HAL
open science

ON THE USE OF GEODESIC TRIANGLES BETWEEN GAUSSIAN DISTRIBUTIONS FOR CLASSIFICATION PROBLEMS

Antoine Collas, Florent Bouchard, Guillaume Ginolhac, Arnaud Breloy, Chengfang Ren, Jean-Philippe Ovarlez

To cite this version:

Antoine Collas, Florent Bouchard, Guillaume Ginolhac, Arnaud Breloy, Chengfang Ren, et al.. ON THE USE OF GEODESIC TRIANGLES BETWEEN GAUSSIAN DISTRIBUTIONS FOR CLASSIFICATION PROBLEMS. 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2022), May 2022, Singapour, Singapore. 10.1109/ICASSP43922.2022.9747872 . hal-03752879

HAL Id: hal-03752879

https://hal.science/hal-03752879

Submitted on 17 Aug 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ON THE USE OF GEODESIC TRIANGLES BETWEEN GAUSSIAN DISTRIBUTIONS FOR CLASSIFICATION PROBLEMS

A. Collas[1], *F. Bouchard*[2], *G. Ginolhac*[3], *A. Breloy*[4], *C. Ren*[1], *J.-P. Ovarlez*[1,5]

[1]SONDRA, CentraleSupélec, Université Paris-Saclay
[2]CNRS, L2S, CentraleSupélec, Université Paris-Saclay
[3]LISTIC, Université Savoie Mont Blanc
[4]LEME, Université Paris Nanterre
[5]DEMR, ONERA, Université Paris-Saclay

ABSTRACT

This paper presents a new classification framework for both first and second order statistics, *i.e.* mean/location and covariance matrix. In the last decade, several covariance matrix classification algorithms have been proposed. They often leverage the Riemannian geometry of symmetric positive definite matrices (SPD) with its affine invariant metric and have shown strong performance in many applications. However, their underlying statistical model assumes a zero mean hypothesis. In practice, it is often estimated and then removed in a preprocessing step. This is of course damaging for applications where the mean is a discriminative feature. Unfortunately, the distance associated to the affine invariant metric for both mean and covariance matrix remains unknown. Leveraging previous works on geodesic triangles, we propose two affine invariant divergences that use both statistics. Then, we derive an algorithm to compute the associated Riemannian centers of mass. Finally, a divergence based *Nearest centroid*, applied on the crop classification dataset *Breizhcrops*, shows the interest of the proposed framework.

***Index Terms*—** Fisher information metric, divergence, Riemannian optimization, Riemannian center of mass

1. INTRODUCTION

Classically, many signal processing applications make use of the second order statistic. Indeed, when the multivariate signals are assumed to be Gaussian, the covariance matrix is an interesting feature to discriminate data.

Recently, the Riemannian geometry associated to the Fisher information metric (FIM) of the centered Gaussian distribution [1] has been used with great successes on classification problems, *e.g.* on EEG data [2], in detection of pedestrians [3] or in Diffusion tensor imaging [4]. The squared distance of the geodesic between two covariance matrices $\mathbf{\Sigma}_1, \mathbf{\Sigma}_2 \in \mathcal{S}_p^{++}$ (set of $p \times p$ PSD matrices) benefits from a simple closed form formula,

$$d^2_{\mathcal{S}_p^{++}}(\mathbf{\Sigma}_1, \mathbf{\Sigma}_2) = \left\| \log\left(\mathbf{\Sigma}_1^{-\frac{1}{2}} \mathbf{\Sigma}_2 \mathbf{\Sigma}_1^{-\frac{1}{2}} \right) \right\|_2^2, \tag{1}$$

where $\log$ is the matrix logarithm and $\|.\|_2$ is the Frobenius norm. Notably, this distance is affine invariant, *i.e* $\forall \boldsymbol{A} \in \mathrm{Gl}_p$ (set of $p \times p$ invertible matrices),

$$d_{\mathcal{S}_p^{++}}(\boldsymbol{A}\mathbf{\Sigma}_1\boldsymbol{A}^T, \boldsymbol{A}\mathbf{\Sigma}_2\boldsymbol{A}^T) = d_{\mathcal{S}_p^{++}}(\mathbf{\Sigma}_1, \mathbf{\Sigma}_2). \tag{2}$$

This invariance property is of particular interest for applications based on mixing models [5, 6], *i.e* the measured signal is assumed to be a linear combination of a non-measurable source signal. In this case, the distances in the source space are equal to those in the measured signal space.

