Outils d'apprentissage automatique pour la reconnaissance de signaux temporels

Maxime Sangnier

THÈSE

présentée par

Maxime SANGNIER

en vue de l'obtention du grade de :

Docteur de Normandie Université

Spécialité informatique

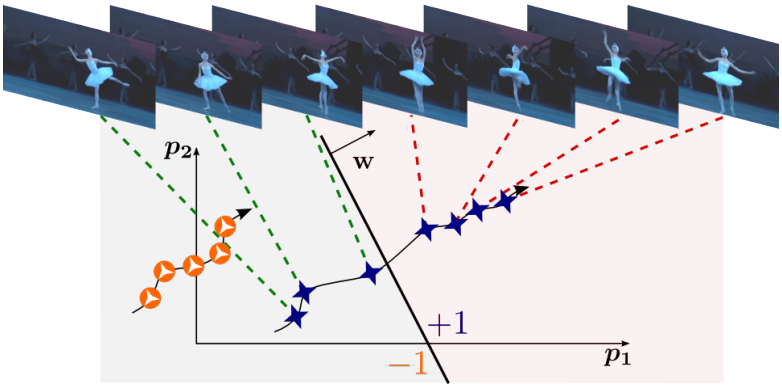
délivré par

l'Université de Rouen

École Doctorale Sciences Physiques, Mathématiques et de l'Information pour l'Ingénieur

Outils d'apprentissage automatique pour la reconnaissance de signaux temporels

(6 janvier 2015)



Devant le jury composé de :

Florence D'ALCHÉ-BUC	Professeur, Télécom ParisTech	(rapporteuse)
Paul HONEINE	Maître de conférence, UT de Troyes	(rapporteur)
Gilles GASSO	Professeur, INSA de Rouen	(examinateur)
Gaël RICHARD	Professeur, Télécom ParisTech	(examinateur)
Michèle SEBAG	Directrice de recherche, CNRS	(examinatrice)
Jérôme GAUTHIER	Ingénieur-chercheur, CEA, LIST	(encadrant)
Alain RAKOTOMAMONJY	Professeur, Université de Rouen	(directeur)

OUTILS D'APPRENTISSAGE AUTOMATIQUE POUR LA RECONNAISSANCE DE SIGNAUX TEMPORELS

Résumé Les travaux présentés ici couvrent deux thématiques de la reconnaissance de signaux temporels par des systèmes numériques dont certains paramètres sont inférés à partir d'un ensemble limité d'observations. La première est celle de la détermination automatique de caractéristiques discriminantes. Pour ce faire, nous proposons un algorithme de génération de colonnes capable d'apprendre une transformée Temps-Fréquence (TF), mise sous la forme d'un banc de filtres, de concert à une machine à vecteurs supports. Cet algorithme est une extension des techniques existantes d'apprentissage de noyaux multiples, combinant de manière non-linéaire une infinité de noyaux. La seconde thématique dans laquelle s'inscrivent nos travaux est l'appréhension de la temporalité des signaux. Si cette notion a été abordée au cours de notre première contribution, qui a pointé la nécessité de déterminer au mieux la résolution temporelle d'une représentation TF, elle prend tout son sens dans une prise de décision au plus tôt. Dans ce contexte, notre seconde contribution fournit un cadre méthodologique permettant de détecter précocement un événement particulier au sein d'une séquence, c'est à dire avant que ledit événement ne se termine. Celui-ci est construit comme une extension de l'apprentissage d'instances multiples et des espaces de similarité aux séries temporelles. De plus, nous accompagnons cet outil d'un algorithme d'apprentissage efficace et de garanties théoriques de généralisation. L'ensemble de nos travaux a été évalué sur des signaux issus d'interfaces cerveau-machine, des paysages sonores et des vidéos représentant des actions humaines.

Mots clefs Apprentissage de représentations, détection précoce, Machine à Vecteurs Supports (SVM), apprentissage de noyaux.

MACHINE LEARNING TOOLS FOR TEMPORAL SIGNAL RECOGNITION

Abstract The work presented here tackles two different subjects in the wide thematic of how to build a numerical system to recognize temporal signals, mainly from limited observations. The first one is automatic feature extraction. For this purpose, we present a column generation algorithm, which is able to jointly learn a discriminative Time-Frequency (TF) transform, cast as a filter bank, with a support vector machine. This algorithm extends the state of the art on multiple kernel learning by non-linearly combining an infinite amount of kernels. The second direction of research is the way to handle the temporal nature of the signals. While our first contribution pointed out the importance of correctly choosing the time resolution to get a discriminative TF representation, the role of the time is clearly enlightened in early recognition of signals. Our second contribution lies in this field and introduces a methodological framework for early detection of a special event in a time-series, that is detecting an event before it ends. This framework builds upon multiple instance learning and similarity spaces by fitting them to the particular case of temporal sequences. Furthermore, our early detector comes with an efficient learning algorithm and theoretical guarantees on its generalization ability. Our two contributions have been empirically evaluated with brain-computer interface signals, soundscapes and human actions movies.

Keywords Feature learning, early detection, Support Vector Machine (SVM), kernel learning.

Thèse préparée au :

Commissariat à l'Énergie Atomique et aux Énergies Alternatives,
Laboratoire d'Intégration des Systèmes et des Technologies.

CEA SACLAY DIGITEO LABS
Bât. 565
91191 GIF-SUR-YVETTE CEDEX
Tél : +33 (0)1 69 08 25 01

Table des matières

Glossaire

ACP Analyse en Composantes Principales. 2, 35, 39

BdF Banc de Filtres. xvi, xviii, 34, 37, 40–43, 45, 47, 48, 59, 62–66, 68, 69, 72–78, 83–91, 93, 94, 129, 130, 140, 144, 148

BFGS Broyden-Fletcher-Goldfarb-Shanno. 39

CHSC Compétition de classification d'enregistrements sonores cardiaques (*Classifying Heart Sounds Challenge*). 69, 70, 72

CNN Réseau de neurones convolutifs (*Convolutional Neural Network*). xviii, xix, 36, 44–46, 50, 51, 84, 85, 87–89, 91, 93, 94, 129, 130

CSP Forme spatiale commune (*Common Spatial Pattern*). 42, 43, 140

DCT Transformée en cosinus discrète (*Discrete Cosine Transform*). 34, 42, 51, 144

DFE Extraction de caractéristiques discriminantes (*Discriminative Feature Extraction*). 42

ECG Électro-Cardiogramme. 49

EEG Électro-Encéphalogramme. 42, 43

HMM Modèle de Markov caché (*Hidden Markov Model*). 43, 55, 140

ICM Interface Cerveau-Machine. xix, 43, 49, 86, 88–94

iid Indépendants et identiquement distribués. 4–6, 21

IKL Apprentissage de noyaux infinis (*Infinite Kernel Learning*). 77, 78, 85, 86

KKT Karush-Kuhn-Tucker. 13, 79, 80, 111, 135, 149, 150

LDA Analyse linéaire discriminante (*Linear Discriminant Analysis*). 4, 140

LDB *Local Discriminant Basis*. 48, 49

LOO Un en dehors (*Leave-One-Out*). 16, 56

LP Programme linéaire (*Linear Program*). 7, 20, 22, 105, 106, 115, 122, 136

MEL MÉLodie. 42, 43, 143, 144

MFCC Coefficients cepstraux mel-fréquences (*Mel-Frequency Cepstral Coefficients*). xiv, xvi, xix, 34–36, 41–43, 50, 51, 93, 94, 97, 102, 120, 143

MIL Apprentissage d'instances multiples (*Multiple Instance Learning*). xvii, xviii, 26–30, 56, 95, 103, 115, 116, 130

MILES Apprentissage d'instances multiples avec sélection intégrée des instances (*Multiple-Instance Learning via Embedded instance Selection*). 115, 117, 120–122, 128, 130

MKL Apprentissage de noyaux multiples (*Multiple Kernel Learning*). 19–25, 50, 62, 72, 77, 78, 81–84, 86, 88, 135, 137

MLP Perceptron multi-couche (*Multi-Layer Perceptron*). 42, 44–46, 50, 51, 85, 88, 140

MMED Détecteur précoce vaste marge (*Maximum Margin Early Detector*). 58, 116, 117, 121–123, 125, 126, 128

MP *Matching Pursuit*. 51, 52

QCQP Programme quadratique à contraintes quadratiques (*Quadratically Constrained Quadratic Program*). 7, 20–22, 56, 66

QP Programme quadratique (*Quadratic Program*). 7, 15, 17, 20, 22, 136

RI Réponse Impulsionnelle. 40–43, 47, 63, 65–70, 72, 77, 78, 81, 84, 86–88, 130, 143

RKHS Espace de Hilbert à noyau reproduisant (*Reproducing Kernel Hilbert Space*). 8–12, 15, 18, 19, 21, 24, 63, 135

SDP Programme semi-défini (*Semi-Definite Program*). xviii, 7, 20, 22, 66, 67, 70, 71

SF Support Fini. 73, 76, 77

SILP Programme linéaire semi-infini (*Semi-Infinite Linear Program*). 20, 22, 25, 77, 135

SMO Optimisation minimale séquentielle (*Sequential Minimal Optimization*). 21, 22, 135

SNR Rapport signal sur bruit (*Signal to Noise Ratio*). 70, 89, 90

SOCP Programme conique de second ordre (*Second-Order Cone Program*). 7, 20, 21

SVM Machine à vecteurs supports (*Support Vector Machine*). xvi–xix, 1, 2, 8–23, 25–30, 38, 39, 43, 45, 46, 49–51, 58, 59, 62, 63, 65, 67, 70–72, 74, 79–81, 83, 85–89, 91, 93, 94, 99, 100, 103–106, 109, 111–116, 120, 121, 128–130, 136–138, 140

TC Temps-Caractéristique. xviii, 31–35, 37, 59, 97–104, 120, 128

TF Temps-Fréquence. i, xiv–xviii, 30, 31, 35–40, 42, 44, 46, 48, 49, 51, 59, 62, 66, 70, 76, 77, 84, 86, 90, 93, 97, 129, 130, 141

WKL Apprentissage de noyaux d'ondelettes (*Wavelet Kernel Learning*). 85–87, 89–91, 93

Notations

ENSEMBLES

$\mathbb{N}_d$	Ensemble des entiers de 1 à d.
$\mathbb{K}$	Corps des réels $\mathbb{R}$ ou des complexes $\mathbb{C}$.
$\mathbb{S}_+^n$	Cône des matrices semi-définies positives de tailles n.
$\mathcal{U}$	Ensemble des signaux discrets de longueur m ($\mathcal{U} \subset \mathbb{K}^m$, m entier).
$\mathcal{O}$	Ensemble des représentations temps-fréquences ($\mathcal{O} = \mathbb{K}^{\left\lfloor \frac{m}{N_1} \right\rfloor} \times \cdots \times \mathbb{K}^{\left\lfloor \frac{m}{N_d} \right\rfloor}$, où $(N_l)_{1 \leq l \leq d}$ est un d-uplet de facteurs de décimations).
$\mathcal{X}$	Ensemble des vecteurs caractéristiques d'une tâche de reconnaissance ($\mathcal{X} \subset \mathbb{K}^d$, d entier).
$\mathcal{Y}$	Ensemble des étiquettes d'une tâche de reconnaissance ($\mathcal{Y} \subset \mathbb{Z}$).
$L^2(\mathbb{R})$	Ensemble des fonctions de carré sommable.

ALGÈBRE

$\imath$	Unité imaginaire.
$\boldsymbol{x}$	Vecteur (sa taille dépend du contexte).
$\boldsymbol{Y}$	Matrice.
$\boldsymbol{K}_+$	Matrice semi-définie positive.
$\boldsymbol{I}_+$	Matrice identité.
$\mathbb{1}$	Vecteur indicateur, composé entièrement de 1.
$\preccurlyeq, \succcurlyeq$	Inégalités composante à composante.
$\lvert \cdot \rvert$	Module composante à composante.
$\lVert \cdot \rVert_p$	(Pseudo-)norme ℓ_p, $\forall p \in \mathbb{R}_+^* : \forall \boldsymbol{x} \in \mathbb{K}^n, \lVert \boldsymbol{x} \rVert_p = (\sum_{i=1}^n \lvert x_i \rvert^p)^{\frac{1}{p}}$.
$\circ$	Selon le contexte, produit d'Hadamard entre matrices ou composition de fonctions.
$\star$	Produit de convolution.

Introduction

MOTIVATIONS

En découvrant ces mots, le lecteur effectue, sans s'en rendre compte, une tâche de reconnaissance. Celle de lettres, puis de mots, avant d'attribuer un concept intelligible aux tâches noires présentes devant ses yeux. Preuve, s'il en était besoin, de la nécessité, inhérente à la perception humaine, d'identifier des formes. L'identification est un moyen naturel d'interaction des êtres vivants avec leur environnement, et à plus haut niveau, de synthèse et de hiérarchisation de l'information. Les mécanismes qui la rendent possibles sont enseignés dès le plus jeune âge, de sorte qu'il devient essentiel de les acquérir à la perfection pour intégrer harmonieusement son environnement. De fait, nous excellons dans les tâches quotidiennes de reconnaissance.

Pourtant, l'identification automatique s'immisce dans un nombre perpétuellement croissant d'applications. Cette immersion des systèmes informatiques dans notre quotidien, se substituant à nous dans l'accomplissement d'actions familières, s'explique de plusieurs manières. La première est sans doute la volonté de libérer notre esprit de tâches mécaniques et fastidieuses. Mécaniques car elles nécessitent la seule application d'un mécanisme acquis, sans ouvrir la porte à une quelconque élévation spirituelle. Fastidieuses, car la répétition prolongée d'un même acte provoque fatalement un ennui et une baisse d'attention, impactant significativement notre efficacité. Ensuite, l'informatique répond adéquatement au fléau, souvent culturel et parfois vital (par exemple lorsqu'il concerne la coordination de secours humanitaires suite à une catastrophe naturelle), du flux tendu ; autrement dit à la volonté ou la nécessité de diligenter une réponse à un problème donné. Il semble en effet évident qu'une machine est apte à agir bien plus rapidement que nous pour certaines tâches. De plus, l'informatique se présente comme une alternative à la défaillance de la perception humaine en grande dimension. En effet, nous ne sommes capables de percevoir que quelques dimensions différentes (les couleurs, l'espace, les odeurs, les sons, *etc.*), et seulement une partie d'entre elles sont assimilées simultanément. Au contraire, un ordinateur est tout à fait apte à appréhender des informations diffusées sur de multiples axes. Par exemple, dans le domaine des images télé-relevées, le radar et l'infrarouge s'ajoutent sans encombre au domaine du visible. Enfin, de manière similaire à l'accumulation des dimensions, un système informatique peut tout à fait manipuler de grandes quantités de données, là où nous sommes limités dans la perception, l'enregistrement et le tri d'information.

Toutes ces raisons expliquent le développement actuel des systèmes automatiques de prise de décision. Comme nous, une grande partie d'entre eux se construit simultanément sur des modèles physiques et sur un ensemble d'observations, mécanisme que l'on appelle *appren-*

tissage supervisé. Chacun (du modèle et de l'ensemble d'observations) a sa propre fonction et ces deux fonctions sont complémentaires. L'un permet d'encoder un *a priori* tandis que l'autre rend possible une adaptation judicieuse du système à l'application spécifiquement visée.

Représentation de l'information

Prenons dès à présent un exemple, celui de reconnaître des enregistrements vocaux, ou plus généralement audio. Il suffirait pour cela de constituer un ensemble de signaux d'apprentissage, de calculer les coefficients cepstraux mel-fréquences (*Mel-Frequency Cepstral Coefficients*, MFCC) sur une fenêtre glissante de 40 ms et de regrouper tous ces représentants dans un unique espace, afin d'apprendre une règle de décision découlant de la distribution empirique des vecteurs caractéristiques. Lorsqu'une nouvelle séquence temporelle se présente, la procédure d'attribution à une classe est naturelle : calculer à nouveau les MFCC sur une fenêtre glissante, prédire une classe pour chacun des vecteurs caractéristiques et attribuer à la séquence la classe apparue en majorité. Cette procédure est somme toute classique mais soulève plusieurs questions chez le néophyte. Pourquoi choisir des MFCC comme descripteurs des signaux ? Pourquoi considérer une fenêtre de 40 ms ? Pourquoi procéder à un vote majoritaire ? La réponse à toutes ces questions est l'*expertise* (un aperçu de celle-ci concernant la reconnaissance audio peut être découvert dans [Richard *et coll.*, 2013]). Pour donner une réponse plus précise à la première question, les MFCC ont été conçus pour reproduire le fonctionnement de la cochlée et caractériser ainsi l'enveloppe spectrale d'un signal vocal. Cette technique est donc née comme un mimétisme de notre mécanisme de perception et d'analyse des sons, lequel est relativement efficace puisqu'il repose sur lui notre principal moyen pour communiquer avec nos semblables. Cette expertise sanctionne plusieurs dizaines d'années d'étude des signaux vocaux, mais surtout, la capacité à modéliser notre système auditif. En revanche, bien qu'inspiré d'un système biologique faisant ses preuves quotidiennement dans des situations diverses et variées, rien ne nous assure *a priori* que le modèle de descripteurs ainsi créé est plus performant qu'un autre sur une tâche d'identification spécifique. Ceci est essentiellement un problème d'*optimalité*. En outre, la modélisation de la construction, ou de la perception d'un phénomène, n'est pas toujours aisée, voire inaccessible. En plus d'un écart d'optimalité, il y a donc aussi un potentiel défaut d'*existence* du modèle. C'est ici que rentre en jeu une nouvelle fois la notion d'apprentissage automatique. Celui-ci apparaît naturellement en réponse à un défaut de modélisation ou à la volonté de construire un objet aussi performant que possible, étant donné un ensemble d'observations. De telles approches, dites d'*apprentissage de descripteurs*, ou d'*extraction automatique de caractéristiques*, sont au cœur des recherches en apprentissage automatique depuis plusieurs décennies. Lesdites approches balayent des outils de nature, d'origine et d'architecture diverses, comme par exemple des techniques (cette fois) adaptatives d'inspiration biologique, des décompositions Temps-Fréquence (TF) énergétiques, des analyses temps-échelle ou encore des synthèses par combinaison d'atomes de différentes natures.

Place du temps

Il existe une différence évidente et significative entre la reconnaissance de séries temporels et de signaux indépendants du temps. Il s'agit de la présence d'une dimension porteuse d'information : le temps. Cette dimension revêt une importance capitale dans la perception que nous avons de notre environnement. Elle rythme l'acquisition de l'information, définie ce qui est passé (autrement dit ce qui est perdu si nous ne l'avons pas enregistré) et ce

qui est futur, c'est à dire indisponible à la volonté et révélé à la seule patience. Ce serait probablement une erreur de considérer le temps comme une simple dimension supplémentaire aux autres (comme, par exemple, la fréquence ou la position), car celui-ci introduit très souvent une redondance dans l'information, voire une variabilité intra-classe (*i.e.* une composante non-discriminante, perturbant la bonne distinction des formes). Si dans le domaine de la sélection de descripteurs, on cherche à se défausser des dimensions qui ne sont pas porteuses d'un pouvoir de discrimination, au contraire, la prise en compte du temps fournit très souvent un gain de puissance discriminante (par exemple dans la comparaison de signaux à travers une représentation TF plutôt que stationnaire) mais uniquement dans une certaine mesure. Au delà de celle-ci, il est plus néfaste qu'informatif. C'est cette ambivalence qui confère au temps la distinction dont il est question ici et qui conduit à chercher un compromis entre la discrimination et l'invariance temporelle des caractéristiques d'un signal.

Comme nous l'avons laissé transparaître, la résolution temporelle est importante pour régler la puissance discriminante des descripteurs. Cependant, cette notion de temps apparaît aussi à l'échelle de la prise de décision. Dans l'exemple de reconnaissance que nous avons donné, la décision concernant une séquence est prise suite au vote de sous-parties de ladite séquence. Autrement dit, la décision globale et une fusion des classes prédites pour chaque sous-partie d'une séquence. Ce processus démocratique fonctionne donc à partir d'une information en quantité (le nombre de votants pour chaque candidat) mais pas en qualité (*i.e.* en termes d'assurance et de pertinence du votant). Il est toutefois souvent possible d'associer une mesure de qualité à un vote (par exemple sous la forme d'une probabilité), mais celle-ci constituerait une information très synthétique vivant dans un espace à une unique dimension. Ceci est la manifestation d'une limitation du traitement démocratique, qui nous conduit à chercher une fusion de l'information plus en amont, là où celle-ci n'a pas encore été réduite à la simple décision d'une classe. Par exemple, cette fusion peut être conduite à l'échelle des vecteurs caractéristiques, là où la qualité discriminante est plus présente. Ce traitement séquentiel du temps ouvre la voie à de nouvelles thématiques, notamment la gestion de la redondance de l'information, ou autrement dit, d'une prise de décision au plus tôt, qui ne nécessite pas d'analyser une séquence dans sa totalité afin d'émettre un avis. En allant encore plus loin, ce concept de reconnaissance précoce conduit au compromis entre la latence d'observation (et inversement la précocité) et la discrimination : réduire l'observation d'une séquence au delà du seuil de redondance conduit nécessairement à dégrader les performances de reconnaissance (puisque la décision est prise avec une information partielle) mais autorise à catégoriser la séquence dans les plus brefs délais.

Comme nous l'apercevons, l'apprentissage automatique, *a fortiori* appliqué au traitement de signaux temporels, est un mélange complexe de compromis entre la discrimination, les *a priori*, l'invariance et la précocité.

CONTRIBUTIONS

Les travaux que nous décrivons dans ce manuscrit ont suivi un approfondissement progressif de la notion de traitement des signaux, vu sous l'angle de l'apprentissage automatique. Ils se situent donc à la frontière de ces deux domaines et au cœur des thématiques d'une communauté particulièrement active. Le premier problème traité, et le plus apparent, est celui du choix (automatique) de descripteurs pour la reconnaissance de signaux. Il nous a paru judicieux de chercher de tels descripteurs dans le plan TF. Ainsi, est apparu rapidement l'ambivalence du temps, opposant la discrimination à l'invariance des descripteurs. Cette dualité est aussi une porte ouverte vers de nouvelles problématiques, moins populaires pour le moment, d'analyse de séquences partiellement observées. Cette thématique

fait l'objet d'un deuxième chapitre de recherche.

Apprentissage de descripteurs

L'extraction automatique de caractéristiques, rendue possible grâce aux profonds développements en informatique, mathématiques appliqués et statistiques, est apparue en réponse à deux écueils rencontrés jusqu'alors. Il y a d'abord celui de l'optimalité des modèles mis au point pour synthétiser des descripteurs de signaux (pensons à l'approche psycho-acoustiques aboutissant aux MFCC) : sont-ils les plus performants pour une tâche prédéfinie et étant donnée l'information disponible ? Puis l'incapacité à modéliser un phénomène pour en extraire des caractéristiques pertinentes. Concrètement, l'apprentissage de descripteurs est né dans les années 1980 avec les premiers travaux sur les réseaux de neurones artificiels. Cette application informatique de concepts biologiques nous a fait prendre conscience que notre incapacité à modéliser n'est pas un problème insurmontable. Paradoxalement, un système au fonctionnement aussi obscur que notre cerveau mais adaptatif offre une porte de sortie élégante. Les réseaux de neurones artificiels ont ouvert la voie à un vaste champ de recherche au sein duquel, le domaine spécifique des signaux temporels a vu apparaître des méthodes d'extraction automatique de caractéristiques adaptant une transformée de Cohen à une tâche de reconnaissance ou, plus récemment, construisant automatiquement un dictionnaire d'atomes favorisant la discrimination entre des groupes de signaux.

Parmi ces développements, peu de place a été laissée aux approches convolutives traditionnelles, pourtant très utilisées pour analyser les signaux au cas par cas (transformée de Fourier, spectrogramme), compresser ou débruiter des données. En pratique, les représentations convolutives utilisées dans ces applications sont très souvent modélisées par un Banc de Filtres (BdF), outil particulièrement adapté et connu en traitement du signal. Ainsi, pour répondre à notre premier objectif d'apprentissage automatique de descripteurs, nous présentons un nouvel algorithme permettant d'inférer, à partir d'un ensemble de signaux observés, un BdF discriminant. Celui-ci est discriminant au sens où les représentations TF obtenues par son application rendent aisée la distinction des signaux appartenant à différents groupes pré-définis. Cependant, dans notre contexte de travail, les signaux ne sont pas directement comparés à travers leurs représentations TF mais par une agrégation de celles-ci, permettant de régler le compromis entre la discrimination et l'invariance des descripteurs. L'algorithme que nous proposons permet aussi de sélectionner cette *fonction d'agrégation* afin de favoriser la reconnaissance des signaux.

Cette technique d'extraction automatique de caractéristiques a été mise en œuvre de concert à une méthode de classification à noyaux, nommée machine à vecteurs supports (*Support Vector Machine*, SVM). Celle-ci constitue un moyen simple, possédant un fort soutien théorique, pour implémenter des outils de reconnaissance non-linéaires. Avec celle-ci, l'extraction automatique de descripteurs est associée à un domaine de recherche très actif depuis les années 2000 : l'apprentissage de noyau. D'un point de vue géométrique, le noyau peut être assimilé à la mesure de similarité utilisée pour comparer les données à classer. Par conséquent, dans ce cadre (plus lié à l'apprentissage automatique qu'au traitement du signal), notre algorithme constitue une extension des travaux, réalisés lors de cette dernière décennie, sur l'apprentissage du noyau. Il permet de combiner, de manière non-linéaire, un ensemble de noyaux choisis parmi une infinité pour leur puissance de discrimination. Dans nos travaux, chaque noyau est associé à l'un des filtres d'un BdF discriminant.

Détection précoce

La technique introduite ci-dessus compare des signaux dans un plan TF dont la résolution temporelle a été astucieusement déterminée. Elle suppose implicitement une structure temporelle partagée par les signaux d'une même classe. Au contraire, lorsque les signaux n'ont pas de structure commune ou que l'information est présente avec redondance, il n'est pas nécessaire d'analyser une série temporelle dans sa totalité pour prendre une décision concernant sa classe. On parle alors alors de reconnaissance précoce. Ce champ de recherche a été peu exploré jusqu'alors, mais bénéficie déjà de quelques travaux solides fondés sur des règles de décision par plus proches voisins, probabilistes et SVM.

Un second chapitre de recherche contribue à l'évolution de ce domaine en proposant un nouveau cadre rendant possible la détection précoce d'événements au sein d'une série temporelle. La détection consiste ici à indiquer, à tout instant, si un événement particulier est en train de se dérouler ou non. L'approche que nous proposons, à la croisée de l'apprentissage d'instances multiples (*Multiple Instance Learning*, MIL) et des espaces de proximité, consiste à détecter une séquence comme événement lorsque celle-ci contient un *instantané* discriminant. Réalisée dans le cadre des méthodes à noyaux, elle se différencie des approches existantes de détection précoce en s'abstenant d'augmenter virtuellement l'ensemble d'apprentissage. En pratique, cet artifice est habituellement réalisé en considérant des observations partielles des séquences d'entraînement, conduisant inéluctablement à des problèmes mathématiques plus complexes à résoudre. Au contraire, l'approche que nous proposons aboutit à un cadre simple à mettre en œuvre et à un problème d'apprentissage facile à résoudre. En outre, il bénéficie du même soutien théorique que celui développé dans la théorie SVM.

Organisation du manuscrit

Structuré en quatre chapitres, ce manuscrit suit les deux domaines fondateurs (apprentissage automatique et traitement du signal) ainsi que les deux contributions mentionnées ci-avant (apprentissage de descripteurs et détection précoce) en plaçant successivement les uns et les autres comme axes principal du discours. Les deux premiers chapitres synthétisent les éléments fondateurs d'apprentissage et de traitement du signal (en insistant sur l'extraction automatique de caractéristiques et sur la reconnaissance précoce), tandis que le deux derniers présentent nos contributions, situées à l'intersection des deux domaines de recherche.

Apprentissage automatique

Tous les éléments présentés dans ce manuscrit découlent de fondements théoriques, formant dans leur ensemble ce que l'on appelle l'*apprentissage statistique*. Celui-ci formalise le compromis entre la diversité d'un modèle et ses performances empiriques. Autrement dit, la difficulté à construire un outil possédant de bonnes capacités de généralisation (*i.e.* une faible erreur de prédiction sur des données inédites) à partir d'un seul jeu d'observations.

L'apprentissage statistique est illustré à travers un outil qui soutient l'ensemble de nos travaux : la SVM. Nous la présentons sous trois facettes : d'abord de manière fonctionnelle (en lien direct avec l'apprentissage statistique), puis numérique (ouvrant la porte à différentes variantes du paradigme SVM) et enfin géométrique (illustrant le principe de fonctionnement sous-jacent). Par *apprentissage automatique*, nous faisons référence à l'aspect numérique des outils d'apprentissage, *i.e.* à leur mise en œuvre plutôt qu'aux éléments théoriques.

Une SVM est un outil indépendant de la distribution statistique des données. En contrepartie, elle possède différents paramètres qu'il n'est pas aisé de sélectionner : l'un est un réel qui contrôle la complexité de la classe des prédicteurs, l'autre est le noyau définissant cette classe. Les approches les plus communes de sélection de ces paramètres sont donc rappelées. Parmi elles, l'apprentissage de noyau multiple, qui est au cœur de notre première contribution, tient une place particulière.

Enfin, nous rappelons le concept MIL, formalisant quelques moyens pour lever une forme particulière d'ambiguïté dans l'étiquetage des données. Notre attention se porte sur ceux dérivés du paradigme SVM et dont l'un est en lien direct avec notre deuxième contribution.

Reconnaissance de signaux

La reconnaissance de signaux repose nécessairement sur des descripteurs. Il en existe un grand nombre, sanctionnant plusieurs dizaines d'années d'expertise. Certains sont stationnaires, d'autres non, caractérisant ainsi la fluctuation des spécificités d'un signal au fil du temps.

Parmi les descripteurs non-stationnaires se trouvent les représentations TF ou temps-échelles. Celles-ci sont des exemples de caractéristiques qui mettent en jeu le compromis entre discrimination et invariance temporelle, et qui nécessitent en conséquence d'être agrégées avant d'être effectivement utilisées pour une tâche de reconnaissance. Il est donc utile de rappeler les techniques usuelles d'agrégation.

Nous passons ensuite en revue les grandes transformées Temps-Caractéristique (TC) qui ont fait naître des techniques d'adaptation automatique à un ensemble de signaux d'apprentissage, avec pour but de mettre en lumière des caractéristiques discriminantes. Ce sont la transformée de Cohen, le BdF, le réseau de neurones convolutifs (*Convolutional Neural Network*, CNN), la décomposition en ondelettes (comprenant la diffusion) et le dictionnaire d'atomes.

Enfin sont rappelées les principales approches de reconnaissance précoce. Celles-ci couvrent plusieurs familles d'approches, en particulier probabiliste et SVM.

Apprentissage d'une représentation TF convolutive

Notre première contribution concerne l'extraction automatique de caractéristiques à travers l'apprentissage d'un BdF discriminant. La détermination dirigée par les données du BdF est donc d'abord formalisée comme un problème d'apprentissage de noyau, supposant que le classifieur associé est une SVM. Le chapitre 1 a introduit plusieurs critères de sélection de modèle. Pour cette formalisation, nous choisissons le plus évident : le risque empirique régularisé.

Avec peu d'hypothèses sur les filtres constituant le banc, le problème d'apprentissage d'un BdF discriminant peut être réduit soit à un programme semi-défini (*Semi-Definite Program*, SDP), que l'on peut résoudre par des logiciels en libre utilisation, soit à la minimisation d'une fonction de coût dérivable (mais non-convexe) sur la boule unité de la norme ℓ_2. Dans ce dernier cas, un minimum local peut être atteint par descente de gradient projeté.

Dans un cadre plus complexe, introduisant une forme de régularisation des filtres, nous montrons que le problème d'apprentissage se réduit à celui d'une combinaison (non-nécessairement linéaire) de noyaux, choisis parmi une infinité. De manière similaire, la fonction d'agrégation peut être déterminée comme la combinaison de plusieurs fonctions. Un algorithme par ensemble actif est alors mis au point afin de fournir une solution locale à ce

problème.

Notre approche par noyau multiple est finalement validée sur des données synthétiques, d'Interface Cerveau-Machine (ICM) et audio. Comparée à des techniques basiques (MFCC et SVM) et avancées (CNN), notre méthode fournit un gain de généralisation.

Un modèle de détecteur précoce

Un nouveau cadre méthodologique pour la détection précoce d'événements au sein d'une série temporelle est présenté dans ce chapitre. Ce cadre est construit sur la notion de similarité entre une séquence et un ensemble d'instantanés. Nous énonçons alors des conditions suffisantes pour que le détecteur cherché soit *fiable*, *i.e.* qu'il ne revienne pas sur sa décision une fois un avis de détection donné au cours de l'analyse d'une séquence. Cette condition est essentielle pour prendre une décision finale au plus tôt.

L'apprentissage d'un tel détecteur simultanément à la sélection d'instantanés pertinents est aisément réalisable grâce à une SVM modifiée, pénalisée par une norme ℓ_1. Il apparaît qu'un tel problème est linéaire et efficacement soluble par un algorithme de contraintes actives. De plus, il est possible de déduire une borne de généralisation des similitudes de notre approche avec le paradigme SVM.

Des expériences numériques conduites sur un jeu de séquences temporelles synthétiques, ainsi que sur des données audio et vidéos valident empiriquement l'intérêt de notre approche quant à la justesse de reconnaissance ainsi que sa capacité à prendre une décision à partir d'une observation partielle.

PUBLICATIONS

Le travail préparatoire à cette thèse de doctorat a fait l'objet de plusieurs publications, dont certaines ne sont pas détaillées dans ce manuscrit, par soucis de clarté du discours.

Revue internationale avec comité de lecture

SANGNIER, M., GAUTHIER, J. et RAKOTOMAMONJY, A. (2015). Filter bank learning for signal classification. *Signal Processing*. [Chapitre 3]

Conférences internationales avec comité de lecture

SANGNIER, M., GAUTHIER, J. et RAKOTOMAMONJY, A. (2014). Kernel learning as minimization of the single validation estimate. *Dans les actes de IEEE International Workshop on Machine Learning for Signal Processing*.

SANGNIER, M., GAUTHIER, J. et RAKOTOMAMONJY, A. (2013). Filter bank kernel learning for nonstationary signal classification. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing*. [Chapitre 3]

Conférences nationales avec comité de lecture

SANGNIER, M., GAUTHIER, J. et RAKOTOMAMONJY, A. (2013). Apprentissage de représentations temps-fréquence adaptées à la classification de signaux par optimisation semi-définie positive. *Dans les actes de Colloque du Groupe d'Études du Traitement du Signal et des Images*. [Chapitre 3]

BARTHÉLEMY, Q., SANGNIER, M., LARUE, A. et MARS, J. (2013). Comparaison de descripteurs pour la classification de décompositions parcimonieuses invariantes par translation. *Dans les actes de Colloque du Groupe d'Études du Traitement du Signal et des Images*.

Travaux en cours

SANGNIER, M., GAUTHIER, J. et RAKOTOMAMONJY, A. (2014). Simple early detector of temporal event. [Chapitre 4]

CHAPITRE 1

Apprentissage automatique

1.1 INTRODUCTION

L'apprentissage automatique (*Machine learning*) est une discipline vouée à mettre en place des théories et des algorithmes permettant à une machine d'apprendre automatiquement des *règles d'analyse et de décision*. Ces règles automatiques couvrent les besoins en traitement des données tels que la représentation, l'extraction d'information et la reconnaissance automatique. De manière moins formelle, l'apprentissage automatique consiste à transférer un défaut de connaissance *a priori* de l'expert vers une machine (un ordinateur). Celle-ci, plus à même qu'un Homme d'explorer un vaste ensemble d'hypothèses, doit alors construire, par elle-même, un outil répondant au besoin de l'expert. En ce sens, l'apprentissage automatique répond à l'impossibilité de modéliser un phénomène pour en distiller l'information utile. L'ordinateur, de concert avec les algorithmes et les théories d'apprentissage automatique, se présente comme un moyen alternatif d'arriver au but recherché, en remplaçant la capacité d'abstraction et de réflexion de l'Homme par l'exploration systématique d'un espace d'hypothèses.

Une grande partie des outils théoriques d'apprentissage automatique ont été mis en lumière par Vapnik, sous le nom d'apprentissage statistique [Vapnik, 1995]. La section 1.2 de ce chapitre rapporte quelques éléments de cette théorie en insistant sur son ambivalence : le compromis inatteignable entre l'efficacité d'une règle inférée à partir d'observations du monde réel et l'hypothétique capacité de celle-ci à se comporter au mieux dans le futur.

Nos travaux se concentrent essentiellement sur les règles de décision, puisque nous cherchons à mettre en place des algorithmes de reconnaissance automatique de signaux. Ainsi, nous introduisons en section 1.3 l'un des outils les plus célèbres d'apprentissage statistique : la machine à vecteurs supports (*Support Vector Machine*, SVM), introduite par Boser, Guyon et Vapnik [Boser *et coll.*, 1992] et capable d'apprendre automatiquement une règle de décision. Nous commencerons par une approche fonctionnelle du concept SVM, en définissant clairement l'espace des hypothèses parcouru. Cette approche est en continuité avec la théorie d'apprentissage statistique évoquée précédemment. Ensuite, nous glissons vers une vision plus commune de la SVM (particulièrement dans le milieu informatique), sous l'angle de la régularisation de Tikhonov. Enfin, nous présentons une interprétation géométrique classique de ce concept : la détermination d'un hyperplan maximisant la marge entre deux ensembles de points.

Cette technique d'apprentissage statistique qu'est la SVM, et qui est au cœur de nos travaux, requiert cependant le choix préalable de valeurs à accorder à quelques paramètres. Dans certaines approches (par exemple bayésiennes et neuronales), cette liberté accordée à l'expert est un moyen d'incorporer le peu de connaissance *a priori* que celui-ci possède. Malheureusement, il est difficile d'accorder une interprétation à ces paramètres dans le cas d'une SVM, faisant de cette liberté une charge plutôt qu'un atout. Des techniques de sélection automatique de modèle ont donc été mises en place pour contourner cette difficulté. La section 1.4 est majoritairement destinée à tracer la ligne principale de recherche de l'une de ces techniques : l'apprentissage de noyau multiple. Une fois de plus, cette notion occupe une place centrale dans nos travaux, puisqu'elle offre un cadre théorique et pratique pour apprendre une représentation des données d'entrées dans laquelle la mise en place d'une règle de décision est plus aisée.

Nous abordons finalement en section 1.5 un cadre d'apprentissage automatique différent de celui usuellement adopté en apprentissage statistique. Ce cadre formalise la prise en compte de l'ambiguïté d'étiquetage rencontrée systématiquement dans certaines applications telles que la reconnaissance d'images et de signaux. Dans la continuité de l'exposé présenté dans ce chapitre, nous nous concentrons sur les algorithmes de résolution liée au paradigme SVM.

1.2 ÉLÉMENTS D'APPRENTISSAGE STATISTIQUE

L'apprentissage statistique est une discipline des mathématiques appliquées à la frontière de quatre domaines : les statistiques, l'analyse fonctionnelle, l'optimisation et l'informatique. Elle regroupe un ensemble de méthodes visant à modéliser un phénomène physique à partir d'observations de celui-ci et des moyens calculatoires actuels, de la manière la plus directe possible. Les statistiques incarnent le fondement de cette discipline, par le cadre théorique qu'elles fournissent, permettant ainsi de *généraliser* des propriétés inférées des observations antérieures à toute observation inédite. Les modèles utilisés pour décrire le phénomène physique d'intérêt tiennent leurs origines de l'analyse fonctionnelle et sont souvent (mais pas nécessairement) déterminés par la résolution d'un problème d'optimisation (*i.e.* un problème consistant à déterminer les minima d'une fonction d'énergie, plus couramment appelée fonction de coût), mettant en jeu tout un panel d'algorithmes et bien entendu, des systèmes informatiques adéquats. Il est un principe important en apprentissage statistique (que l'on peut résumer par *utiliser la manière la plus directe*) : il est sage d'éviter toute étape intermédiaire entre les données et le but à atteindre (la modélisation du phénomène) car il y a fort à parier que les marches intermédiaires soient individuellement plus difficiles à franchir que le but recherché lui-même. Ainsi, l'apprentissage statistique se place *de facto* en opposition aux approches bayésiennes qui cherchent systématiquement à capturer le mécanisme de génération des observations, peu importe le but recherché. En pratique, il est souvent plus difficile d'accéder à une telle information qu'à une représentation du phénomène d'intérêt.

1.2.1 Formalisme

Les observations que nous avons mentionnées auparavant sont des vecteurs caractéristiques $\boldsymbol{x}$, regroupant des descripteurs appelés *variables explicatives*. Il est alors d'usage de distinguer deux branches de l'apprentissage statistique : l'apprentissage supervisé et non-supervisé. Dans ce dernier, les observations sont au centre des débats et l'on va, par exemple, chercher à mettre en place des techniques de séparation aveugle (Analyse en Composantes Principales (ACP), analyse en composantes indépendantes, factorisation de matrices, *etc.*)

et de création automatique de groupes (*clustering*). En apprentissage supervisé, chaque observation $\boldsymbol{x}$ est accompagnée d'une étiquette y (elle aussi observée), aussi appelée variable expliquée. De manière plus rigoureuse, une observation est un couple $(\boldsymbol{x}, y)$ dont la première partie sert à expliquer la deuxième. La finalité de l'apprentissage supervisé est, à partir d'observations étiquetées, d'inférer une règle f donnant l'étiquette y associée à une observation inédite $\boldsymbol{x}$; autrement dit, d'établir un lien de cause à effet entre les deux entitées : $y = f(\boldsymbol{x})$.

Suivant la nature de l'étiquette y, on distingue trois familles d'approches :

- ◇ la ***régression*** : les étiquettes sont prises dans $\mathbb{R}$;
- ◇ la ***classification multi-classe*** : les étiquettes proviennent de $[\![1, K]\!]$ (K étant un entier au moins égale à 3) ;
- ◇ la ***classification binaire*** : les étiquettes sont dans $\{-1, 1\}$, abrégé $\{\pm 1\}$. Il est équivalent de concevoir les étiquettes dans $\{1, 2\}$ mais la notation précédente simplifie les expressions mathématiques.

Pour la suite de ce manuscrit (et conformément à nos travaux), nous nous placerons dans le cadre de l'apprentissage supervisé et nous nous concentrerons sur des problèmes de classification binaire. Une grande partie de ce qui est écrit dans cet chapitre (concernant l'apprentissage automatique) peut être naturellement étendue à la régression et (de manière moins évidente) à la classification multi-classe. En revanche, nous ne traiterons aucunement d'apprentissage statistique non-supervisé.

Définition 1.2.1 (Espace des observations)**.**
Soient d un entier non-nul (dans la suite du manuscrit, sa valeur dépendra du contexte) et $\mathcal{X}$ une partie compacte (*i.e.* un ensemble fermé borné) de $\mathbb{K}^d$. On appelle espace des observations, l'espace probabilisé $\left(\mathcal{X} \times \{\pm 1\}, 2^{\mathcal{X}\times\{\pm 1\}}, \mathbb{P}\right)$. Par analogie à l'apprentissage non-supervisé et par concision, on appellera aussi simplement $\mathcal{X}$ l'espace des observations.

Soient $\mathcal{X}$ l'espace des observations et $\mathfrak{D}$ une distribution de probabilités sur $\mathcal{X} \times \{\pm 1\}$ (de densité de probabilité $p_{\mathfrak{D}}$). L'apprentissage statistique supervisé a pour vocation la détermination d'un lien $f\colon \mathcal{X} \to \{\pm 1\}$ entre deux variables aléatoires X et Y (définies de sorte que (X, Y) admette $\mathfrak{D}$ comme loi conjointe), tel que $Y = f(X)$. Formellement, on se donne une classe de fonctions $\mathcal{F}$ et on cherche l'ensemble des fonctions minimisant l'erreur réelle :

$$\arg\min_{f\in\mathcal{F}} \mathbb{P}(Y \neq f(X)),$$

où la probabilité précédente est définie à partir des crochets d'Iverson (retournant 1 si la condition encadrée est vraie et 0 sinon)

$$\mathbb{P}(Y \neq f(X)) = \iint_{\mathcal{X}\times\{\pm 1\}} [y \neq f(\boldsymbol{x})]\, p_{\mathfrak{D}}(\boldsymbol{x}, y)\, \boldsymbol{dx}\, dy.$$

1.2.2 Approche bayésienne

Dans un cadre bayésien (*i.e.* lorsque la distribution de probabilité *a posteriori* $\mathbb{P}(X|Y)$ est connue) et avec $\mathcal{F} = \{\pm 1\}^{\mathcal{X}}$, il est possible de montrer que le minimum de $\mathbb{P}(Y \neq f(X))$ est (notamment) atteint pour le log-ratio des probabilités :

$$f\colon \boldsymbol{x} \in \mathcal{X} \mapsto \mathrm{Signe}(\log(\mathbb{P}(\boldsymbol{x}|+1)) - \log(\mathbb{P}(\boldsymbol{x}|-1))).$$

À titre d'exemple, supposons que les données de la classe $+1$ sont distribuées suivant une loi normale multivariée $\mathcal{N}(\boldsymbol{\mu_{+1}}, \boldsymbol{\Sigma}_+)$, de matrice de covariance $\boldsymbol{\Sigma}_+$ (de dimension d) et

d'espérance $\boldsymbol{\mu_{+1}}$, *i.e.* :

$$\mathbb{P}(\boldsymbol{x}|+1) = \frac{1}{(2\pi)^{\frac{d}{2}}|\boldsymbol{\Sigma_+}|^{\frac{1}{2}}} \mathrm{e}^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu_{+1}})^T \boldsymbol{\Sigma_+}^{-1}(\boldsymbol{x}-\boldsymbol{\mu_{+1}})}.$$

Si maintenant les données de la classe -1 sont distribuées suivant une loi normale translatée $\mathcal{N}(\boldsymbol{\mu_{-1}}, \boldsymbol{\Sigma_+})$, alors un classifieur de Bayes est donné par le signe d'une application linéaire :

$$f : \boldsymbol{x} \in \mathcal{X} \mapsto \mathrm{Signe}\left(\langle \mathbf{w} \mid \boldsymbol{x} \rangle_{\ell_2} + b\right),$$

avec

$$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu_+} - \boldsymbol{\mu_-}), \quad b = \frac{1}{2}\boldsymbol{\mu_-}^T\Sigma^{-1}\boldsymbol{\mu_-} - \frac{1}{2}\boldsymbol{\mu_+}^T\Sigma^{-1}\boldsymbol{\mu_+}.$$

Ce résultat correspond à la technique de classification nommée analyse linéaire discriminante (*Linear Discriminant Analysis*, LDA). Remarquons que les hypothèses sont particulièrement fortes puisque l'on suppose disposer de deux classes distribuées de manière gaussienne et uniquement translatées l'une par rapport à l'autre. En revanche, quand cette dernière hypothèse est relâchée (les deux lois normales sont définies par des matrices de covariance différentes), on obtient une fonction discriminante quadratique.

1.2.3 Approche fréquentiste

Lorsque les distributions de probabilité sont inconnues, le cadre bayésien est plus difficile à mettre en place et l'apprentissage statistique vient pallier ce manque avec un postulat important : il est de bon sens de déterminer directement un lien $f : \mathcal{X} \to \{\pm 1\}$ sans passer par l'étape intermédiaire (et très probablement difficile car plus générale) de l'estimation des distributions de probabilité régissant le phénomène physique. La difficulté dans l'estimation du phénomène génératif sous-jacent provient généralement de la quantité limitée d'information à disposition (les observations), comparée à la dimension de $\mathcal{X}$.

Ainsi, en pratique l'erreur réelle n'est pas calculable. On travaille donc avec une erreur empirique, calculée sur un jeu d'observations. Ce dernier est couramment appelé *ensemble d'apprentissage*.

Définition 1.2.2 (Ensemble d'apprentissage)**.**
Soient n un entier non-nul et (X, Y) un couple de variables aléatoires distribué selon $\mathfrak{D}$. On appelle ensemble d'apprentissage (de taille n), et on note $\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}$, un échantillon de n réalisations de la variable aléatoire (X, Y), tirées indépendamment. C'est observations sont dites indépendantes et identiquement distribuées (iid).

La probabilité empirique d'erreur $\hat{\mathbb{P}}_n(Y \neq f(X))$ peut alors être définie à partir de l'ensemble d'apprentissage $\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}$ à disposition par :

$$\hat{\mathbb{P}}_n(Y \neq f(X)) = \frac{1}{n}\sum_{i=1}^{n} \left[y_i \neq f(\boldsymbol{x_i})\right].$$

La difficulté inhérente à l'apprentissage statistique est d'établir un lien théorique entre l'erreur empirique (celle que l'on constate) est l'erreur réelle (celle qui a un intérêt en pratique). En particulier, on cherche à quantifier l'aptitude à diminuer l'erreur réelle à partir de la minimisation de l'erreur empirique, nommée *capacité de généralisation*. Les premières garanties théoriques de généralisation concernant l'apprentissage statistique reviennent à Vapnik et Chervonenkis et datent des années 1970 [Vapnik, 1998]. Ici, nous énonçons une amélioration de ces résultats due à Bartlett et Mendelson.

Théorème 1.2.1 (Borne de généralisation [Bartlett et Mendelson, 2002])**.**
Soient $\mathcal{F}$ une classe de fonctions de $\mathcal{X}$ dans $\{\pm 1\}$, (X, Y) un couple de variables aléatoires suivant $\mathfrak{D}$ et $\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}$ un ensemble d'apprentissage (iid) issus de (X, Y). Il existe une constante positive c telle que pour toute taille d'échantillon n et avec probabilité $1-\delta$ par rapport à n (i.e. *$\delta \leq 1$ suffisamment grand relativement à n^{-1}) :*

$$\forall f \in \mathcal{F} \colon \mathbb{P}(Y \neq f(X)) \leq \hat{\mathbb{P}}_n(Y \neq f(X)) + c\sqrt{\frac{\mathrm{dimVC}(\mathcal{F})}{n}},$$

où $\mathrm{dimVC}(\mathcal{F})$ est la dimension de Vapnik-Chervonenkis de $\mathcal{F}$ [Vapnik, 1998].

La dimension de Vapnik-Chervonenkis est un indicateur de la complexité de l'espace considéré (aussi appelée *capacité* de l'espace). De manière informelle, la capacité $\mathrm{dimVC}(\mathcal{F})$ peut être définie comme le nombre maximal d'observations $\{\boldsymbol{x_i}\}_{1\leq i\leq n}$ pour lesquelles, quelles que soient les étiquettes $\{y_i\}_{1\leq i\leq n}$ de $\{\pm 1\}^n$ qui leur sont attribuées, on peut trouver un estimateur f dans $\mathcal{F}$ réalisant une séparation parfaite (*i.e.* d'erreur empirique nulle).

Ainsi, le théorème 1.2.1 met en évidence l'ambivalence de l'apprentissage statistique, pouvant être décrit comme le compromis à réaliser (afin de réduire l'erreur réelle) entre deux entités évoluant de manière résolument opposée : l'erreur empirique et la complexité de l'espace des hypothèses.

Remarque 1.
On déduit du théorème précédent, deux informations :

- ◇ en déterminant une fonction f qui réduit l'erreur empirique, on réduit (très probablement) l'erreur réelle ;
- ◇ lorsque la taille de l'échantillon grandit, la borne s'affine, *i.e.* réduire l'erreur empirique revient à réduire l'erreur réelle.

Si l'on fixe la classe de fonctions $\mathcal{F}$ (et ainsi la complexité de l'espace des hypothèses) et l'échantillon d'apprentissage, alors le seul paramètre libre pour réduire la borne sur l'erreur réelle donnée dans le théorème 1.2.1 est la fonction de discrimination f. Ainsi, un *bon* estimateur (*i.e.* réduisant la probabilité d'erreur) est une fonction f de $\mathcal{F}$ réduisant autant que possible l'erreur empirique $\hat{\mathbb{P}}_n(Y \neq f(X))$. En pratique, une telle fonction est difficile à déterminer car l'erreur empirique ainsi définie est discontinue, tout comme $f \colon \mathcal{X} \to \{\pm 1\}$ (le problème est combinatoire). Pour faciliter la recherche, on étend l'image de $\mathcal{X}$ par f à $\mathbb{R}$ ($\mathcal{F} \subset \mathbb{R}^{\mathcal{X}}$) et on utilise une *fonction de perte* $L \colon \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ qui mesure l'écart entre l'étiquette attendue y (faisant partie d'un espace discontinu) et la valeur prédite $f(\boldsymbol{x})$ (faisant partie d'un espace continu). La décision de classification finale correspond alors au signe de la fonction de prédiction f. Le paradigme de l'apprentissage statistique peut ainsi se résumer à la détermination d'une fonction f^* minimisant le risque empirique :

$$f^* \in \operatorname*{arg\,min}_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\boldsymbol{x_i})).$$

Dans ce qui suit, nous donnons une borne d'erreur (provenant de [Bartlett et Mendelson, 2002]), similaire à celle du théorème 1.2.1, lorsqu'une fonction de perte L est utilisée. Pour ce faire, commençons par définir un indicateur alternatif à la dimension de Vapnik-Chervonenkis pour mesurer la complexité d'un espace, nommée *complexité de Rademacher* [1].

Définition 1.2.3 (Complexité de Rademacher [Kakade *et coll.*, 2009])**.**
Soient $\mathcal{F}$ une classe de fonctions de $\mathcal{X}$ dans $\mathbb{R}$, $\mathfrak{D}'$ une densité de probabilité sur $\mathcal{X}$, X

1. La définition donnée ici est légèrement différente de celle issue des travaux fondateurs [Bartlett et Mendelson, 2002] mais est cohérente avec les résultats présentés dans le manuscrit.

une variable aléatoire suivant $\mathfrak{D}'$ et $(R_i)_{1\leq i\leq n}$ des variables aléatoires iid uniformément distribuées sur $\{\pm 1\}$ (dites variables de Rademacher). Étant donné un échantillon $(\boldsymbol{x_i})_{1\leq i\leq n}$ d'observations iid de X, la complexité de Rademacher (empirique) est définie par :

$$\mathcal{R}_n(\mathcal{F}) = \mathbb{E}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} R_i f(\boldsymbol{x_i})\right].$$

La complexité de Rademacher $\mathcal{R}_n(\mathcal{F})$ d'un ensemble $\mathcal{F}$ quantifie la corrélation des fonctions de $\mathcal{F}$ avec une séquence de bruit de taille n. C'est une manière moins intuitive que la dimension de Vapnik-Chervonenkis de définir la capacité d'une classe de fonctions, mais qui mène plus aisément à des bornes de généralisation. En particulier, le théorème suivant, énoncé par Bartlett et Mendelson puis reformulé par Kakade *et coll.*, borne le risque réel grâce au risque empirique et à la complexité de Rademacher.

Théorème 1.2.2 (Borne de généralisation [Bartlett et Mendelson, 2002, théorème 8], [Kakade *et coll.*, 2009])**.**
Soient $\mathcal{F}$ une classe de fonctions de $\mathcal{X}$ dans $\{\pm 1\}$, (X, Y) un couple de variables aléatoires suivant $\mathfrak{D}$ et $\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}$ un ensemble d'apprentissage (iid) issus de (X, Y). Soit $L\colon \mathbb{R}\times\mathbb{R}\to\mathbb{R}_+$ une fonction de perte k_L-lipschitzienne par rapport à son deuxième argument et bornée par un certain c. Pour tout $\delta\in]0,1]$ et avec probabilité $1-\delta$:

$$\forall f\in\mathcal{F}\colon \mathbb{E}\left[L(Y, f(X))\right] \leq \hat{\mathbb{E}}_n\left[L(Y, f(X))\right] + 2k_L\mathcal{R}_n(\mathcal{F}) + c\sqrt{\frac{\ln(1/\delta)}{2n}},$$

où $\hat{\mathbb{E}}_n\left[L(Y, f(X))\right] = \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\boldsymbol{x_i}))$.

Contrairement au théorème 1.2.1 qui illustre principalement l'ambivalence de l'apprentissage statistique, le théorème 1.2.2 est applicable en pratique. On le retrouvera notamment dans les sections 1.4 et 4.3.

Pour le moment, il clos l'ensemble des éléments théoriques d'apprentissage statistique essentiels à la compréhension de la suite de ce manuscrit. Dans ce qui vient à présent, nous faisons un léger détour sur les grandes notions d'optimisation numérique, qui sous-tendent la détermination d'un estimateur f minimisant le risque empirique.

1.2.4 Optimisation et convexité

Souvent (sans pour autant que ce soit une nécessité), la détermination d'un bon estimateur f passe par la résolution d'un problème d'optimisation mathématique. Par définition, un problème d'optimisation s'écrit sous la forme :

$$\begin{array}{ll} \underset{\mathbf{x}\in\mathcal{O}}{\text{minimiser}} & J(\mathbf{x}) \\ \text{tel que} & \mathbf{x}\in\mathcal{C}, \end{array}$$

où $\mathcal{O}$ est un ensemble d'objets, $J\colon \mathcal{O}\to\mathbb{R}$ est une fonction de coût et $\mathcal{C}$ est l'ensemble des contraintes ($\mathcal{C}\subset\mathcal{O}$). L'ensemble $\mathcal{O}$ peut prendre différents visages. Lorsque la résolution du problème d'optimisation ne peut être que numérique, on se ramène alors à un espace vectoriel réel de dimension finie : $\mathcal{O} = \mathbb{R}^d$.

Le problème posé précédemment réside dans la détermination d'un point $\mathbf{x}$ de $\mathcal{C}$ où est atteint le minimum de la fonction J. À défaut, certaines applications ne s'intéressent qu'au minimum en lui-même, sans exprimer le besoin de déterminer un antécédent. L'apprentissage statistique se place dans la première situation puisque nous cherchons un estimateur

f^* répondant au problème d'optimisation dans lequel $\mathcal{O} = \mathbb{R}^{\mathcal{X}}$, $\mathcal{C} = \mathcal{F}$ et $J\colon f \in \mathbb{R}^{\mathcal{X}} \to \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\boldsymbol{x_i}))$.

Il est rare de réussir à déterminer une solution analytique aux problèmes d'optimisation d'apprentissage statistique. À défaut, on se contente donc d'une solution numérique. C'est ici qu'intervient l'aspect informatique de l'apprentissage statistique. Même si nous ne détaillons pas cet aspect ici, il est important de noter qu'une solution (dite globale) est raisonnablement accessible (*i.e.* en un temps suffisamment court et avec des ressources calculatoires abordables) lorsque le problème d'optimisation est convexe [Boyd et Vandenberghe, 2004]. Lorsque ce n'est pas le cas, les approches numériques ne fournissent généralement qu'une solution locale, *i.e.* minimisant J seulement dans un voisinage. Nous rappelons à présent quelques définitions de convexité, avant de revenir à celle d'un problème convexe.

Définition 1.2.4 (Ensemble convexe)**.**
Un sous-ensemble $\mathcal{S}$ d'un espace vectoriel est convexe ssi

$$\forall(\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{S}^2, \forall\lambda \in [0,1]\colon \lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{z} \in \mathcal{S}.$$

Définition 1.2.5 (Fonction convexe)**.**
Une fonction $f\colon \mathcal{I} \to \mathbb{R}$ ($\mathcal{I}$ étant un sous-ensemble d'un espace vectoriel) est convexe ssi

$$\Big\{(x, y) \in \mathcal{I} \times \mathbb{R} \;/\; f(x) \leq y\Big\} \text{ est un ensemble convexe.}$$

Cette définition est équivalente à

$$\forall(\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{I}^2, \forall\lambda \in [0,1]\colon f(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{z}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{z}).$$

Définition 1.2.6 (Problème d'optimisation convexe)**.**
Un problème d'optimisation est dit convexe si la fonction J et l'ensemble $\mathcal{C}$ sont tous deux convexes.

Si l'on sait résoudre numériquement tout problème convexe en un temps raisonnable, certains sont plus faciles que d'autres. Les problèmes les plus accessibles sont les programmes linéaires (*Linear Programs*, LP) et les programmes quadratiques (*Quadratic Programs*, QP) qui possèdent tous les deux des contraintes linéaires et des fonctions objectifs respectivement linéaire et quadratique. Un programme conique de second ordre (*Second-Order Cone Program*, SOCP) (objectif linéaire et contraintes coniques du second ordre) est un exemple de problèmes à contraintes quadratiques encore accessibles à résoudre. En revanche, il est plus difficile de résoudre un problème d'optimisation lorsque la fonction objectif et les contraintes sont quadratiques. On parle dans ce cas de programme quadratique à contraintes quadratiques (*Quadratically Constrained Quadratic Program*, QCQP). Une autre instance de problèmes d'optimisation dont il sera question dans ce manuscrit est le programme semi-défini (*Semi-Definite Program*, SDP) (objectif et contraintes linéaires sur le cône des matrices semi-définies positives). Bien que celui-ci puisse être efficacement résolu (c'est un problème linéaire), il faut garder à l'esprit que les variables sont des matrices qui croissent quadratiquement par rapport au nombre de données.

Au cours de ce manuscrit, nous allons utiliser la notion d'*équivalence* entre problèmes d'optimisation. Il est difficile de donner une définition complète (le lecteur pourra se référer à [Boyd et Vandenberghe, 2004, p. 130] pour des approches informelles). En conséquence, nous adoptons une définition faible mais suffisante pour nos besoins.

Définition 1.2.7 (Équivalence entre problèmes d'optimisation)**.**
Deux problèmes d'optimisation sont dits équivalents lorsque l'ensemble des solutions de l'un (*i.e.* l'ensemble des points pour lesquels le minimum est atteint) permet de déterminer toutes les solutions de l'autre, et *vice versa*.

1.3 MACHINE À VECTEURS SUPPORTS

1.3.1 Définition fonctionnelle

Une SVM, aussi appelée Séparateur à Vaste Marge, est un outil d'apprentissage statistique caractérisé par la classe des estimateurs $\mathcal{F}$, qui est une restriction d'un espace de Hilbert à noyau reproduisant (*Reproducing Kernel Hilbert Space*, RKHS). Comme nous allons le voir, un RKHS est un espace vectoriel de fonctions défini par un *noyau*. Ci-dessous, nous rappelons donc les notions de noyau (en donnant quelques exemples) et d'espace de Hilbert, avant d'aboutir à la définition d'un RKHS.

Définition 1.3.1 (Noyau)**.**
Un noyau semi-défini positif k est une application

$$k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R},$$

symétrique :

$$\forall (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X} \times \mathcal{X}\colon k(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{z}, \boldsymbol{x})$$

et vérifiant la condition de positivité pour tout entier n :

$$\forall (\boldsymbol{x_i})_{i=1}^{n} \in \mathcal{X}^n, \forall \boldsymbol{\alpha} \in \mathbb{R}^n\colon \sum_{1 \leq i,j \leq n} \alpha_i \alpha_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) \geq 0.$$

Lorsque la dernière inégalité est stricte pour tout n-uplet de vecteurs différents deux à deux, on parle de noyau défini positif. Par la suite, sauf cas contraire et explicitement mentionné, un *noyau* désigne un noyau semi-défini positif.

Exemple 1.3.1.
Les applications $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ suivantes sont des noyaux [Shawe-Taylor et Cristianini, 2004] :

- ***linéaire*** : $k(\boldsymbol{x}, \boldsymbol{z}) = \langle \boldsymbol{x} \mid \boldsymbol{z} \rangle_{\ell_2}$;
- ***polynomial*** : $k(\boldsymbol{x}, \boldsymbol{z}) = (1 + \gamma \langle \boldsymbol{x} \mid \boldsymbol{z} \rangle_{\ell_2})^q$ $(\gamma \geq 0, q \in \mathbb{N})$;
- ***exponentiel*** : $k(\boldsymbol{x}, \boldsymbol{z}) = \mathrm{e}^{\gamma \langle \boldsymbol{x} \mid \boldsymbol{z} \rangle_{\ell_2}}$ $(\gamma \geq 0)$;
- ***laplacien*** : $k(\boldsymbol{x}, \boldsymbol{z}) = \mathrm{e}^{-\gamma \|\boldsymbol{x} - \boldsymbol{z}\|_{\ell_2}}$ $(\gamma \geq 0)$;
- ***gaussien*** : $k(\boldsymbol{x}, \boldsymbol{z}) = \mathrm{e}^{-\gamma \|\boldsymbol{x} - \boldsymbol{z}\|_{\ell_2}^2}$ $(\gamma \geq 0)$.

Définition 1.3.2 (Espace de Hilbert)**.**
Un espace de Hilbert $(\mathcal{H}, \langle \cdot \mid \cdot \rangle_{\mathcal{H}}, \|\cdot\|_{\mathcal{H}})$ est un espace vectoriel préhilbertien (*i.e.* muni d'un produit scalaire $\langle \cdot \mid \cdot \rangle_{\mathcal{H}}$), normé par la norme qui découle du produit scalaire et complet par rapport à la distance induite par cette norme[2]. Par abus de notation, on dit que $\mathcal{H}$ est un espace de Hilbert, en le supposant muni de son produit scalaire et de sa norme.

Définition 1.3.3 (Espace de Hilbert à noyau reproduisant (RKHS))**.**
Soient $\mathcal{H}$ un espace de Hilbert de fonctions de $\mathcal{X}$ dans $\mathbb{R}$ et k un noyau. $\mathcal{H}$ est un RKHS de noyau k (ou k est un noyau reproduisant de $\mathcal{H}$) ssi

- $\forall x \in \mathcal{X}\colon k(x, \cdot) \in \mathcal{H}$;
- $\forall f \in \mathcal{H}, \forall x \in \mathcal{X}\colon f(x) = \langle f \mid k(x, \cdot) \rangle_{\mathcal{H}}$ (propriété de reproduction).

2. Autrement dit, c'est un espace de Banach (espace vectoriel normé et complet) dont la norme découle du produit scalaire.

La définition précédente exhibe le fait qu'un RKHS est caractérisé par un noyau k. Pour compléter cette définition, le théorème suivant assure que pour un noyau k donné, il existe un unique RKHS de noyau reproduisant k. En pratique, cela nous autorise à définir $\mathcal{H}$ uniquement par la connaissance de son noyau.

Théorème 1.3.1 (Théorème de Moore-Aronszajn [Aronszajn, 1950])**.**
Soit $k\colon \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ un noyau. Alors il existe un unique RKHS $\mathcal{H}$ associé à k. De plus, on peut construire $\mathcal{H}$ de la manière suivante. Soit $\mathcal{H}_0$ l'espace vectoriel engendré par les fonctions évaluation en chaque point de $\mathcal{X}$:

$$\mathcal{H}_0 = \mathrm{Vect}\{k(\boldsymbol{x},\cdot), \boldsymbol{x}\in\mathcal{X}\} = \left\{ \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x_i},\cdot)\colon n\in\mathbb{N}, (\alpha_i)_{i=1}^n\in\mathbb{R}^n, (\boldsymbol{x_i})_{i=1}^n\in\mathcal{X}^n \right\},$$

et $\langle\cdot\mid\cdot\rangle_{\mathcal{H}_0}$ défini pour toutes fonctions $f=\sum_{i=1}^{n_1}\alpha_i k(\boldsymbol{u_i},\cdot)$ et $g=\sum_{j=1}^{n_2}\beta_j k(\boldsymbol{v_j},\cdot)$ de $\mathcal{H}_0$ par

$$\langle f\mid g\rangle_{\mathcal{H}_0} = \sum_{\substack{1\le i\le n_1\\ 1\le j\le n_2}} \alpha_i\beta_j k(\boldsymbol{u_i},\boldsymbol{v_j}).$$

*Alors $\mathcal{H}$ est le complété de $\mathcal{H}_0$ au sens de $\|\cdot\|_{\mathcal{H}_0}$ (*i.e. *$\mathcal{H}_0$ augmenté des limites des suites de Cauchy d'éléments de $\mathcal{H}_0$) :*

$$\mathcal{H} = \overline{\mathrm{Vect}\{k(\boldsymbol{x},\cdot), \boldsymbol{x}\in\mathcal{X}\}},$$

muni du produit scalaire défini pour toutes limites de suites de Cauchy $(f_n)_{n=1}^\infty$ et $(g_n)_{n=1}^\infty$ de $\mathcal{H}$ par :

$$\left\langle \lim_{n\to\infty} f_n \mid \lim_{n\to\infty} g_n \right\rangle_{\mathcal{H}} = \lim_{n\to\infty} \langle f_n\mid g_n\rangle_{\mathcal{H}_0}.$$

Comme nous l'avons précisé, une SVM est un outil d'apprentissage statistique caractérisé par l'espace d'hypothèses $\mathcal{F}$, qui est une classe de fonctions issues d'un RKHS. Pour définir une SVM, nous aurons donc besoin d'un noyau k (ou de manière équivalente d'un RKHS $\mathcal{H}$) et d'une borne c de $\mathbb{R}_+$.

Définition 1.3.4 (Machine à vecteur support)**.**
Soient $\mathcal{H}$ un RKHS, $\psi\colon\mathcal{H}\to\mathbb{R}$ une forme convexe (dite fonction de régularisation), c une constante et $L\colon\mathbb{R}\times\mathbb{R}\to\mathbb{R}_+$ une fonction de perte telle que les deux applications $b\in\mathbb{R}\mapsto L(1,b)$ et $b\in\mathbb{R}\mapsto L(-1,b)$ sont convexes.

Une SVM est un algorithme $\mathfrak{S}\colon (\mathcal{X}\times\{\pm1\})^n\to\mathbb{R}^{\mathcal{X}}$ qui à un ensemble d'apprentissage $\{(\boldsymbol{x_i},y_i)\}_{1\le i\le n}$ associe une fonction de décision de $\mathcal{H}$ biaisée :

$$\mathfrak{S}\colon \{(\boldsymbol{x_i},y_i)\}_{1\le i\le n}\mapsto \Big(\boldsymbol{x}\in\mathcal{X}\mapsto f^*(\boldsymbol{x})+b^*\in\mathbb{R}\Big),$$

avec

$$(f^*,b^*)\in \operatorname*{arg\,min}_{f\in\mathcal{F},\, b\in\mathbb{R}} \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\boldsymbol{x_i})+b),$$

et $\mathcal{F}=\{f\in\mathcal{H},\ \psi(f)\le c\}$.

Nous pouvons remarquer que nous avons volontairement laissé une ambiguïté dans le choix de (f^*,b^*), faisant de $\mathfrak{S}$ un algorithme plutôt qu'une application. En effet, en pratique il existe des situations pour lesquelles plusieurs couples optimaux existent (f^*,b^*). Ces situations correspondent à un problème d'optimisation non-strictement convexe, résultant soit d'un noyau qui est seulement semi-défini positif plutôt que défini positif, soit de la présence d'observations identiques (numériquement très proches) dans l'ensemble d'apprentissage, soit enfin à des configurations géométriques particulières autorisant plusieurs

biais adéquats. Dans tous les cas, le choix final de (f^*, b^*) importe peu puisque le risque empirique y atteint son minimum. En outre, l'algorithme précédent peut laisser à penser qu'il est nécessaire de parcourir un espace vectoriel de dimension infinie (le RKHS $\mathcal{H}$) pour déterminer le minimum du risque empirique. En réalité, comme l'assure le théorème du représentant, on sait par avance que f^* vit dans un sous-espace vectoriel de dimension finie engendré par les données.

Théorème 1.3.2 (Théorème du représentant [Kimeldorf et Wahba, 1971, Schölkopf *et coll.*, 2001])**.** *Soit $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ un noyau et $\mathcal{H}$ son RKHS. Soit une fonction de régularisation ψ de la forme $\psi\colon f \in \mathcal{H} \mapsto g(\|f\|_{\mathcal{H}}) \in \mathbb{R}$ avec g strictement croissante. Alors $\mathfrak{S}(\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}) = f^* + b^*$ avec $b^* \in \mathbb{R}$ et*

$$f^* \in \mathrm{Vect}\{k(\boldsymbol{x_i}, \cdot), i \in \mathbb{N}_n\}.$$

Autrement dit,

$$\exists \boldsymbol{\alpha} \in \mathbb{R}^n\colon f^* = \sum_{i=1}^{n} \alpha_i k(\boldsymbol{x_i}, \cdot).$$

L'importance du théorème du représentant est de démontrer que, dans le cas d'une régularisation ψ qui est une fonction croissante de la norme de f, tout estimateur optimal f^* vit dans un sous-espace vectoriel de dimension finie du RKHS $\mathcal{H}$, même si ce dernier est de dimension infinie.

1.3.2 Approche numérique

Déclinaisons du paradigme SVM

Dans la définition de l'algorithme SVM $\mathfrak{S}$, nous avons imposé des contraintes de convexité sur les fonctions de régularisation $\psi\colon \mathcal{H} \to \mathbb{R}$ et de perte $L\colon \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$. Celles-ci ont pour but de conserver le caractère convexe du problème d'optimisation associé au paradigme SVM et ainsi de mettre en place des méthodes de résolution numériques efficaces. En outre, avec ces contraintes de convexité, le problème SVM est équivalent à

$$\underset{f\in\mathcal{H},\, b\in\mathbb{R}}{\text{minimiser}}\ \lambda\psi(f) + \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\boldsymbol{x_i}) + b),$$

où λ est un certain réel positif [Tikhonov et Arsenin, 1977]. Sous cette forme, le principe SVM rentre dans la théorie de la régularisation, énoncée dans l'ouvrage précédemment cité. Cette théorie permet de construire des problèmes *bien posés*, *i.e.* dont une solution unique et stable existe.

Cette nouvelle formulation du problème d'optimisation est la plus communément rencontrée puisque plus facile à résoudre d'un point de vue numérique que celle possédant une contrainte explicite sur f. De plus elle ouvre la voie à de nombreuses combinaisons de fonctions de régularisation et de perte, dont certaines ne répondent pas au théorème du représentant, voire ne sont pas convexes. Par exemple, lorsque le noyau utilisé est linéaire : $k\colon (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X} \times \mathcal{X} \mapsto \langle \boldsymbol{x} \mid \boldsymbol{z} \rangle_{\ell_2}$, le théorème 1.3.1 de Moore-Aronszajn nous assure que le RKHS associé $\mathcal{H}$ est l'ensemble des applications linéaires $\{\langle \mathbf{w} \mid \cdot \rangle_{\ell_2}, \mathbf{w} \in \mathcal{X}\}$ muni du produit scalaire $\left\langle \langle \mathbf{w}_1 \mid \cdot \rangle_{\ell_2} \mid \langle \mathbf{w}_2 \mid \cdot \rangle_{\ell_2} \right\rangle_{\mathcal{H}} = \langle \mathbf{w}_1 \mid \mathbf{w}_2 \rangle_{\ell_2}$ (*a fortiori* $\left\| \langle \mathbf{w} \mid \cdot \rangle_{\ell_2} \right\|_{\mathcal{H}} = \|\mathbf{w}\|_{\ell_2}$). Dans ce cas, il n'est plus nécessaire de répondre aux hypothèses du théorème du représentant (puisque $\mathcal{H}$ est *de facto* un espace vectoriel de dimension finie) et on trouve différentes fonctions de régularisation dans la littérature :

- ◇ les ***normes*** : ℓ_1, ℓ_2 au carré (qui correspond à $\|\cdot\|^2_{\mathcal{H}}$ pour un noyau quelconque) et de manière plus générale, ℓ_p à la puissance p (pour p réel et supérieur à 1) ;

- ◇ les ***quasi-normes*** : ℓ_p ($0 \leq p < 1$, lorsque $p = 0$ on préférera la première formulation SVM vue dans ce manuscrit, comportant une contrainte explicite sur l'estimateur) ;
- ◇ ***elastic-net*** : $\psi\left(\langle \mathbf{w} \mid \cdot \rangle_{\ell_2}\right) = \tau \|\mathbf{w}\|_{\ell_1} + (1-\tau)\|\mathbf{w}\|_{\ell_2}^2$;
- ◇ ***pseudo-norme mixte*** ℓ_{p-q} : soient g un entier non-nul et $\{\mathcal{G}_l\}_{1 \leq l \leq g}$ une partition de $\mathbb{N}_d$. On définit la régularisation en pseudo-norme mixte par :

$$\psi\left(\langle \mathbf{w} \mid \cdot \rangle_{\ell_2}\right) = \sum_{l=1}^{g} \left(\sum_{k \in \mathcal{G}_l} |\mathbf{w}_k|^q \right)^{\frac{p}{q}},$$

  correspondant à la pseudo-norme ℓ_{p-q} à la puissance p.

On pourra se référer à [Szafranski, 2008] pour une revue détaillée des régularisations en apprentissage statistique supervisé.

Revenons à présent à un noyau k quelconque. Différentes fonctions de perte coexistent [Hastie *et coll.*, 2008, section 11.3]. Les plus courantes sont les pertes :

- ◇ ***logistique*** : $L(a,b) = \ln(1 + \mathrm{e}^{-ab})$;
- ◇ ***quadratique*** : $L(a,b) = (a-b)^2$;
- ◇ ***charnière*** : $L(a,b) = \max(0, 1-ab)$;
- ◇ ***charnière quadratique*** : $L(a,b) = \max(0,1-ab)^2$;
- ◇ ***de Huber*** : $L(a,b) = -4ab$ si $ab \leq -1$ et $\max(0,1-ab)^2$ sinon.

La version la plus communément rencontrée du paradigme SVM est celle définie par la régularisation en norme de $\mathcal{H}$ ($\psi\colon f \in \mathcal{H} \mapsto \frac{1}{2}\|f\|_{\mathcal{H}}^2$, qui vérifie la condition de stricte croissance du théorème 1.3.2) et par la fonction de perte charnière $L\colon (a,b) \in \mathbb{R} \times \mathbb{R} \mapsto \max(0, 1-ab)$. Dans ce cas, la classe des estimateurs est $\mathcal{F} = \{f \in \mathcal{H},\ \frac{1}{2}\|f\|_{\mathcal{H}}^2 \leq c\}$. Cette restriction du RKHS $\mathcal{H}$ s'interprète comme l'ensemble des fonctions de *petites variations*. En effet :

$$\begin{aligned} \forall (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X} \times \mathcal{X}, \quad |f(\boldsymbol{x}) - f(\boldsymbol{z})| &= |\langle f \mid k(\boldsymbol{x}, \cdot) - k(\boldsymbol{z}, \cdot) \rangle_{\mathcal{H}}| \\ &\leq \|f\|_{\mathcal{H}} \|k(\boldsymbol{x}, \cdot) - k(\boldsymbol{z}, \cdot)\|_{\mathcal{H}} \\ &\leq c \|k(\boldsymbol{x}, \cdot) - k(\boldsymbol{z}, \cdot)\|_{\mathcal{H}} . \end{aligned}$$

La constante c, que nous nommerons par la suite *constante de Lipschitz apparente*, contrôle donc les variations de la fonction f et ainsi sa *complexité*.

Résolution numérique

Avec ces fonctions de régularisation et de perte, le problème d'apprentissage SVM devient alors

$$\underset{f \in \mathcal{H},\, b \in \mathbb{R}}{\text{minimiser}} \quad \frac{\lambda}{2}\|f\|_{\mathcal{H}}^2 + \frac{1}{n}\sum_{i=1}^{n} \max\left(0, 1 - y_i\left(f(\boldsymbol{x_i}) + b\right)\right).$$

Puisque la fonction de perte charnière n'est pas dérivable partout, on introduit traditionnellement un vecteur de variables d'écart $\boldsymbol{\xi}$ (de $\mathbb{R}_+^n$) majorant les pertes $L(y_i, f(\boldsymbol{x_i}) + b)$ et on reformule le problème de la manière suivante :

$$\begin{aligned} &\underset{f \in \mathcal{H},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n}{\text{minimiser}} && \frac{\lambda}{2}\|f\|_{\mathcal{H}}^2 + \frac{1}{n}\sum_{i=1}^{n} \xi_i \\ &\text{tel que} && \begin{cases} y_i\left(f(\boldsymbol{x_i}) + b\right) \geq 1 - \xi_i, \ \forall i \in \mathbb{N}_n \\ \xi_i \geq 0. \end{cases} \end{aligned} \tag{1.1}$$

Pour ces fonctions de régularisation et de perte, un couple optimal (f^*, b^*) est alors généralement déterminé par la résolution d'un problème dual [Platt, 1999] (sauf rare contre-exemple [Chapelle, 2007]) :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha}\in\mathbb{R}^n}{\text{maximiser}} & \lambda\sum_{i=1}^{n}\alpha_i - \frac{\lambda}{2}\sum_{1\leq i,j\leq n}\alpha_i\alpha_j y_i y_j k(\boldsymbol{x_i},\boldsymbol{x_j}) \\ \text{tel que} & \begin{cases} 0 \leq \alpha_i \leq \dfrac{1}{n\lambda}, \forall i \in \mathbb{N}_n \\ \displaystyle\sum_{i=1}^{n} y_i\alpha_i = 0, \end{cases} \end{array}$$

puis f^* est calculé par la formule :

$$f^* = \sum_{i=1}^{n}\alpha_i y_i k(\boldsymbol{x_i}, \cdot).$$

Dans ce cas précis, l'intérêt du problème d'optimisation dual réside dans la diminution de la taille du problème : le nombre de variables d'optimisation est moindre par rapport au problème primal. Cet intérêt est justifié par une particularité du problème SVM : il est possible de déterminer les variables primales f^* et b^* à partir de la seule connaissance du multiplicateur de Lagrange $\boldsymbol{\alpha}$. Au contraire, dans le cas d'un noyau linéaire avec $\psi = \|\cdot\|_{\ell_1}$, il n'est pas possible de déterminer les variables primales ne connaissant que les variables duales.

Remarque 2.
La détermination du biais b^* est réalisée grâce aux points d'apprentissage qui sont des *vecteurs supports non-pénalisants* (voir le partitionnement de l'espace de représentation donné dans la sous-section suivante). Lorsqu'il n'existe aucun vecteur support non-pénalisant, le classifieur optimal $f^* + b^*$ n'est pas unique car b^* peut être choisi dans un segment de mesure non-nulle [Karasuyama et Takeuchi, 2010].

