(E.38)}
\end{aligned}$$

Therefore we have shown that application Φ is invertible and that:

$$\Phi_{\Omega}^{-1}(\Psi) = \left(I - \Omega\Omega^H\right)^{1/2} \left(I + \Psi\Omega^H\right)^{-1} (\Psi + \Omega) \left(I - \Omega^H\Omega\right)^{-1/2}. \tag{E.39}$$

□

In [45], the following property about Φ^{-1} is mentioned:

Property 49. *The inverse of the function Φ described in Equation (E.9) also has the following expression:*

$$\Phi_{\Omega}^{-1}(\Psi) = \Phi_{-\Omega}(\Psi) \tag{E.40}$$

We present three proofs of this property.

Proof. Here is a first proof in which we reason by equivalence. The expression of $\Phi_{\Omega}^{-1}(\Psi)$ is given in Property 48. From the definition of Φ given by Equation (E.9), we have:

$$\Phi_{-\Omega}(\Psi) = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi + \Omega)\left(I + \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.41}$$

Therefore, we have:

$$\begin{aligned}
&\Phi_{\Omega}^{-1}(\Psi) = \Phi_{-\Omega}(\Psi) && \text{(E.42)}\\
\Leftrightarrow &\left(I - \Omega\Omega^H\right)^{1/2}\left(I + \Psi\Omega^H\right)^{-1}(\Psi + \Omega)\left(I - \Omega^H\Omega\right)^{-1/2} && \text{(E.43)}\\
&\quad = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi + \Omega)\left(I + \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2}\\
\Leftrightarrow &\left(I - \Omega\Omega^H\right)\left(I + \Psi\Omega^H\right)^{-1}(\Psi + \Omega) && \text{(E.44)}\\
&\quad = (\Psi + \Omega)\left(I + \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)\\
\Leftrightarrow &\left(I + \Psi\Omega^H\right)^{-1}\Psi + \left(I + \Psi\Omega^H\right)^{-1}\Omega - \Omega\Omega^H\left(I + \Psi\Omega^H\right)^{-1}\Psi - \Omega\Omega^H\left(I + \Psi\Omega^H\right)^{-1}\Omega && \text{(E.45)}\\
&\quad = \Psi\left(I + \Omega^H\Psi\right)^{-1} + \Omega\left(I + \Omega^H\Psi\right)^{-1} - \Psi\left(I + \Omega^H\Psi\right)^{-1}\Omega^H\Omega - \Omega\left(I + \Omega^H\Psi\right)^{-1}\Omega^H\Omega
\end{aligned}$$

Thanks to Preliminary 6, we notice that:

$$\left(I + \Psi\Omega^H\right)^{-1}\Psi = \Psi\left(I + \Omega^H\Psi\right)^{-1} \tag{E.46}$$

$$\Omega\Omega^H\left(I + \Psi\Omega^H\right)^{-1}\Omega = \Omega\left(I + \Omega^H\Psi\right)^{-1}\Omega^H\Omega \tag{E.47}$$

Equation (E.45) is therefore equivalent to the following equation:

$$\begin{aligned}
&\left(I + \Psi\Omega^H\right)^{-1}\Omega - \Omega\Omega^H\left(I + \Psi\Omega^H\right)^{-1}\Psi = \Omega\left(I + \Omega^H\Psi\right)^{-1} - \Psi\left(I + \Omega^H\Psi\right)^{-1}\Omega^H\Omega && \text{(E.48)}\\
\Leftrightarrow &\left(I + \Psi\Omega^H\right)^{-1}\Omega + \Psi\left(I + \Omega^H\Psi\right)^{-1}\Omega^H\Omega = \Omega\left(I + \Omega^H\Psi\right)^{-1} + \Omega\Omega^H\left(I + \Psi\Omega^H\right)^{-1}\Psi && \text{(E.49)}\\
\Leftrightarrow &\left(I + \Psi\Omega^H\right)^{-1}\Omega + \Psi\Omega^H\left(I + \Psi\Omega^H\right)^{-1}\Omega = \Omega\left(I + \Omega^H\Psi\right)^{-1} + \Omega\left(I + \Omega^H\Psi\right)^{-1}\Omega^H\Psi && \text{(E.50)}\\
\Leftrightarrow &\left(I + \Psi\Omega^H\right)\left(I + \Psi\Omega^H\right)^{-1}\Omega = \Omega\left(I + \Omega^H\Psi\right)^{-1}\left(I + \Omega^H\Psi\right) && \text{(E.51)}\\
\Leftrightarrow &\Omega = \Omega && \text{(E.52)}
\end{aligned}$$

The condition defined by Equation (E.52) being true, we can conclude by equivalence that we have $\Phi_{\Omega}^{-1}(\Psi) = \Phi_{-\Omega}(\Psi)$, which concludes the proof. □

Proof. Here is a second proof in which we start by checking the equality of the differentials of the function $\Phi_{\Omega}^{-1}(\Psi)$ defined in Property 48 and of the function $\Phi_{-\Omega}(\Psi)$ defined in Equation (E.9), then we will check the equality of these two functions at one point.

According to Property 46, the differential of the isometry Φ has the following expression:

$$D\Phi_{\Omega}(\Psi)[h] = \left(I - \Omega\Omega^H\right)^{1/2}\left(I - \Psi\Omega^H\right)^{-1}h\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.53}$$

Therefore, we have:

$$D\Phi_{-\Omega}(\Psi)[h] = \left(I - \Omega\Omega^H\right)^{1/2}\left(I + \Psi\Omega^H\right)^{-1}h\left(I + \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.54}$$

Moreover, using the definition of the function $\Phi_{\Omega}^{-1}(\Psi)$ given in Property 48, we have:

$$\begin{aligned}
&D\left(\Phi_{\Omega}^{-1}\right)(\Psi)[h]\\
&= \left(I - \Omega\Omega^H\right)^{1/2}\left[\left(I + \Psi\Omega^H\right)^{-1}h - \left(I + \Psi\Omega^H\right)^{-1}\left(h\Omega^H\right)\left(I + \Psi\Omega^H\right)^{-1}(\Psi + \Omega)\right]\left(I - \Omega^H\Omega\right)^{-1/2} && \text{(E.55)}\\
&= \left(I - \Omega\Omega^H\right)^{1/2}\left(I + \Psi\Omega^H\right)^{-1}h\left[I - \Omega^H\left(I + \Psi\Omega^H\right)^{-1}(\Psi + \Omega)\right]\left(I - \Omega^H\Omega\right)^{-1/2} && \text{(E.56)}
\end{aligned}$$

However, we have according to Preliminary 6:

$$\begin{aligned}
&I-\Omega^H\left(I+\Psi\Omega^H\right)^{-1}(\Psi+\Omega)\\
=&I-\left(I+\Omega^H\Psi\right)^{-1}\left(\Omega^H\Psi+\Omega^H\Omega\right) &\text{(E.57)}\\
=&\left(I+\Omega^H\Psi\right)^{-1}\left(\left(I+\Omega^H\Psi\right)-\left(\Omega^H\Psi+\Omega^H\Omega\right)\right) &\text{(E.58)}\\
=&\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right) &\text{(E.59)}
\end{aligned}$$

Which gives us, coming back to Equation (E.56) :

$$\begin{aligned}
D\left(\Phi_\Omega^{-1}\right)(\Psi)[h]&=\left(I-\Omega\Omega^H\right)^{1/2}\left(I+\Psi\Omega^H\right)^{-1}h\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)\left(I-\Omega^H\Omega\right)^{-1/2} &\text{(E.60)}\\
&=\left(I-\Omega\Omega^H\right)^{1/2}\left(I+\Psi\Omega^H\right)^{-1}h\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.61)}
\end{aligned}$$

According to Equations (E.61) and (E.54), we have the equality of the differentials: $D\left(\Phi_\Omega^{-1}\right)(\Psi)[h]=D\Phi_{-\Omega}(\Psi)[h]$.

To check the equality of the functions Φ_Ω^{-1} and $\Phi_{-\Omega}$, It is sufficient to verify the equality of these two functions at one point

From the expression of $\Phi_\Omega^{-1}(\Psi)$ given in Property 48, we have $\Phi_\Omega^{-1}(-\Omega)=0$. Likewise, the expression of $\Phi_\Omega(\Psi)$ given in Equation (E.9) gives us: $\Phi_{-\Omega}(-\Omega)=0$.

Since the functions $\Phi_\Omega^{-1}(\Psi)$ and $\Phi_{-\Omega}(\Psi)$ both have the same differentials and have the same value at point $-\Omega$, we have $\Phi_\Omega^{-1}(\Psi)=\Phi_{-\Omega}(\Psi)$ for all $\Psi\in\mathcal{SD}_N$.

□

Proof. We now present a third proof which unlike the previous two proofs does not use the expression of Φ_Ω^{-1} given in Property 48.

In this proof, we will show that $\Phi_\Omega(\Phi_{-\Omega}(\Psi))=\Psi$ and that $\Phi_{-\Omega}(\Phi_\Omega(\Psi))=\Psi$, $\forall(\Omega,\Psi)\in\mathcal{SD}_N^2$.

$$\begin{aligned}
&\Phi_\Omega(\Phi_{-\Omega}(\Psi))\\
=&\left(I-\Omega\Omega^H\right)^{-1/2}\left(\Phi_{-\Omega}(\Psi)-\Omega\right)\left(I-\Omega^H\Phi_{-\Omega}(\Psi)\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.62)}\\
=&\left(I-\Omega\Omega^H\right)^{-1/2} &\text{(E.63)}\\
&\left(\left(I-\Omega\Omega^H\right)^{-1/2}(\Psi+\Omega)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}-\Omega\right)\\
&\left(I-\Omega^H\left(I-\Omega\Omega^H\right)^{-1/2}(\Psi+\Omega)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1}\\
&\left(I-\Omega^H\Omega\right)^{1/2}
\end{aligned}$$

We can obtain another expression of the second factor of Equation (E.63) using Preliminary 6:

$$\begin{aligned}
&\left(I-\Omega\Omega^H\right)^{-1/2}(\Psi+\Omega)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}-\Omega\\
=&\left(I-\Omega\Omega^H\right)^{-1/2}\left[(\Psi+\Omega)\left(I+\Omega^H\Psi\right)^{-1}-\Omega\right]\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.64)}\\
=&\left(I-\Omega\Omega^H\right)^{-1/2}\left[(\Psi+\Omega)-\Omega\left(I+\Omega^H\Psi\right)\right]\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.65)}\\
=&\left(I-\Omega\Omega^H\right)^{-1/2}\left(\Psi-\Omega\Omega^H\Psi\right)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.66)}\\
=&\left(I-\Omega\Omega^H\right)^{-1/2}\left(I-\Omega\Omega^H\right)\Psi\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.67)}\\
=&\left(I-\Omega\Omega^H\right)^{1/2}\Psi\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} &\text{(E.68)}
\end{aligned}$$

Preliminary 6 also allows us to obtain another expression of the third term of Equation (E.63):

$$
\begin{aligned}
&\left(I-\Omega^H\left(I-\Omega\Omega^H\right)^{-1/2}(\Psi+\Omega)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1} \\
&=\left(I-\left(I-\Omega^H\Omega\right)^{-1/2}\left(\Omega^H\Psi+\Omega^H\Omega\right)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1} && \text{(E.69)} \\
&=\left(\left(I-\Omega^H\Omega\right)^{-1/2}\left[I-\left(\Omega^H\Psi+\Omega^H\Omega\right)\left(I+\Omega^H\Psi\right)^{-1}\right]\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1} && \text{(E.70)} \\
&=\left(\left(I-\Omega^H\Omega\right)^{-1/2}\left[\left(I+\Omega^H\Psi\right)-\left(\Omega^H\Psi+\Omega^H\Omega\right)\right]\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1} && \text{(E.71)} \\
&=\left(\left(I-\Omega^H\Omega\right)^{-1/2}\left(I-\Omega^H\Omega\right)\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1} && \text{(E.72)} \\
&=\left(\left(I-\Omega^H\Omega\right)^{1/2}\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\right)^{-1} && \text{(E.73)} \\
&=\left(I-\Omega^H\Omega\right)^{-1/2}\left(I+\Omega^H\Psi\right)\left(I-\Omega^H\Omega\right)^{-1/2} && \text{(E.74)}
\end{aligned}
$$

We now come back to Equation (E.63) in which we replace the second factor by the result obtained in Equation (E.68) and the third factor by the result obtained in Equation (E.74). We obtain:

$$
\begin{aligned}
&\Phi_\Omega\left(\Phi_{-\Omega}\left(\Psi\right)\right) \\
&=\left(I-\Omega\Omega^H\right)^{-1/2} \\
&\left(I-\Omega\Omega^H\right)^{1/2}\Psi\left(I+\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \\
&\left(I-\Omega^H\Omega\right)^{-1/2}\left(I+\Omega^H\Psi\right)\left(I-\Omega^H\Omega\right)^{-1/2} \\
&\left(I-\Omega^H\Omega\right)^{1/2}
\end{aligned}
\qquad \text{(E.75)}
$$

After simplifications, we have:

$$
\Phi_\Omega\left(\Phi_{-\Omega}\left(\Psi\right)\right)=\Psi \quad \forall\left(\Omega,\Psi\right)\in\mathcal{SD}_N^2 \qquad \text{(E.76)}
$$

If we replace Ω by $-\Omega$ in Equation (E.76), we see that we also have:

$$
\Phi_{-\Omega}\left(\Phi_{\Omega}\left(\Psi\right)\right)=\Psi \quad \forall\left(\Omega,\Psi\right)\in\mathcal{SD}_N^2 \qquad \text{(E.77)}
$$

Using Equations (E.76) and (E.77), we can conclude that $\Phi_\Omega^{-1}=\Phi_{-\Omega}$.

□

Using properties 48 and 49, we obtain another expression of the isometry Φ:

$$
\Phi_\Omega\left(\Psi\right)=\Phi_{-\Omega}^{-1}\left(\Psi\right)=\left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(\Psi-\Omega\right)\left(I-\Omega^H\Omega\right)^{-1/2} \qquad \text{(E.78)}
$$

We will use Equation (E.78) to prove properties 50 and 51.

Property 50. *The isometry Φ has the following property with respect to the conjugate transpose operator:*

$$
\Phi_\Omega\left(\Psi\right)^H=\Phi_{\Omega^H}\left(\Psi^H\right) \qquad \text{(E.79)}
$$

Proof.

$$
\begin{aligned}
\Phi_\Omega\left(\Psi\right)^H=&\Phi_{-\Omega}^{-1}\left(\Psi\right)^H && \text{(E.80)} \\
=&\left(\left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(\Psi-\Omega\right)\left(I-\Omega^H\Omega\right)^{-1/2}\right)^H && \text{(E.81)} \\
=&\left(I-\Omega^H\Omega\right)^{-1/2}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} && \text{(E.82)} \\
=&\Phi_{\Omega^H}\left(\Psi^H\right) && \text{(E.83)}
\end{aligned}
$$

□

Property 51.

$$
\Phi_\Omega\left(\Psi\right)\Phi_\Omega\left(\Psi\right)^H=I-\left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \qquad \text{(E.84)}
$$

Proof.

$$\Phi_\Omega(\Psi)\,\Phi_\Omega(\Psi)^H \tag{E.85}$$
$$=\Phi_\Omega(\Psi)\,\Phi_{-\Omega}^{-1}(\Psi)^H \tag{E.86}$$
$$=\left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}\left(I-\Omega^H\Omega\right)^{-1/2}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.87}$$
$$=\left(I-\Omega\Omega^H\right)^{-1/2}(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.88}$$
$$=I-\left(I-\Omega\Omega^H\right)^{-1/2}\left[I-(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Psi^H\right)^{-1}\right]\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.89}$$
$$=I-\left(I-\Omega\Omega^H\right)^{-1/2}\left[\left(I-\Omega\Psi^H\right)-(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(\Psi^H-\Omega^H\right)\right]\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.90}$$

Note that:

$$\begin{aligned}
&\left(I-\Omega\Psi^H\right)-(\Psi-\Omega)\left(I-\Omega^H\Psi\right)^{-1}\left(\Psi^H-\Omega^H\right) && \text{(E.91)}\\
=&\left(I-\Psi\Omega^H\right)\left(I-\Psi\Omega^H\right)^{-1} && \text{(E.92)}\\
&-\Omega\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Psi\right)\Psi^H\\
&-\Psi\left(I-\Omega^H\Psi\right)^{-1}\Psi^H\\
&+\Psi\left(I-\Omega^H\Psi\right)^{-1}\Omega^H\\
&+\Omega\left(I-\Omega^H\Psi\right)^{-1}\Psi^H\\
&-\Omega\left(I-\Omega^H\Psi\right)^{-1}\Omega^H\\
=&\left(I-\Psi\Omega^H\right)^{-1}-\Psi\Omega^H\left(I-\Psi\Omega^H\right)^{-1} && \text{(E.93)}\\
&-\Omega\left(I-\Omega^H\Psi\right)^{-1}\Psi^H+\Omega\Omega^H\left(I-\Psi\Omega^H\right)^{-1}\Psi\Psi^H\\
&-\left(I-\Psi\Omega^H\right)^{-1}\Psi\Psi^H\\
&+\Psi\Omega^H\left(I-\Psi\Omega^H\right)^{-1}\\
&+\Omega\left(I-\Omega^H\Psi\right)^{-1}\Psi^H\\
&-\Omega\Omega^H\left(I-\Psi\Omega^H\right)^{-1}\\
=&\left(I-\Omega\Omega^H\right)\left(I-\Psi\Omega^H\right)^{-1}-\left(I-\Omega\Omega^H\right)\left(I-\Psi\Omega^H\right)^{-1}\Psi\Psi^H && \text{(E.94)}\\
=&\left(I-\Omega\Omega^H\right)\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right) && \text{(E.95)}
\end{aligned}$$

By substituting this result into Equation (E.90), we finally obtain:

$$\Phi_\Omega(\Psi)\,\Phi_\Omega(\Psi)^H \tag{E.96}$$
$$=I-\left(I-\Omega\Omega^H\right)^{-1/2}\left[\left(I-\Omega\Omega^H\right)\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)\right]\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.97}$$
$$=I-\left(I-\Omega\Omega^H\right)^{+1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.98}$$

□

Property 52. *If Ω and Ψ belong to the Siegel disk $\mathcal{SD}_N$, then $\Phi_\Omega(\Psi)$ also belongs to the Siegel disk $\mathcal{SD}_N$.*

Proof. We assume that Ω and Ψ belong to the Siegel disk $\mathcal{SD}_N$. By definition of the Siegel disk, the image $\Phi_\Omega(\Psi)$ belongs to the Siegel disk if and only if $I-\Phi_\Omega(\Psi)\,\Phi_\Omega(\Psi)^H>0$.

Using Property 51, we have:

$$I-\Phi_\Omega(\Psi)\,\Phi_\Omega(\Psi)^H \tag{E.99}$$
$$=\left(I-\Omega\Omega^H\right)^{+1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.100}$$
$$=\left(I-\Omega\Omega^H\right)^{+1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)^{1/2}\left(I-\Psi\Psi^H\right)^{1/2}\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2} \tag{E.101}$$
$$=MM^H \quad \text{with } M=\left(I-\Omega\Omega^H\right)^{+1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)^{1/2} \tag{E.102}$$

To prove that the matrix MM^H is positive definite, it is sufficient to prove that the matrix M is a full rank matrix, which is true as M is the product of three full rank matrices.

Indeed, $\left(I - \Omega\Omega^H\right) > 0$ since $\Omega \in \mathcal{SD}_N$ and $\left(I - \Psi\Psi^H\right) > 0$ since $\Psi \in \mathcal{SD}_N$.

We now prove that the matrix $\left(I - \Psi\Omega^H\right)$ is also a full rank matrix. Since Ω and Ψ belong to the Siegel disk, we have: $\|\Omega\| < 1$ and $\|\Psi\| < 1$. Hence, we have: $\left\|\Psi\Omega^H\right\| = \|\Psi\| \left\|\Omega^H\right\| = \|\Psi\| \|\Omega\| < 1$. By the absurd, if there exists $X \in \mathbb{C}^N \backslash \{0\}$ such that $\left(I - \Psi\Omega^H\right)(X) = 0$, this would implie that: $\|X\| = \left\|\Psi\Omega^H(X)\right\| \leqslant \left\|\Psi\Omega^H\right\| \|X\| < \|X\|$ which is impossible.

□

To prove that the map Φ is an isometry, we will use the following preliminary which establishes a first relation between the map Φ and the distance on the Siegel space defined in Equation (E.4).

Preliminary 7. *The distance on the Siegel disk defined in Equation (E.4) can also be expressed by the application Φ defined in Equation (E.9) as follows:*

$$d^2_{\mathcal{SD}_N}(\Omega, \Psi) = \frac{1}{4} \operatorname{trace}\left(\log^2\left(\frac{I + C^{1/2}}{I - C^{1/2}}\right)\right) \tag{E.103}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}^2\left(C^{1/2}\right)\right) \tag{E.104}$$

with $C = \Phi_\Omega(\Psi)\, \Phi_\Omega(\Psi)^H$.

Proof. We recall Equation (E.9) gives us the following expression of the square of the distance on the Siegel disk:

$$d^2_{\mathcal{SD}_N}(\Omega, \Psi) = \operatorname{trace}\left(\operatorname{arctanh}^2\left(C^{1/2}\right)\right) \tag{E.105}$$

with $C = (\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Omega\Psi^H\right)^{-1}$.

Note that the function $C \mapsto \operatorname{arctanh}^2\left(C^{1/2}\right)$ can be expressed as a Taylor series of the variable C when $\|x\| < 1$.

Since the trace operator est invariant by change of base, we have when the matrices C_1 and C_2 are similar:

$$\operatorname{trace}\left(\operatorname{arctanh}^2\left(C_1^{1/2}\right)\right) = \operatorname{trace}\left(\operatorname{arctanh}^2\left(C_2^{1/2}\right)\right) \tag{E.106}$$

We now define:

$$C_1 := \Phi_\Omega(\Psi)\, \Phi_\Omega(\Psi)^H \tag{E.107}$$

$$C_2 := (\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Omega\Psi^H\right)^{-1}. \tag{E.108}$$

To prove Preliminary 7, it is therefore sufficient to prove that the matrices C_1 and C_2 are similar.

$$C_1 = \Phi_\Omega(\Psi)\, \Phi_\Omega(\Psi)^H \tag{E.109}$$

$$C_1 = \Phi_\Omega(\Psi)\, \Phi^{-1}_{-\Omega}(\Psi)^H \quad \text{d'après l'équation (E.78)} \tag{E.110}$$

$$C_1 = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.111}$$

$$\left(I - \Omega^H\Omega\right)^{-1/2}\left(\Psi^H - \Omega^H\right)\left(I - \Omega\Psi^H\right)^{-1}\left(I - \Omega\Omega^H\right)^{1/2}$$

$$C_1 = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Omega\Psi^H\right)^{-1}\left(I - \Omega\Omega^H\right)^{1/2} \tag{E.112}$$

$$C_1 = \left(I - \Omega\Omega^H\right)^{-1/2} C_2 \left(I - \Omega\Omega^H\right)^{1/2} \tag{E.113}$$

Matrices C_1 and C_2 are similar according to Equation (E.113), which concludes the proof.

□

Theorem 9. *The function Φ described in Equation (E.9):*

$$\Phi_\Omega(\Psi) = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.114}$$

is an isometry for the Siegel distance described in Equation (E.4):

$$d^2_{\mathbb{SD}_N}(\Omega, \Psi) = \frac{1}{4} \operatorname{trace}\left(\log^2\left(\frac{I + C^{1/2}}{I - C^{1/2}}\right)\right) \tag{E.115}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}^2\left(C^{1/2}\right)\right) \tag{E.116}$$

with $C = (\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Omega\Psi^H\right)^{-1}$.

We give two proofs of this theorem.

Proof. In this first proof, we will show that:

$$d^2_{\mathcal{SD}_N}\left(\Phi_\Theta\left(\Omega\right),\Phi_\Theta\left(\Psi\right)\right)=d^2_{\mathcal{SD}_N}\left(\Omega,\Psi\right)\quad\forall\Omega,\Psi,\Theta\in\mathcal{SD}_N \tag{E.117}$$

to prove that the application Φ is an isometry.

According to Preliminary 7, it is sufficient to show that:

$$\operatorname{trace}\left(\operatorname{arctanh}^2\left(C_1^{1/2}\right)\right)=\operatorname{trace}\left(\operatorname{arctanh}^2\left(C_2^{1/2}\right)\right) \tag{E.118}$$

with:

$$C_1:=\Phi_{\Phi_\Theta(\Omega)}\left(\Phi_\Theta\left(\Psi\right)\right)\left(\Phi_{\Phi_\Theta(\Omega)}\left(\Phi_\Theta\left(\Psi\right)\right)\right)^H \tag{E.119}$$

$$C_2:=\Phi_\Omega\left(\Psi\right)\Phi_\Omega\left(\Psi\right)^H \tag{E.120}$$

Note that the function $C\mapsto\operatorname{arctanh}^2\left(C^{1/2}\right)$ can be expressed as a Taylor series of the variable C when $\left|\left|\left|x\right|\right|\right|<1$.

Since the trace operator is invariant by change of base, it is sufficient to show that matrices C_1 and C_2 are similar to prove Equation (E.118).

Preliminary 7 will then allow us to prove Theorem 9.

We have:

$$\begin{aligned}\Phi_{\Phi_\Theta(\Omega)}\left(\Phi_\Theta\left(\Psi\right)\right)=&\left(I-\Phi_\Theta\left(\Omega\right)\Phi_\Theta\left(\Omega\right)^H\right)^{-1/2}\\&\left(\Phi_\Theta\left(\Psi\right)-\Phi_\Theta\left(\Omega\right)\right)\\&\left(I-\Phi_\Theta\left(\Omega\right)^H\Phi_\Theta\left(\Psi\right)\right)^{-1}\\&\left(I-\Phi_\Theta\left(\Omega\right)^H\Phi_\Theta\left(\Omega\right)\right)^{1/2}\end{aligned} \tag{E.121}$$

We define:

$$A:=\left(I-\Phi_\Theta\left(\Omega\right)\Phi_\Theta\left(\Omega\right)^H\right)^{-1/2} \tag{E.122}$$

$$B:=\Phi_\Theta\left(\Psi\right)-\Phi_\Theta\left(\Omega\right) \tag{E.123}$$

$$C:=\left(I-\Phi_\Theta\left(\Omega\right)^H\Phi_\Theta\left(\Psi\right)\right)^{-1} \tag{E.124}$$

$$D:=\left(I-\Phi_\Theta\left(\Omega\right)^H\Phi_\Theta\left(\Omega\right)\right)^{1/2} \tag{E.125}$$

$$\tag{E.126}$$

Therefore, we have:

$$\Phi_{\Phi_\Theta(\Omega)}\left(\Phi_\Theta\left(\Psi\right)\right)=ABCD \tag{E.127}$$

Note that we have $A=A^H$ and $D=D^H$.

Therefore, we have:

$$C_1=\Phi_{\Phi_\Theta(\Omega)}\left(\Phi_\Theta\left(\Psi\right)\right)\left(\Phi_{\Phi_\Theta(\Omega)}\left(\Phi_\Theta\left(\Psi\right)\right)\right)^H \tag{E.128}$$

$$=ABCDD^HC^HB^HA^H \tag{E.129}$$

$$=ABCD^2C^HB^HA \tag{E.130}$$

Matrix C_1 is therefore similar to the product $A^2BCD^2C^HB^H$.