Then, many classification/clustering algorithms, *e.g Nearest centroid* or *K-means*, need to compute centers of mass. Here, centers of mass are computed on sets of covariance matrices $\{\mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_M\}$. Since, these matrices lie on a Riemannian manifold, the classical arithmetic mean is extended to the Riemannian case. Indeed, the Riemannian center of mass of $\{\mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_M\}$, denoted $\mathbf{\Sigma}^\star$, associated to the distance (1), is defined as the minimizer of the variance [7, 8],

$$\mathbf{\Sigma}^\star = \underset{\mathbf{\Sigma} \in \mathcal{S}_p^{++}}{\arg\min} \frac{1}{2M} \sum_{i=1}^{M} d^2_{\mathcal{S}_p^{++}}(\mathbf{\Sigma}, \mathbf{\Sigma}_i). \tag{3}$$

A gradient descent achieves this minimization [4].

As mentioned earlier, this geometry assumes that the signals are centered. Therefore, it does not use the mean/location whereas it can be a discriminative feature, *e.g.* on multispectral images where signals are non-centered [9]. To preserve invariances by affine transformations, the FIM is extended to the case of non centered signals. Unfortunately, the associated Riemannian geometry is not fully known. Especially, its distance is unknown. In the remainder of this paper, we propose to address this issue by using geodesic triangles from [10, 11]. Affine invariant divergences are obtained in Section 3 and a computation of centers of mass is proposed in Section 4. This framework is tested on real data in Section 5.

2. INFORMATION GEOMETRY OF THE MULTIVARIATE GAUSSIAN DISTRIBUTION

Let a set of n data points $\boldsymbol{x}_i \in \mathbb{R}^p$ sampled from a random variable $\boldsymbol{x}$ following a Gaussian distribution

$$\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{4}$$

The parameters $\boldsymbol{\mu} \in \mathbb{R}^p$ and $\boldsymbol{\Sigma} \in \mathcal{S}_p^{++}$ (set of symmetric positive definite matrices) are the location and covariance matrix respectively. The negative log-likelihood is defined on the set $\mathcal{N}^p = \mathbb{R}^p \times \mathcal{S}_p^{++}$ and given $\upsilon = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ writes

$$L(\upsilon) = \frac{n}{2} \log |\boldsymbol{\Sigma}| + \frac{1}{2} \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_i - \boldsymbol{\mu}). \tag{5}$$

The maximum likelihood estimators of the Gaussian distribution are the well known sample mean and SCM,

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i, \quad \hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}})^T. \tag{6}$$

Then, $\mathcal{N}^p$ is turned into a Riemannian manifold. The tangent space $T_\upsilon \mathcal{N}^p$ of $\mathcal{N}^p$ at υ is identified to the product space $\mathbb{R}^p \times \mathcal{S}_p$ with $\mathcal{S}_p$ the set of symmetric matrices. Moreover, $\mathcal{N}^p$ is equipped with the FIM associated to the negative log-likelihood (5). Let $\xi = \left(\boldsymbol{\xi}_{\boldsymbol{\mu}}, \boldsymbol{\xi}_{\boldsymbol{\Sigma}}\right)$, $\eta = \left(\boldsymbol{\eta}_{\boldsymbol{\mu}}, \boldsymbol{\eta}_{\boldsymbol{\Sigma}}\right) \in T_\upsilon \mathcal{N}^p$, this metric writes [1]

$$\langle \xi, \eta \rangle_\upsilon^{\mathcal{N}^p} = \boldsymbol{\xi}_{\boldsymbol{\mu}}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta}_{\boldsymbol{\mu}} + \frac{1}{2} \operatorname{Tr}(\boldsymbol{\Sigma}^{-1} \boldsymbol{\xi}_{\boldsymbol{\Sigma}} \boldsymbol{\Sigma}^{-1} \boldsymbol{\eta}_{\boldsymbol{\Sigma}}). \tag{7}$$

Remarkably, the FIM (7) is invariant under affine transformations. Given $\boldsymbol{A} \in \mathrm{Gl}_p$ and $\boldsymbol{\mu}_0 \in \mathbb{R}^p$ we verify that

$$\langle \mathrm{D}\, \phi(\upsilon)[\xi], \mathrm{D}\, \phi(\upsilon)[\eta] \rangle_{\phi(\upsilon)}^{\mathcal{N}^p} = \langle \xi, \eta \rangle_\upsilon^{\mathcal{N}^p}, \tag{8}$$

where the affine transformation writes,

$$\phi(\upsilon) = (\boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{\mu}_0, \boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}^T), \tag{9}$$

and $\mathrm{D}\, \phi(\upsilon)[\xi]$ is the directional derivative of ϕ at υ in the direction ξ (see *e.g* [12, Ch. 3]).