Remarque 3.
À l'optimalité, les valeurs des fonctions objectif des problèmes primal et dual se rejoignent :

$$\lambda\psi(f^*) + \frac{1}{n}\sum_{i=1}^{n} L(y_i, f^*(\boldsymbol{x_i}) + b^*) = \lambda\sum_{i=1}^{n}\alpha_i - \frac{\lambda}{2}\sum_{1\leq i,j\leq n}\alpha_i\alpha_j y_i y_j k(\boldsymbol{x_i},\boldsymbol{x_j}).$$

1.3.3 Interprétation géométrique

Jusqu'ici, nous avons présenté les SVM comme des outils d'apprentissage statistique possédant une particularité fonctionnelle (l'utilisation d'un RKHS comme espace des hypothèses) qui, par la suite, se sont ouverts à plusieurs variantes à travers le choix des fonctions de régularisation et de perte. Ce serait un tort de négliger l'interprétation géométrique d'une SVM, qui donne une intuition de la notion de régularisation, différente de celle consistant à limiter les variations de f.

Pour ce faire, nous introduisons à présent le concept d'*espace de redescription*, qui représente un nouvel espace de Hilbert, potentiellement de grande dimension, dans lequel les données sont réarrangées et traitées comme deux classes linéairement séparables. Le théorème suivant, dû à Aronszajn, nous affirme l'existence d'un tel espace et d'une fonction de redescription permettant de lier les entrées à leurs images dans ledit espace.

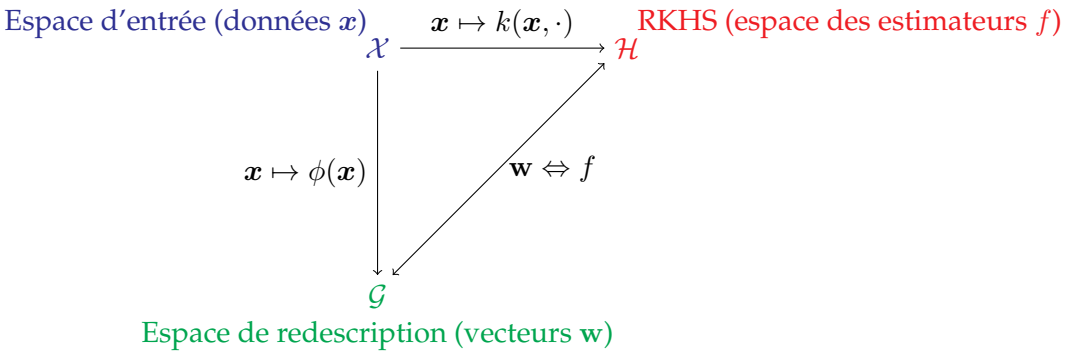


FIGURE 1.1 – Interactions entre les trois espaces considérés.

Théorème 1.3.3 (Caractérisation d'un noyau [Aronszajn, 1950])**.**
$k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est un noyau ssi il existe un espace de Hilbert $\left(\mathcal{G}, \langle \cdot \mid \cdot \rangle_{\mathcal{G}}\right)$ et une fonction de redescription $\phi\colon \mathcal{X} \to \mathcal{G}$ tels que

$$\forall (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X} \times \mathcal{X}\colon k(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}) \mid \phi(\boldsymbol{z}) \rangle_{\mathcal{G}}\,.$$

La figure 1.1 illustre les liens entre les trois espaces considérés dans la théorie SVM. Nous y trouvons d'abord l'espace des observations $\mathcal{X}$, aussi appelé espace d'entrée ; puis, l'espace des hypothèses $\mathcal{H}$, engendré par les fonctions évaluation $k(\boldsymbol{x}, \cdot)$. Enfin, l'espace de redescription $\mathcal{G}$, lié à $\mathcal{X}$ par une fonction de redescription telle que celle mentionnée dans le théorème précédent. Comme nous allons le voir par la suite, il est aussi possible de connecter les éléments de $\mathcal{H}$ à ceux de $\mathcal{G}$. Avant cela, il est important de remarquer qu'aucun des liens que nous avons mentionnés jusqu'à présent, entre les différents espaces, n'est unique.

Il existe différentes façons de construire une fonction de redescription ϕ (par exemple grâce aux théorèmes de Moore-Aronszajn et de Mercer) mais celle-ci n'a pas réellement d'intérêt puisque $\mathcal{G}$ est généralement de dimension infinie. Néanmoins, cette fonction de redescription permet une interprétation géométrique du paradigme SVM dans le cas usuel de la fonction de régularisation $\psi = \frac{1}{2}\|\cdot\|_{\mathcal{H}}^2$ et de la fonction de perte charnière $L\colon (a, b) \in \mathbb{R} \times \mathbb{R} \mapsto \max(0, 1 - ab)$. En appelant $C = \frac{1}{n\lambda}$ le *paramètre de coût* (ou le *paramètre de compromis* entre le terme d'attache aux données et la régularisation), on peut récrire (1.1) sous la forme :

$$\begin{array}{ll} \underset{f \in \mathcal{H},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n}{\text{minimiser}} & \dfrac{1}{2}\|f\|_{\mathcal{H}}^2 + C\sum_{i=1}^{n} \xi_i \\ \text{tel que} & \begin{cases} \forall i \in \mathbb{N}_n / y_i = 1\colon f(\boldsymbol{x_i}) + b \geq 1 - \xi_i \\ \forall i \in \mathbb{N}_n / y_i = -1\colon f(\boldsymbol{x_i}) + b \leq -1 + \xi_i \\ \xi_i \geq 0. \end{cases} \end{array}$$

On voit clairement apparaître la volonté qu'un point d'étiquette positive vérifie $f(\boldsymbol{x_i}) + b \geq 1$, et $f(\boldsymbol{x_i}) + b \leq -1$ pour un point d'étiquette négative. De plus, en notant $\boldsymbol{Y} = \mathrm{Diag}\left((y_i)_{1 \leq i \leq n}\right)$ la matrice diagonale des étiquettes et $\boldsymbol{K}_+ = \left(k(\boldsymbol{x_i}, \boldsymbol{x_j})\right)_{1 \leq i,j \leq n}$ la matrice noyau, un problème dual au précédent est :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\text{maximiser}} & \mathbb{1}^T\boldsymbol{\alpha} - \dfrac{1}{2}\boldsymbol{\alpha}^T\boldsymbol{Y}\boldsymbol{K}_+\boldsymbol{Y}\boldsymbol{\alpha} \\ \text{tel que} & \begin{cases} 0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C \\ \boldsymbol{y}^T\boldsymbol{\alpha} = 0. \end{cases} \end{array} \tag{1.2}$$

En outre, les conditions d'optimalité de Karush-Kuhn-Tucker (KKT) assurent que toute solution f du problème d'optimisation primal s'exprime à travers un vecteur $\boldsymbol{\alpha}$ (de $\mathbb{R}_+^n$), so-

lution du problème dual, par :

$$\forall \boldsymbol{x} \in \mathcal{X}\colon f(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i y_i k(\boldsymbol{x_i}, \boldsymbol{x}).$$

En définissant le vecteur $\mathbf{w}$ de $\mathcal{G}$ par $\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \phi(\boldsymbol{x_i})$, on obtient une nouvelle expression de f :

$$\forall \boldsymbol{x} \in \mathcal{X}\colon f(\boldsymbol{x}) = \langle \mathbf{w} \mid \phi(\boldsymbol{x}) \rangle_{\mathcal{G}}\,.$$

Ainsi, un couple (f, b) solution de (1.1) définit un hyperplan séparateur $\mathfrak{P}$ dans l'espace de redescription $\mathcal{G}$ (voir figure 1.2 page suivante). Sur cette illustration, on voit apparaître la notion de marge : l'espace interstitiel correspondant aux points $\phi(\boldsymbol{x})$ tels que $\left|\langle \mathbf{w} \mid \phi(\boldsymbol{x}) \rangle_{\mathcal{G}} + b\right| \leq 1$. Par abus de langage, on appelle aussi *marge* la distance entre les deux hyperplans définis par $\langle \mathbf{w} \mid \phi(\boldsymbol{x}) \rangle_{\mathcal{G}} + b = 1$ et $\langle \mathbf{w} \mid \phi(\boldsymbol{x}) \rangle_{\mathcal{G}} + b = -1$. Tous les points $\phi(\boldsymbol{x_i})$ tombant à l'intérieur de cette marge sont pénalisés par un écart ξ_i non-nulle. Ainsi, une SVM cherche un hyperplan séparant les classes et pour lequel aucun point n'est situé dans sa marge (c'est l'interprétation de la fonction de perte charnière). En pratique, il suffirait de considérer un hyperplan de marge presque nulle pour s'assurer qu'aucun point n'est à l'intérieur. Pour éviter cette dégénérescence, une SVM cherche donc un hyperplan qui maximise la marge entre les deux classes (c'est l'interprétation de $\frac{1}{2}\|f\|_{\mathcal{H}}$). En effet, soit l'ensemble des points à la frontière de la marge $\mathcal{M} = \{\phi(\boldsymbol{x_i}) \;/\; i \in \mathbb{N}_n, \left|\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b\right| = 1\}$ (dits *sur* la marge). La marge correspond alors à $2\,\mathrm{Dist}(\mathfrak{P}, \phi(\boldsymbol{x}))$ avec $\phi(\boldsymbol{x}) \in \mathcal{M}$. Puisque la distance à un hyperplan est donnée par :

$$\mathrm{Dist}(\mathfrak{P}, \phi(\boldsymbol{x})) = \frac{|\,\langle \mathbf{w} \mid \phi(\boldsymbol{x}) \rangle_{\mathcal{G}} + b|}{\|\mathbf{w}\|_{\mathcal{G}}},$$

alors la marge vaut $\frac{2}{\|\mathbf{w}\|_{\mathcal{G}}}$. Or les normes de f et de $\mathbf{w}$ sont égales puisque :

$$\begin{aligned}
\|f\|_{\mathcal{H}}^2 &= \left\langle \sum_{i=1}^{n} \alpha_i y_i k(\boldsymbol{x_i}, \cdot) \mid \sum_{j=1}^{n} \alpha_j y_j k(\boldsymbol{x_j}, \cdot) \right\rangle_{\mathcal{H}} = \sum_{1 \leq i,j \leq n} \alpha_i \alpha_j y_i y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) \\
&= \sum_{1 \leq i,j \leq n} \alpha_i \alpha_j y_i y_j \langle \phi(\boldsymbol{x_i}) \mid \phi(\boldsymbol{x_j}) \rangle_{\mathcal{G}} = \left\langle \sum_{i=1}^{n} \alpha_i y_i \phi(\boldsymbol{x_i}) \mid \sum_{j=1}^{n} \alpha_j y_j \phi(\boldsymbol{x_j}) \right\rangle_{\mathcal{G}} \\
&= \|\mathbf{w}\|_{\mathcal{G}}^2\,.
\end{aligned}$$

Ainsi, la marge vaut $\frac{2}{\|f\|_{\mathcal{H}}}$. Minimiser $\frac{1}{2}\|f\|_{\mathcal{H}}^2$ correspond bien à maximiser la marge. Cette notion donne le nom de Séparateur Vaste Marge à une SVM.

En analysant les contraintes du problème d'optimisation, on peut partitionner l'espace en trois zones : en dehors de la marge, sur la marge et dans la marge (*cf.* illustration 1.2). Chacune de ces zones correspond à des variables duale α_i et d'écart ξ_i de caractéristiques déterminées :

- ◇ vecteurs $\phi(\boldsymbol{x_i})$ ***en dehors de la marge*** : $\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b > 1$ ($y_i = 1$) ou $\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b < -1$ ($y_i = -1$) [$\alpha_i = 0,\ \xi_i = 0$] ;
- ◇ vecteurs $\phi(\boldsymbol{x_i})$ ***sur la marge***, vérifiant $\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b = 1$ ($y_i = 1$) ou $\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b = -1$ ($y_i = -1$) [$0 \leq \alpha_i \leq C,\ \xi_i = 0$] ;
- ◇ vecteurs $\phi(\boldsymbol{x_i})$ ***dans la marge*** : $\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b < 1$ ($y_i = 1$) ou $\langle \mathbf{w} \mid \phi(\boldsymbol{x_i}) \rangle_{\mathcal{G}} + b > -1$ ($y_i = -1$) [$\alpha_i = C,\ \xi_i > 0$].

On appelle couramment ***vecteurs supports*** les points $\phi(\boldsymbol{x_i})$ pour lesquels $\alpha_i > 0$ (ils sont situés sur et dans la marge). On remarque alors que la direction de l'hyperplan séparateur $\mathfrak{P}$ n'est définie que par ces vecteurs puisque $\mathbf{w} = \sum_{\substack{1 \leq i \leq n \\ \alpha_i > 0}} \alpha_i y_i \phi(\boldsymbol{x_i})$.

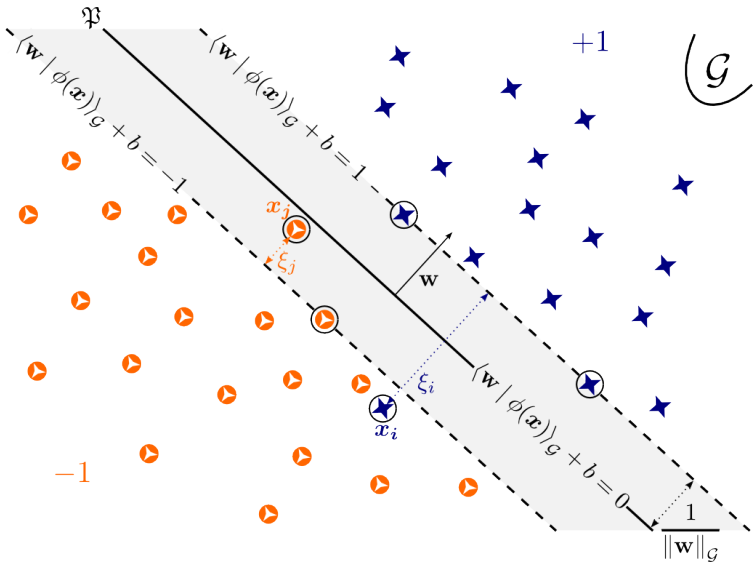


FIGURE 1.2 – Illustration du principe SVM. Celui-ci consiste à séparer les données en maximisant la marge entre les deux groupes (zone grisée). L'hyperplan s'éparateur n'est alors défini que par les vecteurs entourés (dits supports).

1.4 SÉLECTION DE MODÈLE

1.4.1 Risque structurel

Outre les propriétés statistiques intéressantes d'une SVM, nous avons vu que l'apprentissage de celle-ci revient à la résolution d'un problème d'optimisation quadratique convexe (QP). Pour réaliser cet apprentissage, il est nécessaire de fixer au préalable le coefficient de Lipschitz apparent c, bornant la norme des estimateurs f (ou de manière équivalente, l'un des paramètres de compromis C ou λ), ainsi que le noyau définissant le RKHS des hypothèses. Ces deux paramètres sont régulièrement appelés, *hyper-paramètres*. Puisque la complexité (ou capacité) de l'espace $\mathcal{F}$, quantifiée par la dimension de Vapnik-Chervonenkis $VC(\mathcal{F})$, dépend des hyper-paramètres [Vapnik, 1998], ceux-ci présentent une importance capitale dans la détermination d'un estimateur f minimisant le risque réel d'erreur. Pour simplifier le discours, nous ne nous intéressons, dans la suite, qu'au coefficient de Lipschitz apparent c comme hyper-paramètre. Comme l'assure de théorème 1.2.1, la minimisation du risque réel met en jeu deux termes compétitifs :

- ◇ la probabilité d'erreur empirique, décroissante avec c puisqu'en augmentant ce dernier, on donne plus de *souplesse* (degrés de liberté) aux estimateurs ;
- ◇ le terme dépendant explicitement de la capacité de $\mathcal{F}$, croissant avec c.

Schématiquement, lorsque c est petit, les hypothèses possibles sont très *simplistes*, conduisant à un espace de faible capacité et à une erreur empirique importante. Ce phénomène se nomme *sous-apprentissage*. À l'inverse, lorsque c est grand, le modèle de classifieur est très complexe, conduisant à une erreur empirique très faible mais un espace de forte capacité. C'est le *sur-apprentissage*. Le problème de sélection de modèle consiste alors à se placer à l'inflexion de ces deux tendances.

Afin, de déterminer un estimateur convenable, Vapnik propose un algorithme de *minimisation du risque structurel* [Vapnik, 1998] :

1. définir une famille structurée de p classes $\mathcal{F}_1 \subset \cdots \subset \mathcal{F}_p$ de capacité croissante. En pratique, cela revient à considérer une suite croissante de coefficients de Lipschitz apparents $c_1 \leq \cdots \leq c_p$;

2. déterminer l'estimateur correspondant à chaque classe de fonctions :
   $\forall j \in \mathbb{N}_p \colon f_j \in \arg\min_{f \in \mathcal{F}_j} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(\boldsymbol{x_i}))$. Peu importe j plus petit que $p-1$, puisque $\mathcal{V}_j \subset \mathcal{V}_{j+1}$, alors le risque empirique de f_j est plus grand que celui de f_{j+1}. La famille structurée de classes définit donc une suite d'estimateurs dont le risque empirique diminue ;
3. estimer une borne du risque réel à partir de la connaissance de la capacité $\mathrm{dimVC}(\mathcal{F}_j)$ et du risque empirique de f_j et sélectionner l'estimateur f_j qui la minimise.

En pratique, la minimisation du risque structurel est rarement mise en œuvre sous cette forme. Les principales raisons sont la difficulté à calculer la capacité d'un espace et la grossièreté des bornes théoriques, indépendantes de la distribution des données. On préfère à la place le principe de validation croisée qui est une autre forme d'estimation de la probabilité réelle d'erreur, plus facile à concrétiser. Dans la section suivante, nous mentionnons ce principe et détaillons quelques critères de choix des hyper-paramètres.

1.4.2 Critères

Bornes de généralisation

Il existe de nombreuses façons d'aborder le choix des hyper-paramètres d'une SVM. Ces approches, dites de *sélection de modèle*, peuvent par exemple être classées par rapport à la paramétrisation du modèle, au critère de discrimination et à l'algorithme de résolution. Cette section a pour vocation d'introduire les critères de choix rencontrés couramment dans la littérature. Au fil de ce manuscrit, nous retrouverons ces critères (au centre de problèmes d'optimisation) et nous détaillerons les paramétrisations ainsi que les algorithmes de résolution qui les accompagnent.

La technique de sélection de modèle la plus couramment implémentée est la validation croisée [Hastie *et coll.*, 2008]. Le critère correspondant J_{MV} (*Multiple Validation*) est calculé en évaluant l'erreur empirique (sur des données inédites) de classifieurs préalablement construits. Formellement, on se donne une fonction de perte $\Lambda \colon \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ (par exemple $\Lambda(a,b) = 1$ si $\mathrm{Signe}(a) \neq \mathrm{Signe}(b)$ et 0 sinon) et un ensemble de p (entier quelconque) partitions $\{(\mathcal{V}_l, \mathcal{T}_l)\}_{1 \leq l \leq p}$ de $\mathbb{N}_n$, correspondant chacune à un ensemble de validation et d'apprentissage. Ces différentes partitions expliquent l'expression de validation *multiple*. À l'extrême, lorsque les ensembles de validation sont des singletons, on parle d'erreur un en dehors (*Leave-One-Out*, LOO). Pour tout l de $\mathbb{N}_p$, si l'on appelle $g_l = \mathfrak{S}(\{\boldsymbol{x_i}, y_i\}_{i \in \mathcal{T}_l})$ une fonction de décision SVM, alors le critère de validation multiple est défini par :

$$J_{\mathrm{MV}}(C,k) = \frac{1}{p} \sum_{l=1}^{p} \sum_{i \in \mathcal{V}_l} \Lambda(y_i, g_l(\boldsymbol{x_i})).$$

Dans ce critère, la dépendance au paramètre de coût C et au noyau k est introduite par les fonctions de décision g_l. Le principe de validation croisée consiste à approcher un minimum de la fonction précédente par une recherche en grille sur C et sur les paramètres du noyaux k, traditionnellement uniforme [Hastie *et coll.*, 2008] ou aléatoire [Bergstra et Bengio, 2012]. D'autres techniques de résolution existent, fondées soit sur des approches par gradient [Keerthi *et coll.*, 2006, Seeger, 2008] ou sur une reformulation du problème avec des contraintes complémentaires [Bennett *et coll.*, 2006, Kunapuli *et coll.*, 2008, Dong *et coll.*, 2007, Dong *et coll.*, 2008].

Dans le cas particulier du choix du paramètre de coût C, il existe une alternative, nommée *chemin de régularisation*. Hastie *et coll.* ont établi que les multiplicateurs de Lagrange optimaux $\boldsymbol{\alpha}$ d'une SVM sont des fonctions linéaires par morceaux de C. En déterminant les

points de coupure par une analyse géométrique, Hastie *et coll.* proposent une manière de déterminer les fonctions de décision SVM pour n'importe quelle valeur de C sans résoudre un nouveau QP à chaque fois [Hastie *et coll.*, 2004].

Sous certaines conditions, le critère de validation multiple utilisé dans le principe de validation croisée borne l'erreur réel d'un classifieur. Dans le cas d'une SVM (qui nous intéresse particulièrement), il existe plusieurs autres bornes, parmi lesquelles la borne rayon-marge, qui a démontré de bonnes aptitudes dans la détermination des hyper-paramètres [Chapelle *et coll.*, 2002]. Celle-ci, nommée par la suite J_{RM}, est réservée à une SVM à marge dure (*i.e.* $C \to +\infty$) et ne dépend donc que du noyau k (ou de manière équivalente de la matrice noyau $\boldsymbol{K}_+ = (k(\boldsymbol{x_i}, \boldsymbol{x_j}))_{1 \leq i,j \leq n}$)[3] :

$$J_{\text{RM}}(k) = \frac{1}{n} \frac{\rho(k)^2}{\nu(k)^2},$$

où $\rho(k)$ est le rayon de la plus petite boule de $\mathcal{G}$ (de centre $\boldsymbol{z}$ à déterminer) englobant les données et $\nu(k)$ est une marge SVM de définition légèrement différente de celle précédemment mentionnée :

$$\rho(k) = \min_{i \in \mathbb{N}_n, \boldsymbol{z} \in \mathcal{G}} \|\phi(\boldsymbol{x_i}) - \boldsymbol{z}\|_{\mathcal{G}}, \qquad \nu(k) = \min_{i \in \mathbb{N}_n} \frac{y_i(f(\boldsymbol{x_i}) + b)}{\|f\|_{\mathcal{H}}}.$$

Dans certains situations, la borne sur les données d'entrées ($\mathcal{X}$ est compact) induit une borne sur les données redécrites dans $\mathcal{G}$. Dans ce cas, certains auteurs ne considèrent que la marge du critère précédent [Neumann *et coll.*, 2005] :

$$J_{\text{M}}(k) = \frac{1}{\nu(k)^2}.$$

Apprentissage de noyau

Les critères précédemment énoncés sont tirés de bornes de généralisation [Vapnik, 1998]. Plus simplement, de nombreux auteurs ont utilisé le minimum de la fonction objectif d'une SVM (appelé ici *risque régularisé*) [Lanckriet *et coll.*, 2004, Bach *et coll.*, 2004, Rakotomamonjy *et coll.*, 2008] :

$$J_{\text{RR}}(C,k) = \left\{ \begin{array}{ll} \min\limits_{f,b,\boldsymbol{\xi}} & \frac{1}{2}\|f\|_{\mathcal{H}}^2 + C\sum\limits_{i=1}^{n} \xi_i \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(f(\boldsymbol{x_i}) + b\right) \geq 1 - \xi_i,\ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \end{array} \right. \end{array} \right. = \left\{ \begin{array}{ll} \max\limits_{\boldsymbol{\alpha} \in \mathbb{R}^n} & \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{Y} \boldsymbol{K}_+ \boldsymbol{Y} \boldsymbol{\alpha} \\ \text{tel que} & \left\{ \begin{array}{l} 0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C \\ \boldsymbol{y}^T \boldsymbol{\alpha} = 0. \end{array} \right. \end{array} \right.$$

Ce critère a été introduit pour un problème plus restreint que la sélection de modèle : l'apprentissage de noyau [Lanckriet *et coll.*, 2004]. Dans ce contexte, on considère C fixe et on note $J_{\text{RR}}(k)$, le risque régularisé[4].

Plus tôt et dans le seul but d'apprendre un noyau, le critère d'alignement J_{KA} (*Kernel Alignment*) a été introduit, mesurant la corrélation entre deux matrices noyau :

$$J_{\text{KA}}(k) = \frac{\langle h(\boldsymbol{K}_+) \mid h(\boldsymbol{Y}) \rangle_{\ell_2}}{\|h(\boldsymbol{K}_+)\|_{\ell_2} \|h(\boldsymbol{Y})\|_{\ell_2}}, \text{ où } \quad h \colon \boldsymbol{K}_+ \in \mathbb{S}_+ \mapsto \left(\boldsymbol{I}_+ - \frac{\mathbb{1}\mathbb{1}^T}{n}\right) \boldsymbol{K}_+ \left(\boldsymbol{I}_+ - \frac{\mathbb{1}\mathbb{1}^T}{n}\right).$$

Originellement, l'opération de centrage h était absente [Cristianini *et coll.*, 2002]. Elle a été introduite de manière à quantifier plus justement la corrélation entre deux noyaux [Cortes *et coll.*, 2010b, Cortes *et coll.*, 2012].

3. Pour une SVM pénalisée quadratiquement, on peut construire une SVM à marge dure équivalente à une SVM usuelle en remplaçant $\boldsymbol{K}_+$ par $\boldsymbol{K}_+ + \frac{1}{C}\boldsymbol{I}_+$ [Chapelle *et coll.*, 2002].

4. Ceci est justifié par la volonté de mettre en place une technique d'optimisation par descente de gradient. Or, comme nous le verrons plus tard, J_{RR} est facilement dérivable par rapport à k [Rakotomamonjy *et coll.*, 2008] ; ce qui est moins évident par rapport à C [Keerthi *et coll.*, 2006].

Distributions unimodales

Les critères que nous avons cités jusqu'à présent sont issus d'études théoriques avancées sur l'apprentissage statistique. Il existe pourtant des critères antérieurs et plus simples, fondés sur l'hypothèse d'une distribution unimodale de chaque classe (hypothèse utilisée dans l'analyse discriminante de Fisher). Deux exemples de ceux-ci, que nous retrouverons au fil de l'état de l'art, sont la *distance entre les centres des classes* (ou inter-classe) et le *critère de Fisher généralisé* (dont les définitions sont rappelées dans l'annexe A).

Il est difficile de savoir *a priori* quel critère choisir pour sélectionner un modèle SVM performant. Cependant, les comparaisons (partielles) disponibles tendent à privilégier des critères simples tels que la validation multiple, la borne rayon-marge et la distance inter-classe [Chapelle *et coll.*, 2002, Duan *et coll.*, 2003, Neumann *et coll.*, 2005].

Sur-apprentissage

En opposition à une approche par minimisation du risque structurel (ou par validation croisée) qui évalue une borne sur l'erreur réelle pour différents modèles SVM puis sélectionne le plus apte à généraliser, il est attirant de considérer la sélection de modèle comme un problème d'optimisation et de le résoudre par des approches usuelles (gradient, région de confiance, *etc.*). Pourtant, il est important de remarquer qu'une telle approche (aujourd'hui monnaie courante) introduit de nouveaux degrés de liberté dans l'apprentissage SVM puisque les hyper-paramètres sont déterminés automatiquement (ce que l'on souhaite). Ainsi, en mettant en place ces approches, on construit des règles de décision d'une plus grande *complexité* que si les hyper-paramètres étaient fixes. Comme nous l'avons vu, augmenter la complexité (ou la capacité) de l'espace des hypothèses conduit généralement à réduire le risque empirique d'erreur mais dégrade les aptitudes à la généralisation. Cet effet, que l'on nomme *sur-apprentissage*, est à éviter à tout prix. En particulier, pour ce qui est de l'apprentissage de noyau, il existe deux situations pathologiques majeures. La première est due à la propriété suivante [Aronszajn, 1950, Argyriou *et coll.*, 2006] : soit k un noyau et λ un réel positif non-nul. On appelle $\mathcal{H}_k$ et $\mathcal{H}_{\lambda k}$ les RKHS engendrés respectivement par les noyaux k et λk. Alors

$$\forall (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X}^2 \colon \; \langle k(\boldsymbol{x}, \cdot) \mid k(\boldsymbol{z}, \cdot) \rangle_{\mathcal{H}} = \frac{1}{\lambda^2} \langle \lambda k(\boldsymbol{x}, \cdot) \mid \lambda k(\boldsymbol{z}, \cdot) \rangle_{\mathcal{H}_k} = \frac{1}{\lambda^2} \langle k(\boldsymbol{x}, \cdot) \mid k(\boldsymbol{z}, \cdot) \rangle_{\mathcal{H}_{\lambda k}}$$

i.e.

$$\|\cdot\|_{\mathcal{H}_{\lambda k}} = \frac{1}{\lambda} \|\cdot\|_{\mathcal{H}_k} .$$

Cette propriété assure qu'en faisant croître λ, il est possible d'obtenir une quantité $\|f\|_{\mathcal{H}_{\lambda k}}$ arbitrairement proche de 0, peu importe l'estimateur f et le noyau k.

Il existe une deuxième source usuelle de sur-apprentissage, liée à l'utilisation du noyau gaussien $k \colon (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X}^2 \mapsto \exp\left(-\gamma \|\boldsymbol{x} - \boldsymbol{z}\|_{\ell_2}^2\right)$, où γ est un réel positif : lorsque γ est grand comparé aux données, k est approximativement le noyau identité : $k(\boldsymbol{x}, \boldsymbol{z}) = 1$ si $\boldsymbol{x} = \boldsymbol{z}$ et $k(\boldsymbol{x}, \boldsymbol{z}) = 0$ sinon. Dans ce cas, toute solution f de (1.1) est nulle presque partout et est donc inapte à généraliser.

Dans la suite, nous abordons plus précisément l'apprentissage de noyau, à travers la notion centrale de *noyau multiple*. Cette discipline est centrale au sein de nos travaux car, comme nous le verrons dans le chapitre suivant, apprendre un noyau correspond à apprendre une fonction de redescription.

1.4.3 Apprentissage de noyau multiple

Comme nous l'avons entraperçu juste avant, l'apprentissage de noyau est une sous-discipline de la sélection de modèle SVM consistant à déterminer le noyau $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ (ou sa matrice $\boldsymbol{K}_+$ construite avec les données d'apprentissage) définissant le RKHS des hypothèses f, afin de mettre au point une règle de décision généralisant au mieux. L'apprentissage de noyau est apparu avec les travaux de Cristianini *et coll.* sur la détermination automatique du rayon d'un noyau gaussien isotrope [Cristianini *et coll.*, 1999]. Dans ce dernier article, les auteurs introduisent un algorithme simple qui alterne une faible incrémentation du rayon gaussien et l'évaluation de la borne rayon-marge. L'algorithme s'arrête lorsqu'il atteint un minimum local.

Les questions purement théoriques de généralisation mises à part, un choix astucieux du noyau est aussi motivé par la nécessité de combiner des descripteurs de différentes natures, voire de sélectionner les descripteurs les plus pertinents (*feature selection*). Ce problème de sélection de variables a notamment été étudié par Weston *et coll.* par le biais d'un masque binaire (1 symbolisant que la variable est sélectionnée, 0 sinon) [Weston *et coll.*, 2001]. Dans ces travaux, apprendre le masque de sélection équivaut à apprendre le noyau SVM. C'est sous cet angle que Chapelle *et coll.* étendent l'étude précédente en comparant différents critères (dont les erreurs de validation simple et multiple, ainsi que la borne rayon-marge) pour l'apprentissage d'un noyau gaussien anisotrope (*i.e.* dont chaque dimension possède son propre rayon) [Chapelle *et coll.*, 2002].

Ces notions de combinaison et de sélection de variables, de concert avec la volonté de mettre en place des algorithmes efficaces, conduisent au concept de *noyau multiple*, qui est détaillé ci-dessous.

Définition et taxonomie

Sur la base des propriétés les plus simples de conservation du caractère semi-défini positif d'un noyau [Shawe-Taylor et Cristianini, 2004], une paramétrisation particulière a été envisagée, sous le nom de noyau multiple. Celle-ci consiste à construire un noyau comme une combinaison pondérée d'un ensemble de noyaux de base (appelés aussi noyaux générateurs). Ayant fixé les noyaux générateurs, l'apprentissage de noyaux multiples (*Multiple Kernel Learning*, MKL) cherche donc à inférer un vecteur de pondération $\boldsymbol{\mu}$ à partir des données d'apprentissage.

Définition 1.4.1 (Noyau multiple)**.**
Soient d un entier, $\mathcal{A}$ un sous-ensemble dénombrable et fini de $\mathbb{K}^d$ (ensemble des paramètres) et $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ un d-uplet de noyaux. Soient une fonction $m\colon \mathbb{R}^d \times \left(\mathbb{R}^{\mathcal{X}\times\mathcal{X}}\right)^d \to \left(\mathbb{R}^{\mathcal{X}\times\mathcal{X}}\right)^d$, et un vecteur $\boldsymbol{\mu}$ de $\mathbb{R}^d$. On appelle noyau multiple construit sur l'ensemble de noyaux générateurs $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ et on note $k_{[\boldsymbol{\mu}]}$:

$$k_{[\boldsymbol{\mu}]} = m\left(\boldsymbol{\mu}, (k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}\right),$$

lorsque $m\left(\boldsymbol{\mu}, (k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}\right)$ est effectivement un noyau (*i.e.* il est symétrique défini-positif).

Remarque 4.
Les règles les plus communes de préservation du caractère semi-défini positif sont [Shawe-Taylor et Cristianini, 2004, prop. 3.22 - 3.24, p. 75] :

- ◇ la combinaison conique : $k = \sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}$ pour $\boldsymbol{\mu} \succcurlyeq 0$;
- ◇ le produit : $k = \prod_{\boldsymbol{\theta}\in\mathcal{A}} k_{\boldsymbol{\theta}}^{\mu_{\boldsymbol{\theta}}}$ pour $\boldsymbol{\mu} \succcurlyeq 0$;
- ◇ l'expansion polynomiale : $k = p \circ k_{\boldsymbol{\theta}}$ pour un $\boldsymbol{\theta}$ donné et une fonction polynomiale $p\colon \mathbb{R} \to \mathbb{R}$ à coefficients positifs ;

◇ l'expansion exponentielle : $k = \exp \circ k_{\boldsymbol{\theta}}$ pour un $\boldsymbol{\theta}$ donné.

Exemple 1.4.1.
Voici quelques exemples de noyaux multiples couramment rencontrés dans la littérature :

◇ ***linéaire*** : $k_{[\boldsymbol{\mu}]} = \sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}$, pour les valeurs de $\boldsymbol{\mu}$ assurant que $\sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}$ est semi-défini positif ;

◇ ***conique*** : $k_{[\boldsymbol{\mu}]} = \sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}$, $\boldsymbol{\mu} \succcurlyeq 0$;

◇ ***convexe*** : $k_{[\boldsymbol{\mu}]} = \sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}$, $\boldsymbol{\mu} \succcurlyeq 0$ et $\mathbb{1}^T\boldsymbol{\mu} = 1$;

◇ ***multiplicatif*** : $k_{[\boldsymbol{\mu}]} = \prod_{\boldsymbol{\theta}\in\mathcal{A}} k_{\boldsymbol{\theta}}^{\mu_{\boldsymbol{\theta}}}$, $\boldsymbol{\mu} \succcurlyeq 0$.

MKL a été étudié dans nombre de travaux lors de la dernière décennie et nous ne cherchons pas ici à en faire une revue complète. Néanmoins, il est intéressant de rappeler les grands critères décrivant le domaine de recherche MKL. Dans la taxonomie de [Gönen et Alpaydin, 2011], six axes permettent d'expliquer les travaux réalisés :

1. ***inférence de la pondération*** : règle fixe (pas d'apprentissage), heuristique (pondération déterminée directement à partir des noyaux), optimisation, cadre bayésien (les inconnues sont traitées comme des variables aléatoires), *boosting* ;
2. ***combinaison*** : linéaire (sans contrainte, conique ou convexe), non-linéaire (multiplication, puissance, exponentiation), dirigée par les données ;
3. ***fonction objectif*** : mesure de similarité (alignement de noyaux, distance euclidienne, divergence de Kullback-Leibler), risque régularisé, fonction bayésienne (vraisemblance ou probabilité *a posteriori*) ;
4. ***apprentissage*** [5] : un temps (séquentiel, si le vecteur de pondération puis le classifieur sont appris successivement et séparément ; simultané, si tous les paramètres du modèle sont déterminés simultanément) ou deux temps (apprentissage simultané de tous les paramètres avec alternance des mises à jour : à chaque itération, le vecteur de pondération est déterminé à classifieur fixé, puis inversement le classifieur est déterminé à pondération fixée) ;
5. ***modèle de base*** : généralement une SVM (classification ou regression), parfois une analyse discriminante de Fisher (ou autre) ;
6. ***solution algorithmique*** (complexité) : liée à la famille des problèmes d'optimisation mis en jeu (LP, QP, SOCP, SDP, QCQP, programme linéaire semi-infini (*Semi-Infinite Linear Program*, SILP)).

Avant d'expliciter quelques approches d'apprentissage de noyau multiple, nous pouvons remarquer qu'il existe des garanties théoriques sur la capacité de généralisation de cet outil statistique. Les premières d'entres elles sont apparues avec les travaux de Lanckriet *et coll.*, au commencement de cette discipline [Lanckriet *et coll.*, 2004]. Le prochain théorème donne une majoration de la complexité de Rademacher d'un ensemble d'hypothèses MKL, due à Cortes *et coll.* Couplée au théorème 1.2.2, cette majoration fournit une borne plus fine que celles de Lanckriet *et coll.*

Théorème 1.4.1 (Borne de complexité [Cortes *et coll.*, 2010a])**.**
Soient $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ une famille de d noyaux générateurs, c un réel positif et $\mathcal{F}$ la classe de fonctions définie par :

$$\mathcal{F} = \left\{ f \in \mathcal{H}_{[\boldsymbol{\mu}]} \;/\; \|f\|_{\mathcal{H}_{[\boldsymbol{\mu}]}} \leq c, \boldsymbol{\mu} \succcurlyeq 0, \mathbb{1}^T\boldsymbol{\mu} = 1 \right\},$$

5. Cette distinction ne doit pas être confondue avec le nombre d'étapes mentionnées dans le titre de [Cortes *et coll.*, 2010b]. L'algorithme proposé, dans lequel l'apprentissage se déroule en deux étapes, correspond ici à un algorithme en un temps (séquentiel).

où $\mathcal{H}_{[\boldsymbol{\mu}]}$ est le RKHS engendré par le noyau multiple linéaire de pondération $\boldsymbol{\mu}$. Soient aussi $\mathfrak{D}'$ une densité de probabilité sur $\mathcal{X}$, X une variable aléatoire suivant $\mathfrak{D}'$ et un réel positif R tel que :

$$\forall \boldsymbol{\theta} \in \mathcal{A}, \forall \boldsymbol{x} \in \mathcal{X} \colon k_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{x}) \leq R^2.$$

Alors, pour tout échantillon de taille n d'observations iid de X, la complexité de Rademacher de $\mathcal{F}$ est bornée de la manière suivante :

$$\mathcal{R}_n(\mathcal{F}) \leq c\sqrt{\frac{23\mathrm{e}^1\lceil \ln(d) \rceil R^2}{22n}}.$$

Bien que la majoration que nous rapportons soit réservée aux noyaux multiples convexes, les résultats de [Cortes *et coll.*, 2010a] sont en réalité plus larges. Dans la suite, les fondations du courant MKL sont données ainsi que les jalons traçant le lit de l'approche fondée sur le risque régularisé. Celle-ci nous intéresse particulièrement puisque nous avons fait le choix de la mettre au cœur d'une partie de nos travaux (chapitre 3). L'annexe B reprend ce fil historique de manière plus détaillée.

Historique

L'idée de combiner des noyaux entre eux (en particulier lorsque les données sont de natures différentes) naît au début des années 2000 avec [Pavlidis *et coll.*, 2001, Cristianini *et coll.*, 2002]. Dans ces travaux, la pondération de la combinaison linéaire des noyaux est soit fixe [Pavlidis *et coll.*, 2001], soit déterminée de manière heuristique (fondée sur l'alignement de noyaux dans [Cristianini *et coll.*, 2002]). L'introduction de techniques d'apprentissage intervient avec [Bennett *et coll.*, 2002, Crammer *et coll.*, 2003] (approches par *boosting*) et [Lanckriet *et coll.*, 2002] (programmation semi-définie positive) et prend son réel essor avec [Lanckriet *et coll.*, 2004]. Le dernier article cité est une étude générale, traitant de plusieurs critères tels que l'alignement de noyaux et le risque régularisé. Dans le cas de ce dernier, les auteurs montrent que le problème MKL, pour une combinaison conique, peut être exprimé sous la forme d'un QCQP. Ce type de problème n'est cependant accessible que pour un faible nombre de variables à optimiser. Désireux de mettre en place un algorithme efficace fondé sur le principe d'optimisation minimale séquentielle (*Sequential Minimal Optimization*, SMO), déjà utilisé pour résoudre efficacement le problème dual d'une SVM [Platt, 1999], les auteurs de [Bach *et coll.*, 2004] récrivent le problème d'apprentissage de noyau multiple par minimisation du risque régularisé énoncé par [Lanckriet *et coll.*, 2004] sous la forme d'un problème dual de type SOCP. Celui-ci est intimement lié à une formulation SVM munie d'une régularisation ℓ_1 par blocs (promouvant une sélection parcimonieuse de noyaux, appelés noyaux supports [Bach *et coll.*, 2004]). Cependant, la fonction objectif du problème d'optimisation énoncé n'étant pas différentiable partout, les auteurs de [Bach *et coll.*, 2004] y ajoutent une régularisation de Moreau-Yosida [Lemaréchal et Sagastizábal, 1997]. La différentiabilité de la fonction objectif étant assurée, il est alors possible d'obtenir une solution approchée par un algorithme efficace de type SMO. Par la suite, abandonnant la contrainte en norme ℓ_1 pour une régularisation en norme ℓ_p ($p > 1$) ou grâce à certaines divergences de Bregman, Vishwanathan *et coll.* montrent qu'il est alors possible d'appliquer un algorithme SMO, sans utilisation d'une régularisation de Moreau-Yosida (supposée coûteuse à calculer) [Vishwanathan *et coll.*, 2010].

Comme nous l'avons vu, deux difficultés sont soulevées par la formulation de [Lanckriet *et coll.*, 2004] : l'efficacité des solveurs QCQP disponibles et la non-différentiabilité du problème. Si [Bach *et coll.*, 2004] propose de résoudre un problème différentiable mais approché, les auteurs de [Sonnenburg *et coll.*, 2006a, Sonnenburg *et coll.*, 2006b] adoptent une voie

différente en reformulant le problème MKL comme un SILP. Ce problème d'optimisation SILP peut être résolu par un solveur LP librement accessible, couplé ici à une technique de génération de colonnes nécessitant seulement un algorithme SVM standard (QP). Les auteurs l'ont appliqué avec succès à des thématiques biologiques avec plus de 100 000 points d'entrée et plusieurs centaines de noyaux.

Il existe encore d'autres manières de lever l'indifférentiabilité de la fonction objectif du problème MKL. Ainsi, les auteurs de [Rakotomamonjy *et coll.*, 2008] remarquent (indépendamment de [Zien et Ong, 2007] qui mentionne une remarque similaire dans le cas d'un problème multiclasse) que la régularisation ℓ_1 par blocs de [Bach *et coll.*, 2004] correspond à un point stationnaire d'un sous-problème d'optimisation différentiable et convexe. En intégrant ce dernier à la formulation de [Bach *et coll.*, 2004], on obtient un nouveau problème d'optimisation pour l'apprentissage d'un noyau multiple, lui aussi différentiable et convexe :

$$\begin{array}{ll} \underset{\boldsymbol{\mu}\in\mathbb{R}^{\mathcal{A}}}{\text{minimiser}} & J_{\text{lin}}(\boldsymbol{\mu}) \\ \text{tel que} & \left\{ \begin{array}{l} \boldsymbol{\mu} \succcurlyeq 0 \\ \mathbb{1}^T\boldsymbol{\mu} = 1, \end{array} \right. \end{array} \tag{1.3}$$

où $J_{\text{lin}}(\boldsymbol{\mu}) = J_{\text{RR}}\left(\sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}\right)$. Le gradient de J_{lin} est aisément calculable grâce au théorème suivant (essentiellement dû à Danskin).

Théorème 1.4.2 (Dérivabilité de l'encapsulage [Bonnans et Shapiro, 1998, théo. 4.1])**.**
Soient $\mathcal{A}$ un espace métrique compact, $\mathcal{B}$ un espace normé et $J\colon \mathcal{A}\times\mathcal{B}\to\mathbb{R}$ une fonction continue. Supposons que pour tout a de $\mathcal{A}$, $J(a,\cdot)$ est différentiable et que son gradient $\nabla_b J$ est continu sur $\mathcal{A}\times\mathcal{B}$. Alors la fonction $\tilde{J}\colon b\in\mathcal{B}\mapsto \min_{a\in\mathcal{A}} J(a,b)$ est différentiable. De plus, si le minimum de $J(\cdot,b)$ est atteint en un unique point $\bar{a}\in\mathcal{A}$, alors $\nabla\tilde{J}(b) = \nabla_b J(\bar{a},b)$.

Ainsi, le problème (1.3) est résolu par une descente de gradient réduit dans [Rakotomamonjy *et coll.*, 2008]. Cette approche est étendue à une combinaison d'un très grand nombre de noyaux par une technique d'ensemble actif dans [Yger et Rakotomamonjy, 2011]. Celle-ci consiste à tirer profit de la contrainte parcimonieuse en identifiant très tôt les noyaux non-pertinents (ceux pour lesquels $\mu_{\boldsymbol{\theta}} = 0$) et en faisant évoluer un ensemble (dit *actif*) de noyaux vers l'ensemble optimal (ceux pour lesquels $\mu_{\boldsymbol{\theta}} > 0$).

Comme il l'a été clairement résumé dans [Vishwanathan *et coll.*, 2010], les premières formulations MKL (sous forme d'un SDP ou d'un QCQP [Lanckriet *et coll.*, 2002, Lanckriet *et coll.*, 2004]) ont tiré profit de solveurs génériques existants et étaient par conséquent réservés à des problèmes de petite taille. L'introduction d'une technique fondée sur SMO de concert avec une régularisation de Moreau-Yosida [Bach *et coll.*, 2004] a ouvert la porte à des problèmes de taille moyenne. La formulation SILP de [Sonnenburg *et coll.*, 2006a, Sonnenburg *et coll.*, 2006b] a repoussé une nouvelle fois les limites en rendant possible le traitement d'un million de points avec quelques dizaines de noyaux. Malheureusement, la précédente approche passe difficilement à l'échelle par rapport au nombre de noyaux. L'introduction de méthodes par encapsulage telles que [Rakotomamonjy *et coll.*, 2008] (et [Bach, 2009, Varma et Babu, 2009] détaillées ci-après) puis par ensemble actif [Yger et Rakotomamonjy, 2011] permettent alors de considérer un grand nombre de noyaux lorsque la quantité de points est raisonnable. Des ultimes extensions [Vishwanathan *et coll.*, 2010, Jain *et coll.*, 2012] vont encore plus loin en adaptant la précision de résolution du problème SVM interne (de sorte qu'elle reste suffisante pour calculer un gradient informatif) ainsi que la recherche en ligne permettant de choisir le pas de descente.

Il existe une multitude d'autres approches MKL, comme une application du *boosting* [Bi *et coll.*, 2004] ou encore une contrainte en norme ℓ_p plutôt que ℓ_1 sur les poids des noyaux, privilégiant une combinaison plutôt qu'une sélection [Kloft *et coll.*, 2011]. Une autre approche modifie plus en amont la contrainte de parcimonie du problème MKL usuel : en

supposant les noyaux générateurs de trace unitaire, la condition $\mathbb{1}^T\boldsymbol{\mu} = 1$ correspond à $\mathrm{Tr}(\boldsymbol{K}_{[\boldsymbol{\mu}]_+}) = 1$. La partie de gauche de cette dernière inégalité est une borne supérieur de la complexité de Rademacher de l'ensemble d'hypothèses MKL. Certains auteurs proposent de conserver cette contrainte en normalisant les noyaux générateurs non pas par leur trace mais par la somme de leurs plus petites valeurs propres. Ceci a pour effet de considérer une complexité de Rademacher locale et conduit à une borne plus fine de l'écart entre l'erreur optimale et l'erreur empirique [Cortes *et coll.*, 2013].

1.4.4 Apprentissage de noyau multiple généralisé

Les combinaisons coniques ont été le centre d'une grande attention car la non-convexité des problèmes d'apprentissage de noyaux en général peut être levée, tout en laissant subsister des difficultés telles que la non-différentiabilité [Bach *et coll.*, 2004] et la complexité calculatoire [Sonnenburg *et coll.*, 2006a,Sonnenburg *et coll.*, 2006a]. Néanmoins, il est intéressant de remarquer deux grandes généralisations de ces travaux : l'apprentissage de combinaisons non-linéaires et la considération d'une infinité de noyaux générateurs.

Noyaux non-linéaires

Les fondements de l'apprentissage de noyaux multiples multiplicatifs trouvent racines dans [Weston *et coll.*, 2001,Grandvalet et Canu, 2003]. Dans ces travaux, les auteurs ne présentent pas directement d'applications à l'apprentissage d'un produit de noyaux mais le cadre mis en place le permet de manière évidente si l'on considère un noyau gaussien. Le sujet fondamental dont il est question dans ces deux études est la sélection de variables descriptives. Concrètement, tout vecteur caractéristique est redécrit par l'application $\boldsymbol{x} \mapsto \mathrm{Diag}(\sqrt{\boldsymbol{\mu}})\boldsymbol{x}$, où $\sqrt{\cdot}$ correspond à la racine carrée composante à composante et $\boldsymbol{\mu}$ est un vecteur de pondération de l'orthant positif à apprendre, sélectionnant une coordonnée $\boldsymbol{\theta}$ si $\mu_{\boldsymbol{\theta}} = 1$ et l'annihilant si $\mu_{\boldsymbol{\theta}} = 0$. Puisque considérer un vecteur à variables entières nécessite de parcourir toutes les solutions, il est proposé dans [Weston *et coll.*, 2001] de considérer un vecteur $\boldsymbol{\mu}$ réel vivant dans l'orthant positif et affublé de la contrainte $\sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}}^p = \delta$ ($p = 1$ dans [Weston *et coll.*, 2001] et $p = 2$ dans [Grandvalet et Canu, 2003]), où δ renseigne sur la quantité de variables à sélectionner. Considérer un noyau gaussien de concert avec l'application de redescription précédemment citée est ainsi équivalent à apprendre les exposants d'un noyau multiple multiplicatif construit à partir de noyaux gaussiens possédant chacun son propre rayon et associé à une coordonnée particulière.

Les différences principales entre l'approche de [Weston *et coll.*, 2001] et celle de [Grandvalet et Canu, 2003] résident d'une part dans la fonction de perte utilisée dans le modèle SVM (quadratique pour [Weston *et coll.*, 2001] et charnière pour [Grandvalet et Canu, 2003]) ainsi que dans le critère à minimiser (borne rayon marge `vs` risque régularisé). La formulation de [Weston *et coll.*, 2001] conduit naturellement à utiliser une technique de descente de gradient (apprentissage en un temps séquentiel) tandis que [Grandvalet et Canu, 2003] met au point une descente alternée comprenant la minimisation d'une approximation du coût SVM pour la mise à jour de $\boldsymbol{\mu}$ (apprentissage en deux temps).

Les travaux présentés dans [Weston *et coll.*, 2001] ont été étendus par Chapelle *et coll.* avec pour but premier d'apprendre les rayons d'un noyau gaussien ou polynomial anisotrope (et non de sélectionner des variables discriminantes). Ces derniers suppriment la contrainte $\sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}}^p = \delta$ (introduite pour favoriser la sélection de peu de variables) et comparent de nombreux critères à minimiser parmi lesquels les erreurs de validation simple et multiple, ainsi que la borne rayon-marge [Chapelle *et coll.*, 2002]. À l'instar des précédentes, cette approche est apparentée au paradigme MKL avec un noyau multiple multiplicatif lors-

qu'utilisée avec un noyau gaussien.

L'apprentissage d'un noyau multiple multiplicatif a été clairement étudié comme tel dans [Varma et Babu, 2009]. Les auteurs de [Varma et Babu, 2009] étendent les travaux de [Rakotomamonjy *et coll.*, 2008] au produit de noyaux (plus particulièrement de noyaux gaussiens). Le problème de minimisation mis en place est

$$\begin{array}{ll} \underset{\boldsymbol{\mu}}{\text{minimiser}} & J_{\text{prod}}(\boldsymbol{\mu}) + \rho(\boldsymbol{\mu}) \\ \text{tel que} & \boldsymbol{\mu} \succcurlyeq 0, \end{array} \tag{1.4}$$

où $J_{\text{prod}}(\boldsymbol{\mu}) = J_{\text{RR}}\left(\prod_{\boldsymbol{\theta}\in\mathcal{A}} k_{\boldsymbol{\theta}}^{\mu_{\boldsymbol{\theta}}}\right)$ et ρ est une fonction de régularisation dérivable (plus particulièrement linéaire ou quadratique). Malgré la non-convexité du problème (1.4), les auteurs ont choisi de conserver l'approche par encapsulation introduite dans [Rakotomamonjy *et coll.*, 2008] pour la résolution du problème MKL sous la forme (1.3). Les différences entre les formulations (1.3) et (1.4) sont d'une part le type de noyau multiple (convexe `vs` multiplicatif) et d'autre part la régularisation ρ dans (1.4) (*a priori* quelconque tant que dérivable) qui apparaît comme contrainte en norme ℓ_1 dans (1.3). Par la suite, l'algorithme de descente de gradient initialement utilisé dans [Varma et Babu, 2009] pour résoudre localement (1.4) a été accéléré par une méthode de gradient projeté spectral avec différentes heuristiques contribuant elles aussi à l'accélération [Jain *et coll.*, 2012].

Plus récemment, l'apprentissage d'une combinaison polynomiale de noyaux a été étudié [Cortes *et coll.*, 2009, Bach, 2009]. Dans l'apprentissage de noyau hiérarchique [Bach, 2009], on possède un nombre exponentiel de noyaux, organisés au sein d'un graphe direct et acyclique. En exploitant cette structure de graphe combinée à une pénalisation parcimonieuse hiérarchique [Szafranski *et coll.*, 2008], les auteurs proposent un algorithme capable de résoudre un tel problème d'apprentissage avec une complexité temporelle polynomiale.

Il est intéressant de remarquer que l'apprentissage de noyau prend généralement la forme d'un problème de minimisation d'un risque empirique régularisé (risque régularisé, alignement de noyau, probabilité *a posteriori*, *etc.*). Forts de ce constat, Ong *et coll.* introduisent la notion d'hyper-RKHS, définit comme un RKHS pour lequel les fonctionnelles sont des noyaux (*i.e.* un RKHS de noyaux) [Ong *et coll.*, 2003, Ong *et coll.*, 2005, Tsang et Kwok, 2006].

Enfin une dernière généralisation réside dans la notion de noyau multiple localisé [Gönen et Alpaydin, 2008]. Dans ces travaux, les auteurs reprennent le principe d'alternance de [Rakotomamonjy *et coll.*, 2008] en remplaçant le vecteur $\boldsymbol{\mu}$ par une pondération dépendante des données $\mu_{\boldsymbol{\theta}} : \mathcal{X} \to \mathbb{R}$. Le noyau multiple devient alors $k_{[\mu]} : (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X} \times \mathcal{X} \mapsto \sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}}(\boldsymbol{x})\mu_{\boldsymbol{\theta}}(\boldsymbol{z})k_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})$.

Noyaux infinis

La combinaison convexe de noyaux a très tôt attiré l'attention [Lanckriet *et coll.*, 2002, Lanckriet *et coll.*, 2004, Bach *et coll.*, 2004], et ce pour différentes raisons :

- ◇ les critères majoritairement utilisés tels que l'alignement de noyaux et le risque régularisé conduisent à la mise en place de problèmes convexes par rapport au noyau global [Lanckriet *et coll.*, 2002, Micchelli et Pontil, 2005]. Dans ce cas, toute non-convexité provient de la paramétrisation du noyau. Une paramétrisation linéaire permet de conserver la convexité du problème ;
- ◇ imposer que le vecteur $\boldsymbol{\mu}$ appartienne à une boule unité permet d'éviter le surapprentissage dû au problème de normalisation [Gai *et coll.*, 2010] ;
- ◇ une telle combinaison est empiriquement efficace.

Il est donc d'intérêt de proposer des extensions du concept MKL dans cette direction, bénéficiant des propriétés avantageuses que l'on vient d'énumérer.

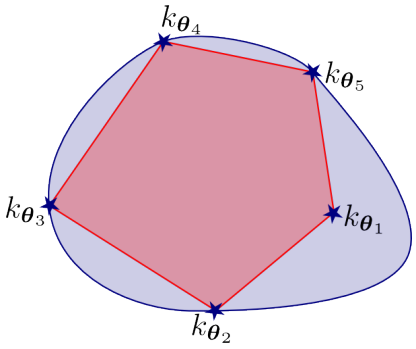


FIGURE 1.3 – Schématisation des enveloppes convexes issues de cinq noyaux $(k_{\boldsymbol{\theta_1}}, \ldots, k_{\boldsymbol{\theta_5}})$ ($\forall i \in \mathbb{N}_5 : \boldsymbol{\theta}_i \in \mathcal{P}$), en rouge, et de la famille complète et infinie $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}}$, en bleu.

D'un point de vue ensembliste, considérer une combinaison convexe de noyaux générateurs $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ est identique à chercher un noyau à l'intérieur de l'enveloppe convexe de l'ensemble de noyaux $\{k_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{A}}$. Micchelli et Pontil étendent donc naturellement cette notion à l'enveloppe convexe d'un ensemble infini (et non-nécessairement dénombrable) de noyaux paramétrés par un vecteur $\boldsymbol{\theta}$ d'un espace compact $\mathcal{P}$ (illustration 1.3) [Micchelli et Pontil, 2005]. La combinaison de noyaux peut s'exprimer $k_{[\mu]} = \int_{\mathcal{P}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}\, d\boldsymbol{\theta}$, où μ est une mesure de probabilité sur $\mathcal{P}$. Par abus d'écriture, on considère la fonction μ comme un vecteur de dimension infinie $\boldsymbol{\mu}$ de $\mathbb{R}_+^{\mathcal{P}}$ et on écrit :

$$k_{[\boldsymbol{\mu}]} = \sum_{\theta\in\mathcal{P}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}, \quad \mathbb{1}^T \boldsymbol{\mu} = 1.$$

Il est alors démontré, dans le cas d'un problème convexe, que tout noyau solution est une combinaison convexe d'au plus $n+2$ noyaux issus de $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}}$ (où n est le nombre de points d'entrée) [Micchelli et Pontil, 2005, théo. 7]. Une approche concrète de ce concept est proposée dans [Argyriou *et coll.*, 2005, Argyriou *et coll.*, 2006]. Dans [Argyriou *et coll.*, 2005], les auteurs s'intéressent à la famille des noyaux gaussiens paramétrés par un scalaire. Ces travaux sont ensuite étendus dans [Argyriou *et coll.*, 2006] aux noyaux exprimés comme une différence de fonctions convexes (et particulièrement un noyau gaussien anisotrope paramétré par une matrice de covariance). Les auteurs mettent au point un algorithme glouton (un temps) qui alterne résolution SVM à noyau multiple fixé, ajout d'un nouveau noyau et mise à jour des poids. La deuxième étape est résolue par un algorithme de plans sécants suggéré par la théorie de l'optimisation de différence de fonctions convexes.

Indépendamment de ces travaux, Gehler *et coll.* développent une méthode semblable en reprenant la formulation convexe de [Rakotomamonjy *et coll.*, 2008] et en donnant la possibilité à l'ensemble fini de noyaux $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ d'être automatiquement extrait de $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}}$ [Gehler et Nowozin, 2008a]. Le problème d'optimisation d'intérêt est alors exprimé dans le domaine dual par :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha},\lambda}{\text{maximiser}} & \sum_{i=1}^n \alpha_i - \lambda \\ \text{tel que} & \begin{cases} 0 \leq \alpha_i \preccurlyeq C, \quad \forall i \in \mathbb{N}_n \\ \sum_{i=1}^n y_i \alpha_i = 0 \\ \sum_{1\leq i,j\leq n} \alpha_i \alpha_j y_i y_j k_{\boldsymbol{\theta}}(\boldsymbol{x_i}, \boldsymbol{x_j}) \leq \lambda, \quad \forall \boldsymbol{\theta} \in \mathcal{P}. \end{cases} \end{array} \tag{1.5}$$

On reconnaît dans (1.5) d'une part un SILP (la différence avec la formulation de [Sonnenburg *et coll.*, 2006a] réside dans la contrainte infinie qui porte sur le vecteur $\boldsymbol{\alpha}$ pour ce dernier et sur le vecteur des paramètres $\boldsymbol{\theta}$ dans (1.5)) et d'autre part le dual du problème MKL exposé dans [Rakotomamonjy *et coll.*, 2008] si l'on remplace $\mathcal{P}$ par l'ensemble fini $\mathcal{A}$. Ceci justifie la mise en place d'un algorithme de génération de colonne (en deux temps) qui alterne la résolution d'un problème MKL (en remplaçant $\mathcal{P}$ par une ensemble $\mathcal{A}$ fini et fixé) et

la mise à jour de l'ensemble de noyaux actifs $\mathcal{A}$ (suppression des noyaux à pondération $\mu_{\boldsymbol{\theta}}$ nulle et ajout d'un nouveau noyau violant les conditions courantes d'optimalité). Une autre approche similaire d'apprentissage de noyau infini, fondée sur les travaux de [Sonnenburg *et coll.*, 2006b], est présentée dans [Özöğür Akyüz et Weber, 2008, Özöğür Akyüz et Weber, 2010a]. Des algorithmes de résolution sont alors proposés dans [Özöğür Akyüz et Weber, 2010b].

1.5 APPRENTISSAGE D'INSTANCE MULTIPLE

1.5.1 Définition

Un problème d'apprentissage statistique usuel suppose que l'on ait accès à un ensemble d'apprentissage $\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}$, dans lequel les observations $(\boldsymbol{x_i}, y_i)$ sont issues de $\mathcal{X}\times\mathcal{Y}$. Le but premier est alors d'inférer une règle $f\colon \mathcal{X}\to\mathcal{Y}$ permettant d'établir un lien entre les données explicatives $\boldsymbol{x_i}$ et les données expliquées y_i. Dans une approche de type SVM, cette inférence est réalisée indépendamment de la distribution statistique régissant le couple de variables aléatoires dont sont issues les observations à notre disposition.

Dans de nombreuses applications, les étiquettes $(y_i)_{1\leq i\leq n}$ sont attribuées par l'expert par défaut, car celui-ci ne possède pas d'information réelle sur les données $(\boldsymbol{x_i})_{1\leq i\leq n}$. Cette ambiguïté naît d'une connaissance située à l'échelle d'un ensemble de données, plutôt qu'au niveau de la donnée elle-même. Ce paradigme, nommé apprentissage d'instances multiples (*Multiple Instance Learning*, MIL), suppose que les étiquettes connues sont attribuées à des groupes de données plutôt qu'aux données elles-mêmes. On parle alors de sacs d'instances. Ce regroupement fait, il existe deux variantes du paradigme MIL : l'étiquetage de sacs [Wang et Zucker, 2000, Gärtner *et coll.*, 2002] ou d'instances inédites.

Définition 1.5.1 (Ensemble de sacs d'apprentissage)**.**
Soient (X, Y) un couple de variables aléatoires distribué selon $\mathfrak{D}$ et $\{(\boldsymbol{x_j}, y_j)\}_{1\leq j\leq m}$ un échantillon de m réalisations de la variable aléatoire (X, Y) tirées indépendamment. Soient $(\mathcal{B}_i)_{1\leq i\leq n}$ une partition de $\mathbb{N}_m$ et $(Y_i)_{1\leq i\leq n}$ le n-uplet réel défini par

$$\forall i\in\mathbb{N}_n\colon Y_i = \max_{j\in\mathcal{B}_i} y_j.$$

$\left\{\left(\{\boldsymbol{x_j}\}_{j\in\mathcal{B}_i}, Y_i\right)\right\}_{1\leq i\leq n}$ est appelé ensemble de sacs d'apprentissage.

Pour un problème binaire (*i.e.* à deux classes), un sac $\{\boldsymbol{x_j}\}_{j\in\mathcal{B}_i}$ (noté par abus d'écriture $\mathcal{B}_i$) est étiqueté positivement ssi (au moins) une instance de $\mathcal{B}_i$ est étiquetée positivement (figure 1.4 page ci-contre). Réciproquement, un sac étiqueté négativement ne contient que des instances de la classe -1. La difficulté du problème MIL réside dans la non-connaissance des variables latentes $(y_i)_{1\leq i\leq m}$, sur lesquelles repose fondamentalement l'étiquetage des sacs. Il faut bien comprendre que si l'étiquette d'un sac négatif est représentative de toutes les instances qu'il contient (elles sont toutes étiquetées négativement), il n'en est pas de même concernant les sacs étiquetés positivement, car ceux-ci contiennent simultanément des instances de la classe $+1$ et de la classe -1.

Cette notion de paradigme ambiguë est apparue avec les travaux de Keeler *et coll.* sur la segmentation et la reconnaissance automatiques de chiffres manuscrits [Keeler *et coll.*, 1991]. Lors de l'apprentissage de leur réseau de neurones, Keeler *et coll.* présentent un ensemble de sous-régions, étiqueté par le chiffre inscrit sur l'image dont sont issues ces dernières. L'algorithme doit simultanément sélectionner la meilleure segmentation et entraîner le réseau de neurones à la reconnaissance des chiffres. Le concept a été développé et nommé

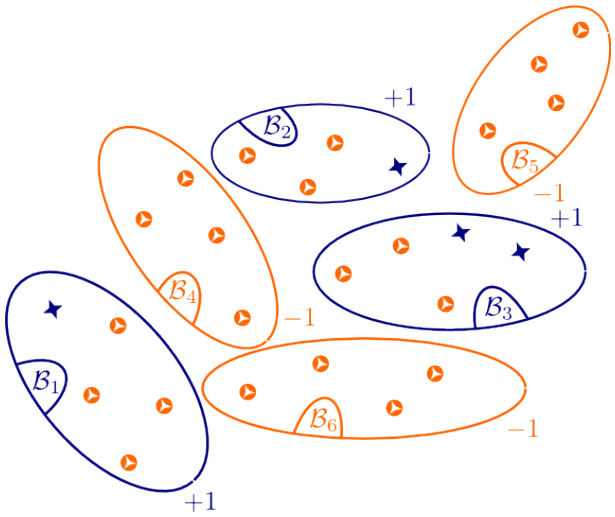


FIGURE 1.4 – Illustration d'un ensemble de sacs d'apprentissage pour le paradigme MIL. Un sac est étiqueté positivement ssi il contient une instance d'étiquette positive (bleue).

MIL par Dietterich *et coll.* [Dietterich *et coll.*, 1997]. Leur étude est centrée autour de la prédiction d'efficacité d'un médicament. Celle-ci dépend de la forme de la molécule active. L'ambiguïté provient des multiples formes que peut prendre une même molécule, sans que l'on sache quelles formes rendent effectif le médicament. Au fil des années, le paradigme MIL a été appliqué à la détection d'objets dans les images, la reconnaissance de personnes sur vidéo, la classification de texte, la catégorisation d'images et de musiques, la sélection d'actions financières et la sécurité [Mandel et Ellis, 2008, Chai *et coll.*, 2014].

1.5.2 Algorithmes

Depuis le lancement de la problématique MIL, introduite par Dietterich *et coll.* à la fin des années 1990 [Dietterich *et coll.*, 1997], il existe un effort continu de recherche qui a conduit à de nombreux travaux. Parmi ceux-ci, on peut rapidement citer les premières garanties théoriques établies sur le paradigme MIL [Blum et Kalai, 1998], une approche probabiliste mesurant la probabilité d'occurrence d'une instance dans différents sacs [Maron et Lozano-Pérez, 1998], une application du *boosting* [Viola *et coll.*, 2006] et des adaptations de l'analyse discriminante de Fisher [Ping *et coll.*, 2010, Chai *et coll.*, 2014]. Loin de vouloir passer l'ensemble des contributions en revue ici, nous nous contentons de décrire les principaux algorithmes vaste marge qui ont jalonné ces recherches et laissons le lecteur à la récente taxonomie mise en place par Amores et décrite dans [Amores, 2013], pour une vision plus détaillée.

À l'instar de la première approche probabiliste [Maron et Lozano-Pérez, 1998], le paradigme SVM est un outil adéquat pour interpréter (géométriquement) le problème MIL. L'une des premières applications est le noyau ensembliste de Gärtner *et coll.* [Gärtner *et coll.*, 2002]. Celui-ci associe à chaque sac $\mathcal{B}$ un vecteur de statistiques $\phi_{\text{stat}}(\mathcal{B})$. Les auteurs proposent en particulier de caractériser un sac $\mathcal{B}$ par la plus petite boîte l'englobant. En supposant que $\mathcal{X} = \mathbb{R}^d$ et en notant x_{jl} ($j \in \mathbb{N}_m, l \in \mathbb{N}_d$) la l^{e} coordonnée du vecteur d'apprentissage $\boldsymbol{x_j}$, ϕ_{stat} est définie selon :

$$\phi_{\text{stat}} \colon \mathcal{B} \in 2^{\mathbb{N}_n} \mapsto \left[\min_{j\in\mathcal{B}} x_{j1}, \ldots, \min_{j\in\mathcal{B}} x_{jd}, \max_{j\in\mathcal{B}} x_{j1}, \ldots, \max_{j\in\mathcal{B}} x_{jd}\right]^T \in \mathbb{R}^{2d}.$$

Partant d'un noyau k (polynomial dans [Gärtner *et coll.*, 2002]), un noyau ensembliste,

nommé *noyau minimax* k_{MM}, peut être défini par :

$$k_{\text{MM}} \colon (\mathcal{B}_1, \mathcal{B}_2) \in 2^{\mathbb{N}_n} \times 2^{\mathbb{N}_n} \mapsto k\left(\phi_{\text{stat}}(\mathcal{B}_1), \phi_{\text{stat}}(\mathcal{B}_2)\right).$$

La méthode proposée par Gärtner *et coll.* est l'une des rares a être spécifiquement orientée vers la prédiction de sacs plutôt que d'instances.

Andrews *et coll.* mettent en place deux adaptations du problème d'optimisation SVM visant à gérer l'ambiguïté inhérente au paradigme MIL : mi-SVM et MI-SVM [Andrews *et coll.*, 2003]. La première propose de retrouver les variables latentes $(y_i)_{1 \leq i \leq m}$:

$$\begin{array}{ll} \underset{\substack{\boldsymbol{y} \in \{0,1\}^m, \\ f \in \mathcal{H},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^m}}{\text{minimiser}} & \dfrac{1}{2}\|f\|_{\mathcal{H}}^2 + C \sum_{j=1}^{m} \xi_i \\ \text{tel que} & \begin{cases} y_j \left(f(\boldsymbol{x_j}) + b\right) \geq 1 - \xi_j, \forall j \in \mathbb{N}_m \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \forall i \in \mathbb{N}_n \,/\, Y_i = -1, \quad \forall j \in \mathcal{B}_i \colon y_j = -1 \\ \forall i \in \mathbb{N}_n \,/\, Y_i = +1, \quad \sum_{j \in \mathcal{B}_i} y_j \geq 1 - \text{Card}(\mathcal{B}_i). \end{cases} \end{array} \tag{1.6}$$

Le problème ainsi formulé est très ressemblant à celui d'une SVM usuelle et directement fondé sur les instances. La différence réside dans la présence de variables entières ($(y_i)_{1 \leq i \leq m}$) et dans l'ajout des deux dernières contraintes. La pénultième affirme que toute instance d'un sac étiqueté négativement est aussi étiquetée par -1, tandis que la dernière assure qu'il existe au moins une instance étiquetée $+1$ dans un sac d'étiquette positive.

La deuxième déclinaison (MI-SVM) aborde une vision orientée sur les sacs eux-mêmes en exhibant une marge entre sacs plutôt qu'entre instances. Cette marge est estimée de manière sur-évaluée à partir de l'instance d'étiquette positive la plus éloignée de l'hyperplan. L'idée sous-jacente à ce choix est qu'en assurant une instance d'étiquette positive bien classée (la plus éloignée de l'hyperplan), on assure *a fortiori* que tous les sacs étiquetés $+1$ possèdent au moins une instance de même étiquette. Les auteurs formulent un tel problème d'optimisation de la manière suivante :

$$\begin{array}{ll} \underset{f \in \mathcal{H},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n}{\text{minimiser}} & \dfrac{1}{2}\|f\|_{\mathcal{H}}^2 + C \sum_{j=1}^{n} \xi_i \\ \text{tel que} & \begin{cases} Y_i \left(\max\limits_{j \in \mathcal{B}_i} f(\boldsymbol{x_j}) + b \right) \geq 1 - \xi_i, \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0. \end{cases} \end{array}$$

La première contrainte impose que toutes les instances des sacs étiquetés négativement sont « à gauche de la marge » : $\forall i \in \mathbb{N}_n \,/\, Y_i = -1, \quad \forall j \in \mathcal{B}_i \colon f(\boldsymbol{x_j}) + b \leq -1 + \xi_i$. Simultanément, elle assure que l'instance d'étiquette $+1$ la plus éloignée de l'hyperplan est « à droite de la marge » : $\forall i \in \mathbb{N}_n \,/\, Y_i = +1, \quad \max\limits_{j \in \mathcal{B}_i} \left(f(\boldsymbol{x_j}) + b\right) \geq 1 - \xi_i$. Malheureusement, cette contrainte n'est pas convexe. Andrews *et coll.* proposent donc la formulation équivalente suivante :

$$\begin{array}{ll} \underset{\substack{\boldsymbol{\delta} \in \mathcal{B}_1 \times \cdots \times \mathcal{B}_n, \\ f \in \mathcal{H},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^m}}{\text{minimiser}} & \dfrac{1}{2}\|f\|_{\mathcal{H}}^2 + C \sum_{j=1}^{m} \xi_i \\ \text{tel que} & \begin{cases} \forall i \in \mathbb{N}_n \,/\, Y_i = -1, \quad \forall j \in \mathcal{B}_i \colon f(\boldsymbol{x_j}) + b \leq -1 + \xi_j \\ \forall i \in \mathbb{N}_n \,/\, Y_i = +1, \qquad\qquad f(\boldsymbol{x_{\delta_i}}) + b \geq 1 - \xi_{\delta_i} \\ \boldsymbol{\xi} \succcurlyeq 0, \end{cases} \end{array} \tag{1.7}$$

dans laquelle δ_i est interprété comme l'indice du meilleur témoin (*i.e.* une instance d'étiquette $y_{\delta_i} = +1$) du sac $\mathcal{B}_i$. Il est important de remarquer que dans cette formulation, les instances d'étiquettes négatives des sacs $\mathcal{B}_i$ étiquetés $Y_i = +1$ sont ignorées.

Dans les deux cas, (1.6) et (1.7) se trouvent être des problèmes d'optimisation quadratiques à variables continues et entières, donc difficiles à résoudre. Une étude développée plus tard par les mêmes auteurs se concentre sur la nature du problème d'optimisation mis en jeu et aboutit à un cadre d'apprentissage par *boosting* [Andrews et Hofmann, 2004].

Une autre manière de lever l'aspect combinatoire de (1.6) est d'accepter la non-connaissance des étiquettes $(y_i)_{1\leq i\leq m}$ et de ré-écrire les contraintes le plus fidèlement possible sans chercher à lever l'ambiguïté, conduisant ainsi au problème *MIL transductif et parcimonieux* [Bunescu et Mooney, 2007] :

$$\begin{array}{ll} \underset{\substack{f\in\mathcal{H},\, b\in\mathbb{R},\\ \boldsymbol{\xi}\in\mathbb{R}^m,\, \tilde{\boldsymbol{\xi}}\in\mathbb{R}^n}}{\text{minimiser}} & \dfrac{1}{2}\left\|f\right\|_{\mathcal{H}}^2 + C\sum_{j=1}^{m}\xi_i + \tilde{C}\sum_{j=1}^{n}\tilde{\xi}_i \\ \text{tel que} & \begin{cases} \forall i\in\mathbb{N}_n \,/\, Y_i=-1, \quad \forall j\in\mathcal{B}_i\colon\ f(\boldsymbol{x_j})+b \leq -1+\xi_j \\ \forall i\in\mathbb{N}_n \,/\, Y_i=+1, \quad \forall j\in\mathcal{B}_i\colon\ |f(\boldsymbol{x_j})+b| \geq +1-\xi_j \\ \qquad\qquad\qquad\qquad\quad \sum_{j\in\mathcal{B}_i} f(\boldsymbol{x_j})+b \geq 2-\mathrm{Card}(\mathcal{B}_i)-\tilde{\xi}_i \\ \boldsymbol{\xi}\succcurlyeq 0. \end{cases} \end{array} \tag{1.8}$$

Les premières contraintes de (1.6) et (1.8) sont semblables. La deuxième de (1.8) relâche la première de (1.6) dans le cas d'un sac étiqueté positivement en assurant une vaste marge, peu importe de quel « côté ». Enfin, la pénultième contrainte de (1.8) force la présence d'au moins une instance témoin dans chaque sac étiqueté positivement. Cette contrainte est parente de la dernière de (1.6) mais ne fait pas intervenir les étiquettes des instances. D'un point de vue théorique, le problème (1.8) n'est pas convexe, mais peut être résolu comme une différence de fonctions convexes [Yuille et Rangarajan, 2002, Sriperumbudur et Lanckriet, 2009]. En pratique, l'approche de Bunescu *et coll.* a montré de bonnes performances de reconnaissance pour des problèmes MIL *parcimonieux*, *i.e.* contenant peu de témoins dans les sacs $\mathcal{B}_i$ étiquetés par $Y_i=+1$.

Conformément au paradigme MIL, tous les travaux précédemment cités supposent qu'un sac étiqueté négativement ne contient que des instances de même étiquette. Or il est des situations dans lesquelles cette hypothèse peut s'avérer fausse (par exemple en catégorisation d'images). Chen et Wang proposent donc une extension des travaux probabilistes de Maron et Lozano-Pérez [Maron et Lozano-Pérez, 1998] permettant de répondre à une telle variante du problème MIL [Chen et Wang, 2004].

L'idée astucieuse de Chen et Wang réside dans l'association de chaque sac à un vecteur de similarité par rapport à un r-uplet $\mathcal{L}$ de prototypes de $\mathcal{X}$: $\mathcal{L}=(\boldsymbol{p_1},\ldots,\boldsymbol{p_r})$. En appelant $\tilde{k}\colon \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ une mesure de proximité, chaque sac $\mathcal{B}$ est associé à un vecteur suivant la fonction de redescription suivante :

$$\psi^{\mathcal{L}}\colon \mathcal{B}\in 2^{\mathbb{N}_n} \mapsto \begin{bmatrix} \max\limits_{j\in\mathcal{B}} \tilde{k}(\boldsymbol{x_j},\boldsymbol{p_1}) \\ \vdots \\ \max\limits_{j\in\mathcal{B}} \tilde{k}(\boldsymbol{x_j},\boldsymbol{p_r}) \end{bmatrix} \in \mathbb{R}^r.$$

La description des sacs dans un tel espace de similarité [Balcan et Blum, 2006, Pekalska et Duin, 2008] transforme la configuration MIL en un problème d'apprentissage supervisé usuel (dans lequel les observations sont des sacs étiquetés), sans encoder aucune relation entre sacs et instances. Dans les premiers travaux de Chen et Wang, l'ensemble de prototypes $\mathcal{L}$ est déterminé de manière probabiliste, conformément à la méthode introduite par Maron et Lozano-Pérez et un classifieur SVM régularisé en norme ℓ_2 est entraîné dans l'espace de similarité [Chen et Wang, 2004]. Suivant l'hypothèse de gaussianité introduite dans [Maron et Lozano-Pérez, 1998], la similarité entre une instance et un prototype est définie comme une distance pondérée par une matrice semi-définie positive $\boldsymbol{\Sigma}_+$:

$$\tilde{k}\colon (\boldsymbol{x},\boldsymbol{p})\in\mathcal{X}\times\mathcal{X}\mapsto(\boldsymbol{x}-\boldsymbol{p})^T\boldsymbol{\Sigma}_+(\boldsymbol{x}-\boldsymbol{p}).$$

En revanche, dans un second article de Chen *et coll.*, $\mathcal{L}$ est simplement défini comme l'ensemble de toutes les instances d'apprentissage à disposition. À la différence du cas précédent, c'est cette fois une SVM régularisée en norme ℓ_1 qui fait suite à la fonction de redescription [Chen *et coll.*, 2006]. La régularisation en parcimonie a alors pour avantage d'exhiber uniquement les instances discriminantes. La similarité est directement tirée du modèle probabiliste de [Maron et Lozano-Pérez, 1998] :

$$\tilde{k} \colon (\boldsymbol{x}, \boldsymbol{p}) \in \mathcal{X} \times \mathcal{X} \mapsto \mathrm{e}^{-\gamma \|\boldsymbol{x}-\boldsymbol{p}\|_{\ell_2}^2},$$

où γ est un paramètre positif à définir.

Bien que les auteurs aient choisi un classifieur SVM pour ces performances notoires dans de nombreuses applications de reconnaissance, ils ne mentionnent pas un point important : de manière générale appliquer une SVM dans un espace de représentation apparaît comme une généralisation du paradigme SVM à des noyaux non-nécessairement semi-définis positifs, ni même symétriques (voir section 4.2.1).

1.6 SYNTHÈSE

Il existe diverses techniques d'apprentissage automatique, parmi lesquelles les SVM. Grâce au cadre théorique de l'apprentissage statistique, le paradigme SVM se présente comme un outil pluridisciplinaire duquel sont nées des approches d'apprentissage de représentations (l'inférence de noyaux multiples) et d'intégration de l'ambiguïté inhérente à l'étiquetage d'images et d'enregistrements audio. Ces deux concepts définissent les contributions présentées dans ce manuscrit.

Notre première contribution (chapitre 3) consiste à fournir une nouvelle méthode d'apprentissage de noyau multiple. Dans celle-ci, on considère une combinaison multiplicative et parcimonieuse d'une infinité de noyaux. Cette approche fait le lien entre deux courants de recherche sur les noyaux multiples : celui des combinaisons finies non-linéaires [Varma et Babu, 2009] et celui des combinaisons infinies linéaires [Gehler et Nowozin, 2008a].