We now study the terms A^2, D^2, B and C. Using Property 51, we obtain:

$$A^2=\left(I-\Phi_\Theta\left(\Omega\right)\Phi_\Theta\left(\Omega\right)^H\right)^{-1} \tag{E.131}$$

$$=\left(\left(I-\Theta\Theta^H\right)^{1/2}\left(I-\Omega\Theta^H\right)^{-1}\left(I-\Omega\Omega^H\right)\left(I-\Theta\Omega^H\right)^{-1}\left(I-\Theta\Theta^H\right)^{1/2}\right)^{-1} \tag{E.132}$$

$$=\left(I-\Theta\Theta^H\right)^{-1/2}\left(I-\Theta\Omega^H\right)\left(I-\Omega\Omega^H\right)^{-1}\left(I-\Omega\Theta^H\right)\left(I-\Theta\Theta^H\right)^{-1/2} \tag{E.133}$$

Remembering that $\Phi_\Theta(\Omega)^H = \Phi_{\Theta^H}(\Omega^H)$ according to Property 50, we obtain an expression of D^2 using Property 51 in which we make the change of variables $\Theta \leftarrow \Theta^H$ and $\Omega \leftarrow \Omega^H$. We obtain:

$$\begin{aligned} D^2 =& I - \Phi_\Theta(\Omega)^H \Phi_\Theta(\Omega) && \text{(E.134)}\\ =& \left(I - \Theta^H\Theta\right)^{1/2} \left(I - \Omega^H\Theta\right)^{-1} \left(I - \Omega^H\Omega\right) \left(I - \Theta^H\Omega\right)^{-1} \left(I - \Theta^H\Theta\right)^{1/2} && \text{(E.135)} \end{aligned}$$

We now compute B:

$$\begin{aligned} B =& \Phi_\Theta(\Psi) - \Phi_\Theta(\Omega) && \text{(E.136)}\\ =& \left(I - \Theta\Theta^H\right)^{-1/2} \left[(\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1} - (\Omega - \Theta)\left(I - \Theta^H\Omega\right)^{-1}\right] \left(I - \Theta^H\Theta\right)^{1/2} && \text{(E.137)}\\ =& \left(I - \Theta\Theta^H\right)^{-1/2} \left[(\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Omega\right) - (\Omega - \Theta)\right] \left(I - \Theta^H\Omega\right)^{-1} \left(I - \Theta^H\Theta\right)^{1/2} && \text{(E.138)} \end{aligned}$$

By using several times Preliminary 6, we can give another expression for the following factor:

$$\begin{aligned} &(\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Omega\right) - (\Omega - \Theta)\\ =& (\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1} - (\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1}\Theta^H\Omega - \Omega + \Theta && \text{(E.139)}\\ =& (\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1} - \left(\Psi\Theta^H - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}\Omega - \Omega + \Theta && \text{(E.140)}\\ =& \Psi\left(I - \Theta^H\Psi\right)^{-1} - \Theta\left(I - \Theta^H\Psi\right)^{-1} + \Theta - \left(\Psi\Theta^H - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}\Omega - \Omega && \text{(E.141)}\\ =& \left(I - \Psi\Theta^H\right)^{-1}\Psi + \Theta\left(I - \Theta^H\Psi\right)^{-1}\left[\left(I - \Theta^H\Psi\right) - I\right] - \left[\left(I - \Psi\Theta^H\right) + \Psi\Theta^H - \Theta\Theta^H\right]\left(I - \Psi\Theta^H\right)^{-1}\Omega && \text{(E.142)}\\ =& \left(I - \Psi\Theta^H\right)^{-1}\Psi + \Theta\left(I - \Theta^H\Psi\right)^{-1}\left(-\Theta^H\Psi\right) - \left(I - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}\Omega && \text{(E.143)}\\ =& \left(I - \Psi\Theta^H\right)^{-1}\Psi - \Theta\Theta^H\left(I - \Psi\Theta^H\right)^{-1}\Psi - \left(I - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}\Omega && \text{(E.144)}\\ =& \left(I - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}\Psi - \left(I - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}\Omega && \text{(E.145)}\\ =& \left(I - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}(\Psi - \Omega). && \text{(E.146)} \end{aligned}$$

Coming back to Equation (E.138), we obtain:

$$\begin{aligned} B =& \left(I - \Theta\Theta^H\right)^{-1/2}\left(I - \Theta\Theta^H\right)\left(I - \Psi\Theta^H\right)^{-1}(\Psi - \Omega)\left(I - \Theta^H\Omega\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2} && \text{(E.147)}\\ =& \left(I - \Theta\Theta^H\right)^{1/2}\left(I - \Psi\Theta^H\right)^{-1}(\Psi - \Omega)\left(I - \Theta^H\Omega\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}. && \text{(E.148)} \end{aligned}$$

We can have another expression of B by noticing that:

$$B = \Phi_\Theta(\Psi) - \Phi_\Theta(\Omega) = -\left(\Phi_\Theta(\Omega) - \Phi_\Theta(\Psi)\right). \quad \text{(E.149)}$$

Then, by switching the roles of Ω and Ψ in the previous formulas, we obtain:

$$\begin{aligned} B =& -\left(I - \Theta\Theta^H\right)^{1/2}\left(I - \Omega\Theta^H\right)^{-1}(\Omega - \Psi)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2} && \text{(E.150)}\\ =& \left(I - \Theta\Theta^H\right)^{1/2}\left(I - \Omega\Theta^H\right)^{-1}(\Psi - \Omega)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}. && \text{(E.151)} \end{aligned}$$

Finally, we give another expression of the factor C:

$$C = \left(I - \Phi_\Theta(\Omega)^H \Phi_\Theta(\Psi)\right)^{-1} \tag{E.152}$$
$$= \left(I - \Phi_{-\Theta}^{-1}(\Omega)^H \Phi_\Theta(\Psi)\right)^{-1} \tag{E.153}$$
$$= \Big(I - \left(I - \Theta^H\Theta\right)^{-1/2}\left(\Omega^H - \Theta^H\right)\left(I - \Theta\Omega^H\right)^{-1}\left(I - \Theta\Theta^H\right)^{1/2} \tag{E.154}$$
$$\left(I - \Theta\Theta^H\right)^{-1/2}(\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}\Big)^{-1}$$
$$= \left(I - \left(I - \Theta^H\Theta\right)^{-1/2}\left(\Omega^H - \Theta^H\right)\left(I - \Theta\Omega^H\right)^{-1}(\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}\right)^{-1} \tag{E.155}$$
$$= \left(\left(I - \Theta^H\Theta\right)^{-1/2}\left[I - \left(\Omega^H - \Theta^H\right)\left(I - \Theta\Omega^H\right)^{-1}(\Psi - \Theta)\left(I - \Theta^H\Psi\right)^{-1}\right]\left(I - \Theta^H\Theta\right)^{1/2}\right)^{-1} \tag{E.156}$$
$$= \left(\left(I - \Theta^H\Theta\right)^{-1/2}\left[\left(I - \Theta^H\Psi\right) - \left(\Omega^H - \Theta^H\right)\left(I - \Theta\Omega^H\right)^{-1}(\Psi - \Theta)\right]\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}\right)^{-1}. \tag{E.157}$$

By using several times Preliminary 6, we give another expression for the following factor:

$$\left(I - \Theta^H\Psi\right) - \left(\Omega^H - \Theta^H\right)\left(I - \Theta\Omega^H\right)^{-1}(\Psi - \Theta) \tag{E.158}$$
$$= I - \Theta^H\Psi - \Omega^H\left(I - \Theta\Omega^H\right)^{-1}(\Psi - \Theta) + \Theta^H\left(I - \Theta\Omega^H\right)^{-1}\Psi - \Theta^H\left(I - \Theta\Omega^H\right)^{-1}\Theta \tag{E.159}$$
$$= I - \Theta^H\Psi - \left(I - \Omega^H\Theta\right)^{-1}\left(\Omega^H\Psi - \Omega^H\Theta\right) + \Theta^H\left(I - \Theta\Omega^H\right)^{-1}\Psi - \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1} \tag{E.160}$$
$$= I - \left(I - \Omega^H\Theta\right)^{-1}\left(\Omega^H\Psi - \Omega^H\Theta\right) - \Theta^H\Psi + \Theta^H\left(I - \Theta\Omega^H\right)^{-1}\Psi - \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1} \tag{E.161}$$
$$= \left(I - \Omega^H\Theta\right)^{-1}\left[\left(I - \Omega^H\Theta\right) - \left(\Omega^H\Psi - \Omega^H\Theta\right)\right] \tag{E.162}$$
$$+ \Theta^H\left(I - \Theta\Omega^H\right)^{-1}\left[I - \left(I - \Theta\Omega^H\right)\right]\Psi - \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1}$$
$$= \left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right) + \Theta^H\left(I - \Theta\Omega^H\right)^{-1}\left(\Theta\Omega^H\right)\Psi - \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1} \tag{E.163}$$
$$= \left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right) + \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1}\Omega^H\Psi - \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1} \tag{E.164}$$
$$= \left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right) - \Theta^H\Theta\left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right) \tag{E.165}$$
$$= \left(I - \Theta^H\Theta\right)\left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right). \tag{E.166}$$

We obtain, coming back to Equation (E.157) :

$$C = \left(\left(I - \Theta^H\Theta\right)^{-1/2}\left(I - \Theta^H\Theta\right)\left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}\right)^{-1} \tag{E.167}$$
$$= \left(\left(I - \Theta^H\Theta\right)^{1/2}\left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Psi\right)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}\right)^{-1} \tag{E.168}$$
$$= \left(I - \Theta^H\Theta\right)^{-1/2}\left(I - \Theta^H\Psi\right)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Theta\right)\left(I - \Theta^H\Theta\right)^{-1/2}. \tag{E.169}$$

We have seen previously that the matrix C_1 defined in Equation (E.119) is similar to the product of matrices $A^2BCD^2C^HB^H$.
We also have, according to the expressions of A^2, D^2, B and C obtained in Equations (E.133), (E.135), (E.151) and (E.169):

$$A^2BCD^2C^HB^H$$
$$= \left(I - \Theta\Theta^H\right)^{-1/2}\left(I - \Theta\Omega^H\right)\left(I - \Omega\Omega^H\right)^{-1}\left(I - \Omega\Theta^H\right)\left(I - \Theta\Theta^H\right)^{-1/2} \tag{E.170}$$
$$\left(I - \Theta\Theta^H\right)^{1/2}\left(I - \Omega\Theta^H\right)^{-1}(\Psi - \Omega)\left(I - \Theta^H\Psi\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}$$
$$\left(I - \Theta^H\Theta\right)^{-1/2}\left(I - \Theta^H\Psi\right)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Theta\right)\left(I - \Theta^H\Theta\right)^{-1/2}$$
$$\left(I - \Theta^H\Theta\right)^{1/2}\left(I - \Omega^H\Theta\right)^{-1}\left(I - \Omega^H\Omega\right)\left(I - \Theta^H\Omega\right)^{-1}\left(I - \Theta^H\Theta\right)^{1/2}$$
$$\left(I - \Theta^H\Theta\right)^{-1/2}\left(I - \Theta^H\Omega\right)\left(I - \Psi^H\Omega\right)^{-1}\left(I - \Psi^H\Theta\right)\left(I - \Theta^H\Theta\right)^{-1/2}$$
$$\left(I - \Theta^H\Theta\right)^{1/2}\left(I - \Psi^H\Theta\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Theta\Omega^H\right)^{-1}\left(I - \Theta\Theta^H\right)^{1/2}$$
$$= \left(I - \Theta\Theta^H\right)^{-1/2}\left(I - \Theta\Omega^H\right)\left(I - \Omega\Omega^H\right)^{-1}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1} \tag{E.171}$$
$$\left(I - \Omega^H\Omega\right)\left(I - \Psi^H\Omega\right)^{-1}\left(\Psi^H - \Omega^H\right)\left(I - \Theta\Omega^H\right)^{-1}\left(I - \Theta\Theta^H\right)^{1/2}$$

According to Equation (E.171), the matrix $A^2BCD^2C^HB^H$ is similar to the following matrix:

$$\left(I-\Omega\Omega^H\right)^{-1}\left(\Psi-\Omega\right)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)\left(I-\Psi^H\Omega\right)^{-1}\left(\Psi^H-\Omega^H\right).$$

This matrix is also similar to the following product:

$$\left(I-\Omega\Omega^H\right)^{-1/2}\left(\Psi-\Omega\right)\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} \tag{E.172}$$

$$\left(I-\Omega^H\Omega\right)^{1/2}\left(I-\Psi^H\Omega\right)^{-1}\left(\Psi^H-\Omega^H\right)\left(I-\Omega\Omega^H\right)^{-1/2} \tag{E.173}$$

$$=\Phi_\Omega\left(\Psi\right)\Phi_\Omega\left(\Psi\right)^H \tag{E.174}$$

$$=C_2. \tag{E.175}$$

Since the similarity relation is an equivalence relation, we have shown that the matrices C_1 and C_2 are similar, which concludes the proof.

□

We now give a second proof of Theorem 9. We recall that the distance between two points Ω and Ψ in a Riemannian manifold $\mathcal{M}$ is defined as the infimum of the lengths of the paths between Ω and Ψ:

$$d\left(\Omega,\Psi\right):=\inf_\gamma\int_0^1\left\|\gamma'(t)\right\|_{\gamma(t)}dt$$

where γ is a path from Ω to Ψ in $\mathcal{M}$ such that $\gamma(0)=\Omega$ and $\gamma(1)=\Psi$. To prove that an application Φ is an isometry, i.e. $d\left(\Omega,\Psi\right)=d\left(\Phi\left(\Omega\right),\Phi\left(\Psi\right)\right)$ for all $\left(\Omega,\Psi\right)\in\mathcal{M}^2$, it is therefore sufficient to prove that:

$$\left\|D\Phi\left(\Psi\right)[v]\right\|_{\Phi(\Psi)}=\left\|v\right\|_\Psi\quad\forall\Psi\in\mathcal{M},\forall v\in\mathcal{T}_\Psi\mathcal{M}.$$

Proof. In this second proof, we will show that:

$$\left\|D\Phi_\Omega\left(\Psi\right)[v]\right\|^2_{\Phi_\Omega(\Psi)}=\left\|v\right\|^2_\Psi\quad\forall\left(\Omega,\Psi\right)\in\mathcal{SD}^2_N,\forall v\in\mathbb{C}^{N\times N} \tag{E.176}$$

to prove that the application Φ is an isometry.

We first recall the expression of the square norm of the tangent vector v at point Ψ:

$$\left\|v\right\|^2_\Psi=\mathrm{trace}\left(\left(I-\Psi\Psi^H\right)^{-1}v\left(I-\Psi^H\Psi\right)^{-1}v^H\right) \tag{E.177}$$

Therefore, we have:

$$\left\|D\Phi_\Omega\left(\Psi\right)[v]\right\|_{\Phi_\Omega(\Psi)}=\mathrm{trace}\left(\left(I-\Phi_\Omega\left(\Psi\right)\Phi_\Omega\left(\Psi\right)^H\right)^{-1}D\Phi_\Omega\left(\Psi\right)[v]\left(I-\Phi_\Omega\left(\Psi\right)^H\Phi_\Omega\left(\Psi\right)\right)^{-1}D\Phi_\Omega\left(\Psi\right)[v]^H\right) \tag{E.178}$$

We recall that:

$$D\Phi_\Omega\left(\Psi\right)[v]=\left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}v\left(I-\Omega^H\Psi\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}. \tag{E.179}$$

Using the conjugate transpose operator in Equation (E.179), we obtain:

$$D\Phi_\Omega\left(\Psi\right)[v]^H=\left(I-\Omega^H\Omega\right)^{1/2}\left(I-\Psi^H\Omega\right)^{-1}v^H\left(I-\Omega\Psi^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2}. \tag{E.180}$$

According to Property 51, we have:

$$I-\Phi_\Omega\left(\Psi\right)\Phi_\Omega\left(\Psi\right)^H=\left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Psi\Psi^H\right)\left(I-\Psi\Omega^H\right)^{-1}\left(I-\Omega\Omega^H\right)^{1/2}. \tag{E.181}$$

Hence, we obtain:

$$\left(I-\Phi_\Omega\left(\Psi\right)\Phi_\Omega\left(\Psi\right)^H\right)^{-1}=\left(I-\Omega\Omega^H\right)^{-1/2}\left(I-\Psi\Omega^H\right)\left(I-\Psi\Psi^H\right)^{-1}\left(I-\Psi\Omega^H\right)\left(I-\Omega\Omega^H\right)^{-1/2}. \tag{E.182}$$

Using Property 50, we obtain performing the transformations $\Omega\leftarrow\Omega^H$ and $\Psi\leftarrow\Psi^H$ in Equation (E.181):

$$I-\Phi_\Omega\left(\Psi\right)^H\Phi_\Omega\left(\Psi\right)=\left(I-\Omega^H\Omega\right)^{1/2}\left(I-\Psi^H\Omega\right)^{-1}\left(I-\Psi^H\Psi\right)\left(I-\Psi^H\Omega\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2}. \tag{E.183}$$

Hence, we obain:

$$\left(I-\Phi_{\Omega}\left(\Psi\right)^{H}\Phi_{\Omega}\left(\Psi\right)\right)^{-1}=\left(I-\Omega^{H}\Omega\right)^{-1/2}\left(I-\Psi^{H}\Omega\right)\left(I-\Psi^{H}\Psi\right)^{-1}\left(I-\Psi^{H}\Omega\right)\left(I-\Omega^{H}\Omega\right)^{-1/2}. \tag{E.184}$$

Using Equations (E.179), (E.180), (E.182) and (E.184), we obtain from Equation (E.178):

$$\begin{aligned}
&\left\|D\Phi_{\Omega}\left(\Psi\right)\left[v\right]\right\|_{\Phi_{\Omega}(\Psi)} && \text{(E.185)}\\
&=\operatorname{trace}\left(\left(I-\Phi_{\Omega}\left(\Psi\right)\Phi_{\Omega}\left(\Psi\right)^{H}\right)^{-1}D\Phi_{\Omega}\left(\Psi\right)\left[v\right]\left(I-\Phi_{\Omega}\left(\Psi\right)^{H}\Phi_{\Omega}\left(\Psi\right)\right)^{-1}D\Phi_{\Omega}\left(\Psi\right)\left[v\right]^{H}\right) && \text{(E.186)}\\
&=\operatorname{trace}\begin{pmatrix}
\left(I-\Omega\Omega^{H}\right)^{-1/2} & \left(I-\Psi\Omega^{H}\right) & \left(I-\Psi\Psi^{H}\right)^{-1} & \left(I-\Psi\Omega^{H}\right) & \left(I-\Omega\Omega^{H}\right)^{-1/2}\\
\left(I-\Omega\Omega^{H}\right)^{1/2} & \left(I-\Psi\Omega^{H}\right)^{-1} & v & \left(I-\Omega^{H}\Psi\right)^{-1} & \left(I-\Omega^{H}\Omega\right)^{1/2}\\
\left(I-\Omega^{H}\Omega\right)^{-1/2} & \left(I-\Psi^{H}\Omega\right) & \left(I-\Psi^{H}\Psi\right)^{-1} & \left(I-\Psi^{H}\Omega\right) & \left(I-\Omega^{H}\Omega\right)^{-1/2}\\
\left(I-\Omega^{H}\Omega\right)^{1/2} & \left(I-\Psi^{H}\Omega\right)^{-1} & v^{H} & \left(I-\Omega\Psi^{H}\right)^{-1} & \left(I-\Omega\Omega^{H}\right)^{1/2}
\end{pmatrix} && \text{(E.187)}\\
&=\operatorname{trace}\left(\left(I-\Psi\Psi^{H}\right)^{-1}v\left(I-\Psi^{H}\Psi\right)^{-1}v^{H}\right) && \text{(E.188)}\\
&=\left\|v\right\|_{\Psi}^{2} && \text{(E.189)}
\end{aligned}$$

□

E.4 The Riemannian logarithm map

E.4.1 Riemannian logarithm map at 0

In the article of Ben Jeuris and Raf Vandebril [45], the logarithm map at 0 is given by the formula :

$$\log_{0}\left(\Omega\right)=+\mathcal{V}\,\Omega \tag{E.190}$$

with:

$$\mathcal{V}=\mathscr{L}\left(\left(\Omega\Omega^{H}\right)^{1/2},\,\log\left(\frac{I+\left(\Omega\Omega^{H}\right)^{1/2}}{I-\left(\Omega\Omega^{H}\right)^{1/2}}\right)\right) \tag{E.191}$$

where $\mathscr{L}\left(A,Q\right)$ is defined as the solution of:

$$AZ+ZA^{H}=Q \tag{E.192}$$

However, the expression of the logarithm map at zero given in [45] can be greatly simplified.

Property 53. *The Riemannian logarithm map of the Siegel disk at zero has the following expression:*

$$\log_{0}\left(\Omega\right)=\operatorname{arctanh}\left(X\right)X^{-1}\Omega\quad\textit{where }X=\left(\Omega\Omega^{H}\right)^{1/2} \tag{E.193}$$

Note that when $\left\|\left|X\right|\right\|<1$, we can express the function $\operatorname{arctanh}$ as a Taylor series:

$$\operatorname{arctanh}\left(X\right)=\sum_{n=0}^{+\infty}\frac{X^{2n+1}}{2n+1} \tag{E.194}$$

Hence for all X in the Siegel disk, we can write the product $\operatorname{arctanh}\left(X\right)X^{-1}$ the following way:

$$\operatorname{arctanh}\left(X\right)X^{-1}=\sum_{n=0}^{+\infty}\frac{X^{2n}}{2n+1} \tag{E.195}$$

This new expression is also valid when the matrix X is not invertible.

Proof. We set: $X=\left(\Omega\Omega^{H}\right)^{1/2}$.
Therefore, $\mathcal{V}$ is the solution of the equation:

$$X\mathcal{V}+\mathcal{V}X^{H}=\log\left(\frac{I+X}{I-X}\right) \tag{E.196}$$

Note that $X^H = X$ and that $\log\left(\frac{I+X}{I-X}\right)$ can be written as a power series expansion of X as the singular values of X are lower than one by definition of the Siegel domain:

$$\log\left(\frac{I+X}{I-X}\right) = \log\left(I+X\right) - \log\left(I-X\right) \tag{E.197}$$

$$= \sum_{n=1}^{+\infty} \frac{(-1)^{n+1}}{n} X^n - \left(-\sum_{n=1}^{+\infty} \frac{1}{n} X^n\right) \tag{E.198}$$

$$= 2\sum_{n=0}^{+\infty} \frac{1}{2n+1} X^{2n+1} \tag{E.199}$$

Hence the expression $\log\left(\frac{I+X}{I-X}\right)$ and X commute and $\mathcal{V}$ can be expressed explicitly as a function of X:

$$\mathcal{V} = \frac{1}{2}\log\left(\frac{I+X}{I-X}\right) X^{-1} \tag{E.200}$$

Hence, we proved that:

$$\log_0\left(\Omega\right) = \frac{1}{2}\log\left(\frac{I+X}{I-X}\right) X^{-1}\Omega \quad \text{where } X = \left(\Omega\Omega^H\right)^{1/2} \tag{E.201}$$

The reciprocal application of the hyperbolic tangent as the following expression:

$$\operatorname{arctanh}\left(X\right) = \frac{1}{2}\log\left(\frac{I+X}{I-X}\right) \quad \text{for } \left|\!\left|\!\left|X\right|\!\right|\!\right| < 1 \tag{E.202}$$

Finally, we proved that:

$$\log_0\left(\Omega\right) = \operatorname{arctanh}\left(X\right) X^{-1}\Omega \quad \text{where } X = \left(\Omega\Omega^H\right)^{1/2} \tag{E.203}$$

□

For consistency, we can check that:

$$d^2_{\mathcal{SD}_N}\left(0,\Omega\right) = \left\|\log_0\left(\Omega\right)\right\|_0^2 \tag{E.204}$$

On the one hand, we obtain from the expression of the distance (E.4):

$$d^2\left(0,\Omega\right) = \operatorname{trace}\left(\operatorname{arctanh}^2\left(C\right)\right) \quad \text{with } C = \left(\Omega\Omega^H\right)^{1/2} \tag{E.205}$$

One the other hand, we obtain using the expression of the norm (E.8):

$$\left\|\log_0\left(\Omega\right)\right\|_0^2 = \operatorname{trace}\left(\log_0\left(\Omega\right)\log_0\left(\Omega\right)^H\right) \tag{E.206}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}\left(X\right) X^{-1}\Omega\Omega^H X^{-1\ H} \operatorname{arctanh}\left(X\right)^H\right) \tag{E.207}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}\left(X\right) X^{-1} X^2 X^{-1} \operatorname{arctanh}\left(X\right)\right) \tag{E.208}$$

$$= \operatorname{trace}\left(\operatorname{arctanh}^2\left(X\right)\right) \tag{E.209}$$

where $X = \left(\Omega\Omega^H\right)^{1/2}$.
Therefore, we checked that $d^2_{\mathcal{SD}_N}\left(0,\Omega\right) = \left\|\log_0\left(\Omega\right)\right\|_0^2$.

E.4.2 The Riemannian logarithm map at any point

To compute the Riemannian logarithm map at a point Ω the key idea here is to transport the problem at zero, compute a certain logarithm at zero and transport the result back to Ω. This idea is illustrated on Figure E.1. If we want to compute the logartithm map: $\log_\Omega\left(\Psi\right)$, we first transport both Ω and Ψ using the isometry Φ_Ω (E.9). The point Ω is sent to zero, and we denote $\Psi^{'}$ the image of Ψ by Φ_Ω :

$$\Psi^{'} := \Phi_\Omega\left(\Psi\right) = \left(I - \Omega\Omega^H\right)^{-1/2}\left(\Psi - \Omega\right)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.210}$$

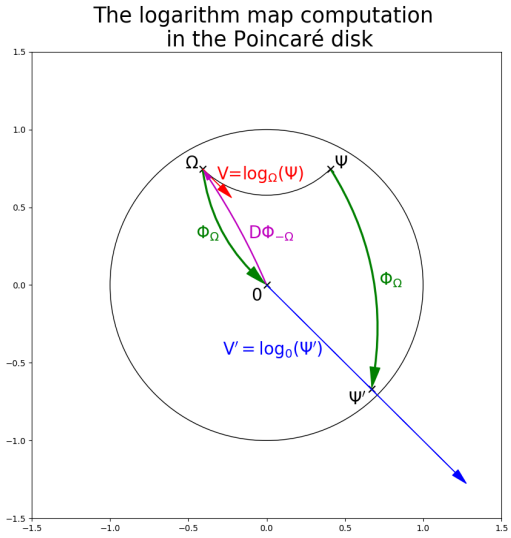

Figure E.1: The Poincaré disk logarithm map computation

Then we compute the logarithm map at zero $\log_0\left(\Psi^{'}\right)$:

$$V^{'} := \log_0\left(\Psi^{'}\right) = \operatorname{arctanh}\left(X\right) X^{-1}\Psi^{'} \quad \text{where } X = \left(\Psi^{'}\Psi^{'\,H}\right)^{1/2} \tag{E.211}$$

And finally, we transport back the logarithm to the point Ω using the differential of the isometry Φ given in Equation (E.17):

$$V := \log_\Omega\left(\Psi\right) = D\Phi_{-\Omega}\left(0\right)\left[V^{'}\right] = \left(I - \Omega\Omega^H\right)^{1/2} V^{'} \left(I - \Omega^H\Omega\right)^{1/2} \tag{E.212}$$

Correspondance between the Siegel disk in dimension one and the Poincaré disk for the logarithm map We can check that the logarithm map of the Siegel disk of dimension one corresponds to the logarithm map of the Poincaré disk described in Section 4.1.2.

We compute the logarithm map $\log_\Omega\left(\Psi\right)$ of the Siegel disk of dimension one using the three steps described in Section E.4.2.

We start by computing the point $\Psi^{'}$:

$$\begin{aligned}
\Psi^{'} =& \Phi_\Omega\left(\Psi\right) && \text{(E.213)}\\
=& \left(I - \Omega\Omega^H\right)^{-1/2}\left(\Psi - \Omega\right)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} && \text{(E.214)}\\
=& \frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi} && \text{(E.215)}
\end{aligned}$$

Then we compute the logarithm map at zero:

$$\begin{aligned}
V^{'} =& \log_0\left(\Psi^{'}\right) && \text{(E.216)}\\
=& \operatorname{arctanh}\left(X\right) X^{-1}\Psi^{'} \quad \text{where } X = \left(\Psi^{'}\Psi^{'\,H}\right)^{1/2} = \left|\Psi^{'}\right| && \text{(E.217)}\\
=& \operatorname{arctanh}\left(\left|\Psi^{'}\right|\right)\frac{\Psi^{'}}{\left|\Psi^{'}\right|} && \text{(E.218)}\\
=& \operatorname{arctanh}\left(\left|\Psi^{'}\right|\right) e^{i\theta} \quad \text{with } \theta = \arg\left(\Psi^{'}\right) && \text{(E.219)}\\
=& \operatorname{arctanh}\left(\left|\frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi}\right|\right) e^{i\theta} \quad \text{with } \theta = \arg\left(\frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi}\right) && \text{(E.220)}
\end{aligned}$$

And finally, we compute the logarithm map $\log_\Omega\left(\Psi\right)$:

$$\log_{\Omega}(\Psi) = D\Phi_{-\Omega}(0)\left[V'\right] \tag{E.221}$$
$$= \left(I - \Omega\Omega^H\right)^{1/2} V' \left(I - \Omega^H\Omega\right)^{1/2} \tag{E.222}$$
$$= \left(1 - |\Omega|^2\right) V' \tag{E.223}$$
$$= \left(1 - |\Omega|^2\right) \operatorname{arctanh}\left(\left|\frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi}\right|\right) e^{i\theta} \quad \text{with } \theta = \arg\left(\frac{\Psi - \Omega}{1 - \bar{\Omega}\Psi}\right) \tag{E.224}$$

This result corresponds to the expression of the Poincaré disk logarithm map described in Equation (4.18).

E.5 The Riemannian exponential map

E.5.1 Riemannian exponential map at 0

Property 54. *The Riemannian exponential map of the Siegel disk at zero has the following expression:*

$$\exp_0(V) = \tanh(Y)\, Y^{-1} V \quad \textit{where } Y = \left(VV^H\right)^{1/2} \tag{E.225}$$

We recall the definition of the hyperbolic tangent:

$$\tanh(X) = \frac{\sinh(X)}{\cosh(X)} = \frac{e^X - e^{-X}}{e^X + e^{-X}} = \frac{e^{2X} - 1}{e^{2X} + 1} = \frac{1 - e^{-2X}}{1 + e^{-2X}} \tag{E.226}$$

Proof. To prove this property, we will use the fact that the Riemannian exponential map at zero is the inverse of the Riemannian logarithm map at zero.

We set: $V = \log_0(\Omega)$. Hence $\Omega = \exp_0(V)$. According to Property 53, we have:

$$V = \operatorname{arctanh}(X)\, X^{-1}\Omega \quad \text{where } X = \left(\Omega\Omega^H\right)^{1/2} \tag{E.227}$$

We recall that $X = X^H$.