A geodesic $\gamma(t) = (\boldsymbol{\mu}(t), \boldsymbol{\Sigma}(t)) : \mathbb{R} \to \mathcal{N}^p$ associated to the FIM (7) must have a zero acceleration [10]

$$\begin{cases} \ddot{\boldsymbol{\mu}}(t) - \dot{\boldsymbol{\Sigma}}(t) \boldsymbol{\Sigma}(t)^{-1} \dot{\boldsymbol{\mu}}(t) = \boldsymbol{0} \\ \ddot{\boldsymbol{\Sigma}}(t) + \dot{\boldsymbol{\mu}}(t) \dot{\boldsymbol{\mu}}(t)^T - \dot{\boldsymbol{\Sigma}}(t) \boldsymbol{\Sigma}(t)^{-1} \dot{\boldsymbol{\Sigma}}(t) = \boldsymbol{0}. \end{cases} \tag{10}$$

An explicit expression of the geodesic on $\mathcal{N}^p$ with initial position $\gamma(0) = \upsilon$ and initial velocity $\dot{\gamma}(0) = \xi$ is derived in [10],

$$\gamma(t) = (\boldsymbol{\mu}(t), \boldsymbol{\Sigma}(t)) = \left(2\boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{R}(t) \sinh\left(\frac{t}{2}\boldsymbol{G}\right) \boldsymbol{G}^{-} \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\xi}_{\boldsymbol{\mu}} + \boldsymbol{\mu}, \; \boldsymbol{\Sigma}^{\frac{1}{2}} \boldsymbol{R}(t) \boldsymbol{R}(t)^T \boldsymbol{\Sigma}^{\frac{1}{2}}\right) \tag{11}$$

where $\boldsymbol{G}^2 = \left(\boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\xi}_{\boldsymbol{\Sigma}} \boldsymbol{\Sigma}^{-\frac{1}{2}}\right)^2 + 2\boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\xi}_{\boldsymbol{\mu}} \boldsymbol{\xi}_{\boldsymbol{\mu}}^T \boldsymbol{\Sigma}^{-\frac{1}{2}}$

and $\boldsymbol{R}(t) = \left(\cosh\left(\frac{t}{2}\boldsymbol{G}\right) - \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{\xi}_{\boldsymbol{\Sigma}} \boldsymbol{\Sigma}^{-\frac{1}{2}} \boldsymbol{G}^{-} \sinh\left(\frac{t}{2}\boldsymbol{G}\right)\right)^{-T}$.

However (11) only gives an expression of a geodesic with initial position and velocity. Unfortunately, in the general case, a closed form expression of a geodesic between two points $\upsilon_1 = (\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $\upsilon_2 = (\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ remains unknown. Hence, the distance between υ_1 and υ_2 associated to the FIM (7) is also unknown. Using other metrics than the FIM could give closed form distances but they would not necessarily have the affine transformation invariance property. Instead, we propose to use geodesic triangles derived from (11).

3. DIVERGENCES

Geodesic triangles between υ_1 and υ_2 using the expression (11) can be derived. Indeed, by carefully choosing intermediate points υ, geodesics are obtained between υ_1 and υ and then between υ and υ_2. Hence, we get geodesic triangles $\upsilon_1 \to \upsilon \to \upsilon_2$.

The squared arc-length of one of these geodesic triangles is then measured to get a divergence denoted $\delta^2_{\mathcal{N}^p}$. By construction, these divergences $\delta^2_{\mathcal{N}^p}$ are invariant by affine transformation,

$$\delta^2_{\mathcal{N}^p}(\phi(\upsilon_1), \phi(\upsilon_2)) = \delta^2_{\mathcal{N}^p}(\upsilon_1, \upsilon_2). \tag{12}$$

To construct those triangles, we recall that the manifold with a fixed location vector $\mathcal{N}^p_{\boldsymbol{\mu}} = \{(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \boldsymbol{\Sigma} \in \mathcal{S}_p^{++}\}$ endowed with metric (7) is a geodesic submanifold of $\mathcal{N}^p$. Hence, in the case $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$, the squared distance on $\mathcal{N}^p$ is

$$d^2_{\mathcal{N}^p}(\upsilon_1, \upsilon_2) = \frac{1}{2} d^2_{\mathcal{S}_p^{++}}(\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2) \tag{13}$$

Thus, to create a triangle between υ_1 and υ_2, it suffices to find an intermediate point $\upsilon = (\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is determined such that a geodesic (11) is known between υ_1 and υ.