Notre seconde contribution (chapitre 4) met en lumière une variante du paradigme MIL, appliquée spécifiquement aux séries temporelles. En redéfinissant la notion de similarité et en modifiant l'approche présentée dans [Chen *et coll.*, 2006], il nous est possible de mettre au point un détecteur précoce d'événements. Celui-ci est accompagné d'une amélioration algorithmique par rapport à [Chen *et coll.*, 2006], permettant d'accélérer la résolution du problème d'optimisation, ainsi que d'une borne sur la complexité de Rademacher de la classe de détecteurs (obtenue directement de [Kakade *et coll.*, 2009]).

Le chapitre suivant dresse un état de l'art partiel de l'apprentissage de représentations Temps-Fréquence (TF) pour la reconnaissance de signaux et des techniques de reconnaissance précoce. Cette dernière est une application naturelle de l'apprentissage d'instance multiple qui lie la partie non-observée à une ambiguïté d'étiquetage.

CHAPITRE 2

Reconnaissance de signaux

2.1 INTRODUCTION

La reconnaissance de signaux est une application des principes d'apprentissage automatique (est plus précisément des techniques de classification, dont une partie primordiale pour nos travaux est présentée dans le chapitre précédent) aux séries temporelles. Elle consiste donc à assigner une étiquette à un *événement* (aussi appelé *séquence*), c'est à dire à un signal, souvent issus de la segmentation d'une longue série temporelle (voir figure 2.1 page 33). Contrairement à l'apprentissage automatique, l'accent de la reconnaissance de signaux n'est pas principalement porté sur l'outil de classification mais sur la nature des vecteurs caractéristiques (supposés donnés en apprentissage automatique) et sur l'intégration de la nouvelle dimension *temps*. Dans ce chapitre, nous nous efforçons donc de mettre en exergue deux traits spécifiques à la reconnaissance de signaux : la notion de représentation (au sens large) et la gestion du temps.

Le socle d'une chaîne de traitement de signaux est une représentation Temps-Fréquence (TF) (ou plus généralement Temps-Caractéristique (TC), dont une vue globale est donnée en section 2.2 et une revue détaillée en section 2.4), qui extrait d'un *signal* des caractéristiques dépendantes du temps. Afin d'éclaircir le discours, nous nous efforcerons de respecter la convention de vocabulaire suivante (qui n'est malheureusement pas en accord avec l'expression « reconnaissance de signaux ») : un *événement* est une séquence temporelle à laquelle nous désirons assigner une étiquette, tandis qu'un *signal* est une sous-partie d'un événement, sur laquelle est effectivement calculée la représentation TC.

La temporalité de ces caractéristiques est nécessaire puisque les signaux observés dans le monde réel sont non-stationnaires, *i.e.* de nature intrinsèque évoluant au fil du temps. En outre, les caractéristiques à extraire sont représentatives de l'action que l'expert souhaite automatiser. Ainsi, compression, débruitage et reconnaissance conduisent à des caractéristiques essentiellement différentes dans l'information qu'elles portent. Pour cette dernière application (la reconnaissance), on met alors en place une nouvelle représentation dont l'espace image peut être décrit de manière sémantique. Une telle représentation est nommée *règle de décision*, ou plus précisément *détecteur* ou *classifieur* suivant les spécificités de la tâche de reconnaissance étudiée. Le plus souvent, l'espace image d'une telle représentation est réduit à $\{\pm 1\}$, conformément au chapitre 1.

La notion de temps apparaît à deux échelles. D'abord au niveau des descripteurs : il existe

un perpétuel compromis entre la discrimination et l'invariance des caractéristiques d'un signal (section 2.3). En effet, si l'on prend en compte l'entière temporalité des caractéristiques, alors celles-ci sont tellement discriminantes qu'elles conduisent à assigner chaque signal à une classe singleton, ne permettant pas de généraliser à des signaux inédits. Au contraire, l'absence totale de temporalité des descripteurs ne permet généralement pas de distinguer les classes (du fait de la non-stationnarité des signaux). Entre ces deux extrémités, la juste prise en compte des aspects temporels des caractéristiques conduit à une règle de décision efficace. Au même titre que beaucoup de spécificités de l'apprentissage automatique, ce juste milieu dépend de la quantité et de la nature de l'information à disposition (*i.e.* de l'ensemble d'apprentissage).

La deuxième échelle à laquelle apparaît le temps est au niveau de la distinction entre *séquence* et *signal*. Lorsqu'un événement est *court* et *structuré* (*i.e.* l'ordre temporel qui y réside possède une importance dans la distinction des classes), il est possible d'en extraire directement une représentation TC (auquel cas *événement* et *signal* sont des termes équivalents). Celle-ci est alors caractéristique de l'événement dont elle est extraite et l'étiquette prédite pour la représentation TC peut être directement assignée à l'événement (première approche de l'illustration 2.1 page ci-contre). Au contraire, lorsqu'un événement est *long* ou *non-structuré*, il est d'usage d'en extraire des sous-parties (des signaux), donnant naissance à autant de représentations TC (deuxième et troisième approches de l'illustration 2.1 page suivante). Dans cette situation, la contribution de chaque représentation TC importe à la décision finale. L'ambivalence d'une telle chaîne de traitement réside dans la façon de combiner ces contributions : à l'étape de la représentation TC ou à celle de la représentation discriminante. Cette intégration fait l'objet d'une troisième représentation, permutable avec la représentation discriminante, qui répond à l'éventuel désordre structurel macroscopique de l'événement.

Nous revenons à présent sur les trois modèles issus des spécificités de la reconnaissance de signaux esquissées précédemment, en nous appuyant sur la figure 2.1 page ci-contre. À l'origine, se présente une longue série temporelle composée d'événements à reconnaître. Ces événements sont extraits, en ce qui nous concerne, par l'expertise d'un agent, bien qu'il existe tout un pan de la recherche uniquement consacré à la segmentation automatique de séries temporelles. De cette extraction naît trois situations, associées chacune à un modèle :

- ◇ ***direct*** : l'événement (court) est caractérisé par un unique signal. Ce dernier est successivement transformé par deux représentations (TC et discriminante) afin d'aboutir à une décision automatisée ;
- ◇ ***séquentiel*** : l'événement (long) est scindé en plusieurs signaux qui le caractérisent. Chacun de ces signaux est représenté dans le plan TC puis l'ensemble est synthétisé par un représentant virtuel (représentation 1). Ce mode de représentation est communément implémenté sous la forme d'une moyenne respectant la nature géométrique du sous-espace engendré par les signaux (par exemple une variété riemannienne) ou d'un histogramme. Enfin vient la représentation discriminante, appliquée au représentant de l'événement (représentation 2) ;
- ◇ ***démocratique*** : dans cette configuration, ressemblante à la précédente, l'opération de synthèse (représentation 2) intervient après la représentation discriminante 1. Cette représentation est souvent définie par un vote majoritaire.

Dans la suite de ce chapitre, nous donnons d'abord un rapide aperçu des caractéristiques discriminantes utilisées en reconnaissance de signaux, et en particulier des enregistrements audio (section 2.2). Nous abordons ensuite différentes manières de déambuler sur le fil du compromis entre discrimination et invariance (pour des signaux courts structurés) grâce aux fonctions d'agrégation (section 2.3). Suite à cela, nous nous focalisons sur les transformations TC (permettant d'extraire des caractéristiques) et traitons des techniques d'apprentissage numérique permettant d'apprendre de telles représentations dans les approches di-

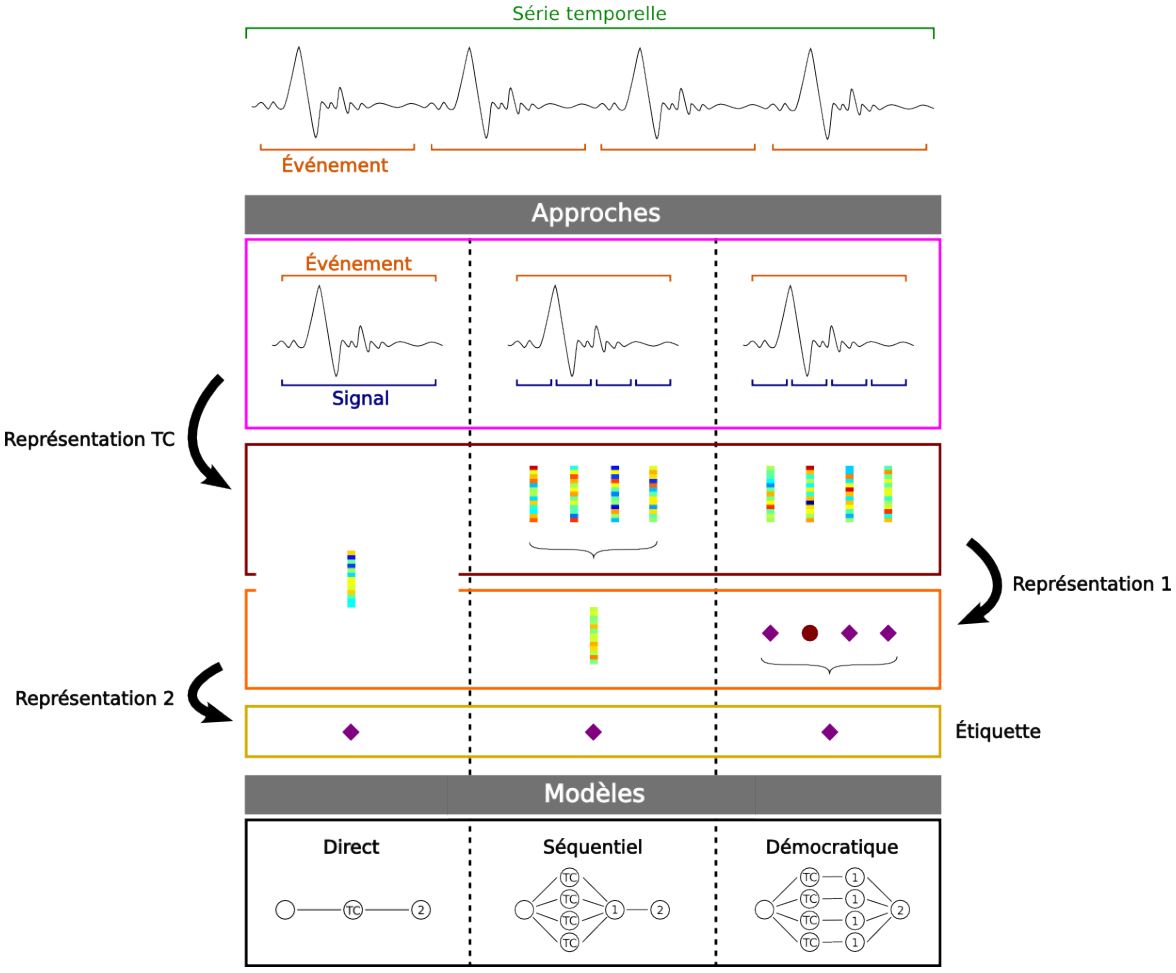


FIGURE 2.1 – Chaînes de traitement pour la reconnaissance d'un signal.

recte et démocratique (section 2.4). Enfin, forts de ces connaissances préalables, nous abordons la question encore nouvelle et passionnante de la reconnaissance précoce qui prend généralement place dans un schéma séquentiel (section 2.5).

2.2 DESCRIPTEURS

Dans une chaîne de traitement usuelle, nous avons identifié une première étape, nommée *représentation TC*. Elle a pour fonction d'extraire des descripteurs discriminants regroupés sous forme de vecteurs caractéristiques. Très souvent et de façon justifiée par l'analyse de signaux transitoires, ces descripteurs possèdent une composante temporelle (d'où la qualification *temps-caractéristique*), *i.e.* ce sont des fonctions du temps. Lorsque le temps est pris en compte dans le calcul, on parlera de descripteurs non-stationnaires (*e.g.* une décomposition en ondelettes). En revanche, lorsqu'aucune notion temporelle n'intervient, on parlera de descripteurs stationnaires (*e.g.* l'énergie totale d'un signal). Comme leur nom l'indique, ces derniers supposent que les caractéristiques du signal analysé sont constantes à l'échelle de l'étude. La notion de descripteurs *temps-caractéristique* tient toujours si l'on considère la caractéristique constante en fonction du temps. On peut remarquer une dernière catégorie, les descripteurs globaux [Mitrović *et coll.*, 2010], qui prennent en compte la dépendance temporelle du signal mais sont tout de même constants (*e.g.* l'attaque, qui correspond au temps nécessaire au signal pour atteindre le premier maximum significatif).

Pour construire cette représentation TC, Mitrović *et coll.* distinguent trois grandes familles d'applications [1] [Mitrović *et coll.*, 2010] : les ***transformées***, qui font le lien entre deux domaines de nature et d'interprétation bien différentes (*e.g.* une transformée de Fourier fait le pont entre les domaines temporel et fréquentiel) ; les ***filtres***, qui agissent à l'intérieur d'un domaine donné pour modifier la représentation d'un signal (*e.g.* normalisation, fenêtrage) ; enfin les ***fonctions d'agrégation***, qui visent à réduire la résolution temporelle pour concentrer l'information (*e.g.* moyenne, médiane, histogramme). Ce dernier type d'opérations fera l'objet de la section 2.3. Nous nous intéressons donc à présent principalement aux représentations TC avant composition avec une fonction d'agrégation.

Il existe, dans la littérature, plusieurs taxonomies des descripteurs adaptés aux signaux. Loin de vouloir en créer une nouvelle ici, nous nous contentons d'esquisser et d'adapter celles récemment communiquées dans [Mitrović *et coll.*, 2010, Chachada et Kuo, 2013], à partir des domaines entre lesquels nous font naviguer les transformées. Indifféremment du domaine d'application (traitement de la musique, de la parole, de sons environnementaux, *etc.*), cette esquisse de taxonomie décrit les descripteurs *stationnaires*, **non-stationnaires** et globaux en les distinguant dans le texte. On trouve ainsi :

- ◇ ***le domaine temporel*** : c'est généralement le domaine d'acquisition des signaux. Ceux-ci sont représentés par l'évolution d'une amplitude en fonction du temps. Les descripteurs que l'on y trouve sont, par exemple, le *taux de passage par zéro*, l'attaque, les **dictionnaire d'atomes** (section 2.4.6) et les statistiques dérivées des paramètres des atomes (par exemples les moyennes et variances de l'échelle et de la fréquence d'atomes de Gabor).
- ◇ ***le domaine fréquentiel*** : il met en relief la distribution spectrale du signal et permet d'analyser ses harmoniques. On y accède généralement grâce à un Banc de Filtres (BdF). Ce domaine comprend des descripteurs fréquentiels physiques : les *coefficients d'auto-régression linéaire*, la *transformation de Fourier* et de **Fourier à court-terme**, le *flux spectral*, les **représentations bilinéaires** (section 2.4.1), les **décompositions en ondelettes** (section 2.4.4), en **paquets ondelettes** et en **diffusion d'ondelettes** (section 2.4.5). On y trouve aussi des descripteurs fréquentiels perceptuels, comme l'*acuité* (qui indique si un son est dominé par les hautes ou basses fréquences), l'*étalement spectral* (qui quantifie la proximité d'un son à une sinusoïde pure et de manière équivalente la dissimilarité par rapport à un bruit blanc), la *hauteur* (qui décrit la fréquence fondamentale réelle ou perçue) et l'*harmonie* (caractérisant la présence ou non de multiples de la fréquence fondamentale) ;
- ◇ ***le domaine de corrélation*** : il représente les relations temporelles entre signaux. En particulier, l'auto-corrélation révèle les périodicités au sein d'un signal ;
- ◇ ***le domaine cepstral*** : on y parvient généralement en appliquant une transformée en cosinus discrète (*Discrete Cosine Transform*, DCT) au logarithme de l'amplitude spectrale. La dernière transformation a pour but de décorréler les coefficients logarithmiques. C'est une façon d'approcher la forme (l'enveloppe) du spectre (et de capturer ainsi l'information de timbre du signal). Des descripteurs associés sont les *coefficients cepstraux mel-fréquences (*Mel-Frequency Cepstral Coefficients, *MFCC)* et leurs première et seconde différences finies Δ**MFCC** et Δ^2**MFCC** ainsi que les *coefficients d'auto-régression linéaire cepstraux* (les principales différences avec les MFCC étant des transformations psycho-acoustiques intermédiaires et une auto-régression appliquée en lieu et place de l'ultime étape de décorrélation par DCT) ;
- ◇ ***le domaine de modulation de fréquence*** : il reflète les modulations temporelles présentes dans le signal, en particulier les structures rythmiques du signal. Deux descripteurs caractéristiques de ce domaine sont l'**énergie à 4 Hz**, qui permet de distinguer

1. Dans [Mitrović *et coll.*, 2010], cette distinction est faite pour l'analyse de signaux audio mais s'étend naturellement au traitement du signal en général.

la parole d'autres types de signaux, et la *métrique de pulse*, qui indique de manière quantitative si un signal est rythmé ou non ;

- ◇ ***le domaine de phase*** : il représente les dynamiques non-linéaires présentes dans le signal et caractérise par exemple les phonèmes ;
- ◇ ***le domaine propre*** : il est généré par des vecteurs propres ou singuliers, par exemple dans le cas d'une Analyse en Composantes Principales (ACP) ou d'une décomposition en valeurs singulières.

Remarque 5.
Certains descripteurs stationnaires sont souvent extraits d'un instantané (*i.e.* une sous-partie de 20 à 400 millisecondes) en supposant ses caractéristiques invariantes à cette échelle. En répétant l'opération d'extraction de tels descripteurs sur une succession d'instantanés, on obtient une représentation TC (*i.e.* possédant une composante temporelle). L'exemple le plus probant est la transformée de Fourier à court-terme, issue de la transformée de Fourier usuelle.

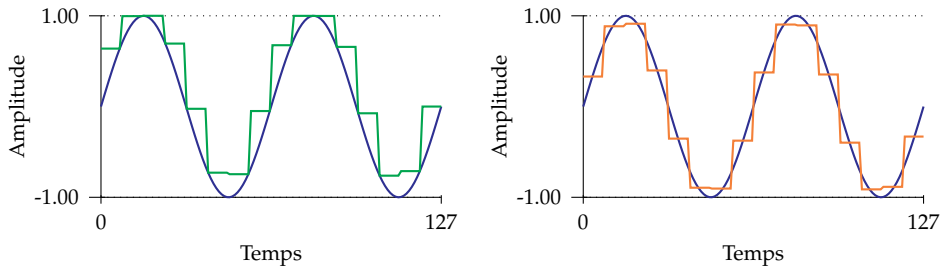
Remarque 6.
Le spectrogramme se présentant comme un outil permettant de visualiser et d'analyser la distribution TF d'un signal, il est aussi envisageable d'extraire des descripteurs *visuels* (issus du traitement d'images), comme la texture, dans le but de reconnaître des signaux [Chachada et Kuo, 2013].

Étant donné leur succès pour le traitement des enregistrements vocaux et en l'absence de descripteurs spécifiques, les MFCC sont rapidement devenus une référence dans la reconnaissance audio en général, et en particulier de scènes environnementales. En outre, il est courant d'adjoindre aux MFCC d'autres descripteurs afin d'améliorer la reconnaissance de signaux (par exemple ceux renseignés dans la normalisation MPEG-7 ou des descripteurs non-stationnaires).

2.3 AGRÉGATION

L'analyse TF est apparue suite à l'incapacité de la transformée de Fourier à transcrire l'évolution temporelle des caractéristiques fréquentielles d'un signal. Une telle technique d'analyse est particulièrement justifiée pour l'étude des signaux réels, qui sont souvent non-stationnaires (*i.e.* leurs caractéristiques fréquentielles changent avec le temps). Toutefois, il existe une contrepartie à la prise en compte du temps pour la classification de signaux : celui-ci est potentiellement source de variabilité intra-classe. Par exemple, ce phénomène est clairement observé si les signaux d'apprentissage ne sont pas correctement alignés. Il est donc nécessaire de mettre au point des caractéristiques fondées sur des comportements TF locaux. Ceci passe généralement par l'utilisation d'opérateurs non-linéaires invariants à certaines déformations. Une fonction d'agrégation ρ est un tel opérateur, qui réalise un compromis entre la puissance de discrimination et l'invariance des descripteurs. Cette notion de compromis a été étudiée théoriquement [Shi et Manduchi, 2004] et apprise par une technique standard de noyau multiple [Varma et Ray, 2007].

L'idée d'agrégation des caractéristiques trouve son origine dans les travaux d'Hubel et Wiesel sur le cortex visuel [Hubel et Wiesel, 1962]. Elle est utilisée dans les modèles d'inspiration biologique tels que les réseaux neuronaux [Fukushima, 1980], et mise en pratique par une moyenne [LeCun *et coll.*, 1989] ou un maximum local [Ranzato *et coll.*, 2008]. Une étude théorique a montré l'intérêt d'une agrégation par maximum plutôt que par moyenne locale [Boureau *et coll.*, 2010b]. Dans le cadre d'autres applications, on trouve aussi des statistiques telles que la variance ou la matrice de covariance [Lee et Pottier, 2009], les pseudo-normes ℓ_p



(a) Agrégation par le maximum. (b) Agrégation par la moyenne.

FIGURE 2.2 – Exemple de fonctions d'agrégation avec une fenêtre $w = 8$.

($p \in \mathbb{R}^*$) et les histogrammes [Rakotomamonjy et Gasso, 2014]. En ce qui concerne la reconnaissance de signaux, ces fonctions d'agrégation sont appliquées soit aux représentations TF, soit aux paramètres génératifs dans le cas d'une modélisation (par exemple à l'échelle d'un filtre de Gabor) [Mitrović *et coll.*, 2010].

NOM	DÉFINITION DE $\rho(\boldsymbol{x})$
norme ℓ_p	$\left(\sum_{k=1}^{m} \lvert x_k \rvert^p\right)^{\frac{1}{p}}$
norme ℓ_p locale	$\left(\left(\sum_{k=(l-1)w+1}^{lw} \lvert x_k \rvert^p\right)^{\frac{1}{p}}\right)_{l=1}^{\lfloor \frac{m}{w} \rfloor}$
Maximum	$\left(\max_{(l-1)w+1 \leq k \leq lw} x_k\right)_{l=1}^{\lfloor \frac{m}{w} \rfloor}$
Moyenne	$\left(\frac{1}{w} \sum_{(l-1)w+1 \leq k \leq lw} x_k\right)_{l=1}^{\lfloor \frac{m}{w} \rfloor}$
Diffusion	$\lvert \boldsymbol{x} \rvert \star \boldsymbol{g}$

TABLE 2.1 – Définitions de fonctions d'agrégation appliquées à un signal $\boldsymbol{x}$ de taille m. Paramètres : taille de fenêtre $w \in \mathbb{N}$, filtre passe-bas gaussien $\boldsymbol{g}$.

Dans notre cas, le principal écueil que nous cherchons à éviter quant aux fonctions d'agrégation que nous utilisons (voir tableau 2.1) est le faible décalage temporel aléatoire (défaut d'alignement) entre les signaux d'apprentissage, inhérent à l'acquisition de signaux réels. En un second temps, les fonctions d'agrégation ont aussi pour but de diminuer la dimension des caractéristiques utilisées par le classifieur (atténuation du fléau de la dimension).

La fonction d'agrégation la plus communément rencontrée est la norme ℓ_p (calculée sur tout le signal), particulièrement avec $p = 2$. Cette approche (extrême) est particulièrement efficace contre les décalages temporels aléatoires puisque l'information en sortie est indépendante du temps (ce qui explique son utilisation fréquente en classification de signaux). Cependant l'absence d'information temporelle est aussi le principal inconvénient de cette approche. Une façon de dépasser cela est de calculer des caractéristiques invariantes localement, par exemple en partitionnant le signal d'entrée et en calculant la norme, le maximum ou la moyenne sur chaque sous-signal ainsi créé (figure 2.2). Ces méthodes ont été introduites à l'origine pour les réseau de neurones convolutifs (*Convolutional Neural Network*, CNN) [LeCun *et coll.*, 1998] afin d'absorber les décalages spatiaux des objets au sein des images tout en réalisant un compromis avec la résolution spatiale. Enfin, l'agrégation par diffusion a été récemment introduite dans [Mallat, 2012], particulièrement pour les signaux à valeurs complexes. L'invariance au décalage temporel provient de l'utilisation simultanée de l'opérateur module et du moyennage gaussien. Il est possible de montrer que ce type d'agrégation est lié au regroupement logarithmique des fréquences appliqué dans le calcul des MFCC [Andén et Mallat, 2014].

Dans le tableau 2.1, les définitions sont données pour un unique signal. Toutefois, ces définitions peuvent naturellement être étendues à un ensemble de signaux (le modèle utilisé ici pour une représentation TF). Étant donné $\mathcal{O} = \mathbb{K}^{m_1} \times \cdots \times \mathbb{K}^{m_d}$ l'ensemble des représentations TF à d canaux (chaque canal porte un signal temporel, de longueur spécifique), on étend les définitions précédentes des fonctions d'agrégation par :

$$\rho \colon (\boldsymbol{x_l})_{l=1}^{d} \in \mathcal{O} \mapsto \mathrm{Vec}\left((\rho(\boldsymbol{x_l}))_{l=1}^{d}\right) \in \mathcal{X},$$

où $\mathrm{Vec}(\cdot)$ est l'opérateur de vectorisation retournant un vecteur unidimensionnel contenant toute l'information présente à l'entrée. Par exemple, l'agrégation par norme ℓ_2, appliquée à une représentation TF, fournit la distribution marginale en énergie pour différentes bandes fréquentielles.

Dans la suite, nous revenons à une phase antérieure de l'extraction de caractéristiques, qui est celle de la représentation TC (ou TF). Nous détaillons les moyens d'obtenir certaines d'entre elles et de les apprendre avec une visée de reconnaissance automatique.

2.4 TRANSFORMÉES TEMPS-CARACTÉRISTIQUE

Au fil des années, différentes représentations TC ont été utilisées pour la reconnaissance de signaux : les transformées énergétiques bilinéaires (section 2.4.1), les décompositions atomiques (en particulier les BdF, présentés dans les sections 2.4.2 et 2.4.3), les décompositions en ondelettes et en diffusion d'ondelettes (sections 2.4.4 et 2.4.5) et par dictionnaire (section 2.4.6). L'un des points cruciaux de cette section, au delà de la définition de telles représentations, est d'introduire les techniques majeures qui permettent de les décliner de manière dirigée par les données pour une tâche de reconnaissance (l'apprentissage discriminant). Les travaux représentatifs de l'apprentissage discriminant de représentations TC sont synthétisés dans le tableau C.1.

2.4.1 Distribution bilinéaire

Il est possible de définir trois grandes familles de représentations TF [Flandrin, 1998] :

- ◇ ***décomposition atomique (ou linéaire)*** : le signal temporel est représenté dans le domaine TF. Deux exemples directs sont la transformée de Gabor (dont la grille uniforme dans le domaine de projection prive la transformée d'orthogonalité) et la transformée en ondelettes (dont l'orthogonalité provient du pavage dyadique du plan) ;
- ◇ ***distribution d'énergie (ou bilinéaire)*** : l'énergie du signal est représentée en temps et en fréquence.
- ◇ ***distribution de puissance*** : la puissance du signal est représentée dans le plan TF.

Cette section s'intéresse au deuxième type de décompositions. Il existe une variété de transformées TF bilinéaires, parmi lesquelles celles de Rihaczek, Choï-Williams et Page-Levin [Flandrin, 1998], mais dont la plus connue reste certainement la première introduite : celle de Wigner-Ville. Cette représentation TF de l'énergie d'un signal est simple à mettre en place mais possède le défaut majeur d'introduire des termes de corrélation croisée lorsque le signal est la somme de différentes contributions, pénalisant ainsi l'interprétation de la représentation du signal dans le plan TF. Face à cela, la classe de Cohen est une extension de la transformée de Wigner-Ville, munie d'un noyau visant à atténuer les termes de corrélation croisée et possédant certaines propriétés : la classe de Cohen se veut l'ensemble des distributions d'énergie bilinéaires *covariantes par translations temporelle et fréquentielle* (voir l'annexe D pour plus de détails).

Étant donné un signal s de $L^2(\mathbb{R})$, la distribution de Cohen est habituellement définie dans l'espace Doppler-retard (ξ, τ) :

$$C_s(t,\omega) = \iint_{-\infty}^{+\infty} A_s(\xi,\tau)\Phi(\xi,\tau)e^{-\imath(\xi t+\tau\omega)}\, d\xi\, d\tau,$$

où A_s est la fonction d'ambiguïté à bande étroite du signal s et Φ est appelé *noyau de Cohen*. La fonction d'ambiguïté est liée à la décomposition de Wigner-Ville par une transformée de Fourier inverse en fréquence, et s'exprime par :

$$A_s\colon (\xi,\tau) \in \mathbb{R}^2 \mapsto \frac{1}{2\pi}\int_{-\infty}^{+\infty} s(\nu+\frac{\tau}{2})\bar{s}(\nu-\frac{\tau}{2})e^{i\xi\nu}d\nu.$$

Le plan Doppler-retard facilite le lissage des interférences puisque les termes quadratiques purs issus des composantes du signal se trouvent à l'origine, tandis que les termes de corrélation croisée sont relégués en périphérie. Le noyau de Cohen Φ a ainsi pour but de capturer uniquement les composantes utiles de la représentation énergétique, en atténuant les interférences desservant l'interprétation.

De manière parallèle, on peut créer une classe de représentations temps-échelle bilinéaires (généralisation des ondelettes) en remplaçant la covariance par translation par celle portant sur le groupe affine [Flandrin, 1998]. On crée ainsi la classe affine. Dans les deux cas (classe de Cohen et classe affine), les formulations unifiées présentent deux intérêts principaux :

- ◇ la plupart des distributions énergétiques connues se retrouvent en spécifiant le noyau Φ ;
- ◇ une contrainte sur une représentation est généralement facilement traduisible en propriété d'admissibilité pour le noyau Φ, autorisant ainsi à construire une classe de solutions en fonction d'un cahier des charges donné.

Apprentissage discriminant

Motivé par la recherche d'outils performants pour les tests de qualité en production industrielle, Heitz ouvre la voie de l'automatisation du choix du noyau de Cohen en se proposant de déterminer de manière dirigée par les données, les rayons d'un noyau gaussien :

$$\Phi_{\boldsymbol{\theta}}\colon (\xi,\tau) \in \mathbb{R}^2 \mapsto \mathrm{e}^{-\left(\frac{\xi^2}{\theta_1^2}+\frac{\tau^2}{\theta_2^2}\right)},$$

où $\boldsymbol{\theta}$ est un vecteur paramètre de $\mathbb{R}^2$ [Heitz, 1995, Heitz, 1996]. Sur la base d'une technique d'optimisation non-déterminée, Heitz obtient un paramètre sous-optimal $\boldsymbol{\theta}$ maximisant un critère de Fisher (voir annexe A) calculé dans le plan TF.

En normalisant en valeur absolue la représentation TF bilinéaire d'un signal s :

$$\overline{C}_s\colon (t,\omega) \in \mathbb{R}^2 \mapsto \frac{|C_s(t,\omega)|}{\iint_{-\infty}^{\infty}|C_s(t',\omega')|\; dt'\, d\omega'},$$

il est possible de l'interpréter comme une densité de probabilité et ainsi d'utiliser des mesures de dissimilarité telles que les divergences de Kolmogorov [Davy et Doncarli, 1998] et de Bhattacharyya [Davy *et coll.*, 2001], aussi bien que l'opposé de la corrélation et qu'une pseudo-norme ℓ_p ($p \in \mathbb{R}^*$) quelconque [Davy *et coll.*, 2001]. On peut alors, à l'instar de travaux précédents [Atlas *et coll.*, 1997], considérer un classifieur par distance minimum par rapport à des prototypes construits comme des moyennes de représentations TF [Davy et Doncarli, 1998, Davy *et coll.*, 2001], ou une machine à vecteurs supports (*Support Vector Machine*, SVM) [Davy *et coll.*, 2002]. Dans toutes les études précédemment citées, le principal

noyau de Cohen considéré est une fonction gaussienne exprimée en coordonnées polaires [Baraniuk et Jones, 1993]. Soient p un entier impair et $\boldsymbol{\theta}$ un vecteur de paramètres de $\mathbb{R}^p$; alors le noyau de Cohen s'écrit :

$$\Phi_{\boldsymbol{\theta}} : (\xi, \tau) \in \mathbb{R} \times \mathbb{R}^* \mapsto \mathrm{e}^{-\frac{\rho(\xi,\tau)}{2(\sigma^2_{\boldsymbol{\theta}} \circ \phi)(\xi,\tau)}},$$

où l'évaluation de la fonction rayon vaut : $\rho(\xi, \tau) = \xi^2 + \tau^2$; et celle de la fonction angle : $\phi(\xi, \tau) = \arctan\left(\frac{\xi}{\tau}\right)$. La fonction d'étalement est définie par [Davy *et coll.*, 2002] :

$$\sigma_{\boldsymbol{\theta}} : \omega \in \left]-\frac{\pi}{2}, \frac{\pi}{2}\right[\mapsto \theta_1 + \sum_{k=1}^{\left\lfloor \frac{p}{2} \right\rfloor} \Big(\theta_{2k} \cos(2k\omega) + \theta_{2k+1} \sin(2k\omega)\Big).$$

Cette fonction représente une approximation polynomiale (au sens des séries de Fourier) d'une fonction d'étalement hypothétique π-périodique. En pratique, l'approximation est réalisée à l'ordre 1 ou 2, *i.e.* le nombre p de paramètres du noyau de Cohen vaut 3 ou 5 [Davy et Doncarli, 1998]. De plus, l'apprentissage de ces paramètres est réalisé par la maximisation du critère de Fisher grâce à la méthode Broyden-Fletcher-Goldfarb-Shanno (BFGS) [Davy et Doncarli, 1998] ou par minimisation d'une estimation de la probabilité d'erreur grâce à la technique de Nelder-Mead (qui ne nécessite pas de calcul de gradient) [Davy *et coll.*, 2001, Davy *et coll.*, 2002].

Une élégante théorie de noyaux TF reproduisant a été développée pour les machines à noyau telles que l'ACP, l'analyse discriminante de Fisher et les classifieurs linéaires (*e.g.* SVM) [Honeine *et coll.*, 2007]. Cette dernière met en place une nouvelle astuce du noyau dans laquelle tout produit scalaire dans le plan TF est substitué par un produit scalaire généralement de moindre complexité. Si l'on considère, par exemple [2], l'espace engendré par la représentation de Wigner-Ville (notée W_s pour un signal s), l'astuce du noyau TF réside dans l'égalité suivante :

$$\forall (s_1, s_2) \in L^2(\mathbb{R}) \times L^2(\mathbb{R}) : \iint_{-\infty}^{\infty} W_{s_1}(t, f) W_{s_2}(t, f)\, dt\, df = \left| \int_{-\infty}^{\infty} s_1(t) \overline{s_2}(t)\, dt \right|^2 .$$

Il est ainsi inutile de calculer explicitement les représentations bilinéaires des signaux pour les traiter, résultant en un gain de temps calculatoire. Forts de cette théorie, Honeine *et coll.* proposent d'adapter les techniques usuelles d'apprentissage de noyau SVM (en particulier l'alignement de noyaux [Cristianini *et coll.*, 2002]) à la création automatique de représentations TF pour la classification. Une première étude a montré l'intérêt de combiner différentes transformées de la classe de Cohen pour la reconnaissance de signaux [Honeine *et coll.*, 2006]. La pondération des noyaux SVM (et *a fortiori* de Cohen) est obtenue par un algorithme glouton qui ajoute à chaque itération un noyau SVM maximisant l'alignement [Honeine *et coll.*, 2006]. Cet algorithme est astucieusement fondé sur une solution analytique du problème d'alignement d'une combinaison de deux noyaux générateurs [Pothin et Richard, 2005]. Cette étude a été étendue par la suite à l'apprentissage d'un noyau de Cohen gaussien [Honeine et Richard, 2007] exprimé de manière identique à celle de travaux précédents concernant l'apprentissage de représentations TF favorisant l'interprétation [Baraniuk et Jones, 1993] et la discrimination [Davy et Doncarli, 1998]. Dans l'approche proposée, toujours fondée sur l'alignement de noyaux SVM, la fonction d'étalement $\sigma_{\boldsymbol{\theta}}$ est discrétisée et déterminée par une montée de gradient projeté sur une contrainte en norme ℓ_2.

Au delà des noyaux de Cohen paramétrés, certaines études se sont portées sur l'apprentissage discriminant de noyaux libres, paramétrant une représentation bilinéaire construite

2. D'autres mises en pratique, comprenant des transformées TF linéaires (Fourier, ondelettes) et quadratiques (spectrogamme, Cohen), sont explicitées dans [Honeine, 2007, p. 42].

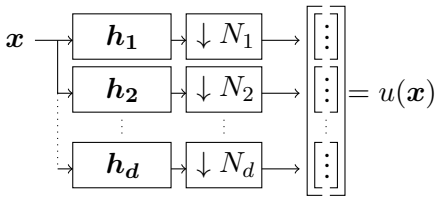


FIGURE 2.3 – Diagramme d'un BdF u à d canaux.

sur la distribution de Rihacsek (plutôt que sur celle de Wigner-Ville) [Atlas *et coll.*, 1997, Droppo et Atlas, 1998] [3]. En s'inspirant de la théorie des opérateurs introduite dans [Narayanan *et coll.*, 1996], Atlas *et coll.* formalisent le problème d'apprentissage à travers la maximisation de la distance entre des représentants de chaque classe (définis par la moyenne des représentations TF de chaque individu) [Atlas *et coll.*, 1997]. En utilisant la version discrète de la transformée de Rihacsek, ce problème se réduit à déterminer une matrice noyau répondant à un problème aux valeurs propres. En conséquence, cette matrice de Cohen optimale ne possède donc qu'une seule valeur non-nulle, correspondant au point Doppler-retard le plus discriminant. Par la suite, un critère de Fisher est utilisé pour ordonner les coordonnées du plan Doppler-retard par puissance discriminante [Droppo et Atlas, 1998]. Il est alors possible de construire une matrice noyau comme un masque binaire contenant autant de coordonnées discriminantes que souhaité. Cette approche est plus robuste que celle initialement proposée [Atlas *et coll.*, 1997] puisqu'elle minimise la variabilité intra-classe des représentations TF.

2.4.2 Banc de filtres

Les BdF sont des modèles linéaires qui incarnent une vaste classe de transformées utilisées en traitement du signal, parmi lesquelles les transformées discrètes de Fourier, en cosinus et en ondelettes. D'un point de vue conceptuel, un BdF est constitué d'un ensemble de filtres en parallèles, suivis par un opérateur de sous-échantillonnage, appelé opérateur de décimation (figure 2.3) [Strang et Nguyen, 1996].

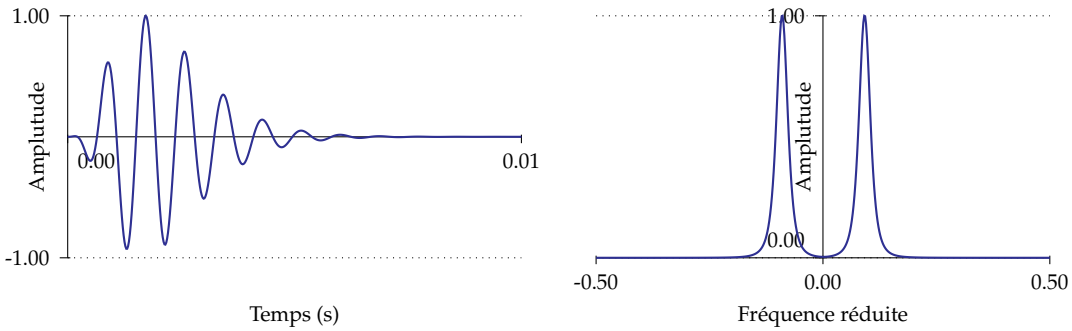
Définition 2.4.1 (Opérateur de décimation)**.**
On appelle opérateur de décimation de facteur N et on note $\boldsymbol{x} \in \mathbb{K}^m \mapsto \downarrow N[\boldsymbol{x}] \in \mathbb{K}^{\lfloor \frac{m}{N} \rfloor}$ l'application linéaire vérifiant :

$$\forall \boldsymbol{x} \in \mathbb{K}^m : \downarrow N[\boldsymbol{x}] = \left(x_{1+N(k-1)}\right)_{1 \leq k \leq \lfloor \frac{m}{N} \rfloor}.$$

Définition 2.4.2 (Banc de filtres)**.**
Soient m un entier, $\mathcal{U}$ un sous-ensemble de signaux discrets de $\mathbb{K}^m$, d un entier et $(N_l)_{1 \leq l \leq d}$ un d-uplet de facteurs de décimations. On appelle $\mathcal{O} = \mathbb{K}^{\lfloor \frac{m}{N_1} \rfloor} \times \cdots \times \mathbb{K}^{\lfloor \frac{m}{N_d} \rfloor}$ l'ensemble des représentations TF. Un banc de d filtres est une application linéaire $u : \mathcal{U} \to \mathcal{O}$, définie par d filtres linéaires à Réponses Impulsionnelles (RI) finies $\{\boldsymbol{h_l}\}_{1 \leq l \leq d}$ (de tailles respectives $\{q_l\}_{1 \leq l \leq d}$) et d facteurs de décimation $\{N_l\}_{1 \leq l \leq d}$. Formellement, un BdF u, aussi noté $(\boldsymbol{h_l}, N_l)_{1 \leq l \leq d}$, vérifie

$$\begin{aligned} &\forall \boldsymbol{x} \in \mathcal{U} : \\ &u(\boldsymbol{x}) = \left(\downarrow N_l \left[\boldsymbol{h_l} \star \boldsymbol{x}\right]\right)_{1 \leq l \leq d} \\ &\qquad = \left(\downarrow N_l \left[\left(\sum_{j=1}^{q_l} h_{lj} \tilde{x}_{k-j+1}\right)_{1 \leq k \leq m}\right]\right)_{1 \leq l \leq d}, \end{aligned}$$

3. Ces deux travaux ont été largement développés respectivement dans [Atlas *et coll.*, 1997, McLaughlin *et coll.*, 1997, McLaughlin, 1997] et [Droppo et Atlas, 1998, Gillespie et Atlas, 1998, Gillespie et Atlas, 1999, Gillespie et Atlas, 2001].



(a) Représentation temporelle.

(b) Représentation spectrale.

FIGURE 2.4 – Filtre gammatone continu et représentation spectrale de la RI de taille 128 à 22050 Hz correspondante.

où $\tilde{\boldsymbol{x}}$ est le signal $\boldsymbol{x}$ étendu vers le passé (par périodisation, symétrisation ou complétion par zéros).

Cette définition (classique), que nous donnons d'un BdF, est celle d'une application linéaire fondée sur un ensemble de filtres à RI finies. Il serait possible de construire un BdF non-linéaire ou à partir de filtres à RI infinies mais nous nous restreignons à cette définition pour notre étude.

Les BdF ont été intensément étudiés en traitement du signal, notamment pour la compression et le débruitage [Vaidyanathan, 1993]. Plus précisément, étant donné un BdF inversible similaire à celui représenté dans la figure 2.3 page précédente (appelé BdF d'analyse), il existe plusieurs façons de construire un BdF inverse (BdF de synthèse) qui reconstruit le signal après traitement [Gauthier *et coll.*, 2009]. En conséquence, de nombreuses autres définitions et propriétés seraient nécessaires pour décrire les travaux effectués dans ces champs applicatifs, en particulier la représentation polyphase d'un BdF [Strang et Nguyen, 1996]. Néanmoins, la partie de notre travail qui traite des BdF (chapitre 3) se concentre sur la reconnaissance de signaux dans le domaine d'analyse, sans considérer l'inversion ni même l'inversibilité du BdF utilisé. Ainsi, nous nous contenterons de ces définitions partielles, qui sont largement suffisantes dans le cadre de nos travaux.

Un BdF peut bénéficier de tout type de filtres. Par exemple, le filtre *gammatone* fut introduit sur une modélisation du système auditif. Son expression temporelle est [Qi *et coll.*, 2013] :

$$\forall t \in \mathbb{R}_+ : g(t) = at^{r-1}\mathrm{e}^{-2\pi bt}\cos(2\pi f_c t + \Phi),$$

où t est en secondes, f_c est la fréquence centrale du filtre (en Hz), Φ est la phase (en radian), généralement fixée à 0, a contrôle le gain du filtre, r est l'ordre de celui-ci (souvent au plus 4) et b est la bande-passante du filtre, souvent définie de manière linéaire par rapport à la fréquence centrale [Slaney, 1993]. En faisant varier la fréquence centrale f_c du filtre ainsi que sa bande passante b, il est possible de construire un BdF à l'image du système auditif. La figure 2.4 fait apparaître la représentation temporelle d'un filtre gammatone continu ainsi que la représentation spectrale de la RI de taille 128 à 22050 Hz correspondante.

Parmi les diverses applications des BdF, l'une possède une certaine notoriété dans le domaine de la reconnaissance de signaux audio. Il s'agit des MFCC calculés sur une fenêtre glissante [Stevens *et coll.*, 1937], qui sont apparus dans le cadre de la reconnaissance automatique de la parole et qui ont évolué par la suite vers l'un des ensembles de descripteurs standards en reconnaissance audio. L'annexe E expose en détails le calcul des MFCC, de sorte que nous en donnons uniquement un rapide tour d'horizon ici. L'analyse cepstrale a été mise en place de manière à imiter le fonctionnement de la cochlée. Elle possède donc

deux spécificités. La première est de calculer l'énergie d'un signal dans le plan TF grâce à un banc de filtres répartis sur une échelle logarithmique. Cette échelle MÉLodie (MEL) modélise notre perception psycho-acoustique du signal. La seconde est une nouvelle transformation de la représentation TF par une valeur absolue, un logarithme et une DCT. Ces opérations combinées ont pour but d'extraire l'enveloppe spectrale du signal, qui caractérise le timbre de celui-ci. En particulier, le premier coefficient MFCC encode l'énergie du signal. Puisque celui-ci est généralement grand par rapport aux autres, il est habituel, pour des tâches de reconnaissance, de normaliser les coefficients cepstraux MEL indépendamment les uns des autres ou d'évincer le premier coefficient MFCC.

Apprentissage discriminant

L'apprentissage de BdF a été étudié sous plusieurs formes par Biem *et coll.* pour des applications en reconnaissance vocale. Dans ces études, deux types de séparateurs sont utilisés : la technique de distance minimum par rapport à des prototypes [Biem *et coll.*, 1993, Biem et Katagiri, 1994, Biem *et coll.*, 1996, Biem et Katagiri, 1997, Biem *et coll.*, 2001] et un perceptron multi-couche (*Multi-Layer Perceptron*, MLP) [Biem et Katagiri, 1993, Biem *et coll.*, 1997]. Tous ces travaux sont fondés sur le principe nommé extraction de caractéristiques discriminantes (*Discriminative Feature Extraction*, DFE) et introduit dans [Biem et Katagiri, 1993, Biem *et coll.*, 1993]. DFE consiste à minimiser une erreur de classification par une descente de gradient conjointe sur les paramètres du classifieur et de la fonction d'extraction. Si l'on reprend [Biem *et coll.*, 2001], le classifieur est fondé sur la distance minimum. Pour un problème à K classes (indexées de 1 à K), la classe attribuée à un signal $\boldsymbol{x}$ est :

$$\underset{k\in\mathbb{N}_K}{\arg\min}\, d^{\mathcal{L}_k}\left((\rho\circ u)(\boldsymbol{x})\right),$$

où u est un BdF, ρ est un fonction d'agrégation énergétique dérivable, $\mathcal{L}_k$ est un ensemble de prototypes caractérisant la k$^{\text{e}}$ classe et $d^{\mathcal{L}_k}$ est une fonction représentative de la distance minimum de son argument à l'ensemble des prototypes $\mathcal{L}_k$ (approximation dérivable par une pseudo-norme ℓ_{-p} pour p suffisamment grand). En utilisant la fonction de perte logistique $L\colon a\in\mathbb{R}\mapsto\frac{1}{1+\mathrm{e}^{-\gamma a}}$ (où γ est une constante positive), le problème d'optimisation traité dans [Biem *et coll.*, 2001] est :

$$\underset{u,\,\mathcal{L}_1,\ldots,\mathcal{L}_K}{\text{minimiser}}\quad \frac{1}{n}\sum_{i=1}^{n} L\left(1-\frac{\left(\frac{1}{K-1}\sum_{\substack{1\le k\le K\\ k\neq y_j}}\left[d^{\mathcal{L}_k}\left((\rho\circ u)(\boldsymbol{x_i})\right)\right]^{-p}\right)^{-\frac{1}{p}}}{d^{\mathcal{L}_{y_i}}\left((\rho\circ u)(\boldsymbol{x_i})\right)}\right),$$

pour p suffisamment grand (c'est une nouvelle apparition de l'approximation dérivable de la fonction minimum mentionnée précédemment). Remarquons que, par croissance de la fonction de perte logistique L, pour approcher une solution du problème d'apprentissage, il est nécessaire de faire tendre le numérateur vers $+\infty$ et le dénominateur vers 0, ce qui est cohérent en terme de distance aux prototypes. Par ce biais, Biem *et coll.* apprennent donc simultanément un BdF ainsi que les prototypes du classifieur par distance minimum. Deux types de BdF sont étudiés : un banc à filtres gaussiens dans le domaine spectral (paramétrés par le centre, la largeur et le gain de la bande-passante) et un BdF à RI finies libres. Les expériences numériques montrent un léger sur-apprentissage de ce dernier type de BdF comparé au premier.

Les BdF ont été introduits à l'étude d'Électro-Encéphalogrammes (EEG) dans [Ang *et coll.*, 2008]. Les BdF sont alors couplés à la technique usuelle, nommée forme spatiale commune (*Common Spatial Pattern*, CSP), pour réaliser un filtrage spectral successivement à un filtrage spatial. Si les travaux exposés dans [Ang *et coll.*, 2008] (tout comme ceux plus récents de

[Thomas *et coll.*, 2009, Zhang *et coll.*, 2011]) considèrent un BdF prédéfini, l'apprentissage de celui-ci pour une application à la classification dans une Interface Cerveau-Machine (ICM) a été introduit par Suk *et coll.* [Suk et Lee, 2013]. Dans ce dernier article et à l'instar de [Ang *et coll.*, 2008], un BdF est utilisé puis des descripteurs sont extraits par CSP pour chaque bande fréquentielle. Le BdF est appris au sein d'un cadre bayésien dans lequel chaque bande fréquentielle est associée à une variable aléatoire. L'algorithme est fondé sur un filtre particulaire couplé à une information mutuelle pour quantifier le pouvoir de discrimination d'une bande de fréquences. La fonction de décision finale est obtenue par pondération des décisions pour chaque bande fréquentielle (issues de l'apprentissage d'une SVM).

D'autres travaux d'apprentissage de descripteurs ont été tournés vers l'ICM, tels que [Flamary *et coll.*, 2012]. Dans ce dernier ouvrage, les auteurs apprennent des filtres spectraux (un filtre par canal EEG) visant à améliorer la classification des signaux (sans utilisation de CSP). L'ensemble des filtres ne forment pas ici un BdF puisque chaque filtre est dédié à un canal. En simplifiant au cas mono-canal (celui que nous étudions) les travaux présentés dans [Flamary *et coll.*, 2012] se réduisent donc à apprendre un unique filtre spectral grâce au problème d'optimisation :

$$\underset{\boldsymbol{h}\in\mathbb{R}^q}{\text{minimiser}} \quad J_{\text{SR}}\left(k_{\boldsymbol{h}}\right) + \lambda \left\|\boldsymbol{h}\right\|_{\ell_2}^2 ,$$

où λ est un facteur de régularisation positif, $k_{\boldsymbol{h}}$ est un noyau comprenant l'opération de filtrage par le filtre $\boldsymbol{h}$ et défini par $k_{\boldsymbol{h}} \colon (\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{X}^2 \mapsto k(\boldsymbol{h} \star \boldsymbol{x}, \boldsymbol{h} \star \boldsymbol{z})$ et k est un noyau linéaire ou gaussien. Cette approche, bien que classique en apprentissage de noyau, est novatrice en traitement du signal. Elle comprend en outre une régularisation en norme ℓ_2 sur la RI $\boldsymbol{h}$ du filtre visant à limiter le sur-apprentissage qui apparaît lorsque l'énergie du filtre augmente (voir remarque 1.4.2 page 18). Une solution de ce problème est approchée par descente de gradient [Flamary *et coll.*, 2012].

Vignolo *et coll.* se sont intéressés à l'optimisation du BdF MEL utilisé dans le calcul des MFCC a des fins de reconnaissance automatique de la parole [Vignolo *et coll.*, 2011a, Vignolo *et coll.*, 2011b]. Les filtres sont toujours triangulaires (figure E.3 page 144) mais leurs positions, leurs largeurs de bande et leurs gains sont déterminés lors de l'apprentissage. Selon l'étude, les filtres sont soit paramétrés directement par les trois variables précédemment mentionnées [Vignolo *et coll.*, 2011a], soit les positions et les gains sont paramétrés par deux splines (les largeurs de bande sont calculées en fonction des positions) et les paramètres de ces splines sont optimisés lors de l'apprentissage [Vignolo *et coll.*, 2011b]. De manière classique pour cette application de reconnaissance de parole, les auteurs utilisent comme classifieur un modèle de Markov caché (*Hidden Markov Model*, HMM) avec un mélange de gaussiennes. Le critère d'apprentissage est alors défini comme le coût de mauvaise classification sur un ensemble de validation. L'apprentissage du BdF MEL consiste à minimiser cette erreur par un algorithme génétique.

Partant du constat que tous les phonèmes n'ont pas la même puissance de discrimination quant à la reconnaissance d'orateur, Kinnunen propose de construire un BdF MEL pour chaque phonème [Kinnunen, 2002]. La chaîne de traitement d'un enregistrement audio consiste donc d'abord à déterminer le phonème correspondant, puis à calculer les MFCC sur la base d'un BdF MEL dépendant du phonème. Ces vecteurs MFCC servent de caractéristiques pour la reconnaissance d'orateur. Les BdF MEL sont créés par pondération des sous-bandes par un critère de Fisher normalisé dépendant de la bande de fréquence et du phonème. Ces pondérations sont calculées à partir d'une base d'apprentissage.

2.4.3 Réseau neuronal

La littérature sur les réseaux de neurones est foisonnante et nous choisissons de ne pas la détailler ici. Il nous paraît néanmoins essentiel de rappeler les liens entre ceux-ci et l'apprentissage de transformées TF, qui s'exprime à travers les CNN, introduits par Fukushima [Fukushima, 1980] et développés par LeCun *et coll.* [LeCun, 1989, LeCun *et coll.*, 1989].

Les CNN (et les réseaux de neurones en général) constituent les premiers travaux à mettre en exergue l'intérêt de se reposer un maximum sur l'apprentissage automatique des descripteurs plutôt que sur des heuristiques conçues manuellement et dépendantes de l'application finale [LeCun *et coll.*, 1998]. Il paraît toutefois évident que tout système automatisé nécessite un apport minimal de connaissance *a priori* concernant la tâche en question. Dans le cas d'un réseau neuronal, l'incorporation de telles connaissances consiste à pré-définir l'architecture du réseau (*i.e.* le nombre de couches et de cartes par couche, la nature des opérations neuronales, *etc.*).

La reconnaissance automatique de formes bidimensionnelles (sur des images), telles que des chiffres manuscrits, suggère par exemple l'encodage d'invariances dans le système d'extraction des caractéristiques. Un réseau de neurones dont l'architecture permet de telles invariances, particulièrement adaptées au traitement des images, est appelé CNN (ou *Time-Delay Neural Network* s'il est appliqué à des séries temporelles [Lang *et coll.*, 1990, Sugiyama *et coll.*, 1991]). Ce type particulier de réseaux artificiels est devenu populaire car il a démontré de bien meilleures performances que d'autres variantes pour la reconnaissance automatique d'images [LeCun *et coll.*, 1998].

Un CNN combine trois idées architecturales pour assurer une invariance à la translation, au changement d'échelle et aux distortions :

- ◇ ***caractéristiques locales*** : les neurones d'une couche ne sont pas connectés à tous les neurones de la couche précédente, mais seulement à une partie connexe. Une telle configuration insiste sur l'extraction de caractéristiques locales, permettant ainsi d'obtenir une information topologique sur des données d'entrée (par exemple des bords ou de coins) ;
- ◇ ***partage de poids*** : mis à part le domaine spatial de la couche précédente sur lequel ils agissent, tous les neurones d'une carte sont contraints à posséder les mêmes caractéristiques. Ceci se traduit par l'identité à travers les neurones, des poids accordés à chaque synapse. En pratique, une telle contrainte conduit à effectuer une opération de convolution sur les données d'entrée, d'où le nom CNN ;
- ◇ ***sous-échantillonnage*** : la dernière étape nécessaire à l'encodage des invariances précédemment citées est une décimation *intelligente* des données filtrées.

En sus de la nécessité d'extraire des caractéristiques invariantes aux déformations couramment observées dans les images, ce choix de structure répond aussi à des problématiques théoriques et calculatoires. En effet, un grand nombre de paramètres (les poids) est synonyme d'une grande complexité du modèle, qui requiert alors de larges bases de données pour prévenir un phénomène de sur-apprentissage. En outre, un réseau complètement et librement connecté nécessiterait un grand espace mémoire et autant d'opérations à chaque étape de l'apprentissage du réseau.

En pratique, un CNN est composé d'une alternance d'opérations de convolution et de décimation pour l'extraction de caractéristiques discriminantes, puis d'un MLP réalisant une classification non-linéaire (figure 2.5 page suivante). Chaque carte de la première couche réalise successivement une convolution, l'ajout d'un biais et l'application d'une non-linéarité (couramment une fonction sigmoïde telle que tangente hyperbolique). De même, chaque opération de décimation est réalisée par le calcul d'une moyenne (ou d'un maximum) dans un voisinage local, suivie d'une multiplication par un facteur puis d'une non-linéarité iden-

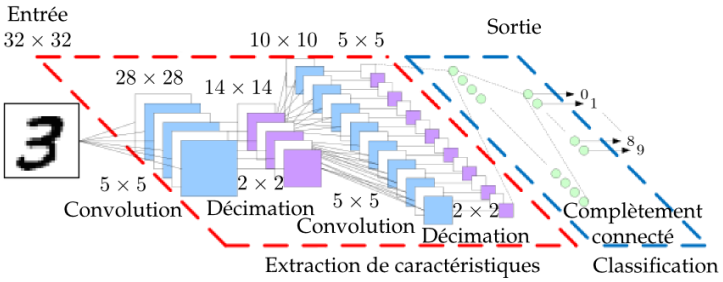


FIGURE 2.5 – Réseau de neurones convolutifs (reproduit et adapté de [Peeman *et coll.*, 2011]).

tique à celle des couches de convolution. Pour une carte de décimation, les voisinages locaux considérés par les différents neurones ne se chevauchent pas. Cette caractéristique implémente concrètement le sous-échantillonnage. En perspective de nos travaux, ces deux premières couches correspondent respectivement à un BdF et à une agrégation non-linéaires.

Les cartes des couches successives aux deux premières peuvent être connectées à plusieurs cartes de la couche précédente. En pratique, ces connexions sont choisies de manière à créer une dissymétrie dans le réseau, ayant pour effet d'apprendre des caractéristiques différentes par cartes.

La dernière partie d'un CNN est un MLP, dont l'ultime couche est le plus souvent constituée de neurones à base radiale euclidienne. Autrement dit, la dernière couche du MLP calcule une distance (au carré) par rapport à des prototypes (constitués par les poids des synapses).

Tous les paramètres d'un CNN sont appris simultanément par une technique de rétropropagation d'erreur introduite pour l'apprentissage statistique par Rumelhart *et coll.* [Rumelhart *et coll.*, 1986]. Ceux-ci incluent les filtres et les biais des couches de convolution, les facteurs et les biais des couches de décimation, ainsi que les poids neuronaux du MLP, incluant les prototypes représentant les classes à discriminer.

Lorsque ces prototypes sont fixes (*i.e.* les poids de l'ultime couche du MLP sont prédéfinis), la fonction de coût utilisée est simplement la somme des sorties du MLP, ce qui correspond à une perte de type *moindres carrés*. En revanche, lorsque ceux-ci doivent être appris, il est nécessaire d'introduire un terme compétitif écartant les solutions triviales pour les prototypes [LeCun *et coll.*, 1998].

Apprentissage vaste marge

Gardant à l'esprit les tenants et les aboutissants de notre problématique, il paraît essentiel de dégager certains travaux des nombreux réalisés sur les CNN : les modèles hybrides de réseaux vaste marge, consistant à remplacer le MLP se trouvant en fin du réseau par une SVM. Cela revient concrètement à changer le type de non-linéarité du classifieur (fonction sigmoïde `vs` redescription associée au noyau SVM considéré), la règle de décision conceptuelle (distance minimum `vs` hyperplan séparateur [4]) et la fonction de perte dirigeant l'apprentissage des paramètres (moindres carrés `vs` charnière simple ou quadratique). Dans cette configuration, en ne considérant que les deux premières couches du réseau et en oubliant leurs non-linéarités, le problème revient à l'apprentissage d'un BdF discriminant conjointement à un classifieur SVM.

Bien que ces travaux n'aient jamais été présentés sous cet angle, ils existent. L'un d'eux

4. Bien que conceptuellement différentes, ces deux règles de décision sont rigoureusement identiques pour un problème de reconnaissance binaire.

prend simplement la forme de deux apprentissages séquentiels : l'un d'un CNN complet (caractéristiques et MLP) puis un autre d'une SVM prenant comme descripteurs les transformations issues du réseau dont le MLP a été supprimé [Huang et LeCun, 2006]. Ces travaux font figure d'exception face à l'étude initiatrice de Zhong et Ghosh, qui intègre un critère similaire à celui d'une SVM usuelle (compromis marge et perte charnière) pour l'apprentissage d'un MLP [Zhong et Ghosh, 2000]. Ce réseau hybride, nommé *Decision Boundary Focused Neural Network* car il se concentre sur les exemples situés à l'intérieur d'une marge de classification (à l'instar d'une SVM), est revisitée plus tard par Collobert et Bengio dans le cadre d'une technique générale et efficace d'apprentissage d'un réseau par descente de gradient [Collobert et Bengio, 2004].

L'introduction de modèles hybrides réseau-SVM aux CNN revient à Long et Leow [Long et Leow, 2002]. S'inspirant du critère des moindres carrés, Long et Leow proposent une configuration dans laquelle une SVM adaptée à la régression est substituée à la dernière couche du MLP (qui implémente la règle de décision par distance minimale). Malheureusement, l'apprentissage par rétro-propagation des paramètres du réseau visant à extraire des caractéristiques discriminantes n'est pas en raccord exacte avec celui de la SVM en fin de chaîne, laissant la place à de nouvelles contributions. Ainsi, à la suite d'un apprentissage séquentiel [Huang et LeCun, 2006] et d'une fusion partielle d'un réseau convolutif et d'une SVM [Long et Leow, 2002], des réseaux convolutifs hybrides dont l'apprentissage d'une SVM est intégrée à la rétro-propagation sont proposés dans le cadre d'une perte charnière simple [Nagi *et coll.*, 2012] [5] et quadratique [Tang, 2013]. En pratique l'optimisation SVM est réalisée par descente de gradient dans l'espace des variables primales.

2.4.4 Transformée en ondelettes

Il est une remarque concernant l'analyse TF qu'il est nécessaire de formuler : un comportement basse fréquence se déploie par nature sur une grande échelle de temps, tandis qu'un comportement haute fréquence est très localisé temporellement. Une transformée de Fourier à court-terme est un outil qui permet de représenter l'évolution spectrale d'un signal au fil du temps, avec cependant un défaut majeur : les échelles temporelles auxquelles sont analysées les comportements à basses et hautes fréquences sont identiques. Une telle transformée ne tient pas compte de la précédente remarque, résultant dans le compromis suivant : une courte échelle temporelle d'analyse implique une haute résolution temporelle des comportements haute fréquence mais une représentation instables des tendances basse fréquence. Réciproquement, une large échelle temporelle d'analyse permet de caractériser correctement les comportements basse fréquence mais ne permet pas de localiser précisément les événements haute fréquence. En outre, une représentation de Fourier nécessite souvent beaucoup de coefficients pour retranscrire les irrégularités au sein d'un signal.

La transformée en ondelettes est une réponse à cette rigidité de la transformée de Fourier à court-terme, fondée sur une analyse multi-résolution [Mallat, 1989]. Ce type d'analyse considère un ensemble imbriqué d'espaces d'approximation de $L^2(\mathbb{R}) : \cdots \subset \mathcal{V}_{-1} \subset \mathcal{V}_0 \subset \mathcal{V}_1 \subset \ldots$, de telle sorte que $\lim_{j\to-\infty} \mathcal{V}_j = \{0\}$ et $\lim_{j\to+\infty} \mathcal{V}_j = L^2(\mathbb{R})$. Pour chaque espace d'approximation $\mathcal{V}_j$ ($j \in \mathbb{Z}$), on note $\mathcal{W}_j$ son complément orthogonal dans dans $\mathcal{V}_{j+1}$: $\mathcal{V}_{j+1} = \mathcal{W}_j \oplus \mathcal{V}_j$. L'ensemble $\mathcal{W}_j$ est aussi connu sous le nom d'espace de détails. Étant donnés une fonction f de $L^2(\mathbb{R})$ et un entier m, le principe d'analyse multi-résolution consiste à approximer f par une certaine fonction f_m de $\mathcal{V}_m$, qui elle-même est décomposable sous

5. Bien que les auteurs prétendent apprendre simultanément les filtres du réseau et l'hyperplan de la SVM, les imprécisions de l'exposé et l'algorithme décrit suggèrent que seule la SVM est entraînée.

la forme :

$$f_m = f_{m-1} + g_{m-1} = f_{m-2} + g_{m-2} + g_{m-1} = \cdots = f_0 + \sum_{j=0}^{m-1} g_j, \tag{2.1}$$

où pour tout j, f_j est une approximation de $\mathcal{V}_j$ et g_j est une fonction de détails de $\mathcal{W}_j$. Dans sa forme finale, la décomposition (2.1) de f_m (et de manière équivalente, l'approximation de f) est la somme d'une approximation *grossière* f_0 et de détails g_j.

Définition 2.4.3 (Ondelette [Mallat, 1999])**.**
Une ondelette est une fonction ψ de $L^2(\mathbb{R})$ de moyenne nulle et normalisée (et de manière informelle centrée autour de 0) :

$$\int_{-\infty}^{+\infty} \psi(t)\, dt = 0, \quad \int_{-\infty}^{+\infty} |\psi(t)|^2\, dt = 1.$$

Le terme *ondelette* vient de *petite onde*, signifiant d'une part sa capacité à être localisée dans le temps, et d'autre part sa nature ondulatoire autour de 0 (au point d'être affublée d'une moyenne nulle). De nombreuses fonctions vérifient la définition d'une ondelette, conduisant ainsi à plusieurs familles. Parmi celles couramment utilisées, certaines portent le nom de leur inventeur ou ont été nommées en honneur à leurs travaux dans le domaine, comme par exemple les ondelettes de Haar, Morlet, Meyer et Daubechies. Une famille d'ondelettes est spécifiquement choisie par un expert suivant plusieurs critères : la régularité, les moments nuls, la compacité du support, la simplicité analytique, la simplicité de la RI associée (par exemple une RI symétrique), *etc.*

Une décomposition en ondelettes exploite l'analyse multi-résolution en fournissant une base de l'espace d'approximation $\mathcal{V}_0$ et des espaces de détails $\mathcal{W}_0$, …, $\mathcal{W}_{m-1}$. Étant donnée une ondelette ψ, chaque espace $\mathcal{W}_j$ est lié à un certain degré de dilatation temporelle de ψ. La théorie des ondelettes assure qu'en translatant l'ondelette dilatée, il est possible d'obtenir une base de $\mathcal{W}_j$. Une représentation en ondelettes cherche alors les vecteurs de coordonnées $\boldsymbol{a}^{(j)}$ de f_j dans $\mathcal{V}_j$, et $\boldsymbol{d}^{(j)}$ de g_j dans $\mathcal{W}_j$ (pour $j \in \mathbb{N}_m$ et suivant (2.1)), par rapport aux bases fournies par ψ.

Comme nous l'avons vu, une ondelette est de moyenne nulle. Cette propriété assure que la transformée de Fourier $\hat{\psi}$ d'une ondelette ψ est nulle à la fréquence nulle : $\hat{\psi}(0) = \int_{-\infty}^{+\infty} \psi(t)\, dt = 0$. Une interprétation fréquentielle immédiate conduit donc à considérer une ondelette comme un filtre passe-bande. Cette notion de filtrage fait le lien avec le théorème suivant, qui permet de construire un BdF pyramidal (figure 2.6 page suivante) associé à une décomposition en ondelettes (l'annexe F donne plus de détails sur la définition des filtres) :

Théorème 2.4.1 (Filtrage pyramidal [Mallat, 1999, théo. 7.7, p. 344])**.**
Étant donnée une ondelette ψ, il existe des filtres $\boldsymbol{h}$ et $\boldsymbol{g}$, tels que pour toute décomposition multi-résolution et tout entier naturel j :

$$\boldsymbol{a}^{(j-1)} =\downarrow 2 \left[\boldsymbol{a}^{(j)} \star \boldsymbol{h}\right], \quad \boldsymbol{d}^{(j-1)} =\downarrow 2 \left[\boldsymbol{a}^{(j)} \star \boldsymbol{g}\right],$$

où $\boldsymbol{a}^{(j)}$ et $\boldsymbol{d}^{(j)}$ sont respectivement les vecteurs de coordonnées d'approximation et de détails à l'ordre j.

Lorsque l'on analyse un signal discret $\boldsymbol{x}$ plutôt qu'une fonction de carré sommable f, il suffit de déterminer un niveau m tel que $\boldsymbol{x} \in \mathcal{V}_m$ puis d'appliquer la décomposition en ondelettes directement à $\boldsymbol{x}$ (là où elle est réalisée à l'approximation f_m de f dans le cas continu, comme dans l'équation (2.1)). De plus, puisque nous travaillons dans un espace vectoriel de dimension finie, les bases des ensembles d'approximation et de détails sont elles aussi finies. Le dernier théorème énoncé est fondateur pour la transformée en ondelettes rapide [Mallat, 1999] qui calcule une représentation en ondelettes en $O(m)$ (où m est la taille des signaux discrets), quand une décomposition de Fourier est calculée en $O(m \log(m))$.

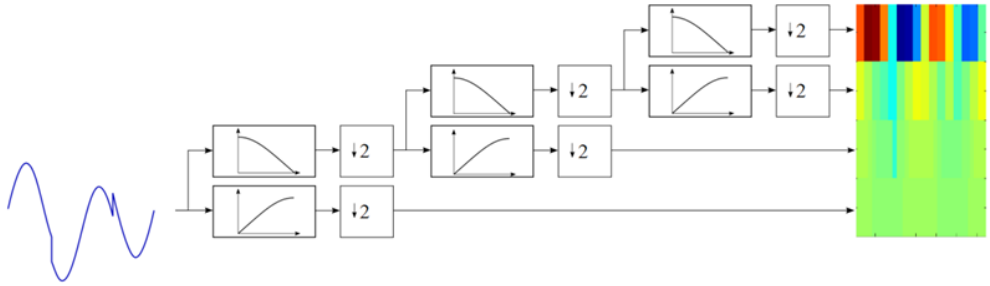


FIGURE 2.6 – Banc de filtres pyramidale associé à une décomposition en ondelettes de Haar.

Remarque 7.

Outre un meilleur pavage du plan TF (comparé à une représentation de Fourier), une décomposition en ondelettes permet aussi d'obtenir une représentation parcimonieuse pour des signaux qui ne sont pas purement harmoniques (à condition que l'ondelette choisie soit adaptée au signal analysé). Cette capacité à obtenir une représentation parcimonieuse est liée au nombre de moments nuls de l'ondelette choisie. Or le support d'une ondelette orthogonale croît au mieux linéairement avec le nombre de moments s'annulant [Mallat, 1999]. Ainsi, les propriétés de parcimonie et de localisation temporelle des irrégularités (compacité du support de l'ondelette) évoluent de manière diamétralement opposée. Les ondelettes de Daubechies sont optimales au sens de ce compromis : leur support est réduit au minimum théorique pour un nombre de moments nuls donné.

Remarque 8.

À l'instar des BdF de synthèse, il existe des propriétés concernant la reconstruction d'un signal décomposé en ondelettes. Cependant, pour des applications en reconnaissance, nous ne nous intéressons pas à cette partie de synthèse.

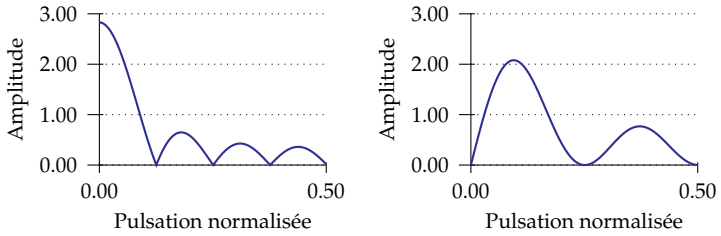
Par analogie avec les BdF présentés précédemment, nous pouvons noter que le BdF pyramidal de la figure 2.6 peut être mis sous la forme d'un BdF à un seul étage (figure 2.3 page 40). Dans le cas d'une ondelette de Haar $\psi : t \in \mathbb{R} \mapsto \chi_{[0,1]} \operatorname{Signe}(\frac{1}{2} - t)$ (où $\chi_{[0,1]}$ est la fonction caractéristique du compact $[0, 1]$, retournant 1 si son argument est entre 0 et 1, et 0 sinon), les distributions spectrales des filtres sont représentées sur la figure 2.7 page suivante.

L'une des spécificités d'une représentation en ondelettes discrète est le pavage dyadique et non-uniforme du plan temps fréquence. En particulier, la résolution temporelle est importante pour les hautes fréquences. Il est toutefois envisageable de paver le plan TF de manière uniforme en décomposant les hautes fréquences de la même manière que les basses fréquences lors du calcul pyramidal des coefficients d'ondelettes (ce qui rendrait le BdF de la figure 2.6 horizontalement symétrique). Cette représentation est appelée *transformation en paquet d'ondelettes* [Saito et Coifman, 1995].

Tout BdF bicanal (à un étage) dit paraunitaire (*i.e.* répondant à certains critères de symétrie et d'inversibilité [Strang et Nguyen, 1996]) et possédant des filtres d'ordre $2q + 1$ (*i.e.* de taille $2(q + 1)$) peut être paramétré par $q + 1$ angles (seulement q si le filtre passe-haut est caractérisé par un moment nul) à choisir dans $[0, \pi[$, sous le nom de structure en treillis (*lattice structure*) [Strang et Nguyen, 1996, théo. 4.7, p. 139]. Ce type de paramétrage peut notamment être utilisé pour des représentations en ondelettes et en paquet d'ondelettes orthogonales.

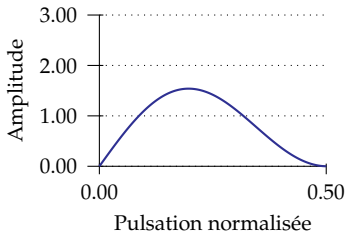
Apprentissage discriminant

Les premiers travaux concernant l'apprentissage d'une décomposition en ondelettes discrète sont de [Saito et Coifman, 1995]. Ils visent, à travers l'algorithme *Local Discriminant*

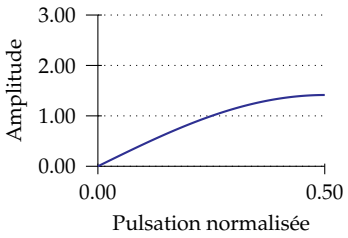


(a) Filtre $\boldsymbol{h_1}$ ($N_1 = 8$).

(b) Filtre $\boldsymbol{h_2}$ ($N_2 = 8$).



(c) Filtre $\boldsymbol{h_3}$ ($N_3 = 4$).



(d) Filtre $\boldsymbol{h_4}$ ($N_4 = 2$).

FIGURE 2.7 – Filtres d'une transformée en ondelettes de Haar représentés dans le domaine fréquentiel.

Basis (LDB), à sélectionner une base d'ondelettes pertinente au sein d'un paquet d'ondelettes et constituent une extension de l'algorithme initialement proposé par [Coifman et Wickerhauser, 1992] pour la compression à une tâche de reconnaissance de signaux. La différence majeure entre les deux algorithmes réside dans le critère à optimiser : [Coifman et Wickerhauser, 1992] minimise l'entropie de Shanon tandis que [Saito et Coifman, 1995] maximise le pouvoir de discrimination d'une base TF fondée sur une mesure de type Kullback-Leibler. Tandis que LDB compare les distributions énergétiques dans le plan TF donné par une décomposition en ondelettes, l'extension décrite dans [Saito et Coifman, 1997] s'intéresse aux densités de probabilité estimées des signaux projetés sur des atomes TF. Une nouvelle extension de LDB consiste à considérer une variante du critère de Fisher comme objectif à maximiser [Vautrin *et coll.*, 2009]. Ces derniers travaux proposent une différence des variabilités inter et intra-classe, plutôt que le traditionnel ratio. De la sorte, on obtient un critère additif qui peut être utilisé au sein de l'algorithme original *Best Basis* [Coifman et Wickerhauser, 1992]. Cette nouvelle méthode est appliquée à la problématique ICM sans toutefois être comparée à celle de Saito et Coifman [Saito et Coifman, 1995].

L'algorithme LDB proposé par Saito et Coifman [Saito et Coifman, 1995] détermine une base d'ondelettes discriminante au sein d'une décomposition en paquet d'ondelettes. Cette dernière décomposition est prédéfinie par un expert sur la base d'une certaine ondelette mère (par exemple une ondelette de Daubechies). Il est possible de déterminer automatiquement une telle ondelette afin d'améliorer le résultat obtenu par LDB [Strauss *et coll.*, 2003]. Dans ces derniers travaux, une décomposition en paquet d'ondelettes est paramétrée par les angles d'une structure en treillis. Un algorithme génétique permet alors de sélectionner des angles maximisant la mesure de discrimination optimale rendue par LDB. Des applications en santé (notamment à la reconnaissance d'Électro-Cardiogrammes (ECG)) ont mis en valeur l'intérêt d'une telle approche, même si celle-ci est plus coûteuse en temps de calcul que LDB. Il existe aussi des approches similaires de construction automatique de transformées en ondelettes discrètes, qui minimisent l'erreur de classification sur un ensemble de validation via des algorithmes génétiques [Ray et Chan, 2001, Jones *et coll.*, 2001].

La structure en treillis a aussi fait l'objet d'une optimisation conjointe d'une transformée en ondelettes discrète avec une SVM [Strauss et Steidl, 2002, Farina *et coll.*, 2007]. Cette

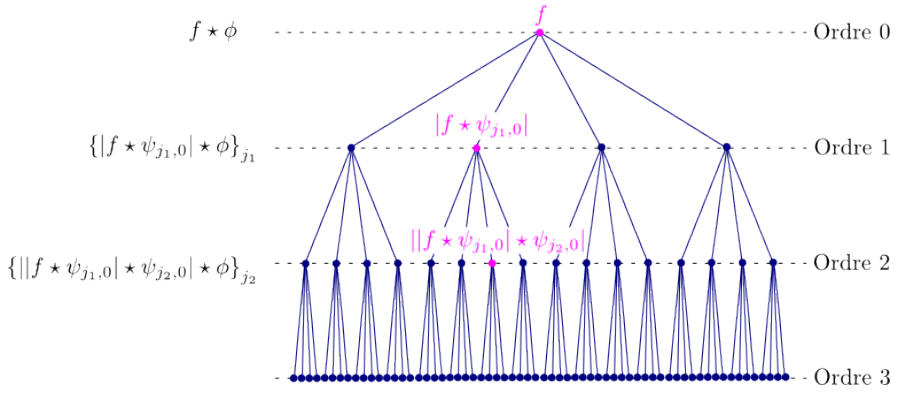


FIGURE 2.8 – Schématisation d'une transformée en diffusion d'ondelettes (reproduite et adaptée de [Bruna et Mallat, 2013]).

optimisation est réalisée en maximisant la distance entre les centres des deux classes considérées dans l'espace de redescription d'un noyau à base radiale [Strauss et Steidl, 2002] ou en minimisant l'erreur de validation croisée [Farina *et coll.*, 2007], par recherche exhaustive des angles optimaux sur une grille uniforme. Une étude plus approfondie est décrite dans l'ouvrage [Neumann *et coll.*, 2005] : différents critères sont comparés (borne rayon-marge, marge SVM, alignement de noyaux, distance entre centres des classes et ratio de Fisher) grâce à un algorithme de recherche exhaustive par grille adaptative. La comparaison expérimentale semble privilégier les critères simples tels que la distance entre les centres des classes et l'alignement de noyaux.

L'une des contributions notables à la détermination automatique d'une transformée en ondelettes discrète discriminante consiste à formaliser l'apprentissage des angles de la structure en treillis comme un problème d'apprentissage de noyaux multiples (*Multiple Kernel Learning*, MKL) comportant un très grand nombre (voir une infinité) de noyaux [Yger et Rakotomamonjy, 2011]. Ce dernier ouvrage propose un algorithme par ensemble actif pour minimiser le risque structurel d'un noyau multiple construit sur une quantité effective restreinte de coefficients d'ondelettes. Il est à noter que le résultat d'un tel apprentissage se réduit au mariage d'une transformation en ondelettes puis d'une SVM dont le noyau n'est pas classique puisque généralement constitué d'une somme de sous-noyaux calculés sur certains coefficients de la décomposition.

2.4.5 Diffusion d'ondelettes

Récemment, une nouvelle transformée à la croisée d'une décomposition en ondelettes et d'un réseau de neurones a été introduite sous le nom *transformée par diffusion* (*Scattering transform*) [Mallat, 2012], visant à fournir des transformations localement invariantes aux translations et linéaires par rapport aux déformations, pour des tâches de reconnaissance. Cette transformée repose sur une transformée en ondelettes mais à la différence de celle-ci, elle introduit des termes d'ordres supérieurs en répétant en cascade la décomposition (mêlée à des opérations de module et de moyennage) sur chaque contribution, à l'image d'un réseau neuronal (figure 2.8). La théorie montre qu'une transformée par diffusion étend les MFCC (ainsi que les descripteurs *Scale-Invariant Feature Transform* (SIFT) couramment utilisés en traitement d'images [Bruna et Mallat, 2013]) en calculant des coefficients de spectre de modulation d'ordres supérieurs, caractérisent l'attaque et les modulations d'amplitudes d'un signal (section 2.2) [Andén et Mallat, 2014].