To express Ω as a function of V, we first calculate the product VV^H:

$$VV^H = \operatorname{arctanh}(X)\, X^{-1}\Omega\Omega^H X^{-1\,H} \operatorname{arctanh}(X)^H \tag{E.228}$$
$$= \operatorname{arctanh}(X)\, X^{-1} X^2 X^{-1} \operatorname{arctanh}(X) \tag{E.229}$$
$$= \operatorname{arctanh}^2(X) \tag{E.230}$$

Hence:

$$\operatorname{arctanh}(X) = \left(VV^H\right)^{1/2} \tag{E.231}$$

Therefore:

$$X = \tanh\left(\left(VV^H\right)^{1/2}\right) \tag{E.232}$$

We will note:

$$X = \tanh(Y) \quad \text{with } Y = \left(VV^H\right)^{1/2} \tag{E.233}$$

Using Equation (E.227) we can write:

$$\Omega = X\left(\operatorname{arctanh}(X)\right)^{-1} V \tag{E.234}$$

Using Equation (E.233), we otain:

$$\Omega = \tanh(Y)\left(\operatorname{arctanh}(\tanh(Y))\right)^{-1} V \tag{E.235}$$
$$= \tanh(Y)\, Y^{-1} V \tag{E.236}$$

We finally proved that:

$$\exp_0(V) = \tanh(Y)\, Y^{-1} V \quad \text{where } Y = \left(VV^H\right)^{1/2} \tag{E.237}$$

□

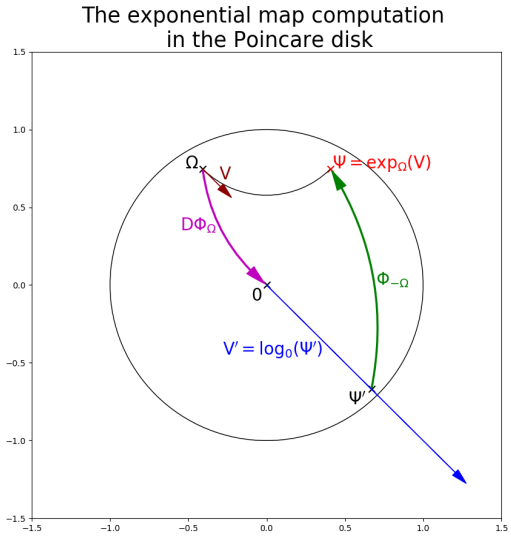

Figure E.2: The Poincaré disk exponential map computation

E.5.2 The Riemannian exponential map at any point

To compute the Riemannian exponential map at a point Ω the key idea here is to transport the problem at zero (as for the logarithm), compute a certain exponential at zero and transport the result back to Ω. This idea is illustrated on Figure E.2. If we want to compute the exponential map: $\exp_\Omega(V)$, we first transport the vector V at zero using the differential of the isometry Φ given in Equation (E.17):

$$
\begin{aligned}
V^{'} :=& D\Phi_\Omega(\Omega)[V] && \text{(E.238)}\\
=& \left(I-\Omega\Omega^H\right)^{1/2}\left(I-\Omega\Omega^H\right)^{-1} V \left(I-\Omega^H\Omega\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} && \text{(E.239)}\\
=& \left(I-\Omega\Omega^H\right)^{-1/2} V \left(I-\Omega^H\Omega\right)^{-1/2} && \text{(E.240)}
\end{aligned}
$$

Then we compute the exponential map at zero $\exp_0\left(V^{'}\right)$:

$$
\Psi^{'} := \exp_0\left(V^{'}\right) = \tanh(Y)\, Y^{-1} V^{'} \quad \text{where } Y = \left(V^{'} V^{'\,H}\right)^{1/2} \tag{E.241}
$$

And finally, we transport back the exponential to the point Ω using the isometry $\Phi_{-\Omega}$ which is the inverse of isometry Φ_Ω (see Property 49) and transport the point 0 back to Ω and the point $\Psi^{'}$ back to $\exp_\Omega(V)$:

$$
\begin{aligned}
\Psi :=& \exp_\Omega(V) && \text{(E.242)}\\
=& \Phi_{-\Omega}\left(\Psi^{'}\right) && \text{(E.243)}\\
=& \left(I-\Omega\Omega^H\right)^{-1/2}\left(\Psi^{'}+\Omega\right)\left(I+\Omega^H\Psi^{'}\right)^{-1}\left(I-\Omega^H\Omega\right)^{1/2} && \text{(E.244)}
\end{aligned}
$$

Correspondance between the Siegel disk in dimension one and the Poincaré disk for the exponential map We can check that the exponential map of the Siegel disk of dimension one corresponds to the exponential map of the Poincaré disk described in Section 4.1.2.

We compute the exponential map $\exp_\Omega(V)$ of the Siegel disk of dimension one using the three steps described in Section E.5.2.

We start by computing the point $V^{'}$:

$$
\begin{aligned}
V^{'} =& \left(I-\Omega\Omega^H\right)^{-1/2} V \left(I-\Omega^H\Omega\right)^{-1/2} && \text{(E.245)}\\
=& \frac{V}{1-|\Omega|^2} && \text{(E.246)}
\end{aligned}
$$

Then we compute the exponential map at zero:

$$\Psi^{'} = \exp_0\left(V^{'}\right) \tag{E.247}$$

$$= \tanh\left(Y\right) Y^{-1} V^{'} \quad \text{where } Y = \left(V^{'} V^{' H}\right)^{1/2} = \left|V^{'}\right| \tag{E.248}$$

$$= \tanh\left(\left|V^{'}\right|\right) \frac{V^{'}}{|V^{'}|} \tag{E.249}$$

$$= \tanh\left(\left|V^{'}\right|\right) e^{i\theta} \quad \text{with } \theta = \arg\left(V^{'}\right) = \arg\left(V\right) \tag{E.250}$$

$$= \tanh\left(\frac{|V|}{1-|\Omega|^2}\right) e^{i\theta} \quad \text{with } \theta = \arg\left(V\right) \tag{E.251}$$

Note that for all x in $\mathbb{R}$, $\tanh(x) = \frac{\exp(2x)-1}{\exp(2x)+1}$, hence we can also write:

$$\Psi^{'} = \frac{\exp\left(\frac{2|V|}{I-|\Omega|^2}\right) - 1}{\exp\left(\frac{2|V|}{1-|\Omega|^2}\right) + 1} e^{i\theta} \quad \text{with } \theta = \arg\left(V\right) \tag{E.252}$$

Finally, we compute the exponential map $\exp_\Omega\left(V\right)$:

$$\exp_\Omega\left(V\right) = \Phi_{-\Omega}\left(\Psi^{'}\right) \tag{E.253}$$

$$= \left(I - \Omega\Omega^H\right)^{-1/2} \left(\Psi^{'} + \Omega\right) \left(I + \Omega^H \Psi^{'}\right)^{-1} \left(I - \Omega^H\Omega\right)^{1/2} \tag{E.254}$$

$$= \frac{\Psi^{'} + \Omega}{1 + \bar{\Omega}\Psi'} \tag{E.255}$$

$$= \frac{\frac{\exp\left(\frac{2|V|}{1-|\Omega|^2}\right)-1}{\exp\left(\frac{2|V|}{1-|\Omega|^2}\right)+1} e^{i\theta} + \Omega}{1 + \bar{\Omega}\frac{\exp\left(\frac{2|V|}{1-|\Omega|^2}\right)-1}{\exp\left(\frac{2|V|}{1-|\Omega|^2}\right)+1} e^{i\theta}} \quad \text{with } \theta = \arg\left(V\right) \tag{E.256}$$

$$= \frac{\left(\exp\left(\frac{2|V|}{1-|\Omega|^2}\right) - 1\right) e^{i\theta} + \left(\exp\left(\frac{2|V|}{1-|\Omega|^2}\right) + 1\right) \Omega}{\left(\exp\left(\frac{2|V|}{1-|\Omega|^2}\right) + 1\right) + \bar{\Omega}\left(\exp\left(\frac{2|V|}{1-|\Omega|^2}\right) - 1\right) e^{i\theta}} \quad \text{with } \theta = \arg\left(V\right) \tag{E.257}$$

$$= \frac{\left(\Omega + e^{i\theta}\right) \exp\left(\frac{2|V|}{1-|\Omega|^2}\right) + \left(\Omega - e^{i\theta}\right)}{\left(1 + \bar{\Omega}e^{i\theta}\right) \exp\left(\frac{2|V|}{1-|\Omega|^2}\right) + \left(1 - \bar{\Omega}e^{i\theta}\right)} \quad \text{with } \theta = \arg\left(V\right) \tag{E.258}$$

This result corresponds to the expression of the Poincaré disk exponential map described in Section 4.1.2.

The geodesics The expression the geodesics can be obtained using the exponential map: the equation of the geodesic starting from Ω with velocity V is given by the application $\zeta(t) : t \mapsto \exp_\Omega\left(tV\right)$.

E.6 The symmetric Siegel disk

We defined the Siegel disk in Definition 16 as the set a complex matrices with singular values lower than one: $\mathcal{SD}_N = \left\{M \in \mathbb{C}^N, I - MM^H > 0\right\}$. We recall that another definition of the Siegel disk also exists in other papers [54], imposing an additional symmetry condition on the matrix M: $M = M^T$. We will show in this section that the symmetric Siegel disk is a totally flat submanifold of the Siegel disk. Hence the formula of the logarithm map E.4.2 and the exponential map E.5.2 computed in previous sections are still meaningful when working in the submanifold of symmetric matrices.

The isometry We first check that the isometry E.9 is meaninful in the symmetric Siegel disk.

Property 55. *If Ω and Ψ are symmetric matrices, then $\Phi_\Omega\left(\Psi\right)$ is also a symmetric matrix.*

Proof. We assume that Ω and Ψ are symmetric matrices.

Then, the expression of Φ becomes:

$$\Phi_\Omega(\Psi) = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.259}$$
$$= \left(I - \Omega\bar{\Omega}\right)^{-1/2}(\Psi - \Omega)\left(I - \bar{\Omega}\Psi\right)^{-1}\left(I - \bar{\Omega}\Omega\right)^{1/2} \tag{E.260}$$

From Equation (E.78), we also have:

$$\Phi_\Omega(\Psi) = \Phi_{-\Omega}^{-1}(\Psi) \tag{E.261}$$
$$= \left(I - \Omega\Omega^H\right)^{1/2}\left(I - \Psi\Omega^H\right)^{-1}(\Psi - \Omega)\left(I - \Omega^H\Omega\right)^{-1/2} \tag{E.262}$$
$$= \left(I - \Omega\bar{\Omega}\right)^{1/2}\left(I - \Psi\bar{\Omega}\right)^{-1}(\Psi - \Omega)\left(I - \bar{\Omega}\Omega\right)^{-1/2} \tag{E.263}$$

Using both expressions of Φ_Ω:

$$\Phi_\Omega(\Psi)^T = \Phi_{-\Omega}^{-1}(\Psi)^T \tag{E.264}$$
$$= \left(I - \Omega\bar{\Omega}\right)^{-1/2}(\Psi - \Omega)\left(I - \bar{\Omega}\Psi\right)^{-1}\left(I - \bar{\Omega}\Omega\right)^{1/2} \tag{E.265}$$
$$= \Phi_\Omega(\Psi) \tag{E.266}$$

which proves the property.

□

We now check that the differential of the isometry Φ is meaningful in the symmetric Siegel space. The differential of the isometry Φ has been used to compute the logarithm E.4.2 and the exponential map. E.5.2 of the Siegel disk. Note that the tangent space of the symmetric Siegel disk is the space of complex symmetric matrices.

Property 56. *If Ω, Ψ and h are symmetric matrices, then $D\Phi_\Omega(\Psi)[h]$ is also a symmetric matrix.*

Proof. We assume that Ω, Ψ and h are symmetric matrices.

Using Equation (E.17), we can write:

$$D\Phi_\Omega(\Psi)[h] = \left(I - \Omega\Omega^H\right)^{1/2}\left(I - \Psi\Omega^H\right)^{-1}h\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.267}$$
$$= \left(I - \Omega\bar{\Omega}\right)^{1/2}\left(I - \Psi\bar{\Omega}\right)^{-1}h\left(I - \bar{\Omega}\Psi\right)^{-1}\left(I - \bar{\Omega}\Omega\right)^{1/2} \tag{E.268}$$

Then, we have:

$$D\Phi_\Omega(\Psi)[h]^T = \left(I - \Omega\bar{\Omega}\right)^{1/2}\left(I - \Psi\bar{\Omega}\right)^{-1}h\left(I - \bar{\Omega}\Psi\right)^{-1}\left(I - \bar{\Omega}\Omega\right)^{1/2} \tag{E.269}$$
$$= D\Phi_\Omega(\Psi)[h] \tag{E.270}$$

which proves the property.

□

The Riemannian logarithm map

Property 57. *If Ω and Ψ are symmetric matrices, then $\log_\Omega(\Psi)$ is also a symmetric matrix.*

Proof. Let Ω and Ψ be symmetric matrices. There are three steps to compute the Riemannian logarithm map. They are described in Section E.4.2.

The first step is to compute the point Ψ':

$$\Psi' := \Phi_\Omega(\Psi) = \left(I - \Omega\Omega^H\right)^{-1/2}(\Psi - \Omega)\left(I - \Omega^H\Psi\right)^{-1}\left(I - \Omega^H\Omega\right)^{1/2} \tag{E.271}$$

The matrix Ψ' is therefore symmetric, thanks to Property 55.

The second step is to compute the logarithm map at zero $\log_0\left(\Psi'\right)$:

$$V' := \log_0\left(\Psi'\right) = \operatorname{arctanh}(X)\,X^{-1}\Psi' \quad \text{where } X = \left(\Psi'\Psi'^H\right)^{1/2} \tag{E.272}$$

As $\|\|X\|\| < 1$, we can express $\operatorname{arctanh}(X)$ as a Taylor series:

$$\operatorname{arctanh}(X) = \sum_{n=0}^{+\infty}\frac{1}{2n+1}X^{2n+1} \tag{E.273}$$

Note that $\operatorname{arctanh}(X)$ is an odd application. Hence, $\operatorname{arctanh}(X) X^{-1}$ is an even application:

$$\operatorname{arctanh}(X) X^{-1} = \sum_{n=0}^{+\infty} \frac{1}{2n+1} X^{2n} \tag{E.274}$$

Hence we can write $V^{'}$ the following way:

$$V^{'} = \operatorname{arctanh}(X) X^{-1} \Psi^{'} \quad \text{where } X = \left(\Psi^{'} \Psi^{'\,H}\right)^{1/2} = \left(\Psi^{'} \bar{\Psi}^{'}\right)^{1/2} \tag{E.275}$$

$$= \left(\sum_{n=0}^{+\infty} \frac{1}{2n+1} X^{2n}\right) \Psi^{'} \quad \text{where } X = \left(\Psi^{'} \bar{\Psi}^{'}\right)^{1/2} \tag{E.276}$$

$$= \sum_{n=0}^{+\infty} \frac{1}{2n+1} \left(\left(\Psi^{'} \bar{\Psi}^{'}\right)^{1/2}\right)^{2n} \Psi^{'} \tag{E.277}$$

$$= \sum_{n=0}^{+\infty} \frac{1}{2n+1} \left(\Psi^{'} \bar{\Psi}^{'}\right)^{n} \Psi^{'} \tag{E.278}$$

Therefore, we have:

$$V^{'\,T} = \left(\sum_{n=0}^{+\infty} \frac{1}{2n+1} \left(\Psi^{'} \bar{\Psi}^{'}\right)^{n} \Psi^{'}\right)^{T} \tag{E.279}$$

$$= \sum_{n=0}^{+\infty} \frac{1}{2n+1} \Psi^{'T} \left(\bar{\Psi}^{'T} \Psi^{'T}\right)^{n} \tag{E.280}$$

$$= \sum_{n=0}^{+\infty} \frac{1}{2n+1} \Psi^{'} \left(\bar{\Psi}^{'} \Psi^{'}\right)^{n} \tag{E.281}$$

$$= \sum_{n=0}^{+\infty} \frac{1}{2n+1} \left(\Psi^{'} \bar{\Psi}^{'}\right)^{n} \Psi^{'} \tag{E.282}$$

$$= V^{'} \tag{E.283}$$

which proves that the logarithm map at zero $\log_0\left(\Psi^{'}\right)$ is symmetric.

The third and last step is to transport back the logarithm to the point Ω using the differential of the isometry Φ given in Equation (E.17):

$$V := \log_{\Omega}(\Psi) = D\Phi_{-\Omega}(0)\left[V^{'}\right] = \left(I - \Omega\Omega^{H}\right)^{1/2} V^{'} \left(I - \Omega^{H}\Omega\right)^{1/2} \tag{E.284}$$

As Ω and $V^{'}$ are symmetric matrices, we finally prove that the logarithm map $\log_{\Omega}(\Psi)$ is a symmetric matrix using Property 56.

□

The Riemannian exponential map

Property 58. *If Ω and V are symmetric matrices, then $\exp_{\Omega}(V)$ is also a symmetric matrix.*

Proof. Let Ω and V be symmetric matrices. There are three steps to compute the Riemannian exponential map. They are described in Section E.5.2.

The first step is to transport the vector V at zero using the differential of the isometry Φ given in Equation (E.17):

$$V^{'} := D\Phi_{\Omega}(\Omega)[V] \tag{E.285}$$

$$= \left(I - \Omega\Omega^{H}\right)^{-1/2} V \left(I - \Omega^{H}\Omega\right)^{-1/2} \tag{E.286}$$

The matrix $V^{'}$ is therefore symmetric, thanks to Property 56.

The second step is to compute the exponential map at zero $\exp_0\left(V^{'}\right)$:

$$\Psi^{'} := \exp_0\left(V^{'}\right) = \tanh(Y) Y^{-1} V^{'} \quad \text{where } Y = \left(V^{'} V^{'\,H}\right)^{1/2} \tag{E.287}$$

As $\|\|\|Y\|\|\| < 1$, we can express $\tanh(Y)$ as a Taylor series:

$$\tanh(Y) = \sum_{n=1}^{+\infty} \frac{2^{2n}\left(2^{2n}-1\right)B_{2n}}{(2n)!} Y^{2n-1} \tag{E.288}$$

where B_n is the n^{th} Bernoulli number.
Note that $\tanh(Y)$ is an odd application. Hence, $\tanh(Y)\,Y^{-1}$ is an even application:

$$\tanh(Y)\,Y^{-1} = \sum_{n=1}^{+\infty} \frac{2^{2n}\left(2^{2n}-1\right)B_{2n}}{(2n)!} Y^{2n-2} \tag{E.289}$$

$$= \sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} Y^{2n} \tag{E.290}$$

Hence we can write $\Psi^{'}$ the following way:

$$\Psi^{'} = \tanh(Y)\,Y^{-1}V^{'} \quad \text{where } Y = \left(V^{'}V^{'H}\right)^{1/2} = \left(V^{'}\bar{V^{'}}\right)^{1/2} \tag{E.291}$$

$$= \left(\sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} Y^{2n}\right) V^{'} \quad \text{where } Y = \left(V^{'}\bar{V^{'}}\right)^{1/2} \tag{E.292}$$

$$= \sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} \left(\left(V^{'}\bar{V^{'}}\right)^{1/2}\right)^{2n} V^{'} \tag{E.293}$$

$$= \sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} \left(V^{'}\bar{V^{'}}\right)^{n} V^{'} \tag{E.294}$$

Therefore, we have:

$$\Psi^{'T} = \left(\sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} \left(V^{'}\bar{V^{'}}\right)^{n} V^{'}\right)^{T} \tag{E.295}$$

$$= \sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} V^{'T} \left(\bar{V^{'}}^{T} V^{'T}\right)^{n} \tag{E.296}$$

$$= \sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} V^{'} \left(\bar{V^{'}}V^{'}\right)^{n} \tag{E.297}$$

$$= \sum_{n=0}^{+\infty} \frac{2^{2(n+1)}\left(2^{2(n+1)}-1\right)B_{2(n+1)}}{(2(n+1))!} \left(V^{'}\bar{V^{'}}\right)^{n} V^{'} \tag{E.298}$$

$$= \Psi^{'} \tag{E.299}$$

which proves that the exponential map at zero $\exp_0\left(V^{'}\right)$ is symmetric.

The third and last step is to transport back the exponential to the point Ω using the isometry $\Phi_{-\Omega}$ which is the inverse of isometry Φ_{Ω} (see Property 49) and transport the point 0 back to Ω and the point $\Psi^{'}$ back to $\exp_{\Omega}(V)$:

$$\Psi := \exp_{\Omega}(V) \tag{E.300}$$

$$= \Phi_{-\Omega}\left(\Psi^{'}\right) \tag{E.301}$$

$$= \left(I - \Omega\Omega^{H}\right)^{-1/2}\left(\Psi^{'} + \Omega\right)\left(I + \Omega^{H}\Psi^{'}\right)^{-1}\left(I - \Omega^{H}\Omega\right)^{1/2} \tag{E.302}$$

As Ω and $\Psi^{'}$ are symmetric matrices, we finally prove that the exponential map $\exp_{\Omega}(V)$ is a symmetric matrix using Property 55.

□

E.7 The sectional curvature

We will focus on the sectional curvature at 0. We can then obtain the sectional curvature at any point using the isometry Φ defined in Equation (E.9) and its differential E.17.

Indeed, the sectional curvature at the point Ω defined by the tangent vectors (which are matrices) E_1 and E_2 is equal to the sectional curvature at the point 0 defined by the vectors $D\Phi_\Omega(\Omega)[E_1]$ et $D\Phi_\Omega(\Omega)[E_2]$.

E.7.1 In Euclidean coordinates

Let σ be a section defined at the point $\Omega = 0$ by two orthonormal matrices E_1 and E_2.

Theorem 10. *The sectional curvature at zero of the plan σ defined by E_1 and E_2 has the following expression:*

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1E_2^H - E_2E_1^H\right\|^2 + \left\|E_1^HE_2 - E_2^HE_1\right\|^2\right) \tag{E.303}$$

Proof. We complete the orthonormal matrices E_1 and E_2 in an orthonormal basis $(E_1, \ldots, E_n)$ as described in Section D.1.

We recall the expression of the Siegel metric given in Section E.2:

$$ds^2 = \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H\right) \tag{E.304}$$

We set $\Omega = \exp_0(x_1E_1 + x_2E_2)$.

We recall the expression of the exponential map at zero described in Section E.5.1:

$$\exp_0(V) = \tanh(Y)\, Y^{-1}V \quad \text{where } Y = \left(VV^H\right)^{1/2} \tag{E.305}$$

Here we have $V = x_1E_1 + x_2E_2$ and $Y = \left(VV^H\right)^{1/2} = \left((x_1E_1 + x_2E_2)(x_1E_1 + x_2E_2)^H\right)^{1/2}$.

As the curvature is a local property, we can use a limited of the fonction $\tanh$.

We recall that $\forall z \in \mathbb{C}$ such that $|z| < 2\pi$, we have:

$$tanh(z) = z - \frac{1}{3}z^3 + O\left(z^5\right) \tag{E.306}$$

Hence, as Y is a matrix close to zero:

$$\Omega = \tanh(Y)\, Y^{-1}V \tag{E.307}$$

$$\Omega = \left(I - \frac{1}{3}Y^2 + O\left(Y^4\right)\right)(x_1E_1 + x_2E_2) \tag{E.308}$$

$$\begin{aligned}\Omega = {} & (x_1E_1 + x_2E_2) \\ & - \frac{1}{3}(x_1E_1 + x_2E_2)(x_1E_1 + x_2E_2)^H(x_1E_1 + x_2E_2) \\ & + O\left((|x_1| + |x_2|)^5\right)\end{aligned} \tag{E.309}$$

$$\begin{aligned}\Omega = {} & (x_1E_1 + x_2E_2) \\ & - \frac{1}{3}\left(x_1^3E_1E_1^HE_1 + x_1^2x_2E_1E_1^HE_2 + x_1^2x_2E_1E_2^HE_1 + x_1x_2^2E_1E_2^HE_2 + \right. \\ & \left. \quad x_1^2x_2E_2E_1^HE_1 + x_1x_2^2E_2E_1^HE_2 + x_1x_2^2E_2E_2^HE_1 + x_2^3E_2E_2^HE_2\right) \\ & + O\left((|x_1| + |x_2|)^5\right)\end{aligned} \tag{E.310}$$

Hence, we obtain using the conjugate transpose operator:

$$\begin{aligned}\Omega^H = &\left(x_1 E_1^H + x_2 E_2^H\right) \\ &- \frac{1}{3}\left(x_1 E_1 + x_2 E_2\right)^H \left(x_1 E_1 + x_2 E_2\right)\left(x_1 E_1 + x_2 E_2\right)^H \\ &+ O\left(\left(|x_1| + |x_2|\right)^5\right)\end{aligned} \tag{E.311}$$

$$\begin{aligned}\Omega^H = &\left(x_1 E_1^H + x_2 E_2^H\right) \\ &- \frac{1}{3}\left(x_1^3 E_1^H E_1 E_1^H + x_1^2 x_2 E_1^H E_1 E_2^H + x_1^2 x_2 E_1^H E_2 E_1^H + x_1 x_2^2 E_1^H E_2 E_2^H\right. \\ &\quad \left. + x_1^2 x_2 E_2^H E_1 E_1^H + x_1 x_2^2 E_2^H E_1 E_2^H + x_1 x_2^2 E_2^H E_2 E_1^H + x_2^3 E_2^H E_2 E_2^H\right) \\ &+ O\left(\left(|x_1| + |x_2|\right)^5\right)\end{aligned} \tag{E.312}$$

We now compute the infinitesimal variations:

$$\begin{aligned}d\Omega = &\left(dx_1 E_1 + dx_2 E_2\right) \\ &- \frac{1}{3}\left[\ (3x_1^2 dx_1) E_1 E_1^H E_1 + (2x_1 x_2 dx_1 + x_1^2 dx_2) E_1 E_1^H E_2 + \right. \\ &(2x_1 x_2 dx_1 + x_1^2 dx_2) E_1 E_2^H E_1 + (x_2^2 dx_1 + 2x_1 x_2 dx_2) E_1 E_2^H E_2 + \\ &(2x_1 x_2 dx_1 + x_1^2 dx_2) E_2 E_1^H E_1 + (x_2^2 dx_1 + 2x_1 x_2 dx_2) E_2 E_1^H E_2 + \\ &\left.(x_2^2 dx_1 + 2x_1 x_2 dx_2) E_2 E_2^H E_1 + (3x_2^2 dx_2) E_2 E_2^H E_2\ \right] \\ &+ O\left(\left(|x_1| + |x_2|\right)^4\right)\end{aligned} \tag{E.313}$$

Hence, using the conjugate transpose operator:

$$\begin{aligned}d\Omega^H = &\left(dx_1 E_1^H + dx_2 E_2^H\right) \\ &- \frac{1}{3}\left[\ (3x_1^2 dx_1) E_1^H E_1 E_1^H + (2x_1 x_2 dx_1 + x_1^2 dx_2) E_2^H E_1 E_1^H + \right. \\ &(2x_1 x_2 dx_1 + x_1^2 dx_2) E_1^H E_2 E_1^H + (x_2^2 dx_1 + 2x_1 x_2 dx_2) E_2^H E_2 E_1^H + \\ &(2x_1 x_2 dx_1 + x_1^2 dx_1) E_2^H E_2 E_1^H + (x_2^2 dx_1 + 2x_1 x_2 dx_2) E_2^H E_1 E_2^H + \\ &\left.(x_2^2 dx_1 + 2x_1 x_2 dx_2) E_1^H E_2 E_2^H + (3x_2^2 dx_2) E_2^H E_2 E_2^H\ \right] \\ &+ O\left(\left(|x_1| + |x_2|\right)^4\right)\end{aligned} \tag{E.314}$$

We now compute the products:

$$\Omega\Omega^H = \left[(x_1 E_1 + x_2 E_2) + O\left(\left(|x_1| + |x_2|\right)^3\right)\right]\left[\left(x_1 E_1^H + x_2 E_2^H\right) + O\left(\left(|x_1| + |x_2|\right)^3\right)\right] \tag{E.315}$$

$$\Omega\Omega^H = x_1^2 E_1 E_1^H + x_1 x_2 E_1 E_2^H + x_1 x_2 E_2 E_1^H + x_2^2 E_2 E_2^H + O\left(\left(|x_1| + |x_2|\right)^4\right) \tag{E.316}$$

and:

$$\Omega^H\Omega = \left[\left(x_1 E_1^H + x_2 E_2^H\right) + O\left(\left(|x_1| + |x_2|\right)^3\right)\right]\left[(x_1 E_1 + x_2 E_2) + O\left(\left(|x_1| + |x_2|\right)^3\right)\right] \tag{E.317}$$

$$\Omega^H\Omega = x_1^2 E_1^H E_1 + x_1 x_2 E_1^H E_2 + x_1 x_2 E_2^H E_1 + x_2^2 E_2^H E_2 + O\left(\left(|x_1| + |x_2|\right)^4\right) \tag{E.318}$$

We recall that $\forall X \in M_n(\mathbb{C})$ such that $|||X||| < 1$, we have:

$$(I - X)^{-1} = \sum_{n=0}^{+\infty} X^n \tag{E.319}$$

Hence:

$$(I - X)^{-1} = I + X + O\left(X^2\right) \tag{E.320}$$

Therefore, we can compute:

$$\left(I-\Omega\Omega^H\right)^{-1} = I+\Omega\Omega^H+O\left(\left(\Omega\Omega^H\right)^2\right) \tag{E.321}$$

$$\left(I-\Omega\Omega^H\right)^{-1} = I+x_1^2E_1E_1^H+x_1x_2E_1E_2^H+x_1x_2E_2E_1^H+x_2^2E_2E_2^H+O\left(\left(|x_1|+|x_2|\right)^4\right) \tag{E.322}$$

and:

$$\left(I-\Omega^H\Omega\right)^{-1} = I+\Omega^H\Omega+O\left(\left(\Omega^H\Omega\right)^2\right) \tag{E.323}$$

$$\left(I-\Omega^H\Omega\right)^{-1} = I+x_1^2E_1^HE_1+x_1x_2E_1^HE_2+x_1x_2E_2^HE_1+x_2^2E_2^HE_2+O\left(\left(|x_1|+|x_2|\right)^4\right) \tag{E.324}$$