Based on this scheme, [10] proposed to use a rescaling of the initial covariance matrix as an intermediate point, *i.e.*

$$\upsilon_c = (\boldsymbol{\mu}_2, c\boldsymbol{\Sigma}_1), \tag{14}$$

with $c = \left|\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_2\right|^{\frac{1}{p}} = \arg\min_{c \in \mathbb{R}_*^+} d^2_{\mathcal{N}^p}(\upsilon_c, \upsilon_2)$. Using this point, a first separable and invariant under affine transformation (9) divergence on $\mathcal{N}^p$ is proposed in Corollary 1.

Corollary 1 (Divergence $\delta^2_{c,\mathcal{N}^p}$)**.** *A separable and invariant under affine transformations* (9)*, divergence on $\mathcal{N}^p$ is*

$$\delta^2_{c,\mathcal{N}^p}(\upsilon_1, \upsilon_2) = 2\operatorname{acosh}\left(\frac{c^{-\frac{1}{2}}}{2}\left(c + 1 + \frac{1}{2}\Delta\boldsymbol{\mu}^T \boldsymbol{\Sigma}_1^{-1} \Delta\boldsymbol{\mu}\right)\right)^2 + \frac{(p-1)}{2}\log(c)^2 + \frac{1}{2}\left\|\log\left(c\boldsymbol{\Sigma}_2^{-\frac{1}{2}}\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_2^{-\frac{1}{2}}\right)\right\|_2^2.$$

where $\Delta\boldsymbol{\mu} = \boldsymbol{\mu}_2 - \boldsymbol{\mu}_1$ and $c = \left|\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_2\right|^{\frac{1}{p}}$.

Proof. Using the intermediate point $\upsilon_c = (\boldsymbol{\mu}_2, c\boldsymbol{\Sigma}_1)$, and applying the construction of triangles explained earlier, we get $\delta^2_{c,\mathcal{N}^p}(\upsilon_1,\upsilon_2) = \rho^2(\upsilon_1,\upsilon_c) + d^2_{\mathcal{N}^p}(\upsilon_c,\upsilon_2)$, where ρ is the arc length of a geodesic (11) computed in Equation (18) of [10]. Then, ρ is simplified. By denoting $\widetilde{\boldsymbol{\mu}} = \boldsymbol{\Sigma}_1^{-\frac{1}{2}}\Delta\boldsymbol{\mu}$, we get

$$\frac{1}{2}\rho^2(\upsilon_1,\upsilon_c) = \left\|\operatorname{acosh}\left(\frac{c^{-\frac{1}{2}}}{2}(\boldsymbol{I}_p + c\boldsymbol{I}_p + \frac{1}{2}\widetilde{\boldsymbol{\mu}}\widetilde{\boldsymbol{\mu}}^T)\right)\right\|_2^2$$
$$= \operatorname{acosh}\left(\frac{c^{-\frac{1}{2}}}{2}(c+1+\frac{1}{2}\widetilde{\boldsymbol{\mu}}^T\widetilde{\boldsymbol{\mu}})\right)^2 + (p-1)\operatorname{acosh}\left(\frac{c^{-\frac{1}{2}}+c^{\frac{1}{2}}}{2}\right)^2$$

Using $\operatorname{acosh}\left(\frac{c^{-\frac{1}{2}}+c^{\frac{1}{2}}}{2}\right)^2 = \log(c^{\frac{1}{2}})^2 = \frac{1}{4}\log(c)^2$ and Equation (13), we get the divergence $\delta^2_{c,\mathcal{N}^p}$. □

In [11], the authors proved that the orthogonal projection of υ_1 onto $\mathcal{N}^p_{\boldsymbol{\mu}_2}$ is

$$\upsilon_\perp = \left(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_1 + \frac{1}{2}\Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T\right). \tag{15}$$

The squared arc length of the geodesic between υ_1 and $\upsilon_\perp$ is also computed in [11],

$$\delta^2_\perp(\upsilon_1,\upsilon_\perp) = \frac{1}{2}\operatorname{acosh}\left(1 + \Delta\boldsymbol{\mu}^T\boldsymbol{\Sigma}_1^{-1}\Delta\boldsymbol{\mu}\right)^2. \tag{16}$$

Hence, using the intermediate point $\upsilon_\perp$ and summing Equation (16) with Equation (13) we get a second separable, and invariant under affine transformation (9) divergence on $\mathcal{N}^p$. This divergence is proposed in Corollary 2.