Continuant sur cette analogie, une transformée par diffusion peut être assimilée à un CNN (sans le classifieur MLP) dont les particularités sont les suivantes :

- ◇ chaque opération de convolution est construite sur une ondelette à une échelle donnée ;
- ◇ les opérateurs de décimation sont remplacés par un module suivi d'un moyennage gaussien ;
- ◇ aucun apprentissage n'est nécessaire.

Cette nouvelle transformée a été appliquée avec succès à la reconnaissance de genres musicaux [Andén et Mallat, 2011] et de parole [Andén et Mallat, 2014]. Les auteurs n'ont pas fondé leurs approches directement sur les coefficients de la décomposition par diffusion mais sur la DCT du logarithme de celle-ci, à l'instar des MFCC [Andén et Mallat, 2011], ou sur les coefficients d'une double transformée par diffusion (l'une en temps et l'autre en fréquence), supposée invariante aux translations temporelles et fréquentielles [Andén et Mallat, 2014]. Des apports similaires ont été démontré sur la reconnaissance de chiffres manuscrits et la classification de textures [Bruna et Mallat, 2011, Bruna et Mallat, 2013]. Il est intéressant de noter que dans toutes les expérimentations numériques conduites, le MLP intrinsèque aux CNN est substitué par une SVM. Une fois de plus, on retrouve une cadre convolutif vaste marge, mais sans apprentissage des filtres.

2.4.6 Dictionnaire

Les dictionnaires sont une généralisation des bases et des trames utilisées dans les transformées usuelles de traitement du signal (Fourier, ondelettes) à des ensembles générateurs surcomplets (qualifiés aussi de redondants) [Mallat et Zhang, 1993]. Deux particularités déjà mentionnées à propos d'une transformée en ondelettes peuvent être formulées pour les dictionnaires : ils visent à décomposer des signaux comme des sommes pondérées d'atomes TF (*i.e.* localisés en temps et en fréquence), comprenant seulement peu de contributions (correspondant ainsi à une représentation parcimonieuse). Cette dernière notion (la parcimonie) est un concept clef de la représentation par dictionnaire : imaginons un signal constitué d'irrégularités de plusieurs sortes. Une décomposition de Fourier ou en ondelettes nécessitera probablement beaucoup d'atomes pour approcher le signal, car ceux-ci ne sont pas adaptés à toutes les irrégularités. Un dictionnaire offrant un large choix permettra en revanche de représenter le signal en question de manière parcimonieuse en sélectionnant les atomes (potentiellement de familles différentes) adaptés. Après sélection des atomes pertinents, un dictionnaire permet de décrire un sous-espace vectoriel de $\mathcal{X}$ dans lequel vivent les données étudiées.

Définition 2.4.4 (Dictionnaire)**.**
Soit $\mathcal{H}$ un espace de Hilbert. Un dictionnaire $\mathcal{D}$ est une collection d'éléments $g_{\boldsymbol{\theta}}$ de $\mathcal{H}$, normalisés ($\|g_{\boldsymbol{\theta}}\|_{\mathcal{H}} = 1$) et indexés par un ensemble (potentiellement infini) $\mathcal{P} : \mathcal{D} = \{g_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{P}}$. Lorsque la complétude de l'espace vectoriel engendré par $\mathcal{D}$ rejoint $\mathcal{H}$, le dictionnaire $\mathcal{D}$ est dit complet.

Lorsque $\mathcal{H}$ est un espace vectoriel de dimension finie et $\mathcal{P}$ est un ensemble fini, le dictionnaire $\mathcal{D}$ peut être représenté par une matrice $\boldsymbol{D}$ dont chaque colonne est un atome $g_{\boldsymbol{\theta}}$.

L'utilisation d'un dictionnaire vise à décomposer tout vecteur f de $\mathcal{H}$ comme une combinaison linéaire d'atomes : $f = \sum_{k\in\mathbb{N}} \mu_k g_{\boldsymbol{\theta_k}}$, où $\{g_{\boldsymbol{\theta_k}}\}_{k\in\mathbb{N}}$ est un sous-ensemble dénombrable de $\mathcal{D}$ et $\boldsymbol{\mu}$ est un vecteur de pondération à déterminer. Un algorithme glouton pour déterminer une telle décomposition a été proposé par Mallat et Zhang sous le nom *Matching Pursuit* (MP) [Mallat et Zhang, 1993]. MP est un algorithme itératif minimisant le résidu $r_j = f - \sum_{0\leq k\leq j} \mu_k g_{\boldsymbol{\theta_k}}$ ($j \in \mathbb{N}\cup\{-1\}$). Il construit simultanément le sous-dictionnaire $\{g_{\boldsymbol{\theta_k}}\}_{k\in\mathbb{N}}$ et le vecteur de pondération $\boldsymbol{\mu}$ en sélectionnant à chaque étape j ($j \in \mathbb{N}$) un atome $g_{\boldsymbol{\theta_j}}$ vérifiant $\left\|\langle g_{\boldsymbol{\theta_j}} \mid r_{j-1}\rangle_{\mathcal{H}}\right\| \geq \epsilon \sup_{\boldsymbol{\theta}\in\mathcal{P}} \left\|\langle g_{\boldsymbol{\theta}} \mid r_{j-1}\rangle_{\mathcal{H}}\right\|$ (où ϵ est un facteur d'optimalité à

prendre dans $]0,1]$) et en définissant $\mu_j = \langle g_{\boldsymbol{\theta_j}} \mid r_{j-1} \rangle_{\mathcal{H}}$. Pour cet algorithme, la suite des normes des résidus $\left(\|r_j\|_{\mathcal{H}}\right)_{j\in\mathbb{N}}$ décroît exponentiellement vers 0.

Dans le cas d'un espace de Hilbert $\mathcal{H}$ de dimension finie et d'un dictionnaire $\mathcal{D}$ de taille finie, un tel algorithme fournit une solution approchée au problème d'approximation parcimonieuse :

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{minimiser}} & \|\boldsymbol{s} - \boldsymbol{D}\boldsymbol{x}\|_{\ell_2}^2 \\ \text{tel que} & \|\boldsymbol{x}\|_{\ell_0} \leq A, \end{array} \tag{2.2}$$

où $\boldsymbol{s}$ est un signal de $\mathcal{H}$ à approximer et A est un seuil (entier naturel) prédéfini. Pour ce problème, MP converge en A itérations.

Remarque 9.
De nombreuses variantes d'algorithmes de résolution et de formalisation du problème (2.2) existent, parmi lesquels nous pouvons citer *Orthogonal Matching Pursuit* [Pati *et coll.*, 1993], *Basis Pursuit* [Chen *et coll.*, 1998] et *Least Absolute Shrinkage and Selection Operator* [Tibshirani, 1996].

Remarque 10.
Une telle décomposition atomique est une réponse alternative à la classe de transformées de Cohen pour effacer les interférences introduites lors du calcul d'une distribution énergétique de Wigner-Ville. Dans ce contexte, connaissant les contributions effectives du signal dans le domaine temporel, il est possible de construire une représentation énergétique par sommation des transformations de Wigner-Ville de chaque atome [Mallat et Zhang, 1993]. On obtient ainsi une représentation dénuée des termes de corrélation croisée.

Si les travaux précédents ont pour but l'approximation parcimonieuse de signaux, des méthodes analogues ont été exclusivement développées pour la classification. Celles-ci combinent les avantages des approches reconstructives et discriminantes afin d'être robustes au bruit et aux points atypiques dans l'ensemble d'apprentissage. Soit un ensemble d'apprentissage de signaux $\{(\boldsymbol{s_i}, y_i)\}_{1\leq i\leq n}$ provenant de K classes (*i.e.* $\forall i \in \mathbb{N}_n \colon y_i \in \mathbb{N}_K$). Le problème dont il est question dans [Huang et Aviyente, 2006], combinant parcimonie, reconstruction et discrimination, est :

$$\underset{\boldsymbol{x_1},\ldots,\boldsymbol{x_n}}{\text{maximiser}} \quad F\left(\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}\right) - \lambda_0 \sum_{i=1}^n \|\boldsymbol{x_i}\|_{\ell_0} - \lambda_1 \sum_{i=1}^n \|\boldsymbol{s_i} - D\boldsymbol{x}_i\|_{\ell_2}^2 ,$$

où F est un critère de Fisher représentant le rapport des dispersions inter-classe et intra-classe. L'algorithme de résolution proposé dans [Huang et Aviyente, 2006] est une adaptation de MP orthogonal [Pati *et coll.*, 1993].

Apprentissage discriminant

L'apprentissage de dictionnaires à des fins d'approximation fait l'objet de recherches depuis deux décennies. Étant donné une collection de signaux $(\boldsymbol{s}_i)_{1\leq i\leq n}$, la formalisation la plus courante du problème d'apprentissage de dictionnaires génératifs est :

$$\begin{array}{ll} \underset{\boldsymbol{D},\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n}{\text{minimiser}} & \displaystyle\sum_{i=1}^n \|\boldsymbol{s}_i - \boldsymbol{D}\boldsymbol{x}_i\|_{\ell_2}^2 \\ \text{tel que} & \begin{cases} \forall i \in \mathbb{N}_n \colon \|\boldsymbol{x}_i\|_{\ell_0} \leq A \\ \mathrm{Diag}(\boldsymbol{D}^T\boldsymbol{D}) \preccurlyeq \mathbb{1}, \end{cases} \end{array}$$

où la dernière contrainte assure que les atomes du dictionnaire sont de norme au plus unitaire. Différentes approches, notamment probabilistes [Olshausen et Field, 1996, Olshausen

et Field, 1997, Lewicki et Sejnowski, 2000] et algébriques [Aharon *et coll.*, 2006] ont été proposées.

Nous nous intéressons dans cette section à l'apprentissage de dictionnaires discriminants. Les premières tentatives sur ce sujet consistent à apprendre un dictionnaire $\boldsymbol{D_k}$ différent pour chacune des K classes de signaux à distinguer [Skretting et Husøy, 2006, Yang *et coll.*, 2010]. Toute nouvelle donnée $\boldsymbol{s}$ est attribuée à la classe dont le dictionnaire minimise le résidu de l'approximation parcimonieuse [Skretting et Husøy, 2006] :

$$\underset{k\in\mathbb{N}_K}{\arg\min} \left(\min_{\substack{\boldsymbol{x} \\ \|\boldsymbol{x}\|_{\ell_0}\leq A}} \|\boldsymbol{s}-\boldsymbol{D_k x}\|_{\ell_2}^2 \right),$$

ou dont le résidu partiel (d'une approximation parcimonieuse généralisée, régularisée par un réel positif λ) est minimum [Yang *et coll.*, 2010] :

$$\underset{k\in\mathbb{N}_K}{\arg\min} \|\boldsymbol{s}-\boldsymbol{D_k x_k}\|_{\ell_2}^2, \quad (\boldsymbol{x_1},\ldots,\boldsymbol{x_K}) = \underset{\boldsymbol{x'_1},\ldots,\boldsymbol{x'_K}}{\arg\min} \left\| \boldsymbol{s}-\sum_{k=1}^{K} \boldsymbol{D_k x'_k} \right\|_{\ell_2}^2 + \lambda \sum_{k=1}^{K} \left\|\boldsymbol{x'_k}\right\|_{\ell_1}.$$

Si les approches précédentes entraînent de manière disjointe des dictionnaires génératifs puis des fonctions discriminantes, les travaux de [Rodriguez et Sapiro, 2008] ouvrent la voie de l'apprentissage simultanément génératif, parcimonieux et discriminant. Cette dernière étude étend celle portée par [Huang et Aviyente, 2006] en formalisant le problème d'apprentissage de dictionnaire discriminant par :

$$\begin{array}{ll} \underset{\boldsymbol{D},\boldsymbol{x_1},\ldots,\boldsymbol{x_n}}{\text{maximiser}} & F\left(\{(\boldsymbol{x_i},y_i)\}_{1\leq i\leq n}\right) - \lambda_1 \sum_{i=1}^{n} \|\boldsymbol{s_i}-D\boldsymbol{x}_i\|_{\ell_2}^2 \\ \text{tel que} & \forall i\in\mathbb{N}_n\colon \|\boldsymbol{x}_i\|_{\ell_0}\leq A, \end{array}$$

où F est toujours un critère de Fisher. Une solution du problème d'optimisation énoncé est approchée par une descente alternée dans laquelle une première étape de codage parcimonieux traite le terme de reconstruction classe par classe tandis que celui de discrimination est considéré globalement ; puis une deuxième étape met à jour le dictionnaire suivant la technique de [Aharon *et coll.*, 2006].

Cette approche combinant les trois propriétés clefs de reconstruction, parcimonie et discrimination est plus longuement développée dans [Mairal *et coll.*, 2008, Mairal *et coll.*, 2009]. Les auteurs considèrent une fonction de perte *softmax* : $L_k\colon \boldsymbol{a}\in\mathbb{R}^K \mapsto \ln\left(\frac{\sum_{k'=1}^{K} \mathrm{e}^{a_{k'}}}{\mathrm{e}^{a_k}}\right)$, qui est une généralisation du coût logistique à plusieurs classes. Étant données une classe numérotée k et une fonction de discrimination bilinéaire de la forme $f_k\colon (\boldsymbol{s},\boldsymbol{x},\theta)\mapsto \boldsymbol{s}\boldsymbol{W_k}\boldsymbol{x}+b_k$ (il existe aussi une version linéaire [Mairal *et coll.*, 2008, Mairal *et coll.*, 2009]), où $\boldsymbol{s}$ est un signal, $\boldsymbol{x}$ est sa représentation parcimonieuse et $\theta = (\boldsymbol{W_1},\ldots,\boldsymbol{W_K},b_1,\ldots,b_K)$ est un ensemble de paramètres des fonctions de discrimination, le codage parcimonieux du signal $\boldsymbol{s}$ de la classe k est donné par la résolution du problème :

$$S_k(\boldsymbol{s},D,\theta) = \min_{\boldsymbol{x}} L_k\left((f_{k'}(\boldsymbol{s},\boldsymbol{x},\theta))_{k'=1}^{K}\right) + \lambda_0\|\boldsymbol{x}\|_{\ell_1} + \lambda_1\|\boldsymbol{s}-\boldsymbol{D}\boldsymbol{x}\|_{\ell_2}^2.$$

Trois termes apparaissent dans cette expression, chacun correspondant à l'une des propriétés clefs précédemment citées. La classe attribuée à un signal inédit $\boldsymbol{s}$ est alors $\underset{k\in\mathbb{N}_K}{\arg\min}\ S_k(\boldsymbol{s},D,\theta)$. Afin de construire simultanément tous les modèles mis en jeu, l'algorithme proposé dans [Mairal *et coll.*, 2008, Mairal *et coll.*, 2009] alterne cette étape de codage

parcimonieux avec la mise à jour du dictionnaire et des fonctions de discrimination en résolvant le problème d'optimisation :

$$\begin{aligned} \underset{\boldsymbol{D},\theta}{\text{minimiser}} \quad & \sum_{k=1}^{K} \sum_{\substack{i=1 \\ y_i=k}}^{n} \left[\mu L_k \left((S_{k'}(\boldsymbol{s_i}, D, \theta))_{k'=1}^{K} \right) + (1-\mu) S_k(\boldsymbol{s_i}, D, \theta) \right] + \lambda \psi(\theta) \\ \text{tel que} \quad & \operatorname{Diag}\left(\boldsymbol{D}^T \boldsymbol{D}\right) \preccurlyeq \mathbb{1}, \end{aligned}$$

où $\psi : \theta \mapsto \sum_{k=1}^{K} \left(\|\boldsymbol{W_k}\|_{\ell_2}^2 + \|b_k\|_{\ell_2}^2 \right)$ est une fonction de régularisation des paramètres de discrimination (pondérée par un réel positif λ) et μ contrôle le compromis entre la marge de décision $L_k \left((S_{k'}(\boldsymbol{s}, D, \theta))_{k'=1}^{K} \right)$ et l'erreur d'approximation discriminante $S_k(\boldsymbol{s_i}, D, \theta)$. Une version stochastique de cet algorithme a été proposée ultérieurement [Mairal *et coll.*, 2012].

2.5 RECONNAISSANCE PRÉCOCE

2.5.1 Motivations

Comme nous avons pu le constater à travers les pages précédentes, l'apprentissage automatique s'est naturellement incliné vers la reconnaissance automatique de signaux et plus généralement de séries temporelles. De nombreuses méthodes ont été développées afin de décrire au mieux les signaux d'une certaine famille, tout en tenant compte de la nature transitoire de ceux-ci. Malgré cette non-stationnarité, il existe toutefois une forme de redondance de l'information au fil du temps (concernant particulièrement les signaux de longue durée) conduisant à la question suivante : est-il possible d'attribuer une étiquette à un signal en observant seulement une partie de celui-ci ?

Cette question soulève trois points essentiels et compétitifs. Le premier est le pouvoir de ***discrimination*** de la méthode de reconnaissance mise en jeu. C'est un point critique dans toute tâche de reconnaissance. Le second est la ***précocité*** de la prise de décision : l'aptitude à décider « au plus tôt », en observant seulement le début d'un signal. Ceci passe souvent par l'augmentation de l'ensemble d'apprentissage à disposition avec des données partielles. La précocité peut évoluer en opposition à la puissance de discrimination si l'on dépasse le seuil de redondance du signal. Il faut alors accepter qu'une observation partielle du signal correspond à une acquisition partielle de l'information. Enfin, cette question renvoie au problème de ***fiabilité***, *i.e.* la garantie que la décision prise avec une séquence partielle est au plus près de celle qui aurait été prise en observant la série temporelle dans sa globalité. Imposer qu'un système de reconnaissance précoce soit très fiable peut impacter sa puissance de discrimination (y compris avec l'information totale) et diminuer sa capacité à prendre une décision précocement.

L'aptitude à prendre une décision automatiquement et de manière précoce est un atout très apprécié dans des applications aussi diverses que la sécurité (surveillance vidéo, détection d'attaque et alerte d'évacuation d'urgence concernant un incendie ou un séisme) [Suriani *et coll.*, 2013], les soins médicaux (prévention de l'infarctus, déclenchement d'une maladie, assistance des personnes âgées) [Neill *et coll.*, 2005] et le divertissement (reconnaissance de mouvements) [Nowozin et Shotton, 2012]. En pratique, cette situation intervient dans le traitement des signaux de longue durée et nécessite des techniques pouvant se déployer en temps réel. Pour ces raisons, les approches utilisées proviennent plus souvent de modèles séquentiels [Hoai et De la Torre, 2014] que de modèles démocratiques [Aucouturier *et coll.*, 2007].

Dans la suite, nous faisons état de travaux jalonnant la recherche en reconnaissance précoce, en distinguant deux problèmes liés mais différents. Le premier est celui de la classification,

qui suppose de distinguer différentes actions ou situations. Le second est celui de la détection (ou de la classification à une classe), dans lequel l'intérêt premier est de signaler l'apparition d'un évènement.

2.5.2 Classification

Définition 2.5.1 (Classification précoce)**.**
Soit un ensemble de séquences d'apprentissage étiquetées par $+1$ ou -1. Un problème de classification binaire précoce consiste à assigner à chaque instant, une étiquette qualifiant une séquence dans son ensemble, parmi $\{+1, -1, ?\}$, où ? indique l'absence de décision. De plus, l'attribution d'une telle étiquette ne peut dépendre des instants futurs à celui observé.

L'idée de classification précoce naît avec les travaux [Rodriguez et Alonso, 2002], qui décrivent une technique de *boosting* appliquée à des apprenants faibles construits sur des prédicats tels que « ... croît » ou « ... stagne ». Ces prédicats sont définis sur des intervalles (régions temporelles) des séquences analysées et une classification précoce est réalisée en omettant les régions non-observées. De manière plus globale, la classification précoce a émergé dans différents domaines de l'intelligence artificielle, particulièrement dans celui de la reconnaissance d'intentions [Bui *et coll.*, 2002, Liao *et coll.*, 2005], *i.e.* l'inférence des plans d'un agent intelligent, à partir de ses actions ou des effets de celles-ci.

Par la suite, la reconnaissance précoce d'actions humaines est devenue une thématique centrale dans la communauté de vision artificielle. [Davis et Tyagi, 2006] reformule le problème dans un cadre d'inférence probabiliste et utilise un ratio de probabilités pour prendre une décision. Cette approche suppose qu'un HMM génératif appris de manière usuelle sur les événements complets peut aussi générer des séquences partielles. [Ryoo, 2011] modélise la partie masquée d'un événement comme une variable latente et propose une extension du paradigme de sacs de mots, sous la forme d'un histogramme dynamique de caractéristiques spatio-temporelles. [Li et Fu, 2012] propose de nouveaux descripteurs, construits comme des histogrammes de vitesses orientées, et couple un HMM à un modèle auto-régressif afin de tirer partie de puissances prédictives complémentaires. En effet, un HMM est supposé prédire correctement des formes globales dans des séries temporelles, tandis que le modèle auto-régressif est attendu sur la prédiction des valeurs futures dans un rayon local de la série temporelle. Ces travaux sont par la suite étendus à la prise en compte du contexte dans le processus décisionnel [Li et Fu, 2014]. Enfin, [Cao *et coll.*, 2013] présente une étude générale (englobant le concept de classification précoce) dans laquelle l'absence d'information n'est pas nécessairement située à la fin de la série temporelle, mais n'importe où. Les auteurs mettent en place un cadre bayésien dans lequel la vraisemblance est calculée à partir de l'erreur d'approximation parcimonieuse de la séquence partiellement observée à classer.

Toutefois, un réel essor est donné avec les travaux de Xing *et coll.*, qui proposent une méthode capable de classer une série temporelle précocement en assurant que la décision aurait été identique en analysant la séquence dans sa globalité [Xing *et coll.*, 2009]. Cette approche se fonde sur la règle de décision du plus proche voisin et le principe de distance minimale de prédiction. Considérons un ensemble de séquences d'apprentissage $\{(\boldsymbol{s_i}, y_i)\}_{1 \leq i \leq n}$, et pour un certain indice temporel t, la troncature $\boldsymbol{s_i}|_t$ de la série $\boldsymbol{s_i}$ ($i \in \mathbb{N}_n$) correspondant à ses t premières valeurs. On appelle *plus proches voisins inverses* de $\boldsymbol{s_i}$, toutes les séquences de l'ensemble d'apprentissage dont le plus proche voisin est $\boldsymbol{s_i}$. Avec ces définitions, la distance minimale de prédiction de $\boldsymbol{s_i}$ correspond au plus petit indice temporel t tel que les plus proches voisins inverses de $\boldsymbol{s_i}|_t$ existent et ne changent pas lorsque le reste de la série est progressivement observé (*i.e.* pour tout indice t' pris entre t et T, les plus proches voisins inverses de $\boldsymbol{s_i}|_{t'}$ existent et coïncident avec ceux de $\boldsymbol{s_i}$). Une séquence inédite $\boldsymbol{s}$ dont on connaît une observation partielle $\boldsymbol{s}|_t$ peut donc être classée dès qu'on lui

trouve un plus proche voisin de distance minimale de prédiction au moins égale à t ; sinon, la prise de décision est remise à plus tard.

Pour accorder plus de souplesse à leur méthode, Xing *et coll.* proposent de regrouper les données d'apprentissage de manière hiérarchique puis de déterminer une distance minimale de prédiction par groupe plutôt que par série temporelle. Un nouveau degré de liberté est apporté à la méthode en ignorant les données d'apprentissage instables (*i.e.* mal classées dans un schéma un en dehors (*Leave-One-Out*, LOO)) dans le calcul de la distance minimale de prédiction [Xing *et coll.*, 2012]. En outre, cette technique relâchée bénéficie d'un apprentissage plus rapide.

La question de la fiabilité en classification précoce a été analysée de manière probabiliste récemment [Parrish *et coll.*, 2013] : soient τ un seuil de probabilité ($\tau \in [0, 1]$), $f : \mathcal{X} \to \mathbb{R}$ une fonction discriminante (linéaire ou quadratique) et $\hat{f} : \mathcal{X} \to \mathcal{Y}$ une fonction de prise de décision, $\boldsymbol{x}$ un signal discret pris dans $\mathcal{X}$ et d'étiquette y, et $\boldsymbol{z}$ son observation partielle. Parrish *et coll.* se proposent de répondre à la question : « peut-ton donner un avis concernant l'étiquette de l'observation partielle $\boldsymbol{z}$ en assurant que celui-ci est identique à la décision qui serait prise avec une information complète $\boldsymbol{x}$, à une probabilité au moins égale à τ » ? Les auteurs répondent positivement en se plaçant *de facto* dans un cadre probabiliste et en considérant les signaux comme des observations de variables aléatoires Z et X (de distribution de probabilité p). Parrish *et coll.* cherchent ainsi à assurer que :

$$\mathbb{P}(\hat{f}(X) = \hat{f}(\boldsymbol{z}) \mid Z = \boldsymbol{z}) \geq \tau,$$

avec

$$\mathbb{P}(\hat{f}(X) = \hat{f}(\boldsymbol{z}) \mid Z = \boldsymbol{z}) = \int_{\left\{\boldsymbol{x} \in \mathcal{X},\ \hat{f}(\boldsymbol{x}) = \hat{f}(\boldsymbol{z})\right\}} p(\boldsymbol{x} \mid \boldsymbol{z})\, \boldsymbol{dx}.$$

Cette propriété est difficile à vérifier puisqu'elle nécessite de calculer l'intégrale. On remarque tout de même que la proposition précédente est équivalente à :

$$\forall \mathcal{A} \subset \mathcal{X} \ / \ \mathbb{P}(X \in \mathcal{A} \mid Z = \boldsymbol{z}) \geq \tau : \hat{f}(\mathcal{A}) = \{\hat{f}(\boldsymbol{z})\}.$$

Puisque cette propriété est aussi difficile à vérifier que le problème originel, Parrish *et coll.* proposent de la relâcher en la vérifiant pour un unique ensemble $\mathcal{A}$, bien choisi. L'algorithme de reconnaissance proposé procède donc en trois étapes :

1. estimer la densité conditionnel $p(\boldsymbol{x} \mid \boldsymbol{z})$;
2. construire un ensemble $\mathcal{A}$ vérifiant $\mathbb{P}(X \in \mathcal{A} \mid Z = \boldsymbol{z}) \geq \tau$;
3. prendre une décision suivant la règle (précisée ici pour les problèmes binaires mais existant aussi pour les applications multi-classes) :

$$\hat{f}(\boldsymbol{z}) = \begin{cases} 1 & \text{si } \min\limits_{\boldsymbol{x} \in \mathcal{A}} f(\boldsymbol{x}) > 0 \\ -1 & \text{si } \max\limits_{\boldsymbol{x} \in \mathcal{A}} f(\boldsymbol{x}) \leq 0 \\ \text{aucune décision} & \text{sinon.} \end{cases}$$

Parrish *et coll.* décrivent trois techniques (une indépendante de la distribution $p(\boldsymbol{x} \mid \boldsymbol{z})$ et deux bayésiennes sous hypothèses de gaussiannité ou d'indépendance des composantes de $\boldsymbol{x}$) pour construire un ensemble $\mathcal{A}$ adéquat, en n'utilisant que les deux premiers moments de $p(\boldsymbol{x} \mid \boldsymbol{z})$. Ainsi cette dernière peut être, par exemple, estimée par mixture de gaussiennes. En ce qui concerne la troisième et dernière étape de la reconnaissance, il est nécessaire de résoudre un problème d'optimisation, ce qui (dans le cas des ensembles $\mathcal{A}$ construits selon les méthodes de Parrish *et coll.*) est rendu possible par le caractère linéaire ou quadratique imposé à f (aboutissant à un programme quadratique à contraintes quadratiques (*Quadratically Constrained Quadratic Program*, QCQP)).

La même année, une seconde étude probabiliste s'est portée sur le compromis entre discrimination et précocité de la reconnaissance d'actions humaines acquises par un récepteur Kinect [Ellis *et coll.*, 2013]. C'est une variante élaborée de la régression logistique appliquée au problème d'apprentissage d'instances multiples (*Multiple Instance Learning*, MIL) que nous avons expliqué précédemment. MIL est utilisé pour déterminer l'instance (*i.e.* la pose canonique) la plus discriminante. Étant donné un ensemble de séquences étiquetées $\{(s_i, y_i)\}_{1\leq i\leq n}$, on extrait des instances $(\boldsymbol{x_j})_{1\leq j\leq m}$, représentant des instantanés. Ces dernières sont regroupées en sacs d'indices $(\mathcal{B}_i)_{1\leq i\leq n}$; $(\boldsymbol{x_j})_{j\in\mathcal{B}_i}$ étant le sac d'instances représentant la séquence s_i. Soit alors $\bar{\boldsymbol{x}}_{\boldsymbol{i}}$ une instance représentative du sac $\mathcal{B}_i$ (une pose canonique discriminante). Suivant le principe de régression logistique, la probabilité d'une classe k connaissant le représentant $\bar{\boldsymbol{x}}_{\boldsymbol{i}}$ (du sac $\mathcal{B}_i$) est paramétrée par un ensemble de vecteurs $(\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_K})$:

$$\mathbb{P}(k|\bar{\boldsymbol{x}}_{\boldsymbol{i}}) = \frac{\mathrm{e}^{\langle \boldsymbol{\theta_k} \mid \bar{\boldsymbol{x}}_{\boldsymbol{i}} \rangle_{\ell_2}}}{1 + \sum_{k'=1}^{K} \mathrm{e}^{\langle \boldsymbol{\theta_{k'}} \mid \bar{\boldsymbol{x}}_{\boldsymbol{i}} \rangle_{\ell_2}}}$$

L'algorithme proposé par Ellis *et coll.* consiste à alterner une étape de sélection et de pondération. Étant donnés des vecteurs $(\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_K})$, la première étape doit déterminer, pour chaque séquence s_i, un certain nombre r (défini préalablement) de poses canoniques $(\bar{\boldsymbol{x}}_{\boldsymbol{i}}^{(\boldsymbol{l})})_{1\leq l\leq r}$ représentatives de la séquence s_i tronquée à r instants différents. Ces poses sont choisies pour leur pouvoir discriminant évalué grâce à la formule précédente. Comme nous le verrons ultérieurement, ces multiples troncatures ont pour effet d'augmenter la base d'apprentissage avec des séquences partiellement observées.

La deuxième étape consiste à repondérer le modèle logistique en fonction des nouvelles instances sélectionnées par minimisation de la log-vraisemblance faisant intervenir les différentes troncatures :

$$J(\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_K}) = \sum_{i=1}^{n} \sum_{l=1}^{r} -\gamma_l \log(\mathbb{P}(y_i|\bar{\boldsymbol{x}}_{\boldsymbol{i}}^{(\boldsymbol{l})})) + \lambda_0 \sum_{k=1}^{K} \log(1 + \lambda_1 \|\boldsymbol{\theta_k}\|_{\ell_2}^2),$$

où γ_l ($l \in \mathbb{N}_r$), λ_0 et λ_1 sont des facteurs réels positifs régulant la précocité du modèle et la parcimonie des vecteurs de pondération, introduite par la régularisation lorentzienne. En pratique, cette étape est réalisée par un nombre fixe d'itérations d'une descente de gradient.

2.5.3 Détection

Définition 2.5.2 (Détection précoce)**.**
Soit un ensemble de séquences d'apprentissage étiquetées par $+1$ ou -1. Un problème de détection précoce consiste à assigner à chaque instant, une étiquette qualifiant une séquence dans son ensemble, parmi $\{+1, ?\}$, où $+1$ signale une détection et $?$ indique l'absence de décision. De plus, l'attribution d'une telle étiquette ne peut dépendre des instants futurs à celui observé.

Dans cette définition, la différence avec la classification précoce réside essentiellement dans le nombre de classes. Ici, seulement une seule classe est d'intérêt (celle des événements à détecter).

Parallèlement à la classification d'événements, la détection est apparue pour des applications médicales, telles que le repérage du déclenchement d'une maladie [Neill *et coll.*, 2005] et le traitement de la dépression par reconnaissance d'expressions faciales [Cohn *et coll.*, 2009]. Suivant cette tendance, une technique de détection précoce de lésions sur des images rétinales a été proposée [Ravishankar *et coll.*, 2009]. Si la détection précoce est liée à la reconnaissance d'événements soudains (*i.e.* arrivant de manière inattendue et abrupte) [Suriani

et coll., 2013], étudiée depuis plus plusieurs années, cette thématique a été réellement intégrée à la communauté d'apprentissage automatique avec les travaux présentés dans [Hoai et De la Torre, 2012], suivis de l'approche par *boosting* pour la reconnaissance précoce d'expression faciale proposée par Su et Sato [Su et Sato, 2013].

Les travaux d'Hoai et De la Torre [Hoai et De la Torre, 2012, Hoai et De la Torre, 2014] ont une importance majeure dans le développement de la détection précoce. Ces derniers étendent la technique SVM à sortie structurée, introduite dans [Tsochantaridis *et coll.*, 2005], de manière à décrire la nature séquentielle des séries temporelles et à provoquer une prise de décision précoce. D'un point de vue pratique, Hoai et De la Torre augmentent virtuellement la base d'apprentissage avec des séquences tronquées et contraignent explicitement le score de la fonction de décision à augmenter au fil du temps. Ainsi, la fiabilité de la décision (*i.e.* sa coïncidence avec une décision prise connaissant une séquence entière) croît avec la quantité d'information collectée.

Hoai et De la Torre formulent un problème légèrement différent de celui défini au début de cette section. Leur objectif est double : détecter la présence d'un événement au sein d'une série temporelle et le localiser. En pratique, la base d'apprentissage n'est pas constituée d'événements mais de séries temporelles $\bar{\boldsymbol{s}}_{\boldsymbol{i}}$ contenant chacune un événement $\boldsymbol{s_i}$. Les étiquettes d'un tel jeu de données sont des intervalles indiquant la position des événements dans les séries temporelles d'apprentissage. Nous intéressant principalement à la première propriété, nous considérons un ensemble d'apprentissage usuel $\{(\boldsymbol{s_i}, y_i)\}_{1\leq i\leq n}$ contenant des séquences étiquetées par $y_i = 1$, correspondant aux événements $\boldsymbol{s_i}$ extraits de $\bar{\boldsymbol{s}}_{\boldsymbol{i}}$, et des séquences étiquetées par $y_i = -1$ ne contenant aucune trace de l'événement à détecter. On suppose aussi que toutes les séquences sont de taille T. Étant donné un indice temporelle t de $\mathbb{N}_T$, nous appelons $\boldsymbol{x_i}|_t$ le vecteur caractéristique associé à la série tronquée $\boldsymbol{s_i}|_t$. Avec ce formalisme (différent de celui employé dans [Hoai et De la Torre, 2014]), la méthode proposée par Hoai et De la Torre, nommée détecteur précoce vaste marge (*Maximum Margin Early Detector*, MMED), consiste à résoudre le problème d'optimisation :

$$\begin{array}{ll} \underset{\mathbf{w},\boldsymbol{\xi},b}{\text{minimiser}} & \dfrac{1}{2}\left\|\mathbf{w}\right\|_{\ell_2}^2 + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \begin{cases} y_i\left(\langle \mathbf{w} \mid \boldsymbol{x_i}|_t\rangle_{\ell_2} - b\right) \geq 1 - \dfrac{\xi_i}{g(t)}, & \forall i \in \mathbb{N}_n, \forall t \in \mathbb{N}_T \\ \langle \mathbf{w} \mid \boldsymbol{x_i}|_t - \boldsymbol{x_i}|_{t'}\rangle_{\ell_2} \geq \Delta(t,t') - \dfrac{\xi_i}{g(t)}, & \forall i \in \mathbb{N}_n \,/\, y_i = 1, \\ & \forall t \in \mathbb{N}_T, \forall t' \in [\![1,t]\!] \\ 0 \preccurlyeq \boldsymbol{\xi}, \end{cases} \end{array} \tag{2.3}$$

où $g : \mathbb{N}_T \mapsto]0,1]$ est une fonction croissante (les auteurs proposent un modèle linéaire par morceaux) et la pénalité $\Delta(t,t')$, croissante avec l'écart $t - t'$, est définie par : $\Delta(t,t') = 1 - \frac{2\min(t,t')}{t+t'}$. Deux remarques peuvent être faites en comparaison à un modèle SVM classique :

- ◇ la première contrainte de (2.3) est ressemblante à celle d'une SVM mais possède deux différences. La première réside dans une augmentation virtuelle de l'ensemble d'apprentissage puisque cette contrainte doit être vérifiée pour toutes les séquences tronquées issues des données d'apprentissage. La seconde est une pondération des variables d'écart ξ_i en fonction du temps. Cette pondération insiste sur la nécessite de correctement classer une séquence longuement observée et relativise les erreurs de détection faites pour les séquences très occultées.
- ◇ la seconde contrainte n'existe pas dans un modèle SVM et se présente comme une déclinaison de l'approche initialement proposée dans [Tsochantaridis *et coll.*, 2005]. À l'instar de la première contrainte de (2.3), elle utilise un ensemble d'apprentissage virtuellement augmenté et impose de la sorte la croissance de la fonction de décision au fil de la découverte d'un événement (séquence étiquetée $y_i = 1$).

Bien que MMED soit une approche attirante, il est important de souligner le nombre important de contraintes mises en jeu. En pratique, ceci résulte en un apprentissage coûteux en temps de calcul et en mémoire.

2.6 SYNTHÈSE

La manière la plus courante de traiter une séquence temporelle est de la scinder en signaux de courte durée, puis de construire une règle de décision sur ces signaux avant de les faire voter pour obtenir une réponse globale. Dans ce cadre, nous avons donné un rapide état de l'art concernant les moyens efficaces de représenter ces signaux dans une optique de reconnaissance automatisée. À l'heure actuelle, ces moyens de représentation sont inférés à partir d'observations et de modèles usuels de traitement du signal, tels que les représentations TF (au sens large) dont nous avons dressé les techniques d'apprentissage.

Une autre façon de traiter les séquences temporelles consiste à synthétiser les contributions des signaux issus du découpage avant de prendre une décision globale. Cette approche est plus adaptée que la précédente au traitement séquentiel des données et ouvre la voie à de nouvelles thématiques dont l'une a fait l'objet de la pénultième section de ce chapitre : la reconnaissance précoce. Celle-ci a été explorée à travers les travaux majeurs qui marquent une dizaine d'années de recherches, aussi bien en classification qu'en détection. Celle-ci a été explorée (aussi bien en classification qu'en détection) à travers les travaux majeurs qui marquent une dizaine d'années de recherches.

Au regard de cet état de l'art, notre contribution est double. La première (chapitre 3) se place dans un traitement démocratique des signaux. Elle consiste à fournir une nouvelle approche d'apprentissage de descripteurs. À partir d'un ensemble de signaux d'apprentissage, notre méthode apprend simultanément les filtres d'un BdF et une SVM. De la sorte les représentations TC obtenues par application du BdF sont discriminantes au sens d'un classifieur SVM (linéaire ou gaussien). De plus, dans un soucis d'applicabilité, notre méthode intègre une fonction d'agrégation qu'il est possible d'apprendre comme combinaison de différentes fonctions d'agrégation usuelles.

Notre deuxième contribution (chapitre 4) se place, elle, dans un schéma séquentiel de reconnaissance. Au sein de la thématique de détection précoce, récemment apparue dans la communauté d'apprentissage automatique, nous proposons un nouveau modèle possédant la propriété de fiabilité totale (*i.e.* si un avis de détection est émis en cours d'analyse d'une séquence, il est certain que l'observation complète de cette séquence conduira à la même décision). L'une des particularités de notre approche, comparée aux travaux mentionnés dans ce chapitre, et de ne pas augmenter virtuellement l'ensemble d'apprentissage avec des séquences partiellement observées. Ceci conduit à un problème d'optimisation linéaire de taille raisonnable et rapidement soluble.

Le chapitre suivant est voué à expliquer en détail notre première contribution, traitant de l'apprentissage de descripteurs.

CHAPITRE 3

Apprentissage d'une représentation temps-fréquence convolutive

3.1 INTRODUCTION

L'apparition des réseaux de neurones artificiels dans le domaine des mathématiques appliquées (informatique et traitement du signal) a eu un impact considérable sur notre vision quant aux façons de manipuler les signaux. Premièrement, les réseaux de neurones sont apparus comme une contribution originale dans un monde à la frontière de l'informatique et du traitement du signal, puisqu'ils s'inspirent de modèles biologiques afin de mettre en place des outils numériques de reconnaissance automatique. Force est d'admettre que pour atteindre un nouvel échelon dans la recherche en traitement du signal et en informatique, il fallait manifestement retourner à un système complexe et obscur (mais particulièrement efficace car fondé sur le fonctionnement du cerveau humain). Les outils qui en ressortent sont perçus comme des boîtes noires automatisées mais performantes. Deuxièmement, ces travaux ont fait prendre conscience de l'intérêt de déléguer certains choix à la machine. En pratique, on lui fournit l'ensemble des informations à notre disposition, sans interprétation particulière, puis on laisse le système converger vers un équilibre. En particulier, cette automatisation concerne la création d'un outil d'extraction de caractéristiques des signaux traités. Il est tout à notre avantage d'automatiser l'extraction de caractéristiques visant à classer des signaux, et ce pour différentes raisons qui peuvent être mises en lumière grâce à la traditionnelle comparaison entre les approches reconstructives (ou génératives), qui sont nées avec le traitement du signal pour des besoins d'acquisition, de débruitage et de compression, et les approches discriminantes (apparues plus tard pour des raisons de confort et de diligence) : d'abord, la détermination des caractéristiques de discrimination n'est pas aussi directe que dans une approche générative. Les facteurs de distinctions ne sont pas des éléments que l'on observe (comme les attributs génératifs) mais que l'on déduit, par comparaison de deux phénomènes. Ensuite, il n'y a pas de caractéristiques objectives ; elles dépendent fatalement de la règle de décision appliquée par la suite. Ces raisons expliquent la tendance actuelle à privilégier des approches dirigées par les données.

Cependant, les travaux actuels ne sont pas entièrement satisfaisants. Soit car ils proviennent du milieu informatique et glissent vers le traitement du signal ; soit car, bien que suivant le chemin inverse, ils se concentrent principalement sur des approches atomiques par dictionnaires. De fait, les approches convolutives, chères au traitement du signal, ont été peu étudiées. Dans ce mouvement qui lie le traitement du signal à l'informatique, nous cherchons donc à apporter un regard nouveau sur l'extraction de caractéristiques discrimi-

nantes issues de représentations Temps-Fréquence (TF) convolutives. Plus spécifiquement, nous nous concentrons sur un modèle particulièrement adapté pour modéliser une représentation TF : le Banc de Filtres (BdF).

Ainsi, le but de ce chapitre est de déterminer un algorithme qui, étant donné un ensemble de signaux d'apprentissage $\{(\boldsymbol{s_i}, y_i)\}_{1\leq i\leq n}$, détermine automatiquement et conjointement une représentation convolutive et une règle de décision :

$$\mathfrak{B}\colon \{(\boldsymbol{s_i}, y_i)\}_{1\leq i\leq n} \mapsto \Big((\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\theta\in\mathcal{A}},\quad \boldsymbol{x}\in\mathcal{X}\mapsto f^*(\boldsymbol{x})+b^*\in\mathbb{R}\Big),$$

où $(\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\theta\in\mathcal{A}}$ est une représentation TF implémentée sous la forme d'un banc de $\mathrm{Card}(\mathcal{A})$ filtres et $\boldsymbol{x}\in\mathcal{X}\mapsto f^*(\boldsymbol{x})+b^*$ est une fonction de décision construite sur une machine à vecteurs supports (*Support Vector Machine*, SVM). Pour ce faire, nous formalisons d'abord le problème (section 3.2), à partir des éléments présentés dans les chapitres précédents, puis nous donnons une première approche de résolution (section 3.3). Celle-ci ne fait que peu d'hypothèses sur la représentation TF et peut être simplement mise en œuvre, grâce à des techniques usuelles de relâchement semi-défini ou de descente de gradient. Ensuite, après avoir établi les faiblesses de cette première ligne d'attaque, nous expliquons dans la section 3.4 comment apprendre judicieusement un BdF, en liant le problème de construction automatique à celui d'apprentissage de noyaux multiples (*Multiple Kernel Learning*, MKL). Enfin, cette dernière approche est validée expérimentalement sur trois jeux de données (section 3.5).

3.2 FORMALISATION DU PROBLÈME

Soient $\mathcal{U}$ et $\mathcal{X}$ les espaces respectivement des signaux discrets et des vecteurs caractéristiques de reconnaissance. Notre objectif ici est d'apprendre une représentation TF (sous la forme d'un BdF) discriminante pour un classifieur SVM. Pour ce faire, nous supposons que les données sont transformées par une fonction de la forme $\rho\circ u\colon \mathcal{U}\to\mathcal{X}$, où $\rho\colon \mathcal{O}\to\mathcal{X}$ est un fonction d'agrégation et $u\colon \mathcal{U}\to\mathcal{O}$ une représentation TF ($\mathcal{O}$ est l'ensemble des représentations TF). De fait, on suppose ainsi de manière sous-jacente que les descripteurs pour la classification sont issus du domaine TF.

Nous introduisons à présent deux concepts utiles à la formalisation de notre problème d'apprentissage : l'énergie d'un BdF et l'ensemble des BdF d'énergie finie. La dernière notion est un abus de langage pour désigner les BdF d'énergie au plus égale à 1. Ce choix est justifié par le fait que tout BdF d'énergie finie peut être mis à l'échelle pour obtenir une énergie unitaire.

Définition 3.2.1 (Énergie)**.**
Soit $u=(\boldsymbol{h_l}, N_l)_{1\leq l\leq d}$ un BdF. On appelle énergie de u et on note $\mathrm{En}(u)$ la quantité

$$\mathrm{En}(u)=\sum_{l=1}^{d}\|\boldsymbol{h_l}\|_{\ell_2}^2\,.$$

Définition 3.2.2 (BdF d'énergie finie)**.**
Soit T la taille maximale des signaux à traiter. L'ensemble des BdF d'énergie finie est

$$\mathcal{T}=\big\{u=(\boldsymbol{h_l}, N_l)_{1\leq l\leq d},\, d\in\mathbb{N},\, \forall l\in\mathbb{N}_d\colon N_l\in\mathbb{N}_T,\, \boldsymbol{h_l}\in\mathbb{K}^T,\, \mathrm{En}(u)\leq 1\big\}\,.$$

Étant donnés un ensemble de signaux d'apprentissage $\{(\boldsymbol{s_i}, y_i)\}_{1\leq i\leq n}$, un noyau $k\colon \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ et un paramètre de coût positif C, construire une représentation TF discriminante dans un environnement à noyau peut être formellement défini par :

$$\underset{u\in\mathcal{T}}{\text{minimiser}}\quad J(u,k), \tag{3.1}$$

où $\mathcal{T}$ est l'ensemble des BdF d'énergie finie (définition 3.2.2) et J est un critère d'apprentissage de noyaux (par exemple la borne rayon-marge, l'alignement de noyaux ou le risque régularisé SVM). Afin de tirer partie des avancées sur les SVM, concernant aussi bien la complexité temporelle de résolution [Platt, 1999, Chang et Lin, 2011] que les développements en apprentissage de noyaux [Lanckriet *et coll.*, 2002, Bach *et coll.*, 2004], nous choisissons le dernier de ces critères : le risque régularisé. Ainsi, à l'instar de [Rakotomamonjy *et coll.*, 2008, Flamary *et coll.*, 2012], nous nous concentrons sur l'indicateur de discrimination :

$$J(u,k) = \left\{ \begin{array}{ll} \min\limits_{f\in\mathcal{H},\, b\in\mathbb{R},\, \boldsymbol{\xi}\in\mathbb{R}^n} & \frac{1}{2}\left\|f\right\|_{\mathcal{H}}^2 + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(f((\rho\circ u)(\boldsymbol{s_i})) + b\right) \geq 1-\xi_i,\ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0. \end{array} \right. \end{array} \right. \tag{3.2}$$

Le problème d'optimisation (3.2) est le problème SVM standard à u (BdF) et k (noyau) fixés. $\mathcal{H}$ est l'espace de Hilbert à noyau reproduisant (*Reproducing Kernel Hilbert Space*, RKHS) associé à k [Aronszajn, 1950], $\boldsymbol{\xi}$ est le vecteur des variables d'écart et un couple (f, b) solution de (3.2) donne la fonction de décision finale par la formule :

$$\boldsymbol{s} \in \mathcal{U} \mapsto \text{Signe}\left(f((\rho\circ u)(\boldsymbol{s})) + b\right),$$

avec f définie à partir du vecteur $\boldsymbol{\alpha}$ (appris lors de l'optimisation) à composantes positives :

$$\forall \boldsymbol{s} \in \mathcal{U},\ f((\rho\circ u)(\boldsymbol{s})) = \sum_{\substack{i=1 \\ \alpha_i>0}}^{n} \alpha_i y_i k((\rho\circ u)(\boldsymbol{s_i}), (\rho\circ u)(\boldsymbol{s})). \tag{3.3}$$

Remarque 11.

Avec un critère construit directement sur le problème d'optimisation SVM (en opposition à la borne rayon-marge et à l'alignement de noyaux), (3.4) aurait pu être formulé différemment :

$$\begin{array}{ll} \underset{f\in\mathcal{H},\, b\in\mathbb{R},\, \boldsymbol{\xi}\in\mathbb{R}^N,\, u\in\mathcal{T}}{\text{minimiser}} & \frac{1}{2}\left\|f\right\|_{\mathcal{H}}^2 + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(f((\rho\circ u)(\boldsymbol{s_i})) + b\right) \geq 1-\xi_i, \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0. \end{array} \right. \end{array} \tag{3.4}$$

Les deux problèmes d'optimisation (3.1) et (3.4) sont *difficiles* dans le sens où ils sont non-convexes. En effet, puisque les données filtrées sont consécutivement agrégées et transformées par un noyau, chaque non-convexité dans la fonction d'agrégation ρ et dans la fonction de redescription ϕ induite par le noyau k résulte en une non-convexité de la fonction objectif par rapport aux Réponses Impulsionnelles (RI) du BdF [Varma et Babu, 2009, Flamary *et coll.*, 2012]. Cet effet peut encore être amplifié si les RI sont paramétrées de manière non-convexe (par exemple par les pulsations de coupures dans le cas d'un filtre passe-bande). En conséquence, la stratégie de résolution de notre problème d'apprentissage est un point crucial dans la mise au point d'un algorithme aussi efficace que possible, aussi bien du point de vue de la capacité de généralisation que de la complexité temporelle. La raison pour laquelle nous avons opté pour l'approche par encapsulage (3.1) par rapport à celle *a priori* plus naturelle (3.4) est l'aptitude de la première à gérer facilement un espace de redescription de dimension infinie (comme par exemple celui induit par le noyau gaussien). De plus, la stratégie par encapsulage nous permet de bénéficier du caractère convexe des SVM ainsi que des avancées récentes concernant les logiciels de résolution (par exemple [Chang et Lin, 2011]) en termes de précision et de temps d'apprentissage. Notons enfin que la proposition 3.2.1 nous assure que les problèmes (3.1) et (3.4) sont équivalents au sens de la définition donnée dans la section 1.2. En conséquence, préférer (3.1) à (3.4) revient à choisir une stratégie de résolution d'un problème non-convexe.

Proposition 3.2.1 (Équivalence de l'encapsulage)**.**
Soient deux ensembles $\mathcal{A}$ et $\mathcal{B}$ et une fonction de coût $J\colon \mathcal{A}\times\mathcal{B}\to\mathbb{R}$. Les problèmes d'optimisation :

$$\underset{a\in\mathcal{A},\, b\in\mathcal{B}}{\text{minimiser}} \quad J(a,b),$$

et

$$\begin{array}{ll}\underset{a\in\mathcal{A},b\in\mathcal{B}}{\text{minimiser}} & J(a,b)\\ \text{tel que} & b\in\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a,\bar{b}),\end{array}$$

sont équivalents.

Démonstration. Commençons par le premier sens de l'équivalence : soit $(a^*,b^*)\in\mathcal{A}\times\mathcal{B}$ vérifiant :

$$\forall(a,b)\in\mathcal{A}\times\mathcal{B}\colon J(a,b)\geq J(a^*,b^*). \tag{3.5}$$

On cherche alors à montrer que $b^*\in\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a^*,\bar{b})$ et que :

$$\forall(a,b)\in\mathcal{A}\times\mathcal{B}\ /\ b\in\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a,\bar{b})\colon J(a,b)\geq J(a^*,b^*). \tag{3.6}$$

La propriété (3.6) est immédiatement déduite de (3.5) puisque $\forall a\in\mathcal{A}$, $\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a,\bar{b})$ est une partie de $\mathcal{B}$. De plus, il vient comme cas particulier de (3.5) que $\forall\bar{b}\in\mathcal{B}\colon J(a^*,\bar{b})\geq J(a^*,b^*)$. Ainsi on a bien $b^*\in\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a^*,\bar{b})$. En conséquence, (a^*,b^*) est une solution globale du deuxième problème d'optimisation.

Réciproquement, soit $(a^*,b^*)\in\mathcal{A}\times\mathcal{B}$ vérifiant $b^*\in\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a^*,\bar{b})$ et (3.6). On cherche à démontrer (3.5). Par définition, $\forall(a,b)\in\mathcal{A}\times\mathcal{B}\colon J(a,b)\geq J(a,b_a)$, où $b_a\in\underset{\bar{b}\in\mathcal{B}}{\arg\min}\, J(a,\bar{b})$. Or par hypothèse, $J(a,b_a)\geq J(a^*,b^*)$, donc $J(a,b)\geq J(a^*,b^*)$. Ainsi (a^*,b^*) est une solution globale du premier problème d'optimisation. ■

Avant de décrire des algorithmes de résolution du problème (3.1), il est intéressant de vérifier que celui-ci est soluble, au moins dans un cas simple. Ainsi, après avoir donné la définition de la distance canonique entre BdF, nous mentionnons une proposition qui nous assure de l'existence d'une solution dans le cas où $J(\cdot,k)$ est continue (ce qui est souvent vérifié) et où le nombre de filtres du banc est borné.

Définition 3.2.3 (Distance entre BdF)**.**
On définit la distance entre deux BdF $u=(\boldsymbol{h_l},N_l)_{1\leq l\leq d}$ et $u'=(\boldsymbol{h'_l},N'_l)_{1\leq l\leq d'}$ par :

$$\mathrm{Dist}(u,u')=\begin{cases}+\infty \text{ si } d\neq d';\\ +\infty \text{ si } d=d' \text{ et } \exists l\in\mathbb{N}_d\ /\ N_l\neq N'_l;\\ \sqrt{\sum_{l=1}^{d}\left\|\boldsymbol{h_l}-\boldsymbol{h'_l}\right\|_{\ell_2}^2} \text{ sinon.}\end{cases}$$

Proposition 3.2.2 (Résolubilité de (3.1))**.**
Pour tout noyau $k\colon\mathcal{X}\times\mathcal{X}\to\mathbb{R}$, si $J(\cdot,k)$ est continu au sens de la distance 3.2.3 et si le nombre de filtres des BdF de $\mathcal{T}$ est borné, alors (3.1) *admet une solution,* i.e.

$$\exists u\in\mathcal{T}\ /\ \forall u'\in\mathcal{T}\colon J(u',k)\geq J(u,k).$$

Démonstration. Dans un premier temps, on considère le sous-problème consistant à résoudre (3.1) lorsque les facteurs de décimations sont figés. Dans ce cas, $\mathcal{T}$ est réduit à :

$$\mathcal{T}=\left\{(\boldsymbol{h_l},N_l)_{1\leq l\leq d},\ \forall l\in\mathbb{N}_d\colon \boldsymbol{h_l}\in\mathbb{K}^T,\ \sum_{l=1}^{d}\|\boldsymbol{h_l}\|_{\ell_2}^2\leq 1\right\},$$

pour des valeurs de d, N_1, …, N_l fixées préalablement. Autrement dit, $\mathcal{T}$ est isomorphe à la boule unité de la norme ℓ_2 d'un espace vectoriel de dimension finie (dT), donc $\mathcal{T}$ est compacte. Par continuité, le problème admet donc un minimum et ce dernier est atteint.

Puisque le nombre de facteurs de décimation possibles est fini (il est compris entre 1 et la taille des signaux d'entrée), résoudre (3.1) revient à considérer le minimum d'un nombre combinatoire mais fini de sous-problèmes de (3.1) à facteurs de décimation fixés. Ce minimum existe et est atteint. ■

Pour illustrer cette proposition, nous abordons la propriété plus forte de la différentiabilité. Plaçons-nous dans le cas où le minimum SVM à BdF u et à noyau k fixés est unique. Ceci est assuré si les données transformées ne comportent pas de doublons et si k est défini positif (c'est le cas du noyau gaussien). La formulation duale (1.2) du problème SVM (de concert au théorème 1.4.2 page 22) nous affirme que la dérivabilité de la fonction objectif (duale) est assurée par celle du noyau par rapport à ces paramètres. Ainsi, si pour tout $(\boldsymbol{s}, \boldsymbol{h})$ de $\mathcal{U} \times \mathcal{U}$, l'application $u \in \mathcal{T} \mapsto k\left((\rho \circ u)(\boldsymbol{s}), (\rho \circ u)(\boldsymbol{h})\right)$ est dérivable, alors $J(\cdot, k)$ est dérivable et *a fortiori* continue. Pour tous les noyaux usuels, $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est dérivable donc la condition d'existence d'une solution revient majoritairement à la fonction d'agrégation $\rho\colon \mathcal{O} \to \mathcal{X}$. Pour les fonctions d'agrégation telles que les normes et moyennes locales, ou encore la diffusion (tableau 2.1 page 36), la dérivabilité est assurée. Le problème se pose pour une fonction d'agrégation comme le maximum local, qui ne répond pas au schéma précédent.

Dans la prochaine section, nous détaillons une première façon de traiter les problèmes (3.1) et (3.4), sans contraintes supplémentaires.

3.3 APPROCHE DIRECTE

3.3.1 Cas d'école

Cette section revient sur le problème d'apprentissage d'un BdF discriminant (3.4) en se plaçant dans un cas simple :

- ◇ le noyau $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ est linéaire : $k(\boldsymbol{x}, \boldsymbol{z}) = \langle \boldsymbol{x} \mid \boldsymbol{z} \rangle_{\ell_2}$;
- ◇ la fonction d'agrégation est transparente, *i.e.* $\rho\colon a \in \mathcal{O} \mapsto \mathrm{Vec}(a) \in \mathcal{X}$;
- ◇ le nombre d de filtres constituants les bancs, les facteurs de décimations $(N_l)_{1 \leq l \leq d}$ et les tailles des filtres $(q_l)_{1 \leq l \leq d}$ sont fixés. Autrement dit, l'ensemble des BdF $\mathcal{T}$ est réduit à :

$$\overline{\mathcal{T}} = \left\{u = (\boldsymbol{h_l}, N_l)_{1 \leq l \leq d},\, \forall l \in \mathbb{N}_d\colon \boldsymbol{h_l} \in \mathbb{K}^{q_l}, \mathrm{En}(u) \leq 1\right\}.$$

Dans ce cas d'école, l'apprentissage d'un BdF se résume à déterminer les RI $(\boldsymbol{h_l})_{1 \leq l \leq d}$ possédant une énergie au plus unitaire :

$$\sum_{l=1}^{d} \|\boldsymbol{h_l}\|_{\ell_2}^2 \leq 1.$$

Formellement, on cherche à résoudre :

$$\begin{array}{ll} \underset{\substack{\mathbf{w} \in \mathcal{X},\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n, \\ u = (\boldsymbol{h_l}, N_l)_{1 \leq l \leq d}}}{\text{minimiser}} & \dfrac{1}{2}\|\mathbf{w}\|_{\ell_2}^2 + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(\langle \mathbf{w} \mid \mathrm{Vec}\left(u(\boldsymbol{s_i})\right)\rangle_{\ell_2} + b\right) \geq 1 - \xi_i, \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \sum_{l=1}^{d} \|\boldsymbol{h_l}\|_{\ell_2}^2 \leq 1. \end{array} \right. \end{array} \tag{3.7}$$

Rappelons que $(\mathbf{w}, b)$ définit une séparatrice linéaire dans le plan TF et que la contrainte sur l'énergie a pour but de contrôler la complexité du modèle (*i.e.* éviter un effet de sur-apprentissage par accroissement de la norme du noyau, conformément à la remarque en section 1.4.2).

Montrons maintenant que sous la forme (3.7), l'apprentissage de BdF discriminant est un programme quadratique à contraintes quadratiques (*Quadratically Constrained Quadratic Program*, QCQP) non nécessairement convexe. Par définition une opération de filtrage puis de décimation par $(\boldsymbol{h}, N)$ s'exprime par :

$$\forall \boldsymbol{s} \in \mathcal{U}\colon \downarrow N\left[\boldsymbol{h} \star \boldsymbol{s}\right] = \left(\sum_{j=1}^{q} h_j \tilde{s}_{2+N(k-1)-j}\right)_{1 \leq k \leq \left\lfloor \frac{m}{N} \right\rfloor},$$

où q est la taille du filtre et $\tilde{\boldsymbol{s}}$ est le signal prolongé dans le passé par périodisation ou symétrie. Si l'on écrit $\mathbf{w}$ en fonction des contributions correspondant à chaque filtre du BdF : $\mathbf{w} = \mathrm{Vec}(\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(d)})$, on obtient :

$$\forall \boldsymbol{s} \in \mathcal{U}\colon \left\langle \mathbf{w} \mid \mathrm{Vec}(u(\boldsymbol{s})) \right\rangle_{\ell_2} = \sum_{l=1}^{d} \left\langle \mathbf{w}^{(l)} \mid \downarrow N_l\left[\boldsymbol{h_l} \star \boldsymbol{s}\right] \right\rangle_{\ell_2} = \sum_{l=1}^{d} \sum_{k=1}^{\left\lfloor \frac{m}{N_l} \right\rfloor} \sum_{j=1}^{q_l} \mathbf{w}_k^{(l)} h_{lj} \tilde{s}_{2+N(k-1)-j}.$$

Ainsi, si l'on regroupe le vecteur normal $\mathbf{w}$ avec les RI sous un seul et même vecteur $\boldsymbol{\gamma} = \mathrm{Vec}(\mathbf{w}, \boldsymbol{h_1}, \ldots, \boldsymbol{h_d})$ de taille Q, il est possible de construire une matrice $\boldsymbol{A}$ uniquement à partir de $\tilde{\boldsymbol{s}}$ telle que :

$$\left\langle \mathbf{w} \mid \mathrm{Vec}(u(\boldsymbol{s})) \right\rangle_{\ell_2} = \sum_{\substack{1 \leq j \leq Q \\ 1 \leq k \leq Q}} A_{j,k} \gamma_j \gamma_k = \boldsymbol{\gamma}^T \boldsymbol{A} \boldsymbol{\gamma}.$$

On en déduit qu'en définissant correctement des matrices $\boldsymbol{A_0}$, $\boldsymbol{A_1}$, …, $\boldsymbol{A_{n+1}}$, le problème (3.7) peut être récrit :

$$\begin{array}{ll} \underset{\boldsymbol{\gamma} \in \mathbb{R}^Q,\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n}{\text{minimiser}} & \boldsymbol{\gamma}^T \boldsymbol{A_0} \boldsymbol{\gamma} + C \mathbb{1}^T \boldsymbol{\xi} \\ \text{tel que} & \begin{cases} y_i \left(\boldsymbol{\gamma}^T \boldsymbol{A_i} \boldsymbol{\gamma} + b \right) \geq 1 - \xi_i, \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \boldsymbol{\gamma}^T \boldsymbol{A_{n+1}} \boldsymbol{\gamma} \leq 1. \end{cases} \end{array}$$

On reconnaît clairement un QCQP pour lequel les matrices $\boldsymbol{A_i}$ ($i \in \mathbb{N}_n$) ne sont pas semi-définies positives. L'ensemble des contraintes est donc quadratique mais non-convexe. Une manière astucieuse de résoudre un QCQP non-convexe est de transiter par un programme semi-défini (*Semi-Definite Program*, SDP) [Luo *et coll.*, 2010]. Un SDP est un programme linéaire à contraintes linéaires sur le cône des matrices semi-définies positives $\mathbb{S}_+^Q$ (ici, de taille Q). Pour aboutir à un tel problème d'optimisation, remarquons tout d'abord que pour toute matrice $\boldsymbol{A}$, on a l'égalité $\boldsymbol{\gamma}^T \boldsymbol{A} \boldsymbol{\gamma} = \mathrm{Tr}(\boldsymbol{A} \boldsymbol{\gamma} \boldsymbol{\gamma}^T)$. Ainsi, en effectuant le changement de variables $\boldsymbol{X}_+ = \boldsymbol{\gamma} \boldsymbol{\gamma}^T$, le problème précédent est équivalent à :

$$\begin{array}{ll} \underset{\boldsymbol{X}_+ \in \mathbb{S}_+^Q,\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n}{\text{minimiser}} & \mathrm{Tr}(\boldsymbol{A_0} \boldsymbol{X}_+) + C \mathbb{1}^T \boldsymbol{\xi} \\ \text{tel que} & \begin{cases} y_i \left(\mathrm{Tr}(\boldsymbol{A_i} \boldsymbol{X}_+) + b \right) \geq 1 - \xi_i, \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \mathrm{Tr}(\boldsymbol{A_{n+1}} \boldsymbol{X}_+) \leq 1 \\ \mathrm{Rang}(\boldsymbol{X}_+) = 1. \end{cases} \end{array} \tag{3.8}$$

Bien que les deux derniers programmes d'optimisation sont équivalents, le problème (3.8), écrit sur le cône des matrices semi-définies positives, possède une spécificité : toute la

non-convexité du problème est reléguée dans la contrainte de rang. En supprimant cette contrainte, on obtient le relâchement semi-défini de (3.7) [Luo *et coll.*, 2010] :

$$\begin{array}{ll} \underset{\boldsymbol{X}_+ \in \mathbb{S}_+^Q,\, b \in \mathbb{R},\, \boldsymbol{\xi} \in \mathbb{R}^n}{\text{minimiser}} & \mathrm{Tr}(\boldsymbol{A_0} \boldsymbol{X}_+) + C \mathbb{1}^T \boldsymbol{\xi} \\ \text{tel que} & \begin{cases} y_i \left(\mathrm{Tr}(\boldsymbol{A_i} \boldsymbol{X}_+) + b \right) \geq 1 - \xi_i, \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \mathrm{Tr}(\boldsymbol{A_{n+1}} \boldsymbol{X}_+) \leq 1 \end{cases} \end{array} \tag{3.9}$$

L'expression *relâchement semi-défini positif* provient de la suppression de la contrainte de rang, qui a pour effet de *libérer* le problème suivant deux directions :

- ◇ le problème (3.9) est convexe tandis que (3.8) ne l'est pas. En ce sens, il est plus libre car plus aisé à résoudre ;
- ◇ le minimum de (3.9) minore celui de (3.8). En effet, soient $(\mathbf{w}^*, \boldsymbol{\xi}^*)$, $(\boldsymbol{\gamma}^\dagger, \boldsymbol{\xi}^\dagger)$ et $(\boldsymbol{X}_+{}^\ddagger, \boldsymbol{\xi}^\ddagger)$ des solutions respectives de (3.7), (3.8) et (3.9). En posant $\boldsymbol{X}_+{}^\dagger = \boldsymbol{\gamma}^\dagger \boldsymbol{\gamma}^{\dagger T}$, on a la relation :

$$\mathrm{Tr}(\boldsymbol{A_0} \boldsymbol{X}_+{}^\ddagger) + C \mathbb{1}^T \boldsymbol{\xi}^\ddagger \leq \mathrm{Tr}(\boldsymbol{A_0} \boldsymbol{X}_+{}^\dagger) + C \mathbb{1}^T \boldsymbol{\xi}^\dagger = \frac{1}{2} \|\mathbf{w}^*\|_{\ell_2}^2 + C \mathbb{1}^T \boldsymbol{\xi}^*.$$

Si l'on sait que l'objectif optimal de (3.9) minore celui de (3.8), il est difficile de quantifier la différence. En pratique, on peut se passer d'une telle quantification et vérifier expérimentalement la pertinence de l'approche. Notons que la résolution du SDP (3.9) est envisageable car, bien que de grande taille, les matrices $\boldsymbol{A_i}$ ($i \in [\![0, n+1]\!]$) sont creuses.

Ayant résolu le problème (3.9) (par exemple grâce à un logiciel tel que SeDuMi [Sturm, 1999]), la connaissance d'une matrice solution $\boldsymbol{X}_+$ ne nous indique pas de points faisables $\boldsymbol{\gamma}$ pour (3.8), encore moins une solution globale. Il existe alors deux grandes techniques pour approcher une solution de (3.8) [Luo *et coll.*, 2010] :

- ◇ réaliser une approximation de rang 1 de $\boldsymbol{X}_+$: $\boldsymbol{X}_+ \approx \boldsymbol{\gamma}\boldsymbol{\gamma}^T$. Ceci est, par exemple, possible par une décomposition en valeurs propres ($\boldsymbol{\gamma} = \sqrt{\lambda_{\max}} \boldsymbol{v}_{\max}$, où $\lambda_{\max}$ est la plus grande valeur propre de $\boldsymbol{X}_+$ est $\boldsymbol{v}_{\max}$ est le vecteur propre associé à $\lambda_{\max}$) et correspond à une projection sur le cône des matrices semi-définies positives. Pour obtenir un point faisable de (3.8), il est alors nécessaire d'extraire de $\boldsymbol{\gamma}$ les RI et de calculer les paramètres de la séparatrice correspondants en résolvant une SVM ;
- ◇ tirer aléatoirement des réalisations $\boldsymbol{\gamma}$ suivant $\mathcal{N}(0, \boldsymbol{X}_+)$, extraire les RI et calculer les paramètres de la séparatrice correspondants. L'observation à retenir est celle minimisant la valeur de la fonction objectif de (3.7). L'interprétation sous-jacente de cette heuristique est qu'une variable aléatoire Γ admettant comme loi $\mathcal{N}(0, \boldsymbol{X}_+)$ résout le problème (3.8) *en moyenne*. Pour s'en convaincre, il suffit de remarquer que pour n'importe quelles matrices $\boldsymbol{A}$ et $\boldsymbol{B}$, le problème d'optimisation :

$$\begin{array}{ll} \underset{\substack{\boldsymbol{X}_+ \in \mathbb{S}_+^Q \\ \Gamma \sim \mathcal{N}(0, \boldsymbol{X}_+)}}{\text{minimiser}} & \mathbb{E}[\Gamma^T \boldsymbol{A} \Gamma] \\ \text{tel que} & \mathbb{E}[\Gamma^T \boldsymbol{B} \Gamma] \geq 0, \end{array}$$

se réduit à :

$$\begin{array}{ll} \underset{\boldsymbol{X}_+ \in \mathbb{S}_+^Q}{\text{minimiser}} & \mathrm{Tr}\left(\boldsymbol{A} \boldsymbol{X}_+\right) \\ \text{tel que} & \mathrm{Tr}\left(\boldsymbol{B} \boldsymbol{X}_+\right) \geq 0, \end{array}$$

puisque par définition $\mathbb{E}[\Gamma\Gamma^T] = \boldsymbol{X}_+$.

3.3.2 Cas général

Cette section aborde un cadre plus général que celui précédemment exposé en revenant sur les hypothèses faites. Nous considérons ici :

- ◇ un noyau $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ dérivable (non-nécessairement linéaire) ;
- ◇ une fonction d'agrégation $\rho\colon \mathcal{O} \mapsto \mathcal{X}$ dérivable ;
- ◇ un nombre d de filtres constituants les bancs, des facteurs de décimations $(N_l)_{1\leq l\leq d}$ et des tailles de filtres $(q_l)_{1\leq l\leq d}$ toujours fixés.

Ces hypothèses sont clairement posées de manière à mettre en place une approche par descente de gradient. Pour des questions d'efficacité et de simplicité, nous choisissons la formulation d'apprentissage de BdF (3.1), dite par encapsulage. On peut alors reformuler le problème d'optimisation de la manière suivante :

$$\begin{array}{ll} \underset{u=(\boldsymbol{h_l},N_l)_{1\leq l\leq d}}{\text{minimiser}} & J(u,k) \\ \text{tel que} & \sum_{l=1}^{d} \|\boldsymbol{h_l}\|_{\ell_2}^2 \leq 1. \end{array} \tag{3.10}$$

Sous cette forme, l'apprentissage de BdF est très proche du filtrage vaste marge proposé dans [Flamary *et coll.*, 2012]. En effet, dans le cas d'un signal unidimensionnel, l'approche de Flamary *et coll.* est exprimée en fonction d'un paramètre de régularisation λ par :

$$\underset{u=(\boldsymbol{h_1},1)}{\text{minimiser}} \quad J(u,k) + \lambda \|\boldsymbol{h_1}\|_{\ell_2}^2 .$$

La différence avec notre approche réside dans :

- ◇ le nombre de filtres considérés (quelconque dans notre cas et un unique dans [Flamary *et coll.*, 2012]) ;
- ◇ la régularisation en norme ℓ_2 d'Ivanov (contrainte explicite), qui prend la forme d'une régularisation de Tikhonov dans [Flamary *et coll.*, 2012], pondérée par un paramètre additionnel λ.

À l'instar des travaux exposés dans [Rakotomamonjy *et coll.*, 2008, Flamary *et coll.*, 2012], il est donc envisageable de mettre en place une descente de gradient afin d'approcher un minimum local du problème non-convexe (3.10). Le théorème 1.4.2 nous assure que si $J(u,k)$ est atteint pour un unique vecteur dual $\boldsymbol{\alpha}$ (solution du problème (1.2)), alors $J(\cdot,k)$ est différentiable et pour le p^{e} coefficient de la RI $\boldsymbol{h_l}$, nous avons :

$$\frac{\partial J}{\partial h_{l_p}}(u,k) = -\frac{1}{2}\boldsymbol{\alpha}^T \frac{\partial \boldsymbol{K}_+}{\partial h_{l_p}} \boldsymbol{\alpha},$$

où $\frac{\partial \boldsymbol{K}_+}{\partial h_{l_p}}$ représente la dérivée partielle du noyau par rapport aux coefficients des RI et peut être formellement défini grâce à la fonction $\bar{k}_{i,j}\colon u \in \overline{\mathcal{T}} \mapsto k\left((\rho\circ u)(\boldsymbol{s_i}),(\rho\circ u)(\boldsymbol{s_j})\right)$ par $\frac{\partial \boldsymbol{K}_+}{\partial h_{l_p}} = \left(\frac{\partial \bar{k}_{i,j}}{\partial h_{l_p}}(u)\right)_{1\leq i,j\leq n}$. Connaissant les dérivées partielles de $J(\cdot,k)$, il est possible d'approcher un minimum local de (3.1) par une descente de gradient projeté avec retour sur trace (*backtracking*) [Boyd et Vandenberghe, 2004]. Cette dernière nous assure une décroissance suffisante de la fonction objectif à chaque itération. Cette approche est décrite par l'algorithme 1. Pour simplifier la description de l'algorithme, nous nous accordons un abus d'écriture en assimilant temporairement le BdF u au vecteur des RI : $u = \text{Vec}\left((\boldsymbol{h_1},\ldots,\boldsymbol{h_l})\right)$. Dans l'algorithme 1, l'opérateur Proj est une simple projection sur la boule unité de la norme ℓ_2, permettant d'assurer que l'énergie du BdF est au plus unitaire :

$$\text{Proj}(u) = \begin{cases} u & \text{si } \|u\|_{\ell_2} \leq 1 \\ \dfrac{u}{\|u\|_{\ell_2}} & \text{sinon.} \end{cases}$$

Données : ensemble de signaux d'apprentissage $\{(\boldsymbol{s_i}, y_i)\}_{1 \leq i \leq n}$.
retourner *BdF* u *et classifieur* f.
Initialiser aléatoirement les RI $(\boldsymbol{h_1}, \ldots, \boldsymbol{h_l})$;
$u \leftarrow \mathrm{Vec}\left((\boldsymbol{h_1}, \ldots, \boldsymbol{h_l})\right)$;
$\eta \leftarrow 1$;
tant que *équilibre non-atteint* **faire**
  Résoudre (1.2) pour obtenir $\boldsymbol{\alpha}$ et $J(u, k)$;
  Calculer $\nabla_u J(u, k)$ à partir de $\boldsymbol{\alpha}$;
  si $\|\nabla_u J(u, k)\|_{\ell_2} \approx 0$ **alors**
    déduire f de $\boldsymbol{\alpha}$;
    équilibre atteint ;
  sinon
    $\bar{u} \leftarrow \mathrm{Proj}\left(u - \eta \nabla_u J(u, k)\right)$;
    tant que $J(\bar{u}, k) \geq J(u, k) + \epsilon \left\langle \nabla_u J(u, k) \mid \bar{u} - u \right\rangle_{\ell_2}$ **faire**
      $\eta \leftarrow \frac{\eta}{2}$;
      $\bar{u} \leftarrow \mathrm{Proj}\left(u - \eta \nabla_u J(u, k)\right)$;
    $u \leftarrow \bar{u}$;
    $\eta \leftarrow 1$;

Algorithme 1 : Apprentissage d'un BdF discriminant par descente de gradient avec retour sur trace.

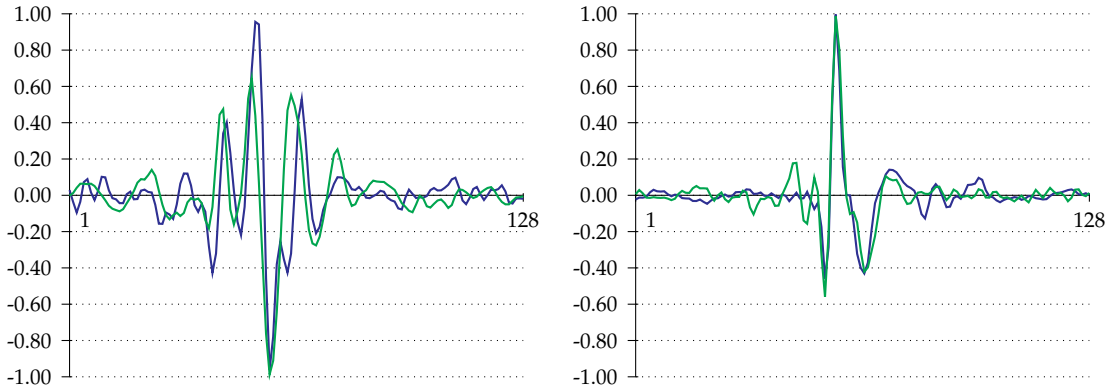
3.3.3 Comparaison numérique

Nous comparons ici les deux approches précédemment présentées, construites sur un relâchement semi-défini et sur une descente de gradient. Nous nous plaçons donc dans le cadre des hypothèses les plus fortes, à savoir un noyau k linéaire est une fonction d'agrégation ρ transparente.

Les résultats de deux expériences numériques sont rapportés : un problème de classification binaire synthétique, dans lequel chaque classe de signaux est composée de l'une des formes *Blocks* ou *HeaviSine* de la boîte à outils Matlab Wavelab [Buckheit et Donoho, 1995], auquel s'ajoute un bruit coloré stationnaire à l'échelle d'un signal mais non-stationnaire à celle de l'ensemble d'apprentissage. Pour chaque signal, le bruit est synthétisé comme un bruit blanc gaussien convolué avec un filtre choisi au hasard parmi les RI $[1, -2, 1]$, $[0, 1, -1]$ et $[1, 0, 1]$.

La deuxième expérience est construite à partir des signaux de la compétition de classification d'enregistrements sonores cardiaques (*Classifying Heart Sounds Challenge*, CHSC), dont des exemples sont donnés sur la figure 3.1 page suivante [Bentley *et coll.*, 2011]. La première classe est constituée des bruits B1 (sons émis à la fermeture de la valve mitrale) tandis que la seconde classe comprend les bruits B2 (fermeture de la valve aortique). Afin d'évaluer la robustesse des approches, un bruit identique à celui de l'expérience synthétique est ajouté aux données. La différence entre les deux expériences réside dans les formes génératives des classes : elles sont constantes pour le jeu de données synthétique et réelles pour la base CHSC.

Dans les deux situations, nous cherchons à apprendre un banc de trois filtres, chacun ayant un facteur de décimation de 2. Les filtres sont de tailles différentes ($(32, 64, 128)$ pour le jeu de données jouet et $(16, 32, 64)$ pour les enregistrements CHSC) de sorte à privilégier une approche multi-résolution redondante. De plus, nous rapportons les résultats pour des filtres à phase linéaire. Ceci est cohérent par rapport aux données et permet de diminuer par deux le nombre de variables à optimiser.



(a) Bruit B1. (b) Bruit B2.

FIGURE 3.1 – Exemples de données CHSC sans bruit ajouté (deux signaux différents pour chaque classe, alignés grâce à leur maximum d'énergie).

Cinq approches différentes sont comparées :

- ◇ gradient projeté : résolution de (3.1) par gradient projeté avec retour sur trace et initialisation aléatoire ;
- ◇ relâchement semi-défini : résolution de l'approximation convexe (3.9) grâce à SeDuMi [Sturm, 1999] puis tirage aléatoire pour trouver un point faisable de (3.7) ;
- ◇ gradient projeté initialisé par relâchement semi-défini : résolution de (3.1) par descente de gradient initialisée par le résultat de l'approche précédente ;
- ◇ domaine temporel : classification SVM linéaire en prenant les représentations temporelles des signaux comme vecteurs caractéristiques ;
- ◇ domaine de Fourier : classification SVM linéaire en prenant le module des représentations de Fourier des signaux comme descripteurs.

Pour toutes les méthodes, le paramètre de coût C est déterminé par validation croisée. Les résultats sont rapportés sur les figures 3.2 page suivante et 3.3 page 72, sous la forme de taux de classification et de temps d'apprentissage moyens, accompagnés de l'erreur type.

Les résultats sur le jeu de données simulées (figure 3.2) montrent d'emblée l'intérêt d'une décomposition TF dirigée par les données ainsi que l'apport du relâchement semi-défini en terme de classification et de complexité d'apprentissage. Les résultats sur le jeu de données réelles (figure 3.3) sont en revanche moins probants car le bruit s'ajoute à une forte variabilité intra-classe. Si cette fois, descente de gradient et relâchement semi-défini semblent conduire à des performances de classification comparables, on peut noter que la variabilité des résultats est légèrement plus importante dans le premier cas que dans le second. En outre, les deux autres remarques tirées pour le jeu de données simulées restent valables dans ce cas (intérêt de l'apprentissage et gain en temps).