As $ds^2 = \text{trace}\left(\left(I-\Omega\Omega^H\right)^{-1}d\Omega\left(I-\Omega^H\Omega\right)^{-1}d\Omega^H\right)$, we now compute the product of the four terms:

$$\begin{aligned}
&\left(I-\Omega\Omega^H\right)^{-1}d\Omega\left(I-\Omega^H\Omega\right)^{-1}d\Omega^H\\
&=(dx_1E_1+dx_2E_2)\left(dx_1E_1^H+dx_2E_2^H\right)\\
&-\frac{1}{3}(dx_1E_1+dx_2E_2)\\
&[\ (3x_1^2dx_1)E_1^HE_1E_1^H+(2x_1x_2dx_1+x_1^2dx_2)E_2^HE_1E_1^H+\\
&(2x_1x_2dx_1+x_1^2dx_2)E_1^HE_2E_1^H+(x_2^2dx_1+2x_1x_2dx_2)E_2^HE_2E_1^H+\\
&(2x_1x_2dx_1+x_1^2dx_1)E_1^HE_1E_2^H+(x_2^2dx_1+2x_1x_2dx_2)E_2^HE_1E_2^H+\\
&(x_2^2dx_1+2x_1x_2dx_2)E_1^HE_2E_2^H+(3x_2^2dx_2)E_2^HE_2E_2^H\ ]\\
&+(dx_1E_1+dx_2E_2)\\
&\left(x_1^2E_1^HE_1+x_1x_2E_1^HE_2+x_1x_2E_2^HE_1+x_2^2E_2^HE_2\right)\\
&\left(dx_1E_1^H+dx_2E_2^H\right)\\
&-\frac{1}{3}[\ (3x_1^2dx_1)E_1E_1^HE_1+(2x_1x_2dx_1+x_1^2dx_2)E_1E_1^HE_2+\\
&(2x_1x_2dx_1+x_1^2dx_2)E_1E_2^HE_1+(x_2^2dx_1+2x_1x_2dx_2)E_1E_2^HE_2+\\
&(2x_1x_2dx_1+x_1^2dx_1)E_2E_1^HE_1+(x_2^2dx_1+2x_1x_2dx_2)E_2E_1^HE_2+\\
&(x_2^2dx_1+2x_1x_2dx_2)E_2E_2^HE_1+(3x_2^2dx_2)E_2E_2^HE_2\ ]\\
&\left(dx_1E_1^H+dx_2E_2^H\right)\\
&+\left(x_1^2E_1E_1^H+x_1x_2E_1E_2^H+x_1x_2E_2E_1^H+x_2^2E_2E_2^H\right)\\
&(dx_1E_1+dx_2E_2)\\
&\left(dx_1E_1^H+dx_2E_2^H\right)\\
&+O\left(\left(|x_1|+|x_2|\right)^4\right)
\end{aligned} \tag{E.325}$$

We now develop each term. We note A the term of order 2, and B_1, B_2, B_3, B_4 the terms of oder 4.

$$A := (dx_1E_1+dx_2E_2)\left(dx_1E_1^H+dx_2E_2^H\right) \tag{E.326}$$

$$A = dx_1^2E_1E_1^H+dx_1dx_2E_1E_2^H+dx_1dx_2E_2E_1^H+dx_2^2E_2E_2^H \tag{E.327}$$

$$\begin{aligned}
B_1 := &-\frac{1}{3}(dx_1E_1+dx_2E_2)\\
&[\ (3x_1^2dx_1)E_1^HE_1E_1^H+(2x_1x_2dx_1+x_1^2dx_2)E_2^HE_1E_1^H+\\
&(2x_1x_2dx_1+x_1^2dx_2)E_1^HE_2E_1^H+(x_2^2dx_1+2x_1x_2dx_2)E_2^HE_2E_1^H+\\
&(2x_1x_2dx_1+x_1^2dx_1)E_1^HE_1E_2^H+(x_2^2dx_1+2x_1x_2dx_2)E_2^HE_1E_2^H+\\
&(x_2^2dx_1+2x_1x_2dx_2)E_1^HE_2E_2^H+(3x_2^2dx_2)E_2^HE_2E_2^H\ ]
\end{aligned} \tag{E.328}$$

$$
\begin{aligned}
B_1 = &-\frac{1}{3}\left(3x_1^2dx_1^2\right)E_1E_1^HE_1E_1^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1^2+x_1^2dx_1dx_2\right)E_1E_1^HE_1E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1^2+x_1^2dx_1dx_2\right)E_1E_1^HE_2E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1^2+2x_1x_2dx_1dx_2\right)E_1E_1^HE_2E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1^2+x_1^2dx_1dx_2\right)E_1E_2^HE_1E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1^2+2x_1x_2dx_1dx_2\right)E_1E_2^HE_1E_2^H \\
&-\frac{1}{3}\left(x_2^2dx_1^2+2x_1x_2dx_1dx_2\right)E_1E_2^HE_2E_1^H \\
&-\frac{1}{3}\left(3x_2^2dx_1dx_2\right)E_1E_2^HE_2E_2^H \\
&-\frac{1}{3}\left(3x_1^2dx_1dx_2\right)E_2E_1^HE_1E_1^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1dx_2+x_1^2dx_2^2\right)E_2E_1^HE_1E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1dx_2+x_1^2dx_2^2\right)E_2E_1^HE_2E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1dx_2+2x_1x_2dx_2^2\right)E_2E_1^HE_2E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1dx_2+x_1^2dx_2^2\right)E_2E_2^HE_1E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1dx_2+2x_1x_2dx_2^2\right)E_2E_2^HE_1E_2^H \\
&-\frac{1}{3}\left(x_2^2dx_1dx_2+2x_1x_2dx_2^2\right)E_2E_2^HE_2E_1^H \\
&-\frac{1}{3}\left(3x_2^2dx_2^2\right)E_2E_2^HE_2E_2^H
\end{aligned}
\tag{E.329}
$$

$$
\begin{aligned}
B_2 := &\left(x_1^2E_1E_1^H+x_1x_2E_1E_2^H+x_1x_2E_2E_1^H+x_2^2E_2E_2^H\right) \\
&(dx_1E_1+dx_2E_2) \\
&\left(dx_1E_1^H+dx_2E_2^H\right)
\end{aligned}
$$

$$
\begin{aligned}
B_2 = &\left(x_1^2 dx_1^2\right) E_1 E_1^H E_1 E_1^H \\
&+ \left(x_1^2 dx_1 dx_2\right) E_1 E_1^H E_1 E_2^H \\
&+ \left(x_1 x_2 dx_1^2\right) E_1 E_1^H E_2 E_1^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_1 E_1^H E_2 E_2^H \\
&+ \left(x_1 x_2 dx_1^2\right) E_1 E_2^H E_1 E_1^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_1 E_2^H E_1 E_2^H \\
&+ \left(x_2^2 dx_1^2\right) E_1 E_2^H E_2 E_1^H \\
&+ \left(x_2^2 dx_1 dx_2\right) E_1 E_2^H E_2 E_2^H \\
&+ \left(x_1^2 dx_1 dx_2\right) E_2 E_1^H E_1 E_1^H \\
&+ \left(x_1^2 dx_2^2\right) E_2 E_1^H E_1 E_2^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_2 E_1^H E_2 E_1^H \\
&+ \left(x_1 x_2 dx_2^2\right) E_2 E_1^H E_2 E_2^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_2 E_2^H E_1 E_1^H \\
&+ \left(x_1 x_2 dx_2^2\right) E_2 E_2^H E_1 E_2^H \\
&+ \left(x_2^2 dx_1 dx_2\right) E_2 E_2^H E_2 E_1^H \\
&+ \left(x_2^2 dx_2^2\right) E_2 E_2^H E_2 E_2^H
\end{aligned}
\tag{E.330}
$$

$$
\begin{aligned}
B_3 := -\frac{1}{3} [\ &(3x_1^2 dx_1) E_1 E_1^H E_1 + (2x_1 x_2 dx_1 + x_1^2 dx_2) E_1 E_1^H E_2 + \\
&(2x_1 x_2 dx_1 + x_1^2 dx_2) E_1 E_2^H E_1 + (x_2^2 dx_1 + 2x_1 x_2 dx_2) E_1 E_2^H E_2 + \\
&(2x_1 x_2 dx_1 + x_1^2 dx_1) E_2 E_1^H E_1 + (x_2^2 dx_1 + 2x_1 x_2 dx_2) E_2 E_1^H E_2 + \\
&(x_2^2 dx_1 + 2x_1 x_2 dx_2) E_2 E_2^H E_1 + (3x_2^2 dx_2) E_2 E_2^H E_2\ ] \\
&\left(dx_1 E_1^H + dx_2 E_2^H\right)
\end{aligned}
\tag{E.331}
$$

$$
\begin{aligned}
B_3 = &-\frac{1}{3}\left(3x_1^2dx_1^2\right)E_1E_1^HE_1E_1^H \\
&-\frac{1}{3}\left(3x_1^2dx_1dx_2\right)E_1E_1^HE_1E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1^2+x_1^2dx_1dx_2\right)E_1E_1^HE_2E_1^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1dx_2+x_1^2dx_2^2\right)E_1E_1^HE_2E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1^2+x_1^2dx_1dx_2\right)E_1E_2^HE_1E_1^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1dx_2+x_1^2dx_2^2\right)E_1E_2^HE_1E_2^H \\
&-\frac{1}{3}\left(x_2^2dx_1^2+2x_1x_2dx_1dx_2\right)E_1E_2^HE_2E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1dx_2+2x_1x_2dx_2^2\right)E_1E_2^HE_2E_2^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1^2+x_1^2dx_1dx_2\right)E_2E_1^HE_1E_1^H \\
&-\frac{1}{3}\left(2x_1x_2dx_1dx_2+x_1^2dx_2^2\right)E_2E_1^HE_1E_2^H \\
&-\frac{1}{3}\left(x_2^2dx_1^2+2x_1x_2dx_1dx_2\right)E_2E_1^HE_2E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1dx_2+2x_1x_2dx_2^2\right)E_2E_1^HE_2E_2^H \\
&-\frac{1}{3}\left(x_2^2dx_1^2+2x_1x_2dx_1dx_2\right)E_2E_2^HE_1E_1^H \\
&-\frac{1}{3}\left(x_2^2dx_1dx_2+2x_1x_2dx_2^2\right)E_2E_2^HE_1E_2^H \\
&-\frac{1}{3}\left(3x_2^2dx_1dx_2\right)E_2E_2^HE_2E_1^H \\
&-\frac{1}{3}\left(3x_2^2dx_2^2\right)E_2E_2^HE_2E_2^H
\end{aligned}
\tag{E.332}
$$

$$
\begin{aligned}
B_4 := &\left(x_1^2E_1E_1^H+x_1x_2E_1E_2^H+x_1x_2E_2E_1^H+x_2^2E_2E_2^H\right) \\
&(dx_1E_1+dx_2E_2) \\
&\left(dx_1E_1^H+dx_2E_2^H\right)
\end{aligned}
\tag{E.333}
$$

$$
\begin{aligned}
B_4 = &\left(x_1^2 dx_1^2\right) E_1 E_1^H E_1 E_1^H \\
&+ \left(x_1^2 dx_1 dx_2\right) E_1 E_1^H E_1 E_2^H \\
&+ \left(x_1^2 dx_1 dx_2\right) E_1 E_1^H E_2 E_1^H \\
&+ \left(x_1^2 dx_2^2\right) E_1 E_1^H E_2 E_2^H \\
&+ \left(x_1 x_2 dx_1^2\right) E_1 E_2^H E_1 E_1^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_1 E_2^H E_1 E_2^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_1 E_2^H E_2 E_1^H \\
&+ \left(x_1 x_2 dx_2^2\right) E_1 E_2^H E_2 E_2^H \\
&+ \left(x_1 x_2 dx_1^2\right) E_2 E_1^H E_1 E_1^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_2 E_1^H E_1 E_2^H \\
&+ \left(x_1 x_2 dx_1 dx_2\right) E_2 E_1^H E_2 E_1^H \\
&+ \left(x_1 x_2 dx_2^2\right) E_2 E_1^H E_2 E_2^H \\
&+ \left(x_2^2 dx_1^2\right) E_2 E_2^H E_1 E_1^H \\
&+ \left(x_2^2 dx_1 dx_2\right) E_2 E_2^H E_1 E_2^H \\
&+ \left(x_2^2 dx_1 dx_2\right) E_2 E_2^H E_2 E_1^H \\
&+ \left(x_2^2 dx_2^2\right) E_2 E_2^H E_2 E_2^H
\end{aligned}
\tag{E.334}
$$

We denote $B := B_1 + B_2 + B_3 + B_4$. Adding the last equations, we obtain:

$$
\begin{aligned}
B = &(0)\, E_1 E_1^H E_1 E_1^H \\
&+ \left(-\frac{2}{3}x_1x_2dx_1^2 + \frac{2}{3}x_1^2dx_1dx_2\right) E_1 E_1^H E_1 E_2^H \\
&+ \left(-\frac{1}{3}x_1x_2dx_1^2 + \frac{1}{3}x_1^2dx_1dx_2\right) E_1 E_1^H E_2 E_1^H \\
&+ \left(-\frac{1}{3}x_2^2dx_1^2 - \frac{1}{3}x_1x_2dx_1dx_2 + \frac{2}{3}x_1^2dx_2^2\right) E_1 E_1^H E_2 E_2^H \\
&+ \left(+\frac{2}{3}x_1x_2dx_1^2 - \frac{2}{3}x_1^2dx_1dx_2\right) E_1 E_2^H E_1 E_1^H \\
&+ \left(-\frac{1}{3}x_2^2dx_1^2 + \frac{2}{3}x_1x_2dx_1dx_2 - \frac{1}{3}x_1^2dx_2^2\right) E_1 E_2^H E_1 E_2^H \\
&+ \left(+\frac{1}{3}x_2^2dx_1^2 - \frac{1}{3}x_1x_2dx_1dx_2\right) E_1 E_2^H E_2 E_1^H \\
&+ \left(-\frac{1}{3}x_2^2dx_1dx_2 + \frac{1}{3}x_1x_2dx_2^2\right) E_1 E_2^H E_2 E_2^H \\
&+ \left(+\frac{1}{3}x_1x_2dx_1^2 - \frac{1}{3}x_1^2dx_1dx_2\right) E_2 E_1^H E_1 E_1^H \\
&+ \left(-\frac{1}{3}x_1x_2dx_1dx_2 + \frac{1}{3}x_1^2dx_2^2\right) E_2 E_1^H E_1 E_2^H \\
&+ \left(-\frac{1}{3}x_2^2dx_1^2 + \frac{2}{3}x_1x_2dx_1dx_2 - \frac{1}{3}x_1^2dx_2^2\right) E_2 E_1^H E_2 E_1^H \\
&+ \left(-\frac{2}{3}x_2^2dx_1dx_2 + \frac{2}{3}x_1x_2dx_2^2\right) E_2 E_1^H E_2 E_2^H \\
&+ \left(+\frac{2}{3}x_2^2dx_1^2 - \frac{1}{3}x_1x_2dx_1dx_2 - \frac{1}{3}x_1^2dx_2^2\right) E_2 E_2^H E_1 E_1^H \\
&+ \left(+\frac{1}{3}x_2^2dx_1dx_2 - \frac{1}{3}x_1x_2dx_2^2\right) E_2 E_2^H E_1 E_2^H \\
&+ \left(+\frac{2}{3}x_2^2dx_1dx_2 - \frac{2}{3}x_1x_2dx_2^2\right) E_2 E_2^H E_2 E_1^H \\
&+ (0)\, E_2 E_2^H E_2 E_2^H
\end{aligned}
\tag{E.335}
$$

Then we compute $\mathrm{trace}\,(A)$ and $\mathrm{trace}\,(B)$ as:

$$
ds^2 = \mathrm{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H\right) \tag{E.336}
$$

$$
ds^2 = \mathrm{trace}\left(A + B + O\left(\left(|x_1| + |x_2|\right)^4\right)\right) \tag{E.337}
$$

$$
ds^2 = \mathrm{trace}\,(A) + \mathrm{trace}\,(B) + O\left(\left(|x_1| + |x_2|\right)^4\right) \tag{E.338}
$$

Hence, we compute:

$$
\begin{aligned}
\mathrm{trace}\,(A) =& \mathrm{trace}\left(dx_1^2 E_1 E_1^H + dx_1dx_2 E_1 E_2^H + dx_1dx_2 E_2 E_1^H + dx_2^2 E_2 E_2^H\right) \\
\mathrm{trace}\,(A) =& dx_1^2\, \mathrm{trace}\left(E_1 E_1^H\right) + dx_1dx_2\, \mathrm{trace}\left(E_1 E_2^H + E_2 E_1^H\right) + dx_2^2\, \mathrm{trace}\left(E_2 E_2^H\right) \\
\mathrm{trace}\,(A) =& dx_1^2 + dx_2^2
\end{aligned}
\tag{E.339}
$$

As E_1 and E_2 are the two first vectors of an orthonormal basis.

To compute $\mathrm{trace}\,(B)$, we first note the following equalities due to the invariance of the trace operator by commutation:

- $\mathrm{trace}\left(E_1 E_1^H E_1 E_2^H\right) = \mathrm{trace}\left(E_1 E_2^H E_1 E_1^H\right)$
- $\mathrm{trace}\left(E_1 E_1^H E_2 E_1^H\right) = \mathrm{trace}\left(E_2 E_1^H E_1 E_1^H\right)$
- $\mathrm{trace}\left(E_1 E_1^H E_2 E_2^H\right) = \mathrm{trace}\left(E_2 E_2^H E_1 E_1^H\right)$

- $\text{trace}\left(E_1E_2^HE_2E_1^H\right) = \text{trace}\left(E_2E_1^HE_1E_2^H\right)$
- $\text{trace}\left(E_1E_2^HE_2E_2^H\right) = \text{trace}\left(E_2E_2^HE_1E_2^H\right)$
- $\text{trace}\left(E_2E_1^HE_2E_2^H\right) = \text{trace}\left(E_2E_2^HE_2E_1^H\right)$

Using these equalities, many simplifications appear and we obtain:

$$\begin{aligned}
\text{trace}\,(B) = &+\left(-\frac{2}{3}x_1x_2dx_1^2+\frac{2}{3}x_1^2dx_1dx_2+\frac{2}{3}x_1x_2dx_1^2-\frac{2}{3}x_1^2dx_1dx_2\right)\text{trace}\left(E_1E_1^HE_1E_2^H\right)\\
&+\left(-\frac{1}{3}x_1x_2dx_1^2+\frac{1}{3}x_1^2dx_1dx_2+\frac{1}{3}x_1x_2dx_1^2-\frac{1}{3}x_1^2dx_1dx_2\right)\text{trace}\left(E_1E_1^HE_2E_1^H\right)\\
&+\left(-\frac{1}{3}x_2^2dx_1^2-\frac{1}{3}x_1x_2dx_1dx_2+\frac{2}{3}x_1^2dx_2^2+\frac{2}{3}x_2^2dx_1^2-\frac{1}{3}x_1x_2dx_1dx_2-\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_1E_1^HE_2E_2^H\right)\\
&+\left(-\frac{1}{3}x_2^2dx_1^2+\frac{2}{3}x_1x_2dx_1dx_2-\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_1E_2^HE_1E_2^H\right)\\
&+\left(+\frac{1}{3}x_2^2dx_1^2-\frac{1}{3}x_1x_2dx_1dx_2-\frac{1}{3}x_1x_2dx_1dx_2+\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_1E_2^HE_2E_1^H\right)\\
&+\left(-\frac{1}{3}x_2^2dx_1dx_2+\frac{1}{3}x_1x_2dx_2^2+\frac{1}{3}x_2^2dx_1dx_2-\frac{1}{3}x_1x_2dx_2^2\right)\text{trace}\left(E_1E_2^HE_2E_2^H\right)\\
&+\left(-\frac{1}{3}x_2^2dx_1^2+\frac{2}{3}x_1x_2dx_1dx_2-\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_2E_1^HE_2E_1^H\right)\\
&+\left(-\frac{2}{3}x_2^2dx_1dx_2+\frac{2}{3}x_1x_2dx_2^2+\frac{2}{3}x_2^2dx_1dx_2-\frac{2}{3}x_1x_2dx_2^2\right)\text{trace}\left(E_2E_1^HE_2E_2^H\right)
\end{aligned}$$

With simplifications, we obtain:

$$\begin{aligned}
\text{trace}\,(B) = &+\left(+\frac{1}{3}x_2^2dx_1^2-\frac{2}{3}x_1x_2dx_1dx_2+\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_1E_1^HE_2E_2^H\right)\\
&+\left(-\frac{1}{3}x_2^2dx_1^2+\frac{2}{3}x_1x_2dx_1dx_2-\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_1E_2^HE_1E_2^H\right)\\
&+\left(+\frac{1}{3}x_2^2dx_1^2-\frac{2}{3}x_1x_2dx_1dx_2+\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_1E_2^HE_2E_1^H\right)\\
&+\left(-\frac{1}{3}x_2^2dx_1^2+\frac{2}{3}x_1x_2dx_1dx_2-\frac{1}{3}x_1^2dx_2^2\right)\text{trace}\left(E_2E_1^HE_2E_1^H\right)
\end{aligned}$$

$$\text{trace}\,(B) = \frac{1}{3}\,\text{trace}\left(E_1E_1^HE_2E_2^H - E_1E_2^HE_1E_2^H + E_1E_2^HE_2E_1^H - E_2E_1^HE_2E_1^H\right)\left(x_1dx_2 - x_2dx_1\right)^2 \tag{E.340}$$

Note that:

$$\begin{aligned}
\left\|E_1E_2^H - E_2E_1^H\right\|^2 &= \text{trace}\left(\left(E_1E_2^H - E_2E_1^H\right)\left(E_1E_2^H - E_2E_1^H\right)^H\right)\\
&= \text{trace}\left(\left(E_1E_2^H - E_2E_1^H\right)\left(E_2E_1^H - E_1E_2^H\right)\right)\\
&= \text{trace}\left(+E_1E_2^HE_2E_1^H - E_1E_2^HE_1E_2^H - E_2E_1^HE_2E_1^H + E_2E_1^HE_1E_2^H\right)
\end{aligned}$$

As

$$\text{trace}\left(E_1E_2^HE_2E_1^H\right) = \text{trace}\left(E_2E_1^HE_1E_2^H\right), \tag{E.341}$$

we can write:

$$\left\|E_1E_2^H - E_2E_1^H\right\|^2 = \text{trace}\left(+2\,E_1E_2^HE_2E_1^H - E_1E_2^HE_1E_2^H - E_2E_1^HE_2E_1^H\right) \tag{E.342}$$

We show as well that:

$$\begin{aligned}\left\|E_1^H E_2 - E_2^H E_1\right\|^2 &= \text{trace}\left(\left(E_1^H E_2 - E_2^H E_1\right)\left(E_1^H E_2 - E_2^H E_1\right)^H\right)\\ &= \text{trace}\left(\left(E_1^H E_2 - E_2^H E_1\right)\left(E_2^H E_1 - E_1^H E_2\right)\right)\\ &= \text{trace}\left(+E_1^H E_2 E_2^H E_1 - E_1^H E_2 E_1^H E_2 - E_2^H E_1 E_2^H E_1 + E_2^H E_1 E_1^H E_2\right)\end{aligned}$$

As $\text{trace}\left(E_1^H E_2 E_2^H E_1\right) = \text{trace}\left(E_2^H E_1 E_1^H E_2\right)$, we can write:

$$\begin{aligned}\left\|E_1^H E_2 - E_2^H E_1\right\|^2 &= \text{trace}\left(+2\, E_1^H E_2 E_2^H E_1 - E_1^H E_2 E_1^H E_2 - E_2^H E_1 E_2^H E_1\right)\\ &= \text{trace}\left(+2\, E_1 E_1^H E_2 E_2^H - E_2 E_1^H E_2 E_1^H - E_1 E_2^H E_1 E_2^H\right)\end{aligned} \tag{E.343}$$

Hence we have:

$$\text{trace}\left(E_1 E_1^H E_2 E_2^H - E_1 E_2^H E_1 E_2^H + E_1 E_2^H E_2 E_1^H - E_2 E_1^H E_2 E_1^H\right) \tag{E.344}$$

$$= \frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right) \tag{E.345}$$

Therefore:

$$\text{trace}\,(B) = \frac{1}{3}\left(\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right)\right)(x_1 dx_2 - x_2 dx_1)^2 \tag{E.346}$$

We recall Equation (E.338):

$$ds^2 = \text{trace}\,(A) + \text{trace}\,(B) + O\left(\left(|x_1| + |x_2|\right)^4\right)$$

Therefore, we have from Equations (E.339) and (E.346):

$$\begin{aligned}ds^2 =& dx_1^2 + dx_2^2\\ &- \frac{1}{3}\left(-\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right)\right)(x_1 dx_2 - x_2 dx_1)^2\\ &+ O\left(\left(|x_1| + |x_2|\right)^4\right)\end{aligned} \tag{E.347}$$

We recall Equation (D.1) which gives a link between the metric element ds^2 and the sectional curvature [31]:

$$\begin{aligned}ds^2 =& dx_1^2 + dx_2^2 - \frac{1}{3}K(\sigma)\,(x_1 dx_2 - x_2 dx_1)^2\\ &+ O\left(\left(|x_1| + |x_2|\right)^3\right)dx_1^2 + O\left(\left(|x_1| + |x_2|\right)^3\right)dx_1 dx_2 + O\left(\left(|x_1| + |x_2|\right)^3\right)dx_2^2\end{aligned}$$

By identification, we finally showed that:

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right) \tag{E.348}$$

□

Corollary 2.

$$-4 \leqslant K(\sigma) \leqslant 0 \tag{E.349}$$

Proof. If the base point considered is not zero, we first transport the problem at zero using the differential of the isometry E.17.

Let E_1 an E_2 be an orthonormal basis of the plan σ at zero such that:

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|_2^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|_2^2\right) \tag{E.350}$$

Using the positivity of the norm, we have: $K(\sigma) \leqslant 0$.
Due the triangular inequality, we have:

$$K(\sigma) \geqslant -\frac{1}{2}\left(\left(\left\|E_1 E_2^H\right\|_2 + \left\|E_2 E_1^H\right\|_2\right)^2 + \left(\left\|E_1^H E_2\right\|_2 + \left\|E_2^H E_1\right\|_2\right)^2\right) \tag{E.351}$$

Using the inequality on the Frobenuis norm: $\|AB\|_2 \leqslant \|A\|_2 \|B\|_2$ and the fact that $\|E_1\|_2 = 1$ and $\|E_2\|_2 = 1$, we obtain:

$$K(\sigma) \geqslant -\frac{1}{2}\left((1+1)^2 + (1+1)^2\right) = -4 \tag{E.352}$$

□

We recall that the Siegel space is a generalization of the Poincaré disk $\mathbb{D}$ defined as the complex unit disk endowed with the metric:

$$ds^2 = \frac{|d\mu_k|^2}{\left(1 - |\mu_k|^2\right)^2} \tag{E.353}$$

If we perform Theorem 10 on the orthonormal basis vectors $E_1 = 1$ and $E_2 = i$, we find that its curvature equals to -4.

E.7.2 In polar coodinates

Let σ be a section defined at the point $\Omega = 0$ by the two first vectors of an orthonormal basis $(E_1, \ldots, E_n)$ as described in Section D.1.

We recall the expression of the Siegel space scalar product given in Equation (E.6):

$$\langle v, w\rangle_\Omega = \frac{1}{2}\operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} v \left(I - \Omega^H\Omega\right)^{-1} w^H\right) \tag{E.354}$$

$$+\frac{1}{2}\operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} w \left(I - \Omega^H\Omega\right)^{-1} v^H\right) \tag{E.355}$$

At the point $\Omega = 0$, we obtain:

$$\langle v, w\rangle_0 = \frac{1}{2}\operatorname{trace}\left(vw^H + wv^H\right) \tag{E.356}$$

Hence the Siegel space scalar product at the point $\Omega = 0$ corresponds to the classical Euclidean scalar product. As a consequence, the Siegel space norm at the point $\Omega = 0$ also correponds to the classical Euclidean norm.

We now use the polar coordinates in the plan σ defined by E_1 and E_2. We define:

$$E_r := \cos(\theta) E_1 + \sin(\theta) E_2 \tag{E.357}$$

$$E_\theta := -\sin(\theta) E_1 + \cos(\theta) E_2 \tag{E.358}$$

Note that we have $\|E_r\|_0 = \|E_\theta\|_0 = 1$ and $\langle E_r, E_\theta\rangle_0 = 0$.

Theorem 11. *The sectional curvature at zero of the plan σ defined by E_r and E_θ has the following expression:*

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right) \tag{E.359}$$

Proof. We first recall the expression of the Siegel metric given in Section E.2:

$$ds^2 = \operatorname{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H\right) \tag{E.360}$$

We set $\Omega = \exp_0(rE_r)$.

We recall the expression of the exponential map at zero described in Section E.5.1:

$$\exp_0(V) = \tanh(Y)\, Y^{-1} V \quad \text{where } Y = \left(VV^H\right)^{1/2} \tag{E.361}$$

Here we have $V = rE_r$ and $Y = \left(VV^H\right)^{1/2} = \left((rE_r)(rE_r)^H\right)^{1/2} = r\left(E_r E_r^H\right)^{1/2}$.

As the curvature is a local property, we can use a limited of the fonction $\tanh$.