Corollary 2 (Divergence $\delta^2_{\perp,\mathcal{N}^p}$)**.** *A separable, and invariant under affine transformations* (9)*, divergence on $\mathcal{N}^p$ is*

$$\delta^2_{\perp,\mathcal{N}^p}(\upsilon_1,\upsilon_2) = \frac{1}{2}\Bigg[\operatorname{acosh}\left(1 + \Delta\boldsymbol{\mu}^T\boldsymbol{\Sigma}_1^{-1}\Delta\boldsymbol{\mu}\right)^2 + \left\|\log\left(\boldsymbol{\Sigma}_2^{-\frac{1}{2}}\left(\boldsymbol{\Sigma}_1 + \frac{1}{2}\Delta\boldsymbol{\mu}\Delta\boldsymbol{\mu}^T\right)\boldsymbol{\Sigma}_2^{-\frac{1}{2}}\right)\right\|_2^2\Bigg].$$

4. RIEMANNIAN OPTIMIZATION AND ESTIMATION OF CENTERS OF MASS

In machine learning, some important clustering algorithms / classifiers, *e.g. K-means* or *Nearest centroïd*, require a proximity measure and an algorithm to compute centers of mass. Since the divergences proposed in Corollaries 1 and 2 can be used as a proximity measure, it only remains to explicit an algorithm to compute centers of mass. Such an algorithm is described in Section 4.2. It relies on optimization on the Riemannian manifold $\mathcal{N}^p$. Hence, we begin by presenting tools to perform gradient based optimization on $\mathcal{N}^p$.

4.1. Riemannian optimization on $\mathcal{N}^p$

In this subsection we consider a function $f : \mathcal{N}^p \mapsto \mathbb{R}$. The objective is to find the parameter $\upsilon^\star$ minimizing f on $\mathcal{N}^p$,

$$\upsilon^\star = \arg\min_{\upsilon\in\mathcal{N}^p} f(\upsilon). \tag{17}$$

Since $\mathcal{N}^p$ is a Riemannian manifold, we leverage the framework of optimization on Riemannian manifolds [12] to compute (17). Thus, we provide two important tools for Riemannian optimization, both associated to the metric (7) : *(i)* the Riemannian gradient in Proposition 1, *(ii)* a second order retraction in Proposition 2 (approximation of the geodesic (11) with lower calculation cost and better numerical stability). With these tools, we can apply gradient based algorithms on $\mathcal{N}^p$ to minimize f. The corresponding Riemannian gradient descent is given in Algorithm 1.

Proposition 1 (Riemannian gradient)**.** *Let $\upsilon \in \mathcal{N}^p$, the Riemannian gradient of f at υ is*

$$\operatorname{grad}_{\mathcal{N}^p} f(\upsilon) = P^{\mathcal{N}^p}_\upsilon\left(\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\mu}}, 2\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}\right)$$

where $\forall\xi \in \mathbb{R}^p \times \mathbb{R}^{p\times p}$, $P^{\mathcal{N}^p}_\upsilon(\xi) = (\boldsymbol{\xi}_{\boldsymbol{\mu}}, \operatorname{sym}(\boldsymbol{\xi}_{\boldsymbol{\Sigma}}))$, with $\operatorname{sym}(\boldsymbol{\xi}) = \frac{1}{2}(\boldsymbol{\xi} + \boldsymbol{\xi}^T)$, is the orthogonal projection according to the FIM (7) *onto $T_\upsilon\mathcal{N}^p$ and $\operatorname{grad}_\epsilon f(\upsilon) = (\boldsymbol{G}_{\boldsymbol{\mu}}, \boldsymbol{G}_{\boldsymbol{\Sigma}})$ is the Euclidean gradient of f in $\mathbb{R}^p \times \mathbb{R}^{p\times p}$.*