La figure 3.4 présente un exemple de banc de filtres appris à partir du jeu de données CHSC (sans ajout de bruit). Sur cet exemple, le filtre le plus court possède une RI identiquement nulle. De manière générale, on remarque expérimentalement que les filtres longs sont préférés d'autant que le rapport signal sur bruit (*Signal to Noise Ratio*, SNR) est faible.

Il est très difficile d'estimer théoriquement la qualité d'un relâchement semi-défini car celui-ci est dépendant de la structure du problème traité. En revanche, on peut analyser expérimentalement les différences moyennes entre les minima locaux obtenus. La figure 3.5 page 73 met en lumière ces différences sur le second jeu de données en faisant figurer les résultats du programme SDP, du relâchement avec tirage aléatoire et d'une descente de gradient initialisée par le précédent résultat. On vérifie tout d'abord sur cette figure que l'op-

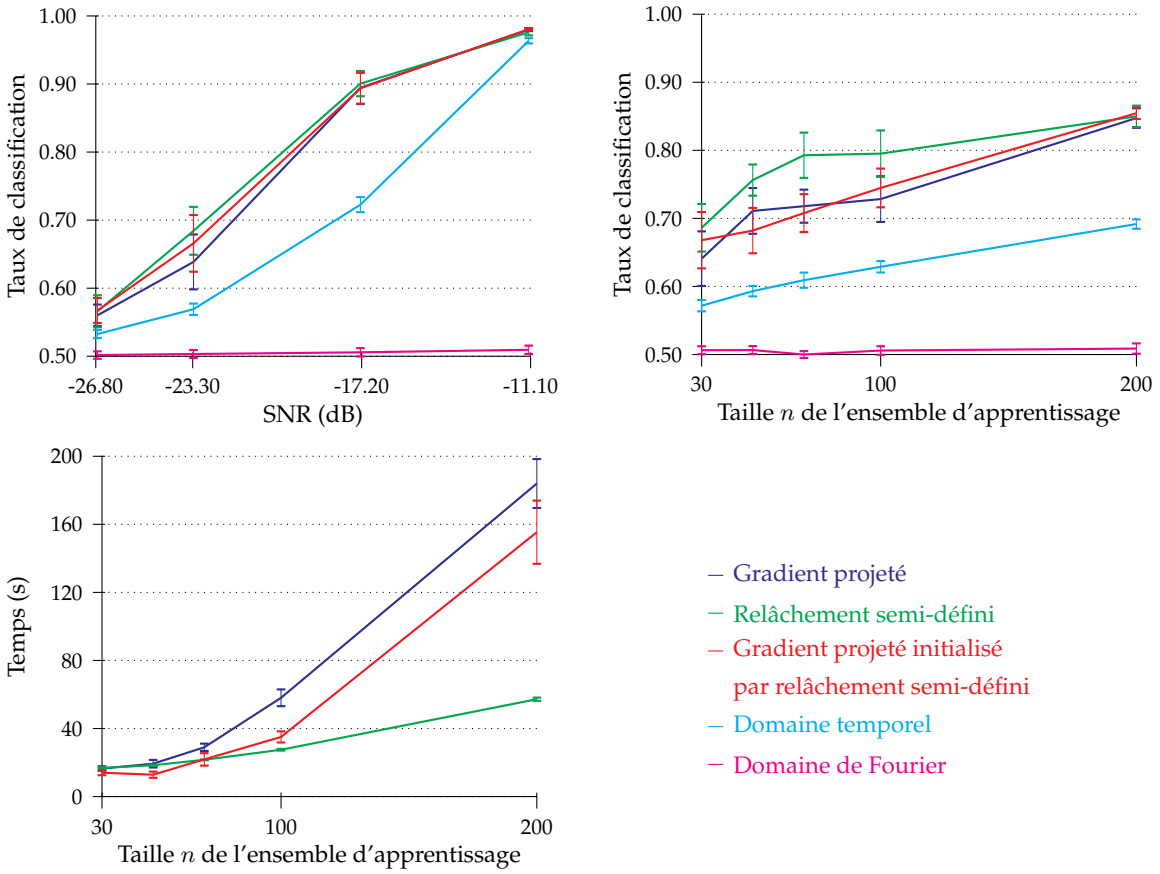


FIGURE 3.2 – Taux de bonne classification sur l'ensemble de test et temps d'apprentissage (données simulées).

timum du problème SDP minore la valeur de la fonction objectif SVM de toutes les autres approches. Il est ensuite intéressant de noter que le relâchement borne supérieurement la descente de gradient, elle même supérieure aux résultats obtenus avec une initialisation adéquate.

3.4 RÉGULARISATION PAR FAMILLE GÉNÉRATRICE

Les deux approches présentées dans la section qui précède souffrent des faiblesses suivantes :

- ◇ la technique par relâchement semi-défini fait largement croître le nombre de variables du problème d'optimisation. Ce nombre est lié de manière quadratique à la taille T des signaux et à la longueur des filtres. En pratique, on constate effectivement que le solveur SeDuMi [Sturm, 1999] n'est capable de gérer que des problèmes de petites tailles ;
- ◇ la technique par descente de gradient requiert que tous les opérateurs mis en jeu soient dérivables ;
- ◇ les taux de classification sur les ensembles de test et d'apprentissage divergent avec le niveau de bruit et la diminution du nombre d'exemples d'apprentissage (figure 3.6 page 74). Ceci suggère une trop grande complexité de notre modèle, entraînant un phénomène de sur-apprentissage.

Pour ces raisons, nous introduisons ici une approche différente, dans laquelle nous choisissons au préalable une famille de filtres contrôlée par peu de paramètres. Le choix d'une

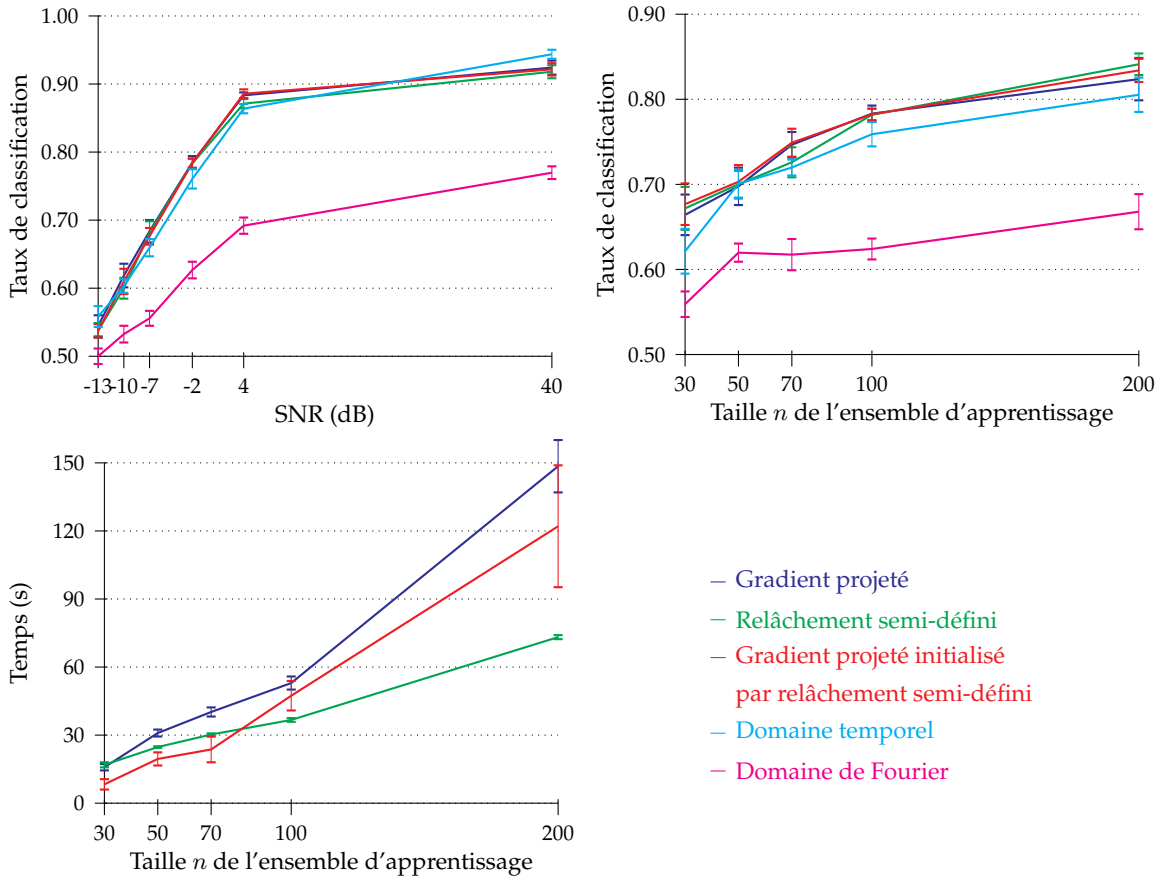


FIGURE 3.3 – Taux de bonne classification sur l'ensemble de test et temps d'apprentissage (données CHSC).

famille a d'abord pour effet de régulariser le problème, et ainsi de diminuer sa complexité. En outre, il conduit à la mise en place d'un algorithme MKL, qui bénéficie de qualités faisant écho aux faiblesses exposées ci-avant :

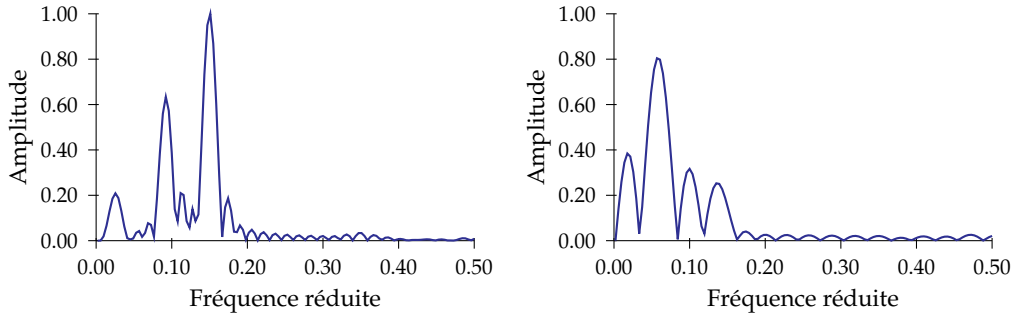
- ◇ la capacité à traiter des problèmes de plus grande dimension que par relâchement semi-défini ;
- ◇ la possibilité d'intégrer un grand choix de fonctions d'agrégation, y compris celles qui ne sont pas dérivables ;
- ◇ un cadre permettant d'apprendre une fonction d'agrégation.

3.4.1 Restriction du problème

Afin de mettre en place le cadre proposé, nous effectuons trois hypothèses, respectivement sur l'ensemble des BdF, sur les fonctions d'agrégation et sur les noyaux SVM. Ces trois éléments constituent les trois étapes consécutives de notre chaîne de traitement.

La première hypothèse, contraint la nature des BdF appris. En effet, on suppose disposer d'une famille donnée de filtres (par exemple des filtres passe-bandes tels que des ondelettes) et on cherche à construire un BdF *engendré* par des filtres de cette famille.

Définition 3.4.1 (Ensemble de BdF engendré)**.**
Soient $\mathcal{P}$ un ensemble borné (potentiellement continu), $\boldsymbol{h}_\cdot : \boldsymbol{\theta} \in \mathcal{P} \mapsto \boldsymbol{h}_{\boldsymbol{\theta}}$ une famille de RI normalisées paramétrée par $\boldsymbol{\theta}$ et $N_\cdot : \boldsymbol{\theta} \in \mathcal{P} \mapsto N_{\boldsymbol{\theta}}$ une famille de facteurs de décimation. On appelle ensemble de BdF engendré par $\{\boldsymbol{h}_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{P}}$ et $\{N_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{P}}$ le sous-ensemble $\overline{\mathcal{T}}$ de $\mathcal{T}$



(a) Filtre 1 (longueur : 64). (b) Filtre 2 (longueur : 32).

FIGURE 3.4 – Exemple de banc de filtres appris.

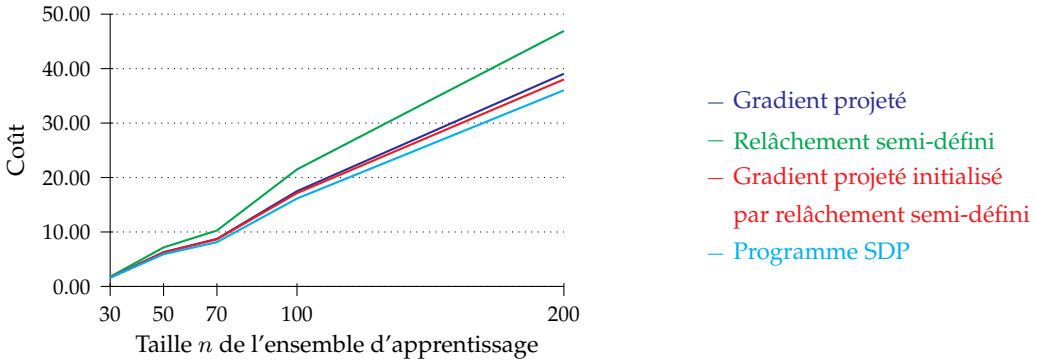


FIGURE 3.5 – Valeurs optimales moyennes de la fonction de coût.

défini par :

$$\overline{\mathcal{T}} = \left\{ (\tilde{\mu}_{\boldsymbol{\theta}} \boldsymbol{h}_{\boldsymbol{\theta}}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{A}},\ \tilde{\boldsymbol{\mu}} \succcurlyeq 0, \|\tilde{\boldsymbol{\mu}}\|_{\ell_2} = 1, \mathcal{A} \subset \mathcal{P}, \mathrm{Card}(\mathcal{A}) < \infty \right\},$$

où $\mathrm{Card}(\mathcal{A}) < \infty$ signifie que l'ensemble $\mathcal{A}$ est de cardinalité finie.

Remarque 12.

La condition $\|\tilde{\boldsymbol{\mu}}\|_{\ell_2} = 1$ assure que le BdF construit est d'énergie unitaire. Dans les approches précédentes, nous avons privilégié des BdF d'énergie *au plus* unitaire : $\sum_{l=1}^{d} \|\boldsymbol{h}_{\boldsymbol{l}}\|_{\ell_2}^2 \leq 1$. Ceci est justifié par la volonté de ne pas introduire de non-convexités qui peuvent être facilement levées. En effet, cette formulation est convexe tandis que $\sum_{l=1}^{d} \|\boldsymbol{h}_{\boldsymbol{l}}\|_{\ell_2}^2 = 1$ ne l'est pas. En réalité, il est quand même préférable d'assurer que l'énergie du BdF est unitaire. Cela permet à chaque contribution du problème d'optimisation de garder son propre rôle.

La définition d'un ensemble de BdF engendré par une famille de filtres est semblable à celle d'un espace vectoriel engendré par une famille de vecteur. Cela consiste donc à choisir des filtres, puis à les assembler suivant une pondération adéquate par rapport à la tâche souhaitée. En revanche, d'un point de vue pratique, cette définition pose une difficulté : le nombre de filtres constituant un banc pris dans $\overline{\mathcal{T}}$ est *a priori* inconnu. Il est ainsi avantageux de récrire la définition de $\overline{\mathcal{T}}$ de manière différente et plus appropriée :

$$\overline{\mathcal{T}} = \left\{ (\tilde{\mu}_{\boldsymbol{\theta}} \boldsymbol{h}_{\boldsymbol{\theta}}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{P}},\ \tilde{\boldsymbol{\mu}} \succcurlyeq 0, \|\tilde{\boldsymbol{\mu}}\|_{\ell_2} = 1 \text{ et } \tilde{\boldsymbol{\mu}} \text{ à Support Fini (SF)} \right\},$$

où SF signifie que le nombre de valeurs non-nulles du vecteur $\tilde{\boldsymbol{\mu}}$ est fini. Cette nouvelle définition de $\overline{\mathcal{T}}$ met en lumière notre utilisation future de la notion de parcimonie afin de gérer l'aspect continu de l'ensemble de paramètres $\mathcal{P}$.

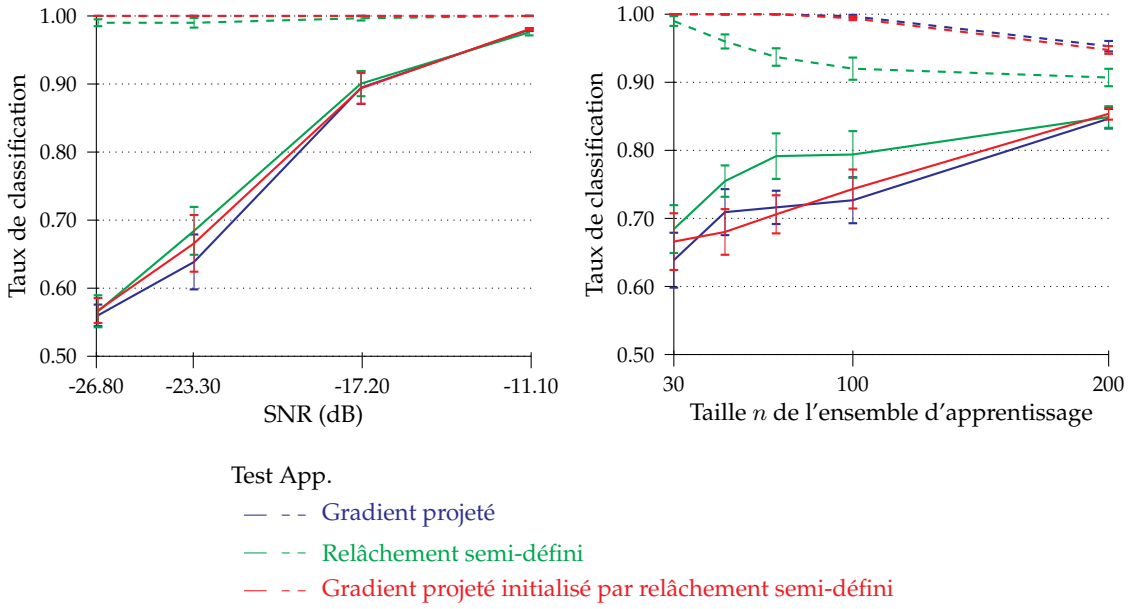


FIGURE 3.6 – Taux de bonne classification sur les ensembles de test et d'apprentissage (données simulées).

Le cadre que nous mettons en place suppose que la fonction d'agrégation $\rho\colon \mathcal{O} \to \mathcal{X}$ est positivement homogène de degré 1. Cette hypothèse est relativement faible puisque toutes les fonctions d'agrégation présentées dans le tableau 2.1 page 36 la vérifie. De plus, on remarque qu'à l'image de la fonction *maximum local*, la dérivabilité n'est pas nécessaire pour obtenir une fonction d'agrégation positivement homogène.

Définition 3.4.2 (Fonction positivement homogène)**.**
Une fonction $\rho\colon \mathcal{O} \to \mathcal{X}$ est positivement homogène de degré p ($p \in \mathbb{N}^*$) lorsque :

$$\forall \lambda \in \mathbb{R}_+, \forall a \in \mathcal{O}\colon \rho(\lambda a) = \lambda^p \rho(a).$$

L'ultime hypothèse faite dans ce contexte concerne le noyau SVM $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Nous supposons qu'il est soit linéaire, soit gaussien isotrope (paramétré par un certain γ positif). La restriction de notre étude à ces deux noyaux est d'ordre purement théorique (nous ne sommes pas à même d'étendre le cadre présenté ici à d'autres noyaux SVM). En revanche, d'un point de vue pratique, il est recommandé d'utiliser le noyau gaussien (qui donne des résultats toujours au moins égaux à ceux du noyau linéaire) excepté si le praticien ne dispose pas d'assez de temps pour réaliser la validation-croisée nécessaire à la détermination du paramètre de coût C et de celui du noyau gaussien γ. Dans ce cas, on peut préférer le noyau linéaire, qui est plus rapide à calculer et qui ne conduit à déterminer qu'un seul paramètre par validation croisée (C).

Dans le cas linéaire, et avec les hypothèses faites, nous pouvons établir l'équivalence suivante entre le problème d'apprentissage d'un BdF discriminant et celui d'un noyau multiple convexe.

Proposition 3.4.1 (Équivalence linéaire)**.**
Soient $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ le noyau linéaire, ρ une fonction d'agrégation positivement homogène de degré 1 et $\overline{\mathcal{T}}$ un ensemble de BdF engendré. Si $\mathcal{T}$ est restreint à $\overline{\mathcal{T}}$ alors (3.1) *est équivalent à*

$$\begin{array}{ll} \underset{\boldsymbol{\mu} \in \mathbb{R}^{\mathcal{P}}}{\text{minimiser}} & J\left(u_0, k_{\boldsymbol{\mu}}\right) \\ \text{tel que} & \left\{ \begin{array}{l} \mathbb{1}^T \boldsymbol{\mu} = 1 \\ \boldsymbol{\mu} \succcurlyeq 0, \end{array} \right. \end{array} \tag{3.11}$$

où $u_0 = \{(\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})\}_{\boldsymbol{\theta}\in\mathcal{P}}$, $\boldsymbol{\mu} = \tilde{\boldsymbol{\mu}}^2$ *et* $k_{\boldsymbol{\mu}}$ *est un noyau multiple linéaire construit à partir d'un ensemble de noyaux générateurs* $\{k_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{P}}$*, définis par :*

$$\forall \boldsymbol{\theta}\in\mathcal{P}, \forall \boldsymbol{x} = \mathrm{Vec}\left((\boldsymbol{x_\theta})_{\boldsymbol{\theta}\in\mathcal{A}}\right), \boldsymbol{z} = \mathrm{Vec}\left((\boldsymbol{z_\theta})_{\boldsymbol{\theta}\in\mathcal{A}}\right) \in \mathcal{X} : k_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{z}) = \langle \boldsymbol{x_\theta} \mid \boldsymbol{z_\theta}\rangle_{\ell_2}.$$

Remarque 13.
Pour un vecteur caractéristique $\boldsymbol{x} = \mathrm{Vec}\left((\boldsymbol{x_\theta})_{\boldsymbol{\theta}\in\mathcal{A}}\right) \in \mathcal{X}$, issu d'un BdF engendré par une famille $\boldsymbol{h}_\cdot : \boldsymbol{\theta}\in\mathcal{P} \mapsto \boldsymbol{h_\theta}$, la partie $\boldsymbol{x_\theta}$ correspond à un signal discret filtré par $\boldsymbol{h_\theta}$, décimé par $N_{\boldsymbol{\theta}}$ puis agrégé par ρ : $\boldsymbol{x_\theta} = \rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h_\theta}\star\boldsymbol{s}\right]\right)$.

Remarque 14.
Le noyau $k_{[\boldsymbol{\mu}]}$ s'exprime par :

$$\forall \boldsymbol{x} = \mathrm{Vec}\left((\boldsymbol{x_\theta})_{\boldsymbol{\theta}\in\mathcal{A}}\right), \boldsymbol{z} = \mathrm{Vec}\left((\boldsymbol{z_\theta})_{\boldsymbol{\theta}\in\mathcal{A}}\right) \in \mathcal{X} : k_{[\boldsymbol{\mu}]} = \sum_{\boldsymbol{\theta}\in\mathcal{P}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{z}) = \sum_{\boldsymbol{\theta}\in\mathcal{P}} \mu_{\boldsymbol{\theta}} \langle \boldsymbol{x_\theta} \mid \boldsymbol{z_\theta}\rangle_{\ell_2}.$$

Démonstration. On montre d'abord la relation entre BdF et noyau multiple : soient $u \in \overline{\mathcal{T}}$, $u = (\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$, et ρ une fonction d'agrégation positivement homogène de degré 1, alors :

$$\begin{aligned}
\forall \boldsymbol{s},\boldsymbol{t}\in\mathcal{U}, &\langle (\rho\circ u)(\boldsymbol{s}) \mid (\rho\circ u)(\boldsymbol{t})\rangle_{\ell_2} \\
&= \langle \rho\left((\downarrow N_{\boldsymbol{\theta}}\left[\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h_\theta}\star\boldsymbol{s}\right])_{\boldsymbol{\theta}\in\mathcal{A}}\right) \mid \rho\left((\downarrow N_{\boldsymbol{\theta}}\left[\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h_\theta}\star\boldsymbol{t}\right])_{\boldsymbol{\theta}\in\mathcal{A}}\right)\rangle_{\ell_2} \\
&= \sum_{\boldsymbol{\theta}\in\mathcal{A}} \tilde{\mu}_{\boldsymbol{\theta}}^2 \langle \rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h_\theta}\star\boldsymbol{s}\right]\right) \mid \rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h_\theta}\star\boldsymbol{t}\right]\right)\rangle_{\ell_2},
\end{aligned}$$

autrement dit, pour le noyau linéaire k :

$$\forall \boldsymbol{s},\boldsymbol{t}\in\mathcal{U}, k\Big((\rho\circ u)(\boldsymbol{s}),(\rho\circ u)(\boldsymbol{t})\Big) = k_{\boldsymbol{\mu}}\Big((\rho\circ u_0)(\boldsymbol{s}),(\rho\circ u_0)(\boldsymbol{t})\Big),$$

avec $\boldsymbol{\mu} = \tilde{\boldsymbol{\mu}}^2$. En outre, puisque :

$$J(u,k) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j k\Big((\rho\circ u)(\boldsymbol{s_i}),(\rho\circ u)(\boldsymbol{s_j})\Big),$$

où $\boldsymbol{\alpha}$ est obtenu de manière adéquate et vérifiant $0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C\mathbb{1}$, alors on a l'égalité :

$$J(u,k) = J(u_0,k_{\boldsymbol{\mu}}).$$

Montrons à présent qu'un point optimal pour (3.1) est optimal pour (3.11). Soit $u \in \overline{\mathcal{T}}$, $u = (\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$, un point optimal pour (3.1) et $\boldsymbol{\mu}\in\mathbb{R}^{\mathcal{P}}$ vérifiant $\boldsymbol{\mu_{\mathcal{A}}} = \tilde{\boldsymbol{\mu}}^2$ et $\boldsymbol{\mu_{\mathcal{P}\setminus\mathcal{A}}} = 0$. Alors $\boldsymbol{\mu}$ est un point faisable de (3.11) puisque $\boldsymbol{\mu}\succcurlyeq 0$ et $\mathbb{1}^T\boldsymbol{\mu} = \|\tilde{\boldsymbol{\mu}}\|_{\ell_2} = 1$. De plus, soit $\boldsymbol{\mu'}$ un point optimal pour (3.11). D'après le théorème 2.1 de [Gehler et Nowozin, 2008b], $\boldsymbol{\mu'}$ a un support fini ; appelons-le $\mathcal{A}$ et définissons u' le BdF engendré par $\sqrt{\boldsymbol{\mu'_{\mathcal{A}}}}$ (il existe bien). Alors

$$J(u_0,k_{\boldsymbol{\mu'}}) = J(u',k) \geq J(u,k) = J(u_0,k_{\boldsymbol{\mu}}),$$

par optimalité de u et

$$J(u_0,k_{\boldsymbol{\mu'}}) \leq J(u_0,k_{\boldsymbol{\mu}}),$$

par optimalité de $\boldsymbol{\mu'}$, donc $J(u_0,k_{\boldsymbol{\mu'}}) = J(u_0,k_{\boldsymbol{\mu}})$ et $\boldsymbol{\mu}$ est optimal pour (3.11).

Réciproquement, soit $\boldsymbol{\mu}$ un point optimal pour (3.11). D'après le théorème 2.1 de [Gehler et Nowozin, 2008b], $\boldsymbol{\mu}$ a un support fini ; appelons-le $\mathcal{A}$ et définissons u le BdF de $\overline{\mathcal{T}}$ de poids $\sqrt{\boldsymbol{\mu_{\mathcal{A}}}}$. Pour tout BdF u' de $\overline{\mathcal{T}}$ (de poids $\tilde{\boldsymbol{\mu'}}$), appelons $\boldsymbol{\mu'} = \tilde{\boldsymbol{\mu'}}^2$. Alors

$$J(u',k) = J(u_0,k_{\boldsymbol{\mu'}}) \geq J(u_0,k_{\boldsymbol{\mu}}) = J(u,k),$$

donc u est optimal pour (3.11). ■

La proposition 3.4.1 nous assure donc que le problème (3.1) admet bien une solution dans $\overline{\mathcal{T}}$ et que celle-ci est calculable sous la forme d'une combinaison linéaire d'une infinité de noyaux. Nous laissons la résolution d'un tel problème à la section suivante et abordons maintenant le cas du noyau gaussien.

Proposition 3.4.2 (Équivalence gaussienne)**.**
Soient $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ un noyau gaussien de paramètre γ (strictement positif), ρ une fonction d'agrégation positivement homogène de degré 1 et $\overline{\mathcal{T}}$ un ensemble de BdF engendré. Si $\mathcal{T}$ est restreint à $\overline{\mathcal{T}}$ alors (3.1) *est équivalent à*

$$\begin{array}{ll} \underset{\boldsymbol{\mu}\in\mathbb{R}^{\mathcal{P}}}{\text{minimiser}} & J\left(u_0, k_{\boldsymbol{\mu}}\right) \\ \text{tel que} & \left\{ \begin{array}{l} \mathbb{1}^T\boldsymbol{\mu} = 1 \\ \boldsymbol{\mu} \succcurlyeq 0 \\ \boldsymbol{\mu} \text{ SF}, \end{array} \right. \end{array} \tag{3.12}$$

où $u_0 = \{(\boldsymbol{h}_{\boldsymbol{\theta}}, N_{\boldsymbol{\theta}})\}_{\boldsymbol{\theta}\in\mathcal{P}}$, $\boldsymbol{\mu} = \tilde{\boldsymbol{\mu}}^2$ et $k_{\boldsymbol{\mu}}$ est un noyau multiple multiplicatif construit à partir de l'ensemble de noyaux générateurs $\{k_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{P}}$, définis par :

$$\forall \boldsymbol{\theta} \in \mathcal{P},\ \forall \boldsymbol{x} = \mathrm{Vec}\left((\boldsymbol{x}_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}\right), \boldsymbol{z} = \mathrm{Vec}\left((\boldsymbol{z}_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}\right) \in \mathcal{X}\colon k_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\gamma \left\|\boldsymbol{x}_{\boldsymbol{\theta}} - \boldsymbol{z}_{\boldsymbol{\theta}}\right\|_{\ell_2}^2\right).$$

Remarque 15.
Le noyau $k_{[\boldsymbol{\mu}]}$ s'exprime par :

$$\begin{aligned} &\forall \boldsymbol{x} = \mathrm{Vec}\left((\boldsymbol{x}_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}\right), \boldsymbol{z} = \mathrm{Vec}\left((\boldsymbol{z}_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}\right) \in \mathcal{X}\colon \\ &k_{[\boldsymbol{\mu}]} = \prod_{\boldsymbol{\theta}\in\mathcal{P}} k_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{z})^{\mu_{\boldsymbol{\theta}}} = \exp\left(\sum_{\boldsymbol{\theta}\in\mathcal{P}} -\gamma\mu_{\boldsymbol{\theta}} \left\|\boldsymbol{x}_{\boldsymbol{\theta}} - \boldsymbol{z}_{\boldsymbol{\theta}}\right\|_{\ell_2}^2\right). \end{aligned}$$

Démonstration. Soient $u \in \overline{\mathcal{T}}$, $u = (\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h}_{\boldsymbol{\theta}}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$, et ρ une fonction d'agrégation positivement homogène de degré 1, alors

$$\begin{aligned} &\forall \boldsymbol{s}, \boldsymbol{t} \in \mathcal{U}, \left\|(\rho \circ u)(\boldsymbol{s}) - (\rho \circ u)(\boldsymbol{t})\right\|_{\ell_2}^2 \\ &\qquad = \left\|\rho\left((\downarrow N_{\boldsymbol{\theta}}\left[\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h}_{\boldsymbol{\theta}} \star \boldsymbol{s}\right])_{\boldsymbol{\theta}\in\mathcal{A}}\right) - \rho\left((\downarrow N_{\boldsymbol{\theta}}\left[\tilde{\mu}_{\boldsymbol{\theta}}\boldsymbol{h}_{\boldsymbol{\theta}} \star \boldsymbol{t}\right])_{\boldsymbol{\theta}\in\mathcal{A}}\right)\right\|_{\ell_2}^2 \\ &\qquad = \sum_{\boldsymbol{\theta}\in\mathcal{A}} \tilde{\mu}_{\boldsymbol{\theta}}^2 \left\|\rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h}_{\boldsymbol{\theta}} \star \boldsymbol{s}\right]\right) - \rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h}_{\boldsymbol{\theta}} \star \boldsymbol{t}\right]\right)\right\|_{\ell_2}^2, \end{aligned}$$

autrement dit, pour un noyau gaussien k :

$$\forall \boldsymbol{s}, \boldsymbol{t} \in \mathcal{U},\ k\Big((\rho \circ u)(\boldsymbol{s}), (\rho \circ u)(\boldsymbol{t})\Big) = k_{\boldsymbol{\mu}}\Big((\rho \circ u_0)(\boldsymbol{s}), (\rho \circ u_0)(\boldsymbol{t})\Big),$$

avec $\boldsymbol{\mu} = \tilde{\boldsymbol{\mu}}^2$. Le reste de la preuve est identique au cas linéaire en remplaçant le recours au théorème 2.1 de [Gehler et Nowozin, 2008b] (qui ne tient plus du fait de la non-convexité du problème (3.12)) par la contrainte $\boldsymbol{\mu}$ SF. ■

3.4.2 Apprentissage de la transformée temps-fréquence

Grâce aux éléments donnés ci-avant, le problème d'apprentissage d'une transformée TF devient celui d'une combinaison infinie de noyaux :

$$\begin{array}{ll} \underset{\boldsymbol{\mu}\in\mathbb{R}^{\mathcal{P}}}{\text{minimiser}} & J\left(u_0, k_{\boldsymbol{\mu}}\right) \\ \text{tel que} & \left\{ \begin{array}{l} \mathbb{1}^T\boldsymbol{\mu} = 1 \\ \boldsymbol{\mu} \succcurlyeq 0 \\ \boldsymbol{\mu} \text{ SF}, \end{array} \right. \end{array} \tag{3.13}$$

où $u_0 = (\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}}$ et $k_{[\boldsymbol{\mu}]}$ est un noyau multiple convexe ou multiplicatif. La contrainte $\boldsymbol{\mu}$ SF est nécessaire pour assurer qu'une solution à support fini existe pour l'apprentissage d'un noyau multiplicatif gaussien, mais est inutile pour un noyau convexe (la convexité du problème d'apprentissage permet en effet d'établir la finitude des solutions [Gehler et Nowozin, 2008b]).

Quand l'ensemble $\mathcal{P}$ est fini, ce problème peut être résolu grâce à des algorithmes MKL existants comme celui de [Szafranski *et coll.*, 2010] dans le cas du noyau linéaire (problème d'optimisation convexe) et celui de [Varma et Babu, 2009] pour le noyau gaussien (problème d'optimisation non-convexe). Dans ce dernier cas, le problème étudié ici est légèrement différent de celui originellement introduit dans [Varma et Babu, 2009] (et rappelé dans la section 1.4) puisque nous avons remplacé la régularisation de Tikhonov (terme additif dans la fonction objectif) par une régularisation d'Ivanov (contrainte explicite). Cette alternative est apparue naturellement de nos hypothèses de travail. Il existe toutefois deux raisons de préférer une régularisation explicite sur $\boldsymbol{\mu}$:

- ◇ premièrement, il n'y a pas de coefficient de régularisation à déterminer (ce qui est difficile en pratique car nécessitant soit une étude théorique approfondie, soit des ressources de calcul importantes) ;
- ◇ deuxièmement, puisque $\boldsymbol{\mu}$ est ainsi assuré de rester sur la sphère unité de la norme ℓ_1, le paramètre de coût C conserve son rôle originel de compromis entre le terme d'attache aux données et la régularisation de f.

L'algorithme 2 résout le problème (3.13) pour un nombre fini de paramètres de filtres (*i.e.* $\mathrm{Card}(\mathcal{P}) < \infty$). La résolution du problème MKL est obtenue grâce à une technique de point fixe pour le noyau linéaire (proposition 3.4.1) [Szafranski *et coll.*, 2010]. Dans le cas du noyau gaussien (proposition 3.4.2), nous avons implémenté un algorithme de descente de gradient réduit [Luenberger, 1984], muni d'une recherche en ligne par retour sur trace.

Données : ensemble de signaux d'apprentissage $\{(\boldsymbol{s_i}, y_i)\}_{1\leq i\leq n}$.
1 **retourner** *BdF u et classifieur f.*
2 $u_0 \leftarrow (\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}}$ *{banc de filtres normalisés}* ;
3 $(\boldsymbol{\mu}, f) \leftarrow$ résolution du problème MKL avec $(u_0, (k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}})$;
4 $\mathcal{A} \leftarrow \{\boldsymbol{\theta} \in \mathcal{A}, \mu_{\boldsymbol{\theta}} > 0\}$;
5 $u \leftarrow \left(\sqrt{\mu_{\boldsymbol{\theta}}}\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}}\right)_{\boldsymbol{\theta}\in\mathcal{A}}$;

Algorithme 2 : Algorithme $\mathfrak{B}$ d'apprentissage de BdF pour un nombre fini de paramètres de filtres.

Toutefois, en pratique $\mathcal{P}$ est infini du fait de la nature continue des paramètres $\boldsymbol{\theta}$ des RI. Ainsi, l'une de nos contributions principales est l'algorithme 3 (nommé Filter-MKL et détaillé dans la section suivante), permettant de résoudre le problème d'apprentissage (3.13) d'une transformée TF.

En réalité, un algorithme semblable existe quand le noyau k est linéaire et se nomme apprentissage de noyaux infinis (*Infinite Kernel Learning*, IKL) [Gehler et Nowozin, 2008a]. IKL est un problème d'apprentissage introduit et résolu par Gehler et Nowozin [Gehler et Nowozin, 2008a] dans le but d'étendre le principe d'une combinaison convexe de noyaux à une infinité de noyaux générateurs, dont le vecteur de poids $\boldsymbol{\mu}$ est à support fini. En invoquant la dualité forte du problème d'optimisation (il est convexe), celui-ci est réduit à un programme linéaire semi-infini (*Semi-Infinite Linear Program*, SILP). Les auteurs démontrent qu'une solution existe bien et résolvent le problème dual par un algorithme de génération de contraintes.

L'algorithme que nous proposons ici étend l'état de l'art en apprentissage automatique en étant le seul à gérer un produit infini de noyaux gaussiens et *a fortiori* l'unique à four-

Données : ensemble de signaux d'apprentissage $\{(\boldsymbol{s_i}, y_i)\}_{1 \leq i \leq n}$.
1 **retourner** *BdF* u *et classifieur* f.
2 $\mathcal{A} \leftarrow$ grille linéaire de paramètres des RI ;
3 $\bar{\boldsymbol{\mu}} \leftarrow \frac{1}{\mathrm{Card}(\mathcal{A})}\mathbb{1}$ { *poids initiaux* } ;
4 **tant que** *équilibre non-atteint* **faire**
5 $u \leftarrow (\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{A}}$ {*banc de filtres normalisés*} ;
6 $(\boldsymbol{\mu}, f) \leftarrow$ résolution du problème MKL avec $(u, (k_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{A}})$, initialisé avec $\bar{\boldsymbol{\mu}}$;
7 $\mathcal{A} \leftarrow \{\boldsymbol{\theta} \in \mathcal{A}, \mu_{\boldsymbol{\theta}} > 0\}$;
8 $\Theta \leftarrow$ échantillon aléatoire de $\mathcal{P}$;
9 $\hat{\boldsymbol{\theta}} \leftarrow \arg\max_{\boldsymbol{\theta} \in \Theta} V(\boldsymbol{\theta})$;
10 **si** $V(\hat{\boldsymbol{\theta}}) > \sum_{\boldsymbol{\theta} \in \mathcal{A}} \mu_{\boldsymbol{\theta}} V(\boldsymbol{\theta})$ **alors** {*condition d'optimalité violée*}
11 $\mathcal{A} \leftarrow \mathcal{A} \cup \{\hat{\boldsymbol{\theta}}\}$;
12 $\bar{\boldsymbol{\mu}} \leftarrow [\boldsymbol{\mu}_{\boldsymbol{\mathcal{A}}}; 0]$;
13 **sinon**
14 équilibre atteint ;
15 $u \leftarrow \left(\sqrt{\mu_{\boldsymbol{\theta}}}\boldsymbol{h_\theta}, N_{\boldsymbol{\theta}}\right)_{\boldsymbol{\theta} \in \mathcal{A}}$;

Algorithme 3 : Algorithme $\mathfrak{B}$ d'apprentissage de BdF pour une famille continûment paramétrée de filtres (Filter-MKL).

nir une solution au problème d'apprentissage de BdF discriminant (3.13) quand k est le noyau gaussien. Concrètement, notre algorithme est une extension de celui proposé dans [Varma et Babu, 2009] permettant de gérer une famille continûment paramétrée de noyaux $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{P}}$. Notre algorithme s'applique aussi au problème d'apprentissage d'un BdF discriminant (3.13) quand le noyau est linéaire (les différences avec IKL sont discutées en section 3.4.6).

L'approche que nous avançons ici s'attaque au problème d'optimisation dans sa forme primale (étant donnée la non-convexité). Elle est fondée sur le principe d'ensemble actif [Nocedal et Wright, 2000] et est inspirée de [Yger et Rakotomamonjy, 2011]. Pour les besoins de la description de ce principe, nous supposons qu'un oracle nous a fourni l'ensemble fini $\mathcal{P}^*$ des paramètres solutions du problème (3.13). Commençons alors avec un candidat $\mathcal{A}$ ($\mathcal{A}$ est un sous-ensemble fini de paramètres de $\mathcal{P}$), supposé coïncider avec l'ensemble solution $\mathcal{P}^*$ et résolvons le problème à noyau multiple associé à $(\mu_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{A}}$ (ligne 6 de l'algorithme 3). À l'instar de l'algorithme 2, pour un noyau linéaire, nous utilisons le logiciel MKL de [Szafranski *et coll.*, 2010] tandis que pour le noyau gaussien, la stratégie d'optimisation utilisée est une descente de gradient réduit [Luenberger, 1984] accompagnée d'une recherche en ligne par retour sur trace. Cette étape peut être vue comme une descente par bloc de coordonnées en considérant que $\mu_{\boldsymbol{\theta}}$ est figé pour $\boldsymbol{\theta}$ dans $\mathcal{P} \backslash \mathcal{A}$. Il en résulte un ensemble actif $\mathcal{A}^*$ de paramètres dont les poids $\mu_{\boldsymbol{\theta}}$ sont non-nuls et son ensemble complémentaire non-actif $\mathcal{A} \backslash \mathcal{A}^*$ pour lequel les poids $\mu_{\boldsymbol{\theta}}$ sont nuls (effet de la contrainte de parcimonie $\mathbb{1}^T \boldsymbol{\mu} = 1$). Si $\mathcal{A}$ inclut $\mathcal{P}^*$, alors les conditions d'équilibre du problème sont vérifiées pour tout paramètre $\boldsymbol{\theta}$ de $\mathcal{P}$. Par contraposition, si les conditions d'équilibre ne sont pas vérifiées pour un certain $\boldsymbol{\theta}$ de $\mathcal{P}$, alors $\mathcal{A}$ n'inclut pas $\mathcal{P}^*$ et en particulier, le violateur $\boldsymbol{\theta}$ est absent de l'ensemble candidat $\mathcal{A}$. En conséquence, nous mettons à jour l'ensemble $\mathcal{A}$ grâce à la règle $\mathcal{A} \leftarrow \mathcal{A}^* \cup \{\boldsymbol{\theta}\}$ et résolvons une nouvelle fois le problème à noyau multiple associé. En alternant ces deux étapes (MKL et ajout d'un violateur), notre algorithme réalise une descente sur un nombre infini de paramètres.

Dans la prochaine section, nous détaillons la condition d'équilibre qui apparaît à la ligne 10 de l'algorithme 3, et qui permet à la fois de faire évoluer l'ensemble $\mathcal{A}$ vers une solution et

de déterminer l'arrêt de la descente.

3.4.3 Conditions d'équilibre

Cette section détaille le moyen de vérifier l'optimalité d'un vecteur de poids $\boldsymbol{\mu}$ donné, par rapport au problème (3.13). Si les conditions d'équilibre ne sont pas vérifiées pour un poids $\mu_{\boldsymbol{\theta}}$, alors le paramètre violateur $\boldsymbol{\theta}$ doit être ajouté à l'ensemble actif $\mathcal{A}$ et le nouveau problème à noyau multiple doit être résolu. La manière de trouver un tel paramètre $\boldsymbol{\theta}$ est discutée en section 3.4.4, de sorte que nous nous concentrons ici sur les seules conditions d'équilibre.

Proposition 3.4.3 (Non-optimalité linéaire [Yger et Rakotomamonjy, 2011])**.**
Soient $\boldsymbol{\mu} \in \mathbb{R}_+^{\mathcal{P}}$ faisable pour (3.11)*, $k_{[\boldsymbol{\mu}]}$ un noyau multiple convexe construit comme dans la proposition 3.4.1 et $\boldsymbol{\alpha}$ le (supposé unique) vecteur dual optimal du problème SVM avec comme noyau $k_{[\boldsymbol{\mu}]}$. Soit maintenant l'application :*

$$V : \boldsymbol{\theta} \in \mathcal{P} \mapsto \boldsymbol{\alpha}^T \boldsymbol{Y} \boldsymbol{K}_{\boldsymbol{\theta}+} \boldsymbol{Y} \boldsymbol{\alpha},$$

où $\boldsymbol{Y} = \mathrm{Diag}(\boldsymbol{y})$ est la matrice des étiquettes et $\boldsymbol{K}_{\boldsymbol{\theta}+} = \left(k_{\boldsymbol{\theta}}\left((\rho \circ u_0)(\boldsymbol{s_i}), (\rho \circ u_0)(\boldsymbol{s_j})\right)\right)_{1 \leq i,j \leq n}$ la matrice noyau de $k_{\boldsymbol{\theta}}$. Si

$$\exists \boldsymbol{\theta} \in \mathcal{P} \ / \ V(\boldsymbol{\theta}) > \sum_{\substack{\boldsymbol{\theta'} \in \mathcal{P} \\ \mu_{\boldsymbol{\theta'}} > 0}} \mu_{\boldsymbol{\theta'}} V(\boldsymbol{\theta'}),$$

alors $\boldsymbol{\mu}$ n'est pas optimal pour (3.11)*.*

Proposition 3.4.4 (Non-optimalité gaussienne)**.**
Soient $\boldsymbol{\mu} \in \mathbb{R}_+^{\mathcal{P}}$ faisable pour (3.12)*, $k_{[\boldsymbol{\mu}]}$ un noyau multiple multiplicatif construit comme dans la proposition 3.4.2 et $\boldsymbol{\alpha}$ le (supposé unique) vecteur dual optimal du problème SVM avec comme noyau $k_{[\boldsymbol{\mu}]}$. Soit maintenant l'application :*

$$V : \boldsymbol{\theta} \in \mathcal{P} \mapsto -\boldsymbol{\alpha}^T \boldsymbol{Y} \left(\boldsymbol{D}_{\boldsymbol{\theta}} \circ \boldsymbol{K}_{[\boldsymbol{\mu}]+} \right) \boldsymbol{Y} \boldsymbol{\alpha},$$

où $\boldsymbol{Y} = \mathrm{Diag}(\boldsymbol{y})$ est la matrice des étiquettes, $\boldsymbol{K}_{[\boldsymbol{\mu}]+} = \left(k_{[\boldsymbol{\mu}]}\left((\rho \circ u_0)(\boldsymbol{s_i}), (\rho \circ u_0)(\boldsymbol{s_j})\right)\right)_{1 \leq i,j \leq n}$ la matrice noyau de $k_{[\boldsymbol{\mu}]}$ et $\boldsymbol{D}_{\boldsymbol{\theta}} = \left(\left\| \rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h}_{\boldsymbol{\theta}} \star \boldsymbol{s_i}\right]\right) - \rho\left(\downarrow N_{\boldsymbol{\theta}}\left[\boldsymbol{h}_{\boldsymbol{\theta}} \star \boldsymbol{s_j}\right]\right) \right\|_{\ell_2}^2 \right)_{1 \leq i,j \leq n}$ la matrice de similarité des signaux filtrés par $\boldsymbol{h}_{\boldsymbol{\theta}}$. Si

$$\exists \boldsymbol{\theta} \in \mathcal{P} \ / \ V(\boldsymbol{\theta}) > \sum_{\substack{\boldsymbol{\theta'} \in \mathcal{P} \\ \mu_{\boldsymbol{\theta'}} > 0}} \mu_{\boldsymbol{\theta'}} V(\boldsymbol{\theta'}),$$

alors $\boldsymbol{\mu}$ n'est pas optimal pour (3.12)*.*

Démonstration. La preuve réside dans la simple expression des conditions de Karush-Kuhn-Tucker (KKT) du problème (3.12). En effet, malgré la non-convexité de (3.12), les conditions KKT restent nécessaires (mais ne sont pas suffisantes) pour un point optimal $\boldsymbol{\mu}$. Appelons $\mathfrak{L}$ le lagrangien (primal) associé au problème (3.12), λ le multiplicateur de Lagrange de la contrainte de parcimonie $\mathbb{1}^T \boldsymbol{\mu} = 1$ et $\boldsymbol{\tau}$ le vecteur dual de la contrainte de positivité $\boldsymbol{\mu} \succcurlyeq 0$. Le lagrangien s'exprime alors par :

$$\forall (\boldsymbol{\mu}, \lambda, \boldsymbol{\tau}) \in \mathbb{R}^{\mathcal{P}} \times \mathbb{R} \times \mathbb{R}^{\mathcal{P}} : \mathfrak{L}(\boldsymbol{\mu}, \lambda, \boldsymbol{\tau}) = J\left(u_0, k_{[\boldsymbol{\mu}]}\right) + \lambda(\mathbb{1}^T \boldsymbol{\mu} - 1) - \boldsymbol{\tau}^T \boldsymbol{\mu}.$$

En tout point d'équilibre du problème à noyau multiple (potentiellement un minimum ou un maximum local), les conditions KKT sont vérifiées :

$$\mathbb{1}^T \boldsymbol{\mu} = 1 \tag{3.14}$$
$$\boldsymbol{\mu} \succcurlyeq 0 \tag{3.15}$$
$$\boldsymbol{\tau} \succcurlyeq 0 \tag{3.16}$$
$$\forall \boldsymbol{\theta} \in \mathcal{P}, \tau_{\boldsymbol{\theta}} \mu_{\boldsymbol{\theta}} = 0 \tag{3.17}$$
$$\nabla_{\boldsymbol{\mu}} \mathfrak{L}(\boldsymbol{\mu}, \lambda, \boldsymbol{\tau}) = 0. \tag{3.18}$$

Les conditions (3.14) et (3.15) sont dites de faisabilité primale et (3.16) de faisabilité duale. La condition (3.18) correspond à la nullité du gradient et peut être récrite à partir de la fonction :

$$\tilde{J} \colon \boldsymbol{\mu} \in \mathbb{R}^{\mathcal{P}} \mapsto J\left(u_0, k_{[\boldsymbol{\mu}]}\right),$$

par

$$\nabla \tilde{J}(\boldsymbol{\mu}) + \lambda \mathbb{1} - \boldsymbol{\tau} = 0. \tag{3.19}$$

En combinant (3.15), (3.16), (3.17) et (3.19), on obtient :

$$\forall \boldsymbol{\theta} \in \mathcal{P}, \begin{cases} \dfrac{\partial \tilde{J}}{\partial \mu_{\boldsymbol{\theta}}}(\boldsymbol{\mu}) = -\lambda, \text{ si } \mu_{\boldsymbol{\theta}} > 0 \\ \dfrac{\partial \tilde{J}}{\partial \mu_{\boldsymbol{\theta}}}(\boldsymbol{\mu}) \geq -\lambda, \text{ si } \mu_{\boldsymbol{\theta}} = 0. \end{cases} \tag{3.20}$$

À l'équilibre, le multiplicateur de Lagrange λ est donné par :

$$\lambda = -\sum_{\substack{\boldsymbol{\theta} \in \mathcal{P} \\ \mu_{\boldsymbol{\theta}} > 0}} \mu_{\boldsymbol{\theta}} \frac{\partial \tilde{J}}{\partial \mu_{\boldsymbol{\theta}}}(\boldsymbol{\mu}).$$

Il reste maintenant à calculer la dérivée partielle de $\tilde{J}$. Ceci est possible grâce au théorème 1.4.2. Avant tout, récrivons le problème SVM à noyau $k_{[\boldsymbol{\mu}]}$ fixé :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\text{maximiser}} & \mathbb{1}^T \boldsymbol{\alpha} - \dfrac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{Y} \boldsymbol{K}_{[\boldsymbol{\mu}]_+} \boldsymbol{Y} \boldsymbol{\alpha} \\ \text{tel que} & \begin{cases} 0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C \\ \boldsymbol{y}^T \boldsymbol{\alpha} = 0, \end{cases} \end{array}$$

avec $\boldsymbol{Y}$ et $\boldsymbol{K}_{[\boldsymbol{\mu}]_+}$ définis dans l'énoncé de la proposition. En supposant une unique solution $\boldsymbol{\alpha}$, le théorème 1.4.2 assure que $\tilde{J}$ est dérivable et que :

$$\forall \boldsymbol{\theta} \in \mathcal{P} \colon \frac{\partial \tilde{J}}{\partial \mu_{\boldsymbol{\theta}}}(\boldsymbol{\mu}) = -\frac{1}{2} \boldsymbol{\alpha}^T \boldsymbol{Y} \frac{\partial \boldsymbol{K}_{[\boldsymbol{\mu}]_+}}{\partial \mu_{\boldsymbol{\theta}}} \boldsymbol{Y} \boldsymbol{\alpha}.$$

Étant dans le cas d'un noyau k gaussien :

$$\boldsymbol{K}_{[\boldsymbol{\mu}]_+} = \exp\left(-\gamma \sum_{\boldsymbol{\theta} \in \mathcal{P}} \mu_{\boldsymbol{\theta}} \boldsymbol{D}_{\boldsymbol{\theta}}\right),$$

où l'exponentielle est prise point à point. On obtient ainsi :

$$\frac{\partial \boldsymbol{K}_{[\boldsymbol{\mu}]_+}}{\partial \mu_{\boldsymbol{\theta}}} = -\gamma \boldsymbol{D}_{\boldsymbol{\theta}} \circ \boldsymbol{K}_{[\boldsymbol{\mu}]_+},$$

avec $\circ$ le produit matriciel de Hadamard. D'où :

$$\forall \boldsymbol{\theta} \in \mathcal{P} \colon \frac{\partial \tilde{J}}{\partial \mu_{\boldsymbol{\theta}}}(\boldsymbol{\mu}) = \frac{\gamma}{2} \boldsymbol{\alpha}^T \boldsymbol{Y} \left(\boldsymbol{D}_{\boldsymbol{\theta}} \circ \boldsymbol{K}_{[\boldsymbol{\mu}]_+} \right) \boldsymbol{Y} \boldsymbol{\alpha}.$$

De plus :

$$\lambda = -\frac{\gamma}{2} \sum_{\substack{\boldsymbol{\theta} \in \mathcal{P} \\ \mu_{\boldsymbol{\theta}} > 0}} \mu_{\boldsymbol{\theta}} \boldsymbol{\alpha}^T \boldsymbol{Y} \left(\boldsymbol{D}_{\boldsymbol{\theta}} \circ \boldsymbol{K}_{[\boldsymbol{\mu}]_+} \right) \boldsymbol{Y} \boldsymbol{\alpha}.$$

En reprenant (3.20), on observe que pour tout point $\boldsymbol{\mu}$ à l'équilibre :

$$\forall \boldsymbol{\theta} \in \mathcal{P} \colon \boldsymbol{\alpha}^T \boldsymbol{Y} \left(\boldsymbol{D}_{\boldsymbol{\theta}} \circ \boldsymbol{K}_{[\boldsymbol{\mu}]_+} \right) \boldsymbol{Y} \boldsymbol{\alpha} \geq \sum_{\substack{\boldsymbol{\theta'} \in \mathcal{P} \\ \mu_{\boldsymbol{\theta'}} > 0}} \mu_{\boldsymbol{\theta'}} \boldsymbol{\alpha}^T \boldsymbol{Y} \left(\boldsymbol{D}_{\boldsymbol{\theta'}} \circ \boldsymbol{K}_{[\boldsymbol{\mu}]_+} \right) \boldsymbol{Y} \boldsymbol{\alpha},$$

i.e. (avec la fonction V définie comme dans l'énoncé) :

$$\forall \boldsymbol{\theta} \in \mathcal{P} \colon V(\boldsymbol{\theta}) \leq \sum_{\substack{\boldsymbol{\theta'} \in \mathcal{P} \\ \mu_{\boldsymbol{\theta'}} > 0}} \mu_{\boldsymbol{\theta'}} V(\boldsymbol{\theta'}).$$

■

Dans les deux cas précédemment cités (noyaux linéaire et gaussien), la fonction solution du problème d'apprentissage (3.13) s'écrit :

$$\begin{aligned} \forall \boldsymbol{s} \in \mathcal{U}, f\left((\rho \circ u)(\boldsymbol{s})\right) &= \sum_{i=1}^{n} \alpha_i y_i k_{\boldsymbol{\mu}} \left((\rho \circ u_0)(\boldsymbol{s}), (\rho \circ u_0)(\boldsymbol{s_i})\right) \\ &= \sum_{i=1}^{n} \alpha_i y_i k \left((\rho \circ u)(\boldsymbol{s}), (\rho \circ u)(\boldsymbol{s_i})\right), \end{aligned} \tag{3.21}$$

où $\boldsymbol{\alpha}$ est le vecteur dual optimal du problème SVM appliqué à $((\rho \circ u_0)(\boldsymbol{s_i}), y_i))_{1 \leq i \leq n}$ avec le noyau multiple $k_{[\boldsymbol{\mu}]}$, $\boldsymbol{\mu}$ est le vecteur de poids optimal provenant de la résolution du problème MKL et $u = (\sqrt{\mu_{\boldsymbol{\theta}}} \boldsymbol{h}_{\boldsymbol{\theta}}, N_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \mathcal{A}}$. Notons que quand la transformée optimale u est connue, la fonction de décision (3.21) se réduit à celle d'un simple problème SVM de noyau k.

3.4.4 Détails d'implémentation

Dans cette section, nous discutons de deux points précis concernant l'algorithme proposé. Le premier est la manière concrète de vérifier les conditions d'optimalité (en particulier comment trouver un paramètre violateur $\boldsymbol{\theta}$). Le second concerne la convergence de l'algorithme et la normalisation des noyaux.

Génération de colonne

Les principales difficultés dans la prise en compte de la nature continue du paramètre $\boldsymbol{\theta}$ des RI sont :

- ◇ trouver un élément $\boldsymbol{\theta}$ de $\mathcal{P}$ violant les conditions d'équilibre : ceci peut être réalisé par tirages aléatoires [Rakotomamonjy *et coll.*, 2013] ;
- ◇ résoudre le problème variationnel :

$$\underset{\boldsymbol{\theta} \in \mathcal{P}}{\text{maximiser}} \quad V(\boldsymbol{\theta}),$$

afin d'arrêter le processus. En effet, si un maximiseur $\hat{\boldsymbol{\theta}}$ vérifie $V(\hat{\boldsymbol{\theta}}) \leq \sum_{\substack{\boldsymbol{\theta}\in\mathcal{P} \\ \mu_{\boldsymbol{\theta}}>0}} \mu_{\boldsymbol{\theta}} V(\boldsymbol{\theta})$, alors aucun paramètre violateur $\boldsymbol{\theta}$ n'existe et le système est donc à l'équilibre.

En pratique, ce problème est difficile à résoudre (il est non-convexe). En conséquence et conformément à [Rakotomamonjy *et coll.*, 2013], notre algorithme est construit sur l'heuristique suivante : si aucun paramètre d'un échantillon aléatoire à une itération donnée ne viole les conditions d'équilibre, alors ceci est vrai pour tout paramètre de $\mathcal{P}$.

Dans [Gehler et Nowozin, 2008a], ce sous-problème est résolu grâce à une méthode de Newton initialisée avec différents points. Ce type d'approche est particulièrement gourmand en temps de calcul, d'autant plus si le processus de descente de gradient est répété pour plusieurs initialisations. Au contraire, notre approche par tirage aléatoire est peu coûteuse puisqu'elle nécessite uniquement le calcul de la matrice noyau et l'évaluation du critère $V(\boldsymbol{\theta})$. De plus, puisque notre approche ne calcule aucun gradient, il est aisé d'utiliser des opérateurs non-différentiables par rapport à $\boldsymbol{\theta}$ (comme par exemple l'agrégation par maximum).

À l'instar de [Gehler et Nowozin, 2008a], dans lequel un algorithme de gradient est initialisé avec les paramètres violateurs de l'itération précédente, notre technique par tirage aléatoire peut être dirigée par une distribution de probabilité inférée grâce à la connaissance issue de l'itération précédente. En effet, supposons qu'à la première itération, les noyaux générateurs sont calculés à partir d'une grille régulière de paramètres de $\mathcal{P}$. Alors la solution de l'apprentissage du noyau multiple associé donne une idée grossière de la puissance de discrimination sur l'ensemble de paramètres $\mathcal{P}$. Ainsi, chaque nouvelle itération a principalement pour but d'affiner la solution obtenue à l'itération précédente, plutôt que d'en découvrir de nouvelles.

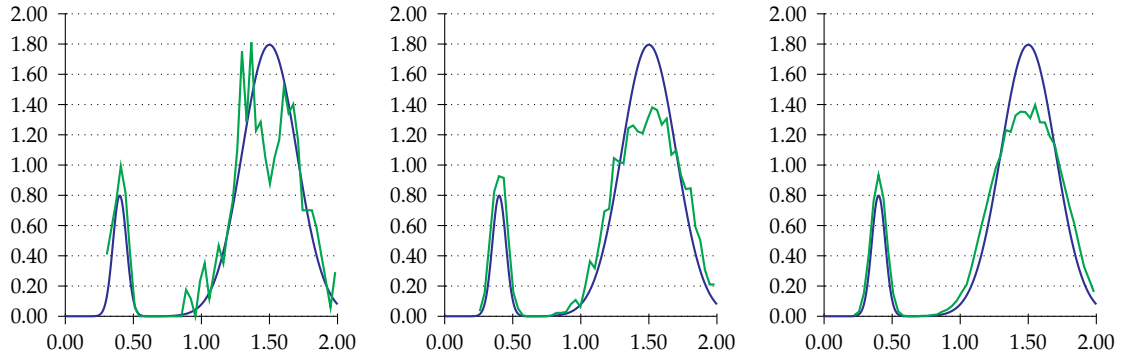
Une estimation de cette distribution de probabilité est directement liée à la fonction $\boldsymbol{\theta} \in \mathcal{P} \mapsto \max\left(0, V(\boldsymbol{\theta}) - \sum_{\substack{\boldsymbol{\theta}\in\mathcal{P} \\ \mu_{\boldsymbol{\theta}}>0}} \mu_{\boldsymbol{\theta}} V(\boldsymbol{\theta})\right)$. Une option envisageable afin de diriger l'échantillonnage aléatoire est alors d'appliquer une technique de régression à la fonction précédente et d'utiliser le résultat au sein d'un algorithme d'échantillonnage de type Metropolis-Hastings. En pratique, plusieurs milliers de réalisations sont nécessaires pour approcher la distribution estimée grâce à un algorithme de Metropolis-Hastings (voir figure 3.7 page ci-contre), tandis que seulement quelques centaines suffisent à chaque itération de notre algorithme. En conséquence, notre algorithme tire des paramètres aléatoirement suivant une loi uniforme sur $\mathcal{P}$ et selon une seconde loi uniforme sur une petite boîte centrée sur le paramètre violateur de l'itération précédente.

Considérations calculatoires

Étant donnée la non-convexité de notre problème d'apprentissage, nous ne cherchons pas à atteindre un minimum global. Théoriquement, notre algorithme peut même s'arrêter sur un maximum local bien qu'en pratique ceci soit peu probable puisque c'est un équilibre instable. En revanche, nous sommes assurés que la valeur de la fonction objectif décroît strictement à chaque itération.

Proposition 3.4.5.
À chaque itération de l'algorithme 3, la valeur de la fonction objectif du problème d'optimisation (3.13) *décroît strictement.*

Démonstration. À une itération j quelconque, on appelle $\bar{\boldsymbol{\mu}}^{(j)}$ le vecteur d'initialisation et $\boldsymbol{\mu}^{(j)}$ une solution du problème MKL (ligne 6 de l'algorithme 3). Selon l'algorithme, on a alors : $\forall \boldsymbol{\theta} \in \mathcal{A}$: $\bar{\mu}_{\boldsymbol{\theta}}^{(j+1)} = \mu_{\boldsymbol{\theta}}^{(j)}$ si $\boldsymbol{\theta}$ vérifie les conditions d'optimalité de l'itération j et $\bar{\mu}_{\boldsymbol{\theta}}^{(j+1)} = 0$ sinon. Ainsi, $\tilde{J}\left(\bar{\boldsymbol{\mu}}^{(j+1)}\right) = \tilde{J}\left(\boldsymbol{\mu}^{(j)}\right)$, puisque les seules coordonnées de $\bar{\boldsymbol{\mu}}^{(j+1)}$ qui diffèrent



(a) 500 points échantillonnés. (b) 5000 points échantillonnés. (c) 50 000 points échantillonnés.

FIGURE 3.7 – Exemple d'échantillonnage par l'algorithme de Metropolis-Hastings. La courbe bleu représente la densité de probabilité réelle et la courbe verte est un histogramme des points échantillonnés.

de celles de $\boldsymbol{\mu}^{(j)}$ sont nulles. De plus, puisque $\bar{\boldsymbol{\mu}}^{(j+1)}$ a été spécifiquement construit pour ne pas être à l'équilibre de (3.13) tout en étant admissible, alors la résolution du sous-problème MKL fait strictement décroître la valeur de la fonction objectif : $\tilde{J}\left(\boldsymbol{\mu}^{(j+1)}\right) < \tilde{J}\left(\bar{\boldsymbol{\mu}}^{(j+1)}\right)$. D'où $\tilde{J}\left(\boldsymbol{\mu}^{(j+1)}\right) < \tilde{J}\left(\boldsymbol{\mu}^{(j)}\right)$. ■

En effet, à chaque étape, un nouveau noyau multiple est construit sur le précédent en ajoutant un noyau de poids nul. Ce nouveau noyau multiple n'est pas un point critique puisque le noyau ajouté viole les conditions d'équilibre. De plus, la valeur de la fonction objectif est identique à celle à la fin de l'itération précédente puisque les deux noyaux multiples ne diffèrent que d'un noyau générateur muni d'un poids nul. Ainsi, on initialise un nouveau problème MKL avec un point non-critique et on est assuré que la résolution de celui-ci fera strictement décroître la valeur de la fonction objectif.

Apprendre un noyau multiple suppose de comparer entre elles les *pouvoirs discriminants* de chaque noyau. Pour cette raison, les noyaux doivent être d'*amplitudes* semblables, sans quoi un noyau peut obtenir un rôle prépondérant uniquement dû à son *amplitude* mais sans être discriminant. C'est l'un des écueils de la minimisation de la fonction objectif SVM et il est nécessaire d'y faire attention. Dans le but de prévenir cet effet néfaste pouvant conduire au sur-apprentissage, les noyaux sont normalisés suivant cette règle :

- ◇ si c'est un noyau linéaire, il est divisé par sa trace ;
- ◇ si c'est un noyau gaussien, la matrice de distance associée est divisée par sa norme de Frobenius.

Dans les deux cas, la normalisation est répercutée sur les poids $\boldsymbol{\mu}$ appris lors de la création du BdF discriminant final.

3.4.5 Détermination automatique de la fonction d'agrégation

De précédents travaux montrent l'intérêt de choisir de manière adéquate la fonction d'agrégation [Boureau *et coll.*, 2010a, Boureau *et coll.*, 2010b], puisqu'elle fait partie des premières étapes de la chaîne de traitement du signal. Suivant cette observation, une tentative de détermination automatique de la fonction d'agrégation est récemment apparue dans le cadre de l'apprentissage de dictionnaires invariants aux translations [Barthélemy *et coll.*, 2013]. De manière particulièrement intéressante, le cadre introduit ici permet aussi d'apprendre une fonction d'agrégation comme la concaténation de plusieurs fonctions, que nous appellerons *fonction d'agrégation multiple*.

Définition 3.4.3 (Fonction d'agrégation multiple)**.**
Soient p un entier naturel non-nul et $(\rho_r)_{1\leq r\leq p}$ un p-uplet de fonctions d'agrégation génératrices. L'ensemble des fonctions d'agrégation multiples construites à partir de ce p-uplet est :

$$\mathcal{L} = \{(\tilde{\eta}_r\rho_r)_{1\leq r\leq p},\ \tilde{\boldsymbol{\eta}} \succcurlyeq 0 \text{ et } \|\tilde{\boldsymbol{\eta}}\|_{\ell_2} = 1\}\,,$$

où $\tilde{\boldsymbol{\eta}}$ est un vecteur de pondération. De plus toute fonction d'agrégation multiple $\rho = (\tilde{\eta}_r\rho_r)_{1\leq r\leq p}$ est définie par :

$$\forall \boldsymbol{s} \in \mathcal{U}\colon \rho(\boldsymbol{s}) = \mathrm{Vec}\left(\tilde{\eta}_1\rho_1(\boldsymbol{s}), \ldots, \tilde{\eta}_p\rho_p(\boldsymbol{s})\right).$$

D'après cette définition, pour toute fonction d'agrégation multiple ρ de $\mathcal{L}$,

$$\forall \boldsymbol{s}, \boldsymbol{t} \in \mathcal{U}, \langle(\rho\circ u)(\boldsymbol{s}) \mid (\rho\circ u)(\boldsymbol{t})\rangle_{\ell_2} = \sum_{r=1}^{p} \tilde{\eta}_r^2 \langle(\rho_r\circ u)(\boldsymbol{s}) \mid (\rho_r\circ u)(\boldsymbol{t})\rangle_{\ell_2}\,. \tag{3.22}$$

Cette relation fait le lien entre les fonctions d'agrégation multiples et le problème MKL, à l'instar de ce qui a déjà été vu. En conséquence, apprendre une fonction d'agrégation se réduit à apprendre un noyau multiple de la même façon que cela a été précédemment fait pour un BdF discriminant. Dans ce contexte, étant donnée une transformée TF u, la fonction de décision optimale f apprise conjointement à la fonction d'agrégation multiple ρ (de vecteur de poids $\tilde{\boldsymbol{\eta}} = \sqrt{\boldsymbol{\eta}}$) s'exprime par la formule :

$$\begin{aligned}\forall \boldsymbol{s} \in \mathcal{U}\colon f\left((\rho\circ u)(\boldsymbol{s})\right) &= \sum_{i=1}^{n} \alpha_i y_i k_{[\boldsymbol{\eta}]}\left((\rho_0\circ u)(\boldsymbol{s}), (\rho_0\circ u)(\boldsymbol{s_i})\right)\\ &= \sum_{i=1}^{n} \alpha_i y_i k\left((\rho\circ u)(\boldsymbol{s}), (\rho\circ u)(\boldsymbol{s_i})\right),\end{aligned}$$

où $\rho_0 = (\rho_r)_{1\leq r\leq p}$.

Apprendre la fonction d'agrégation paraît avantageux puisque cela est une étape de plus dans l'automatisation de la méthode. Ainsi, l'utilisateur n'a pas besoin de choisir une fonction d'agrégation particulière (en pratique, il est difficile d'avoir une intuition sur quelle fonction d'agrégation fonctionnera correctement). Néanmoins, il est nécessaire de rester prudent car ajouter de nouveaux degrés de liberté au schéma d'optimisation peut rendre la méthode sujette au sur-apprentissage. Pour cette raison, nous proposons une approche en trois étapes :

- ◇ tracer une grille linéaire des paramètres des RI et apprendre une fonction d'agrégation multiple avec le BdF ainsi construit ;
- ◇ appliquer l'algorithme 3 avec la fonction d'agrégation apprise à la première étape ;
- ◇ apprendre une nouvelle fonction d'agrégation avec le BdF discriminant appris à la précédente étape.

Une pré-étude (que nous ne rapportons pas ici) montre que cette approche est plus efficace que celle consistant à inclure l'apprentissage de la fonction d'agrégation à chaque itération de l'algorithme 3.

3.4.6 Relation avec l'état de l'art

Réseaux de neurones convolutifs

Considérer un BdF pour modéliser une représentation TF discriminante nous mène à un problème déjà abordé : les réseaux de neurones convolutifs (*Convolutional Neural Network*,

CNN) [LeCun *et coll.*, 1998]. Les CNN constituent l'état de l'art dans diverses applications de reconnaissance de formes, comme par exemple celle consistant à reconnaître des caractères manuscrits. L'efficace machine des CNN est construite à partir d'un ou plusieurs étages de BdF. Ceux-ci filtrent les signaux et réduisent leur dimension. Dans un CNN, chaque BdF contient une non-linéarité grâce à une *fonction d'activation*. Conjointement à leur création, les descripteurs sont utilisés pour apprendre un perceptron multi-couche (*Multi-Layer Perceptron*, MLP), qui est un classifieur non linéaire.

Comparée à un CNN, l'approche présentée ici apporte un nouveau regard sur le problème d'apprentissage d'un BdF discriminant. Une première différence est la nature des non-linéarités dans l'extraction des caractéristiques. Dans un CNN, on trouve une fonction sigmoïde à la sortie de chaque neurone, qu'il soit de convolution ou de décimation. En revanche, les BdF que nous construisons sont des applications linéaires. La non-linéarité réside uniquement dans la fonction d'agrégation. Dans ce contexte, une cascade de BdF peut être récrite comme un BdF à un seul étage ; c'est dans cette situation que nous nous plaçons.

Une deuxième différence, constituant en réalité le point fort de notre approche, est de construire un BdF à partir d'une méthode à noyaux, fournissant des classifieurs non-linéaires appris par optimisation convexe. Notons que notre problème d'apprentissage (3.1) pouvait être formulé différemment, par exemple grâce à la borne rayon-marge [Chapelle *et coll.*, 2002] ou à l'alignement de noyaux [Cortes *et coll.*, 2012], mais la façon dont nous avons formalisé le problème semble plus naturelle et plus efficace que les critères non-convexes décrit dans la section 1.4. Au contraire, le MLP présent dans un CNN est appris grâce à une descente de gradient sur un critère non-convexe. Les conséquences sont notables puisqu'il est souvent nécessaire de réaliser plusieurs apprentissages initialisés différemment (aléatoirement) afin d'obtenir un résultat satisfaisant. Comme nous l'avons expliqué au cours des précédentes sections, le schéma d'optimisation que nous proposons gère en partie sa non-convexité intrinsèque grâce à une étape interne de tirages aléatoires, ce qui le rend plus stable que d'initialiser aléatoirement diverses descentes de gradient.

En outre, notre approche se présente comme une réponse au principal défaut des CNN, qu'est le risque de sur-apprendre l'ensemble d'apprentissage, résultant en une incapacité à classer correctement des signaux inédits. Ce phénomène apparaît en premier lieu pour de petits ensembles d'apprentissage. Il est souvent observé pour des CNN (par exemple sur les données synthétiques étudiées en section 3.5.2) alors que notre méthode est supposée ne pas être sujette à cet écueil puisque construite sur une SVM. En effet, une SVM cherche à maximiser la marge entre les deux classes, ce que ne fait pas un MLP. De plus, le sur-apprentissage chez les CNN peut intervenir du fait de la forte complexité du modèle (il y a beaucoup de paramètres). Au contraire, puisque nos filtres sont contrôlés par peu de paramètres, notre méthode est en quelque sorte régularisée par la famille de filtres choisie et tend ainsi à prévenir le sur-apprentissage.

Enfin, la méthode que nous proposons est accompagnée de plusieurs autres avantages. Par exemple, puisque les gradients sont calculés par rapport aux poids de chaque filtre, il n'est pas nécessaire d'utiliser des opérateurs dérivables. En particulier, les fonctions d'agrégation comme celles calculant des maxima locaux peuvent être utilisées sans soucis. En outre, la méthode proposée est relativement bien automatisée et ne requiert pas une grande expérience pour être paramétrée, à la différence des CNN. Il n'est, par exemple, pas nécessaire de pré-définir le nombre de filtres (comme dans un CNN) car celui-ci est déterminé automatiquement au cours de l'apprentissage.

Apprentissage de noyaux infinis et d'ondelettes

IKL [Gehler et Nowozin, 2008a] et l'apprentissage de noyaux d'ondelettes (*Wavelet Kernel Learning*, WKL) [Yger et Rakotomamonjy, 2011] sont deux problèmes distincts. Alors

que le premier a pour but d'apprendre (par programmation linéaire semi-infinie) un noyau multiple sous la forme d'une combinaison convexe d'un nombre potentiellement infini de noyaux, le second apprend une combinaison d'un très grand nombre de noyaux construits sur des décompositions en ondelettes, et ce grâce à une méthode d'ensemble actif. En dépit de leurs différences, ces deux approches partagent des algorithmes relativement semblables, fondés sur une technique de génération de colonne couplée avec un MKL parcimonieux comme problème interne. La différence majeure réside dans la façon de générer une nouvelle colonne. Tandis qu'IKL essaie de résoudre un problème secondaire non-convexe, WKL échantillonne aléatoirement plusieurs noyaux jusqu'à en trouver un qui contredit les conditions d'optimalité.

Le travail présenté ici est algorithmiquement inspiré de ces deux contributions puisque nous cherchons à apprendre une combinaison non-linéaire d'un nombre potentiellement infini de noyaux ($\approx$ IKL) grâce à une méthode d'ensemble actif ($\approx$ WKL). Pourtant, notre approche diffère d'IKL et de WKL dans le but : notre objectif est en premier lieu d'apprendre une représentation TF discriminante, conjointement à un classifieur SVM. De plus, notre approche a été construite de sorte que l'outil de classification appris puisse se réduire facilement à deux étapes : d'abord l'analyse des signaux grâce à un BdF, puis une SVM. Cette réduction n'est pas possible avec WKL, pouvant ainsi conduite à une difficulté quant à l'interprétation des outils appris. Notons aussi que notre algorithme s'avère être une extension de la version non-linéaire du paradigme MKL de [Varma et Babu, 2009] à une infinité de noyaux. Une solution de ce problème ne peut certainement pas être approchée ni par IKL ni par WKL. Enfin, même s'il n'y a aucune preuve de convergence, nous avons montré la stricte décroissance de la valeur de la fonction objectif de notre problème au fil des itérations, et ce malgré la non-convexité du problème MKL interne lorsque le noyau est gaussien.

3.5 EXPÉRIENCES NUMÉRIQUES

Cette section a pour but de démontrer empiriquement l'intérêt de l'apprentissage de BdF engendrés par une famille génératrice de filtres. Dans un premier temps, nous donnons quelques exemples de paramétrisation des RI dont les paramètres peuvent être appris grâce à notre méthode. Ensuite, nous évaluons notre approche sur un problème synthétique ainsi que deux applications réelles : l'une liée aux Interfaces Cerveau-Machine (ICM) et l'autre à la classification de scènes audio.

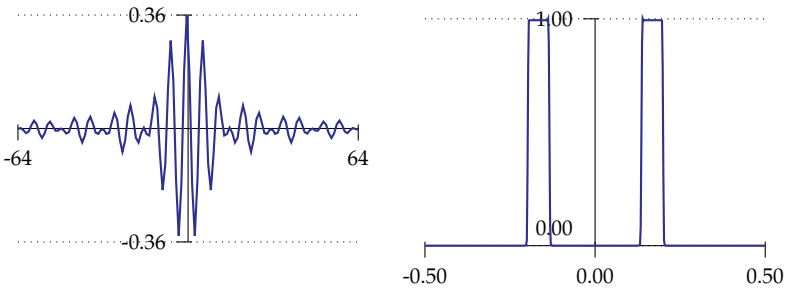
3.5.1 Paramétrisation des méthodes

La première paramétrisation des RI que nous considérons est celle d'un filtre passe-bande créé par la méthode de la fenêtre. Soit ω_{on} et ω_{off} les pulsations réduites du filtre ($0 \leq \omega_{\text{on}}, \omega_{\text{off}} \leq \pi$). Alors $\boldsymbol{\theta} = (\omega_{\text{on}}, \omega_{\text{off}})$ et $\mathcal{P} = [0, \pi]^2$. La RI du filtre (de longueur q) est ainsi :

$$\forall (\omega_{\text{on}}, \omega_{\text{off}}) \in \mathcal{P},\ \forall t \in \mathbb{N}_q :$$
$$\boldsymbol{h}_{(\boldsymbol{\omega}_{\text{on}}, \boldsymbol{\omega}_{\text{off}})}(t) = \frac{\sin(\omega_{\text{off}} t - \omega_{\text{off}} \frac{q}{2}) - \sin(\omega_{\text{on}} t - \omega_{\text{on}} \frac{q}{2})}{t} \frac{W(t)}{\nu},$$

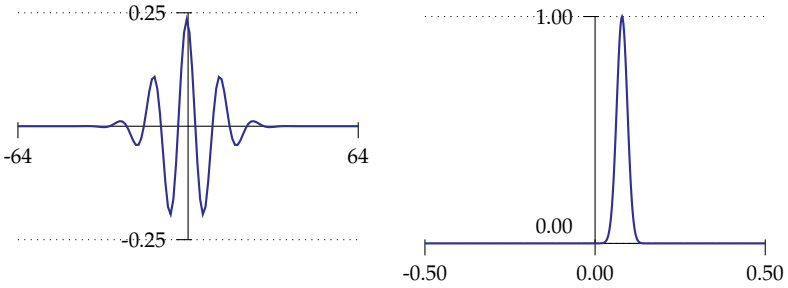
où W est une fonction d'apodisation (par exemple la fonction de Hanning ou celle de Blackman) et ν est un facteur de normalisation ayant pour but de rendre unitaire l'énergie du filtre. La figure 3.8 page suivante illustre un tel filtre.

Une autre option est l'ondelette de Morlet. Elle consiste en une sinusoïde à valeur complexe modulée par une fenêtre gaussienne de paramètre σ. Le facteur d'enveloppe σ contrôle le



(a) Représentation temporelle. (b) Représentation spectrale.

FIGURE 3.8 – Filtre passe-bande.



(a) Représentation temporelle de la partie réelle. (b) Représentation spectrale.