We recall that $\forall z \in \mathbb{C}$ such that $|z| < 2\pi$, we have:

$$tanh(z) = z - \frac{1}{3}z^3 + \frac{2}{15}z^5 + O\left(z^7\right) \tag{E.362}$$

Hence, as Y is a matrix close to zero:

$$\Omega = \tanh(Y)\, Y^{-1} V \tag{E.363}$$

$$\Omega = \left(I - \frac{1}{3} Y^2 + \frac{2}{15} Y^4 + O\left(Y^6\right) \right) (rE_r) \tag{E.364}$$

$$\begin{aligned} \Omega = & (rE_r) \\ & - \frac{1}{3} (rE_r) (rE_r)^H (rE_r) \\ & + \frac{2}{15} (rE_r) (rE_r)^H (rE_r) (rE_r)^H (rE_r) \\ & + O\left(r^7\right) \end{aligned} \tag{E.365}$$

$$\begin{aligned} \Omega = & rE_r \\ & - \frac{1}{3} r^3 E_r E_r^H E_r \\ & + \frac{2}{15} r^5 E_r E_r^H E_r E_r^H E_r \\ & + O\left(r^7\right) \end{aligned} \tag{E.366}$$

Hence, we obtain using the conjugate transpose operator:

$$\begin{aligned} \Omega^H = & rE_r^H \\ & - \frac{1}{3} r^3 E_r^H E_r E_r^H \\ & + \frac{2}{15} r^5 E_r^H E_r E_r^H E_r E_r^H \\ & + O\left(r^7\right) \end{aligned} \tag{E.367}$$

We now compute the infinitesimal variations.
We first recall that:

$$dE_r = d(\cos(\theta) E_1 + \sin(\theta) E_2) \tag{E.368}$$

$$dE_r = d(\cos(\theta)) E_1 + d(\sin(\theta)) E_2 \tag{E.369}$$

$$dE_r = -\sin(\theta)\, d_\theta E_1 + \cos(\theta)\, d_\theta E_2 \tag{E.370}$$

$$dE_r = d_\theta E_\theta \tag{E.371}$$

$$dE_\theta = d(-\sin(\theta) E_1 + \cos(\theta) E_2) \tag{E.372}$$

$$dE_\theta = d(-\sin(\theta)) E_1 + d(\cos(\theta)) E_2 \tag{E.373}$$

$$dE_\theta = -\cos(\theta)\, d_\theta E_1 + -\sin(\theta)\, d_\theta E_2 \tag{E.374}$$

$$dE_\theta = -d_\theta E_r \tag{E.375}$$

Hence:

$$\begin{aligned} d\Omega = & (d_r E_r + r d_\theta E_\theta) \\ & - \frac{1}{3} \left(3r^2 d_r E_r E_r^H E_r + r^3 d_\theta E_\theta E_r^H E_r + r^3 d_\theta E_r E_\theta^H E_r + r^3 d_\theta E_r E_r^H E_\theta \right) \\ & + \frac{2}{15} (\, 5r^4 d_r E_r E_r^H E_r E_r^H E_r \\ & \qquad + r^5 d_\theta E_\theta E_r^H E_r E_r^H E_r \\ & \qquad + r^5 d_\theta E_r E_\theta^H E_r E_r^H E_r \\ & \qquad + r^5 d_\theta E_r E_r^H E_\theta E_r^H E_r \\ & \qquad + r^5 d_\theta E_r E_r^H E_r E_\theta^H E_r \\ & \qquad + r^5 d_\theta E_r E_r^H E_r E_r^H E_\theta \,) \\ & + O\left(r^6\right) \end{aligned} \tag{E.376}$$

Hence, we have at the order 5:

$$\begin{aligned} d\Omega = &(d_r E_r + r d_\theta E_\theta) \\ &- \frac{1}{3}\left(3r^2 d_r E_r E_r^H E_r + r^3 d_\theta E_\theta E_r^H E_r + r^3 d_\theta E_r E_\theta^H E_r + r^3 d_\theta E_r E_r^H E_\theta\right) \\ &+ \frac{2}{3} r^4 d_r E_r E_r^H E_r E_r^H E_r \\ &+ O\left(r^5\right) \end{aligned} \tag{E.377}$$

Therefore, using the conjugate transpose operator:

$$\begin{aligned} d\Omega^H = &\left(d_r E_r^H + r d_\theta E_\theta^H\right) \\ &- \frac{1}{3}\left(3r^2 d_r E_r^H E_r E_r^H + r^3 d_\theta E_r^H E_r E_\theta^H + r^3 d_\theta E_r^H E_\theta E_r^H + r^3 d_\theta E_\theta^H E_r E_r^H\right) \\ &+ \frac{2}{3} r^4 d_r E_r^H E_r E_r^H E_r E_r^H \\ &+ O\left(r^5\right) \end{aligned} \tag{E.378}$$

We now compute the products:

$$\Omega\Omega^H = \left[r E_r - \frac{1}{3} r^3 E_r E_r^H E_r + \frac{2}{15} r^5 E_r E_r^H E_r E_r^H E_r + O\left(r^7\right)\right] \tag{E.379}$$

$$\left[r E_r^H - \frac{1}{3} r^3 E_r^H E_r E_r^H + \frac{2}{15} r^5 E_r^H E_r E_r^H E_r E_r^H + O\left(r^7\right)\right] \tag{E.380}$$

$$\Omega\Omega^H = r^2 E_r E_r^H - \frac{2}{3} r^4 E_r E_r^H E_r E_r^H + O\left(r^6\right) \tag{E.381}$$

We obtain as well with $\Omega \leftarrow \Omega^H$:

$$\Omega^H\Omega = r^2 E_r^H E_r - \frac{2}{3} r^4 E_r^H E_r E_r^H E_r + O\left(r^6\right) \tag{E.382}$$

We recall that $\forall X \in M_n(\mathbb{C})$ such that $|||X||| < 1$, we have:

$$(I - X)^{-1} = \sum_{n=0}^{+\infty} X^n \tag{E.383}$$

Hence:

$$(I - X)^{-1} = I + X + X^2 + O\left(X^3\right) \tag{E.384}$$

Therefore, we can compute:

$$\left(I - \Omega\Omega^H\right)^{-1} = I + \Omega\Omega^H + \left(\Omega\Omega^H\right)^2 + O\left(\left(\Omega\Omega^H\right)^3\right) \tag{E.385}$$

$$\left(I - \Omega\Omega^H\right)^{-1} = I + r^2 E_r E_r^H - \frac{2}{3} r^4 E_r E_r^H E_r E_r^H + O\left(r^6\right) + r^4 E_r E_r^H E_r E_r^H + O\left(r^6\right) \tag{E.386}$$

$$\left(I - \Omega\Omega^H\right)^{-1} = I + r^2 E_r E_r^H + \frac{1}{3} r^4 E_r E_r^H E_r E_r^H + O\left(r^6\right) \tag{E.387}$$

and:

$$\left(I - \Omega^H\Omega\right)^{-1} = I + \Omega^H\Omega + \left(\Omega^H\Omega\right)^2 + O\left(\left(\Omega^H\Omega\right)^3\right) \tag{E.388}$$

$$\left(I - \Omega^H\Omega\right)^{-1} = I + r^2 E_r^H E_r - \frac{2}{3} r^4 E_r^H E_r E_r^H E_r + O\left(r^6\right) + r^4 E_r^H E_r E_r^H E_r + O\left(r^6\right) \tag{E.389}$$

$$\left(I - \Omega^H\Omega\right)^{-1} = I + r^2 E_r^H E_r + \frac{1}{3} r^4 E_r^H E_r E_r^H E_r + O\left(r^6\right) \tag{E.390}$$

As $ds^2 = \text{trace}\left(\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H\right)$, we now compute the product of the four terms:

$$
\begin{aligned}
&\left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H \\
=&\left(I + r^2 E_r E_r^H + \frac{1}{3} r^4 E_r E_r^H E_r E_r^H + O\left(r^6\right)\right) \qquad \text{(E.391)} \\
&(\ (d_r E_r + r d_\theta E_\theta) \\
&- \frac{1}{3}\left(3r^2 d_r E_r E_r^H E_r + r^3 d_\theta E_r E_r^H E_\theta + r^3 d_\theta E_r E_\theta^H E_r + r^3 d_\theta E_\theta E_r^H E_r\right) \\
&+ \frac{2}{3} r^4 d_r E_r E_r^H E_r E_r^H E_r \\
&+ O\left(r^5\right)\ ) \\
&\left(I + r^2 E_r^H E_r + \frac{1}{3} r^4 E_r^H E_r E_r^H E_r + O\left(r^6\right)\right) \qquad \text{(E.392)} \\
&(\ \left(d_r E_r^H + r d_\theta E_\theta^H\right) \\
&- \frac{1}{3}\left(3r^2 d_r E_r^H E_r E_r^H + r^3 d_\theta E_r^H E_r E_\theta^H + r^3 d_\theta E_r^H E_\theta E_r^H + r^3 d_\theta E_\theta^H E_r E_r^H\right) \\
&+ \frac{2}{3} r^4 d_r E_r^H E_r E_r^H E_r E_r^H \\
&+ O\left(r^5\right)\ )
\end{aligned}
$$

We now develop this product.
We denote S the product:

$$
\begin{aligned}
S :=& \left(I - \Omega\Omega^H\right)^{-1} d\Omega \left(I - \Omega^H\Omega\right)^{-1} d\Omega^H \\
S =& S_0 + S_1 r + S_2 r^2 + S_3 r^3 + S_4 r^4 + O\left(r^5\right) \qquad \text{(E.393)}
\end{aligned}
$$

where S_i is the coefficient of order i in r.
We denote A the first term of the product:

$$
\begin{aligned}
A :=& \left(I - \Omega\Omega^H\right)^{-1} \qquad \text{(E.394)} \\
A =& \left(I + r^2 E_r E_r^H + \frac{1}{3} r^4 E_r E_r^H E_r E_r^H + O\left(r^6\right)\right) \qquad \text{(E.395)} \\
A =& A_0 + A_2 r^2 + A_4 r^4 + O\left(r^6\right) \qquad \text{(E.396)}
\end{aligned}
$$

where:

$$
\begin{aligned}
A_0 :=& I \qquad \text{(E.397)} \\
A_2 :=& E_r E_r^H \qquad \text{(E.398)} \\
A_4 :=& \frac{1}{3} E_r E_r^H E_r E_r^H \qquad \text{(E.399)}
\end{aligned}
$$

We denote B the second term of the product:

$$
\begin{aligned}
B :=& d\Omega \qquad \text{(E.400)} \\
B =& (\ (d_r E_r + r d_\theta E_\theta) \\
&- \frac{1}{3}\left(3r^2 d_r E_r E_r^H E_r + r^3 d_\theta E_r E_r^H E_\theta + r^3 d_\theta E_r E_\theta^H E_r + r^3 d_\theta E_\theta E_r^H E_r\right) \\
&+ \frac{2}{3} r^4 d_r E_r E_r^H E_r E_r^H E_r \\
&+ O\left(r^5\right)\ ) \qquad \text{(E.401)} \\
B =& B_0 + B_1 r + B_2 r^2 + B_3 r^3 + B_4 r^4 + O\left(r^5\right) \qquad \text{(E.402)}
\end{aligned}
$$

where:

$$B_0 := d_r E_r \tag{E.403}$$
$$B_1 := d_\theta E_\theta \tag{E.404}$$
$$B_2 := -d_r E_r E_r^H E_r \tag{E.405}$$
$$B_3 := -\frac{1}{3} d_\theta \left(E_r E_r^H E_\theta + E_r E_\theta^H E_r + E_\theta E_r^H E_r \right) \tag{E.406}$$
$$B_4 := \frac{2}{3} d_r E_r E_r^H E_r E_r^H E_r \tag{E.407}$$

We denote C the tird term of the product:

$$C := \left(I - \Omega^H \Omega \right)^{-1} \tag{E.408}$$
$$C = \left(I + r^2 E_r^H E_r + \frac{1}{3} r^4 E_r^H E_r E_r^H E_r + O\left(r^6\right) \right) \tag{E.409}$$
$$C = C_0 + C_2 r^2 + C_4 r^4 + O\left(r^6\right) \tag{E.410}$$

where:

$$C_0 := I \tag{E.411}$$
$$C_2 := E_r^H E_r \tag{E.412}$$
$$C_4 := \frac{1}{3} E_r^H E_r E_r^H E_r \tag{E.413}$$

Finally, we denote D the fourth and last term of the product:

$$D := d\Omega \tag{E.414}$$
$$\begin{aligned} D = (& \left(d_r E_r^H + r d_\theta E_\theta^H \right) \\ & - \frac{1}{3} \left(3 r^2 d_r E_r^H E_r E_r^H + r^3 d_\theta E_r^H E_r E_\theta^H + r^3 d_\theta E_r^H E_\theta E_r^H + r^3 d_\theta E_\theta^H E_r E_r^H \right) \\ & + \frac{2}{3} r^4 d_r E_r^H E_r E_r^H E_r E_r^H \\ & + O\left(r^5\right)) \end{aligned} \tag{E.415}$$
$$D = D_0 + D_1 r + D_2 r^2 + D_3 r^3 + D_4 r^4 + O\left(r^5\right) \tag{E.416}$$

where:

$$D_0 := d_r E_r^H \tag{E.417}$$
$$D_1 := d_\theta E_\theta^H \tag{E.418}$$
$$D_2 := -d_r E_r^H E_r E_r^H \tag{E.419}$$
$$D_3 := -\frac{1}{3} d_\theta \left(E_r^H E_r E_\theta^H + E_r^H E_\theta E_r^H + E_\theta^H E_r E_r^H \right) \tag{E.420}$$
$$D_4 := \frac{2}{3} d_r E_r^H E_r E_r^H E_r E_r^H \tag{E.421}$$

We now devopp the product and compute the terms order by order.
The only term of order zero is:

$$S_0 = A_0 B_0 C_0 D_0 \tag{E.422}$$
$$S_0 = (I)\ (d_r E_r)\ (I)\ \left(d_r E_r^H \right) \tag{E.423}$$
$$S_0 = d_r^2 E_r E_r^H \tag{E.424}$$

We now compute the coefficient of order 1:

$$S_1 = A_0 B_0 C_0 D_1 + A_0 B_1 C_0 D_0 \tag{E.425}$$
$$S_1 = (I)\ (d_r E_r)\ (I)\ \left(d_\theta E_\theta^H\right) + (I)\ (d_\theta E_\theta)\ (I)\ \left(d_r E_r^H\right) \tag{E.426}$$
$$S_1 = d_r d_\theta \left(E_r E_\theta^H + E_\theta E_r^H\right) \tag{E.427}$$

We now compute the coefficient of order 2:

$$\begin{aligned} S_2 = \quad & A_0 B_0 C_0 D_2 \\ & + A_0 B_0 C_2 D_0 \\ & + A_0 B_2 C_0 D_0 \\ & + A_2 B_0 C_0 D_0 \\ & + A_0 B_1 C_0 D_1 \end{aligned} \tag{E.428}$$

$$\begin{aligned} S_2 = \quad & (I)\,(d_r E_r)\,(I)\left(-d_r E_r^H E_r E_r^H\right) \\ & + (I)\,(d_r E_r)\left(E_r^H E_r\right)\left(d_r E_r^H\right) \\ & + (I)\left(-d_r E_r E_r^H E_r\right)(I)\left(d_r E_r^H\right) \\ & + \left(E_r E_r^H\right)(d_r E_r)\,(I)\left(d_r E_r^H\right) \\ & + (I)\,(d_\theta E_\theta)\,(I)\left(d_\theta E_\theta^H\right) \end{aligned} \tag{E.429}$$

$$\begin{aligned} S_2 = & - d_r^2 E_r E_r^H E_r E_r^H \\ & + d_r^2 E_r E_r^H E_r E_r^H \\ & - d_r^2 E_r E_r^H E_r E_r^H \\ & + d_r^2 E_r E_r^H E_r E_r^H \\ & + d_\theta^2 E_\theta E_\theta^H \end{aligned} \tag{E.430}$$
$$S_2 = + d_\theta^2 E_\theta E_\theta^H \tag{E.431}$$

Then we compute the coefficient of order 3:

$$\begin{aligned} S_3 = \quad & A_0 B_0 C_0 D_3 \\ & + A_0 B_3 C_0 D_0 \\ & + A_0 B_1 C_0 D_2 \\ & + A_0 B_0 C_2 D_1 \\ & + A_0 B_1 C_2 D_0 \\ & + A_0 B_2 C_0 D_1 \\ & + A_2 B_0 C_0 D_1 \\ & + A_2 B_1 C_0 D_0 \end{aligned} \tag{E.432}$$

$$\begin{aligned} S_3 = \quad & (I)\,(d_r E_r)\,(I)\left(-\frac{1}{3} d_\theta \left(E_r^H E_r E_\theta^H + E_r^H E_\theta E_r^H + E_\theta^H E_r E_r^H\right)\right) \\ & + (I)\left(-\frac{1}{3} d_\theta \left(E_r E_r^H E_\theta + E_r E_\theta^H E_r + E_\theta E_r^H E_r\right)\right)(I)\left(d_r E_r^H\right) \\ & + (I)\,(d_\theta E_\theta)\,(I)\left(-d_r E_r^H E_r E_r^H\right) \\ & + (I)\,(d_r E_r)\left(E_r^H E_r\right)\left(d_\theta E_\theta^H\right) \\ & + (I)\,(d_\theta E_\theta)\left(E_r^H E_r\right)\left(d_r E_r^H\right) \\ & + (I)\left(-d_r E_r E_r^H E_r\right)(I)\left(d_\theta E_\theta^H\right) \\ & + \left(E_r E_r^H\right)(d_r E_r)\,(I)\left(d_\theta E_\theta^H\right) \\ & + \left(E_r E_r^H\right)(d_\theta E_\theta)\,(I)\left(d_r E_r^H\right) \end{aligned} \tag{E.433}$$

$$
\begin{aligned}
S_3 = &-\frac{1}{3} d_r d_\theta \left(E_r E_r^H E_r E_\theta^H + E_r E_r^H E_\theta E_r^H + E_r E_\theta^H E_r E_r^H \right) \\
&-\frac{1}{3} d_r d_\theta \left(E_r E_r^H E_\theta E_r^H + E_r E_\theta^H E_r E_r^H + E_\theta E_r^H E_r E_r^H \right) \\
&- d_r d_\theta E_\theta E_r^H E_r E_r^H \\
&+ d_r d_\theta E_r E_r^H E_r E_\theta^H \\
&+ d_r d_\theta E_\theta E_r^H E_r E_r^H \\
&- d_r d_\theta E_r E_r^H E_r E_\theta^H \\
&+ d_r d_\theta E_r E_r^H E_r E_\theta^H \\
&+ d_r d_\theta E_r E_r^H E_\theta E_r^H
\end{aligned}
\tag{E.434}
$$

$$
\begin{aligned}
S_3 = d_r d_\theta \, (& \frac{2}{3} E_r E_r^H E_r E_\theta^H \\
&+ \frac{1}{3} E_r E_r^H E_\theta E_r^H \\
&- \frac{2}{3} E_r E_\theta^H E_r E_r^H \\
&- \frac{1}{3} E_\theta E_r^H E_r E_r^H \,)
\end{aligned}
\tag{E.435}
$$

Finally we compute the coefficient of order 4:

$$
\begin{aligned}
S_4 = \quad & A_0 B_0 C_0 D_4 \\
&+ A_0 B_0 C_4 D_0 \\
&+ A_0 B_4 C_0 D_0 \\
&+ A_4 B_0 C_0 D_0 \\
&+ A_0 B_1 C_0 D_3 \\
&+ A_0 B_3 C_0 D_1 \\
&+ A_0 B_0 C_2 D_2 \\
&+ A_0 B_2 C_0 D_2 \\
&+ A_2 B_0 C_0 D_2 \\
&+ A_0 B_2 C_2 D_0 \\
&+ A_2 B_0 C_2 D_0 \\
&+ A_2 B_2 C_0 D_0 \\
&+ A_0 B_1 C_2 D_1 \\
&+ A_2 B_1 C_0 D_1
\end{aligned}
\tag{E.436}
$$

$$
\begin{aligned}
S_4 = \quad & (I)\,(d_r E_r)\,(I)\left(\frac{2}{3} d_r E_r^H E_r E_r^H E_r E_r^H\right) \\
& + (I)\,(d_r E_r)\left(\frac{1}{3} E_r^H E_r E_r^H E_r\right)\left(d_r E_r^H\right) \\
& + (I)\left(\frac{2}{3} d_r E_r E_r^H E_r E_r^H E_r\right)(I)\left(d_r E_r^H\right) \\
& + \left(\frac{1}{3} E_r E_r^H E_r E_r^H\right)(d_r E_r)\,(I)\left(d_r E_r^H\right) \\
& + (I)\,(d_\theta E_\theta)\,(I)\left(-\frac{1}{3} d_\theta \left(E_r^H E_r E_\theta^H + E_r^H E_\theta E_r^H + E_\theta^H E_r E_r^H\right)\right) \\
& + (I)\left(-\frac{1}{3} d_\theta \left(E_r E_r^H E_\theta + E_r E_\theta^H E_r + E_\theta E_r^H E_r\right)\right)(I)\left(d_\theta E_\theta^H\right) \\
& + (I)\,(d_r E_r)\left(E_r^H E_r\right)\left(-d_r E_r^H E_r E_r^H\right) \\
& + (I)\left(-d_r E_r E_r^H E_r\right)(I)\left(-d_r E_r^H E_r E_r^H\right) \\
& + \left(E_r E_r^H\right)(d_r E_r)\,(I)\left(-d_r E_r^H E_r E_r^H\right) \\
& + (I)\left(-d_r E_r E_r^H E_r\right)\left(E_r^H E_r\right)\left(d_r E_r^H\right) \\
& + \left(E_r E_r^H\right)(d_r E_r)\left(E_r^H E_r\right)\left(d_r E_r^H\right) \\
& + \left(E_r E_r^H\right)\left(-d_r E_r E_r^H E_r\right)(I)\left(d_r E_r^H\right) \\
& + (I)\,(d_\theta E_\theta)\left(E_r^H E_r\right)\left(d_\theta E_\theta^H\right) \\
& + \left(E_r E_r^H\right)(d_\theta E_\theta)\,(I)\left(d_\theta E_\theta^H\right)
\end{aligned}
\tag{E.437}
$$

$$
\begin{aligned}
S_4 = \quad & \frac{2}{3} d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& + \frac{1}{3} d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& + \frac{2}{3} d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& + \frac{1}{3} d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& - \frac{1}{3} d_\theta^2 \left(E_\theta E_r^H E_r E_\theta^H + E_\theta E_r^H E_\theta E_r^H + E_\theta E_\theta^H E_r E_r^H\right) \\
& - \frac{1}{3} d_\theta^2 \left(E_r E_r^H E_\theta E_\theta^H + E_r E_\theta^H E_r E_\theta^H + E_\theta E_r^H E_r E_\theta^H\right) \\
& - d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& + d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& - d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& - d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& + d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& - d_r^2 E_r E_r^H E_r E_r^H E_r E_r^H \\
& + d_\theta^2 E_\theta E_r^H E_r E_\theta^H \\
& + d_\theta^2 E_r E_r^H E_\theta E_\theta^H
\end{aligned}
\tag{E.438}
$$

$$
\begin{aligned}
S_4 =& d_\theta^2 \, (\, + \frac{2}{3} E_r E_r^H E_\theta E_\theta^H \\
& - \frac{1}{3} E_r E_\theta^H E_r E_\theta^H \\
& + \frac{1}{3} E_\theta E_r^H E_r E_\theta^H \\
& - \frac{1}{3} E_\theta E_r^H E_\theta E_r^H \\
& - \frac{1}{3} E_\theta E_\theta^H E_r E_r^H \,)
\end{aligned} \tag{E.439}
$$

As:

$$
ds^2 = \operatorname{trace}\left(\left(I - \Omega\Omega^H \right)^{-1} d\Omega \left(I - \Omega^H\Omega \right)^{-1} d\Omega^H \right) \tag{E.440}
$$

We have:

$$
ds^2 = \operatorname{trace}\left(S \right) \tag{E.441}
$$

$$
ds^2 = \operatorname{trace}\left(S_0 + S_1 r + S_2 r^2 + S_3 r^3 + S_4 r^4 + O\left(r^5\right) \right) \tag{E.442}
$$

$$
ds^2 = \operatorname{trace}\left(S_0 \right) + \operatorname{trace}\left(S_1 \right) r + \operatorname{trace}\left(S_2 \right) r^2 + \operatorname{trace}\left(S_3 \right) r^3 + \operatorname{trace}\left(S_4 \right) r^4 + O\left(r^5\right) \tag{E.443}
$$

Hence we compute the trace of the coefficients of order 0 to 4.
We compute the trace of the element of order 0 using Equation (E.424):

$$
\operatorname{trace}\left(S_0 \right) = \operatorname{trace}\left(d_r^2 E_r E_r^H \right) \tag{E.444}
$$

$$
\operatorname{trace}\left(S_0 \right) = d_r^2 \operatorname{trace}\left(E_r E_r^H \right) \tag{E.445}
$$

$$
\operatorname{trace}\left(S_0 \right) = d_r^2 \left\| E_r \right\|_0^2 \tag{E.446}
$$

$$
\operatorname{trace}\left(S_0 \right) = d_r^2 \tag{E.447}
$$

We compute the trace of the element of order 1 using Equation (E.427):

$$
\operatorname{trace}\left(S_1 \right) = \operatorname{trace}\left(d_r d_\theta \left(E_r E_\theta^H + E_\theta E_r^H \right) \right) \tag{E.448}
$$

$$
\operatorname{trace}\left(S_1 \right) = d_r d_\theta \operatorname{trace}\left(E_r E_\theta^H + E_\theta E_r^H \right) \tag{E.449}
$$

$$
\operatorname{trace}\left(S_1 \right) = d_r d_\theta \left\langle E_r, E_\theta \right\rangle_0 \tag{E.450}
$$

$$
\operatorname{trace}\left(S_1 \right) = 0 \tag{E.451}
$$

We compute the trace of the element of order 2 using Equation (E.431):

$$
\operatorname{trace}\left(S_2 \right) = \operatorname{trace}\left(d_\theta^2 E_\theta E_\theta^H \right) \tag{E.452}
$$

$$
\operatorname{trace}\left(S_2 \right) = d_\theta^2 \operatorname{trace}\left(E_\theta E_\theta^H \right) \tag{E.453}
$$

$$
\operatorname{trace}\left(S_2 \right) = d_\theta^2 \left\| E_\theta \right\|_0^2 \tag{E.454}
$$

$$
\operatorname{trace}\left(S_2 \right) = d_\theta^2 \tag{E.455}
$$

We compute the trace of the element of order 3 using Equation (E.435):

$$\begin{aligned}\operatorname{trace}(S_3) = \operatorname{trace}(\ d_r d_\theta\ (\ &\frac{2}{3} E_r E_r^H E_r E_\theta^H \\ &+ \frac{1}{3} E_r E_r^H E_\theta E_r^H \\ &- \frac{2}{3} E_r E_\theta^H E_r E_r^H \\ &- \frac{1}{3} E_\theta E_r^H E_r E_r^H\ )\ ) \end{aligned} \tag{E.456}$$

$$\begin{aligned}\operatorname{trace}(S_3) =& d_r d_\theta\ (\ \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &+ \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right) \\ &- \frac{2}{3} \operatorname{trace}\left(E_r E_\theta^H E_r E_r^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_\theta E_r^H E_r E_r^H\right)\ ) \end{aligned} \tag{E.457}$$

$$\begin{aligned}\operatorname{trace}(S_3) =& d_r d_\theta\ (\ \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &+ \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right) \\ &- \frac{2}{3} \operatorname{trace}\left(E_r E_\theta^H E_r E_r^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_\theta E_r^H E_r E_r^H\right)\ ) \end{aligned} \tag{E.458}$$

$$\begin{aligned}&= d_r d_\theta\ (\ \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &+ \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right) \\ &- \frac{2}{3} \operatorname{trace}\left(E_r E_\theta^H E_r E_r^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_\theta E_r^H E_r E_r^H\right)\ ) \end{aligned} \tag{E.459}$$

Due to the invariance of the trace operator by commutativity, note that:

- $\operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) = \operatorname{trace}\left(E_r E_\theta^H E_r E_r^H\right)$
- $\operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right) = \operatorname{trace}\left(E_\theta E_r^H E_r E_r^H\right)$

Then we have:

$$\begin{aligned}\operatorname{trace}(S_3) =& d_r d_\theta\ (\ \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &+ \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right) \\ &- \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right)\ ) \end{aligned} \tag{E.460}$$

$$\begin{aligned}&= d_r d_\theta\ (\ \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &+ \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right) \\ &- \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_r E_\theta^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_r^H\right)\ ) \end{aligned} \tag{E.461}$$

$$\operatorname{trace}(S_3) = 0 \tag{E.462}$$

Finally, we compute the trace of the element of order 4 using Equation (E.439):

$$\begin{aligned}\operatorname{trace}(S_4) = \operatorname{trace}(\ d_\theta^2\ (\ &+ \frac{2}{3} E_r E_r^H E_\theta E_\theta^H \\ &- \frac{1}{3} E_r E_\theta^H E_r E_\theta^H \\ &+ \frac{1}{3} E_\theta E_r^H E_r E_\theta^H \\ &- \frac{1}{3} E_\theta E_r^H E_\theta E_r^H \\ &- \frac{1}{3} E_\theta E_\theta^H E_r E_r^H\ )\ ) \end{aligned} \tag{E.463}$$

$$\begin{aligned}\operatorname{trace}(S_4) =& d_\theta^2\ (\ + \frac{2}{3} \operatorname{trace}\left(E_r E_r^H E_\theta E_\theta^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_r E_\theta^H E_r E_\theta^H\right) \\ &+ \frac{1}{3} \operatorname{trace}\left(E_\theta E_r^H E_r E_\theta^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_\theta E_r^H E_\theta E_r^H\right) \\ &- \frac{1}{3} \operatorname{trace}\left(E_\theta E_\theta^H E_r E_r^H\right)\ ) \end{aligned} \tag{E.464}$$

Due to the invariance of the trace operator by commutativity, note that:

$$\operatorname{trace}\left(E_r E_r^H E_\theta E_\theta^H\right) = \operatorname{trace}\left(E_\theta E_\theta^H E_r E_r^H\right) \tag{E.465}$$

Hence:

$$\begin{aligned}\operatorname{trace}(S_4) =& \frac{1}{3} d_\theta^2 \big(+ \operatorname{trace}\left(E_r E_r^H E_\theta E_\theta^H\right) \\ & - \operatorname{trace}\left(E_r E_\theta^H E_r E_\theta^H\right) \\ & + \operatorname{trace}\left(E_\theta E_r^H E_r E_\theta^H\right) \\ & - \operatorname{trace}\left(E_\theta E_r^H E_\theta E_r^H\right)\end{aligned} \tag{E.466}$$

Note that:

$$\begin{aligned}\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 &= \operatorname{trace}\left(\left(E_r E_\theta^H - E_\theta E_r^H\right)\left(E_r E_\theta^H - E_\theta E_r^H\right)^H\right) \\ &= \operatorname{trace}\left(\left(E_r E_\theta^H - E_\theta E_r^H\right)\left(E_\theta E_r^H - E_r E_\theta^H\right)\right) \\ &= \operatorname{trace}\left(+E_r E_\theta^H E_\theta E_r^H - E_r E_\theta^H E_r E_\theta^H - E_\theta E_r^H E_\theta E_r^H + E_\theta E_r^H E_r E_\theta^H\right)\end{aligned}$$

As $\operatorname{trace}\left(E_r E_\theta^H E_\theta E_r^H\right) = \operatorname{trace}\left(E_\theta E_r^H E_r E_\theta^H\right)$, we can write:

$$\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 = \operatorname{trace}\left(+2\, E_\theta E_r^H E_r E_\theta^H - E_r E_\theta^H E_r E_\theta^H - E_\theta E_r^H E_\theta E_r^H\right) \tag{E.467}$$

We show as well that:

$$\begin{aligned}\left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2 &= \operatorname{trace}\left(\left(E_r^H E_\theta - E_\theta^H E_r\right)\left(E_r^H E_\theta - E_\theta^H E_r\right)^H\right) \\ &= \operatorname{trace}\left(\left(E_r^H E_\theta - E_\theta^H E_r\right)\left(E_\theta^H E_r - E_r^H E_\theta\right)\right) \\ &= \operatorname{trace}\left(+E_r^H E_\theta E_\theta^H E_r - E_r^H E_\theta E_r^H E_\theta - E_\theta^H E_r E_\theta^H E_r + E_\theta^H E_r E_r^H E_\theta\right)\end{aligned}$$

As $\operatorname{trace}\left(E_r^H E_\theta E_\theta^H E_r\right) = \operatorname{trace}\left(E_\theta^H E_r E_r^H E_\theta\right)$, we can write:

$$\begin{aligned}\left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2 &= \operatorname{trace}\left(+2\, E_r^H E_\theta E_\theta^H E_r - E_r^H E_\theta E_r^H E_\theta - E_\theta^H E_r E_\theta^H E_r\right) \\ &= \operatorname{trace}\left(+2\, E_r E_r^H E_\theta E_\theta^H - E_\theta E_r^H E_\theta E_r^H - E_r E_\theta^H E_r E_\theta^H\right)\end{aligned} \tag{E.468}$$

Hence we have:

$$\operatorname{trace}\left(E_r E_r^H E_\theta E_\theta^H - E_r E_\theta^H E_r E_r^H + E_\theta E_r^H E_r E_\theta^H - E_\theta E_r^H E_\theta E_r^H\right) \tag{E.469}$$

$$= \frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right) \tag{E.470}$$

Therefore Equation (E.466) becomes:

$$\operatorname{trace}(S_4) = \frac{1}{3}\left(\frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right)\right) d_\theta^2 \tag{E.471}$$

We recall Equation (E.443):

$$ds^2 = \operatorname{trace}(S_0) + \operatorname{trace}(S_1)\, r + \operatorname{trace}(S_2)\, r^2 + \operatorname{trace}(S_3)\, r^3 + \operatorname{trace}(S_4)\, r^4 + O\left(r^5\right)$$

Therefore, we have from Equations (E.447), (E.451), (E.455), (E.462) and (E.471):

$$\begin{aligned}ds^2 = \;& dr^2 \\ & + d\theta^2 r^2 \\ & + \frac{1}{3}\left(\frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right)\right) d\theta^2 r^4 \\ & + O\left(r^5\right)\end{aligned} \tag{E.472}$$

$$\begin{aligned}ds^2 =& dr^2 \\ & + \left(1 - \frac{1}{3}\left(-\frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right) r^2\right)\right) r^2 d\theta^2 \\ & + O\left(r^5\right)\end{aligned} \tag{E.473}$$

We recall Equation (D.2) which gives a link between the metric element ds^2 expressed in polar coordinates and the sectional curvature [31]:

$$ds^2 = dr^2 + \left(1 - \frac{1}{3}K(\sigma)r^2\right) r^2 d\theta^2 + O\left(r^5\right) dr^2 + O\left(r^5\right) drd\theta + O\left(r^5\right) d\theta^2 \tag{E.474}$$

Note that according to this expression, we only needed to compute the coefficient of order four $\mathrm{trace}\,(S_4)$ to obtain the sectional curvature $K\,(\sigma)$.