Proof. Using the definition of the gradient associated to the Euclidean metric [12, Ch. 3], we get $\forall\xi \in T_\upsilon\mathcal{N}^p$

$$\mathrm{D}\, f(\upsilon)[\xi] = \boldsymbol{G}_{\boldsymbol{\mu}}^T\boldsymbol{\xi}_{\boldsymbol{\mu}} + \operatorname{Tr}\left(\boldsymbol{G}_{\boldsymbol{\Sigma}}^T\boldsymbol{\xi}_{\boldsymbol{\Sigma}}\right)$$
$$= (\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\mu}})^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi}_{\boldsymbol{\mu}} + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{\Sigma}^{-1}\left(2\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\Sigma}}^T\boldsymbol{\Sigma}\right)\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi}_{\boldsymbol{\Sigma}}\right)$$
$$= (\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\mu}})^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi}_{\boldsymbol{\mu}} + \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{\Sigma}^{-1}\operatorname{sym}(2\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\Sigma}}^T\boldsymbol{\Sigma})\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi}_{\boldsymbol{\Sigma}}\right)$$
$$= \langle P^{\mathcal{N}^p}_\upsilon\left(\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\mu}}, 2\boldsymbol{\Sigma}\boldsymbol{G}_{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}\right), \xi\rangle^{\mathcal{N}^p}_\upsilon.$$

Using the definition of the Riemannian gradient $\mathrm{D}\, f(\upsilon)[\xi] = \langle\operatorname{grad}_{\mathcal{N}^p} f(\upsilon), \xi\rangle^{\mathcal{N}^p}_\upsilon$ [12, Ch. 3], we get the Proposition 1. □

Proposition 2 (Second order retraction)**.** *A second order retraction at $\upsilon \in \mathcal{N}^p$ of $\xi \in T_\upsilon\mathcal{N}^p$ is,*

$$R^{\mathcal{N}^p}_\upsilon(\xi) = \Big(\boldsymbol{\mu} + \boldsymbol{\xi}_{\boldsymbol{\mu}} + \frac{1}{2}\boldsymbol{\xi}_{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma} + \boldsymbol{\xi}_{\boldsymbol{\Sigma}} + \frac{1}{2}\left(\boldsymbol{\xi}_{\boldsymbol{\Sigma}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\xi}_{\boldsymbol{\Sigma}} - \boldsymbol{\xi}_{\boldsymbol{\mu}}\boldsymbol{\xi}_{\boldsymbol{\mu}}^T\right)\Big).$$

Proof. $\forall\upsilon \in \mathcal{N}^p$, $R^{\mathcal{N}^p}_\upsilon$ is a smooth mapping from $T_\upsilon\mathcal{N}^p$ onto $\mathcal{N}^p$. To be a second order retraction, it remains to check the three following properties [12, Ch. 4 and 5]: $\forall\xi \in T_\upsilon\mathcal{N}^p$

$$R^{\mathcal{N}^p}_\upsilon(0) = \upsilon, \mathrm{D}\, R^{\mathcal{N}^p}_\upsilon(0_\upsilon)[\xi] = \xi, \frac{\mathrm{D}^2}{dt^2}R^{\mathcal{N}^p}_\upsilon(t\xi)\Big|_{t=0} = 0$$

where 0_υ denotes the zero element of $T_\upsilon\mathcal{N}^p$ and $\frac{\mathrm{D}^2}{dt^2}\gamma$ denotes the acceleration of the curve $t \mapsto \gamma(t)$ on $\mathcal{N}^p$ (see [12, Ch. 5]). The first two properties are easily verified. By denoting $R^{\mathcal{N}^p}_\upsilon(t\xi) = (\boldsymbol{\mu}(t), \boldsymbol{\Sigma}(t))$, and using Equation (10), the third property is equivalent to

$$\begin{cases}\ddot{\boldsymbol{\mu}}(0) - \dot{\boldsymbol{\Sigma}}(0)\boldsymbol{\Sigma}(0)^{-1}\dot{\boldsymbol{\mu}}(0) = \mathbf{0}\\ \ddot{\boldsymbol{\Sigma}}(0) + \dot{\boldsymbol{\mu}}(0)\dot{\boldsymbol{\mu}}(0)^T - \dot{\boldsymbol{\Sigma}}(0)\boldsymbol{\Sigma}(0)^{-1}\dot{\boldsymbol{\Sigma}}(0) = \mathbf{0},\end{cases}$$

which is also verified. □

Algorithm 1: Riemannian gradient descent [12]

Input : Initial iterate $\upsilon_1 \in \mathcal{N}^p$.
Output: Sequence of iterates $\{\upsilon_k\}$.
$k := 1$;
while *no convergence* **do**
  Compute a step size α (see [12, Ch. 4]) and set $\upsilon_{k+1} := R_{\upsilon_k}^{\mathcal{N}^p}(-\alpha \operatorname{grad}_{\mathcal{N}^p} f(\upsilon_k))$;
  $k := k + 1$;
end