FIGURE 3.9 – Ondelette de Morlet.

nombre d'oscillations de l'onde au sein du paquet. Le vecteur paramètre est $\boldsymbol{\theta} = (\tau, \sigma)$ et vit dans $\mathcal{P} = [1, \beta_\tau] \times [1, \beta_\sigma]$ (où β_τ et β_σ sont des bornes supérieures, par exemple 50). La formule donnant la RI d'une ondelette de Morlet est :

$$\forall (\tau, \sigma) \in \mathcal{P},\ \forall t \in \mathbb{N}_q :$$
$$\boldsymbol{h}_{(\tau,\sigma)}(t) = \mathrm{e}^{-8\left(\frac{\pi\tau(t-\frac{q}{2})}{\sigma q}\right)^2} \left(\mathrm{e}^{4\imath\pi\frac{\tau(t-\frac{q}{2})}{n}} - \mathrm{e}^{-\sigma^2/2}\right)\frac{1}{\nu},$$

où ν est toujours un facteur de normalisation permettant d'obtenir une énergie unitaire. Utiliser un filtre à valeur complexe conjointement à une fonction d'agrégation par diffusion est particulièrement intéressant puisque comme l'ondelette de Morlet est quasiment analytique pour $\sigma > 5$ (*i.e.* sa transformée de Fourier est presque nulle pour les pulsations négatives), le BdF ainsi construit est proche d'une transformée par diffusion d'ondelettes de premier ordre [Andén et Mallat, 2014]. Ce type de transformée aboutit à de bonnes performances de reconnaissance de signaux audio. La figure 3.9 illustre un filtre de Morlet.

Enfin, les méthodes que nous allons confronter dans cette section sont résumées dans le tableau 3.1. La majorité d'entre elles est construite sur un classifieur SVM. La méthode surnommée *SVM* est l'approche naïve considérant les séries temporelles comme vecteurs caractéristiques. *Max* améliore *SVM* en agrégeant les séries ; l'agrégation est réalisée en calculant des maxima locaux. À la différence de ces deux approches, *DFT* utilise l'amplitude des décompositions de Fourier des signaux comme vecteurs caractéristiques, et *MFCC* leur coefficients cepstraux dans le plan Mel-Fréquence. Ces quatre méthodes sont des points de comparaison dans notre étude.

Nous considérons aussi des méthodes avancées comme les CNN [LeCun *et coll.*, 1998] (*CNN*) et WKL [Yger et Rakotomamonjy, 2011] (*WKL*). Dans toute notre étude, un CNN possède une unique couche de convolution avec trois filtres. Le facteur de décimation de

ABRÉVIATION	ANALYSE	AGRÉGATION	CLASSIFIEUR
SVM	-	-	SVM
Max	-	Max	SVM
DFT	Transformée de Fourier discrète	-	SVM
MFCC	Cepstre Mel-Frequence	-	SVM
CNN	Réseau de neurones convolutifs	Moyenne	MLP
WKL	Transformée en ondelettes	Norme ℓ_2	MKL
Band-Max	Banc **appris** de filtres passe-bandes	Max	SVM
Band-ℓ_2	Banc **appris** de filtres passe-bandes	Norme ℓ_2	SVM
Morlet-Pooling	Banc **appris** d'ondelettes de Morlet	Diffusion	SVM
Band-Pooling	Banc **appris** de filtres passe-bandes	**Appris**	SVM
Morlet-Pooling	Banc **appris** d'ondelettes de Morlet	**Appris**	SVM

TABLE 3.1 – Méthodes mises en jeu (les variantes de celle que nous proposons sont dans la deuxième partie du tableau).

la couche de sous-échantillonnage est identique à celui des fonctions d'agrégation Max et Moyenne associées à notre approche d'apprentissage de BdF. De la sorte, la structure du CNN est comparable à celle de nos BdF. Enfin, les méthodes que nous proposons ici sont appelées *Band-Max*, *Band-*ℓ_2 and *Morlet-Scattering*, selon la famille de filtre (passe-bande ou ondelette de Morlet) et la sorte d'agrégation (voir tableau 2.1).

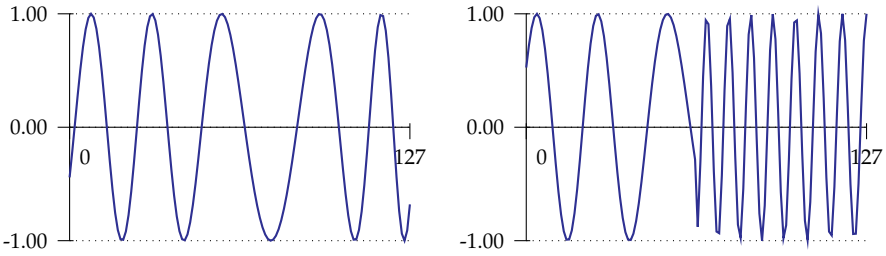
Pour les expériences numériques sur données réelles, (ICM et classification de scènes), nous présentons nos méthodes sans et avec apprentissage de la fonction d'agrégation (voir section 3.4.5). Dans ce dernier cas, notre approche se nomme *Band-Pooling* ou *Morlet-Pooling*, selon la famille de filtres, et la fonction d'agrégation multiple est générée à partir de fonctions Max, Moyenne et Diffusion avec différentes largeurs de fenêtre, ainsi que des normes ℓ_1 et ℓ_2. Enfin, l'apprentissage de la fonction d'agrégation est initialisé avec un vecteur de poids uniforme (aucune information *a priori* n'est intégrée).

Les sections suivantes détaillent les trois jeux de données sur lesquels nous avons évalué notre approche ainsi que les résultats numériques qui y sont associés.

3.5.2 Données synthétiques

Le premier jeu de données utilisé est constitué de deux classes de signaux synthétisés sur la base d'une forme générative par classe (figure 3.10 page suivante). La première forme est une sinusoïde de pulsation réduite variant légèrement autour de 0.039. La seconde forme est identique sur la première moitié puis est constituée d'une sinusoïde de plus haute fréquence sur la deuxième moité. La pulsation réduite de cette deuxième sinusoïde varie légèrement autour de 0.117. Les pulsations variables de ces deux formes sont une première sorte de variabilités intra-classes qui caractérise le jeu de données. La seconde sorte de variabilités est une translation temporelle aléatoire des signaux, de manière qu'une distance euclidienne soit inefficace pour quantifier la proximité physique de deux entrées. Enfin, un bruit coloré gaussien est ajouté à chaque signal. Celui-ci provient d'un bruit blanc gaussien convolué avec un filtre choisi au hasard parmi les RI $[1, -2, 1]$, $[0, 1, -1]$ et $[1, 0, 1]$. Le bruit ainsi construit est stationnaire à l'échelle d'un signal mais non-stationnaire à celle du jeu de données.

Évidemment, le jeu de données a été simulé de manière à mettre en valeur les qualités propres de notre approche par rapport au concurrents (tableau 3.1). En effet, les courbes de la figure 3.11 montrent que le module de la transformée de Fourier (*DFT*) n'est pas un extracteur de caractéristiques pertinent pour ce problème. De plus, la représentation



(a) Classe 1. (b) Classe 2.

FIGURE 3.10 – Données jouet.

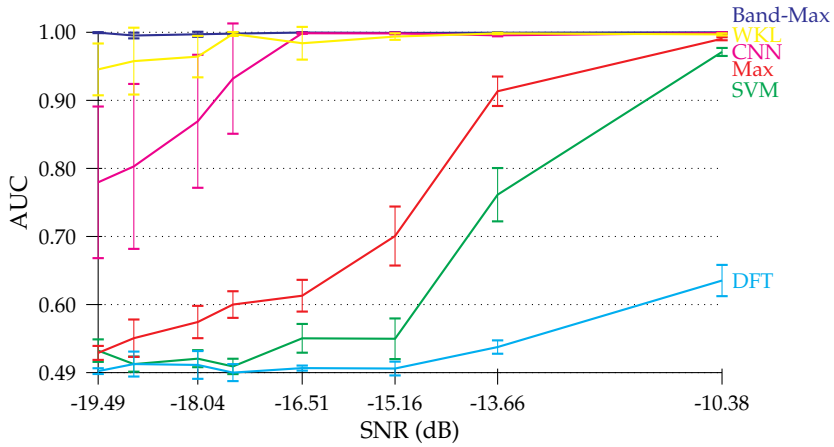
de Shannon (*SVM*) éprouve des difficultés à gérer le bruit coloré ainsi que les translations temporelles. Heureusement, l'utilisation d'une fonction d'agrégation suite à la représentation de Shannon (*Max*) permet d'améliorer la reconnaissance des signaux. Le CNN, quant à lui, tend à sur-apprendre l'ensemble d'apprentissage lorsque celui-ci est de petite taille ou lorsque le SNR est faible. La méthode d'Yger et Rakotomamonjy, WKL, est adaptée à cette tâche de classification mais ne semble pas être en mesure de maintenir ses résultats en présence d'un bruit important. En comparaison à toutes ces méthodes, l'approche proposée dans la section précédente paramétrée par de simples filtres passe-bandes et une fonction d'agrégation par maxima locaux (*Band-Max*) est clairement supérieure. En effet, le filtrage et l'agrégation absorbent le bruit tout comme les variabilités intra-classes.

La figure 3.12 dépeint un exemple de BdF appris sur ce jeu de données. Le principal filtre discriminant est centré autour de la pulsation réduite 0.117, qui est bien la caractéristique décisive pour séparer les deux classes de signaux.

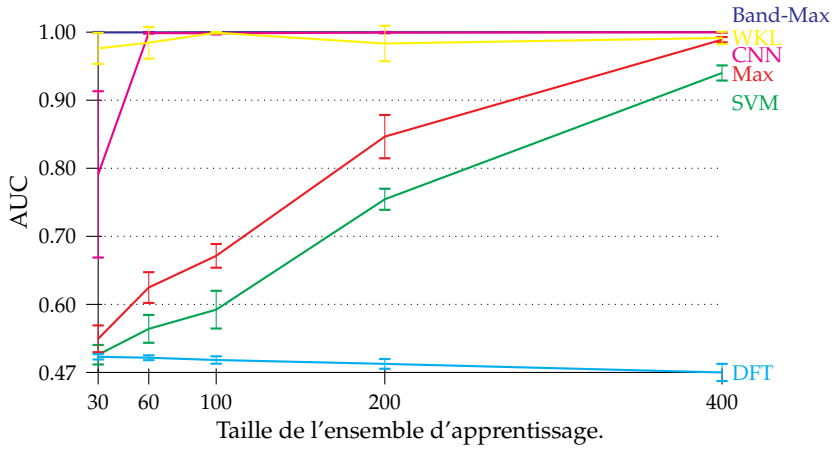
3.5.3 Problème d'interface cerveau-machine

À l'instar de [Yger et Rakotomamonjy, 2011], le premier jeu de signaux réels que nous considérons est un problème ICM. L'activité électro-encéphalographique d'un homme sain a été enregistrée à l'aide d'un casque contenant 32 électrodes en étain, placées aux endroits usuels du scalpe. Lors des enregistrements, le sujet a effectué des flexions plantaires imaginaires du pied droit, soit le plus vite possible, soit en suivant un mouvement continu de 4 secondes. Les deux protocoles (flexions lentes et rapides) génèrent différents potentiels corticaux liés aux mouvements (*movement-related cortical potentials*) que nous souhaitons distinguer. Nous sommes donc en présence d'un problème de classification binaire de signaux ICM. L'ensemble d'apprentissage contient 75 signaux (de taille 512) de chaque classe (autant que de flexions imaginaires effectuées par le sujet), normalisés de sorte que l'amplitude maximale (en valeur absolue) atteinte sur l'ensemble soit unitaire. Pour les besoins de l'étude, les résultats de classification sont donnés sous forme de statistiques construites sur 10 essais, pour lesquels l'ensemble des signaux est aléatoirement scindé en deux groupes sans recouvrement. L'un est réservé à l'apprentissage (70%), l'autre à l'évaluation (30%). Pour conclure sur le protocole expérimental, les paramètres inconnus (C et γ) sont déterminés par une validation croisée en 5 étapes.

Les expériences numériques ont été conduites sur les canaux 9, 12, 17, 29 et 30, comme dans [Yger et Rakotomamonjy, 2011]. La figure 3.13 page 92 décrit les résultats obtenus avec un noyau SVM gaussien uniquement, puisque ceux-ci sont meilleurs que les résultats du noyau linéaire. Les boîtes à moustaches sur cette figure montrent qu'un banc d'ondelettes de Morlet de concert à une agrégation par diffusion (*Morlet-Scattering*) atteint des taux de classification comparables mais légèrement supérieurs à ceux d'un banc de filtres



(a) Par rapport au SNR.



(b) Par rapport à la taille de l'ensemble d'apprentissage.

FIGURE 3.11 – Taux de classification sur les données jouet.

passe-bandes simples associé à une agrégation par maxima locaux (*Band-Max*), et nettement meilleurs qu'un même BdF associé, cette fois, à une norme ℓ_2 (*Band-*ℓ_2), qui agrège toute l'information temporelle en fournissant la distribution marginale des fréquences. L'échec de cette dernière approche met en évidence la nécessité d'une représentation TF (plutôt qu'une représentation purement fréquentielle) pour distinguer les signaux ICM de cette expérience.

Sur cet exemple numérique, les différentes variantes de notre approche sont systématiquement plus à même de séparer les deux classes de signaux que les méthodes concurrentes, comme WKL. Pour celle-ci, nous avons utilisé la même configuration que celle décrite dans l'article qui introduit WKL [Yger et Rakotomamonjy, 2011] (approche pleinement stochastique avec des ondelettes de Daubechies à 6 moments nuls et un noyau gaussien marginal, configuration qui a montré de meilleures performances que celle sans agrégation des décompositions en ondelettes) et retrouvé des résultats similaires, même si les données n'ont pas été normalisées de la même manière (moyenne nulle et écart type unitaire de chaque variable explicative dans [Yger et Rakotomamonjy, 2011] et amplitude unitaire à travers l'ensemble d'apprentissage ici). La supériorité de notre approche par rapport à WKL peut s'expliquer de manière intuitive suivant trois directions :

- ◇ notre approche permet de considérer différentes familles de filtres et manières d'agréger les représentations TF ;
- ◇ le paramètre du noyau gaussien γ est fixé à $\frac{1}{2}$ dans WKL alors qu'il est déterminé par

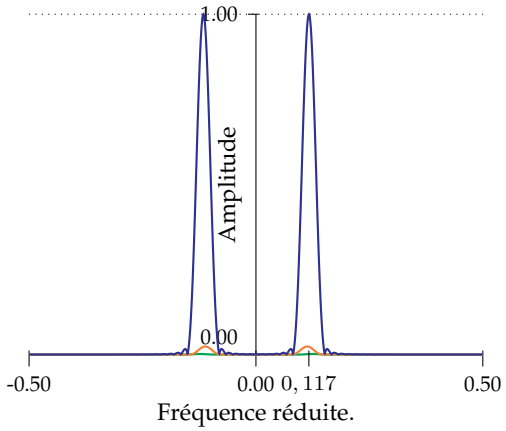


FIGURE 3.12 – Exemple de BdF appris sur le jeu de données jouet (chaque couleur incarne un filtre dans le domaine spectral).

validation croisée dans notre cas ;

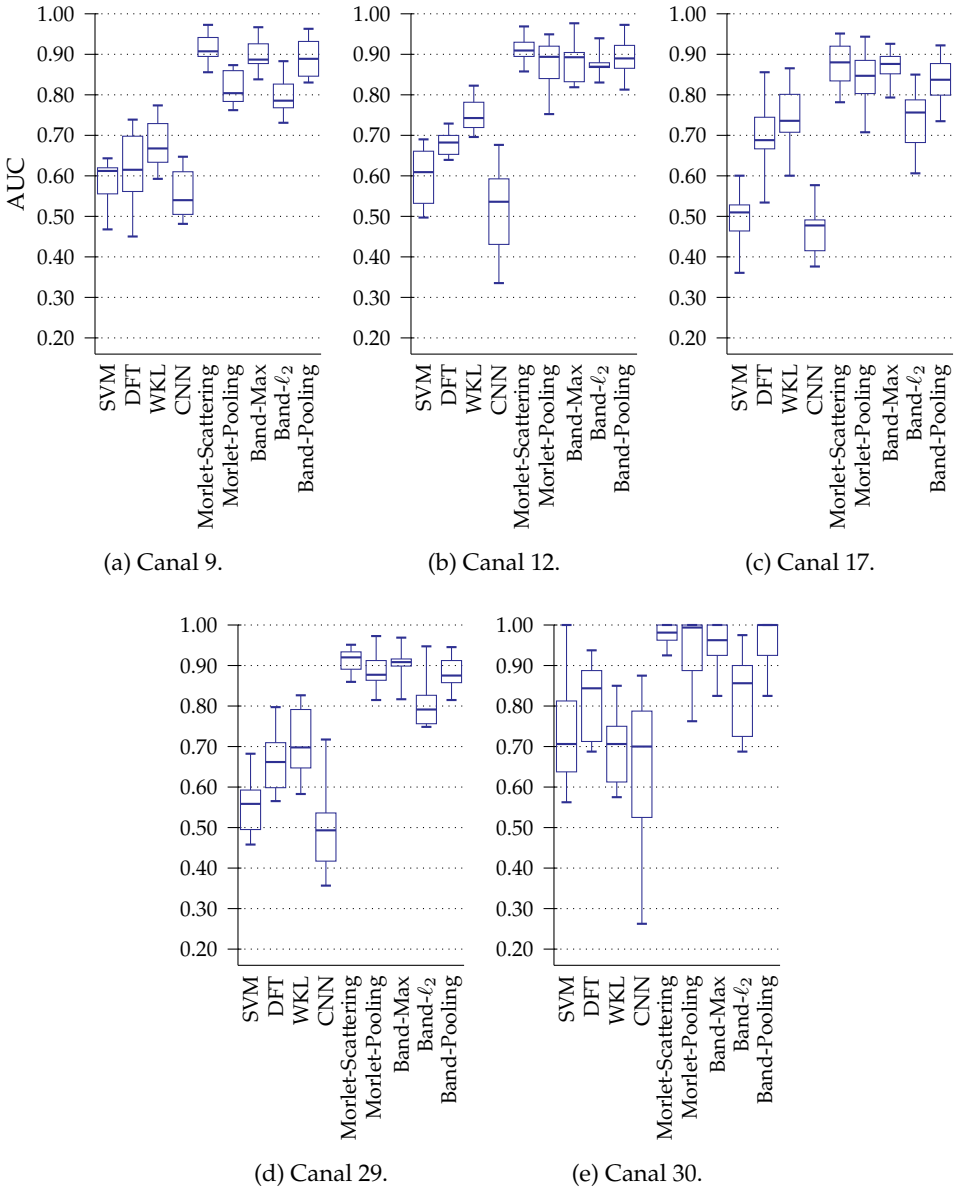
- ◇ notre approche munie d'un noyau gaussien nécessite l'utilisation d'une combinaison non-linéaire de noyaux SVM, qui s'est révélée plus efficace qu'une combinaison linéaire dans certaines situations [Varma et Babu, 2009] ; tandis que WKL utilise systématiquement un noyau multiple linéaire.

Notre approche semble aussi plus adaptée qu'un CNN pour ce jeu de données. En effet, le CNN montre de piètres performances de reconnaissance, probablement à cause du haut niveau de bruit dans les signaux ICM et de la difficulté à l'entraîner. Enfin, les approches naïves, considérant les représentations temporelles comme vecteurs caractéristiques (*SVM*) ou les modules de la transformée de Fourier (*DFT*), échouent à généraliser la règle de décision apprise à des données inédites. Ceci met en avant l'intérêt des approches dirigées par les données pour la reconnaissance de signaux.

Mises à part quelques exceptions, l'apprentissage de la fonction d'agrégation (méthodes *Band-Pooling* et *Morlet-Pooling*) donne des résultats similaires (et même meilleurs pour le canal 30) à ceux de la meilleure fonction d'agrégation choisie manuellement. Ceci nous conduit à remarquer que l'utilisateur n'a pas particulièrement besoin d'une intuition lui indiquant quelle fonction d'agrégation choisir, ou d'en essayer plusieurs indépendamment. L'apprentissage de la fonction d'agrégation permet d'automatiser notre méthode en donnant des résultats comparables à des choix astucieux.

3.5.4 Scènes acoustiques

La seconde expérience sur signaux réels concerne l'analyse de scènes acoustiques, domaine de l'informatique qui cherche à imiter la capacité des humains à suivre des sources sonores spécifiques dans un environnement audio complexe [Giannoulis *et coll.*, 2013]. Deux sortes de problèmes existent dans ce type d'analyse : la détection d'événements acoustiques et la classification de scènes acoustiques, aussi appelées *paysages sonores* (cette expression a été introduite par le compositeur canadien R. Murray Schafer en 1977 [Schafer, 1977]). En ce qui nous concerne, nous nous intéressons seulement au problème de classification de scènes audio. Le but est alors de caractériser l'environnement d'un enregistrement sonore en lui attribuant une étiquette sémantique. Par exemple, la base d'enregistrements que nous utilisons est constituée de 10 classes [Giannoulis *et coll.*, 2013] : rue agitée, rue calme, parc, marché en plein air, supermarché, bus, métropolitain, station de métropolitain, restaurant et bureau. Pour chaque scène, 10 séquences de 30 secondes ont été enregistrées par trois



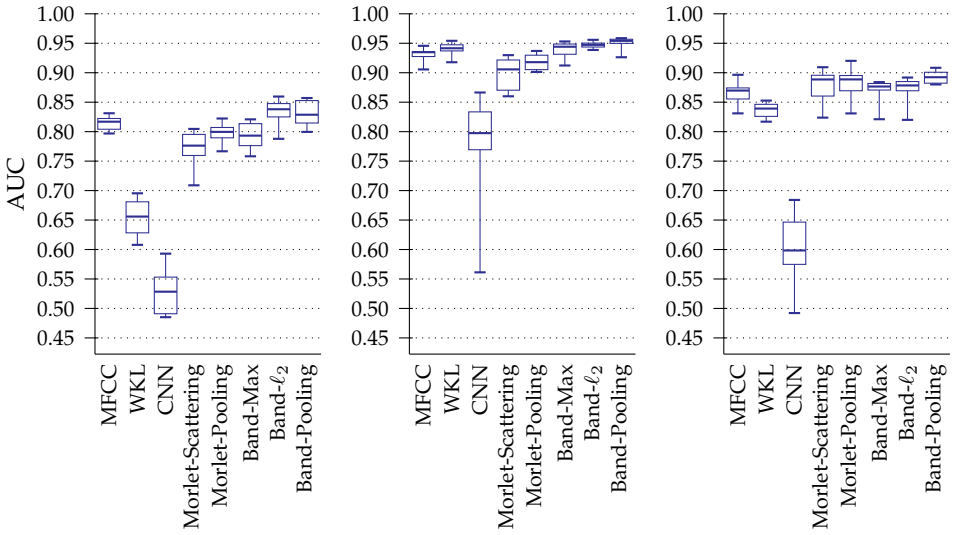
(a) Canal 9. (b) Canal 12. (c) Canal 17.

(d) Canal 29. (e) Canal 30.

FIGURE 3.13 – Taux de classification pour l'expérience ICM.

personnes différentes dans divers endroits du Grand Londres. Ces enregistrements se sont étalés sur une période de plusieurs mois durant l'été et l'automne 2012, avec des conditions météorologiques variables et à des horaires de la journée différents.

Comme nous l'avons mentionné dans le chapitre 2, il est nécessaire de traiter ces (longues) séquences sous la forme de (sous-)signaux extraits de celles-ci. Idéalement, on se placerait dans un schéma démocratique mais pour simplifier le problème de classification, nous décidons de nous arrêter avant l'étape de vote des signaux. En pratique, chaque séquence est découpée en signaux de 300 millisecondes, qui sont filtrés et décimés de manière à réduire leur taille, puis chacun des signaux prend l'étiquette de la séquence dont il provient. On obtient donc une nouvelle base comportant 100 fois plus de données. Le reste du protocole expérimental est semblable à celui mis en place pour le problème ICM : les séquences sont partagées aléatoirement en deux groupes dont l'un fournit des signaux d'apprentissage et l'autre d'évaluation. La normalisation est réalisée de sorte que l'amplitude maximale (en valeur absolue) atteinte sur l'ensemble d'apprentissage est unitaire et 10 essais sont réalisés pour obtenir des statistiques sur les taux de classification. Enfin, nous ne traitons que



(a) Parc vs rue calme. (b) Bureau vs super-marché. (c) Bureau vs rue calme.

FIGURE 3.14 – Taux de reconnaissance des scènes acoustiques.

des problèmes de classification binaires, comme par exemple métropolitain vs station de métropolitain.

Les résultats de cette expérience numérique sont indiqués sur la figure 3.14. À l'instar du problème ICM, seul les meilleurs résultats (ceux du noyau gaussien, plutôt que linéaire) sont exposés. Ici, la méthode de référence est une SVM appliquée aux coefficients cepstraux mel-fréquences (*Mel-Frequency Cepstral Coefficients*, MFCC) calculés sur la totalité des 300 millisecondes des signaux. Cette approche, qui fait référence pour la classification de signaux audio, semble particulièrement efficace pour certains des problèmes binaires étudiés dans cette section (par exemple bureau vs super-marché). Pourtant, même dans ces cas favorables, il existe toujours une variante de notre approche (soit avec des filtres passe-bandes simples - *Band-Max* et *Band-ℓ_2* - soit avec des ondelettes de Morlet - *Morlet-Scattering*) qui améliore les taux de classification obtenus avec les *MFCC*.

La méthode avancée qu'est WKL est généralement efficace mais moins précise que notre approche quant à la reconnaissance. Une fois de plus, ceci peut être expliqué intuitivement par la modularité de la méthode que nous proposons, qui est capable de travailler avec plusieurs fonctions d'agrégation conjointement à des ondelettes ou à des filtres de natures différentes. À l'opposé de cela, WKL apprend une décomposition en ondelettes associée à une agrégation marginale en norme ℓ_2.

De manière identique à l'expérience ICM, le CNN montre des résultats très faibles, qui sont difficiles à expliquer. De plus, ces simulations numériques confirment que l'apprentissage de la fonction d'agrégation reste une alternative efficace au choix manuel de celle-ci, et améliore même les taux de classification.

3.6 SYNTHÈSE

Ce chapitre a présenté notre première contribution. Celle-ci réside principalement dans la mise en place d'un algorithme d'apprentissage d'une représentation TF (implémentée sous la forme d'un BdF) conjointement à une SVM. Il est apparu que l'apprentissage d'un tel modèle sans contraintes particulières sur les filtres fournit certes un apport comparé à l'extraction purement manuelle de caractéristiques mais souffre de sur-apprentissage lorsque

l'on dispose de peu de données ou que les signaux sont fortement bruités.

Pour cette raison, l'algorithme que nous avons décrit dans ce chapitre suppose que l'on se restreigne à une (ou plusieurs) famille(s) de filtres. Cette forme de régularisation permet d'éviter l'écueil du sur-apprentissage et conduit à la mise en place d'un algorithme d'apprentissage de noyaux multiples, étendant l'état de l'art dans le domaine en autorisant à combiner non-linéairement des noyaux choisis parmi un ensemble infini. De manière intéressante, il est aussi possible d'automatiser le choix de la fonction d'agrégation comme concaténation de plusieurs fonctions. Cet attribut s'ajoute au choix automatique du nombre de filtres constituant le banc et facilite l'utilisation de notre méthode.

Une dernière section d'expériences numériques a montré l'intérêt de notre approche sur des signaux ICM ainsi que sur des paysages sonores. Pour ces derniers, le gain de reconnaissance s'avère toutefois léger (en particulier comparé à celui obtenu avec des MFCC) au regard de la complexité de la méthode mise en œuvre.

Au fil des chapitres précédents, nous avons rencontré trois structures pyramidales. La première est le CNN (figure 2.5 page 45). La seconde est celle permettant d'implémenter une transformée en ondelettes rapide (figure 2.6 page 48). Enfin, la troisième est la décomposition par diffusion d'ondelettes (figure 2.8 page 50). Comme nous l'avons vu, la transformée en ondelettes rapide peut être modélisée par un BdF à un seul étage car elle est composée d'une cascade d'opérations linéaires. Cette configuration est identique à celle du BdF qu'il est possible d'apprendre grâce à notre approche. En revanche, cette réduction n'est pas envisageable pour le CNN et la diffusion d'ondelettes car des opérateurs d'agrégation non-linéaires sont présents à plusieurs points internes à leur structure.

En ce sens, le BdF que nous apprenons (composé d'un étage d'opérations de filtrage puis d'une agrégation non-linéaire), ne constitue que la décomposition de premier ordre d'une structure pyramidale telle qu'un CNN ou une transformée en diffusion d'ondelettes. Or de récents travaux montrent l'intérêt de capturer l'information des ordres supérieurs [Bengio, 2009, Andén et Mallat, 2014, Bruna et Mallat, 2013], que ce soit pour l'un ou l'autre de ces modèles. Ainsi, une extension naturelle de notre approche consisterait à appliquer une cascade de BdF et d'agrégations non-linéaires à un signal, puis d'apprendre des filtres discriminants conjointement à une SVM. À l'heure actuelle, la principale limitation à ces prochains travaux est la complexité temporelle de notre algorithme. Comme toute technique d'apprentissage de noyau, notre approche est plus longue à entraîner qu'une SVM classique et nécessite elle aussi une étape de validation croisée afin de déterminer le coefficient de coût C et le paramètre des noyaux gaussiens γ, multipliant encore les besoins en ressources calculatoires. [1] Par conséquent, la mise en place de cette extension nécessite des solutions à chercher par exemple du côté du calcul parallèle.

1. À l'instar de [Gehler et Nowozin, 2008a, Sangnier *et coll.*, 2014] il aurait été possible d'inclure γ dans les paramètres du noyau à apprendre. Cependant, il serait nécessaire de comparer cette approche à une validation croisée (comme celle réalisée pour ce manuscrit) afin d'évaluer le risque de sur-apprentissage dû à la détermination automatique de γ en comparaison au gain de temps par rapport au choix par validation croisée.

CHAPITRE 4

Un modèle de détecteur précoce

4.1 INTRODUCTION

Lorsqu'il s'agit de reconnaître une série temporelle, deux questions viennent à l'esprit :

- ◇ quels descripteurs utiliser ?
- ◇ comment prendre le temps en compte ?

Le temps est une composante primordiale qu'on aurait tort d'assimiler à une simple dimension supplémentaire, et ceci pour les raisons déjà évoquées : la prise en compte du temps permet souvent d'augmenter la puissance de discrimination des outils mis en place. Toutefois, celui-ci est aussi source de redondance de l'information, voire de variabilité intra-classe. Dans ce dernier cas, il est donc soit nécessaire d'augmenter le nombre de signaux observés destinés à l'apprentissage, soit de réduire la résolution temporelle. Puisqu'il est généralement difficile d'acquérir et de prendre en compte (d'un point de vue informatique) toujours plus de signaux, l'outil que nous avons mis en place dans le chapitre 3 se concentre sur la deuxième solution. En suivant un schéma démocratique de traitement des séries temporelles (figure 2.1 page 33), l'approche que nous avons présentée est capable de donner une réponse aux deux questions, portant sur les descripteurs et le temps.

Il reste toutefois un point important qui n'a pas été abordé dans le chapitre 3 : celui de la redondance d'information dans le temps. Autrement dit : est-il nécessaire d'observer une séquence dans sa totalité pour prendre une décision ? À défaut de déterminer les caractéristiques de la redondance, il est néanmoins intéressant d'analyser le compromis entre la ***puissance de discrimination*** d'un outil et la ***précocité*** de la prise de décision. Cette question (seule, sans détermination automatique des descripteurs) est abordée ici à travers le problème générique de la détection précoce (au sein d'un schéma séquentiel, conformément à la figure 2.1 page 33).

Le but de ce chapitre est donc de construire un outil de détection précoce, adapté à toute série temporelle (audio, vidéo, *etc.*). Pour ce faire, nous laissons l'expert choisir des descripteurs appropriés et mettons en place le cadre qui suit, permettant une prise de décision au plus tôt (section 4.2). Nous faisons le pari qu'il est possible de détecter l'apparition d'un événement uniquement à partir de la connaissance de quelques *instantanés* discriminants (par exemple, des poses particulières dans la reconnaissance d'actions vidéo ou de brefs sons dans l'identification de paysages sonores). Ce choix, qui nous rapproche du concept d'apprentissage d'instances multiples (*Multiple Instance Learning*, MIL), nous

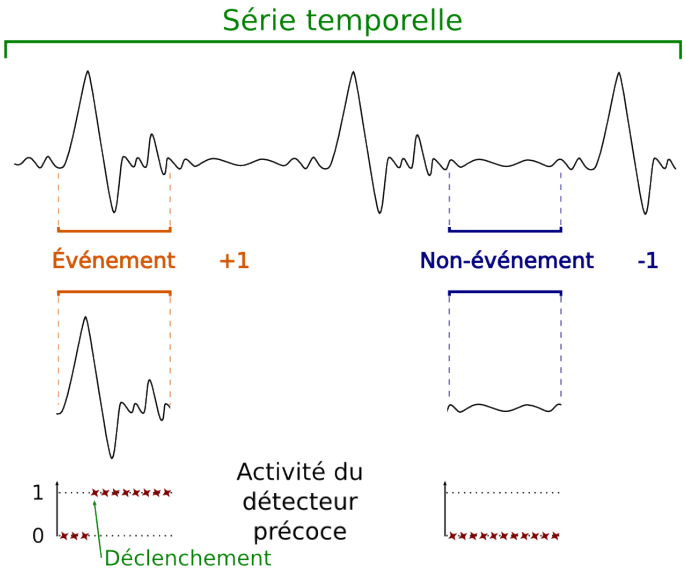


FIGURE 4.1 – Illustration de l'activité souhaitée du détecteur. Celui-ci est supposé se déclencher au plus tôt sur un événement et persister lorsqu'une notification de détection a été émise.

conduit à étendre les espaces de similarité proposés dans [Balcan et Blum, 2006, Pekalska et Duin, 2008] à la gestion des séries temporelles, tout en laissant une place primordiale au temps. À partir de ce cadre, il est alors possible d'énoncer des conditions faibles permettant d'aboutir à un détecteur ***fiable***, *i.e.* ne changeant pas d'avis après avoir notifié une détection. Ceci est une condition essentielle afin de prendre une décision à partir d'une information partielle.

Les conditions faibles que nous imposons à notre détecteur conduisent à mettre en place un problème d'apprentissage simple, ainsi qu'un algorithme efficace pour le résoudre (section 4.3). Au delà de l'aspect numérique, le détecteur que nous proposons bénéficie aussi de garanties théoriques, issues directement des travaux sur les séparateurs linéaires [Kakade *et coll.*, 2009]. Enfin, avant de valider empiriquement notre approche (section 4.5), les proximités et différences de celle-ci avec les modèles existants sont discutées dans la section 4.4.

4.2 DÉTECTION PRÉCOCE

Il existe plusieurs définitions de la détection d'événements (et *a fortiori* de la détection précoce). L'une d'entre elles consiste à *localiser* un événement (supposé unique) au sein de longues séquences [Hoai et De la Torre, 2014]. Un tel détecteur renvoie donc un intervalle de temps indiquant la position d'un événement dans une séquence (intervalle orange sur la figure 4.1). Réciproquement, un ensemble d'apprentissage adéquat contient des séquences (ou séries temporelles) associées chacune à un intervalle précisant la position de l'événement. La tâche accomplie par ce type de détecteur est double puisqu'elle requiert à la fois de détecter la présence d'un événement, et de déterminer précisément son commencement et sa fin. Plus le commencement annoncé par le détecteur est éloigné du commencement réel, plus la latence du système est grande. Un détecteur précoce cherche à diminuer cette latence tout en conservant de bonnes aptitudes à la détection et à la localisation.

Notre travail est construit sur une définition différente de la détection, ne comprenant pas la notion de localisation. En reprenant l'image 4.1, notre base d'apprentissage est consti-

tuée d'événements (en pratique, étiquetés $+1$) et de *non-événements* (*i.e.* des séquences qu'il n'est pas nécessaire de détecter, étiquetées -1). Ayant extrait ces séquences étiquetées, nous cherchons à apprendre un détecteur capable de donner un avis (potentiellement neutre) à chaque instant t d'une série temporelle. Par conséquent, lorsque notre détecteur est évalué sur un événement, il est supposé se déclencher au cours de l'analyse de celui-ci (évidemment avant la fin) pour envoyer une notification de détection. Le moment auquel le détecteur se déclenche indique sa latence et inversement sa précocité. De plus, pour que la décision prise par le détecteur soit interprétable par un expert, il est nécessaire que celle-ci ne varie pas, autrement dit que lorsqu'une notification de détection est émise, celle-ci reste active jusqu'à la fin de la séquence. Cet aspect correspond à la *fiabilité* du détecteur.

Afin d'atteindre cet objectif, nous proposons la chaîne de traitement suivante (dont les éléments clefs sont détaillés dans la prochaine section) :

1. chaque séquence (événements et non-événements) est associée à une représentation Temps-Caractéristique (TC), permettant de nous affranchir de la nature des séries temporelles (audio, vidéo, *etc.*), tout en favorisant la discrimination ;
2. les représentations TC sont ensuite redécrites dans un espace de similarité, par rapport à des objets discriminants (ici des *instantanés*) ;
3. dans cet espace de similarité, on réalise conjointement l'apprentissage du détecteur d'événements et la sélection des instantanés les plus discriminants.

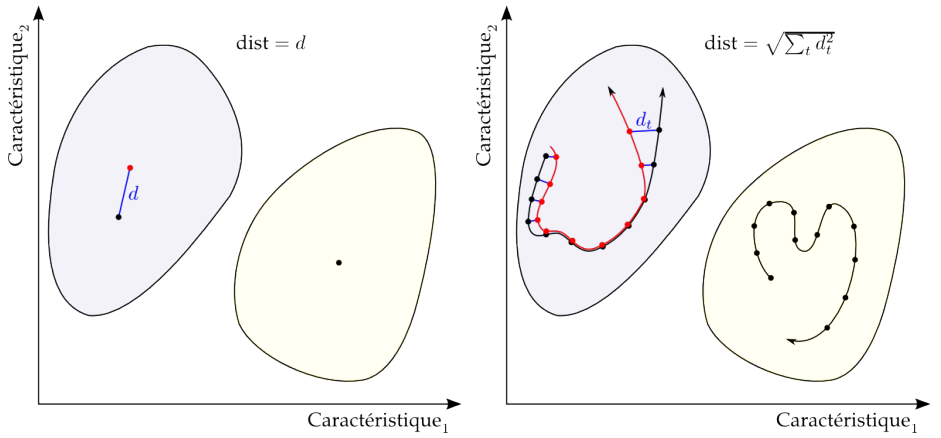
4.2.1 Espace de similarité

Commençons par appeler $\mathcal{Z}$ l'espace des séries temporelles observées (supposées continues) et T un majorant du temps. $\mathcal{X}^{[0,T]}$ désigne donc l'ensemble des applications de $[0, T]$ dans l'espace de caractéristiques discriminantes $\mathcal{X}$. De plus, pour alléger les notations, une fonction dépendante du temps (par exemple f) sera notée $f_\sim$ et son évaluation au temps t sera notée f_t.

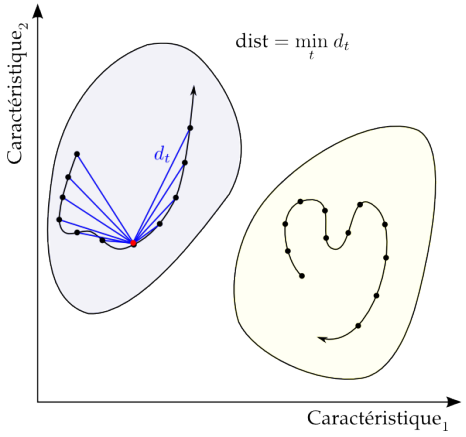
Notre approche est construite sur la prise en compte des non-stationnarités au sein des séquences, tout comme dans nos précédents travaux portant sur la classification. En ce sens, nous caractérisons une séquence par une représentation temporelle. Puisque le cadre de cette étude n'est pas restreint au traitement de signaux unidimensionnels, nous ne parlerons pas de représentations Temps-Fréquence (TF) mais de représentations TC. Toute séquence s de $\mathcal{Z}$ peut alors être associée à une fonction de $\mathcal{X}^{[0,T]}$, que nous noterons $x_\sim$, et que nous appellerons *représentation TC*. À un certain temps t de $[0, T]$, $\boldsymbol{x_t}$ est donc le vecteur caractéristique de la séquence s à l'instant t.

Pour les signaux unidimensionnels, toutes les représentations TF (comme la transformée de Fourier à court terme) et temps-échelles (comme la décomposition en ondelettes) sont des représentations TC puisqu'elles associent à chaque instant t un vecteur caractéristique. De même, les coefficients cepstraux mel-fréquences (*Mel-Frequency Cepstral Coefficients*, MFCC) calculés sur une fenêtre balayant continûment le signal est aussi une application TC (mais pas TF). Dans le traitement de vidéo, une fonction qui à chaque image associe un vecteur caractéristique est aussi une application TC.

L'approche présentée ici est fondée sur une vision de la discrimination de signaux différente de celle couramment utilisée (comme par exemple dans le chapitre précédent). Dans cette dernière, on mesure des similarités (et des différences) entre les séries temporelles via des distances euclidiennes entre des caractéristiques stationnaires (*e.g.* les MFCC calculés sur l'ensemble de la séquence) ou transitoires (*e.g.* une décomposition en ondelettes). Ces approches correspondent aux images (a) et (b) de la figure 4.2 page suivante. Sur l'image (a), chaque séquence est représentée par un point puisque les caractéristiques sont station-



(a) Caractéristique vs caractéristique. (b) TC vs TC.



(c) TC vs caractéristique.

FIGURE 4.2 – Illustration des différentes façons de séparer des séries temporelles. Deux classes sont représentées (en bleu et orange pâle) ainsi qu'un représentant par classe. Un trajet représente l'évolution temporelle d'une séquence dans le plan des caractéristiques. L'élément discriminant auquel les exemples sont comparés pour construire l'espace de similarité est en rouge.

naires. En revanche, sur l'image (b), chaque séquence est représentée par un trajet (temporel) dans le plan des caractéristiques. Comparer deux trajets revient à sommer les distances au carré de chaque instant, ou de manière équivalente, à concaténer les caractéristiques temporelles et à appliquer une distance euclidienne. Au contraire, notre approche se place dans le cadre de l'image (c) de la figure 4.2. Une série temporelle n'est pas comparée à une autre série, mais à un instantané, supposé discriminant pour une classe donnée. Cette façon de traiter les séquences répond au problème de divergence observé dans l'image (b) de la figure 4.2 : en comparant deux séquences temporelles entre elles, on suppose implicitement qu'elles sont ressemblantes à chaque instant (c'est la condition pour obtenir une grande similarité), alors qu'en réalité, seulement des parties d'entre elles sont proches (des parties supposées communes à toutes les séquences d'une même classe). L'instantané que nous avons mentionné apparaît donc comme un prototype discriminant supposé partagé entre les exemples d'une même classe et susceptible d'apparaître n'importe quand au cours de la séquence analysée.

Pour mettre en œuvre cette approche, nous redéfinissons à présent la notion de représentation par similarités, qui consiste à associer à chaque séquence un vecteur de similarités avec des prototypes donnés (des instantanés). Les espaces de similarité ont été introduits

dans [Balcan et Blum, 2006, Pekalska et Duin, 2008] pour des objets de même type. Dans notre contexte, nous comparons des objets de types différents : une série temporelle (fonction du temps) et un instantané (localisé temporellement). Par conséquent, la mesure de similarité (ou de proximité) que nous introduisons est une fonction dépendante du temps et non-symétrique.

Définition 4.2.1 (Mesure de similarité)**.**
Une mesure de similarité est une fonction $k_\sim : \mathcal{X}^{[0,T]} \times \mathcal{X} \to \mathbb{R}^{[0,T]}$, qui quantifie la similarité de ses arguments.

Avec cette définition, pour une représentation TC $x_\sim$ d'une séquence s et un prototype $\boldsymbol{p}$ de $\mathcal{X}$, $k_t(x_\sim, \boldsymbol{p})$ est un scalaire qui reflète la proximité de la séquence s au prototype $\boldsymbol{p}$ à l'instant t (de $[0, T]$).

Exemple 4.2.1.
Les deux fonctions définies ci-dessous sont des mesures de similarité :

$$\begin{aligned}(x_\sim, \boldsymbol{p}) \in \mathcal{X}^{[0,T]} \times \mathcal{X} \quad &\mapsto \quad \Big(t \mapsto -\left\|x_t - \boldsymbol{p}\right\|_{\ell_2}^2\Big),\\ (x_\sim, \boldsymbol{p}) \in \mathcal{X}^{[0,T]} \times \mathcal{X} \quad &\mapsto \quad \Big(t \mapsto \langle x_t \mid \boldsymbol{p}\rangle_{\ell_2}\Big).\end{aligned}$$

Ayant une mesure de similarité à notre disposition, il est maintenant possible de définir une *représentation* par similarités, qui agrège sous la forme d'un vecteur les proximités d'une séquence à plusieurs prototypes.

Définition 4.2.2 (Représentation par similarités)**.**
Soient $k_\sim$ une mesure de similarité et $\mathcal{L}$ un r-uplet de prototypes de $\mathcal{X}$: $\mathcal{L} = (\boldsymbol{p_1}, \ldots, \boldsymbol{p_r})$. La représentation par similarités fondée sur $k_\sim$ et $\mathcal{L}$ est notée $\psi_\sim^{\mathcal{L}}$. Elle est définie par :

$$\forall t \in [0,T], \quad \psi_t^{\mathcal{L}} : x_\sim \in \mathcal{X}^{[0,T]} \mapsto \begin{bmatrix} k_t(x_\sim, \boldsymbol{p_1}) \\ \vdots \\ k_t(x_\sim, \boldsymbol{p_r}) \end{bmatrix} \in \mathbb{R}^r.$$

Ainsi, pour une représentation TC $x_\sim$, $\psi_t^{\mathcal{L}}(x_\sim)$ est le vecteur de similarité entre $x_\sim$ et les prototypes $\boldsymbol{p_1}$, …, $\boldsymbol{p_r}$ à l'instant t. Cette notion est illustrée sur la figure 4.3, qui fournit un exemple d'événement temporel s, avec sa représentation TC $x_\sim$, ainsi que l'évolution de sa représentation par similarités $\psi_\sim^{\mathcal{L}}(x_\sim)$. Sur cette illustration $\boldsymbol{p_1}$ est un prototype discriminant qui est supposé contribuer à l'activation du détecteur. En revanche $\boldsymbol{p_2}$ s'avère être un instantané inutile pour la tâche de détection. Il illustre l'une des particularités de notre approche : elle exploite des instantanés discriminants présents à l'intérieur des événements à détecter mais sans les connaître *a priori*. En conséquence, ceux-ci sont à découvrir grâce à une base d'apprentissage contenant des événements (séquences à détecter) et des non-événements (séquences sans intérêt). Ceci fera en partie l'objet de la section 4.3.

Remarque 16.
En plus d'être des caractéristiques particulières et bien étudiées (*proximity space representation*, [Pekalska et Duin, 2008]), les espaces de similarité (ou de représentation) sont une généralisation de la notion de machines à vecteurs supports (*Support Vector Machine*, SVM) dans laquelle le noyau est remplacé par une mesure de similarité quelconque. Pour illustrer cette assertion, nous considérons un classifieur linéaire $f = \langle \boldsymbol{\alpha} \mid \cdot \rangle_{\ell_2} + b$. Appliqué à un point $\psi_t^{\mathcal{L}}(x_\sim)$ de l'espace de similarité, on obtient :

$$f(\psi_t^{\mathcal{L}}(x_\sim)) = \left\langle \boldsymbol{\alpha} \mid \psi_t^{\mathcal{L}}(x_\sim) \right\rangle_{\ell_2} + b = \sum_{i=1}^{r} \alpha_i k_t(x_\sim, \boldsymbol{p_i}) + b,$$

i.e. $(f - b) \in \mathrm{Vect}\left(\{k_t(\cdot, \boldsymbol{p_i}), i \in \mathbb{N}_r\}\right)$. On reconnaît immédiatement la forme d'une SVM avec deux différences majeures :

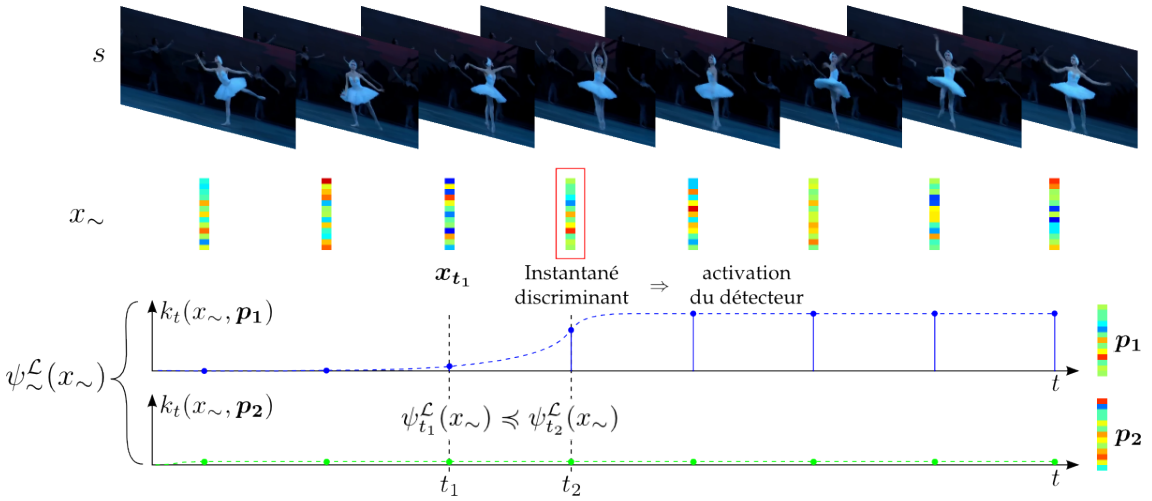


FIGURE 4.3 – Exemple de représentation par similarités. $\boldsymbol{x_{t_1}}$ est le vecteur caractéristique à l'instant t_1 correspondant à la séquence s. $\psi^{\mathcal{L}}_t(x_\sim)$ est le vecteur de similarités entre la représentation TC $x_\sim$ et deux prototypes $\boldsymbol{p_1}$ et $\boldsymbol{p_2}$. Dans cet exemple, le prototype $\boldsymbol{p_2}$ n'est pas discriminant pour la tâche de détection courante. En effet, l'instantané supposé activer le détecteur est celui correspondant à l'image sur laquelle la danseuse a les bras levés.

- ◇ $k_\sim$ est une mesure de similarité quelconque, qui n'est ni semi-définie positive, ni symétrique. Cette liberté permet de comparer des objets de natures différentes ;
- ◇ il n'y a aucune contrainte sur le signe de α_i suivant la nature de $\boldsymbol{p_i}$, tandis qu'une SVM impose que son signe soit identique à la classe à laquelle appartient le point associé.

4.2.2 Modèle pour la détection précoce

Nous introduisons à présent un détecteur linéaire dépendant du temps et s'appliquant dans l'espace de similarité. Pour toute séquence, le détecteur retourne une fonction dépendante du temps et à valeur réelle. Le signe de celle-ci indique la décision prise par notre détecteur : $+1$ signifie que la séquence analysée est un événement à détecter, tandis que -1 indique que la séquence est un non-événement.

Définition 4.2.3 (Détecteur linéaire)**.**
Soit $\psi^{\mathcal{L}}_{\sim}$ une représentation par similarités. $f^{\mathcal{L}}_{\sim}$ est un détecteur linéaire dans l'espace de similarité défini par $\psi^{\mathcal{L}}_{\sim}$ ssi $\exists(\mathbf{w}, b) \in \mathbb{R}^r \times \mathbb{R}$:

$$\forall t \in [0,T], \quad f^{\mathcal{L}}_t : x_\sim \in \mathcal{X}^{[0,T]} \mapsto \left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_t(x_\sim) \right\rangle_{\ell_2} - b \in \mathbb{R}.$$

Ainsi, pour une représentation TC $x_\sim$ (de $\mathcal{X}^{[0,T]}$), $\text{Signe}\left(f^{\mathcal{L}}_t(x_\sim)\right)$ est la décision prise par $f^{\mathcal{L}}_{\sim}$ à l'instant t. Puisque l'un de nos principaux objectifs est de prendre une décision le plus tôt possible, il est nécessaire que celle-ci ne prenne en compte que l'information (partielle) observée dans l'intervalle $[0,t]$ (*i.e.* que le détecteur soit causal) et que cette décision ne fluctue pas de manière erratique au fil du temps (propriété que l'on appelle la *fiabilité*). Pour ce faire, nous imposons que le détecteur ne change pas d'avis dès lors qu'il annonce une détection (*i.e.* que la *détection* soit fiable, comme illustré sur la figure 4.1 page 96). Ainsi, dans le cas d'une détection, la décision finale $\text{Signe}\left(f^{\mathcal{L}}_T(x_\sim)\right)$ (nécessairement $+1$), *i.e.* celle prise ayant analysé la totalité d'une séquence, peut être actée dès qu'un instantané discriminant apparaît à l'instant t, puisque $\text{Signe}\left(f^{\mathcal{L}}_T(x_\sim)\right) = \text{Signe}\left(f^{\mathcal{L}}_t(x_\sim)\right) = +1$. Analyser la totalité d'une séquence est donc inutile puisque le détecteur ne changera pas d'avis au cours du temps. En revanche, cette propriété n'est pas requise pour la décision contraire (-1).

Définition 4.2.4 (Détecteur totalement fiable)**.**
Soit $f^{\mathcal{L}}_{\sim}$ un détecteur linéaire dans un espace de similarité défini par $\psi^{\mathcal{L}}_{\sim}$. $f^{\mathcal{L}}_{\sim}$ est dit totalement fiable (ou possédant la propriété de fiabilité totale) ssi :

$$\forall x_{\sim} \in \mathcal{X}^{[0,T]}, \forall t \in [0,T]: \quad \psi^{\mathcal{L}}_{t}(x_{\sim}) = \psi^{\mathcal{L}}_{t}(x_{\sim} \cdot \chi_{[0,t]}) \qquad \text{(Causalité)}$$

$$\forall x_{\sim} \in \mathcal{X}^{[0,T]}, \forall t_1 \in [0,T]: \quad \left[f^{\mathcal{L}}_{t_1}(x_{\sim}) \geq 0\right] \Rightarrow \left[f^{\mathcal{L}}_{t_2}(x_{\sim}) \geq 0,\ \forall t_2 \in [t_1, T]\right] \qquad \text{(Fiabilité)},$$

où $\chi_{[0,t]}$ est la fonction caractéristique de l'ensemble $[0,t]$ (retournant 1 si son argument est dans $[0,t]$ et 0 sinon) et $\cdot$ représente le produit usuel entre deux fonctions.

Dans [Parrish *et coll.*, 2013], la classification précoce (et *a fortiori* la notion de fiabilité) est observée d'un point de vue probabiliste. Parrish *et coll.* posent et répondent à la question : « avec une probabilité τ, la décision prise sur la base de données incomplètes sera-t-elle identique à celle prise avec la totalité de l'information ? » La réponse des auteurs est construite sur un cadre probabiliste nécessitant l'estimation d'une densité de probabilité. Au contraire, nous adoptons ici une approche déterministe en affirmant que dès lors que le détecteur annonce une détection, il ne changera pas d'avis.

Ce point de vue déterministe est aussi à l'origine des travaux d'Hoai *et coll.* Dans [Hoai et De la Torre, 2014], une approche déterministe mais dirigée par les données (par augmentation de l'ensemble d'apprentissage) est présentée. L'algorithme proposé impose que la valeur de la fonction de décision croisse avec le temps dès lors qu'un événement est détecté dans la séquence analysée (à des erreurs près, pénalisées dans la fonction objectif). Puisque la fiabilité est assurée sur les données d'apprentissage et seulement à des erreurs près (les variables d'écart), celle-ci n'est absolument pas garantie sur des exemples inédits. Au contraire, le travail que nous présentons ici ne nécessite pas d'augmenter l'ensemble d'apprentissage artificiellement et assure la fiabilité *totale* du détecteur par la propriété suivante :

Proposition 4.2.1 (Détecteur fiable simple)**.**
Soient $\psi^{\mathcal{L}}_{\sim}$ une représentation par similarités construite sur une mesure de proximité $k_{\sim}$ et $f^{\mathcal{L}}_{\sim}$ un détecteur linéaire défini par $(\mathbf{w}, b) \in \mathbb{R}^r \times \mathbb{R}$. Si, pour toute représentation TC $x_{\sim}$ et tout prototype $\boldsymbol{p}$ de $\mathcal{X}$:

$$\forall t \in [0,T]: \quad k_t(x_{\sim}, \boldsymbol{p}) = k_t(x_{\sim} \cdot \chi_{[0,t]}, \boldsymbol{p}) \qquad \textit{(Causalité)}$$

$$\forall (t_1, t_2) \in [0,T]^2: \quad \left[t_1 \leq t_2\right] \Rightarrow \left[k_{t_1}(x_{\sim}, \boldsymbol{p}) \leq k_{t_2}(x_{\sim}, \boldsymbol{p})\right] \qquad \textit{(Croissance)}$$

$$\mathbf{w} \succcurlyeq 0, \qquad \textit{(Positivité)}$$

alors $f^{\mathcal{L}}_{\sim}$ est un détecteur totalement fiable.

Réciproquement, cette conclusion est également vérifiée si la mesure de similarité est une fonction décroissante du temps et si $\mathbf{w}$ vit dans l'orthant négatif.

Démonstration. Nous démontrons uniquement le cas principal de 4.2.1, lorsque $\mathbf{w} \succcurlyeq 0$. Le second cas (lorsque $\mathbf{w} \preccurlyeq 0$) est alors trivial. Supposons donc que nous satisfaisons les hypothèses de causalité, de croissance et de positivité de 4.2.1. Soit $(t_1, t_2) \in [0,T]^2$ tel que $t_1 \leq t_2$. Puisque $k_{\sim}(x_{\sim}, \boldsymbol{p})$ est croissant pout tout $\boldsymbol{p}$,

$$\psi^{\mathcal{L}}_{t_1}(x_{\sim}) \preccurlyeq \psi^{\mathcal{L}}_{t_2}(x_{\sim}).$$

De plus, comme $\mathbf{w} \succcurlyeq 0$ alors

$$\left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{t_1}(x_{\sim}) \right\rangle_{\ell_2} \leq \left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{t_2}(x_{\sim}) \right\rangle_{\ell_2}.$$

En conséquence, si $0 \leq f^{\mathcal{L}}_{t_1}(x_{\sim})$, alors

$$0 \leq \left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{t_1}(x_{\sim}) \right\rangle_{\ell_2} - b \leq \left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{t_2}(x_{\sim}) \right\rangle_{\ell_2} - b = f^{\mathcal{L}}_{t_2}(x_{\sim}).$$

■

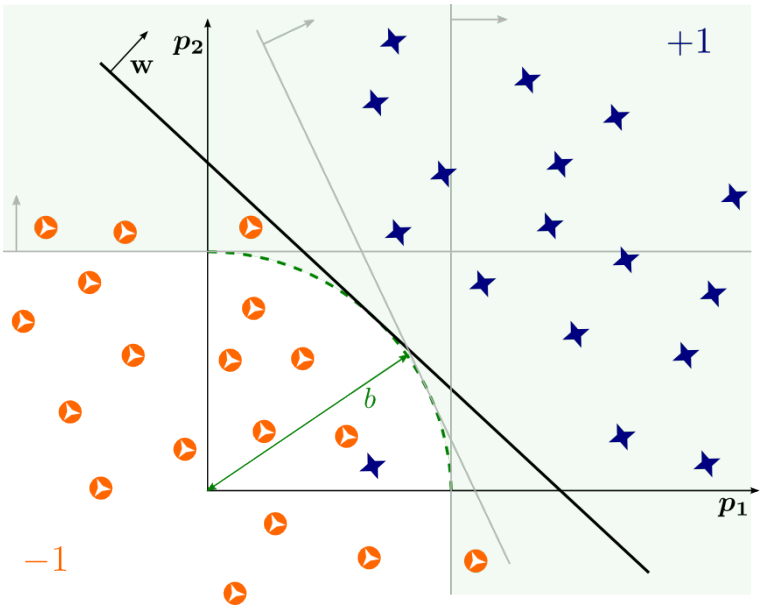


FIGURE 4.4 – Illustration de la contrainte géométrique $\mathbf{w} \succcurlyeq 0$ avec quelques hyperplans admissibles, à seuil de détection b fixé.

Remarque 17.

D'un point de vue géométrique, la propriété de positivité de la proposition 4.2.1 signifie que la classe des événements à détecter est située *en haut à droite* par rapport à la classe des non-événements (figure 4.4). Cette contrainte est cohérente avec l'utilisation d'un espace de similarité : puisque les prototypes de $\mathcal{L}$ sont supposés caractériser les événements à détecter (classe $+1$), les similarités $k_t(x_\sim, \boldsymbol{p})$ de ces événements doivent être plus importantes que celles des non-événements (ce qui se traduit géométriquement par *en haut à droite*).

La proposition 4.2.1 assure que, sous des hypothèses faibles sur la représentation par similarités et sur le détecteur linéaire, toute décision de détection est totalement fiable, *i.e.* si une détection est émise sur la base d'une séquence incomplète, alors analyser la séquence dans sa totalité conduira aussi à l'émission d'un avis de détection. Dans la section à venir, nous illustrons ces conditions faibles sur une représentation par similarités, en détaillant un exemple concret.

4.2.3 Une représentation par similarités adéquate

On considère ici un ensemble de séquences d'apprentissage $(s_i)_{1\leq i\leq n}$ (de $\mathcal{Z}$), extraites par exemple d'enregistrements audio, ainsi que leurs représentations TC $(x_\sim^{(i)})_{1\leq i\leq n}$. Ces dernières peuvent, par exemple, être les MFCC calculés sur une fenêtre glissante. Soit alors Δt un pas de temps. Nous choisissons, comme prototypes, les instantanés issus de la discrétisation de toutes les séquences :

$$\mathcal{L} = (\boldsymbol{p_j})_{1\leq j\leq r} = (x^{(i)}_{k_i\Delta t})_{1\leq i\leq n,\ 0\leq k_i\leq \lfloor \frac{T}{\Delta t} \rfloor}.$$

Soit un paramètre réel positif γ. Considérons la mesure de similarité $q_\sim$, définie pour tout t de $[0, T]$ par :

$$\forall (x_\sim, \boldsymbol{p}) \in \mathcal{X}^{[0,T]} \times \mathcal{X}, \quad q_t(x_\sim, \boldsymbol{p}) = \exp\left(-\gamma \left\|x_t - \boldsymbol{p}\right\|^2_{\ell_2}\right).$$

$q_\sim$ est probablement très efficace d'un point de vue géométrique puisque c'est un noyau gaussien, connu pour être flexible est adapté aux problèmes de reconnaissance non-

linéaires. Pourtant, la fonction $q_\sim$ ne satisfait pas les hypothèses de la proposition 4.2.1 puisqu'elle n'est manifestement pas croissante en fonction du temps. Définissons donc maintenant la mesure de similarité $k_\sim$, qui agrège toute l'information du passé grâce à une norme ℓ_p (p étant un entier strictement positif) :

$$\forall (x_\sim, \boldsymbol{p}) \in \mathcal{X}^{[0,T]} \times \mathcal{X}, \quad k_t(x_\sim, \boldsymbol{p}) = \left(\int_0^t (q_\delta(x_\sim, \boldsymbol{p}))^p \, d\delta \right)^{\frac{1}{p}}.$$

Alors la représentation par similarités $\psi^{\mathcal{L}}_\sim$, construite sur la mesure de similarité $k_\sim$ et sur les prototypes $\mathcal{L}$, répond aux hypothèses de causalité et de croissance de la proposition 4.2.1 (la croissance est illustrée sur la figure 4.3 page 100). Cette représentation peut donc être utilisée pour construire un détecteur fiable. Notons que lorsque $p \to +\infty$, *i.e.* la norme ℓ_p est remplacée par un maximum (l'agrégation temporelle consiste alors à retenir la plus grande valeur de similarité rencontrée jusqu'à un instant t, et ce pour chaque prototype), la représentation par similarités est identique à celle utilisée dans [Chen *et coll.*, 2006] pour transformer le problème MIL en une simple SVM, mais avec la dépendance de $\psi^{\mathcal{L}}_\sim$ au temps en plus. Nous reviendrons sur cette similitude plus en détails dans la section 4.4.

Selon le modèle de détecteur précoce défini ci-avant, une procédure de test d'une séquence donnée consisterait en toute simplicité à calculer la représentation par similarités $\psi^{\mathcal{L}}_t(x_\sim)$ d'une représentation TC $x_\sim$ à l'instant t, puis d'y appliquer la forme linéaire $\langle \mathbf{w} \mid \psi^{\mathcal{L}}_t(x_\sim) \rangle_{\ell_2} - b$ afin d'obtenir une décision. Cette décision est en réalité prise avec une information partielle puisque, à l'instar de la mesure de similarité considérée, la fonction de décision est causale. Comme nous l'avons expliqué précédemment, l'intuition à l'origine de ce type de représentations est la suivante : dès qu'un instantané discriminant apparaît dans la séquence, sa proximité avec les prototypes est capturée et retenue par l'agrégation en norme ℓ_p, activant alors le détecteur.

Nous expliquons à présent comment déterminer un couple $(\mathbf{w}, b)$ ainsi qu'un r-uplet de prototypes $\mathcal{L}$ définissant le détecteur $f^{\mathcal{L}}_\sim$.

4.3 ALGORITHME D'APPRENTISSAGE ET ANALYSE DE COMPLEXITÉ

4.3.1 Problème d'apprentissage

Choisir les prototypes de la représentation par similarités n'est pas tâche facile. Dans une représentation douce par k-moyennes (*soft K-means*), les prototypes sont les centres de groupes déterminés par une méthode de groupement automatique (*clustering*) [Coates *et coll.*, 2010]. Pourtant, à l'instar de ce qui a été présenté dans la section 4.2.3, il est souvent plus aisé et judicieux de considérer directement les instantanés issus des données d'apprentissage comme prototypes [Chen *et coll.*, 2006, Kar et Jain, 2012]. Nous nous plaçons donc dans le cas très général dans lequel l'ensemble de prototypes $\mathcal{L}$ est un (potentiellement très) grand r-uplet d'instantanés. Évidemment, dans une telle configuration, certains de ces instantanés ne sont pas pertinents comme prototype, puisqu'ils ne sont pas discriminants. En conséquence, il est nécessaire de se défausser de certains d'entre eux. Au lieu de procéder en deux étapes séparées, nous proposons de simultanément *sélectionner* les prototypes utiles et d'*apprendre* les paramètres $\mathbf{w}$ et b du détecteur linéaire dans l'espace de similarité correspondant. Pour ce faire, nous adoptons une approche très semblable à une SVM régularisée par une norme ℓ_1 [Zhu *et coll.*, 2004].

Dans les paragraphes qui suivent, nous nous plaçons dans le premier cas de la proposition 4.2.1, autrement dit, nous disposons d'une mesure de proximité $k_\sim$ causale et croissante. En outre, nous supposons posséder un ensemble d'apprentissage $(x^{(i)}_\sim, y_i)_{1 \leq i \leq n}$, où

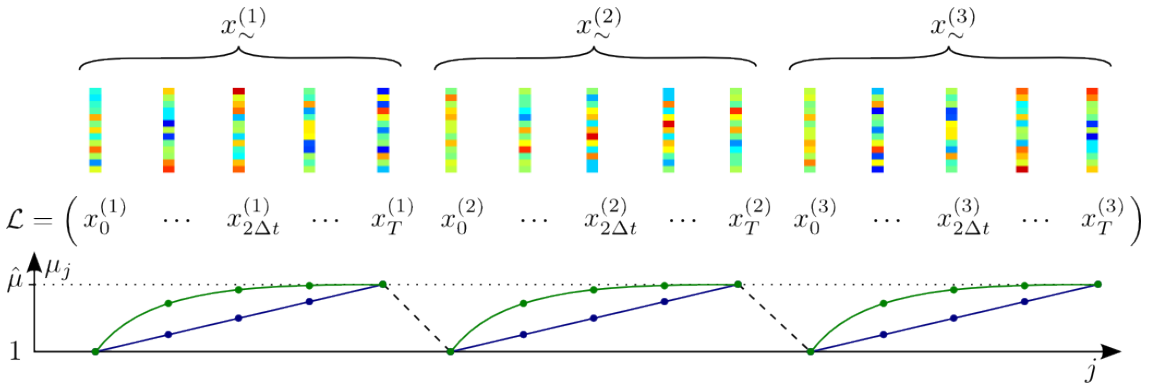


FIGURE 4.5 – Exemples de pondération de la norme $\|\cdot\|_{\boldsymbol{\mu},\ell_1}$ en fonction de l'ordre des prototypes (rangés ici par séquence et par instants d'apparition croissants). La courbe bleue est une pondération linéaire et la courbe verte, une pondération logarithmique. Chacune pénalise la sélection d'un instantané apparaissant à la fin d'une séquence.

$x_{\sim}^{(i)}$ est la représentation TC d'une séquence s_i et y_i est son étiquette, fixée à $+1$ si ladite séquence est un événement et -1 sinon. Le problème d'apprentissage que nous mettons à présent en place pour déterminer le détecteur $f_{\sim}^{\mathcal{L}}$ possède trois spécificités. La première réside dans la ***non-augmentation*** artificielle de l'ensemble d'apprentissage par des séquences partiellement observées. Cette technique est notamment utilisée dans [Hoai et De la Torre, 2014, Ellis *et coll.*, 2013] afin de favoriser la précocité de la prise de décision, mais présente le désavantage certain d'augmenter le nombre de contraintes ou de complexifier le calcul de la fonction objectif, suivant la nature de la pénalisation (Tikhonov ou Ivanov). Au contraire, le problème d'optimisation que nous construisons prend uniquement en compte les séquences complètes comme points d'apprentissage. En contrepartie, il est nécessaire de pénaliser une prise de décision tardive. Dans le cadre mis en place, qui suppose que le détecteur s'active grâce à des instantanés discriminants (les prototypes), cela correspond à pénaliser la sélection d'instantanés apparaissant tardivement dans les séquences analysées. Nous mettons ceci en pratique en remplaçant la norme ℓ_1 régularisant $\mathbf{w}$ par une ***norme pondérée*** $\|\cdot\|_{\boldsymbol{\mu},\ell_1}$ (c'est la deuxième caractéristique de notre approche). Celle-ci est définie par un vecteur de pondération $\boldsymbol{\mu}$, à composantes positives et croissantes avec l'instant d'apparition de l'instantané correspondant. On peut, par exemple, envisager des profils linéaires ou logarithmiques évoluant entre 1 et un certain $\hat{\mu}$ à définir (figure 4.5).

Selon la proposition 4.2.1, la fiabilité de notre détecteur est encodée dans la croissance de la mesure de similarité et dans la ***positivité*** du vecteur de pondération $\mathbf{w}$ (ou réciproquement dans la décroissance et la négativité). La troisième et dernière particularité de notre approche est donc d'imposer que le vecteur de poids du détecteur vive dans l'orthant positif. En conséquence, les deux actions conjointes de sélection des prototypes utiles et d'apprentissage du détecteur peuvent être réalisées en résolvant un problème SVM régularisé par une norme ℓ_1 pondérée avec une contrainte supplémentaire de positivité $\mathbf{w} \succcurlyeq 0$:

$$\begin{array}{ll} \underset{\mathbf{w},\boldsymbol{\xi},b}{\text{minimiser}} & (1-\lambda)\,\|\mathbf{w}\|_{\boldsymbol{\mu},\ell_1} + \lambda\,\|b\|_{\ell_1} + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \begin{cases} y_i\left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)})\right\rangle_{\ell_2} - b\right) \geq 1-\xi_i, \forall i \in \mathbb{N}_n \\ 0 \preccurlyeq \boldsymbol{\xi} \\ 0 \preccurlyeq \mathbf{w}, \end{cases} \end{array}$$

où λ est un paramètre de compromis choisi dans $[0,1]$ et C est le traditionnel paramètre de coût d'une SVM. Dans ce problème d'apprentissage, le seuil de détection b est aussi pénalisé. Cette pénalité a été introduite principalement pour une raison théorique, qui est de répondre aux hypothèses de [Kakade *et coll.*, 2009, théo. 3] (voir section suivante). En

pratique, cette pénalité est relativement commune et n'a généralement pas d'impact négatif sur les performances d'un modèle SVM [Mangasarian et Musicant, 2001].

En pratique, notre détecteur précoce est appris comme un classifieur binaire. Les instantanés discriminants (les prototypes) découverts par sélection parcimonieuse lors de l'apprentissage (grâce à la régularisation ℓ_1) sont généralement extraits des séquences événements, tandis que l'ensemble des non-événements joue un rôle dans la détermination du seuil de détection b.

4.3.2 Algorithme par ensemble actif

La pénalité sur le seuil de détection rend la fonction objectif du problème d'apprentissage non-dérivable. Cependant, en utilisant l'astuce courante consistant à remplacer b par $u - v$ (avec $u, v \geq 0$) il est aisé de montrer qu'à l'optimalité, $u = 0$ (et $\|b\|_{\ell_1} = v$) ou $v = 0$ (et $\|b\|_{\ell_1} = u$), donc $\|b\|_{\ell_1} = u + v$. Par conséquent, la régularisation en norme ℓ_1 sur le seuil de détection $\|b\|_{\ell_1}$ peut être remplacée par $u + v$. Le problème d'apprentissage du détecteur précoce $f^{\mathcal{L}}_{\sim}$ devient alors :

$$\begin{array}{ll} \underset{\mathbf{w}, \boldsymbol{\xi}, u, v}{\text{minimiser}} & (1-\lambda)\boldsymbol{\mu}^T\mathbf{w} + \lambda(u+v) + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \left\{ \begin{array}{l} y_i \left(\left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_T(x^{(i)}_{\sim}) \right\rangle_{\ell_2} - u + v \right) \geq 1 - \xi_i, \forall i \in \mathbb{N}_n \\ 0 \preccurlyeq \boldsymbol{\xi} \\ 0 \preccurlyeq \mathbf{w} \\ 0 \leq u, v. \end{array} \right. \end{array} \tag{4.1}$$

Le problème (4.1) est un programme linéaire (*Linear Program*, LP) et peut donc être aisément résolu par des logiciels populaires tels que *lpsolve* [Berkelaar *et coll.*, 2004]. La difficulté de ce problème ne réside pourtant pas dans sa forme mais dans sa taille. Un calcul rapide montre qu'il possède $r + n + 2$ variables d'optimisation et $r + 2n + 2$ contraintes linéaires. Or si n est supposé de taille raisonnable, on s'attend à ce que le nombre de prototypes initiaux r soit très grand. Ceci est d'autant plus préjudiciable que beaucoup de valeurs de $\mathbf{w}$ seront nulles à l'optimalité, du fait de la contrainte de parcimonie (ainsi beaucoup de prototypes sont pris en compte dans le problème d'optimisation sans pour autant intervenir *in fine* dans la définition de l'espace de similarité). Pour réduire la taille du problème, on propose, dans un premier temps, de considérer un problème dual de (4.1) (les variables duales des contraintes d'intérêt sont mentionnées à droite de celles-ci) :

$$\begin{array}{ll} \underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\text{maximiser}} & \mathbb{1}^T\boldsymbol{\alpha} \\ \text{tel que} & \left\{ \begin{array}{ll} 0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C\mathbb{1} & \\ \boldsymbol{Q}^{\mathcal{L}}\boldsymbol{\alpha} \preccurlyeq (1-\lambda)\boldsymbol{\mu} & : \mathbf{w} \\ -\lambda \leq \boldsymbol{\alpha}^T\boldsymbol{y} \leq \lambda & : u, v, \end{array} \right. \end{array} \tag{4.2}$$

où $\boldsymbol{Q}^{\mathcal{L}}$ est une matrice de $\mathbb{R}^{r \times n}$ définie par : $\boldsymbol{Q}^{\mathcal{L}} = \left[y_1 \psi^{\mathcal{L}}_T(x^{(1)}_{\sim}), \ldots, y_n \psi^{\mathcal{L}}_T(x^{(n)}_{\sim}) \right]$. L'annexe G détaille les calculs permettant d'aboutir à cette formulation. Le problème (4.2) possède $r + n + 2$ contraintes et n variables d'optimisation. Cette simple réduction n'est pas satisfaisante puisque le nouveau problème possède moins de variables mais presque autant de contraintes. En revanche, les conditions d'optimalité affirment que si $\mathbf{w}_j > 0$, alors $(\boldsymbol{Q}^{\mathcal{L}}\boldsymbol{\alpha})_j = (1-\lambda)\mu_j$ et réciproquement, si $\mathbf{w}_j = 0$, alors $(\boldsymbol{Q}^{\mathcal{L}}\boldsymbol{\alpha})_j \leq (1-\lambda)\mu_j$ (voir l'annexe G pour le détail des calculs). Ceci suggère un algorithme de génération de colonnes (ou de contraintes actives) [Nocedal et Wright, 2000], qui partirait d'un ensemble $\mathcal{A}$ de coordonnées supposées non-nulles de $\mathbf{w}$ (*i.e.* d'indices de prototypes discriminants) et qui évoluerait vers l'ensemble optimal.

Données : ensemble d'apprentissage $\left(\psi_T^{\mathcal{L}}(x_{\sim}^{(i)}), y_i\right)_{1 \leq i \leq n}$.

retourner *détecteur linéaire* $(\mathbf{w}, b)$.

$\mathcal{A} \leftarrow$ échantillon aléatoire de $\mathbb{N}_r$;

tant que *équilibre non-atteint* **faire**

  Résoudre (4.2) avec $\boldsymbol{Q}^{\mathcal{L}} \leftarrow \boldsymbol{Q}^{\mathcal{L}'}$, où $\mathcal{L}' = (\boldsymbol{p_j})_{j \in \mathcal{A}}$;

  $\theta \leftarrow \arg\max_{j \in \mathbb{N}_r} (\boldsymbol{Q}^{\mathcal{L}} \boldsymbol{\alpha})_j$;

  si $(\boldsymbol{Q}^{\mathcal{L}} \boldsymbol{\alpha})_\theta \leq (1 - \lambda)\mu_\theta$ **alors**

    équilibre atteint ;

  sinon

    $\mathcal{A} \leftarrow \mathcal{A} \cup \{\theta\}$;

Algorithme 4 : Apprentissage du détecteur linéaire par contraintes actives.

Dans l'algorithme 4, qui décrit une procédure de résolution par contraintes actives du problème (4.2) (et *a fortiori* de (4.1)), chaque itération l consiste, dans un premier temps, à résoudre un LP à $\mathrm{Card}(\mathcal{A})+n+2$ contraintes et n variables d'optimisation, avec $\mathrm{Card}(\mathcal{A}) \ll r$. Dans un second temps, on détermine une coordonnée θ de $\mathbf{w}$ qui viole les conditions d'optimalité, et on l'ajoute à l'ensemble des contraintes actives. En notant $\boldsymbol{\alpha}^{(l)}$ une solution optimale du sous-problème (4.2) à l'itération l et $\boldsymbol{\alpha}^{(l+1)}$ une solution de celui à l'itération suivante, il est évident que $\mathbb{1}^T \boldsymbol{\alpha}^{(l+1)} \leq \mathbb{1}^T \boldsymbol{\alpha}^{(l)}$. Ceci est dû au fait que le problème d'optimisation à l'itération $l+1$ est plus contraint que celui à l'itération précédente. De plus, si la solution $\boldsymbol{\alpha}^{(l)}$ à l'itération l est unique, alors $\mathbb{1}^T \boldsymbol{\alpha}^{(l+1)} < \mathbb{1}^T \boldsymbol{\alpha}^{(l)}$. En effet, puisque $\boldsymbol{\alpha}^{(l+1)}$ est un point faisable du problème d'optimisation à l'itération l, si l'on avait l'égalité, alors $\boldsymbol{\alpha}^{(l+1)}$ serait aussi une solution de ce problème et par unicité : $\boldsymbol{\alpha}^{(l+1)} = \boldsymbol{\alpha}^{(l)}$. Ceci impliquerait que $\boldsymbol{\alpha}^{(l)}$ est un point faisable du problème d'optimisation de l'itération $l+1$, ce qui, par construction, n'est pas. Par conséquent, la valeur de la fonction objectif décroît (strictement sous certaines conditions) à chaque itération de l'algorithme 4. Notons qu'en pratique, l'algorithme 4 est implémenté de manière légèrement différente puisque l'on ajoute en réalité plusieurs coordonnées de $\mathbf{w}$ violant les conditions d'optimalité et que l'on retire de $\mathcal{A}$ les coordonnées de poids nuls, afin de contrôler la taille du sous-problème. Cette stratégie d'implémentation est identique à celle utilisée dans la section 3.4 et dans [Gehler et Nowozin, 2008b].

4.3.3 Algorithme incrémental

Dans la section précédente, nous avons présenté un algorithme par ensemble actif, afin de résoudre efficacement un problème d'optimisation proche d'une SVM comportant beaucoup de variables et de contraintes. Dans la littérature SVM, une approche différente en pratique mais proche dans les idées consiste à mettre à jour la fonction de décision apprise en ajoutant un exemple d'apprentissage [Cauwenberghs et Poggio, 2001, Ralaivola et d'Alché Buc, 2001]. Cette technique est appelée *SVM incrémentale*. Dans la lignée des travaux initiés par [Cauwenberghs et Poggio, 2001] et poursuivis dans [Laskov *et coll.*, 2006, Karasuyama et Takeuchi, 2009, Karasuyama et Takeuchi, 2010], nous étendons la notion d'incrément à la dimension des données, plutôt qu'à leur nombre. Dans notre contexte, cela correspond à ajouter un instantané à l'ensemble des prototypes $\mathcal{L}$.