By identification, we finally show that:

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right) \tag{E.475}$$

□

We can check that this formula corresponds to the result of Theorem 10 in Euclidean coordinates:

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right) \tag{E.476}$$

We start by computing the products using Equations (E.357) and (E.358):

$$\begin{aligned} E_r E_\theta^H &= (\cos(\theta) E_1 + \sin(\theta) E_2)\,(-\sin(\theta) E_1 + \cos(\theta) E_2)^H \\ E_r E_\theta^H &= (\cos(\theta) E_1 + \sin(\theta) E_2)\left(-\sin(\theta) E_1^H + \cos(\theta) E_2^H\right) \\ E_r E_\theta^H &= -\cos(\theta)\sin(\theta) E_1 E_1^H + \cos^2(\theta) E_1 E_2^H - \sin^2(\theta) E_2 E_1^H + \cos(\theta)\sin(\theta) E_2 E_2^H \end{aligned} \tag{E.477}$$

$$\begin{aligned} E_\theta E_r^H &= (-\sin(\theta) E_1 + \cos(\theta) E_2)\,(\cos(\theta) E_1 + \sin(\theta) E_2)^H \\ E_\theta E_r^H &= (-\sin(\theta) E_1 + \cos(\theta) E_2)\left(\cos(\theta) E_1^H + \sin(\theta) E_2^H\right) \\ E_\theta E_r^H &= -\cos(\theta)\sin(\theta) E_1 E_1^H - \sin^2(\theta) E_1 E_2^H + \cos^2(\theta) E_2 E_1^H + \cos(\theta)\sin(\theta) E_2 E_2^H \end{aligned} \tag{E.478}$$

Finally, we have:

$$\begin{aligned} E_r E_\theta^H - E_\theta E_r^H &= \left(\cos^2(\theta) + \sin^2(\theta)\right) E_1 E_2^H + \left(-\cos^2(\theta) - \sin^2(\theta)\right) E_2 E_1^H \\ E_r E_\theta^H - E_\theta E_r^H &= E_1 E_2^H - E_2 E_1^H \end{aligned} \tag{E.479}$$

With $E_r \leftarrow E_r^H$ and $E_\theta \leftarrow E_\theta^H$, we show as well that:

$$E_r^H E_\theta - E_\theta^H E_r = E_1^H E_2 - E_2^H E_1 \tag{E.480}$$

Which confirmes that the two expressions of $K\,(\sigma)$ correspond: the expression given in polar coordinates in Theorem 11 and the expression given in Euclidean coordinates in Theorem 10:

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_r E_\theta^H - E_\theta E_r^H\right\|^2 + \left\|E_r^H E_\theta - E_\theta^H E_r\right\|^2\right) \tag{E.481}$$

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right) \tag{E.482}$$

E.7.3 Infinitesimal right triangles and sectional curvature in the Siegel space

In addition to the proof of the expression of the sectional curvature using Euclidean coordinates E.7.1 and using polar coordinates E.7.2, we will prove the expression of the sectional curvature by using the length of the hypotenuse of an infinitesimal right triangle.

The link between the Taylor series of the square length of the hypothenuse of an infinitesimal right triangle and the sectional curvature has been studied in Section D.4 whose notations we will used.

Theorem 12. *We consider a right triangle OAB rectangle in O whose point O is located at zero, i.e. the null matrix. In the Siegel manifold, the limited expansion of the square length of the hypotenuse is:*

$$d^2(A,B) = a^2 + b^2 + \frac{1}{6}\left(\left\|E_1 E_2^H - E_2 E_1^H\right\|^2 + \left\|E_1^H E_2 - E_2^H E_1\right\|^2\right) a^2 b^2 + O\left((|a| + |b|)^5\right) \tag{E.483}$$

Corollary 3. *According to Theorem 8, this is another proof that the sectional curvature at zero of the plan σ defined by E_1 and E_2 has the following expression:*

$$K(\sigma) = -\frac{1}{2}\left(\left\|E_1E_2^H - E_2E_1^H\right\|^2 + \left\|E_1^HE_2 - E_2^HE_1\right\|^2\right) \tag{E.484}$$

Proof. We consider the infinitesimal right triangle OAB rectangle in O whose point O is located at zero.
We set u and v the vectors of the tangent space at zero such that:

- $A = \exp_0(\lambda u)$
- $B = \exp_0(\lambda v)$
- $\langle u, v\rangle_0 = 0 \quad \Leftrightarrow \operatorname{trace}\left(uv^H + vu^H\right) = 0$

where λ is a positive real.
We calculate the Taylor series of the length squared of the hypotenuse $d^2(A,B)$ as λ approaches 0.
According to Section E.2 on the metric of the Siegel manifold, we already have:

$$d^2(O,A) = \|\lambda u\|_0^2 = \lambda^2 \operatorname{trace}\left(uu^H\right) \tag{E.485}$$
$$d^2(O,B) = \|\lambda v\|_0^2 = \lambda^2 \operatorname{trace}\left(vv^H\right) \tag{E.486}$$
$$d^2(A,B) = \operatorname{trace}\left(\operatorname{arctanh}^2\left(C_{A,B}^{1/2}\right)\right) \tag{E.487}$$
$$\text{where} \quad C_{A,B} = (B-A)\left(I - A^HB\right)^{-1}\left(B^H - A^H\right)\left(I - AB^H\right)^{-1}$$

According to Section E.5.1 dealing with the expression of the Riemmannian exponential map of the Siegel manifold at zero, we have:

$$A = \exp_0(\lambda u) \tag{E.488}$$
$$A = \tanh(\lambda X)(\lambda X)^{-1}(\lambda u) \quad \text{where } X = \left(uu^H\right)^{1/2} \tag{E.489}$$
$$A = \tanh(\lambda X)X^{-1}u. \tag{E.490}$$

Likewise, we have:

$$B = \tanh(\lambda Y)Y^{-1}v \quad \text{where } Y = \left(vv^H\right)^{1/2}. \tag{E.491}$$

We recall the Taylor series of the hyperbolic tangent function at zero:

$$\tanh(x) = x - \frac{1}{3}x^3 + O\left(x^5\right) \quad \forall |x| < \frac{\pi}{2}. \tag{E.492}$$

We therefore have:

$$A = \tanh(\lambda X)X^{-1}u \tag{E.493}$$
$$A = \left(X\lambda - \frac{1}{3}X^3\lambda^3 + O\left(\lambda^5\right)\right)X^{-1}u \tag{E.494}$$
$$A = u\lambda - \frac{1}{3}X^2u\lambda^3 + O\left(\lambda^5\right) \tag{E.495}$$
$$A = u\lambda - \frac{1}{3}uu^Hu\lambda^3 + O\left(\lambda^5\right) \tag{E.496}$$

Likewise, we have:

$$B = v\lambda - \frac{1}{3}vv^Hv\lambda^3 + O\left(\lambda^5\right) \tag{E.497}$$

We now calculate the Taylor series of the four terms of the product $C_{A,B}$ of Equation (E.487).
The first term is $(B-A)$:

$$B - A = (v-u)\lambda - \frac{1}{3}\left(vv^Hv - uu^Hu\right)\lambda^3 + O\left(\lambda^5\right). \tag{E.498}$$

The third term is $\left(B^H - A^H\right) = (B-A)^H$, which gives by transconjugating the previous formula:

$$B^H - A^H = (v^H - u^H)\lambda - \frac{1}{3}\left(v^Hvv^H - u^Huu^H\right)\lambda^3 + O\left(\lambda^5\right). \tag{E.499}$$

We now calculate the products $A^H B$ and AB^H:

$$A^H B = \left(u^H \lambda + O\left(\lambda^3\right)\right)\left(v\lambda + O\left(\lambda^3\right)\right) \tag{E.500}$$
$$A^H B = u^H v \lambda^2 + O\left(\lambda^4\right). \tag{E.501}$$

Likewise, we have:

$$AB^H = uv^H \lambda^2 + O\left(\lambda^4\right). \tag{E.502}$$

We recall the following Taylor series:

$$(1-x)^{-1} = 1 + x + O\left(x^2\right) \quad \forall\, |x| < 1. \tag{E.503}$$

This formula will be used to compute the Taylor series of the second and fourth terms of $C_{A,B}$ defined in Equation (E.487).

We now compute the Taylor series of the second term:

$$\left(I - A^H B\right)^{-1} = I + A^H B + O\left(\left(A^H B\right)^2\right) \tag{E.504}$$
$$\left(I - A^H B\right)^{-1} = I + u^H v \lambda^2 + O\left(\lambda^4\right). \tag{E.505}$$

Finally, the Taylor series of the fourth term is:

$$\left(I - AB^H\right)^{-1} = I + AB^H + O\left(\left(AB^H\right)^2\right) \tag{E.506}$$
$$\left(I - AB^H\right)^{-1} = I + uv^H \lambda^2 + O\left(\lambda^4\right). \tag{E.507}$$

We now calculate the Taylor series of the product of the first and the second term of $C_{A,B}$ defined in Equation (E.487).

$$\left(B - A\right)\left(I - A^H B\right)^{-1} \tag{E.508}$$
$$= \left((v-u)\lambda - \frac{1}{3}\left(vv^H v - uu^H u\right)\lambda^3 + O\left(\lambda^5\right)\right)\left(I + u^H v\lambda^2 + O\left(\lambda^4\right)\right) \tag{E.509}$$
$$= (v-u)\lambda + \left((v-u)u^H v - \frac{1}{3}\left(vv^H v - uu^H u\right)\right)\lambda^3 + O\left(\lambda^5\right). \tag{E.510}$$

We then calculate the Taylor series of the product of the third and fourth terms of $C_{A,B}$ defined in Equation (E.487).

$$\left(B^H - A^H\right)\left(I - AB^H\right)^{-1} \tag{E.511}$$
$$= \left((v^H - u^H)\lambda - \frac{1}{3}\left(v^H v v^H - u^H u u^H\right)\lambda^3 + O\left(\lambda^5\right)\right)\left(I + uv^H\lambda^2 + O\left(\lambda^4\right)\right) \tag{E.512}$$
$$= (v^H - u^H)\lambda + \left((v^H - u^H)uv^H - \frac{1}{3}\left(v^H v v^H - u^H u u^H\right)\right)\lambda^3 + O\left(\lambda^5\right). \tag{E.513}$$

We can now calculate the Taylor series of the product of four factors $C_{A,B}$ defined in Equation (E.487). by multiplying the Taylor series obtained in Equations (E.510) and (E.513).

$$C_{A,B} := (B-A)\left(I-A^HB\right)^{-1}\left(B^H-A^H\right)\left(I-AB^H\right)^{-1} \tag{E.514}$$

$$= \left((v-u)\lambda + \left((v-u)u^Hv - \frac{1}{3}\left(vv^Hv - uu^Hu\right)\right)\lambda^3 + O\left(\lambda^5\right)\right) \tag{E.515}$$
$$\left((v^H-u^H)\lambda + \left((v^H-u^H)uv^H - \frac{1}{3}\left(v^Hvv^H - u^Huu^H\right)\right)\lambda^3 + O\left(\lambda^5\right)\right)$$

$$\begin{aligned}=&(v-u)\left(v^H-u^H\right)\lambda^2 \\ &+\Big[(v-u)\left(v^H-u^H\right)uv^H \\ &-\frac{1}{3}(v-u)\left(v^Hvv^H-u^Huu^H\right) \\ &+(v-u)u^Hv\left(v^H-u^H\right) \\ &-\frac{1}{3}\left(vv^Hv-uu^Hu\right)\left(v^H-u^H\right)\Big]\lambda^4 \\ &+O\left(\lambda^6\right)\end{aligned} \tag{E.516}$$

We now study the Taylor series of $\operatorname{arctanh}^2\left(C_{A,B}^{1/2}\right)$.
We recall the Taylor series of the function $\operatorname{arctanh}$:

$$\operatorname{arctanh}(x) = x + \frac{1}{3}x^3 + O\left(x^5\right) \quad \forall\, |x| < 1. \tag{E.517}$$

The limited expansion of the function $x \mapsto \operatorname{arctanh}^2(x)$ is therefore:

$$\operatorname{arctanh}^2(x) = x^2 + \frac{2}{3}x^4 + O\left(x^6\right) \quad \forall\, |x| < 1. \tag{E.518}$$

Hence, we have:

$$\operatorname{arctanh}^2\left(C_{A,B}^{1/2}\right) = C_{A,B} + \frac{2}{3}C_{A,B}^2 + O\left(C_{A,B}^3\right). \tag{E.519}$$

Using the Taylor series of $C_{A,B}$ obtained in Equation (E.516), we have:

$$\begin{aligned}\operatorname{arctanh}^2\left(C_{A,B}^{1/2}\right) =&(v-u)\left(v^H-u^H\right)\lambda^2 \\ &+\Big[(v-u)\left(v^H-u^H\right)uv^H \\ &-\frac{1}{3}(v-u)\left(v^Hvv^H-u^Huu^H\right) \\ &+(v-u)u^Hv\left(v^H-u^H\right) \\ &-\frac{1}{3}\left(vv^Hv-uu^Hu\right)\left(v^H-u^H\right) \\ &+\frac{2}{3}\left((v-u)\left(v^H-u^H\right)\right)^2\Big]\lambda^4 \\ &+O\left(\lambda^6\right).\end{aligned} \tag{E.520}$$

We now develop the products of the previous equation. For terms in λ^2, we have:

$$(v-u)\left(v^H-u^H\right) = vv^H - vu^H - uv^H + uu^H. \tag{E.521}$$

For the terms in λ^4, we set:

$$C_1 = (v-u)\left(v^H-u^H\right)uv^H \tag{E.522}$$

$$C_2 = -\frac{1}{3}(v-u)\left(v^Hvv^H-u^Huu^H\right) \tag{E.523}$$

$$C_3 = (v-u)u^Hv\left(v^H-u^H\right) \tag{E.524}$$

$$C_4 = -\frac{1}{3}\left(vv^Hv-uu^Hu\right)\left(v^H-u^H\right) \tag{E.525}$$

$$C_5 = \frac{2}{3}\left((v-u)\left(v^H-u^H\right)\right)^2. \tag{E.526}$$

We denote by C the term in λ^4. According to the previous equations, we have $C = C_1 + C_2 + C_3 + C_4 + C_5$. We now develop the terms C_1, C_2, C_3, C_4 and C_5.

$$\begin{aligned}
C_1 =&(v-u)\left(v^H-u^H\right)uv^H && \text{(E.527)}\\
=&\left(vv^H-vu^H-uv^H+uu^H\right)uv^H && \text{(E.528)}\\
=&vv^Huv^H-vu^Huv^H-uv^Huv^H+uu^Huv^H && \text{(E.529)}
\end{aligned}$$

$$\begin{aligned}
C_2 =&-\frac{1}{3}(v-u)\left(v^Hvv^H-u^Huu^H\right) && \text{(E.530)}\\
=&-\frac{1}{3}\left(vv^Hvv^H-vu^Huu^H-uv^Hvv^H+uu^Huu^H\right) && \text{(E.531)}\\
=&-\frac{1}{3}vv^Hvv^H+\frac{1}{3}vu^Huu^H+\frac{1}{3}uv^Hvv^H-\frac{1}{3}uu^Huu^H && \text{(E.532)}
\end{aligned}$$

$$\begin{aligned}
C_3 =&(v-u)u^Hv\left(v^H-u^H\right) && \text{(E.533)}\\
=&\left(vu^Hv-uu^Hv\right)\left(v^H-u^H\right) && \text{(E.534)}\\
=&vu^Hvv^H-vu^Hvu^H-uu^Hvv^H+uu^Hvu^H && \text{(E.535)}
\end{aligned}$$

$$\begin{aligned}
C_4 =&-\frac{1}{3}\left(vv^Hv-uu^Hu\right)\left(v^H-u^H\right) && \text{(E.536)}\\
=&-\frac{1}{3}\left(vv^Hvv^H-vv^Hvu^H-uu^Huv^H+uu^Huu^H\right) && \text{(E.537)}\\
=&-\frac{1}{3}vv^Hvv^H+\frac{1}{3}vv^Hvu^H+\frac{1}{3}uu^Huv^H-\frac{1}{3}uu^Huu^H && \text{(E.538)}
\end{aligned}$$

$$\begin{aligned}
C_5 =&\frac{2}{3}\left((v-u)\left(v^H-u^H\right)\right)^2 && \text{(E.539)}\\
=&\frac{2}{3}\left(vv^H-vu^H-uv^H+uu^H\right)^2 && \text{(E.540)}\\
=&\frac{2}{3}\left(vv^Hvv^H-vv^Hvu^H-vv^Huv^H+vv^Huu^H\right. && \text{(E.541)}\\
&-vu^Hvv^H+vu^Hvu^H+vu^Huv^H-vu^Huu^H && \text{(E.542)}\\
&-uv^Hvv^H+uv^Hvu^H+uv^Huv^H-uv^Huu^H && \text{(E.543)}\\
&\left.+uu^Hvv^H-uu^Hvu^H-uu^Huv^H+uu^Huu^H\right) && \text{(E.544)}\\
=&+\frac{2}{3}vv^Hvv^H-\frac{2}{3}vv^Hvu^H-\frac{2}{3}vv^Huv^H+\frac{2}{3}vv^Huu^H && \text{(E.545)}\\
&-\frac{2}{3}vu^Hvv^H+\frac{2}{3}vu^Hvu^H+\frac{2}{3}vu^Huv^H-\frac{2}{3}vu^Huu^H && \text{(E.546)}\\
&-\frac{2}{3}uv^Hvv^H+\frac{2}{3}uv^Hvu^H+\frac{2}{3}uv^Huv^H-\frac{2}{3}uv^Huu^H && \text{(E.547)}\\
&+\frac{2}{3}uu^Hvv^H-\frac{2}{3}uu^Hvu^H-\frac{2}{3}uu^Huv^H+\frac{2}{3}uu^Huu^H && \text{(E.548)}
\end{aligned}$$

We can now calculate the term in λ^4:

$$\begin{aligned}
C =&C_1+C_2+C_3+C_4+C_5 && \text{(E.549)}\\
C =&+0\ vv^Hvv^H-\frac{1}{3}vv^Hvu^H+\frac{1}{3}vv^Huv^H+\frac{2}{3}vv^Huu^H && \text{(E.550)}\\
&+\frac{1}{3}vu^Hvv^H-\frac{1}{3}vu^Hvu^H-\frac{1}{3}vu^Huv^H-\frac{1}{3}vu^Huu^H && \text{(E.551)}\\
&-\frac{1}{3}uv^Hvv^H+\frac{2}{3}uv^Hvu^H-\frac{1}{3}uv^Huv^H-\frac{2}{3}uv^Huu^H && \text{(E.552)}\\
&-\frac{1}{3}uu^Hvv^H+\frac{1}{3}uu^Hvu^H+\frac{2}{3}uu^Huv^H+0\ uu^Huu^H. && \text{(E.553)}
\end{aligned}$$

According to Equation (E.487), we have:

$$d^2(A,B) = \operatorname{trace}\left(\operatorname{arctanh}^2\left(C_{A,B}^{1/2}\right)\right) \tag{E.554}$$

$$\text{where} \quad C_{A,B} = (B-A)\left(I - A^H B\right)^{-1}\left(B^H - A^H\right)\left(I - AB^H\right)^{-1}. \tag{E.555}$$

We have obtained a Taylor series of $\operatorname{arctanh}^2\left(C_{A,B}^{1/2}\right)$ as λ approaches zero. The term in λ^2 is given in Equation (E.521) and the term in λ^4 is given in Equation (E.550). To obtain the Taylor series of $d^2(A,B)$, we will therefore apply the trace operator to the Taylor series obtained previously. By linearity of the trace operator, it suffices to calculate the trace of the term in λ^2 and the trace of the term in λ^4.

For the term in λ^2 given in Equation (E.521), we obtain:

$$\begin{aligned}
&\operatorname{trace}\left(vv^H - vu^H - uv^H + uu^H\right) && \text{(E.556)}\\
=&\operatorname{trace}\left(vv^H\right) - \operatorname{trace}\left(vu^H + uv^H\right) + \operatorname{trace}\left(uu^H\right) && \text{(E.557)}\\
=&\operatorname{trace}\left(vv^H\right) - \langle v,u\rangle_0 + \operatorname{trace}\left(uu^H\right) && \text{(E.558)}\\
=&\operatorname{trace}\left(vv^H\right) + \operatorname{trace}\left(uu^H\right). && \text{(E.559)}
\end{aligned}$$

To calculate the trace of the term in λ^4 given in Equation (E.550), we recall that $\operatorname{trace}(XY) = \operatorname{trace}(YX)$, therefore many simplifications can be obtained.

$$\operatorname{trace}(C) = \operatorname{trace}\left(\frac{1}{3}vv^H uu^H - \frac{1}{3}vu^H vu^H + \frac{1}{3}vu^H uv^H - \frac{1}{3}uv^H uv^H\right) \tag{E.560}$$

$$\operatorname{trace}(C) = \frac{1}{3}\left(\operatorname{trace}\left(vv^H uu^H\right) - \operatorname{trace}\left(vu^H vu^H\right) + \operatorname{trace}\left(vu^H uv^H\right) - \operatorname{trace}\left(uv^H uv^H\right)\right) \tag{E.561}$$

We now show that:

$$\begin{aligned}
&\operatorname{trace}\left(vv^H uu^H\right) - \operatorname{trace}\left(vu^H vu^H\right) + \operatorname{trace}\left(vu^H uv^H\right) - \operatorname{trace}\left(uv^H uv^H\right)\\
=&\frac{1}{2}\left(\left\|vu^H - uv^H\right\|^2 + \left\|v^H u - u^H v\right\|^2\right) && \text{(E.562)}
\end{aligned}$$

Indeed, we have:

$$\begin{aligned}
&\left\|vu^H - uv^H\right\|^2\\
=&\operatorname{trace}\left(\left(vu^H - uv^H\right)\left(vu^H - uv^H\right)^H\right) && \text{(E.563)}\\
=&\operatorname{trace}\left(\left(vu^H - uv^H\right)\left(uv^H - vu^H\right)\right) && \text{(E.564)}\\
=&\operatorname{trace}\left(vu^H uv^H - vu^H vu^H - uv^H uv^H + uv^H vu^H\right) && \text{(E.565)}\\
=&\operatorname{trace}\left(vu^H uv^H\right) - \operatorname{trace}\left(vu^H vu^H\right) - \operatorname{trace}\left(uv^H uv^H\right) + \operatorname{trace}\left(uv^H vu^H\right) && \text{(E.566)}\\
=&2\operatorname{trace}\left(vu^H uv^H\right) - \operatorname{trace}\left(vu^H vu^H\right) - \operatorname{trace}\left(uv^H uv^H\right). && \text{(E.567)}
\end{aligned}$$

We also have:

$$\begin{aligned}
&\left\|v^H u - u^H v\right\|^2\\
=&\operatorname{trace}\left(\left(v^H u - u^H v\right)\left(v^H u - u^H v\right)^H\right) && \text{(E.568)}\\
=&\operatorname{trace}\left(\left(v^H u - u^H v\right)\left(u^H v - v^H u\right)\right) && \text{(E.569)}\\
=&\operatorname{trace}\left(v^H uu^H v - v^H uv^H u - u^H vu^H v + u^H vv^H u\right) && \text{(E.570)}\\
=&\operatorname{trace}\left(v^H uu^H v\right) - \operatorname{trace}\left(v^H uv^H u\right) - \operatorname{trace}\left(u^H vu^H v\right) + \operatorname{trace}\left(u^H vv^H u\right) && \text{(E.571)}\\
=&\operatorname{trace}\left(vv^H uu^H\right) - \operatorname{trace}\left(uv^H uv^H\right) - \operatorname{trace}\left(vu^H vu^H\right) + \operatorname{trace}\left(vv^H uu^H\right) && \text{(E.572)}\\
=&2\operatorname{trace}\left(vv^H uu^H\right) - \operatorname{trace}\left(uv^H uv^H\right) - \operatorname{trace}\left(vu^H vu^H\right). && \text{(E.573)}
\end{aligned}$$

By adding the terms of Equations (E.567) and (E.573), we prove the equality of Equation (E.562).

According to Equations (E.561) and (E.562), we have:

$$\operatorname{trace}(C) = \frac{1}{6}\left(\left\|vu^H - uv^H\right\|^2 + \left\|v^H u - u^H v\right\|^2\right). \tag{E.574}$$

According to Equation (E.559) which details the term in λ^2 and to Equation (E.574) which details the term in λ^4, we have the following Taylor series of the squared distance $d^2(A, B)$:

$$d^2(A, B) = \left(\operatorname{trace}\left(vv^H\right) + \operatorname{trace}\left(uu^H\right)\right)\lambda^2 + \frac{1}{6}\left(\left\|vu^H - uv^H\right\|^2 + \left\|v^H u - u^H v\right\|^2\right)\lambda^4 + O\left(\lambda^6\right). \tag{E.575}$$

□

Appendix F

Applications to radar detection

The estimation of the clutter covariance matrix is an already known problem in radar detection [26]: it is useful to obtain a detection estimator with a constant false alarm rate (CFAR). A fixed point estimator is used in [57]. The article of Le Yang [6] uses the Riemannian median in the manifold $\mathbb{R}^{++} \times \mathbb{D}^{n-1}$ the notice outliers and perform radar detection.

Contents

We denote $\mathcal{T}_n^+$ denotes the set of Toeplitz Hermitian Positive Definite matrices of size n.

F.1 Detection using a known steering vector

As in [24], we would like to detect a perfectly known complex steering vector p multiplied by an unknown complex number a characterizing the signal amplitude and initial phase and disturbed by an additive noise $c \sim \mathcal{CN}(0, M)$. Given the complex observation vector y, the two hypothesis confronted in this detection problem are the following:

$$\begin{cases} H_0 : & y = c \\ H_1 : & y = ap + c \end{cases} \tag{F.1}$$

We first use the previous sections to estimate the clutter correlation matrix M of the area surrounding the cell under study. We would like to construct a CFAR (Constant False Alarm Rate) test: the probability of declaring H_1 under H_0 should not depend on the nature of the clutter c characterized by its correlation matrix M. We define the adaptive normalized matched filter (ANMF) dector T:

$$T(y) = \frac{\left| y^H \widehat{M}^{-1} p \right|}{\left| y^H \widehat{M}^{-1} y \right|^{\frac{1}{2}} \left| p^H \widehat{M}^{-1} p \right|^{\frac{1}{2}}} \tag{F.2}$$

If we denote $\widetilde{y} = \widehat{M}^{-\frac{1}{2}} y$ and $\widetilde{p} = \widehat{M}^{-\frac{1}{2}} p$, then T can be expressed the following way:

$$T(y) = \left| \left\langle \frac{\widetilde{y}}{\|\widetilde{y}\|}, \frac{\widetilde{p}}{\|\widetilde{p}\|} \right\rangle \right| = |\cos(\theta)| \tag{F.3}$$

where θ denotes the real-valued Euclidean angle between the normalized vectors $\frac{\widetilde{y}}{\|\widetilde{y}\|}$ and $\frac{\widetilde{p}}{\|\widetilde{p}\|}$ seen as real vectors, their real and imaginary parts being considered separately.