4.2. Estimation of Riemannian centers of mass

Some important algorithms in machine learning require the computation the center of mass of a set of points $S = \{\upsilon_i\}_{i=1}^M \subset \mathcal{N}^p$. This center is associated to a proximity measure which in our case is one of the divergences, $\delta^2_{\mathcal{N}^p}$, defined in Section 3. Similarly to (3), the Riemannian center of mass $\upsilon^\star$ is defined as the minimizer of the variance of S

$$\upsilon^\star = \underset{\upsilon \in \mathcal{N}^p}{\arg\min} \frac{1}{2M} \sum_{i=1}^{M} \delta^2_{\mathcal{N}^p}(\upsilon, \upsilon_i). \tag{18}$$

Using tools from the previous subsection, we can perform optimization on $\mathcal{N}^p$. Hence, gradient based algorithms can be applied to achieve (18) (*e.g* using Algorithm 1). Using Proposition 1, computing the Riemannian gradient of the variance defined in (18) amounts to computing its Euclidean gradient. The latter is easily numerically computed using automatic differentiation libraries like Autograd [13] or JAX [14].

5. APPLICATION

In this section, we provide an application of the theoretical framework presented earlier on the large-scale satellite image time series dataset for crop type mapping called *Breizhcrops* [15].

More specifically, for each crop $n = 45$ observations $\boldsymbol{x}_i \in \mathbb{R}^p$ are measured over time. Each $\boldsymbol{x}_i$ contains measurements of reflectance of $p = 13$ spectral bands. Then, these measurements are concatenated into one batch $\boldsymbol{X}_j = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$. Hence, we get one matrix $\boldsymbol{X}_j$ per crop and each one belongs to an unknown class $y \in [\![1, K]\!]$. These $K = 9$ classes represent crop types such as nuts, barley or wheat and are heavily imbalanced, *i.e* some classes are much more represented than others. The data are divided into two sets: a training set and a test set with 485 649 and 122 614 batches respectively. We apply a single preprocessing step: all data are centered using the global mean. For simplicity, the matrix $\boldsymbol{X}_j$ is simply noted $\boldsymbol{X}$ in the following.

To classify these crops, we apply a *Nearest centroid* algorithm on descriptors. Indeed, the use of statistical descriptors is a classical procedure in machine learning as they are often more discriminative than raw data (see *e.g* [2, 3]). Hence, this classification algorithm works in three steps: *(i)* For each batch $\boldsymbol{X}$, a descriptor is computed (*e.g* the sample mean or the SCM (6)). *(ii)* Then, on the training set, the center of mass of the descriptors of each class is computed. *(iii)* Finally, on the test set, each descriptor is associated to the nearest center of mass. Thus, we get a classification of the $\boldsymbol{X}$.

Estimator of $\boldsymbol{X}$	Geometry	OA (%)	AA (%)
$\boldsymbol{X}$	$\mathbb{R}^{p \times n}$	10.1	18.5
$\hat{\boldsymbol{\mu}}$	$\mathbb{R}^p$	13.2	14.8
$\hat{\boldsymbol{\Sigma}}$, ($\boldsymbol{\mu}$ known)	$\mathcal{S}_p^{++}$	43.9	28.1
$\hat{\boldsymbol{\Sigma}}$, ($\boldsymbol{\mu}$ unknown)	$\mathcal{S}_p^{++}$	46.7	30.1
$(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$	$\mathcal{N}^p$ with $\delta^2_{c,\mathcal{N}^p}$	**54.3**	**37.0**
$(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$	$\mathcal{N}^p$ with $\delta^2_{\perp,\mathcal{N}^p}$	53.3	35.7

Table 1. Performance of the different estimators and Riemannian geometries on the *Breizhcrops* dataset [15]. OA = Overall Accuracy, AA = Average Accuracy.