Pour ce faire, nous considérons $\lambda = 0$ (par simplicité) et nous suivons la démarche usuelle

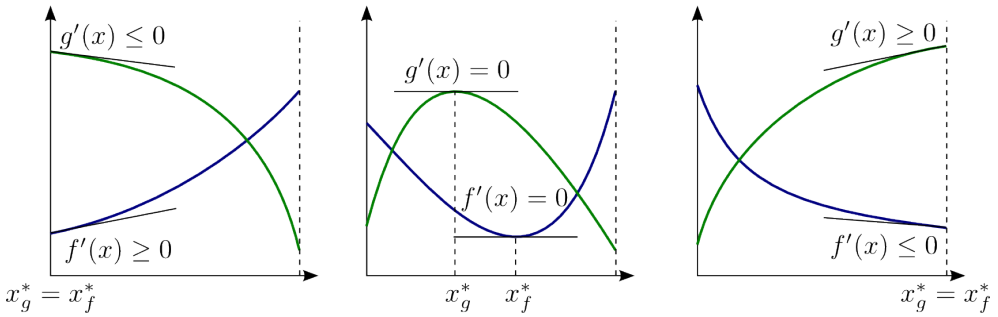


FIGURE 4.6 – Conditions d'optimalité du premier ordre pour la minimisation d'une fonction bleue f et la maximisation d'une fonction verte g sur un segment. Lorsque l'optimum est atteint dans l'intérieur du segment, la dérivée de la fonction est nulle. En revanche, lorsque celui-ci se trouve sur un bord, la dérivée est soit positive (minimisation à gauche ou maximisation à droite), soit négative (minimisation à droite ou maximisation à gauche).

en ré-écrivant (4.2) sous la forme d'un problème à point-selle (en notant à présent $\boldsymbol{Q} = \boldsymbol{Q}^{\mathcal{L}}$) :

$$\underset{\mathbf{w}\succcurlyeq 0,\, b\in\mathbb{R}}{\text{maximiser}} \quad \min_{0\preccurlyeq \boldsymbol{\alpha}\preccurlyeq C\mathbb{1}} \; -\mathbb{1}^T\boldsymbol{\alpha} + \mathbf{w}^T(\boldsymbol{Q}\boldsymbol{\alpha} - \boldsymbol{\mu}) - b\boldsymbol{y}^T\boldsymbol{\alpha}. \tag{4.3}$$

Soient alors les ensembles indexant les points respectivement dans, sur et en dehors de la marge :

$$\mathcal{I} = \left\{ i \in \mathbb{N}_n, y_i \left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)}) \right\rangle_{\ell_2} - b \right) < 1 \right\}$$
$$\mathcal{M} = \left\{ i \in \mathbb{N}_n, y_i \left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)}) \right\rangle_{\ell_2} - b \right) = 1 \right\}$$
$$\mathcal{O} = \left\{ i \in \mathbb{N}_n, y_i \left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)}) \right\rangle_{\ell_2} - b \right) > 1 \right\}.$$

Soit aussi la partition $(\mathcal{S}, \overline{\mathcal{S}})$ des dimensions (coordonnées de $\mathbf{w}$) actives et inactives :

$$\mathcal{S} = \{j \in \mathbb{N}_r, \boldsymbol{Q}_{\boldsymbol{j},\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}} = \mu_j\} \qquad \overline{\mathcal{S}} = \{j \in \mathbb{N}_r, \boldsymbol{Q}_{\boldsymbol{j},\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}} < \mu_j\}.$$

Supposons que $\mathcal{M}$ et $\mathcal{S}$ sont non-vides. En s'aidant de la figure 4.6, les conditions d'optimalité du premier ordre du problème (4.3) sont obtenues en calculant les gradients de la fonction objectif par rapport aux différentes variables :

$$\boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}} = \boldsymbol{\mu}_{\boldsymbol{\mathcal{S}}} \quad (\mathbf{w}_{\mathcal{S}} \geq 0) \qquad \boldsymbol{Q}_{\overline{\boldsymbol{\mathcal{S}}},\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}} < \boldsymbol{\mu}_{\overline{\boldsymbol{\mathcal{S}}}} \quad (\mathbf{w}_{\overline{\mathcal{S}}} = 0)$$
$$\boldsymbol{y}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}{}^T\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}} = 0 \quad (b \in \mathbb{R}) \qquad \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{I}}}{}^T\mathbf{w}_{\boldsymbol{\mathcal{S}}} - b\boldsymbol{y}_{\boldsymbol{\mathcal{I}}} < \mathbb{1} \quad (\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}} = C\mathbb{1})$$
$$\boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{M}}}{}^T\mathbf{w}_{\boldsymbol{\mathcal{S}}} - b\boldsymbol{y}_{\boldsymbol{\mathcal{M}}} = \mathbb{1} \quad (0 \preccurlyeq \boldsymbol{\alpha}_{\boldsymbol{\mathcal{M}}} \preccurlyeq C\mathbb{1}) \qquad \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{O}}}{}^T\mathbf{w}_{\boldsymbol{\mathcal{S}}} - b\boldsymbol{y}_{\boldsymbol{\mathcal{O}}} > \mathbb{1} \quad (\boldsymbol{\alpha}_{\boldsymbol{\mathcal{O}}} = 0)$$

Dans ces relations, nous avons déjà intégré le fait que $\boldsymbol{\alpha}_{\boldsymbol{\mathcal{O}}} = 0$ et $\mathbf{w}_{\overline{\mathcal{S}}} = 0$.

On désire à présent ajouter de nouvelles dimensions aux données, correspondant ainsi à *étendre* la matrice $\boldsymbol{Q}$ et le vecteur $\mathbf{w}$ à un ensemble d'indices, noté $\mathcal{A}$. À ce stade, puisque les dimensions indexées par $\mathcal{A}$ ne sont pas encore prises en compte, on peut donc considérer qu'elles sont présentes dans $\boldsymbol{Q}$ et $\mathbf{w}$, mais inactives : $\mathbf{w}_{\boldsymbol{\mathcal{A}}} = 0$. Parmi les coordonnées indexées par $\mathcal{A}$, certaines respectent les conditions d'optimalité, d'autres non. Il est alors naturel de réaffecter celles qui satisfont les conditions d'optimalités à $\overline{\mathcal{S}}$ et de conserver dans $\mathcal{A}$ uniquement celles qui violent l'optimalité. Le but est maintenant de faire évoluer $\mathbf{w}_{\boldsymbol{\mathcal{A}}}$ jusqu'à un état satisfaisant les conditions d'optimalité. Évidemment, durant cette évolution,

les autres variables sont susceptibles de changer de valeurs. Appelons donc Δ l'opérateur de *petite variation*. Si l'on suppose que les parois entre les ensembles précédemment définis ($\mathcal{I}$, $\mathcal{M}$, $\mathcal{O}$, $\mathcal{S}$ et $\overline{\mathcal{S}}$) sont imperméables (autrement dit, qu'aucun point ne quitte l'un de ces ensembles pour en peupler un autre), alors toute variation des variables d'optimisation vérifie :

$$\begin{aligned} \boldsymbol{Q}_{\mathcal{S},\mathcal{M}}\boldsymbol{\Delta\alpha}_{\mathcal{M}} &= 0 \\ \boldsymbol{y}_{\mathcal{M}}{}^{T}\boldsymbol{\Delta\alpha}_{\mathcal{M}} &= 0 \\ \boldsymbol{Q}_{\mathcal{S},\mathcal{M}}{}^{T}\boldsymbol{\Delta}\mathbf{w}_{\mathcal{S}} + \boldsymbol{Q}_{\mathcal{A},\mathcal{M}}{}^{T}\boldsymbol{\Delta}\mathbf{w}_{\mathcal{A}} - \Delta b\boldsymbol{y}_{\mathcal{M}} &= 0. \end{aligned}$$

Ces trois équations peuvent être ré-écrites sous forme matricielle suivant une unique équation faisant intervenir d'une part $\boldsymbol{\Delta}\mathbf{w}_{\mathcal{A}}$, et d'autre part les autres variations :

$$\boldsymbol{M}\begin{bmatrix} \Delta b \\ \boldsymbol{\Delta\alpha}_{\mathcal{M}} \\ \boldsymbol{\Delta}\mathbf{w}_{\mathcal{S}} \end{bmatrix} = -\begin{bmatrix} 0 \\ 0 \\ \boldsymbol{Q}_{\mathcal{A},\mathcal{M}}{}^{T} \end{bmatrix}\boldsymbol{\Delta}\mathbf{w}_{\mathcal{A}}, \tag{4.4}$$

où

$$\boldsymbol{M} = \begin{bmatrix} 0 & \boldsymbol{Q}_{\mathcal{S},\mathcal{M}} & 0 \\ 0 & \boldsymbol{y}_{\mathcal{M}}{}^{T} & 0 \\ -\boldsymbol{y}_{\mathcal{M}} & 0 & \boldsymbol{Q}_{\mathcal{S},\mathcal{M}}{}^{T} \end{bmatrix}.$$

Cette relation nous indique l'impact d'une variation de $\mathbf{w}_{\mathcal{A}}$ sur les variables b, $\boldsymbol{\alpha}$ et $\mathbf{w}$. Comme nous l'avons supposé, cette équation est uniquement valable si les parois des ensembles sont imperméables, autrement dit, si les inégalités d'optimalité écrites plus haut sont vérifiées. Nous les récrivons à présent en faisant apparaître entre parenthèses les mises à jour à réaliser lorsqu'un point transforme l'une de ces inégalités strictes en égalités :

$$\begin{aligned} \boldsymbol{Q}_{\mathcal{S},\mathcal{I}}{}^{T}(\mathbf{w}_{\mathcal{S}} + \boldsymbol{\Delta}\mathbf{w}_{\mathcal{S}}) + \boldsymbol{Q}_{\mathcal{A},\mathcal{I}}{}^{T}(\mathbf{w}_{\mathcal{A}} + \boldsymbol{\Delta}\mathbf{w}_{\mathcal{A}}) - (b + \Delta b)\boldsymbol{y}_{\mathcal{I}} &< \mathbb{1} \quad (\mathcal{I} \to \mathcal{M}) \\ \boldsymbol{Q}_{\mathcal{S},\mathcal{O}}{}^{T}(\mathbf{w}_{\mathcal{S}} + \boldsymbol{\Delta}\mathbf{w}_{\mathcal{S}}) + \boldsymbol{Q}_{\mathcal{A},\mathcal{O}}{}^{T}(\mathbf{w}_{\mathcal{A}} + \boldsymbol{\Delta}\mathbf{w}_{\mathcal{A}}) - (b + \Delta b)\boldsymbol{y}_{\mathcal{O}} &> \mathbb{1} \quad (\mathcal{O} \to \mathcal{M}) \\ \mathbf{w}_{\mathcal{S}} + \boldsymbol{\Delta}\mathbf{w}_{\mathcal{S}} &> 0 \quad (\mathcal{S} \to \overline{\mathcal{S}}) \\ 0 < \boldsymbol{\alpha}_{\mathcal{M}} + \boldsymbol{\Delta\alpha}_{\mathcal{M}} &< C\mathbb{1} \quad (\mathcal{M} \to \mathcal{I} \text{ ou } \mathcal{O}) \\ \boldsymbol{Q}_{\overline{\mathcal{S}},\mathcal{I}}\boldsymbol{\alpha}_{\mathcal{I}} + \boldsymbol{Q}_{\overline{\mathcal{S}},\mathcal{M}}(\boldsymbol{\alpha}_{\mathcal{M}} + \boldsymbol{\Delta\alpha}_{\mathcal{M}}) &< \boldsymbol{\mu}_{\overline{\mathcal{S}}} \quad (\overline{\mathcal{S}} \to \mathcal{S}). \end{aligned}$$

De plus, il est nécessaire de conserver dans $\mathcal{A}$, uniquement les coordonnées violant les conditions d'optimalité :

$$\boldsymbol{Q}_{\mathcal{A},\mathcal{I}}\boldsymbol{\alpha}_{\mathcal{I}} + \boldsymbol{Q}_{\mathcal{A},\mathcal{M}}(\boldsymbol{\alpha}_{\mathcal{M}} + \boldsymbol{\Delta\alpha}_{\mathcal{M}}) > \boldsymbol{\mu}_{\mathcal{A}} \quad (\mathcal{A} \to \mathcal{S}).$$

Lorsque l'une de ces inégalités est atteinte, il est nécessaire de mettre à jour les ensembles mis en jeu. Par exemple, quand cette dernière expression de violation est transformée en égalité pour une coordonnée, *i.e.* $\exists\theta \in \mathcal{A}$:

$$\boldsymbol{Q}_{\theta,\mathcal{I}}\boldsymbol{\alpha}_{\mathcal{I}} + \boldsymbol{Q}_{\theta,\mathcal{M}}(\boldsymbol{\alpha}_{\mathcal{M}} + \boldsymbol{\Delta\alpha}_{\mathcal{M}}) = \mu_{\theta},$$

on effectue la double mise à jour :

$$\mathcal{A} \leftarrow \mathcal{A}\backslash\{\theta\}, \quad \mathcal{S} \leftarrow \mathcal{S} \cup \{\theta\}.$$

Nous disposons à présent de tous les ingrédients pour faire évoluer itérativement les variables vers un état d'équilibre. En pratique, on se donne $\boldsymbol{\Delta}\mathbf{w}_{\mathcal{A}} = \eta\mathbb{1}$ ($\eta > 0$), où η est un pas d'avancement. On cherche alors le plus grand pas d'avancement admissible, *i.e.* ne violant pas l'imperméabilité des ensembles. Pour ce faire, nous supposons que le système (4.4) est soluble, *i.e.* que $\boldsymbol{M}$ est inversible. Lorsque $\boldsymbol{M}$ n'est pas inversible, il suffit d'ajouter

une faible diagonale positive à la matrice : $\boldsymbol{M} = \boldsymbol{M} + \epsilon \boldsymbol{I}_+$ (avec $0 < \epsilon \ll 1$, typiquement $\epsilon = 10^{-6}$). Ceci permet d'obtenir une direction d'évolution suffisante. En remplaçant $\boldsymbol{\Delta}\mathrm{w}_{\boldsymbol{\mathcal{A}}}$ par $\eta\mathbb{1}$ dans (4.4), on obtient :

$$\begin{bmatrix} \Delta b \\ \boldsymbol{\Delta\alpha}_{\boldsymbol{\mathcal{M}}} \\ \boldsymbol{\Delta}\mathrm{w}_{\boldsymbol{\mathcal{S}}} \end{bmatrix} = \eta\boldsymbol{\phi}, \text{ avec } \quad \boldsymbol{\phi} = -\boldsymbol{M}^{-1} \begin{bmatrix} 0 \\ 0 \\ \boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{M}}}{}^T \mathbb{1} \end{bmatrix}.$$

En détaillant les contributions de $\boldsymbol{\phi}$ en $\boldsymbol{\phi} = [\phi_b,\ \boldsymbol{\phi}_{\boldsymbol{\alpha}},\ \boldsymbol{\phi}_{\mathrm{w}}]^T$, les inégalités d'optimalité assurant l'imperméabilité des ensembles prennent alors la forme $0 \leq \eta < \frac{\triangle}{\square}$:

$$\eta < \frac{1 - \left(\boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{I}}}{}^T \mathrm{w}_{\boldsymbol{\mathcal{S}}} + \boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{I}}}{}^T \mathrm{w}_{\boldsymbol{\mathcal{A}}} - b\boldsymbol{y}_{\boldsymbol{\mathcal{I}}}\right)_j}{\left(\left[-\boldsymbol{y}_{\boldsymbol{\mathcal{I}}},\ 0,\ \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{I}}}{}^T,\ \boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{I}}}{}^T\right]\left[\phi_b,\ \boldsymbol{\phi}_{\boldsymbol{\alpha}},\ \boldsymbol{\phi}_{\mathrm{w}},\ \mathbb{1}\right]^T\right)_j}, \quad \text{pour } j \ / \ \square > 0$$

$$\eta < \frac{1 - \left(\boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{O}}}{}^T \mathrm{w}_{\boldsymbol{\mathcal{S}}} + \boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{O}}}{}^T \mathrm{w}_{\boldsymbol{\mathcal{A}}} - b\boldsymbol{y}_{\boldsymbol{\mathcal{O}}}\right)_j}{\left(\left[-\boldsymbol{y}_{\boldsymbol{\mathcal{O}}},\ 0,\ \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{O}}}{}^T,\ \boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{O}}}{}^T\right]\left[\phi_b,\ \boldsymbol{\phi}_{\boldsymbol{\alpha}},\ \boldsymbol{\phi}_{\mathrm{w}},\ \mathbb{1}\right]^T\right)_j}, \quad \text{pour } j \ / \ \square < 0$$

$$\eta < \frac{-\left(\mathrm{w}_{\boldsymbol{\mathcal{S}}}\right)_j}{\left(\boldsymbol{\phi}_{\mathrm{w}}\right)_j}, \quad \text{pour } j \ / \ \square < 0$$

$$\eta < \frac{C - \left(\boldsymbol{\alpha}_{\boldsymbol{\mathcal{M}}}\right)_j}{\left(\boldsymbol{\phi}_{\boldsymbol{\alpha}}\right)_j}, \quad \text{pour } j \ / \ \square > 0$$

$$\eta < \frac{-\left(\boldsymbol{\alpha}_{\boldsymbol{\mathcal{M}}}\right)_j}{\left(\boldsymbol{\phi}_{\boldsymbol{\alpha}}\right)_j}, \quad \text{pour } j \ / \ \square < 0$$

$$\eta < \frac{\left(\boldsymbol{\mu}_{\overline{\boldsymbol{\mathcal{S}}}} - \boldsymbol{Q}_{\overline{\boldsymbol{\mathcal{S}}},\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\right)_j}{\left(\boldsymbol{Q}_{\overline{\boldsymbol{\mathcal{S}}},\boldsymbol{\mathcal{M}}}\boldsymbol{\phi}_{\boldsymbol{\alpha}}\right)_j}, \quad \text{pour } j \ / \ \square > 0$$

$$\eta < \frac{\left(\boldsymbol{\mu}_{\boldsymbol{\mathcal{A}}} - \boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\right)_j}{\left(\boldsymbol{Q}_{\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{M}}}\boldsymbol{\phi}_{\boldsymbol{\alpha}}\right)_j}, \quad \text{pour } j \ / \ \square < 0.$$

Les inégalités précédentes nous donnent l'ensemble des valeurs admissibles pour η. La plus grande de ces valeurs correspond à la migration d'un point de l'ensemble d'apprentissage ou d'une coordonnée de w d'un ensemble vers un autre. C'est cette valeur que nous assignons à η avant de mettre lesdits ensembles à jour. Une première idée de l'algorithme incrémental est alors de répéter les trois étapes suivantes :

- ◇ calculer $\boldsymbol{\phi}$ par résolution du système linéaire (4.4) ;
- ◇ déterminer le plus grand η qui satisfait les contraintes d'imperméabilité ;
- ◇ mettre à jour des variables b, $\boldsymbol{\alpha}_{\boldsymbol{\mathcal{M}}}$, $\mathrm{w}_{\boldsymbol{\mathcal{S}}}$ et $\mathrm{w}_{\boldsymbol{\mathcal{A}}}$, ainsi que les ensembles suivant l'inégalité atteinte par η.

Cycles sur la marge

L'un des écueils (encore non élucidé à notre connaissance) auxquels sont confrontés les algorithmes incrémentaux est la non-inversibilité de $\boldsymbol{M}$, en partie due à la nature du noyau SVM[1]. C'est pourquoi ceux-ci sont généralement appliqués à des problèmes définis par des noyaux à bases radiales, tels que le noyau gaussien (connu pour être défini-positif). Dans notre cas, cette porte de sortie est inaccessible puisque nous ne travaillons qu'avec un

1. Cette non-inversibilité apparaît aussi lorsque l'ensemble d'apprentissage comporte des points en double.

noyau linéaire. En pratique, on peut observer que cette situation de non-inversibilité de $\boldsymbol{M}$ apparaît parfois lorsqu'un point arrive *sur* la marge. Le système est alors dans un nouvel équilibre mais celui-ci n'est pas défini de manière unique (d'où la non-inversibilité). En ajoutant une faible diagonale à $\boldsymbol{M}$, il est possible d'obtenir une direction d'évolution mais celle-ci résulte en un cycle infini : le dernier point arrivé sur la marge est éjecté, puis rajouté, et éjecté, *etc.* D'un point de vue numérique, ceci correspond à une situation pathologique dans laquelle $y_i \left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)}) \right\rangle_{\ell_2} - y_i b = 1$ et $\alpha_i = 0$ ou $\alpha_i = C$, *i.e.* le point $\psi_T^{\mathcal{L}}(x_{\sim}^{(i)})$ est à la fois *sur* et *en dehors* de la marge.

Dans cette situation, afin d'éviter les cycles infinis, nous proposons une étape de *pénétration de la marge*, consistant à oublier un instant l'ajout de nouvelles dimensions et à faire évoluer le système (au sein de son équilibre qui n'est pas défini de manière unique) vers un nouvel état d'équilibre extrême, conduisant à la migration d'un point. Pour ce faire, nous mettons à part le dernier point ajouté à la marge (noté $*$), et ré-écrivons le système linaire des variations (avec $\mathcal{M}^* = \mathcal{M} \backslash \{*\}$) :

$$\begin{bmatrix} 0 & \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{M}}^*} & 0 \\ 0 & \boldsymbol{y}_{\boldsymbol{\mathcal{M}}^*}{}^T & 0 \\ -\boldsymbol{y}_{\boldsymbol{\mathcal{M}}} & 0 & \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{M}}}{}^T \end{bmatrix} \begin{bmatrix} \Delta b \\ \boldsymbol{\Delta\alpha}_{\boldsymbol{\mathcal{M}}^*} \\ \boldsymbol{\Delta w}_{\boldsymbol{\mathcal{S}}} \end{bmatrix} = - \begin{bmatrix} \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},*} \\ y_* \\ 0 \end{bmatrix} \Delta\alpha_*.$$

En prenant comme direction d'évolution $\Delta\alpha_* = \eta C$ ($\eta \in [0,1]$) si le point indexé par $*$ provenait auparavant de l'extérieur de la marge et $\Delta\alpha_* = -\eta C$ s'il provenait de l'intérieur, on obtient un nouveau système d'inéquations indiquant les pas η admissibles. Pour ces inégalités, $\boldsymbol{\Delta w}_{\boldsymbol{\mathcal{A}}} = 0$ puisque l'ajout des dimensions a été mis temporairement de côté. La stratégie consiste à prendre le plus grand pas possible et à mettre les variables et les ensembles à jour en accord avec le type de migration qui apparaît.

Cycles sur les dimensions

Ce phénomène d'instabilité numérique apparaît aussi lorsqu'une dimension (d'indice $*$) active devient inactive, avec $\mathbf{w}_* = 0$ et $\boldsymbol{Q}_{*,\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}\cup\boldsymbol{\mathcal{M}}} = \mu_*$. Dans ce cas, puisque la coordonnée $*$ a souhaité être inactive, on la considère comme telle et on la met temporairement de côté (dans un ensemble noté $\mathcal{W}$). Lorsque toutes les dimensions ont été ajoutées (*i.e.* $\mathcal{A} = \emptyset$), on repeuple $\mathcal{A}$ avec les coordonnées laissées de côté ($\mathcal{A} = \mathcal{W}$) puis on réaffecte les éléments de $\mathcal{A}$ à l'équilibre à $\mathcal{S}$ et $\overline{\mathcal{S}}$. Le déroulement de l'algorithme peut alors reprendre normalement.

Marge vide

Lors du développement présenté jusqu'alors, nous avons supposé $\mathcal{M}$ et $\mathcal{S}$ non-vides. Il est très improbable que $\mathcal{S}$ se dépeuple totalement au cours du déroulement de l'algorithme, car cela signifierait qu'avec les dimensions courantes ($\mathcal{S} \cup \overline{\mathcal{S}}$), le meilleur séparateur est l'application $\mathbf{x} \in \mathcal{X} \mapsto -b$, qui assigne indifféremment l'étiquette $\mathrm{Signe}(-b)$ à tous les points. En revanche, il n'est pas impossible qu'une situation dans laquelle aucun point n'est sur la marge apparaisse [2].

Dans ce cas, on procède de la même façon que celle décrite dans [Karasuyama et Takeuchi, 2010]. Les conditions d'optimalité sont :

$$\begin{aligned} \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{I}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}} = \boldsymbol{\mu}_{\boldsymbol{\mathcal{S}}} \quad (\mathbf{w}_{\mathcal{S}} > 0) \qquad & \qquad \boldsymbol{Q}_{\overline{\boldsymbol{\mathcal{S}}},\boldsymbol{\mathcal{I}}}\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}} < \boldsymbol{\mu}_{\overline{\boldsymbol{\mathcal{S}}}} \quad (\mathbf{w}_{\overline{\mathcal{S}}} = 0) \\ \boldsymbol{y}_{\boldsymbol{\mathcal{I}}}{}^T\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}} = 0 \quad (b \in \mathbb{R}) \qquad & \qquad \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{I}}}{}^T\mathbf{w}_{\boldsymbol{\mathcal{S}}} + b\boldsymbol{y}_{\boldsymbol{\mathcal{I}}} < \mathbb{1} \quad (\boldsymbol{\alpha}_{\boldsymbol{\mathcal{I}}} = C\mathbb{1}) \\ & \qquad \boldsymbol{Q}_{\boldsymbol{\mathcal{S}},\boldsymbol{\mathcal{O}}}{}^T\mathbf{w}_{\boldsymbol{\mathcal{S}}} + b\boldsymbol{y}_{\boldsymbol{\mathcal{O}}} > \mathbb{1} \quad (\boldsymbol{\alpha}_{\boldsymbol{\mathcal{O}}} = 0). \end{aligned}$$

2. En pratique, nous n'avons jamais observé ce cas pathologique.

On remarque que b n'est alors pas déterminé de manière unique. En effet, on déduit des inégalités précédentes que :

$$\max_{i\in\mathcal{B}_-} y_i(1 - \boldsymbol{Q_{S,i}}^T \mathbf{w_S}) < b < \min_{i\in\mathcal{B}_+} y_i(1 - \boldsymbol{Q_{S,i}}^T \mathbf{w_S}),$$

où :

$$\begin{aligned}\mathcal{B}_- &= \{i \in \mathcal{I}, y_i = -1\} \cup \{i \in \mathcal{O}, y_i = 1\}\\ \mathcal{B}_+ &= \{i \in \mathcal{I}, y_i = 1\} \cup \{i \in \mathcal{O}, y_i = -1\}\,.\end{aligned}$$

On peut alors choisir b de manière à insérer un point sur la marge et outrepasser ainsi ce cas pathologique.

Algorithme

L'algorithme 5 récapitule et ordonne les éléments nécessaires à l'apprentissage d'un détecteur linéaire par incréments dimensionnels. Même si nous n'avons aucune preuve de convergence de cet algorithme, les expérimentations numériques que nous avons conduites (section 4.5) montrent qu'il fournit systématiquement une solution satisfaisant les conditions de Karush-Kuhn-Tucker (KKT), en un nombre fini d'itérations.

Bien que fondamentalement différent de l'approche par contraintes actives, cet algorithme est construit à partir des mêmes conditions d'optimalité du premier ordre, tirant ainsi parti des dimensions actives et inactives du détecteur à chaque étape de l'apprentissage. On espère donc qu'il sera apte à entraîner rapidement un détecteur linéaire en grande dimension.

4.3.4 Complexité du modèle

Jusqu'à présent, nous avons présenté un modèle de détecteur précoce ainsi qu'une manière permettant de déterminer ses paramètres (*i.e.* calculer un hyperplan séparateur et sélectionner des prototypes pertinents parmi un grand ensemble d'instantanés mis à notre disposition). Cette approche est très semblable à une SVM régularisée par une norme ℓ_1, de sorte qu'elle bénéficie de garanties de généralisation comparables. Nous revenons donc ici sur la théorie de l'apprentissage statistique (section 1.2) afin de montrer comment des résultats existants peuvent s'appliquer au cadre construit.

Les espaces de similarité ont très tôt été accompagnés de théories sur les aptitudes des mesures de proximité (*goodness of a similarity function*) et sur les capacités de généralisation que l'on peut en attendre [Balcan et Blum, 2006, Kar et Jain, 2012]. Au contraire, nous nous intéressons ici à une approche plus générale, qui ne fait aucune supposition sur la fonction de similarité. Celle-ci découle directement des travaux présentés dans [Kakade *et coll.*, 2009], qui fournissent un ensemble d'outils théoriques pour analyser les SVM linéaires.

Si l'on reprend le formalisme introduit dans la section 1.2 et l'argument d'équivalence entre les formulations de Tikhonov et d'Ivanov pour les problèmes d'optimisation convexes [Tikhonov et Arsenin, 1977], le problème d'apprentissage (convexe) (4.1) d'un détecteur précoce $(\mathbf{w}, b)$ peut s'écrire :

$$\underset{f\in\mathcal{F}}{\text{minimiser}} \quad \frac{1}{n}\sum_{i=1}^{n} L\left(y_i, f\left(\psi_T^{\mathcal{L}}(x_{\sim}^{(i)})\right)\right),$$

où $L\colon (a,b) \in \mathbb{R}\times\mathbb{R} \mapsto \max(0, 1-ab)$ est la fonction de perte charnière et la classe des détecteurs précoces est définie à partir d'un majorant positif c_1 par :

$$\mathcal{F} = \{\mathbf{x} \mapsto \langle \mathbf{w} \mid \mathbf{x}\rangle_{\ell_2} - b,\ \mathbf{w} \in \mathbb{R}_+^r, b \in \mathbb{R}, (1-\lambda)\left\|\mathbf{w}\right\|_{\boldsymbol{\mu},\ell_1} + \lambda\left\|b\right\|_{\ell_1} \leq c_1\}.$$

Données : matrice $\boldsymbol{Q}$, vecteurs $\boldsymbol{\mu}$, $\boldsymbol{\alpha}$, $\mathbf{w}$, seuil b et ensembles $\mathcal{I}$, $\mathcal{M}$, $\mathcal{O}$, $\mathcal{S}$, $\overline{\mathcal{S}}$ et $\mathcal{A}$.
retourner *vecteurs* $\boldsymbol{\alpha}$, $\mathbf{w}$ *et seuil* b *à l'équilibre.*
$\mathcal{W} \leftarrow \emptyset$;
$\mathcal{A}^* \leftarrow$ restriction de $\mathcal{A}$ vérifiant : $\boldsymbol{Q}_{\mathcal{A}^*,\mathcal{I}\cup\mathcal{M}}\boldsymbol{\alpha}_{\mathcal{I}\cup\mathcal{M}} \leq \boldsymbol{\mu}_{\mathcal{A}^*}$;
$\mathcal{A} \leftarrow \mathcal{A}\backslash\mathcal{A}^*$;
affecter des éléments de $\mathcal{A}^*$ à $\mathcal{S}$ et $\overline{\mathcal{S}}$;
tant que $\mathcal{A}$ *non-vide* **faire**
  $\boldsymbol{\phi} \leftarrow$ solution de (4.4) ;
  $\eta \leftarrow$ pas maximum admissible ;
  si *aucun risque de cycle* **alors**
    $[b,\ \boldsymbol{\alpha}_{\mathcal{M}},\ \mathbf{w}_{\mathcal{S}}]^T \leftarrow [b,\ \boldsymbol{\alpha}_{\mathcal{M}},\ \mathbf{w}_{\mathcal{S}}]^T + \eta\boldsymbol{\phi}$;
    $\boldsymbol{\alpha}_{\mathcal{A}} \leftarrow \boldsymbol{\alpha}_{\mathcal{A}} + \eta\mathbb{1}$;
    mettre à jour les ensembles $\mathcal{I}$, $\mathcal{M}$, $\mathcal{O}$, $\mathcal{S}$, $\overline{\mathcal{S}}$ et $\mathcal{A}$ suivant l'événement apparu ;
  sinon
    si *le risque est sur un point peuplant la marge* **alors**
      lancer une procédure de pénétration ;
    si *le risque est sur une dimension (notée* $*$*)* **alors**
      $\mathcal{W} \leftarrow \mathcal{W} \cup \{*\}$;
      $\overline{\mathcal{S}} \leftarrow \overline{\mathcal{S}}\backslash\{*\}$;
  si $\mathcal{A}$ *est vide* **alors**
    $\mathcal{A}^* \leftarrow$ restriction de $\mathcal{W}$ vérifiant : $\boldsymbol{Q}_{\mathcal{A}^*,\mathcal{I}\cup\mathcal{M}}\boldsymbol{\alpha}_{\mathcal{I}\cup\mathcal{M}} \leq \boldsymbol{\mu}_{\mathcal{A}^*}$;
    $\mathcal{A} \leftarrow \mathcal{W}\backslash\mathcal{A}^*$;
    affecter des éléments de $\mathcal{A}^*$ à $\mathcal{S}$ et $\overline{\mathcal{S}}$;

Algorithme 5 : Apprentissage du détecteur linéaire par incréments dimensionnels.

Cette formulation du problème d'optimisation nous rapproche des travaux de Kakade *et coll.* concernant les bornes de généralisation des SVM linéaires. Ces derniers présentent notamment une majoration de la complexité de Rademacher de ce type de séparateurs, via un théorème que nous donnons à présent, après avoir rappelé la notion de forte convexité.

Définition 4.3.1 (Fonction fortement convexe [Kakade *et coll.*, 2009])**.**
Une fonction $f\colon \mathcal{X} \to \mathbb{R}$ est dite σ-fortement convexe ($\sigma \in \mathbb{R}_+^*$) par rapport à une norme $\|\cdot\|_*$ ssi :

$$\forall(\boldsymbol{x},\boldsymbol{z}) \in \mathcal{X}^2, \forall\lambda \in [0,1]\colon f(\lambda\boldsymbol{x} + (1-\lambda)\boldsymbol{z}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda)f(\boldsymbol{z}) - \frac{\sigma}{2}\lambda(1-\lambda)\|\boldsymbol{x}-\boldsymbol{z}\|_*^2.$$

Cette définition est équivalente à affirmer que $\boldsymbol{x} \mapsto f(\boldsymbol{x}) - \frac{\sigma}{2}\|\boldsymbol{x}\|_*^2$ est convexe.

Théorème 4.3.1 (Borne de complexité [Kakade *et coll.*, 2009])**.**
Soient $\|\cdot\|$ *une norme et* $\|\cdot\|_*$ *sa norme duale. Supposons que :*

$$\exists c_{\mathcal{X}} \in \mathbb{R}_+ \ /\ \forall \boldsymbol{x} \in \mathcal{X},\ \|\boldsymbol{x}\| \leq c_{\mathcal{X}}.$$

Soient maintenant $\mathcal{S}$ *un ensemble fermé et convexe (de même dimension que* $\mathcal{X}$*) et* $F\colon \mathcal{S} \to \mathbb{R}$ *une fonction* σ*-fortement convexe (*$\sigma > 0$*) par rapport à* $\|\cdot\|_*$ *telle que* $\inf_{\boldsymbol{x}\in\mathcal{S}} F(\boldsymbol{x}) = 0$*. Soient alors* $c_{\mathcal{W}}$ *un réel positif et* $\mathcal{W}$ *un sous-ensemble de* $\mathcal{S}$ *vérifiant :*

$$\mathcal{W} = \{\mathbf{w} \in \mathcal{S}\ /\ F(\mathbf{w}) \leq c_{\mathcal{W}}^2\}.$$

Nous définissons la classe de fonctions d'intérêt $\mathcal{F}_{\mathcal{W}}$ *par* $\mathcal{F}_{\mathcal{W}} = \{\mathbf{x} \mapsto \langle \mathbf{w} \mid \mathbf{x}\rangle_{\ell_2},\ \mathbf{w} \in \mathcal{W}\}$*. Alors la complexité de Rademacher de cette classe de fonctions est bornée de la manière suivante :*

$$\mathcal{R}_n(\mathcal{F}_{\mathcal{W}}) \leq c_{\mathcal{X}} c_{\mathcal{W}} \sqrt{\frac{2}{\sigma n}}.$$

La première hypothèse de ce théorème est automatiquement vérifiée puisque nous nous sommes assurés que $\mathcal{X}$ est un espace compact. Le reste du théorème est assez général et suppose que l'on travaille avec des séparateurs linéaires $\langle \mathbf{w} \mid \cdot \rangle_{\ell_2}$ dont une mesure $F(\mathbf{w})$ est bornée. Nous énonçons à présent le corollaire qui établit le lien entre ce théorème et les SVM régularisées en norme ℓ_1.

Corollaire 4.3.2 (Borne de complexité pour contrainte ℓ_1 [Kakade *et coll.*, 2009])**.**
Soient c_1 un nombre réel positif et $\mathcal{F} = \{\mathbf{x} \mapsto \langle \mathbf{w} \mid \mathbf{x} \rangle_{\ell_2}, \mathbf{w} \in \mathbb{R}^r_+, \|\mathbf{w}\|_{\ell_1} \leq c_1\}$. Si

$$\exists c_\infty \in \mathbb{R}_+ \,/\, \forall \boldsymbol{x} \in \mathcal{X},\ \|\boldsymbol{x}\|_\infty \leq c_\infty,$$

alors

$$\mathcal{R}_n(\mathcal{F}) \leq c_1 c_\infty \sqrt{\frac{2(\ln(r) + \mathrm{e}^{-1})}{n}}.$$

Dans [Kakade *et coll.*, 2009], les auteurs énoncent ce corollaire sans le terme e^{-1}, qui nous semble manquer. Nous reprenons donc leur démonstration en détaillant les étapes.

Démonstration. Choisissons la norme $\|\cdot\| = \|\cdot\|_\infty$ et sa norme duale $\|\cdot\|_* = \|\cdot\|_1$. Soit alors c_1 un réel positif. Prenons $\mathcal{S} = \{\mathbf{w} \in \mathbb{R}^r \,/\, \mathbf{w} \succcurlyeq 0, \|\mathbf{w}\|_{\ell_1} \leq c_1\}$ ($\mathcal{S}$ est fermé et convexe) et considérons la fonction entropie :

$$F \colon \mathbf{w} \in \mathcal{S} \mapsto \sum_{j=1}^{r} \frac{w_j}{c_1} \ln\left(\frac{r w_j}{c_1}\right) + \mathrm{e}^{-1}.$$

Démontrons que F est $\frac{1}{c_1^2}$-fortement convexe par rapport à $\|\cdot\|_{\ell_1}$ et que $\inf_{\mathbf{w} \in S} F(\mathbf{w}) = 0$.

Tout d'abord, F est doublement dérivable sur tout $\mathcal{S}$ sauf là où une coordonnée de $\mathbf{w}$ s'annule. On vérifie aisément que la hessienne $\nabla^2 F$ de F est $(\delta_{p,q} \frac{1}{c_1 w_p})_{1 \leq p,q \leq r}$ (où $\delta_{p,q}$ est le dirac valant 1 ssi $p = q$). Chaque élément sur la diagonale est minoré par $\frac{1}{c_1^2}$ puisque $0 \leq w_j \leq c_1$, $\forall j \in \mathbb{N}_r$. Ainsi $\nabla^2 F - \frac{1}{c_1^2} I$ est semi-définie positive, ce qui prouve la convexité de $F - \frac{1}{2c_1^2} \|\cdot\|^2_{\ell_1}$ et la forte convexité de F.

Considérons maintenant le problème $\min_{\mathbf{w} \in \mathbb{R}^d_+} \sum_{j=1}^r \frac{w_j}{c_1} \ln\left(\frac{r w_j}{c_1}\right)$. Par annulation du gradient, on obtient un unique minimum global en $\mathbf{w}^* = \frac{c_1}{r} \mathrm{e}^{-1} \mathbb{1}$, valant $-\mathrm{e}^{-1}$. Puisque $\mathbf{w}^* \in \mathcal{S}$, $\mathbf{w}^* = \arg\min_{\mathbf{w} \in \mathcal{S}} F(\mathbf{w})$ et $F(\mathbf{w}^*) = 0$.

Définissons à présent $\mathcal{W} = \{\mathbf{w} \in S \,/\, F(\mathbf{w}) \leq c_{\mathcal{W}}^2\}$ pour $c_{\mathcal{W}} = \sqrt{\ln(r) + \mathrm{e}^{-1}}$. Puisque $\forall \mathbf{w} \in \mathcal{S}$, $\forall j \in \mathbb{N}_r$, $0 \leq w_j \leq c_1$, $0 \leq \frac{r w_j}{c_1} \leq r$, puis $\sum_{j=1}^r \frac{w_j}{c_1} \ln\left(\frac{r w_j}{c_1}\right) \leq \ln(r) \sum_{j=1}^r \frac{w_j}{c_1} \leq \ln(r)$ (on utilise la continuité là où les coordonnées de $\mathbf{w}$ s'annulent). D'où $\forall \mathbf{w} \in \mathcal{S}$, $F(\mathbf{w}) \leq c_{\mathcal{W}}^2$. Ainsi $\mathcal{W} = \mathcal{S}$ et $\mathcal{F} = \{\mathbf{x} \mapsto \langle \mathbf{w} \mid \mathbf{x} \rangle_{\ell_2}, \mathbf{w} \in \mathcal{S}\} = \mathcal{F}_{\mathcal{W}}$ (tel que défini dans l'énoncé du théorème 4.3.1). On peut donc appliquer la majoration du théorème 4.3.1 en prenant $c_{\mathcal{X}} = c_\infty$, $c_{\mathcal{W}} = \sqrt{\ln(r) + \mathrm{e}^{-1}}$ et $\sigma = \frac{1}{c_1^2}$. ■

Forts de ce corollaire, on obtient immédiatement une majoration de la complexité de Rademacher de notre classe de détecteurs précoces.

Corollaire 4.3.3 (Borne de complexité pour détecteur précoce)**.**
Soient $r > 0$ la dimension des données, $\boldsymbol{\mu} \in \mathbb{R}^r_+$ un vecteur de pondération, $\lambda \in]0, 1[$, c_1 un nombre réel positif et $\mathcal{F}$ l'ensemble des détecteurs précoces considérés :

$$\mathcal{F} = \{\mathbf{x} \mapsto \langle \mathbf{w} \mid \mathbf{x} \rangle_{\ell_2} - b,\ \mathbf{w} \in \mathbb{R}^r_+, b \in \mathbb{R}, (1-\lambda) \|\mathbf{w}\|_{\boldsymbol{\mu},\ell_1} + \lambda \|b\|_{\ell_1} \leq c_1\}.$$

Si la plus petite composante de $\boldsymbol{\mu}$ est 1 et si :

$$\exists c_\infty \geq 1 \;/\; \forall x_\sim \in \mathcal{X}^{[0,T]},\, \forall t \in [0,T]\colon\; \left\|\psi_t^{\mathcal{L}}(x_\sim)\right\|_\infty \leq c_\infty,$$

alors :

$$\mathcal{R}_n(\mathcal{F}) \leq \frac{c_1 c_\infty}{\min(\lambda, 1-\lambda)} \sqrt{\frac{2(\ln(r+2) + \mathrm{e}^{-1})}{n}}.$$

Démonstration. Soient $\boldsymbol{\mu}$, λ, c_1 et $\mathcal{F}$ tels que définis dans l'énoncé. Soit aussi la matrice $A_{\boldsymbol{\mu},\lambda} \in \mathbb{R}^{(r+2)\times(r+2)}$ définie par $A_{\boldsymbol{\mu},\lambda} = \mathrm{Diag}([(1-\lambda)\mu_1, \ldots, (1-\lambda)\mu_r, \lambda, \lambda])$. Alors en effectuant les changements de variables $\hat{\mathbf{w}} = [\mathbf{w}, u, v]$ (avec $b = u - v$ et $u, v \geq 0$) et $\hat{\mathbf{x}} = [\psi_t^{\mathcal{L}}(x_\sim), -1, 1]$, on obtient :

$$\mathcal{F} = \{\hat{\mathbf{x}} \mapsto \langle \hat{\mathbf{w}} \mid \hat{\mathbf{x}} \rangle_{\ell_2}\,,\; \hat{\mathbf{w}} \in \mathbb{R}_+^{r+2}, \|A_{\boldsymbol{\mu},\lambda}\hat{\mathbf{w}}\|_{\ell_1} \leq c_1\}.$$

Notons d'abord qu'avec ce changement de variable et les hypothèses du corollaire 4.3.3, les données augmentées sont toujours majorées par c_∞ (car $c_\infty \geq 1$) :

$$\|\hat{\mathbf{x}}\|_\infty \leq \max(\left\|\psi_t^{\mathcal{L}}(x_\sim)\right\|_\infty, 1) \leq c_\infty.$$

Ensuite, puisque la plus petite valeur de $\boldsymbol{\mu}$ est 1 et que $1 \leq \frac{1-\lambda}{\min(\lambda,1-\lambda)}$, on a pour tout j de $\mathbb{N}_r$: $1 \leq \frac{1-\lambda}{\min(\lambda,1-\lambda)}\mu_j$, puis $\hat{\mathbf{w}}_j \leq \frac{1-\lambda}{\min(\lambda,1-\lambda)}\mu_j\hat{\mathbf{w}}_j$ ($\hat{\mathbf{w}}_j \geq 0$). De même, puisque $1 \leq \frac{\lambda}{\min(\lambda,1-\lambda)}$, alors $\forall j \in [\![r+1, r+2]\!]\colon \hat{\mathbf{w}}_j \leq \frac{\lambda}{\min(\lambda,1-\lambda)}\hat{\mathbf{w}}_j$. Ainsi $\hat{\mathbf{w}} \preccurlyeq \frac{1}{\min(\lambda,1-\lambda)}A_{\boldsymbol{\mu},\lambda}\hat{\mathbf{w}}$. On en déduit que $\mathbb{1}^T\hat{\mathbf{w}} \leq \frac{1}{\min(\lambda,1-\lambda)}\mathbb{1}^T A_{\boldsymbol{\mu},\lambda}\hat{\mathbf{w}}$, *i.e.*

$$\|\hat{\mathbf{w}}\|_{\ell_1} \leq \frac{1}{\min(\lambda, 1-\lambda)} \|A_{\boldsymbol{\mu},\lambda}\hat{\mathbf{w}}\|_{\ell_1} \leq \frac{c_1}{\min(\lambda, 1-\lambda)}.$$

Il suffit maintenant d'appliquer le corollaire 4.3.2 en notant que la dimension des données $\hat{\mathbf{x}}$ est $r+2$ et que la borne de contrôle de la complexité de $\mathcal{F}$ est $\frac{c_1}{\min(\lambda,1-\lambda)}$. ■

Comme mentionné dans [Kakade *et coll.*, 2009], ce type de majoration sert de substitut dans le théorème 1.2.2 [Bartlett et Mendelson, 2002] afin d'obtenir une borne de généralisation pour l'ensemble des détecteurs précoces $\mathcal{F}$.

Remarque 18.

La borne de généralisation obtenue par majoration de la complexité de Rademacher de l'ensemble des détecteurs précoces $\mathcal{F}$ dépend de la dimension des données (*i.e.* du nombre r de prototypes), qui d'une part est susceptible d'être très grande et d'autre part contredit l'une des propriétés de la théorie SVM qui est précisément de ne pas dépendre de ladite dimension. Deux observations peuvent alors être émises :

- ◇ cette dépendance est faible puisqu'en $\ln(r+2)$;
- ◇ cette borne est vérifiée pour tout estimateur f de $\mathcal{F}$. En revanche, nous ne nous intéressons en réalité qu'à un estimateur optimal f^* résolvant le problème d'apprentissage. Or la norme ℓ_1 favorise la parcimonie, donc le support du vecteur de pondération $\mathbf{w}$ de f^* est restreint et la dimension effective de $\mathbf{w}$ est donc bien inférieure à r en pratique.

Ayant montré que la similarité de notre approche avec les SVM linéaires permet de donner des garanties théoriques de généralisation au cadre mis en place, nous retournons à présent à des aspects numériques. Plus précisément, la prochaine section est vouée à mettre en lumière les points communs et les différences de notre détecteur précoce en comparaison à deux approches existantes. Finalement, une ultime partie permet de valider empiriquement nos travaux.

4.4 DISCUSSION

Apprentissage d'instances multiples

Le cadre de détection précoce que nous avons présenté est construit sur l'exhibition d'instantanés discriminants au sein d'une séquence. Concrètement, si un tel instantané est découvert dans une séquence, alors celle-ci est étiquetée par $+1$. Sinon, elle est attribuée à la classe -1. Cette règle d'attribution est similaire à celle du paradigme MIL, présenté dans la section 1.5, dans lequel une instance est associée à un instantané et un sac à une série temporelle.

Concrètement, le cadre présenté dans ce chapitre est proche de celui introduit par Chen *et coll.*, nommé apprentissage d'instances multiples avec sélection intégrée des instances (*Multiple-Instance Learning via Embedded instance Selection*, MILES) [Chen *et coll.*, 2006]. Comme nous l'avons rapidement vu dans la section 1.5, MILES consiste à utiliser un espace de représentation afin de réduire un problème MIL à une simple SVM régularisée par une norme ℓ_1. De la même façon que ce que nous avons présenté jusqu'ici, MILES utilise toutes les instances comme prototypes et sélectionne celles qui sont discriminantes grâce à la régularisation parcimonieuse. Plusieurs différences existent pourtant entre notre détecteur précoce et MILES.

Premièrement, les instances que nous considérons (les instantanés d'une séquence) sont ordonnées par le temps. C'est cet ordre qui nous conduit à introduire la notion de temporalité dans la mesure de similarité, tout comme celle de précocité et de fiabilité. Ces trois concepts sont absents des travaux de Chen *et coll.*, si bien que leur problème d'apprentissage est une simple SVM régularisée en norme ℓ_1 lorsque le notre contient deux modifications majeurs : la pondération de la norme (qui a un sens physique dans le cadre de la détection précoce) et la positivité des poids (assurant la fiabilité de la décision). De même, comme nous le verrons dans la section suivante, nous proposons un algorithme par ensemble actif simple mais apportant un réel gain de temps d'apprentissage, sans perdre en qualité sur la solution obtenue. Au contraire, Chen *et coll.* ne discutent pas particulièrement la résolution du LP mis en place pour l'apprentissage.

Deuxièmement, la mesure de similarité utilisée par MILES est identique à celle présentée dans l'exemple de la section 4.2.3 (avec $p \to +\infty$), prenant le maximum de plusieurs évaluations d'un noyau gaussien. Celle-ci découle naturellement des hypothèses statistiques réalisées dans [Maron et Lozano-Pérez, 1998] et reprises dans le cadre de MILES. Cette mesure de proximité, que nous avons adoptée pour nos expériences numériques, est un choix judicieux. D'une part car le noyau gaussien est réputé pour sa malléabilité et ses très bons résultats empiriques ; d'autre part car considérer le maximum d'un ensemble de valeurs est une manière robuste d'agréger les similarités au fil du temps. Toutefois, le cadre que nous avons présenté est plus large et offre les conditions faibles nécessaires (la causalité et la croissance) pour utiliser d'autres mesures de proximité.

Les travaux de Chen *et coll.* sont capitaux dans le développement du paradigme MIL, au même titre que ceux présentés dans [Andrews *et coll.*, 2003]. Ces différentes déclinaisons des SVM, visant à répondre à l'ambiguïté intrinsèque du problème MIL, ont été appliquées à la reconnaissance audio [Mandel et Ellis, 2008]. Dans cet article, les auteurs relèvent les différences de granularité des métadonnées des bases musicales (par exemple, on indique souvent le genre musical d'un album, mais pas des chansons qui le composent). MIL est alors utilisé pour inférer des informations à une granularité plus fine que celle à disposition. Les applications numériques semblent privilégier l'approche mi-SVM, proposée par Andrews *et coll.*, à MILES.

Plus récemment, le paradigme MIL a été placé au cœur d'un problème de reconnaissance

d'actions vidéos [Ellis *et coll.*, 2013]. Dans cet ouvrage, les auteurs présentent une approche probabiliste de MIL, adaptée à la nature des séries temporelles (voir section 2.5). La log-vraisemblance est alors déclinée à différents instants t pour forcer la précocité de la prise de décision (bien que les auteurs ne discutent pas la fiabilité de leur système). Cette technique correspond à augmenter virtuellement l'ensemble des séries d'apprentissage à l'aide de séquences partielles. Comme nous allons le détailler à présent, elle a déjà été utilisée dans [Hoai et De la Torre, 2012] pour les mêmes raisons.

Détecteur précoce vaste marge

Le détecteur précoce vaste marge (*Maximum Margin Early Detector*, MMED), introduit dans [Hoai et De la Torre, 2012, Hoai et De la Torre, 2014] et expliqué dans la section 2.5, est un modèle visant à détecter et à localiser un événement à l'intérieur d'une série temporelle. Il est entraîné par le biais d'une SVM modifiée. Pour le comparer à notre approche, nous mettons de côté l'aspect de localisation en réutilisant le contexte mis en place dans ce chapitre (qui suppose que chaque exemple d'apprentissage est un événement, étiqueté $+1$, ou un non-événement, étiqueté -1), et nous ré-écrivons le problème d'optimisation dans l'espace de similarité (introduisant ainsi la notion d'instance multiple à MMED) :

$$\begin{array}{ll} \underset{\mathbf{w},\boldsymbol{\xi},b}{\text{minimiser}} & \dfrac{1}{2}\left\|\mathbf{w}\right\|_{\ell_2}^2 + C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \begin{cases} y_i\left(\left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{\tau\Delta t}(x^{(i)}_{\sim})\right\rangle_{\ell_2} - b\right) \geq 1 - \dfrac{\xi_i}{g(\tau)}, \quad \forall i \in \mathbb{N}_n, \quad \forall \tau \in \mathbb{N}_{\lfloor \frac{T}{\Delta t} \rfloor} \\ \left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{\tau\Delta t}(x^{(i)}_{\sim}) - \psi^{\mathcal{L}}_{\tau'\Delta t}(x^{(i)}_{\sim})\right\rangle_{\ell_2} \geq \Delta(\tau,\tau') - \dfrac{\xi_i}{g(\tau)}, \\ \qquad\qquad \forall i \in \mathbb{N}_n \,/\, y_i = 1, \quad \forall \tau \in \mathbb{N}_{\lfloor \frac{T}{\Delta t} \rfloor}, \quad \forall \tau' \in [\![0,\tau]\!] \\ 0 \preccurlyeq \boldsymbol{\xi}, \end{cases} \end{array} \tag{4.5}$$

où $g : \mathbb{N}_{\lfloor \frac{T}{\Delta t} \rfloor} \mapsto]0,1]$ est une fonction croissante (les auteurs proposent un modèle linéaire par morceaux) et $\Delta(\tau,\tau') = 1 - \frac{2\min(\tau,\tau')}{\tau+\tau'}$, représentant une pénalité croissante avec l'écart $\tau - \tau'$.

Dans la formulation du problème (4.5), le temps a été discrétisé (conformément aux travaux présentés dans [Hoai et De la Torre, 2014]), remplaçant ainsi l'instant t par $\tau\Delta t$. Par rapport à notre formulation (4.1), cette discrétisation est nécessaire puisque le problème (4.5) fait intervenir les séquences d'apprentissage partiellement observées, à chaque instant $\tau\Delta t$. Ainsi, à l'instar d'une SVM usuelle, la première contrainte de (4.5) assure que chaque séquence partiellement observée est bien classée (avec un écart autorisé $\frac{\xi_i}{g(\tau)}$). Puisque g est une fonction croissante du temps, cette première contrainte autorise plus facilement les erreurs de classification sur les séquences peu observées que largement acquises. La deuxième contrainte de (4.5) n'apparaît pas dans le paradigme SVM. Celle-ci a pour but de forcer la décision du détecteur $\left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{\tau\Delta t}(x^{(i)}_{\sim})\right\rangle_{\ell_2} - b$ à croître au cours de l'analyse d'un événement (en effet, cette contrainte n'est effective que pour les séquences d'étiquette $+1$). En pratique, la valeur de la fonction de décision à un instant $\tau\Delta t$ doit être supérieure à toutes celles des instants passés $\tau'\Delta t$ ($\tau' \in [\![0,\tau]\!]$), avec une *marge* $\Delta(\tau,\tau')$ et une erreur autorisée $\frac{\xi_i}{g(\tau)}$.

MMED ne fait pas intervenir la notion d'instantané discriminant, que nous supposons dans notre modèle. Toutefois, des ressemblances existent évidemment entre les deux approches puisque les apprentissages découlent du paradigme SVM. Afin de mettre en relief notre approche par rapport à MMED, nous dressons à présent la liste des différences majeures :

- ◇ la première contrainte de (4.5) dépend du temps, lorsque la contrainte équivalente dans notre problème d'apprentissage (4.1) considère uniquement les vecteurs caractéristiques à l'instant final T. Ceci est l'une des manifestations de l'augmentation

virtuelle de l'ensemble d'apprentissage réalisée par MMED, conduisant à un grand nombre de contraintes ;

- ◇ la pénultième contrainte de (4.5) est fonction du temps $\tau\Delta t$ mais aussi de tous les instants précédents (indexés par τ', pour $\tau' \in [\![0, \tau]\!]$). Cette caractéristique du problème (4.5) augmente grandement le nombre de contraintes puisqu'elle se décline de manière combinatoire pour chaque couple (τ, τ'). On peut, toutefois, remarquer que dans une situation idéale ($\xi_i = 0$), la partie de droite de la pénultième contrainte de (4.5) est positive, conduisant à l'inégalité :

$$\left\langle \mathbf{w} \mid \psi^{\mathcal{L}}_{\tau\Delta t}(x^{(i)}_{\sim}) - \psi^{\mathcal{L}}_{\tau'\Delta t}(x^{(i)}_{\sim}) \right\rangle_{\ell_2} \geq 0, \quad \forall \tau \in \mathbb{N}_{\lfloor \frac{T}{\Delta t} \rfloor}, \forall \tau' \in [\![0, \tau]\!]. \tag{4.6}$$

  De manière informelle, puisque cette inégalité doit être vérifiée pour un nombre combinatoire de différences de vecteurs caractéristiques, elle est *quasiment* équivalente à $\langle \mathbf{w} \mid \mathbf{x} \rangle_{\ell_2} \geq 0, \forall \mathbf{x} \in \mathbb{R}^r_+$ (la positivité provient de la croissance de la représentation par similarités $\psi^{\mathcal{L}}_{\sim}$), *i.e.* $\mathbf{w} \succcurlyeq 0$. Réciproquement, si $\mathbf{w} \succcurlyeq 0$, la relation (4.6) est bien vérifiée. Il apparaît ainsi que la pénultième contrainte de MMED est une version dirigée par les données de $\mathbf{w} \succcurlyeq 0$ (ou inversement que notre contrainte est une version déterministe), qui incorpore une marge souple $\Delta(\tau, \tau') - \frac{\xi_i}{g(\tau)}$ pour améliorer la généralisation tout en autorisant les erreurs de fiabilité. Au contraire, notre approche déterministe assure automatiquement la fiabilité du détecteur sur des séquences inédites. En pratique, il est difficile d'affirmer que notre contrainte est plus restrictive que celle de MMED ;

- ◇ enfin, si certaines garanties théoriques sont données dans [Hoai et De la Torre, 2014] sur l'intérêt de contraindre le programme d'optimisation (4.5) de la sorte, aucune borne de généralisation n'est exposée. Au contraire, le cadre que nous proposons s'inscrit directement dans les travaux de Bartlett, Kakade *et coll.*, fournissant une majoration de l'erreur réelle par la complexité de Rademacher de la classe de détecteurs utilisés.

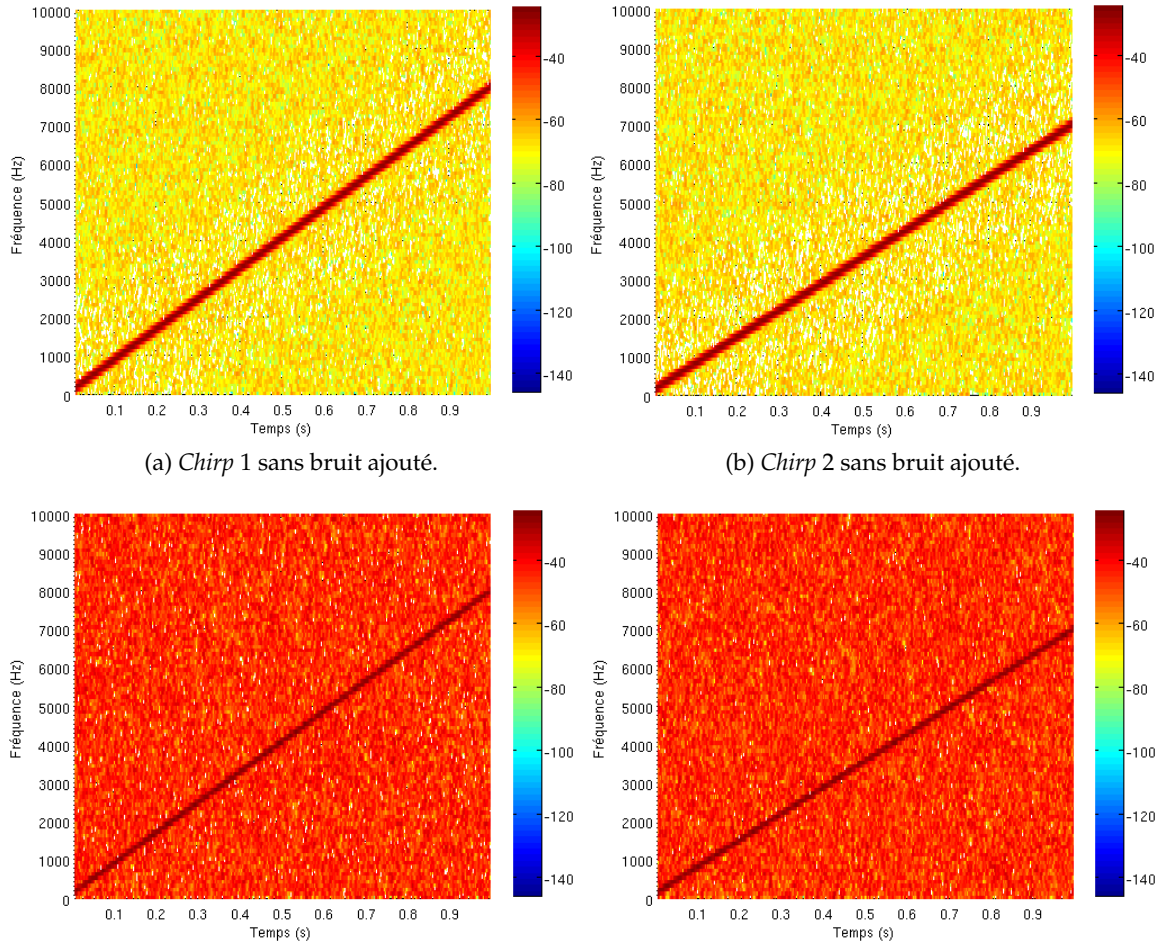
4.5 EXPÉRIENCES NUMÉRIQUES

Nous rapportons ici plusieurs expériences numériques ayant pour but de valider le modèle de détecteur et la méthode d'apprentissage présentés. Dans un premier temps, on s'assurera donc que l'apprentissage par ensemble actif offre un réel gain de temps et on évaluera l'algorithme incrémental. Ensuite, nous observerons la bonne capacité de détection de notre approche sur des données réelles, en la comparant à celle de MILES et de MMED. Enfin, l'impact de la pondération de la norme ℓ_1 comme pénalisation d'une prise de décision tardive sera évalué.

Dans toutes les expériences, pour MILES et notre détecteur, nous avons fixé $\lambda = 0,1$; excepté dans l'algorithme incrémental, mis en œuvre uniquement dans le cas $\lambda = 0$.

4.5.1 Comparaison des approches de résolution

Cette section compare simplement la résolution du problème d'optimisation (4.2) par une méthode directe (utilisant *lpsolve*) et par les algorithmes 4 (par contraintes actives) et 5 (incrémental). La comparaison est effectuée sur un jeu de données simulées, constitué de deux classes construites à partir de formes de base qui sont légèrement et aléatoirement translatées, puis auxquelles est ajouté un bruit blanc gaussien. Les deux formes considérées (voir figure 4.7 page suivante), générant chacune une classe, sont des *chirps* linéaires (l'un entre 100 Hz et 8 kHz, l'autre entre 100 Hz et 7 kHz). Les séquences ainsi générées sont ensuite



(a) *Chirp* 1 sans bruit ajouté.

(b) *Chirp* 2 sans bruit ajouté.

(c) *Chirp* 1 avec bruit ajouté.

(d) *Chirp* 2 avec bruit ajouté.

FIGURE 4.7 – Spectrogrammes des *chirps* constituant le jeu de données synthétique (avant et après l'ajout de bruit). L'échelle est donnée en décibels.

décrites dans un espace de similarité identique à celui présenté dans l'exemple de la section 4.2.3 avec $p \to +\infty$ (*i.e.* utilisant une mesure de proximité gaussienne et une agrégation par maximum).

Dans l'algorithme par ensemble actif, le sous-problème est lui aussi résolu par *lpsolve* mais ne possède jamais plus de 100 contraintes duales à $\mathbf{w}$ (*i.e.* $\mathrm{Card}(\mathcal{A}) \leq 100$). De plus, la pondération de la norme est fixée à $\boldsymbol{\mu} = \mathbb{1}$, de sorte que $\|\cdot\|_{\boldsymbol{\mu},\ell_1} = \|\cdot\|_{\ell_1}$. La comparaison est réalisée à partir des temps d'apprentissage moyens calculés sur 10 essais. Pour chaque essai, un nouveau jeu de données est généré et les paramètres C et γ sont déterminés par validation croisée.

Les temps d'apprentissage moyens (ainsi que les erreurs types) de l'approche directe et de l'algorithme par contraintes actives sont donnés sur la figure 4.8, en fonction de la taille de l'espace de similarité, qui correspond au nombre r de prototypes composant $\mathcal{L}$. Pour être cohérent avec l'exemple de la section 4.2.3, qui propose un protocole expérimental dans lequel les prototypes de $\mathcal{L}$ sont les instantanés issus de la discrétisation des séquences d'apprentissage, la taille de l'ensemble d'apprentissage croît aussi proportionnellement à l'espace de similarité. La figure 4.8 illustre clairement le gain de temps de calcul fourni par l'algorithme par contraintes actives. Empiriquement, l'approche directe a une complexité approximativement en $O(r^4)$ (par rapport à la dimension des données) alors que celle de l'algorithme 4 semble être en $O(r^{2,5})$. De manière générale, un problème parcimonieux a tout intérêt

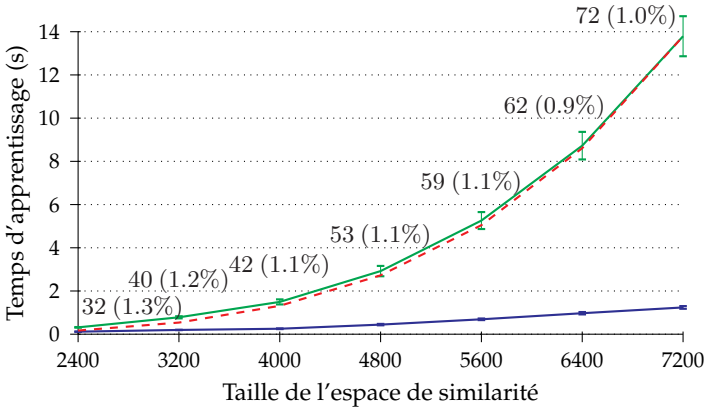


FIGURE 4.8 – Temps d'apprentissage d'un détecteur précoce par une méthode directe (en vert, accompagnée d'une estimation en $O(r^4)$ en rouge) et par contraintes actives (en bleu), en fonction de la taille de l'espace de représentation (*i.e.* de la dimension des données). Sur le graphique sont indiqués les nombres moyens de poids non-nuls de $\mathbf{w}$ (*i.e.* $\|\mathbf{w}\|_{\ell_0}$) ainsi que les pourcentages correspondants par rapport à la dimension des données.

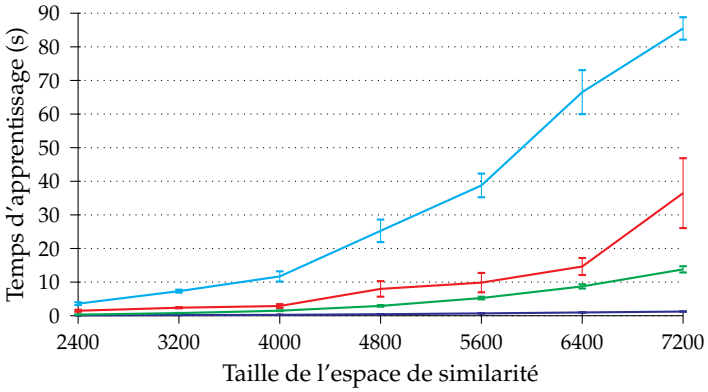


FIGURE 4.9 – Temps d'apprentissage d'un détecteur précoce par une méthode directe (vert), par contraintes actives (bleu), par incréments dimensionnels directs (cyan) et par une approche couplée de contraintes actives et d'incréments dimensionnels (rouge).

à être résolu par une telle méthode. Ceci est d'autant plus avantageux que le nombre de poids non-nuls de $\mathbf{w}$ semble évoluer linéairement par rapport à la taille de l'espace de similarité. En effet, il apparaît sur cette expérience que $\|\mathbf{w}\|_{\ell_0} \approx 10^{-2}\,\mathrm{Card}(\mathbf{w})$, quelque soit la dimension des données.

La figure 4.9 reproduit les courbes de l'illustration 4.8 en y ajoutant les temps d'apprentissage moyens de l'algorithme 5 (en cyan), initialisé avec 500 dimensions (les autres sont ajoutées en une fois). Ceux-ci sont clairement prohibitifs. Plus spécifiquement 40% du temps est passé à vérifier la contrainte d'inégalité sur les dimensions inactives : $\boldsymbol{Q}_{\overline{\mathcal{S}},\mathcal{M}}(\boldsymbol{\alpha}_{\mathcal{M}} + \boldsymbol{\Delta\alpha}_{\mathcal{M}}) < \boldsymbol{\mu}_{\overline{\mathcal{S}}}$, tandis que 15% est dévoué à la résolution du système linéaire. Cette expérience met en évidence l'un des défauts de l'approche incrémentale comparée à l'algorithme par contraintes actives : alors que ce dernier ne tient pas compte des dimensions inactives des données, l'algorithme 5 les identifie mais les utilise pour déterminer le plus grand pas admissible η. Ainsi, l'approche par incréments dimensionnels ne tire pas avantage de la structure parcimonieuse du problème pour entraîner rapidement notre détecteur.

La courbe rouge de la figure 4.9 répond à ce problème en modifiant l'approche incrémentale. Dans cette évolution, les dimensions sont ajoutées par paquets de 40. De plus, à chaque itération, les dimensions ajoutées sont celles qui violent le plus les conditions d'optimalité

et les dimensions inactives de l'itération précédente sont temporairement mises de côté (à l'instar de l'algorithme 4). Avec cette approche, seulement 5% du temps est consacré à la vérification des contraintes d'inégalité tandis que 30% est alloué à la résolution du système linéaire. Conformément aux études précédentes, il est envisageable de réduire le temps de résolution du système par une mise à jour de rang un de la matrice inverse $\boldsymbol{M}^{-1}$ [Cauwenberghs et Poggio, 2001, Laskov *et coll.*, 2006]. Toutefois, le temps nécessaire à l'apprentissage resterait au mieux proche de celui requit par une méthode directe (courbe verte) et par conséquent, largement supérieur à celui de l'algorithme 4 par contraintes actives.

4.5.2 Fiabilité

Le soucis principal de cette section est d'illustrer la fiabilité du détecteur mis en place. Pour ce faire, notre approche de détection précoce est évaluée sur deux problèmes, respectivement liés à la détection de scènes audio et d'actions humaines (vidéo). Le problème audio consiste à déterminer dans quel lieu spécifique des séquences de 30 secondes ont été enregistrées (café, bus, marché, *etc.*) [Rakotomamonjy et Gasso, 2014]. Ces paysages sonores ont été acquis par une unique personne à l'aide d'un téléphone portable (Samsung Galaxy S III), à la fréquence d'échantillonnage de 22050 Hz, donnant ainsi un caractère particulièrement réaliste au protocole expérimental. À terme, il sera d'une grande utilité d'être en mesure de reconnaître des signaux obtenus dans de telles conditions (aussi bien du point de vue de l'environnement que de la qualité du matériel d'acquisition). Les enregistrements se sont déroulés dans la ville de Rouen (et *a minima* dans le métropolitain de Paris) entre décembre 2012 et novembre 2013, sauvegardés au format MP3 avec un taux de 64 kbps. 19 classes ont été construites, regroupant des signaux audio de différents endroits chacune.

Notre détecteur, ainsi que les approches concurrentes, ont été évalués sur plusieurs problèmes binaires de type *un-contre-un*. C'est un cadre relativement simple qui nous permet de mettre en avant les aptitudes de chaque approche comparée. La représentation TC utilisée consiste à calculer les MFCC sur une fenêtre glissante de 370 millisecondes, avec un chevauchement de moité, à l'instar de [Andén et Mallat, 2014]. Pour des raisons calculatoires, seul un instantané sur 10 est considéré, de sorte qu'un paysage sonore a une durée équivalente de 3 secondes. La représentation par similarités et celle présentée dans l'exemple de la section 4.2.3 avec tous les instantanés discrétisés comme prototypes et $p \to +\infty$ (autrement dit en remplaçant la norme ℓ_p par la fonction maximum).

Le problème vidéo quant à lui est celui introduit dans [Gorelick *et coll.*, 2007a]. Il est construit à partir d'un ensemble de séquences représentant chacune une action humaine (courir, sauter, *etc.*). Ces séquences vidéo peuvent être interprétées comme des silhouettes bidimensionnelles évoluant avec le temps et qui génèrent de ce fait différentes formes espace-temps. La représentation TC utilisée pour cette application est celle décrite dans [Gorelick *et coll.*, 2007a] et dont le code est disponible sur internet [Gorelick *et coll.*, 2007b]. Pour l'évaluation des différentes approches, plusieurs problèmes binaires de type *un-contre-tous* ont été étudiés dans un espace de similarité identique à celui utilisé pour le problème audio.

Le protocole expérimental prévoit 80% des séquences pour entraîner le détecteur et les 20% restants pour évaluer son potentiel de généralisation. Les deux jeux de données considérés (audio et vidéo), sont aléatoirement scindés de cette manière à 10 reprises afin d'obtenir une information statistique. De plus, les paramètres inconnus (C et γ) sont choisis par validation croisée. Le nombre de variables composant les données de classification (dans l'espace de représentation par similarités) varie de 1400 à 2500 pour les scènes audio et autour de 700 pour les actions vidéo.

Dans cette section, le modèle de détecteur proposé est comparé à deux autres approches. La première est MILES [Chen *et coll.*, 2006]. Elle correspond à une SVM régularisée par

PROBLÈMES	MILES	MMED	NOTRE MODÈLE
Billard vs restaurant	4.0 ± 0.7	10.7 ± 1.2	$\mathbf{6.4 \pm 0.8}$
Restaurant vs billard	6.4 ± 1.4	27.6 ± 2.0	$\mathbf{18.5 \pm 1.9}$
Jeu d'enfants vs rue agitée	1.9 ± 0.5	11.8 ± 2.1	$\mathbf{2.9 \pm 0.7}$
Billard vs jeu d'enfants	0.4 ± 0.4	6.2 ± 0.8	$\mathbf{5.4 \pm 0.8}$
Marche vs reste	2.8 ± 2.8	7.5 ± 3.8	$\mathbf{0 \pm 0}$
Saut à deux pieds vs reste	17.5 ± 3.8	22.8 ± 4.5	$\mathbf{18.1 \pm 3.6}$
Saut à un pied vs reste	10.3 ± 5.6	25.0 ± 6.5	$\mathbf{20.3 \pm 7.2}$

TABLE 4.1 – Erreurs moyennes (et erreurs types) de détection des différents modèles. Notons que MILES ne réalise pas réellement une détection précoce et que ses résultats sont uniquement fournis comme références.

une norme ℓ_1 (sur le vecteur de poids ainsi que sur le seuil de détection), appliquée dans l'espace de similarité. La différence entre ce type de SVM est notre détecteur réside uniquement dans la contrainte de positivité (alternativement de négativité) des poids, $\mathbf{w} \succcurlyeq 0$ (elle est absente dans MILES et présente dans l'apprentissage de notre détecteur). MILES ne permet pas de réaliser une détection précoce ou fiable, mais donne une indication concernant l'impact de la contrainte de positivité (que nous avons introduite) sur le taux de bonne reconnaissance. De plus, l'approche concurrençant directement nos travaux, à savoir MMED, est aussi évaluée. Comme nous l'avons vu, MMED a été mis en place pour détecter et localiser un événement au sein d'une série temporelle. C'est pourquoi, un jeu d'apprentissage conforme est un ensemble de séquences contenant chacune un événement, dont la position est indiquée par une étiquette-intervalle. Pour l'adapter à notre cadre, nous procédons de la manière suivante : chaque séquence étiquetée $+1$ se voit attribuer l'étiquette-intervalle $[0, T]$, signifiant que la séquence dans la totalité est un événement. Réciproquement, chaque série étiquetée -1 est envoyée à MMED avec l'étiquette-intervalle $[0, 0]$. Cette configuration est cohérente avec le logiciel mis à disposition par les auteurs et avec la théorie présentée dans [Hoai et De la Torre, 2014].

Le tableau 4.1 décrit les capacités de généralisation des trois modèles comparés. Notre modèle a été entraîné avec un vecteur de pondération de la norme fixé à $\boldsymbol{\mu} = \mathbb{1}$. L'impact de la norme pondérée sera évalué dans la section suivante et est volontairement laissé de côté pour le moment. Les performances atteintes par MILES, qui sont meilleures que celles de MMED et de notre détecteur, indiquent le prix que coûte le caractère précoce des deux autres approches. Dans la cas de notre modèle, les pertes de généralisation observées par rapport à MILES sont donc dues à la contrainte de positivité du vecteur de pondération $\mathbf{w}$. Pour diminuer ces pertes, il serait nécessaire que la représentation par similarités utilisée induise une configuration géométrique particulière, à savoir que la classe à détecter soit majoritairement *plus haut et plus à droite* que la classe des non-événements (figure 4.4 page 102), ce qui n'est pas nécessairement assuré bien que cohérent avec l'utilisation d'une fonction de similarité (et non de dissimilarité). Ceci est nettement mis en lumière dans la dissymétrie des résultats des exemples *billard vs restaurant* et *restaurant vs billard*).

Pour ces deux expériences, les résultats de classification de MILES sont aussi différents (de 2.4%). Cette différence n'est due qu'au jeu des statistiques mais ne se justifie pas par la construction de la méthode. De même, les résultats obtenus pour *marche vs reste* sont soumis aux mêmes effets néfastes, expliqués par le nombre d'essais à partir desquels sont calculées les statistiques (seulement 10). En effet, notre modèle atteint une erreur de classification nulle sur tous les ensembles de test, tandis que MILES obtient $2,8\%$ en moyenne. Une analyse approfondie des résultats montre que MILES échoue à classer 28% des exemples de test lors d'un seul essai, mais les classe tous correctement sur les 9 autres. Dans cet essai pathologique, MILES et notre modèle classent correctement toutes les séquences d'appren-

PROBLÈMES	MILES	MMED	NOTRE MODÈLE
Billard vs restaurant	2.0 ± 0.2	364.1 ± 69.2	$\mathbf{0.3 \pm 1{\cdot}10^{-2}}$
Restaurant vs billard	1.8 ± 0.1	23.4 ± 5.2	$\mathbf{0.4 \pm 3{\cdot}10^{-2}}$
Jeu d'enfants vs rue agitée	$0.3 \pm 1{\cdot}10^{-2}$	37.5 ± 14.8	$\mathbf{0.1 \pm 2{\cdot}10^{-3}}$
Billard vs jeu d'enfants	$0.2 \pm 2{\cdot}10^{-2}$	44.8 ± 10.2	$\mathbf{0.1 \pm 2{\cdot}10^{-3}}$
Marche vs reste	$0.1 \pm 7{\cdot}10^{-3}$	1.7 ± 0.3	$\mathbf{2{\cdot}10^{-2} \pm 1{\cdot}10^{-3}}$
Saut à deux pieds vs reste	$0.1 \pm 3{\cdot}10^{-3}$	1.5 ± 0.2	$\mathbf{3{\cdot}10^{-2} \pm 2{\cdot}10^{-3}}$
Saut à un pied vs reste	$0.2 \pm 1{\cdot}10^{-2}$	1.9 ± 0.2	$\mathbf{3{\cdot}10^{-2} \pm 1{\cdot}10^{-3}}$

TABLE 4.2 – Temps d'apprentissage moyens en secondes (et erreurs types). Dans cette expérience, notre modèle a été entraîné par une méthode directe (sans technique de contraintes actives), au même titre que MILES.

tissage. Il semble donc que, dans ce cas particulier, la contrainte de positivité $\mathbf{w} \succcurlyeq 0$ a privilégié des instantanés plus pertinents que ceux sélectionnés par MILES. Cependant, les autres expériences laissent penser qu'en augmentant le nombre d'essais, notre modèle atteindra des taux de classification inférieurs à ceux de MILES. Ces effets mettent en lumière l'intérêt de conduire de plus amples expérimentations. Néanmoins, celles-ci sont, pour le moment, freinées par les ressources et le temps de calcul prohibitifs nécessaires à MMED (voir le paragraphe suivant).

La comparaison avec MMED, exposée dans le tableau 4.1, est clairement en faveur de notre modèle. En effet, celui-ci est plus apte à généraliser que MMED. En outre, le tableau 4.2, qui indique les temps d'apprentissage de chaque modèle par des méthodes directes (*CPLEX* pour MMED et *lpsove* sans technique de contraintes actives pour MILES et notre modèle), montre que notre détecteur précoce est bien plus rapide à entraîner que MMED. Ce phénomène s'explique par le très grand nombre de contraintes du problème d'optimisation introduit par Hoai *et coll.* L'apprentissage de notre modèle est aussi plus rapide que celui de MILES. Pour expliquer cet état de fait, rappelons que dans cette expérience, l'unique différence entre MILES et notre approche réside dans l'ajout de la contrainte $\mathbf{w} \succcurlyeq 0$. Ainsi, pour MILES, qui ne contraint pas $\mathbf{w}$, il est nécessaire d'appliquer l'astuce usuelle de dédoublement des variables ($\mathbf{w} = \boldsymbol{a} - \boldsymbol{b}$, $\boldsymbol{a}, \boldsymbol{b} \succcurlyeq 0$) pour rendre le problème d'optimisation linéaire [Chen *et coll.*, 2006]. La différence de temps d'apprentissage s'explique donc par le nombre de variables d'optimisation, plus important pour le LP de MILES que celui de notre détecteur.

La figure 4.10 illustre l'évolution des décisions prises pas notre détecteur au cours du temps pour quatre séquences différentes. Les sauts que l'on voit apparaître correspondent à l'apparition d'instantanés discriminants. Remarquons que le franchissement du zéro par une courbe de décision indique le moment (plus précisément le temps normalisé par la durée totale des séquences) auquel le détecteur affirme que la séquence analysée est un évènement détecté.