If we assume that $\widehat{M} \simeq M$ and denote $x \sim \mathcal{CN}(0, I)$ such that $c = M^{\frac{1}{2}} x$, then the statistic T follows the following laws:

- Under H_0, $T \sim \left| \left\langle \frac{x}{\|x\|}, \frac{\widetilde{p}}{\|\widetilde{p}\|} \right\rangle \right|$
- Under H_1, $T \sim \left| \left\langle \frac{x+a\widetilde{p}}{\|x+a\widetilde{p}\|}, \frac{\widetilde{p}}{\|\widetilde{p}\|} \right\rangle \right|$

Hence, the angle θ should be uniformly distributed over $[-\pi, \pi]$ under H_0 and T should be rather close to 1 under H_1.

Therefore, we reject the hypothesis H_0 when T is close to 1, i.e. θ belongs to an interval of shape $[\pi - \beta, \pi + \beta] \cup [-\beta, \beta]$. We now want to construct a test of level α, i.e. we search β such that:

$$\mathbb{P}_{H_0} \left(\theta \in [\pi - \beta, \pi + \beta] \cup [-\beta, \beta] \right) = \alpha \tag{F.4}$$

Hence, since θ is uniformly distributed over $[-\pi, \pi]$ under H_0, $\frac{4\beta}{2\pi} = \alpha$ and then $\beta = \frac{\pi}{2}\alpha$.
Finally, the hypothesis H_0 is rejected when y belongs the rejection area:

$$R = \left\{ y \in \mathbb{C}^n, T(y) \in \left[\cos\left(\frac{\pi}{2}\alpha\right), 1 \right] \right\} \tag{F.5}$$

F.2 Detection using a known autocorrelation matrix

We now assume that the autocorrelation matrix $P \in \mathcal{T}_n^+$ of the signal we try to detect is known. The two hypothesis confronted in this detection problem are:

$$\begin{cases} H_0 : & y = c, \quad c \sim \mathcal{N}(0, M) \\ H_1 : & y = p + c, \quad p \sim \mathcal{N}(0, P) \text{ and } \ c \sim \mathcal{N}(0, M) \end{cases} \tag{F.6}$$

We consider the statistical model $\left(\mathbb{C}^n, \{\mathcal{N}(0, \Sigma)\}_{\Sigma \in \mathcal{T}_n^+}\right)$ such that for $\Sigma \in \mathcal{T}_n^+$, $\mathcal{N}(0, \Sigma)$ denotes the law on $\mathbb{C}^n$ of density:

$$f_\Sigma(z) = \frac{1}{\pi^n \det(\Sigma)} e^{z^H \Sigma^{-1} z} \tag{F.7}$$

The two hypothesis can then be reformulated as:

$$\begin{cases} H_0 : \Sigma = \Sigma_0 = M \\ H_1 : \Sigma = \Sigma_1 = P + M \end{cases} \tag{F.8}$$

We construct a test of level α looking for $\gamma \in \mathbb{R}_+^*$ such that:

$$\mathbb{P}_{\Sigma_0}\left(\frac{f_{\Sigma_1}(z)}{f_{\Sigma_0}(z)} \geqslant \gamma \right) = \alpha \tag{F.9}$$

The hypothesis H_0 is then rejected when y belongs the rejection area:

$$R = \left\{ y \in \mathbb{C}^n, \frac{f_{\Sigma_1}(y)}{f_{\Sigma_0}(y)} \geqslant \gamma \right\} \tag{F.10}$$

To construct another test of level α with a more geometrical approach, we can estimate the autocorrelation matrix $\widehat{\Sigma_y} \in \mathcal{T}_n^+$ of y using the Burg algorithm. We then look for γ such that:

$$\mathbb{P}_{\Sigma_0}\left(\frac{d_{\mathcal{T}_n^+}\left(\Sigma_1, \widehat{Y}\right)}{d_{\mathcal{T}_n^+}\left(\Sigma_0, \widehat{Y}\right)} \geqslant \gamma \right) = \alpha \tag{F.11}$$

The hypothesis H_0 is therefore rejected when y belongs the rejection area:

$$R = \left\{ y \in \mathbb{C}^n, \frac{d_{\mathcal{T}_n^+}\left(\Sigma_1, \widehat{Y}\right)}{d_{\mathcal{T}_n^+}\left(\Sigma_0, \widehat{Y}\right)} \geqslant \gamma \right\} \tag{F.12}$$

Appendix G

The radar clutter spatio-temporal model

In this appendix, we assume that the times series recorded by the radar during a burst are wide-sense stationary on both the temporal axis and the spatial axis, the spatial axis considered being the radial axis. In Chapter 8, the radar clutter is clustered cell per cell. The spatio-temporal model presented here generalize the temporal model described in Chapter 8: we use the spatial correlation between spatially close cells in addition to the Doppler information contained in each cell. In the article written by Romain Couillet et al. [24], the space correlation coefficient of sea clutter data is studied for different range resolutions: the periodicity of the sea surface is highlighted by the periodicity of the spatial correlation coefficient for a range resolution of 3 meters. The spatio-temporal model is also a particular case of the multidimensional stationary centered complex Gaussian autoregressive model presented in Section 2.2. Using the assumption of stationarity on both the temporal and spatial axes, we construct a Riemannian manifold to represent the information contained in a group of spatially close cells. This model is also presented in detail in [83].

Contents

G.1 The input data

We recall the context of radar clutter classification presented in Chapter 8. During each burst, we represent the information associated with the observed angular sector in the form of a matrix U:

$$U = \begin{bmatrix} \boxed{\begin{matrix} u_{0,0} & u_{0,1} & \cdots & u_{0,n-1} \\ u_{1,0} & u_{1,1} & \cdots & u_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{p-1,0} & u_{p-1,1} & \cdots & u_{p-1,n-1} \end{matrix}} \\ \vdots \quad \vdots \quad \ddots \quad \vdots \\ \boxed{u_{N-1,0} \quad u_{N-1,1} \quad \cdots \quad u_{N-1,n-1}} \end{bmatrix}. \tag{G.1}$$

Each row of the matrix U represents the same distance cell and each column represents the same pulse. In this matrix, N represents the number of distance cells and n represents the number of pulses emitted during the burst.

In Chapter 8, we clustered each cell, i.e. each row of the matrix U, independently from one another based on its reflection coefficients which characterize the Doppler information. In this appendix, we would like to use the spatial correlation between consecutive cells in addition to the Doppler information contained in each cell.

Here the input data is not a single vector corresponding to a row of the matrix U, but a matrix corresponding to several consecutive rows of the matrix U. For example, the following matrix contains the p first rows of the matrix U:

$$Z = \begin{bmatrix} u_{0,0} & u_{0,1} & \cdots & u_{0,n-1} \\ u_{1,0} & u_{1,1} & \cdots & u_{1,n-1} \\ \vdots & \vdots & \ddots & \vdots \\ u_{p-1,0} & u_{p-1,1} & \cdots & u_{p-1,n-1} \end{bmatrix}. \tag{G.2}$$

For the hypothesis of wide-stationarity on the spatial axis to be respected, the p cells studied together in the matrix Z should belong to the same kind of clutter.

G.2 The model

We first present in Section G.2.1 a spatio-temporal model which generalizes the one-dimensional temporal model presented in Chapter 8 for radar clutter clustering (see Section 8.2). In this first model, a texture coefficient is present (see [13] for more details about the clutter texture). We then present in Section G.2.2 a simplified spatio-temporal model. For simplicity, we will refer to this second model to present the theory of following sections.

G.2.1 The general radar model

We assume that the matrix Z presented in Equation (G.2) has the following structure:

$$Z = \underbrace{T^{1/2} R_s^{1/2} N R_t^{T^{1/2}}}_{\substack{\text{information coming from} \\ \text{the environment}}} + \underbrace{B_{radar}}_{\substack{\text{thermal noise coming} \\ \text{from the radar itself}}} \tag{G.3}$$

with:

- Z: radar observation matrix of size (p, n).
- T: clutter texture; it is a diagonal matrix of size (p, p). Its diagonal coefficients are independent positive real random variables. They are also independent from N and B_{radar}.
- R_s: the scaled spatial autocorrelation matrix of size (p, p). It is a Toeplitz HPD matrix since the signal is assumed to be wide-sense stationary on the spatial axis. Its diagonal coefficients are equal to 1.
- R_t: the scaled temporal autocorrelation matrix of size (n, n). It is a Toeplitz HPD matrix since the signal is assumed to be wide-sense stationary on the temporal axis. Its diagonal coefficients are equal to 1.
- N, B_{radar}: matrices of size (p, n) whose coefficients are independent standard complex Gaussian random variables.

We now present an equivalent vectorized model:

$$\widetilde{Z} = \underbrace{\widetilde{T}^{1/2} R_{st}^{1/2} \widetilde{N}}_{\substack{\text{information coming from} \\ \text{the environment}}} + \underbrace{\widetilde{B}_{radar}}_{\substack{\text{thermal noise coming} \\ \text{from the radar itself}}} \tag{G.4}$$

with:

- $\widetilde{Z}$: radar observation matrix of size $(p \times n, 1)$.
- $\widetilde{T}$: clutter texture; it is a diagonal matrix of size $(p \times n, p \times n)$. Its diagonal coefficients are independent positive real random variables. They are also independent of $\widetilde{N}$ and $\widetilde{B}_{radar}$.
- R_{st}: the scaled spatio-temporal autocorrelation matrix of size $(p \times n, p \times n)$. Its diagonal coefficients are equal to 1.
- $\widetilde{N}$, $\widetilde{B}_{radar}$: vectors of size $(p \times n, 1)$ whose coefficients are independent standard complex Gaussian random variables.

We have the following correspondence between the terms of the matrix Equation (G.3) and the equivalent vectorized Equation (G.4):

$$Z = \begin{bmatrix} \boxed{Z_0^T} \\ \boxed{Z_1^T} \\ \vdots \\ \boxed{Z_{p-1}^T} \end{bmatrix} \quad N = \begin{bmatrix} \boxed{N_0^T} \\ \boxed{N_1^T} \\ \vdots \\ \boxed{N_{p-1}^T} \end{bmatrix} \quad B_{radar} = \begin{bmatrix} \boxed{B_0^T} \\ \boxed{B_1^T} \\ \vdots \\ \boxed{B_{p-1}^T} \end{bmatrix} \tag{G.5}$$

$$\widetilde{Z} = \begin{bmatrix} Z_0 \\ \hline Z_1 \\ \hline \vdots \\ \hline Z_{p-1} \end{bmatrix} \quad \widetilde{N} = \begin{bmatrix} N_0 \\ \hline N_1 \\ \hline \vdots \\ \hline N_{p-1} \end{bmatrix} \quad \widetilde{B}_{radar} = \begin{bmatrix} B_0 \\ \hline B_1 \\ \hline \vdots \\ \hline B_{p-1} \end{bmatrix} \tag{G.6}$$

$$R_{st} = R_s \otimes R_t \tag{G.7}$$

$$\widetilde{T} = T \otimes I_n \tag{G.8}$$

We would like to represent the information contained in the matrix Z using its covariance matrix R_{st}: we do not use the clutter texture T to represent the radar data in this appendix.

To estimate the covariance matrix R_{st} from an observed matrix Z, we have to deal with the texture coefficient T and the radar noise B_{radar}. We generally assume that the diagonal coefficients of the texture matrix T are much larger than one. The radar noise B_{radar} can then be neglected: $Z \simeq T^{1/2} R_s^{1/2} N {R_t^T}^{1/2}$. The diagonal coefficients of the matrix T represents the expectation of the average quadratic power recorded in each cell. By setting t_i the i^{th} diagonal coefficient of the texture matrix T, we have $t_i = \mathbb{E}\left[|u_{i,j}|^2\right]$ for all $j \in [\![1, n]\!]$. We can estimate the coefficient t_i by averaging the quadratic power of the i^{th} row of the matrix Z presented in Equation (G.2):

$$\widehat{t_i} = \frac{1}{n} \sum_{j=1}^{n} |u_{i,j}|^2 , \tag{G.9}$$

this estimation being more precise when the number of pulses n is large. If we divide each row of matrix Z by the square root of its average quadratic power, i.e. by $\widehat{t_i}^{1/2}$, we obtain a matrix which is almost equal to $R_s^{1/2} N {R_t^T}^{1/2}$. We will present a method to estimate the Toeplitz HPD matrices R_s and R_t from a matrix of the form $R_s^{1/2} N {R_t^T}^{1/2}$ in Section G.4.

We now propose a simpler model without texture coefficient T and radar noise B_{radar}.

G.2.2 The simplified spatio-temporal model

We assume that the matrix Z presented in Equation (G.2) has the following spatio-temporal correlation:

$$Z = p_0^{1/2} R_s^{1/2} N {R_t^T}^{1/2} \tag{G.10}$$

with:

- Z: radar observation matrix of size (p, n).
- p_0: a positive real number which corresponds to the expectation of the quadratic power, i.e. $p_0 = \mathbb{E}\left[|u_{i,j}|^2\right]$ for all $(i, j) \in [\![1, p]\!] \times [\![1, n]\!]$.
- R_s: the scaled spatial autocorrelation matrix of size (p, p). It is a Toeplitz HPD matrix since the signal is assumed to be wide-sense stationary on the spatial axis. Its diagonal coefficients are equal to 1.
- R_t: the scaled temporal autocorrelation matrix of size (n, n). It is a Toeplitz HPD matrix since the signal is assumed to be wide-sense stationary on the temporal axis. Its diagonal coefficients are equal to 1.
- N: matrix of size (p, n) whose coefficients are independent standard complex Gaussian random variables.

We now present an equivalent vectorized model:

$$\widetilde{Z} = p_0^{1/2} R_{st}^{1/2} \widetilde{N} \tag{G.11}$$

with:

- $\widetilde{Z}$: radar observation matrix of size $(p \times n, 1)$.
- p_0: a positive real number which corresponds to the mean quadratic power.
- R_{st}: the scaled spatio-temporal autocorrelation matrix of size $(p \times n, p \times n)$. Its diagonal coefficients are equal to 1.
- $\widetilde{N}$: vector of size $(p \times n, 1)$ whose coefficients are independent standard complex Gaussian random variables.

We have the following correspondence between the terms of the matrix Equation (G.10) and the equivalent vectorized Equation (G.11):

$$Z = \begin{bmatrix} Z_0^T \\ \hline Z_1^T \\ \hline \vdots \\ \hline Z_{p-1}^T \end{bmatrix} \quad N = \begin{bmatrix} N_0^T \\ \hline N_1^T \\ \hline \vdots \\ \hline N_{p-1}^T \end{bmatrix} \tag{G.12}$$

$$\widetilde{Z} = \begin{bmatrix} Z_0 \\ \hline Z_1 \\ \hline \vdots \\ \hline Z_{p-1} \end{bmatrix} \quad \widetilde{N} = \begin{bmatrix} N_0 \\ \hline N_1 \\ \hline \vdots \\ \hline N_{p-1} \end{bmatrix} \tag{G.13}$$

$$R_{st} = R_s \otimes R_t \tag{G.14}$$

The model presented in Section G.2.2 looks like to model presented by Romain Couillet et al. [24], except for the structure of the spatial covariance matrix R_s which is only assumed to be Toeplitz here. We will consider this simplified spatio-temporal model in next sections.

The covariance matrix $p_0 R_{st}$ is the Kronecker product of two Toeplitz HPD matrices:

$$p_0 R_{st} = p_0 \, R_s \otimes R_t = R_s \otimes (p_0 R_t) \,, \tag{G.15}$$

it is therefore a Toeplitz-Block Block-Toeplitz HPD matrix with a very specific structure. We now study the structure of the associated autoregressive coefficients A_i^i and Siegel coefficients Ω_i.

G.3 The autoregressive coefficients and the Siegel coefficients

In Section 2.2, we have seen that a multidimensional stationary centered complex Gaussian autoregressive time series can be represented by its block-Toeplitz covariance matrix $\mathbf{R} \in \mathcal{B}_{p,n}^+$, by the coefficients $\left(P_0, A_1^1, \ldots, A_{p-1}^{p-1}\right) \in \mathcal{H}_n^+ \times \mathcal{D}_n^{p-1}$ where the coefficients A_i^i are autoregressive coefficients or by the coefficients $(P_0, \Omega_1, \ldots, \Omega_{p-1}) \in \mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$ where the coefficients Ω_i are Siegel coefficients. Using the relations given in Section 2.2 between these three representation spaces, we study here the structure of the autoregressive coefficients A_i^i and the Siegel coefficients Ω_i associated with the spatio-temporal correlation matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$.

In Section 2.2.5, a relation between the autocorrelation coefficients $(R_i)_{0 \leqslant i \leqslant M}$ and the autoregressive coefficients $\left(A_i^M\right)_{1 \leqslant i \leqslant M}$ has been given by the multidimensional Yule-Walker equation (2.123):

$$\widetilde{A}_M \, \widetilde{R}_M = -\widetilde{V}_M \quad \forall M \in [\![1, p-1]\!] \tag{G.16}$$

with:

$$\begin{aligned}
\widetilde{A}_M &= [A_1^M, \dots, A_M^M] \\
\widetilde{V}_M &= [R_1, \dots, R_M] \\
\widetilde{R}_M &= \begin{bmatrix} R_0 & R_1 & R_2 & \dots & R_{M-1} \\ R_1^H & R_0 & R_1 & \dots & R_{M-2} \\ R_2^H & R_1^H & R_0 & \dots & R_{M-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{M-1}^H & R_{M-2}^H & R_{M-3}^H & \dots & R_0 \end{bmatrix}
\end{aligned} \tag{G.17}$$

Hence, the autoregressive coefficients can be computed from the autocorrelation coefficients using Equation (2.126):

$$\widetilde{A}_M = -\widetilde{V}_M \widetilde{R}_M^{-1}. \tag{G.18}$$

The autoregressive coefficients associated with the covariance matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$ have a specific structure.

Property 59. *The spatial autoregressive coefficients $A_{sl}^{\ l}$ can be expressed from the one-dimensional spatial reflection coefficient μ_{sl}:*

$$A_{sl}^{\ l} = \mu_{sl} I_n \quad \forall l \in [\![1, p-1]\!]. \tag{G.19}$$

We now give two proofs of this property. The first proof is based on the multidimensional Yule-Walker equation, the second proof uses the multidimensional Levinson algorithm.

Proof. We start by setting:

$$R_s := \begin{bmatrix} r_{s0} & r_{s1}^* & r_{s2}^* & \dots & r_{sM-1}^* \\ r_{s1} & r_{s0} & r_{s1}^* & \dots & r_{sM-2}^* \\ r_{s2} & r_{s1} & r_{s0} & \dots & r_{sM-3}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{sM-1} & r_{sM-2} & r_{sM-3} & \dots & r_{s0} \end{bmatrix}. \tag{G.20}$$

Using this notation, we have:

$$p_0 R_{st} = R_s \otimes (p_0 R_t) = \begin{bmatrix} r_{s0} p_0 R_t & r_{s1}^* p_0 R_t & r_{s2}^* p_0 R_t & \dots & r_{sM-1}^* p_0 R_t \\ r_{s1} p_0 R_t & r_{s0} p_0 R_t & r_{s1}^* p_0 R_t & \dots & r_{sM-2}^* p_0 R_t \\ r_{s2} p_0 R_t & r_{s1} p_0 R_t & r_{s,0} p_0 R_t & \dots & r_{sM-3}^* p_0 R_t \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{sM-1} p_0 R_t & r_{sM-2} p_0 R_t & r_{sM-3} p_0 R_t & \dots & r_{s0} p_0 R_t \end{bmatrix}. \tag{G.21}$$

Therefore, we have:

$$R_{st} = \begin{bmatrix} R_0 & R_1^H & R_2^H & \dots & R_{M-1}^H \\ R_1 & R_0 & R_1^H & \dots & R_{M-2}^H \\ R_2 & R_1 & R_0 & \dots & R_{M-3}^H \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{M-1} & R_{M-2} & R_{M-3} & \dots & R_0 \end{bmatrix} \quad \text{with } R_i = r_{si} p_0 R_t. \tag{G.22}$$

By considering that the terms $\widetilde{R}_M$ and $\widetilde{V}_M$ of Equation (G.18) have correlation coefficients defined by $R_i = r_{si} p_0 R_t$, we note that they are Kronecker products:

$$\widetilde{R}_M = R_s^T \otimes (p_0 R_t), \tag{G.23}$$

$$\widetilde{V}_M = [r_{s1}, r_{s2}, \dots, r_{sM}] \otimes (p_0 R_t). \tag{G.24}$$

We recall the following properties of the Kronecker product:

$$\begin{aligned}
&(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}, \\
&(A \otimes B)(C \otimes D) = (AC) \otimes (BD).
\end{aligned}$$

Coming back to Equation (2.126), we obtain:

$$\widetilde{A}_M = -\widetilde{V}_M \widetilde{R}_M^{-1} \tag{G.25}$$
$$= -\left([r_{s1}, r_{s2}, \dots, r_{sM}] \otimes (p_0 R_t)\right) \left(R_s^T \otimes (p_0 R_t)\right)^{-1} \tag{G.26}$$
$$= -\left([r_{s1}, r_{s2}, \dots, r_{sM}] \otimes (p_0 R_t)\right) \left(R_s^{T^{-1}} \otimes (p_0 R_t)^{-1}\right) \tag{G.27}$$
$$= -\left([r_{s1}, r_{s2}, \dots, r_{sM}] R_s^{T^{-1}}\right) \otimes \left((p_0 R_t)(p_0 R_t)^{-1}\right) \tag{G.28}$$
$$= \left[a_{s1}^M, a_{s2}^M, \dots, a_{sM}^M\right] \otimes I_n \tag{G.29}$$

where the coefficients $\left[a_{s1}^M, a_{s2}^M, \dots, a_{sM}^M\right]$ are the autoregressive coefficients obtained resolving the one-dimensional Yule-Walker equation (see Equation (2.38)).

Therefore, the spatial autoregressive coefficient A_{sM}^M can be expressed from the one-dimensional spatial reflection coefficient μ_{sM}:

$$A_{sM}^M = \mu_{sM} I_n \quad \forall M \in [\![1, p-1]\!]. \tag{G.30}$$

□

As the matrix $p_0 R_{st}$ is a TBBT HPD matrix, another proof of Property 59 can be obtained using the multidimensional Levinson algorithm 1.

Proof. We can check that the multidimensional Levinson algorithm 1 with the input matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$ and the one-dimensional Levinson algorithm 13 with the input matrix R_s are related by the following relations:

- $P_0 = p_0 R_t$,
- $A_1^1 = a_1^1 I_n$.

  We can prove by induction that, for all $l \in [\![2, p-1]\!]$,
- $A_l^l = a_l^l I_n$,
- $\Delta_l = \delta_l R_t$ with $\delta_l = r_l + \sum_{j=1}^{l-1} r_{l-j} a_j^{l-1}$,
- $P_{l-1} = p_{l-1} R_t$ with $p_{l-1} = r_0 + \sum_{j=1}^{l-1} r_j a_j^{l-1*}$,
- $\widetilde{A}_l = \widetilde{a}_l \otimes I_n$ with $\widetilde{a}_l = \left[a_1^l, \dots, a_l^l\right]$.

□

The Siegel coefficients associated with the covariance matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$ also have a specific structure.

Property 60. *The spatial Siegel coefficients Ω_{sl} can be expressed from the one-dimensional spatial reflection coefficient μ_{sl}:*

$$\Omega_{sl} = A_{sl}^l = \mu_{sl} I_n \quad \forall l \in [\![1, p-1]\!]. \tag{G.31}$$

Proof. We can compute the coefficients $(P_0, \Omega_1, \dots, \Omega_{p-1})$ from a Toeplitz-Block Block-Toeplitz HPD matrix using Algorithm 2. We can check that Algorithm 2 on the input matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$ is related to the one-dimensional Levinson algorithm 13 on the input matrix R_s by the following relations:

- $P_0 = p_0 R_t$,
- $A_1^1 = a_1^1 I_n$,
- $\Omega_1 = a_1^1 I_n$.

  We can prove by induction that, for all $l \in [\![2, p-1]\!]$,
- $A_l^l = a_l^l I_n$,
- $\Delta_l = \delta_l R_t$ with $\delta_l = r_l + \sum_{j=1}^{l-1} r_{l-j} a_j^{l-1}$,
- $P_{l-1} = p_{l-1} R_t$ with $p_{l-1} = r_0 + \sum_{j=1}^{l-1} r_j a_j^{l-1*}$ ($\overline{P_{l-1}} = P_{l-1}$ since $p_{l-1} \in \mathbb{R}$ and R_t is a Toeplitz HPD matrix),
- $\widetilde{A}_l = \widetilde{a}_l \otimes I_n$ with $\widetilde{a}_l = \left[a_1^l, \dots, a_l^l\right]$,
- $\Omega_l = a_l^l I_n$.

□

The covariance matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$ can therefore be represented by the coefficients:

$$(P_0, \Omega_1, \dots, \Omega_{p-1}) = \left(p_0 R_t, \mu_{s1} I_n, \dots, \mu_{sp-1} I_n\right). \tag{G.32}$$

Note that the Toeplitz HPD matrix $p_0 R_t$ can be represented by the coefficients $\left(p_0, \mu_{t1}, \dots, \mu_{tn-1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$. The bijection is given by the Levinson algorithm 13. The spatio-temporal correlation matrix $p_0 R_{st} = R_s \otimes (p_0 R_t)$ can therefore be represented by the reflection coefficients $\left(p_0, \mu_{t1}, \dots, \mu_{tn-1}, \mu_{s1}, \dots, \mu_{sp-1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{n+p-2}$. Note that we could have seen it directly since the scaled temporal correlation matrix R_t is in bijection with the temporal reflection coefficients $\left(\mu_{t1}, \dots, \mu_{tn-1}\right)$ via the Levinson algorithm 13 and the scaled spatial correlation matrix R_s is in bijection with the spatial reflection coefficients $\left(\mu_{s1}, \dots, \mu_{sp-1}\right)$ via the Levinson algorithm as well. However, the study of the transformation of the scaled spatio-temporal correlation matrix $p_0 R_{st}$ into the reflection coefficients $\left(p_0, \mu_{t1}, \dots, \mu_{tn-1}, \mu_{s1}, \dots, \mu_{sp-1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{n+p-2}$ via the coefficients $\left(p_0 R_t, \mu_{s1} I_n, \dots, \mu_{sp-1} I_n\right) \in \mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$ presented in this section is useful: the metric of the space $\mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$ presented in Section 4.2.3 will be used to endow the space $\mathbb{R}_+^* \times \mathcal{D}^{n+p-2}$ with a Riemannian metric in Section G.5. As we assumed the time series to be wide-sense stationary on both spatial and temporal axes, we expect this metric to have the same structure on both axes.

In next section, we present a method to estimate the spatio-temporal covariance matrix $p_0 R_{st}$ or equivalently the coefficients $\left(p_0, \mu_{t1}, \dots, \mu_{tn-1}, \mu_{s1}, \dots, \mu_{sp-1}\right)$ from an observed matrix $Z = p_0^{1/2} R_s^{1/2} N {R_t^T}^{1/2}$ (see Section G.2.2).

G.4 The model parameters estimation

We consider an observation matrix Z corresponding to the model described in Section G.2.2:

$$Z = p_0^{1/2} R_s^{1/2} N {R_t^T}^{1/2}. \tag{G.33}$$

The parameters of this model are the mean quadratic power coefficient $p_0 \in \mathbb{R}_+^*$, the temporal correlation matrix R_t and the spatial correlation matrix R_s which are Toeplitz HPD matrices with diagonal coefficients equal to one. According to the previous section, this model can also be represented by the coefficients $\left(p_0, \mu_{t1}, \dots, \mu_{tn-1}, \mu_{s1}, \dots, \mu_{sp-1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{n+p-2}$ where the coefficients μ_t are the temporal reflection coefficients and the coefficients μ_s are the spatial reflection coefficients.

We can estimate the mean quadratic power coefficient p_0 by:

$$\widehat{p_0} = \frac{1}{np} \sum_{i=1}^{p} \sum_{j=1}^{n} |u_{i,j}|^2. \tag{G.34}$$

We can estimate the temporal reflection coefficients using the regularized Burg algorithm 17 on the rows of the matrix Z defined in Equation (G.2). We first estimate the temporal reflection coefficients $\left(\widehat{\mu}_{t1}^{\,i}, \widehat{\mu}_{t2}^{\,i}, \dots, \widehat{\mu}_{tn-1}^{\,i}\right)$ of each cell $i \in [\![1, p]\!]$. Then we can estimate the mean temporal reflection coefficients $\left(\widehat{\mu}_{t1}^{\,mean}, \widehat{\mu}_{t2}^{\,mean}, \dots, \widehat{\mu}_{tn-1}^{\,mean}\right)$ on the manifold $\mathbb{D}^{n-1}$ using Algorithm 8 which approximates the mean by gradient descent. The manifold $\mathbb{D}^{n-1}$ corresponds to the space $\mathcal{D}^{n-1}$ endowed with the metric described in Section 4.1.3 omitting the power coefficient p_0.

We can estimate the spatial reflection coefficients using the same method but performing the regularized Burg algorithm on the columns of the matrix Z instead of its rows.

We can then estimate the scaled temporal correlation matrix $\widehat{R}_t$ using the reversed Levinson Algorithm 15 on the input coefficients $\left(1, \widehat{\mu}_{t1}^{\,mean}, \widehat{\mu}_{t2}^{\,mean}, \dots, \widehat{\mu}_{tn-1}^{\,mean}\right)$. Indeed, we consider that the mean quadratic power coefficient p_0^{mean} is equal to one to obtain a scaled covariance matrix whose diagonal coefficients are equal to one.

Similarly, we can estimate the scaled spatial correlation matrix $\widehat{R}_s$ using the reversed Levinson Algorithm on the input coefficients $\left(1, \widehat{\mu_{s1}}^{mean}, \widehat{\mu_{s2}}^{mean}, \dots, \widehat{\mu_{sp-1}}^{mean}\right)$.