The different descriptors used in the application are the following. The first two descriptors are the batches themselves $\boldsymbol{X}$ and their sample means $\hat{\boldsymbol{\mu}}$ (6). Their associated geometry is the Euclidean one with the Frobenius distance. Thus, the center of mass is the classical element-wise arithmetic mean. Then, the next two estimators are the SCMs $\hat{\boldsymbol{\Sigma}}$ (6) with location assumed to be known or not. In the case of known location, the SCM is simply estimated as $\hat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i \boldsymbol{x}_i^T$. The associated geometry is $\mathcal{S}_p^{++}$ as presented in Equations (1) and (3). Finally, the last two descriptors use both sample mean and SCM, $(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}})$ from (6). These estimators are used with the geometry $\mathcal{N}^p$ and the two divergences $\delta^2_{c,\mathcal{N}^p}$ and $\delta^2_{\perp,\mathcal{N}^p}$ presented in Corollaries 1 and 2 respectively. Riemannian centers of mass are computed using Algorithm 1, implemented using Pymanopt [16] (the python version of Manopt [17]).

Table 1 presents the Overall Accuracy and Average Accuracy of the different descriptors and geometries used in the *Nearest centroid*. Estimators using $\hat{\boldsymbol{\Sigma}}$ along with the FIM clearly outperform the others. Also, the three estimators assuming $\boldsymbol{\mu}$ is unknown perform better than the others. This shows the interest of not considering $\boldsymbol{\mu} = \boldsymbol{0}$ for such applications, even if the global mean has been subtracted in a preprocessing step. Finally, using the divergences proposed in Corollaries 1 and 2 with their Riemannian centers of mass greatly improves both Overall Accuracy and Average Accuracy. These results confirm the interest of considering geodesic triangles when the distance associated to the FIM is not available in closed form.

6. CONCLUSIONS

This paper has proposed two affine invariant divergences that handle both first and second order statistics of the Gaussian distribution. The Riemannian geometry associated to the FIM has been studied and an algorithm to compute Riemannian centers of mass associated to these divergences has been proposed. Finally, these tools have been applied on a classification problem to show the interest of the proposed method.

7. REFERENCES

[1] L. T. Skovgaard, "A Riemannian geometry of the multivariate Normal model," *Scandinavian Journal of Statistics*, vol. 11, no. 4, pp. 211–223, 1984.

[2] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass brain–computer interface classification by riemannian geometry," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 4, pp. 920–928, 2012.

[3] O. Tuzel, F. Porikli, and P. Meer, "Human detection via classification on riemannian manifolds," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[4] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *International Journal of computer vision*, vol. 66, no. 1, pp. 41–66, 2006.

[5] Y.E. Shimabukuro and J.A. Smith, "The least-squares mixing models to generate fraction images derived from remote sensing multispectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 29, no. 1, pp. 16–20, 1991.

[6] J.-F. Cardoso, "Blind signal separation: statistical principles," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009–2025, 1998.

[7] H. Karcher, "Riemannian center of mass and mollifier smoothing," *Communications on Pure and Applied Mathematics*, vol. 30, no. 5, pp. 509–541, 1977.

[8] M. Moakher, "A differential geometric approach to the geometric mean of symmetric positive-definite matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 3, pp. 735–747, 2005.

[9] D. G. Manolakis, D. Marden, J. P. Kerekes, and G. A. Shaw, "Statistics of hyperspectral imaging data," in *Algorithms for Multispectral, Hyperspectral, and Ultraspectral Imagery VII*, Sylvia S. Shen and Michael R. Descour, Eds. International Society for Optics and Photonics, 2001, vol. 4381, pp. 308 – 316, SPIE.

[10] M. Calvo and J. M. Oller, "An explicit solution of information geodesic equations for the multivariate normal model," *Statistics & Risk Modeling*, vol. 9, no. 1-2, pp. 119–138, 1991.

[11] M. Tang, Y. Rong, J. Zhou, and X. Li, "Information geometric approach to multisensor estimation fusion," *IEEE Transactions on Signal Processing*, vol. 67, no. 2, pp. 279–292, 2019.

[12] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, 2008.

[13] D. Maclaurin, D. Duvenaud, and Adams R.P., "Autograd: Effortless gradients in pure numpy," *AutoML workshop ICML*, 2015.

[14] J. Bradbury, R. Frostig, P. Hawkins, M.J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," 2018.

[15] M. Rußwurm, C. Pelletier, M. Zollner, S. Lefèvre, and M. Körner, "Breizhcrops: A time series dataset for crop type mapping," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences ISPRS (2020)*, 2020.

[16] J. Townsend, N. Koep, and S. Weichwald, "Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 4755–4759, Jan. 2016.

[17] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, "Manopt, a Matlab toolbox for optimization on manifolds," *Journal of Machine Learning Research*, vol. 15, pp. 1455–1459, 2014.