We can then estimate the scaled spatio-temporal correlation matrix using formula (G.14): it is the Kronecker product of the scaled spatial correlation matrix and the scaled temporal correlation matrix: $\widehat{R}_{st} = \widehat{R}_s \otimes \widehat{R}_t$.

An alternative estimation method of the spatial and temporal correlation matrices is presented in [24] which does not use any Riemannian tools.

In next section, we endow the space $\mathbb{R}_+^* \times \mathcal{D}^{n+p-2}$ with a Riemannian metric inspired by information geometry.

G.5 The Riemannian manifold associated with the spatio-temporal model

In Section 2.2, we have seen that a multidimensional stationary centered complex Gaussian autoregressive time series can be represented by its block-Toeplitz HPD matrix $\mathbf{R} \in \mathcal{B}_{p,n}^+$ or by the coefficients $(P_0, \Omega_1, \dots, \Omega_{p-1}) \in \mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$. In Section 3.2.3, we have endowed the space $\mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$ with a Riemannian metric inspired by information geometry.

The geometric tools of this metric are detailed in Section 4.2.3. We obtained a manifold denoted $\mathbb{H}_n^{++} \times \mathbb{SD}_n^{p-1}$ whose metric is given by:

$$ds^2_{\mathbb{H}_n^{++}\times\mathbb{SD}_n^{p-1}} = p\, ds^2_{\mathbb{H}_n^{++}} + \sum_{l=1}^{p-1} (p-l)\, ds^2_{\mathbb{SD}_n} \tag{G.35}$$

with:

$$ds^2_{\mathbb{H}_n^{++}} = \operatorname{trace}\left(P_0^{-1}\, dP_0\, P_0^{-1}\, dP_0\right) \tag{G.36}$$

$$ds^2_{\mathbb{SD}_n} = \operatorname{trace}\left(\left(I-\Omega_l\Omega_l^H\right)^{-1} d\Omega_l \left(I-\Omega_l^H\Omega_l\right)^{-1} d\Omega_l^H\right). \tag{G.37}$$

According to Section G.3, the spatio-temporal correlation matrix $R_{st} = R_s \otimes R_t$ is associated with the coefficients $\left(R_t, \mu_{s1} I_n, \ldots, \mu_{sp-1} I_n\right) \in \mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$. If we take into account the mean quadratic power coefficient $p_0 \in \mathbb{R}_+^*$, then the matrix $p_0 R_{st}$ is associated with the coefficients $\left(p_0 R_t, \mu_{s1} I_n, \ldots, \mu_{sp-1} I_n\right) \in \mathcal{H}_n^+ \times \mathcal{SD}_n^{p-1}$.

The metric described in Equation (G.35) represents the matrix $p_0 R_t$ into the manifold $\mathbb{H}_n^{++}$ up to multiplication of the metric by a constant. However, this manifold does not take into account the Toeplitz structure of the matrix $p_0 R_t$. Since the matrix $p_0 R_t$ is a Toeplitz HPD matrix, we could instead represent it by the coefficients $\left(p_0, \mu_{t1}, \ldots, \mu_{tn-1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{n-1}$ and use the metric presented in Section 4.1.3:

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n ds^2_{\mathbb{R}^{++}} + \sum_{k=1}^{n-1} (n-k)\, ds^2_{\mathbb{D}} \tag{G.38}$$

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n-1}} = n\frac{dp_0^2}{p_0^2} + \sum_{k=1}^{n-1} (n-k)\, \frac{|d\mu_{t_k}|^2}{\left(1-|\mu_{t_k}|^2\right)^2}. \tag{G.39}$$

Note that the metric $ds^2_{\mathbb{SD}_n}$ described in Equation (G.37) is equal to the Poincaré metric up to multiplication by a constant for the diagonal matrices $\mu_s I_n$:

$$ds^2_{\mathbb{SD}_n} = n ds^2_{\mathbb{D}} \quad \text{with } ds^2_{\mathbb{D}} = \frac{|d\mu_s|^2}{\left(1-|\mu_s|^2\right)^2}. \tag{G.40}$$

Using Equations (G.38) and (G.40), we can transform the metric described in Equation (G.35):

$$ds^2_{\mathbb{H}_n^{++}\times\mathbb{SD}_n^{p-1}} = p\, ds^2_{\mathbb{H}_n^{++}} + \sum_{l=1}^{p-1} (p-l)\, ds^2_{\mathbb{SD}_n}$$

into the metric:

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n+p-2}} = p\left(n\, ds^2_{\mathbb{R}^{++}} + \sum_{k=1}^{n-1} (n-k)\, ds^2_{\mathbb{D}}\right) + \sum_{l=1}^{p-1} (p-l)\left(n\, ds^2_{\mathbb{D}}\right) \tag{G.41}$$

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{n+p-2}} = np\, ds^2_{\mathbb{R}^{++}} + p\sum_{k=1}^{n-1} (n-k)\, ds^2_{\mathbb{D}} + n\sum_{l=1}^{p-1} (p-l)\; ds^2_{\mathbb{D}} \tag{G.42}$$

with

$$ds^2_{\mathbb{R}^{++}} = \frac{dp_0^2}{p_0^2}, \quad \text{and} \quad ds^2_{\mathbb{D}} = \frac{|d\mu|^2}{\left(1-|\mu|^2\right)^2}. \tag{G.43}$$

We propose to use this metric on the coefficients $\left(p_0, \mu_{t1}, \ldots, \mu_{tn-1}, \mu_{s1}, \ldots, \mu_{sp-1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{n+p-2}$ to represent the spatio-temporal information contained in the matrix $Z = p_0^{1/2} R_s^{1/2} N R_t^{T\,1/2}$ described in Section G.2.2. Note that this metric has similar structure on both the temporal and the spatial axes, which is expected since we assumed the time series to be wide-sense stationary on both axes (the temporal correlation matrix R_t and the spatial correlation matrix R_s are both scaled Toeplitz HPD matrices).

9.6 Generalization to higher dimensional models

In this section, we generalize the spatio-temporal model to models with possibly more than two dimensions. Let's consider the observation matrix Z of dimension n and shape $(d_1, d_2, \dots, d_n)$. We assume that each axis i of the matrix Z is associated with the correlation matrix $p_0 R_i$ where p_0 is the mean quadratic power coefficient:

$$p_0 = \mathbb{E}\left[|z_{i_1,i_2,\dots,i_n}|^2\right] \quad \forall\, (i_1, i_2, \dots, i_n) \in [\![1, d_1]\!] \times [\![1, d_2]\!] \times \dots \times [\![1, d_n]\!]$$

and R_i is a scaled Toeplitz HPD matrix (its diagonal coefficients are equal to one). We define $\widetilde{Z}$ as the vectorization of the matrix Z, it is a large vector a size $\prod_{i=1}^n d_i$ whose scaled correlation matrix is $R_{d_1 d_2 \dots d_n} = R_{d_1} \otimes R_{d_2} \otimes \dots \otimes R_{d_n}$. This model can be parametrized by the coefficients $(p_0, R_1, \dots, R_n)$. Each scaled Toeplitz HPD matrix R_i can also be represented by the reflection coefficients $\left(\mu_{i1}, \dots, \mu_{i d_i - 1}\right)$ via the Levinson algorithm 13. This model can therefore be parametrized by the coefficients $\left(p_0, \mu_{11}, \dots, \mu_{1 d_1 - 1}, \mu_{21}, \dots, \mu_{2 d_2 - 1}, \dots, \mu_{n1}, \dots, \mu_{n d_n - 1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{(\prod_{i=1}^n d_i) - n}$.

We propose to use the following Riemannian metric on the space $\mathbb{R}_+^* \times \mathcal{D}^{(\prod_{i=1}^n d_i) - n}$:

$$ds^2_{\mathbb{R}^{++} \times \mathbb{D}^{\left(\prod_{i=1}^n d_i\right) - n}} = \left(\prod_{i=1}^n d_i\right) \left(ds^2_{\mathbb{R}^{++}} + \sum_{i=1}^n \frac{1}{d_i} \sum_{k=1}^{d_i - 1} (d_i - k)\, ds^2_{\mathbb{D}}\right) \tag{9.44}$$

with

$$ds^2_{\mathbb{R}^{++}} = \frac{dp_0^2}{p_0^2}, \quad \text{and} \quad ds^2_{\mathbb{D}} = \frac{|d\mu|^2}{\left(1 - |\mu|^2\right)^2}.$$

This metric can be constructed by induction. It has been constructed for $n = 1$ in Section 3.2.2. It has been constructed for $n = 2$ for the spatio-temporal model in Section G.5. To construct the metric of the model of dimension $n+1$ from the metric of the model of dimension n, we can note that Algorithm 2 transforms the Toeplitz-Block Block-Toeplitz HPD matrix $p_0 R_{d_1 \dots d_n d_{n+1}}$ into the coefficients $\left(p_0 R_{d_1 \dots d_n}, \mu_{n+1\,1} I_{\prod_{i=1}^n d_i}, \dots, \mu_{n+1\,d_{n+1}-1} I_{\prod_{i=1}^n d_i}\right) \in \mathcal{H}^+_{\prod_{i=1}^n d_i} \times \mathcal{SD}^{d_{n+1}-1}_{\prod_{i=1}^n d_i}$. In Section 3.2.3, we have endowed the space $\mathcal{H}^+_{\prod_{i=1}^n d_i} \times \mathcal{SD}^{d_{n+1}-1}_{\prod_{i=1}^n d_i}$ with a Riemannian metric inspired by information geometry. We obtained a manifold denoted $\mathbb{H}^{++}_{\prod_{i=1}^n d_i} \times \mathbb{SD}^{d_{n+1}-1}_{\prod_{i=1}^n d_i}$ whose metric is given by:

$$ds^2_{\mathbb{H}^{++}_{\prod_{i=1}^n d_i} \times \mathbb{SD}^{d_{n+1}-1}_{\prod_{i=1}^n d_i}} = d_{n+1}\, ds^2_{\mathbb{H}^{++}_{\prod_{i=1}^n d_i}} + \sum_{l=1}^{d_{n+1}-1} (d_{n+1} - l)\, ds^2_{\mathbb{SD}_{\prod_{i=1}^n d_i}} \tag{9.45}$$

with:

$$ds^2_{\mathbb{H}^{++}_{\prod_{i=1}^n d_i}} = \operatorname{trace}\left(P_0^{-1}\, dP_0\, P_0^{-1}\, dP_0\right) \tag{9.46}$$

$$ds^2_{\mathbb{SD}_{\prod_{i=1}^n d_i}} = \operatorname{trace}\left(\left(I - \Omega_l \Omega_l^H\right)^{-1} d\Omega_l \left(I - \Omega_l^H \Omega_l\right)^{-1} d\Omega_l^H\right). \tag{9.47}$$

Note that the metric $ds^2_{\mathbb{SD}_{\prod_{i=1}^n d_i}}$ described in Equation (9.47) is equal to the Poincaré metric up to multiplication by a constant for the diagonal matrices $\mu_{n+1} I_{\prod_{i=1}^n d_i}$:

$$ds^2_{\mathbb{SD}_{\prod_{i=1}^n d_i}} = \left(\prod_{i=1}^n d_i\right) ds^2_{\mathbb{D}} \quad \text{with } ds^2_{\mathbb{D}} = \frac{|d\mu_{n+1}|^2}{\left(1 - |\mu_{n+1}|^2\right)^2}. \tag{9.48}$$

Using Equations (9.45) and (9.48), the coefficients $\left(p_0 R_{d_1 \dots d_n}, \mu_{n+1\,1}, \dots, \mu_{n+1\,d_{n+1}-1}\right) \in \mathcal{H}^+_{\prod_{i=1}^n d_i} \times \mathcal{D}^{d_{n+1}-1}$ are endowed with the following metric:

$$ds^2_{\mathbb{H}^{++}_{\prod_{i=1}^n d_i} \times \mathbb{D}^{d_{n+1}-1}} = d_{n+1}\, ds^2_{\mathbb{H}^{++}_{\prod_{i=1}^n d_i}} + \left(\prod_{i=1}^n d_i\right) \sum_{l=1}^{d_{n+1}-1} (d_{n+1} - l)\, ds^2_{\mathbb{D}} \tag{9.49}$$

Instead of using the metric $ds^2_{\mathbb{H}^{++}_{\prod_{i=1}^n d_i}}$ on the coefficient $p_0 R_{d_1 \dots d_n}$, we choose to use on the coefficients $\left(p_0, \mu_{11}, \dots, \mu_{1 d_1 - 1}, \mu_{21}, \dots, \mu_{2 d_2 - 1}, \dots, \mu_{n1}, \dots, \mu_{n d_n - 1}\right) \in \mathbb{R}_+^* \times \mathcal{D}^{(\prod_{i=1}^n d_i) - n}$ the metric proposed for the multidimensional model of order n in Equation (9.44):

$$ds^2_{\mathbb{R}^{++} \times \mathbb{D}^{\left(\prod_{i=1}^n d_i\right) - n}} = \left(\prod_{i=1}^n d_i\right) \left(ds^2_{\mathbb{R}^{++}} + \sum_{i=1}^n \frac{1}{d_i} \sum_{k=1}^{d_i - 1} (d_i - k)\, ds^2_{\mathbb{D}}\right). \tag{9.50}$$

Therefore, we obtain:

$$ds^2_{\mathbb{R}^{++}\times\mathbb{D}^{\left(\prod_{i=1}^{n+1} d_i\right)-(n+1)}} = \left(\prod_{i=1}^{n+1} d_i\right)\left(ds^2_{\mathbb{R}^{++}} + \sum_{i=1}^{n+1}\frac{1}{d_i}\sum_{k=1}^{d_i-1}(d_i-k)\,ds^2_{\mathbb{D}}\right) \tag{9.51}$$

which is the metric proposed in Equation (9.44) for the model of order $n+1$. This highlights that the metric proposed can be constructed by induction using the metric presented in Section 3.2.3.

The radar spatio-temporal model presented in this appendix considers two axes: the temporal axis whose dimension is related to the number of pulses of the burst and the spatial axis whose dimension is equal to the number of consecutives cells considered along the radial axis. In Chapter 8, the radar clutter is studied using only the zero elevation beam. However, if the space studied is a volume (clouds, rain, hail, snow...) then we could also use the elevation axis. We would have a three dimensional observation matrix with a temporal axis and two spatial axes: a radial axis and an elevation axis. We could then use the multidimensional model presented here with $n = 3$. If the assumption of stationarity is meaningful on each axis, we could then use the multidimensional model presented here with $n = 3$ to represent the volume studied in a Riemannian manifold.

Bibliography

[1] A. Sinap and W. Van Assche. Orthogonal matrix polynomials and applications. *Journal of Computational and Applied Mathematics, 66*, pages 27–52, 1996.

[2] Hirotugu Akaike. Maximum likelihood identification of Gaussian autoregressive moving average models. *Biometrika 60*, pages 255–265, 1973.

[3] Alexis Decurninge, Frédéric Barbaresco. Robust Burg Estimation of Radar Scatter Matrix for Mixtures of Gaussian Stationary Autoregressive Vectors. *IET Radar, Sonar & Navigation, Volume 11, Issue 1*, page 78–89, January 2016.

[4] Davide Barilari Andrei A. Agrachev and Luca Rizzi. Curvature: A Variational Approach. *Memoirs of the American Mathematical Society*, 2018.

[5] Bernard Armour and Salvatore D. Morgera. An exact forward-backward maximum likelihood autoregressive parameter estimation method. *IEEE*, September 1991.

[6] Marc Arnaudon, Frédéric Barbaresco, and Le Yang. Riemannian Medians and Means With Applications to Radar Signal Processing. *IEEE*, August 2013.

[7] Robert Azencott and Didier Dacunha-Castelle. *Séries d'observations irrégulières - Modélisation et prévision*. Masson, 1997.

[8] Frédéric Barbaresco. Algorithme de Burg regularisé FSDS (fonctionnelle stabilisatrice de douceur spectrale) Comparaison avec l'algorithme de Burg MFE (Minimum free energy). September 1995.

[9] Frédéric Barbaresco. Super resolution spectrum analysis regularization: Burg, Capon and AGO-antagonistic algorithms. pages 2005–2008, 1996.

[10] Frédéric Barbaresco. Information Geometry of Covariance Matrix: Cartan-Siegel Homogeneous Bounded Domains, Mostow/Berger Fibration and Fréchet Median. *Matrix Information Geometry; Springer*, page 199–256, 2012.

[11] Frédéric Barbaresco. Information Geometry Manifold of Toeplitz Hermitian Positive Definite Covariance Matrices: Mostow/Berger Fibration and Berezin Quantization of Cartan-Siegel Domains. March 2013.

[12] Frédéric Barbaresco, Thibault Forget, Emmanuel Chevallier, and Jesus Angulo. Doppler spectrum segmentation of radar sea clutter by mean-shift and information geometry metric. 2017.

[13] J. Barrie Billingsley. Low-Angle Radar Land Clutter, Measurements and Empirical Models. *William Andrew Publishing*, 2002.

[14] Jean Bricmont. *Making Sense of Statistical Mechanics*. Springer, 2022.

[15] Alice Le Brigant. *Probability on the spaces of curves and the associated metric spaces using information geometry; radar applications*. PhD thesis, 2017.

[16] John Parker Burg. *Maximum Entropy Spectral Analysis*. PhD thesis, Stanford University, 1975.

[17] Yann Cabanes, Frédéric Barbaresco, Marc Arnaudon, and Jérémie Bigot. Non-Supervised High Resolution Doppler Machine Learning for Pathological Radar Clutter. *IEEE, RADAR 2019, Toulon, France*, September 2019.

[18] Yann Cabanes, Frédéric Barbaresco, Marc Arnaudon, and Jérémie Bigot. Non-supervised Machine Learning Algorithms for Radar Clutter High-Resolution Doppler Segmentation and Pathological Clutter Analysis. *IEEE, IRS 2019, Ulm, Germany*, June 2019.

[19] Yann Cabanes, Frédéric Barbaresco, Marc Arnaudon, and Jérémie Bigot. Toeplitz Hermitian Positive Definite Matrix Machine Learning based on Fisher Metric. *IEEE, GSI 2019, Toulouse, France*, August 2019.

[20] Yann Cabanes, Frédéric Barbaresco, Marc Arnaudon, and Jérémie Bigot. Unsupervised Machine Learning for Pathological Radar Clutter Clustering: the P-Mean-Shift Algorithm. *IEEE, C&ESAR 2019, Rennes, France*, November 2019.

[21] Yann Cabanes and Frank Nielsen. Classification in the Siegel Space for Vectorial Autoregressive Data. *Geometric Science of Information*, 2021.

[22] Emmanuel Chevallier, Thibault Forget, Frédéric Barbaresco, and Jesus Angulo. Kernel Density Estimation on the Siegel Space with an Application to Radar Processing. *Entropy*, 2016.

[23] Christrian Chaire and Frédéric Barbaresco. New generation Doppler radar processing: Ultra-fast robust Doppler Spectrum Barycentre computation scheme in Poincaré's unit disk. *The 7th European Radar Conference*, pages 196–199, 2010.

[24] Romain Couillet, Maria Sabrina Greco, Jean-Philippe Ovarlez, and Frédéric Pascal. RMT for Whitening Space Correlation and Applications to Radar Detection. *IEEE*, 2015.

[25] D. Damanik, A. Pushnitski and B. Simon. The analytic theory of matrix orthogonal polynomials. *Surveys in Approximation Theory, 4*, pages 1–85, 2008.

[26] Alexis Decurninge and Frédéric Barbaresco. Robust Burg Estimation of Radar Scatter Matrix for Autoregressive structured SIRV based on Fréchet medians. *arXiv*, May 2016. https://arxiv.org/pdf/1601.02804.pdf.

[27] Holger Dette and William J. Studden. *The Theory of Canonical Moments with Applications in Statistics, Probability, and Analysis*. Wiley and Sons, 1997.

[28] Holger Dette and Jens Wagener. Matrix measures on the unit circle, moment spaces, orthogonal polynomials and the Geronimus relations. *Linear Algebra and its Applications, 432*, pages 1609–1626, 2010.

[29] Michel Marie Deza and Elena Deza. *Encyclopedia of Distances*. Springer, 2016.

[30] P. Thomas Fletcher, Conglin Lu, Stephen M. Pizer, and Sarang Joshi. Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape. *IEEE Transactions on Medical Imaging*, August 2004.

[31] François Rouvière. *Initiation à la géométrie de Riemann*. Calvage & Mounet, 2018.

[32] Bernd Fritzsche and Bernd Kirstein. *An extension problem for non-negative Hermitian block Toeplitz matrices*. Mathematische Nachrichten, 131, pp. 287-297, 1987.

[33] Frédéric Barbaresco. Modèles autorégressifs: du coefficient de réflexion à la géométrie Riemannienne de l'information. *Traitement du Signal*, 1998.

[34] Frédéric Barbaresco. Innovative tools for radar signal processing Based on Cartan's geometry of SPD matrices & Information Geometry. *IEEE Radar Conference*, 2008.

[35] Frédéric Barbaresco. Géométrie différentielle des matrices de covariance et espaces métriques à courbure négative. *GRETSI Conference*, 2009.

[36] Frédéric Barbaresco. Interactions between Symmetric Cone and Information Geometries: Bruhat-Tits and Siegel Spaces Models for High Resolution Autoregressive Doppler Imagery. *In: Nielsen F. (eds) Emerging Trends in Visual Computing. ETVC 2008. Lecture Notes in Computer Science, vol 5416. Springer, Berlin, Heidelberg*, 2009.

[37] Frédéric Barbaresco. New foundation of radar Doppler signal processing based on advanced differential geometry of symmetric spaces. *International Radar Conference*, 2009.

[38] Frédéric Barbaresco. Robust statistical Radar Processing in Fréchet metric space: OS-HDR-CFAR and OS-STAP Processing in Siegel homogeneous bounded domains. *12th International Radar Symposium (IRS)*, pages 639–644, 2011.

[39] Frédéric Barbaresco. Koszul Information Geometry and Souriau Geometric Temperature/Capacity of Lie Group Thermodynamics. *Entropy, 16(8):4521-4565*, 2014.

[40] Frédéric Barbaresco. Gaussian Distributions on the Space of Symmetric Positive Definite Matrices from Souriau's Gibbs State for Siegel Domains by Coadjoint Orbit and Moment Map. *Geometric Science of Information*, 2021.

[41] Frédéric Barbaresco and G. Bouyt. Espace Riemannien symétrique et géométrie des espaces de matrices de covariance: équations de diffusion et calculs de médianes. *GRETSI Conference*, 2009.

[42] Maria S. Greco and Fulvion Gini. Radar Clutter Modeling.

[43] Ulf Grenander and Gábor Szegő. *Toeplitz forms and their applications*. Second edition, Chelsea Publishing Co., New York, 1984.

[44] Simon Haykin. *Adaptive Filter Theory*. Pearson, 2014.

[45] Ben Jeuris and Raf Vandebril. The Kähler mean of Block-Toeplitz matrices with Toeplitz structured blocks. *SIAM Journal on Matrix Analysis and Applications 37(3):1151-1175*, 2016.

[46] Jérôme Lapuyade-Lahorgue and Frédéric Barbaresco. Radar detection using Siegel distance between autoregressive processes, application to HF and X-band radar. *IEEE Radar Conference*, 2008.

[47] S. Kullback and R. A. Leibler. On Information and Sufficiency. *Annals of Mathematical Statistics*, 1951.

[48] Jean-Michel Loubes and Bruno Pelletier. A kernel-based classifier on a Riemannian manifold. *Statistics & Decisions 26*, July 2008.

[49] M. Humet and M. Van Barel. Algorithms for the Geronimus transformation for orthogonal polynomials on the unit circle. *Journal of Computational and Applied Mathematics, 267*, page 195–217, 2014.

[50] John Makhoul. Linear Prediction: A Tutorial Review. *Proceedings of the IEEE, Vol. 63, No. 4*, April 1975.

[51] Daniel T. L. Lee Martin Morf, Augusto Vieira and Thomas Kailath. Recursive Multichannel Maximum Entropy Spectral Estimation. *IEEE Transactions on Geoscience Electronics, vol. GE-16, NO. 2*, April 1978.

[52] Nina Miolane, Nicolas Guigui, Alice Le Brigant, Johan Mathe, Benjamin Hou, Yann Thanwerdas, Stefan Heyder, Olivier Peltre, Niklas Koep, Hadi Zaatiti, Hatem Hajri, Yann Cabanes, Thomas Gerald, Paul Chauchat, Christian Shewmake, Daniel Brooks, Bernhard Kainz, Claire Donnat, Susan Holmes, and Xavier Pennec. Geomstats: A Python Package for Riemannian Geometry in Machine Learning. *Journal of Machine Learning Research*, 21(223):1–9, 2020. http://jmlr.org/papers/v21/19-027.html, https://github.com/geomstats/geomstats.

[53] Nicolas Charon and Frédéric Barbaresco. Une nouvelle approche pour la détection de cibles dans les images radar basée sur des distances et moyennes dans des espaces de matrices de covariance. *Traitement du Signal*, 2009.

[54] Frank Nielsen. Hilbert geometry of the Siegel disk: The Siegel-Klein disk model. 2020.

[55] P. Delsarte, Y.V. Genin and Y.G. Kamp. Orthogonal polynomial matrices on the unit circle. *IEEE Transactions on Circuits and Systems, 25*, pages 149–160, 1978.

[56] P. Delsarte, Y.V. Genin and Y.G. Kamp. Parametrization of positive definite block-Toeplitz systems. *SIAM Journal on Applied Mathematics, 36*, pages 34–46, 1979.

[57] F. Pascal, R. Forster, J. . Ovarlez, and P. Arzabal. Theoretical analysis of an improved covariance matrix estimator in non-gaussian noise. In *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 4, pages iv/69–iv/72 Vol. 4, 2005.

[58] Xavier Pennec. Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements. *Journal of Mathematical Imaging and Vision, Springer Verlag, 25 (1), pp.127-154*, 2006.

[59] Xavier Pennec. Barycentric Subspaces and Affine Spans in Manifolds. *Geometric Science of Information*, 2015.

[60] Bernard Picinbono. Second-order complex random vectors and normal distributions. *IEEE Transactions on Signal Processing, Volume: 44, Issue: 10*, pages 2637–2640, 1996.

[61] Pierre-Yves Lagrave and Frédéric Barbaresco. Generalized SU(1, 1) Equivariant Convolution on Fock-Bargmann Spaces for Robust Radar Doppler Signal Classification. *CAID*, 2021.

[62] Pierre-Yves Lagrave, Yann Cabanes and Frédéric Barbaresco. An Equivariant Neural Network with Hyperbolic Embedding for Robust Doppler Signal Classification. *IRS*, 2021.

[63] Pierre-Yves Lagrave, Yann Cabanes and Frédéric Barbaresco. SU(1, 1) Equivariant Neural Networks and Application to Robust Toeplitz Hermitian Positive Definite Matrix Classification. *Geometric Science of Information*, 2021.

[64] Riadh Ben Jelili, Salima Hamouche and Ferhat Mihoubi. Le pseudo-maximum de vraisemblance théorique et simulé. Application au cas des modèles de déséquilibre. *Économie & prévision*, (112):149–174, 1994. https://www.persee.fr/doc/ecop_0249-4744_1994_num_112_1_5659.

[65] Jorma Rissanen. Minimax codes for finite alphabets. *IEEE Transactions on Information Theory*, 24(3):389–392, 1978.

[66] Jorma Rissanen. Stochastic Complexity and Modeling. *The Annals of Statistics*, 14(3):1080–1100, 1986. https://doi.org/10.1214/aos/1176350051.

[67] Jorma Rissanen. *Stochastic Complexity in Statistical Inquiry*. World Scientific Publishing Co., Inc., Teaneck, NJ, 1989.

[68] R. Tyrrell Rockafellar. *Convex Analysis*. Princeton University Press, Second Printing, 1972.

[69] Peter J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. 1987.

[70] Salem Said. Statistical models and probabilistic methods on Riemannian manifolds. 2021. https://arxiv.org/abs/2101.10855.

[71] Salem Said, Simon Heuveline and Cyrus Mostajeran. Riemannian statistics meets random matrix theory: towards learning from high-dimensional covariance matrices. 2022. https://arxiv.org/abs/2203.00204.

[72] Satake Ichiro. Algebraic Structures of Symmetric Domains. *Princeton University Press*, 1981.

[73] Shun-ichi Amari. *Information Geometry and Its Applications*. Springer, 2016.

[74] Shun-ichi Amari and Andrzej Cichocki. Information Geometry of divergence functions. *Bulletin of the Polish Academy of Sciences: Technical Sciences, 58(1):183-195*, 2010.

[75] Carl Ludwig Siegel. Symplectic geometry. *American Journal of Mathematics*, 65(1):1–86, 1943. http://www.jstor.org/stable/2371774.

[76] Barry Simon. *Szegő's Theorem and Its Descendants - Spectral Theory for L^2 Perturbations of Orthogonal Polynomials*. Princeton University Press, 2010.

[77] G. Szegő. Orthogonal Polynomials. *American Mathematical Society*, 1939.

[78] Samuel Verblunsky. On positive harmonic functions: a contribution to the algebra of Fourier series. *Proceedings London Mathematical Society, 38*, pages 125–157, 1935.

[79] Samuel Verblunsky. On positive harmonic functions (second paper). *Proceedings London Mathematical Society, 40*, pages 290–320, 1936.

[80] Mati Wax and Thomas Kailath. Efficient Inversion of Toeplitz-Block Toeplitz Matrix. *IEEE*, 1983.

[81] Werner Ballmann. Lectures on Kähler manifolds. *ESI Lectures in Mathematics and Physics, European Mathematical Society, Zürich*, 2006.

[82] Le Yang. *Medians of probability measures in Riemannian manifolds and applications to radar target detection*. PhD thesis, 2011.

[83] Yann Cabanes, Frédéric Barbaresco, Marc Arnaudon and Jérémie Bigot. Matrix Extension for Pathological Radar Clutter Machine Learning. *hal-02875440*, 2020.