Enfin, le tableau 4.3 décrit les temps moyens de détection pour les différents problèmes étudiés. Il apparaît que MMED [Hoai et De la Torre, 2014] est clairement plus rapide pour prendre une décision que notre détecteur. Ceci est cohérent puisque nous n'avons pas pénalisé une prise de décision tardive ($\boldsymbol{\mu} = \mathbb{1}$). De plus, nous avons pu voir que notre modèle est plus apte à la détection que MMED. Suivant le compromis discrimination vs précocité, cela suggère que notre approche est plus longue à prendre une décision, ce qui est vérifié dans le tableau 4.3. En revanche, il est important de noter que l'approche dirigée par les données intrinsèque à MMED ne fournit aucune garantie quant à la fiabilité du détecteur. En pratique, les auteurs de [Hoai et De la Torre, 2014] ne considèrent pas comme décision finale (en phase de test) le signe du détecteur linéaire mais la plus grande décision de toutes celles prises lors du passé. Cet artifice rend la décision finale croissante en fonction du temps bien

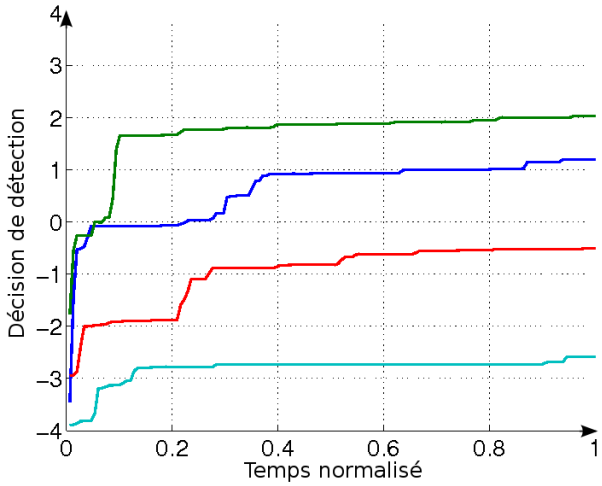


FIGURE 4.10 – Exemples d'évolution au cours du temps de la décision de détection de notre modèle pour quatre séquences. La notification de détection est émise lorsque la courbe franchit zéro.

PROBLÈMES	MMED	NOTRE MODÈLE
Billard vs restaurant	$\mathbf{0.23 \pm 0.02}$	0.40 ± 0.03
Restaurant vs billard	$\mathbf{0.24 \pm 0.02}$	0.54 ± 0.03
Jeu d'enfants vs rue agitée	$\mathbf{0.25 \pm 0.03}$	0.41 ± 0.03
Billard vs jeu d'enfants	$\mathbf{0.18 \pm 0.01}$	0.35 ± 0.02
Marche vs reste	0.11 ± 0.02	$\mathbf{0.09 \pm 0.01}$
Sauts à deux pieds vs reste	$\mathbf{0.08 \pm 0.01}$	0.10 ± 0.01
Saut à un pied vs reste	$\mathbf{0.07 \pm 0.01}$	0.15 ± 0.02

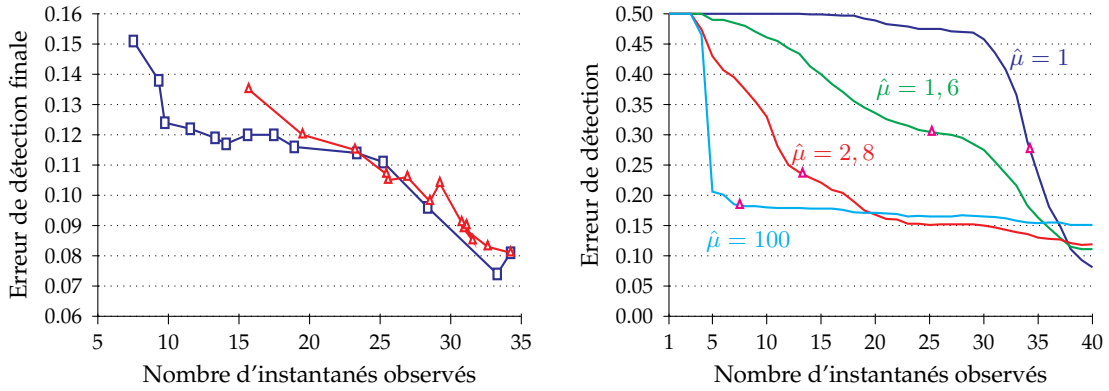
TABLE 4.3 – Temps normalisés moyens (et erreurs types) de prise de décision concernant l'émission d'un avis de détection. Une valeur de 1 indiquerait que la décision a été prise en analysant la séquence dans sa totalité.

qu'elle ne le soit pas nécessairement réellement.

Dans la section qui vient, nous revenons sur la pénalisation d'une prise de décision tardive et nous montrons que $\boldsymbol{\mu}$ est un paramètre réglant le compromis discrimination vs précocité.

4.5.3 Précocité

Dans les expériences précédentes, nous avons fixé $\boldsymbol{\mu} = \mathbb{1}$, de sorte qu'une prise de décision tardive n'était pas pénalisée. Ce choix est justifié par deux raisons. Premièrement, par ce moyen, nous avons principalement évalué la capacité de détection et vérifié la fiabilité de notre détecteur. Deuxièmement, la pénalisation d'une prise de décision tardive n'est effective que sur des séquences structurées, *i.e.* des signaux pour lesquels plus l'observation est complète, plus la puissance de discrimination est importante. Ceci n'est pas le cas pour l'expérience de reconnaissance de paysages sonores. Pour illustrer cette assertion, prenons l'exemple de la salle de jeux et supposons qu'un cri d'enfant est un instantané discriminant. Cet instantané peut apparaître à n'importe quel moment dans la séquence, de sorte qu'analyser un enregistrement dans sa totalité n'apporte pas de nouvelle information une fois qu'un cri a été perçu (excepté si plusieurs cris apparaissent sur la même séquence). Au contraire, sur une action vidéo telle que *se baisser* (qui est structurée), il est évident que les instantanés les plus discriminants apparaissent à la fin des séquences et qu'une observation



(a) Évolution de l'erreur de détection (observation complète) en fonction du nombre moyen d'instantanés nécessaires pour émettre un avis de détection. Lorsque l'on parcourt les courbes (bleue pour le profil linéaire et rouge pour celui logarithmique) de gauche à droite, $\hat{\mu}$ décroît de 100 à 1.

(b) Évolution de l'erreur de détection moyenne au cours de l'analyse des séquences de test. Les triangles magenta indiquent le nombre moyen d'instantanés nécessaires pour émettre un avis de détection.

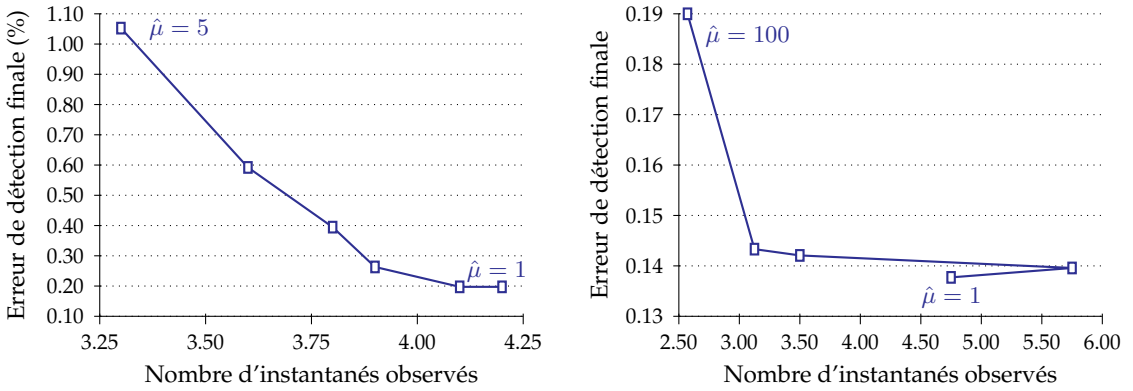
FIGURE 4.11 – Évaluation de la précocité du détecteur sur le jeu de données synthétique.

complète profite au pouvoir de discrimination.

Ainsi, sur des séries temporelles structurées, une pondération adéquate de la norme, comme celles présentées sur l'illustration 4.5 page 104 (linéaire et logarithmique), permet de pénaliser la sélection d'instantanés survenant à la fin des séquences, et par conséquent de favoriser la précocité de la détection. La première expérience que nous conduisons compare donc les deux profils de pondération et valide leur utilité. Nous utilisons pour cela le jeu de données synthétisé à partir de deux *chirps* linéaires, introduit plus tôt. Il est évident que les séquences ainsi générées sont structurées puisque plus on avance dans le temps, plus les fréquences des deux formes de base diffèrent. Conformément à l'hypothèse faite dans le corollaire 4.3.3, la plus petite valeur de $\boldsymbol{\mu}$ est fixée à 1 et seul le maximum $\hat{\mu}$ est variable.

L'illustration (a) de la figure 4.11 trace l'évolution de l'erreur obtenue en observant complètement les séquences de test, en fonction du nombre moyen d'instantanés à observer pour émettre un avis de détection. Chaque point du graphique correspond à une moyenne calculée sur 10 essais et pour lesquels les paramètres C et γ du modèle ont été déterminés par validation croisée. Dans cette expérience, $\hat{\mu}$ décroît de 100 à 1 lorsque l'on parcourt les courbes de gauche à droite. Il est possible d'observer que les deux profils conduisent à des courbes globalement proches pour les petites valeurs de $\hat{\mu}$ mais semblent diverger pour les grandes valeurs. En outre, le profil linéaire (bleu) donne plus d'*amplitude* pour régler le compromis discrimination vs précocité que le profil logarithmique, à intervalle de $\hat{\mu}$ fixé (ici $[1, 100]$). Dans la suite, nous choisirons donc un profil linéaire. Par exemple, l'illustration (b) de la figure 4.11 indique l'évolution de l'erreur de détection moyenne au cours de l'analyse des séquences pour $\hat{\mu}$ valant (de gauche à droite) 100, $2,8$, $1,6$ et 1. Sur chaque courbe, le triangle magenta indique le nombre moyen d'instantanés qu'il est nécessaire d'observer pour émettre un avis de détection. Conformément à l'illustration (a), celui-ci décroît lorsque $\hat{\mu}$ augmente. De plus, l'erreur d'observation totale (que l'on lit à l'abscisse 40) augmente bien avec $\hat{\mu}$.

La figure 4.12 page suivante indique l'évolution de l'erreur de détection finale en fonction du nombre moyen d'instantanés observés pour deux événements vidéo (*se baisser* et *saluer de grands gestes de la main*), au même titre que l'illustration (a) de la figure 4.11 pour le jeu de données synthétique. L'illustration (a) de la figure 4.12 page suivante affiche une tendance décroissante attendue, bien que les intervalles de valeurs mis en jeu (sur le nombre d'ins-



(a) Se baisser ($\hat{\mu} \in [1, 5]$). (b) Saluer de grands gestes de la main ($\hat{\mu} \in [1, 100]$).

FIGURE 4.12 – Évaluation de la précocité du détecteur sur les actions video : évolution de l'erreur de détection (observation complète) en fonction du nombre moyen d'instantanés nécessaires pour émettre un avis de détection.

tances à observer comme sur l'erreur de détection) soient relativement faibles. De fait, cette classe est particulièrement simple à détecter, y compris en observant une très faible portion des séquences. Au contraire, l'illustration (b) de la figure 4.12 affiche des intervalles (particulièrement d'erreurs) plus importants mais montre que le nombre moyen d'instantanés à observer n'évolue pas de manière monotone avec $\hat{\mu}$. Il est raisonnable de supposer que ce phénomène est dû soit au jeu des statistiques, soit à la présence d'une structure périodique des instantanés discriminants. En conséquence, la pondération linéaire présentée dans la figure 4.5 page 104 n'est peut-être pas adéquate pour ce problème de détection. Un pondération efficace supposerait la connaissance de la périodicité des événements discriminants, *i.e.* de la notion de redondance d'information présente dans la séquence, ce qui dépasse le cadre de notre étude.

Jusqu'ici, nous avons évalué la précocité de notre détecteur sur différents jeux de données, notamment réels. Nous terminons à présent cette évaluation en le comparant à MMED [Hoai et De la Torre, 2014]. Nous choisissons pour cela le problème synthétique introduit au début de cette section car il a, pour le moment, fournit des résultats plus probants que les jeux de données réelles. Afin de faire varier le compromis précocité `vs` discrimination réalisé par MMED, nous avons modifié l'application g proposée par les auteurs et considéré une fonction $g : \tau \in \mathbb{N}_{\lfloor \frac{T}{\Delta t} \rfloor} \mapsto \hat{\tau} + \frac{\tau \Delta t}{T}(1 - \hat{\tau}) \in [\hat{\tau}, 1]$, où $\hat{\tau}$ est un réel de $]0, 1]$ (*cf.* problème d'optimisation (4.5)). Cette application est linéaire et croissante, suivant les recommandations des auteurs. Elle atteint son minimum en 0 (*i.e.* en début de séquence), de valeur $\hat{\tau}$, et son maximum en fin de séquence. Ainsi, la fonction g impacte la précocité de la décision de la manière suivante : plus $\hat{\tau}$ augmente, moins les séquences partiellement observées sont autorisées à être mal classées. Par conséquent, en faisant croître $\hat{\tau}$, on force la précocité de la prise de décision.

Les résultats de la comparaison de notre détecteur à MMED sont présentés sur la figure 4.13 page suivante [3]. Dans l'ensemble, MMED est plus apte que notre détecteur à prendre une décision précocement. Ceci peut en partie s'expliquer par la manière qu'a chaque méthode d'aborder la fiabilité de la décision. Notre détecteur est totalement fiable, ce qui conduit à imposer des contraintes géométriques tendant à diminuer son pouvoir de discrimina-

3. Les différences de résultats de notre détecteur entre les figures 4.11 page précédente et 4.13 page suivante s'expliquent par la taille de l'ensemble d'apprentissage : celui considéré dans la comparaison de MMED à notre approche est plus restreint que celui utilisé pour la figure 4.11 page précédente, car MMED est particulièrement gourmand en temps de calcul et en ressources calculatoires.

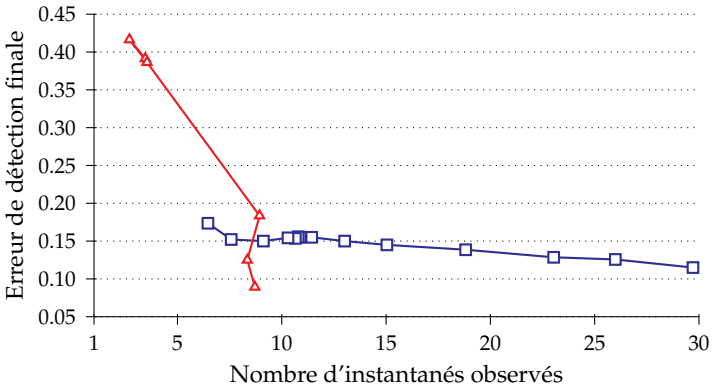


FIGURE 4.13 – Comparaison de la précocité de notre détecteur (bleu) à celle de MMED (rouge) sur le jeu de données synthétique. Les courbes représentent l'évolution de l'erreur de détection (observation complète) en fonction du nombre moyen d'instantanés nécessaires pour émettre un avis de détection. Lorsque l'on parcourt les courbes de gauche à droite, $\hat{\mu}$ décroît de 100 à 1 pour notre détecteur (courbe bleue) et $\hat{\tau}$ de 1 à 0 pour MMED (courbe rouge).

tion. En revanche, MMED n'assure la fiabilité de la décision que de manière dirigée par les données et en autorisant des écarts. Par conséquent, il est plus libre d'atteindre de faibles erreurs de reconnaissance que notre détecteur.

4.5.4 Fonctionnement en temps réel

Jusqu'à présent, nous avons validé et évalué les propriétés de notre détecteur précoce dans un cadre conforme à celui de son apprentissage, *i.e.* en lui présentant des séquences qui sont soit des événements, soit des non-événements. Nous désirons ici mettre en avant son fonctionnement en *temps réel*, sur une série temporelle comportant des événements localisés à certains instants, semblable à celle de la figure 4.1 page 96. Pour ce faire, nous reprenons le problème de détection audio *jeu d'enfants* `vs` *rue agitée* et nous construisons une unique séquence de test par concaténation (dans un ordre aléatoire) d'événements (enregistrements de jeux d'enfants) et de non-événements (enregistrements de rues agitées).

Notre détecteur précoce est ensuite appliqué à chaque instant de cette série temporelle. Deux points clefs sont à relever :

- ◇ d'un instant à l'autre, le calcul de la représentation par similarité est très peu coûteuse. En effet, si l'on considère une fonction de similarité $k_\sim$ (conforme à l'exemple de la section 4.2.3) construite comme l'agrégation en norme ℓ_p d'une mesure de similarité quelconque $q\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$:

$$k_{\tau\Delta t}\colon (x_\sim, \boldsymbol{p}) \in \mathcal{X}^{[0,T]} \times \mathcal{X} \mapsto \left(\sum_{\tau'=0}^{\tau} q(x_{\tau'\Delta t}, \boldsymbol{p})^p \right)^{\frac{1}{p}},$$

  où $\tau\Delta t$ est l'instant (discret) considéré, alors à l'instant suivant$(\tau+1)\Delta t$, on a simplement :

$$k_{(\tau+1)\Delta t}(x_\sim, \boldsymbol{p}) = \left(k_{\tau\Delta t}(x_\sim, \boldsymbol{p})^p + q(x_{(\tau+1)\Delta t}, \boldsymbol{p})^p \right)^{\frac{1}{p}};$$

- ◇ lorsqu'un événement a été détecté au cours de l'analyse d'une série temporelle, il n'est pas nécessaire de continuer à calculer la fonction de décision puisque l'on est assurés qu'elle restera supérieure à 0 (c'est la propriété de fiabilité de notre détecteur précoce). En conséquence, il est possible de réinitialiser le système de reconnaissance

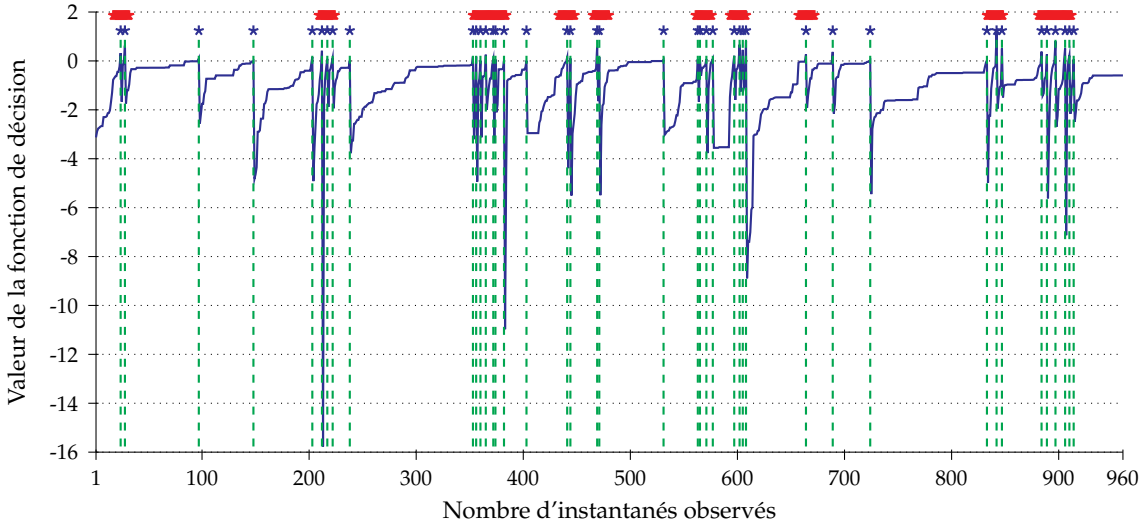


FIGURE 4.14 – Évolution de la fonction de décision au cours de l'analyse d'une série temporelle. Les événements sonores à détecter (jeux d'enfants) sont indiqués par les bandes rouges. Le reste de la séquence est constitué d'enregistrements de rues agitées. Les étoiles bleues mentionnent les notifications de détection (*i.e.* lorsque la valeur de la fonction de décision est supérieure à 0). Elles correspondent ici aux ré-initialisations du détecteur (marqueurs verts).

en *effaçant le passé de la séquence* : supposons qu'une détection a eu lieu à l'instant $\bar{\tau}\Delta t$. Alors la nouvelle mesure de similarité à considérer agrège l'information depuis l'instant $(\bar{\tau}+1)\Delta t$ uniquement :

$$k_{\tau\Delta t}\colon (x_\sim, \boldsymbol{p}) \in \mathcal{X}^{[0,T]} \times \mathcal{X} \mapsto \left(\sum_{\tau'=\bar{\tau}+1}^{\tau} q(x_{\tau'\Delta t}, \boldsymbol{p})^p \right)^{\frac{1}{p}}.$$

La figure 4.14 décrit l'évolution de la fonction de décision $t \mapsto \langle \mathbf{w} \mid \psi_t^{\mathcal{L}}(x_\sim) \rangle_{\ell_2} - b$ au cours du temps. Les marqueurs verts indiquent les ré-initialisations du système, correspondant aux instants où une détection est émise (marqués par une étoile bleue). On constate que la fonction de décision croît systématiquement entre deux ré-initialisations. Bien que ceci n'est en principe requis que lors des événements, la propriété 4.2.1 assure que c'est le cas quelque soit le type de séquences. Les paliers dans la fonction de décision indiquent l'absence d'instantanés discriminants.

Sur l'illustration 4.14, les segments rouges (étoiles concomitantes) indiquent les emplacements des événements. On s'attend donc à trouver au moins une étoile bleue par bande rouge et aucune en dehors. Ceci est majoritairement le cas. En revanche, certaines détections apparaissent uniquement par agrégation de petites valeurs de similarités sur une longue période, sans qu'aucun événement ne soit présent (par exemple les troisième et quatrième étoiles bleues de la figure 4.14). Pour éviter cet effet de bord, il est envisageable de réinitialiser le système de reconnaissance lorsqu'aucun événement n'a été détecté pendant une certaine période (par exemple la durée moyenne des événements). La figure 4.15 page suivante illustre ce mécanisme et la diminution des faux-positifs auquel il conduit.

4.6 SYNTHÈSE

La contribution présentée dans ce chapitre réside dans la proposition d'un nouveau cadre visant à détecter au plus tôt des évènements temporels. En étendant le principe de repré-

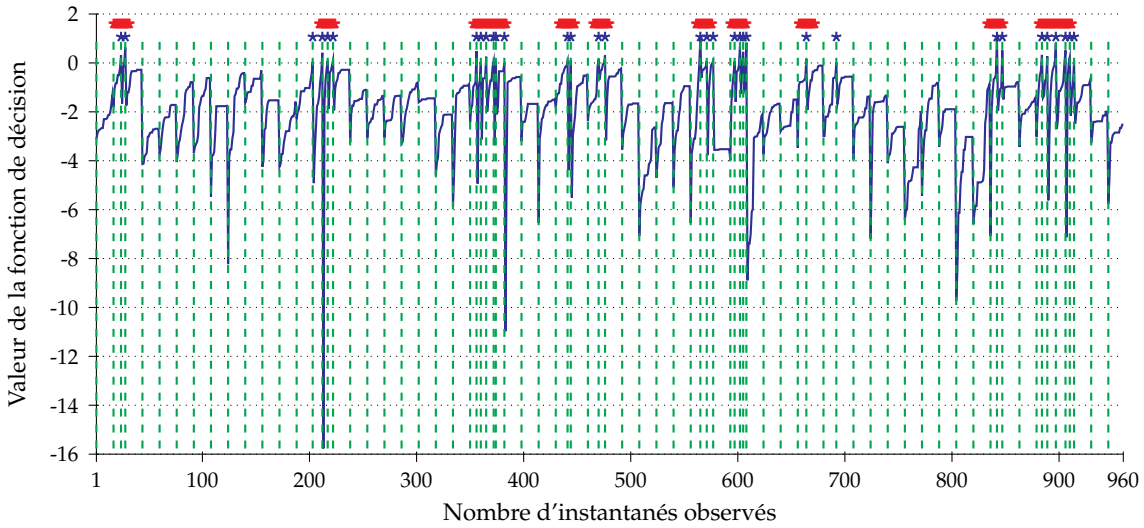


FIGURE 4.15 – Évolution de la fonction de décision au cours de l'analyse d'une série temporelle (voir la légende de la figure 4.14 page précédente). Les marqueurs verts indiquent les ré-initialisations, apparaissant lorsqu'aucun événement n'a été détecté pendant la durée moyenne desdits événements ou qu'une détection est émise.

sentation par similarités aux séries temporelles et en y ajoutant quelques conditions, il nous a été possible de mettre en place un problème d'apprentissage simple, résolu efficacement par un algorithme de contraintes actives. Les détecteurs ainsi entraînés présentent la garantie d'une fiabilité totale : lorsqu'une notification de détection est émise au cours de l'analyse d'une séquence, il est certain qu'en continuant à analyser ladite séquence, cette notification restera active. En outre, le cadre que nous proposons est accompagné d'une borne de généralisation directement issue de la théorie SVM.

Grâce à diverses expériences numériques, nous avons montré que cette approche fournit un réel gain de temps d'apprentissage et de taux de reconnaissance par rapport à son plus proche concurrent, MMED, introduit dans [Hoai et De la Torre, 2014]. Dans ces expériences, il apparaît que MMED est plus rapide que notre approche pour prendre une décision. Ceci vient en contrepartie des meilleurs taux de reconnaissance que nous obtenons et de la propriété de fiabilité totale de notre détecteur. Cependant, l'aptitude de ce dernier à régler le compromis entre discrimination et précocité a été clairement mis en lumière sur un jeu de données structurées synthétique et sur la reconnaissance d'actions humaines.

Comme il l'a été souligné, le cadre de reconnaissance précoce que nous proposons impose une certaine contrainte sur la représentation par similarités utilisée : la classe des séquences à détecter (les événements) doit être majoritairement située *en haut à droite* par rapport aux non-événements. Les expériences numériques révèlent qu'il est difficile de prévoir si une représentation par similarités satisfait cette condition (et donnera ainsi des taux de reconnaissance proches d'un système non-contraint tel que MILES), notamment car les prototypes discriminants ne sont pas connus par avance. C'est pourquoi, nos recherches actuelles se concentrent sur la généralisation de ces travaux à un cadre rendant possible l'apprentissage d'une représentation par similarités la plus adéquate possible (*i.e.* d'une représentation TC ou de manière équivalente d'une mesure de similarité). Concernant le pendant théorique de cette remarque, les garanties de généralisation que nous avons données sont directement issues de la théorie SVM, et donc relativement générales. Cependant, les travaux initiés dans [Balcan *et coll.*, 2008,Kar et Jain, 2012] permettent de quantifier l'aptitude à généraliser d'une approche par similarités en fonction d'une mesure d'efficacité de la fonction de proximité utilisée. Il paraît essentiel d'étendre ces travaux à notre cadre et de les lier à la généralisation proposée ci-avant.

Conclusion et perspectives

La mise en place d'outils d'apprentissage visant à reconnaître des signaux conduit naturellement à deux principales difficultés. La première, et la plus apparente, est le choix de descripteurs pour caractériser des groupes de signaux de sorte à obtenir une faible variabilité intra-classe et une forte différence inter-classe. La seconde est la manière d'appréhender la dimension temps. Suivant la tâche et la nature des signaux, il est soit possible de prendre des décisions sur des sous-parties des séries temporelles, puis de les combiner (par exemple par vote majoritaire) ; soit d'agréger l'information contenue dans les vecteurs caractéristiques, puis de prendre une décision globale ensuite. Ces deux façons de prendre le temps en considération se différencient par la permutation de deux représentations : celle de décision (à valeur dans $\{-1, 1\}$) et celle de combinaison des descripteurs.

Les travaux présentés dans ce manuscrit proposent une réponse à ces deux difficultés par des voies originales.

APPRENTISSAGE DE DESCRIPTEURS

Selon une chaîne de traitement constituée de trois étapes : transformation par un Banc de Filtres (BdF), agrégation des représentations Temps-Fréquence (TF) et application d'une machine à vecteurs supports (*Support Vector Machine*, SVM), nous avons proposé une méthode d'apprentissage de BdF discriminant, favorisant la classification de signaux. Celle-ci est fondée sur une hypothèse majeure : les filtres sont contrôlés par peu de paramètres. Cette hypothèse est certes restrictive mais bénéfique pour la reconnaissance de signaux, puisqu'elle joue le rôle d'une régularisation permettant d'améliorer la capacité de généralisation de la règle de décision ainsi apprise. Il a été possible de montrer que dans ce cadre (et pour des noyaux SVM linéaire ou gaussien), le problème d'apprentissage de BdF discriminant se réduit à celui de la détermination d'une combinaison (non-nécessairement convexe) de noyaux, choisis parmi une infinité. La méthode que nous proposons consiste donc à alterner la résolution d'une SVM à noyau fixé, et la mise à jour de la combinaison de noyaux par un algorithme d'ensemble actif. De la sorte, notre algorithme est apte à déterminer automatiquement le nombre de filtres et leurs paramètres adéquats. Ce cadre permet aussi de construire une fonction d'agrégation, réglant le compromis entre la discrimination et l'invariance des descripteurs TF issus du BdF, comme une combinaison d'autres fonctions d'agrégation.

Le BdF que nous considérons dans ces travaux ne possède qu'un seul étage, là où un réseau de neurones convolutifs (*Convolutional Neural Network*, CNN) est constitué d'une cascade de

BdF et d'opérateurs d'agrégation non-linéaires. À mi-chemin entre un CNN et une transformée par diffusion d'ondelettes, il nous semble intéressant de poursuivre nos recherches vers l'apprentissage de BdF imbriqués par des méthodes à noyaux. Ceci rend le système moins interprétable mais à l'instar de la transformée par diffusion d'ondelettes et des réseaux profonds, conférerait très certainement de meilleures capacités de reconnaissance. Toutefois, en tant que méthode d'apprentissage de noyau, notre approche passe difficilement à l'échelle. L'extension à l'imbrication de BdF doit donc se concevoir conjointement à ce problème, par exemple sous la forme d'une optimisation distribuée, afin d'être en mesure de voir le jour.

Mis à part l'apprentissage du BdF en lui même, l'un des freins au passage à l'échelle est la sélection de paramètres C et γ pertinents pour la classification. Dans les travaux que nous avons présentés ici, ceux-ci sont déterminés par validation croisée, augmentant de manière conséquente le besoin en ressources de calculs. Dans une étude récente, nous avons exploré une méthode alternative de sélection de modèle, construite sur la minimisation de l'erreur empirique calculée sur un unique ensemble de validation [Sangnier *et coll.*, 2014]. Cette approche s'écrit sous la forme d'un problème d'optimisation bi-niveau (*i.e.* contenant une contrainte sous la forme d'un problème d'optimisation). Une simple heuristique, de concert à une technique d'apprentissage incrémental d'une SVM, a permis une mise en œuvre efficace de la détermination conjointe du coefficient C et des paramètres du noyau (γ ainsi que la Réponse Impulsionnelle (RI) de chaque filtre du banc).

DÉTECTION PRÉCOCE

Pour notre deuxième contribution, nous avons proposé un cadre permettant de mettre en place un détecteur précoce d'événements au sein de séries temporelles. Celui-ci est construit sur une hypothèse simple, similaire au concept d'apprentissage d'instances multiples (*Multiple Instance Learning*, MIL) : une séquence temporelle est déclarée comme étant un événement (que l'on recherche) si elle contient un ou plusieurs instantanés spécifiques. Cette condition suppose, de manière sous-jacente, que la discrimination des séquences est possible non pas grâce à leur structure temporelle (comme c'était le cas dans la contribution précédente, motivant l'utilisation d'une représentation TF) mais grâce à des instantanés discriminants, susceptibles d'apparaître à n'importe quels moments dans les séquences. Dans le cadre que nous proposons, le détecteur précoce est une fonction linéaire dans un espace de similarité entre les séquences et des instantanés (appelés prototypes). Dans cet espace, il est possible d'énoncer des conditions suffisantes pour que le détecteur soit fiable, *i.e.* ne change pas d'avis une fois qu'il a déclaré qu'une séquence était un événement, au cours de l'analyse de celle-ci. Dans un contexte proche du paradigme SVM, et bénéficiant des mêmes garanties théoriques de généralisation, nous proposons alors un algorithme par contraintes actives permettant simultanément d'apprendre les paramètres du détecteur linéaire précoce et de sélectionner les prototypes pertinents. En outre, le problème d'apprentissage du détecteur bénéficie d'un paramètre contrôlant le compromis entre la discrimination et la précocité de la prise de décision. Celui-ci a été validé expérimentalement sur des données synthétiques et vidéos. De plus, l'intérêt général de notre modèle a été démontré, par rapport à son concurrent direct, sur les mêmes données vidéo ainsi que sur la détection de paysages sonores.

La contrainte de positivité $\mathbf{w} \succcurlyeq 0$ introduite afin d'obtenir un détecteur fiable impose une condition géométrique dans l'espace des prototypes sélectionnés : la classe des événements à détecter doit se trouver *en haut à droite* par rapport à la classe des non-événements. Cette condition assure une faible perte de reconnaissance entre l'apprentissage d'instances multiples avec sélection intégrée des instances (*Multiple-Instance Learning via Embedded instance Selection*, MILES) et notre détecteur. En pratique, il est difficile de vérifier cette condition.

Cependant, il est envisageable d'introduire des techniques d'apprentissage de descripteurs (pour que les prototypes discriminants définissent un *bon* espace de similarité) ou de mesures de similarité (qui définit elle aussi l'espace de représentation des séquences). Certains travaux ont mis en place des outils théoriques pour vérifier la pertinence d'une mesure de similarité [Balcan *et coll.*, 2008, Kar et Jain, 2012]. Une extension de ceux-ci à notre cadre offrirait un moyen alternatif de s'affranchir de notre contrainte géométrique.

ANNEXE A

Critères discriminants

Il existe des critères antérieurs et plus simples que ceux mentionnés dans la section 1.4, fondés sur l'hypothèse d'une distribution unimodale de chaque classe (hypothèse utilisée dans l'analyse discriminante de Fisher). Nous en rappelons deux ici, notamment utilisés dans [Neumann *et coll.*, 2005] : la distance entre les centres des classes (ou inter-classe) J_{C} et le critère de Fisher (généralisé) J_{F}. Pour ce faire, on suppose que l'on dispose d'un ensemble d'apprentissage $\{(\boldsymbol{x_i}, y_i)\}_{1\leq i\leq n}$ et d'un noyau $k\colon \mathcal{X}\times\mathcal{X}\to\mathbb{R}$ dont une fonction de redescription est $\phi\colon \mathcal{X}\to\mathcal{G}$.

Définissons d'abord les cardinaux de chaque classe ainsi que les centres de celles-ci :

$$n_+ = \mathrm{Card}\left(\{i\in\mathbb{N}_n, y_i = 1\}\right), \quad n_- = \mathrm{Card}\left(\{i\in\mathbb{N}_n, y_i = -1\}\right),$$

$$\boldsymbol{\mu_+} = \frac{1}{n_+}\sum_{\substack{1\leq i\leq n\\ y_i=1}} \phi(\boldsymbol{x_i}), \quad \boldsymbol{\mu_-} = \frac{1}{n_-}\sum_{\substack{1\leq i\leq n\\ y_i=-1}} \phi(\boldsymbol{x_i}).$$

La distance inter-classe est alors :

$$J_{\mathrm{C}}(k) = \|\boldsymbol{\mu_+} - \boldsymbol{\mu_-}\|^2_{\mathcal{G}}.$$

Elle est calculée en fonction du noyau k par :

$$J_{\mathrm{C}}(k) = \frac{1}{n_+^2}\sum_{\substack{1\leq i,j\leq n\\ y_i=1,\, y_j=1}} k(\boldsymbol{x_i},\boldsymbol{x_j}) + \frac{1}{n_-^2}\sum_{\substack{1\leq i,j\leq n\\ y_i=-1,\, y_j=-1}} k(\boldsymbol{x_i},\boldsymbol{x_j}) - \frac{2}{n_+n_-}\sum_{\substack{1\leq i,j\leq n\\ y_i=1,\, y_j=-1}} k(\boldsymbol{x_i},\boldsymbol{x_j}).$$

Pour définir le critère de discrimination de Fisher, nous avons besoin du centre de l'ensemble d'apprentissage $\boldsymbol{\mu}$:

$$\boldsymbol{\mu} = \frac{1}{n}\sum_{1\leq i\leq n} \phi(\boldsymbol{x_i}),$$

ainsi que des matrices de dispersion intra (*Within*) et inter-classe (*Between*) $\boldsymbol{S_{W+}}$ et $\boldsymbol{S_{B+}}$:

$$\boldsymbol{S_{W+}} = \sum_{\substack{1\leq i\leq n\\ y_i=1}} (\phi(\boldsymbol{x_i}) - \boldsymbol{\mu_+})(\phi(\boldsymbol{x_i}) - \boldsymbol{\mu_+})^T + \sum_{\substack{1\leq i\leq n\\ y_i=-1}} (\phi(\boldsymbol{x_i}) - \boldsymbol{\mu_-})(\phi(\boldsymbol{x_i}) - \boldsymbol{\mu_-})^T.$$

$$\boldsymbol{S_{B+}} = n_+(\boldsymbol{\mu_+} - \boldsymbol{\mu})(\boldsymbol{\mu_+} - \boldsymbol{\mu})^T + n_-(\boldsymbol{\mu_-} - \boldsymbol{\mu})(\boldsymbol{\mu_-} - \boldsymbol{\mu})^T,$$

Le critère de Fisher correspond alors au rapport des dispersions intra et inter-classe :

$$J_{\mathrm{F}}(k) = \frac{\mathrm{Tr}(\boldsymbol{S_{B}}_{+})}{\mathrm{Tr}(\boldsymbol{S_{W}}_{+})}.$$

En terme d'espérances et de variances empiriques, le critère de Fisher s'écrit aussi :

$$J_{\mathrm{F}}(k) = \frac{n_+ \left\|\boldsymbol{\mu}_+ - \boldsymbol{\mu}\right\|_{\mathcal{G}}^2 + n_- \left\|\boldsymbol{\mu}_- - \boldsymbol{\mu}\right\|_{\mathcal{G}}^2}{\sum_{\substack{1\leq i\leq n \\ y_i=1}} \left\|\phi(\boldsymbol{x_i}) - \boldsymbol{\mu}_+\right\|_{\mathcal{G}}^2 + \sum_{\substack{1\leq i\leq n \\ y_i=-1}} \left\|\phi(\boldsymbol{x_i}) - \boldsymbol{\mu}_-\right\|_{\mathcal{G}}^2}.$$

ANNEXE B

Compléments sur l'apprentissage de noyau multiple

Soient d un entier, $\mathcal{A}$ un sous-ensemble dénombrable et fini de $\mathbb{K}^d$ (ensemble des paramètres) et $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ un d-uplet de noyaux. La formulation d'apprentissage de noyaux multiples (*Multiple Kernel Learning*, MKL) introduite dans [Bach *et coll.*, 2004]) est :

$$\begin{array}{ll} \underset{\{f_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{A}},\, b,\, \boldsymbol{\xi}}{\text{minimiser}} & \dfrac{1}{2}\left(\displaystyle\sum_{\boldsymbol{\theta}\in\mathcal{A}} \|f_{\boldsymbol{\theta}}\|_{\mathcal{H}_{\boldsymbol{\theta}}}\right)^2 + C\sum_{i=1}^{n}\xi_i \\ \text{tel que} & \left\{\begin{array}{l} y_i\left(\displaystyle\sum_{\boldsymbol{\theta}\in\mathcal{A}} f_{\boldsymbol{\theta}}(\boldsymbol{x_i}) + b\right) \geq 1-\xi_i,\ \forall i\in\mathbb{N}_n \\ \boldsymbol{\xi}\succcurlyeq 0, \end{array}\right. \end{array} \tag{B.1}$$

où $\mathcal{H}_{\boldsymbol{\theta}}$ est le espace de Hilbert à noyau reproduisant (*Reproducing Kernel Hilbert Space*, RKHS) engendré par le noyau $k_{\boldsymbol{\theta}}$. Les conditions d'optimalité de Karush-Kuhn-Tucker (KKT) sur le cône du second ordre font alors apparaître un vecteur d'« anti-proportionnalité » $\boldsymbol{\mu}$ vérifiant $\boldsymbol{\mu}\succcurlyeq 0$, $\mathbb{1}^T\boldsymbol{\mu}=1$ et pour lequel toute solution f de (B.1) appartient au RKHS $\mathcal{H}_{[\boldsymbol{\mu}]}$ de noyau $k_{[\boldsymbol{\mu}]}=\sum_{\boldsymbol{\theta}\in\mathcal{A}}\mu_{\boldsymbol{\theta}}k_{\boldsymbol{\theta}}$. Résoudre (B.1) permet donc d'apprendre un noyau multiple convexe et parcimonieux $k_{[\boldsymbol{\mu}]}$. La fonction objectif de (B.1) n'étant pas différentiable pour $f_{\boldsymbol{\theta}}=0$, les auteurs de [Bach *et coll.*, 2004] y ajoute une régularisation de Moreau-Yosida [Lemaréchal et Sagastizábal, 1997]. La différentiabilité de la fonction objectif étant assurée, il est alors possible d'obtenir une solution approchée de (B.1) par un algorithme efficace d'optimisation minimale séquentielle (*Sequential Minimal Optimization*, SMO).

Afin d'accélérer la résolution du problème d'apprentissage MKL introduit dans [Lanckriet *et coll.*, 2004], les auteurs de [Sonnenburg *et coll.*, 2006a, Sonnenburg *et coll.*, 2006b] adoptent une voie différente de celle de [Bach *et coll.*, 2004] en reformulant (B.1) comme un programme linéaire semi-infini (*Semi-Infinite Linear Program*, SILP) :

$$\begin{array}{ll} \underset{\delta,\,\boldsymbol{\mu}}{\text{maximiser}} & \delta \\ \text{tel que} & \left\{\begin{array}{l} \forall\boldsymbol{\alpha}\in\mathbb{R}^n \,/\, 0\preccurlyeq\boldsymbol{\alpha}\preccurlyeq C\mathbb{1},\ \boldsymbol{y}^T\boldsymbol{\alpha}=0\colon \displaystyle\sum_{\boldsymbol{\theta}\in\mathcal{A}}\mu_{\boldsymbol{\theta}}S_{\boldsymbol{\theta}}(\boldsymbol{\alpha})\geq\delta \\ \boldsymbol{\mu}\succcurlyeq 0 \\ \mathbb{1}^T\boldsymbol{\mu}=1, \end{array}\right. \end{array} \tag{B.2}$$

avec

$$S_{\boldsymbol{\theta}}(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{1 \leq i,j \leq n} \alpha_i \alpha_j y_i y_j k_{\boldsymbol{\theta}}(\boldsymbol{x_i}, \boldsymbol{x_j}) - \sum_{i=1}^{n} \alpha_i.$$

Notons que la première contrainte de (B.2) peut aussi s'écrire :

$$\min_{\substack{0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C\mathbb{1} \\ \boldsymbol{y}^T \boldsymbol{\alpha} = 0}} \sum_{\boldsymbol{\theta} \in \mathcal{A}} \mu_{\boldsymbol{\theta}} S_{\boldsymbol{\theta}}(\boldsymbol{\alpha}) \geq \delta,$$

et que $\underset{\substack{0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C\mathbb{1} \\ \boldsymbol{y}^T \boldsymbol{\alpha} = 0}}{\arg\min} \sum_{\boldsymbol{\theta} \in \mathcal{A}} \mu_{\boldsymbol{\theta}} S_{\boldsymbol{\theta}}(\boldsymbol{\alpha})$ correspond à l'ensemble des vecteurs duaux $\boldsymbol{\alpha}$ d'une machine à vecteurs supports (*Support Vector Machine*, SVM) de noyaux $\sum_{\boldsymbol{\theta} \in \mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}$. Ceci éclaire quant à l'algorithme de génération de colonnes utilisé dans [Sonnenburg *et coll.*, 2006a, Sonnenburg *et coll.*, 2006b] qui alterne la résolution d'un programme linéaire (*Linear Program*, LP) pour déterminer $(\delta, \boldsymbol{\mu})$ (en vérifiant la première contrainte de (B.2) pour un certain nombre de vecteurs $\boldsymbol{\alpha}$ fixés) et la résolution d'un programme quadratique (*Quadratic Program*, QP) pour générer une nouvelle colonne (SVM à $\boldsymbol{\mu}$ fixé afin d'ajouter un nouveau vecteur $\boldsymbol{\alpha}$ à l'ensemble déterminant le relâchement de la contrainte de (B.2)).

Une autre formulation issue de (B.1) et permettant de lever l'indifférentiabilité est décrite dans [Rakotomamonjy *et coll.*, 2008]. Les auteurs de [Rakotomamonjy *et coll.*, 2008] remarquent (indépendamment de [Zien et Ong, 2007] qui mentionne une remarque similaire dans le cas d'un problème multiclasse) que :

$$\left(\sum_{\boldsymbol{\theta} \in \mathcal{A}} \|f_{\boldsymbol{\theta}}\|_{\mathcal{H}_{\boldsymbol{\theta}}} \right)^2 = \min_{\substack{\boldsymbol{\mu} \succcurlyeq 0, \\ \mathbb{1}^T \boldsymbol{\mu} = 1}} \sum_{\boldsymbol{\theta} \in \mathcal{A}} \frac{\|f_{\boldsymbol{\theta}}\|_{\mathcal{H}_{\boldsymbol{\theta}}}}{\mu_{\boldsymbol{\theta}}}, \tag{B.3}$$

avec les conventions $[\mu_{\boldsymbol{\theta}} = 0] \Rightarrow [f_{\boldsymbol{\theta}} = 0]$ et $\frac{0}{0} = 0$. La norme mixte et non-différentiable de (B.1) est donc égale au minimum d'une nouvelle fonction objectif différentiable et convexe. En outre, même si le nombre de variables d'optimisation croît (on introduit $\boldsymbol{\mu}$), les auteurs de [Rakotomamonjy *et coll.*, 2008] montrent qu'il existe des algorithmes de résolution efficaces. En intégrant (B.3) à (B.1), on obtient un nouveau problème d'optimisation pour l'apprentissage d'un noyau multiple convexe :

$$\begin{array}{ll} \underset{\{f_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \mathcal{A}}, b, \boldsymbol{\xi}, \boldsymbol{\mu}}{\text{minimiser}} & \frac{1}{2} \sum_{\boldsymbol{\theta} \in \mathcal{A}} \frac{\|f_{\boldsymbol{\theta}}\|^2_{\mathcal{H}_{\boldsymbol{\theta}}}}{\mu_{\boldsymbol{\theta}}} + C \sum_{i=1}^{n} \xi_i \\ \text{tel que} & \begin{cases} y_i \left(\sum_{\boldsymbol{\theta} \in \mathcal{A}} f_{\boldsymbol{\theta}}(\boldsymbol{x_i}) + b \right) \geq 1 - \xi_i, \ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \boldsymbol{\mu} \succcurlyeq 0 \\ \mathbb{1}^T \boldsymbol{\mu} = 1 \end{cases} \end{array} \tag{B.4}$$

Par convexité [Rakotomamonjy *et coll.*, 2008], (B.4) peut être aussi formulé :

$$\begin{array}{ll} \underset{\boldsymbol{\mu}}{\text{minimiser}} & J_{\text{lin}}(\boldsymbol{\mu}) \\ \text{tel que} & \begin{cases} \boldsymbol{\mu} \succcurlyeq 0 \\ \mathbb{1}^T \boldsymbol{\mu} = 1, \end{cases} \end{array} \tag{B.5}$$

où

$$J_{\text{lin}}(\boldsymbol{\mu}) = \left\{ \begin{array}{ll} \min\limits_{\{f_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta} \in \mathcal{A}}, b, \boldsymbol{\xi}} & \frac{1}{2} \sum\limits_{\boldsymbol{\theta} \in \mathcal{A}} \frac{\|f_{\boldsymbol{\theta}}\|^2_{\mathcal{H}_{\boldsymbol{\theta}}}}{\mu_{\boldsymbol{\theta}}} + C \sum\limits_{i=1}^{n} \xi_i \\ \text{tel que} & \begin{cases} y_i \left(\sum\limits_{\boldsymbol{\theta} \in \mathcal{A}} f_{\boldsymbol{\theta}}(\boldsymbol{x_i}) + b \right) \geq 1 - \xi_i, \ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0, \end{cases} \end{array} \right.$$

Il apparaît alors [Rakotomamonjy *et coll.*, 2008] que

$$J_{\text{lin}}(\boldsymbol{\mu}) = \left\{ \begin{array}{ll} \min\limits_{f,\, b,\, \boldsymbol{\xi}} & \frac{1}{2}\left\|f\right\|^2_{\mathcal{H}_{[\boldsymbol{\mu}]}} + C\sum\limits_{i=1}^{n}\xi_i \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(f(\boldsymbol{x_i}) + b\right) \geq 1 - \xi_i,\ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \end{array} \right. \end{array} \right\} = J_{\text{RR}}\left(\sum_{\boldsymbol{\theta}\in\mathcal{A}} \mu_{\boldsymbol{\theta}} k_{\boldsymbol{\theta}}\right).$$

Le gradient de J_{lin} étant aisément calculable grâce au théorème 1.4.2, le problème (B.5) (et par ce biais (B.4)) est résolu par une descente de gradient réduit dans [Rakotomamonjy *et coll.*, 2008]. Dans le cas de (B.5), ce théorème est applicable à condition que la solution SVM à noyau fixe soit unique. Ceci est assuré si l'ensemble d'apprentissage ne contient pas de données répétées et si le noyau utilisé est défini positif (ce qui est le cas pour le noyau gaussien par exemple).

Par la suite, cette formulation MKL a été étendue à une infinité de noyaux [Gehler et Nowozin, 2008a]. Gehler *et coll.* donnent la possibilité à l'ensemble de noyaux $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{A}}$ d'être automatiquement extrait de $(k_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\mathcal{P}}$, indexé par un ensemble $\mathcal{P}$ infini :

$$\begin{array}{ll} \inf\limits_{\mathcal{A}\subset\mathcal{P},\,\mathcal{A}\text{ fini}} \min\limits_{\{f_{\boldsymbol{\theta}}\}_{\boldsymbol{\theta}\in\mathcal{A}},\, b,\, \boldsymbol{\xi},\, \boldsymbol{\mu}} & \frac{1}{2}\sum\limits_{\boldsymbol{\theta}\in\mathcal{A}} \frac{\left\|f_{\boldsymbol{\theta}}\right\|^2_{\mathcal{H}_{\boldsymbol{\theta}}}}{\mu_{\boldsymbol{\theta}}} + C\sum\limits_{i=1}^{n}\xi_i \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(\sum\limits_{\boldsymbol{\theta}\in\mathcal{A}} f_{\boldsymbol{\theta}}(\boldsymbol{x_i}) + b\right) \geq 1 - \xi_i,\ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0 \\ \boldsymbol{\mu} \succcurlyeq 0 \\ \mathbb{1}^T\boldsymbol{\mu} = 1. \end{array} \right. \end{array} \tag{B.6}$$

Une formulation duale de ce problème est (1.5).

Récemment, certains défauts des approches MKL classiques (en particulier celles fondées sur le risque régularisé) ont été soulevés : Gai *et coll.* relèvent notamment que la normalisation de chaque noyau générateur joue (à tort) un grand rôle dans le processus de sélection (problème d'*initialisation*) et qu'il est possible de faire arbitrairement croître la marge SVM en multipliant le noyau principal par une constante positive (problème de *normalisation*) [Gai *et coll.*, 2010]. Pour pallier ces deux principaux défauts, Gai *et coll.* construisent un nouveau problème d'optimisation, nommé *apprentissage de noyau fondé sur le rayon*. Celui-ci est identique à une SVM en remplaçant la régularisation en norme ℓ_2, $\psi\colon f \mapsto \frac{1}{2}\left\|f\right\|^2_{\mathcal{H}}$, par $\psi\colon (f,k) \mapsto \frac{1}{2}\rho^2(k)\left\|f\right\|^2_{\mathcal{H}}$, où $\rho(k)$ est le rayon de la plus petite boule (en norme ℓ_2) comprenant les données. Cette régularisation correspond à la borne rayon-marge précédemment mentionnée. Le nouveau problème d'apprentissage de noyau multiple (dans le cas linéaire) devient :

$$\begin{array}{ll} \underset{\boldsymbol{\mu}}{\text{minimiser}} & J_{\text{RKL}}\left(\boldsymbol{\mu}\right) \\ \text{tel que} & \boldsymbol{\mu} \succcurlyeq 0, \end{array} \tag{B.7}$$

où

$$J_{\text{RKL}}\left(\boldsymbol{\mu}\right) = \left\{ \begin{array}{ll} \min\limits_{f,\, b,\, \boldsymbol{\xi}} & \frac{1}{2}\rho^2(k_{[\boldsymbol{\mu}]})\left\|f\right\|^2_{\mathcal{H}_{[\boldsymbol{\mu}]}} + C\sum\limits_{i=1}^{n}\xi_i \\ \text{tel que} & \left\{ \begin{array}{l} y_i\left(f(\boldsymbol{x_i}) + b\right) \geq 1 - \xi_i,\ \forall i \in \mathbb{N}_n \\ \boldsymbol{\xi} \succcurlyeq 0. \end{array} \right. \end{array} \right.$$

Il est alors démontré que (B.7) résout les problèmes de normalisation et d'initialisation (la fonction de régularisation ψ est invariante à toute multiplication du noyau par un facteur positif) et qu'il est invariant aux contraintes en norme sur le vecteur de pondération $\boldsymbol{\mu}$ (*i.e.* ajouter une contrainte du type $\left\|\boldsymbol{\mu}\right\|_{\ell_p} \leq 1$ ne change aucunement les solutions de (B.7)).

Notons que (B.7) (résolu par gradient projeté dans [Gai *et coll.*, 2010]) contient deux optimisations internes (l'une pour le calcul du rayon et l'autre pour la détermination de la SVM à noyau fixé) et est différent de l'approche rayon-marge proposée dans [Chapelle *et coll.*, 2002]. Dans cette dernière étude, la fonction de perte considérée est quadratique (tandis que Gai *et coll.* utilisent la perte charnière) et pénalisée par $C\rho^2(k_{[\boldsymbol{\mu}]})$ au lieu de C dans [Gai *et coll.*, 2010].

ANNEXE C

Synthèse de l'apprentissage de descripteurs

SEC.	TRAVAUX REPRÉSENTATIFS	CADRE	PRÉ-TRAITEMENT	CLASSIFIEUR	COÛT	MÉTHODE
2.4.1	[Heitz, 1995]	Optimisation	Trans. bilinéaire gaussienne	Distance minimum	Fisher	Indéterminée
	[Droppo et Atlas, 1998]	Optimisation	Trans. bilinéaire	Distance minimum	Fisher	Exhaustif
	[Davy et Doncarli, 1998]	Optimisation	Trans. bilinéaire gaussienne	Distance minimum	Fisher	Gradient
	[Davy *et coll.*, 2001]	Bayésien	Trans. bilinéaire gaussienne	Distance minimum	Estimation prob. d'err.	Nelder-Mead
	[Davy *et coll.*, 2002]	Bayésien	Trans. bilinéaire gaussienne	SVM	Estimation prob. d'err.	Nelder-Mead
	[Honeine *et coll.*, 2006]	Optimisation	Trans. bilinéaire gaussienne	SVM	Alignement de noyaux	Algorithme glouton
	[Honeine, 2007]	Optimisation	Trans. bilinéaire gaussienne	SVM	Alignement de noyaux	Gradient
2.4.2	[Biem *et coll.*, 2001]	Optimisation	BdF linéaire	Distance minimum	Logistique, distance algébrique	Gradient
	[Suk et Lee, 2013]	Bayésien	BdF linéaire + CSP	SVM	Vraisemblance	Filtre particulaire
	[Flamary *et coll.*, 2012]	Optimisation	Filtre linéaire	SVM	Risque structurel	Gradient
	[Vignolo *et coll.*, 2011a]	Optimisation	BdF linéaire	HMM	Validation	Algorithme génétique
2.4.3	[Fukushima, 1980, LeCun, 1989]	Optimisation	BdF non-linéaire	MLP	Moindre carré	Rétropropagation d'erreur
	[Zhong et Ghosh, 2000]	Optimisation	BdF non-linéaire	MLP	Risque structurel	Gradient
	[Tang, 2013]	Optimisation	BdF non-linéaire	SVM	Risque structurel	Gradient
2.4.4	[Saito et Coifman, 1995]	Optimisation	BdF d'ondelettes	LDA / arbre	Divergence de Kullback-Leibler	Algorithme glouton
	[Strauss et Steidl, 2002]	Optimisation	BdF d'ondelettes	SVM	Marge SVM	Grille
	[Strauss *et coll.*, 2003]	Optimisation	BdF d'ondelettes	LDA / SVM	Divergence de Kullback-Leibler	Algorithme génétique
	[Neumann *et coll.*, 2005]	Optimisation	BdF d'ondelettes	SVM	5 coûts différents	Grille
	[Farina *et coll.*, 2007]	Optimisation	BdF d'ondelettes	SVM	Validation croisée	Grille
	[Yger et Rakotomamonjy, 2011]	Optimisation	BdF d'ondelettes	SVM	Risque structurel	Ensemble actif, gradient
2.4.6	[Rodriguez et Sapiro, 2008]	Optimisation	Dictionnaire	Résidu minimum	Fisher + moindres carrés	Alg. glouton, val. singulières
	[Mairal *et coll.*, 2009]	Optimisation	Dictionnaire	Résidu minimum	Logistique + moindres carrés + ℓ_1	Gradient

TABLE C.1 – Algorithmes représentatifs d'apprentissage de descripteurs discriminants.

ANNEXE D

Compléments sur la transformée de Cohen

Étant donné un signal s de $L^2(\mathbb{R})$, la distribution de Wigner-Ville correspond à la transformée de Fourier de l'auto-corrélation instantanée du signal s par rapport au retard τ :

$$W_s\colon (t,\omega) \in \mathbb{R}^2 \mapsto \int_{-\infty}^{+\infty} s(t+\frac{\tau}{2})\bar{s}(t-\frac{\tau}{2})\mathrm{e}^{-2\imath\pi\omega\tau}\,d\tau,$$

où $\bar{\cdot}$ désigne l'opération de conjugaison. Lorsque le signal s est la somme de plusieurs contributions, cette transformée génère des termes de corrélation croisée, pénalisant ainsi l'interprétation de la représentation du signal dans le plan Temps-Fréquence (TF).

La classe de Cohen, est une tentative d'unification des transformées TF bilinéaires (Wigner-Ville, Rihaczek, *etc.*). En appelant E_s l'énergie totale du signal s, une distribution de Cohen C_s répond donc à la conservation d'énergie :

$$E_s = \iint_{-\infty}^{+\infty} C_s(t,\omega)\,dt\,d\omega,$$

et s'exprime, à l'instar de la distribution de Wigner-Ville, en fonction de l'auto-corrélation du signal s. Il existe une multitude de chemins qui répondent à ces deux propriétés. La classe de Cohen se veut l'ensemble des distributions d'énergie bilinéaires dites *covariantes par translations temporelle et fréquentielle* :

$$C_s\colon (t,\omega) \in \mathbb{R}^2 \mapsto \iint_{-\infty}^{+\infty} \left(K(t_1-t,t_2-t)\mathrm{e}^{-2\imath\pi\omega(t_1-t_2)}\right) s(t_1)\bar{s}(t_2)\,dt_1\,dt_2,$$

où l'expression entre parenthèses représente un noyau de pondération de l'auto-corrélation. Moyennant quelques manipulations, la distribution de Cohen offre son premier visage, convolution de la distribution de Wigner-Ville et d'un nouveau noyau $\hat{\Phi}$ lié à K :

$$C_s(t,\omega) = \iint_{-\infty}^{+\infty} \hat{\Phi}(\nu-t,\xi-\omega)W_s(\nu,\xi)\,d\nu\,d\xi.$$

Sous cette forme, la classe de Cohen apparaît comme une extension de la transformée de Wigner-Ville, munie d'un noyau $\hat{\Phi}$ visant à atténuer les termes de corrélation croisée.

Plus souvent, la classe de Cohen est définie dans l'espace Doppler-retard (ξ,τ) :

$$C_s(t,\omega) = \iint_{-\infty}^{+\infty} A_s(\xi,\tau)\Phi(\xi,\tau)e^{-\imath(\xi t+\tau\omega)}\,d\xi\,d\tau,$$

où A_s est la fonction d'ambiguïté à bande étroite du signal s et Φ est appelé *noyau de Cohen*. Ces deux fonctions sont liées respectivement à W_s et à $\hat{\Phi}$ par une transformée de Fourier inverse en fréquence et directe en temps. En particulier, la fonction d'ambiguïté à bande étroite s'exprime par :

$$A_s\colon (\xi,\tau) \in \mathbb{R}^2 \mapsto \frac{1}{2\pi}\int_{-\infty}^{+\infty} s(\nu+\frac{\tau}{2})\bar{s}(\nu-\frac{\tau}{2})e^{i\xi\nu}d\nu.$$

ANNEXE E

Coefficients cepstraux MEL

Les coefficients cepstraux mel-fréquences (*Mel-Frequency Cepstral Coefficients*, MFCC) [Stevens *et coll.*, 1937] sont apparus dans le cadre de la reconnaissance automatique de la parole et ont évolué par la suite vers l'un des ensembles de descripteurs standards en reconnaissance audio. L'analyse cepstrale a été mise en place de manière à imiter le fonctionnement de la cochlée. Elle suppose que la voix est issue d'un système linéaire, autrement dit, un signal vocal s'écrit comme la convolution d'un signal d'excitation et d'une composante vocale pure [Vignolo *et coll.*, 2011a]. Par transformée de Fourier et logarithme, ces deux contributions s'associent alors de manière additive et sont plus aisément séparables que dans le domaine temporel. Ainsi, les coefficients cepstraux MÉLodie (MEL) visent à représenter cette deuxième contribution : l'information de timbre (*i.e.* l'enveloppe cepstrale) d'un signal et sont calculés suivant le diagramme de la figure E.1.

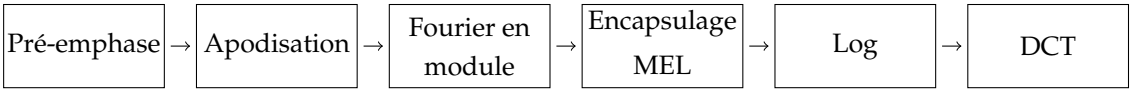


FIGURE E.1 – Diagramme de calcul des coefficients cepstraux MEL.

Le premier module de pré-emphase a pour but de rehausser les hautes fréquences. Il est mis en pratique par une convolution avec la Réponse Impulsionnelle (RI) $[1, -\epsilon]$ (filtrage passe-haut), où ϵ peut être fixé à 0.97. La seconde étape d'apodisation est usuelle avant de calculer une transformée de Fourier (qui suppose la périodicité du signal). Seul le module de la représentation de Fourier est considéré, afin d'obtenir la distribution spectrale de l'énergie. Cette dernière est décrite de manière discrète et uniforme en fréquence. Bien qu'informative, cette description ne représente pas encore le fonctionnement de la cochlée. Cette distribution spectrale est donc ensuite agrégée de manière uniforme sur l'échelle MEL, qui est un modèle de perception psycho-acoustique de l'énergie d'un signal (d'où l'origine : *mélodie*). L'échelle MEL est donnée par la formule :

$$f_{\text{MEL}} = 2595 \log_{10}\left(1 + \frac{f_{Hz}}{700}\right),$$

où f_{MEL} représente la fréquence MEL et f_{Hz} la fréquence en Hz. Elle a été déterminée de sorte que 1000 Mel correspondent à 1000 Hz. La figure E.2 page suivante donne une idée du lien logarithmique entre l'échelle MEL et l'échelle hertzienne. Cette dernière s'arrête généralement à 8000 Hz puisque l'on considère qu'il n'y a que peu d'information au dela de

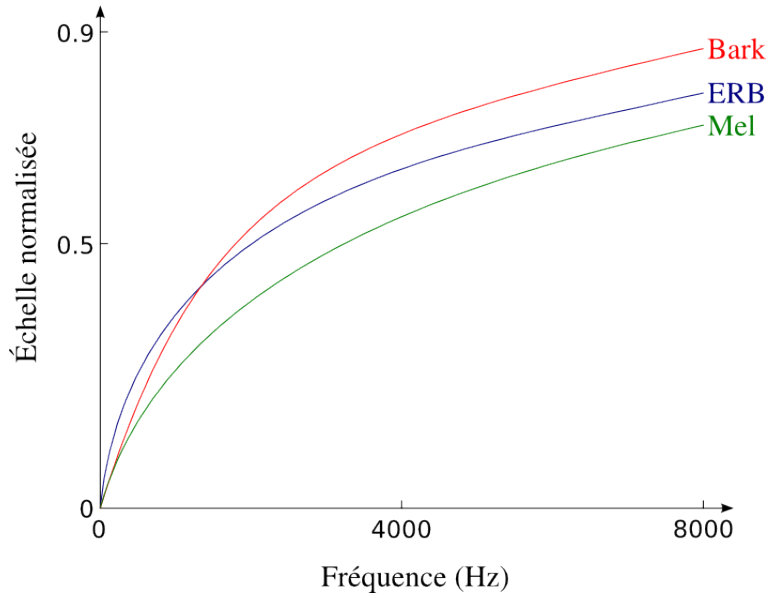


FIGURE E.2 – Comparaison des échelles Mel, Bark et ERB normalisées à 22050 Hz.

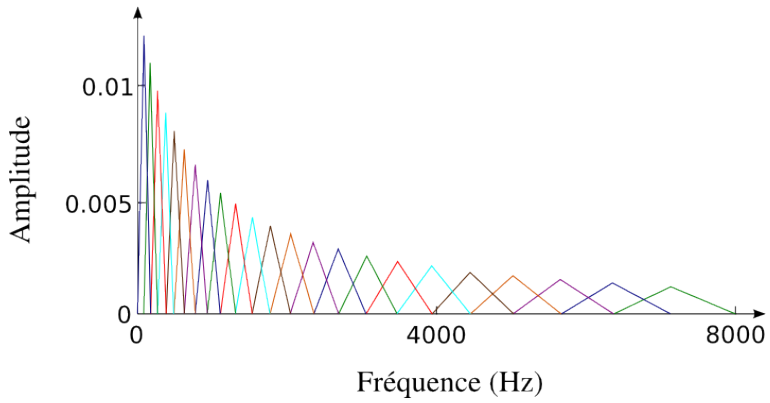


FIGURE E.3 – Banc de filtres Mel. De manière usuelle, on considère 40 filtres MEL entre 0 et 8000 Hz.

6800 Hz pour les signaux de parole humaine. Il est évidemment possible d'aller plus loin en fréquence pour des signaux audio quelconques. L'illustration représente aussi les échelles Bark et ERB, qui sont deux alternatives à l'échelle MEL. D'un point de vue pratique, cette agrégation est réalisée par pondération du module de la représentation de Fourier obtenue précédemment, par un ensemble de distributions triangulaires (dans le domaine spectral) que l'on peut représenter sous la forme du Banc de Filtres (BdF) en figure E.3. De manière similaire au fonctionnement de la cochlée, cette échelle a une faible résolution spectrale pour les hautes fréquences.

Suite à cette agrégation, le cepstre du signal est obtenu par logarithme et l'enveloppe de celui-ci par une transformée en cosinus discrète (*Discrete Cosine Transform*, DCT) (souvent de type II) dont on ne retient à terme que les 13 premiers coefficients. Comme l'illustre la figure E.4 page suivante, cette dernière opération (DCT) décorrèle l'enveloppe (évolution basse fréquence) de l'excitation du cepstre (évolution haute fréquence). Le timbre du signal (son enveloppe spectrale) est donc bien caractérisé par les premiers coefficients de la représentation en cosinus discrète.

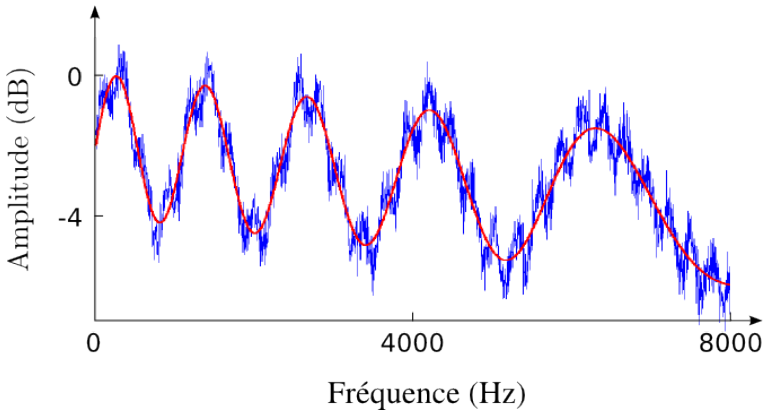


FIGURE E.4 – Enveloppe et excitation d'un cepstre.

ANNEXE F

Compléments sur la transformée en ondelettes

Une transformée en ondelettes exploite une ondelette ψ afin de représenter un signal grâce une analyse multi-résolution. Cette dernière est générée par une fonction d'échelle ϕ (la notion d'échelle est directement liée à celle de résolution) issue de l'ondelette ψ.

Définition F.0.1 (Approximation multi-résolution [Mallat, 1999, p. 301])**.**
Une suite $(\mathcal{V}_j)_{j\in\mathbb{Z}}$ de sous-ensembles fermés de $L^2(\mathbb{R})$ est une approximation multi-résolution de $L^2(\mathbb{R})$ si :

$$\forall (j,k)\in\mathbb{Z}^2,\quad f\in\mathcal{V}_j \Rightarrow (t\mapsto f(t-2^j k))\in\mathcal{V}_j \tag{F.1}$$

$$\forall j\in\mathbb{Z},\quad \mathcal{V}_j\subset\mathcal{V}_{j+1} \tag{F.2}$$

$$\forall j\in\mathbb{Z},\quad f\in\mathcal{V}_j \Leftrightarrow (t\mapsto f(2t))\in\mathcal{V}_{j+1} \tag{F.3}$$

$$\lim_{j\to-\infty}\mathcal{V}_j=\bigcap_{j=-\infty}^{+\infty}\mathcal{V}_j=\{0\} \tag{F.4}$$

$$\lim_{j\to+\infty}\mathcal{V}_j=\overline{\bigcup_{j=-\infty}^{+\infty}\mathcal{V}_j}=L^2(\mathbb{R}) \tag{F.5}$$

$$\exists\phi\in\mathcal{V}_0\ /\quad \{t\mapsto\phi(t-k)\}_{k\in\mathbb{Z}}\text{ est une base de Riesz de }\mathcal{V}_0. \tag{F.6}$$

La propriété (F.1) signifie que $\mathcal{V}_j$ est invariant à toutes les translations proportionnelles à 2^j. L'espace $\mathcal{V}_j$ peut donc être assimilé à une grille uniforme de pas 2^j, caractérisant l'approximation à la résolution 2^j. L'inclusion (F.2) est une propriété de causalité traduisant le fait qu'une approximation à la résolution 2^{j+1} contient toute l'information nécessaire pour calculer une approximation à une résolution plus grossière 2^j. L'équivalence (F.3) assure que dilater une fonction de $\mathcal{V}_j$ permet de définir une approximation à une résolution plus fine de 2^{j+1}. Lorsque la résolution 2^j tend vers 0, (F.4) indique que l'on perd tous les détails. En revanche, lorsque la résolution 2^j tend vers $+\infty$, (F.5) impose que l'approximation d'une fonction converge vers la fonction elle-même.

La fonction ϕ génératrice d'une base de $\mathcal{V}_0$ est appelée *fonction d'échelle*. On peut alors vérifier qu'en définissant $\phi_{j,k}\colon t\mapsto\sqrt{2^j}\phi(2^j t-k)$, $\{\phi_{1,k}\}_{k\in\mathbb{Z}}$ est une base de Riesz de $\mathcal{V}_1$. Soit $\boldsymbol{h}$ un vecteur (potentiellement infini mais de norme finie) défini par $h_k=\langle\phi\mid\phi_{1,k}\rangle_{L^2(\mathbb{R})}$ (pour tout k de $\mathbb{Z}$). Alors on obtient l'équation d'échelle [Mallat, 1999] ou de dilatation [Strang et

Nguyen, 1996] :

$$\forall t \in \mathbb{R}: \quad \phi(t) = \sum_{k\in\mathbb{Z}} \sqrt{2} h_k \phi(2t-k).$$

Fixons maintenant un entier naturel j non nul. L'ensemble $\mathcal{V}_j$ est un espace d'approximation de résolution définie. Considérons le complément orthogonal $\mathcal{W}_j$ de $\mathcal{V}_j$ dans $\mathcal{V}_{j+1}$:

$$\mathcal{V}_{j+1} = \mathcal{W}_j \oplus \mathcal{V}_j.$$

Si l'on définit :

$$\forall t \in \mathcal{R}: \quad \psi(t) = \sqrt{2} \sum_{k\in\mathbb{Z}} (-1)^k h_{1-k} \phi(2t-k),$$

on peut alors montrer qu'en nommant $\psi_{j,k}: t \mapsto \sqrt{2^j}\psi(2^j t - k)$, $\{\psi_{j,k}\}_{k\in\mathbb{Z}}$ est une base de Riesz de l'espace de détails $\mathcal{W}_j$ [Mallat, 1999, théo. 7.3, p. 320]. Ainsi, étant donnée une fonction f de $L^2(\mathbb{R})$ et un entier naturel m, f peut être approximée par une certaine fonction f_m de $\mathcal{V}_m$ qui elle-même est décomposable sous la forme :

$$f_m = f_{m-1} + g_{m-1} = f_{m-2} + g_{m-2} + g_{m-1} = \cdots = f_0 + \sum_{j=0}^{m-1} g_j, \tag{F.7}$$

où pour tout j, f_j est une approximation de $\mathcal{V}_j$ et g_j est une fonction de détails de $\mathcal{W}_j$. Les différentes fonctions d'approximation et de détails de (F.7) sont uniques puisque $\mathcal{V}_{j+1} = \mathcal{W}_j \oplus \mathcal{V}_j$ pour tout j. En revanche, lorsque f_m est obtenue par projection orthogonale de f sur $\mathcal{V}_m$, (F.7) est alors appelée *décomposition en ondelettes* de f.

Par simplicité, une considère maintenant une ondelette ψ orthogonale, *i.e.* les bases précédemment énoncées pour $\mathcal{V}_j$ et $\mathcal{W}_j$ (quel que soit j) sont orthonormales. Une représentation en ondelettes cherche alors les vecteurs de coordonnées $\boldsymbol{a}^{(j)}$ de f_j dans $\mathcal{V}_j$, et $\boldsymbol{d}^{(j)}$ de g_j dans $\mathcal{W}_j$ (pour $j \in \mathbb{N}_m$ et suivant (F.7)). Ceux-ci sont donnés par :

$$\forall k \in \mathbb{Z}: \quad a_k^{(j)} = \langle f \mid \phi_{j,k} \rangle_{L^2(\mathbb{R})}, \quad d_k^{(j)} = \langle f \mid \psi_{j,k} \rangle_{L^2(\mathbb{R})}.$$

Nous abordons à présent une manière efficace d'obtenir cette représentation en ondelettes. Si ϕ est intégrable, alors la transformée de Fourier continue $\hat{h}$ de $\boldsymbol{h}$ vérifie [Mallat, 1999] :

$$\forall \omega \in \mathbb{R}: \quad |\hat{h}(\omega)|^2 + |\hat{h}(\omega+\pi)|^2 = 2.$$

La propriété précédente indique que $\boldsymbol{h}$ est un filtre miroir conjugué. Ce type de filtres joue un rôle important en traitement du signal. Il rend possible la décomposition d'un signal discret en deux bandes de fréquences distinctes à l'aide d'un BdF. En effet, appelons $\boldsymbol{g}$ le vecteur défini par :

$$\forall k \in \mathbb{Z}: \quad g_k = (-1)^k h_{1-k}.$$

Alors $\boldsymbol{h}$ et $\boldsymbol{g}$ sont deux filtres dits *miroirs et en quadrature*. $\boldsymbol{h}$ est un filtre passe-bas tandis que $\boldsymbol{g}$ est un filtre passe-haut. Le théorème 2.4.1 permet alors de construire un BdF pyramidal, associé à une décomposition en ondelettes (figure 2.6 page 48), à partir de $\boldsymbol{h}$ et $\boldsymbol{g}$. Les filtres mentionnés dans le théorème 2.4.1 correspondent respectivement aux vecteurs $\boldsymbol{h}$ et $\boldsymbol{g}$ (définis ici) *retournés*.

ANNEXE G

Dualité d'une SVM linéaire à poids positifs

Nous détaillons ici le calcul d'un dual du problème :

$$\begin{array}{ll} \underset{\mathbf{w},\boldsymbol{\xi},u,v}{\text{minimiser}} & (1-\lambda)\boldsymbol{\mu}^T\mathbf{w}+\lambda(u+v)+C\mathbb{1}^T\boldsymbol{\xi} \\ \text{tel que} & \left\{\begin{array}{ll} y_i\left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)})\right\rangle_{\ell_2}-u+v\right)\geq 1-\xi_i, \forall i\in\mathbb{N}_n & :\boldsymbol{\alpha} \\ 0\preccurlyeq\boldsymbol{\xi} & :\boldsymbol{\tau} \\ 0\preccurlyeq\mathbf{w} & :\boldsymbol{\beta} \\ 0\leq u,v & :\delta,\epsilon, \end{array}\right. \end{array}$$

ainsi que ses conditions d'optimalité KKT. Dans l'expression de ce problème, la variable duale de chaque contrainte est indiquée à sa droite.

Le lagrangien s'exprime alors par :

$$\begin{aligned} &\forall(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon)\in\mathbb{R}^r\times\mathbb{R}^n\times\mathbb{R}\times\mathbb{R}\times\mathbb{R}^n\times\mathbb{R}^n\times\mathbb{R}^r\times\mathbb{R}\times\mathbb{R}: \\ &\mathfrak{L}(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon)=(1-\lambda)\boldsymbol{\mu}^T\mathbf{w}+\lambda(u+v)+C\mathbb{1}^T\boldsymbol{\xi} \\ &\qquad\qquad+\sum_{i=1}^n\alpha_i\left(1-\xi_i-y_i\left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)})\right\rangle_{\ell_2}-u+v\right)\right) \\ &\qquad\qquad-\boldsymbol{\tau}^T\boldsymbol{\xi}-\boldsymbol{\beta}^T\mathbf{w}-\delta u-\epsilon v. \end{aligned}$$

On cherche maintenant un minimum de $\mathfrak{L}$ par rapport aux variables primales. Pour ce faire, on annule ses gradients :

$$\begin{aligned} \nabla_{\mathbf{w}}\mathfrak{L}(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon)&=(1-\lambda)\boldsymbol{\mu}-\sum_{i=1}^n\alpha_i y_i\psi_T^{\mathcal{L}}(x_{\sim}^{(i)})-\boldsymbol{\beta}=0 \\ \nabla_{\boldsymbol{\xi}}\mathfrak{L}(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon)&=\boldsymbol{C}\mathbb{1}-\boldsymbol{\alpha}-\boldsymbol{\tau}=0 \\ \nabla_u\mathfrak{L}(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon)&=\lambda+\boldsymbol{y}^T\boldsymbol{\alpha}-\delta=0 \\ \nabla_v\mathfrak{L}(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon)&=\lambda-\boldsymbol{y}^T\boldsymbol{\alpha}-\epsilon=0, \end{aligned}$$

i.e.

$$\begin{aligned}\boldsymbol{\beta} &= (1-\lambda)\boldsymbol{\mu} - \sum_{i=1}^{n} \alpha_i y_i \psi_T^{\mathcal{L}}(x_{\sim}^{(i)}) \\ \boldsymbol{\tau} &= \boldsymbol{C}\mathbb{1} - \boldsymbol{\alpha} \\ \delta &= \lambda + \boldsymbol{y}^T\boldsymbol{\alpha} \\ \epsilon &= \lambda - \boldsymbol{y}^T\boldsymbol{\alpha},\end{aligned} \tag{G.1}$$

d'où

$$\min_{\mathbf{w},\boldsymbol{\xi},u,v} \mathfrak{L}(\mathbf{w},\boldsymbol{\xi},u,v,\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta},\delta,\epsilon) = \mathbb{1}^T\boldsymbol{\alpha}.$$

Pour obtenir un problème d'optimisation dual, les variables duales doivent être admissibles, *i.e.* $\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta} \succcurlyeq 0$ et $\delta,\epsilon \geq 0$. Nous obtenons alors de (G.1) les conditions suivantes :

$$\begin{aligned}\boldsymbol{Q}^{\mathcal{L}}\boldsymbol{\alpha} &\preccurlyeq (1-\lambda)\boldsymbol{\mu} \\ \boldsymbol{\alpha} &\preccurlyeq \boldsymbol{C}\mathbb{1} \\ -\lambda \leq \boldsymbol{y}^T\boldsymbol{\alpha} &\leq \lambda,\end{aligned}$$

où $\boldsymbol{Q}^{\mathcal{L}}$ est une matrice de $\mathbb{R}^{r\times n}$ définie par : $\boldsymbol{Q}^{\mathcal{L}} = \left[y_1\psi_T^{\mathcal{L}}(x_{\sim}^{(1)}),\ldots,y_n\psi_T^{\mathcal{L}}(x_{\sim}^{(n)})\right]$.

Ainsi, un problème d'optimisation dual du problème énoncé plus haut est :

$$\begin{array}{ll}\underset{\boldsymbol{\alpha}\in\mathbb{R}^n}{\text{maximiser}} & \mathbb{1}^T\boldsymbol{\alpha} \\ \text{tel que} & \left\{\begin{array}{l} 0 \preccurlyeq \boldsymbol{\alpha} \preccurlyeq C\mathbb{1} \\ \boldsymbol{Q}^{\mathcal{L}}\boldsymbol{\alpha} \preccurlyeq (1-\lambda)\boldsymbol{\mu} \\ -\lambda \leq \boldsymbol{\alpha}^T\boldsymbol{y} \leq \lambda. \end{array}\right.\end{array}$$

Nous cherchons maintenant à caractériser l'optimalité : les conditions KKT assurent qu'un minimum global du problème primal est atteint lorsque les gradients de $\mathfrak{L}$ par rapport aux variables primales sont nuls, que les variables primales sont admissibles ($\mathbf{w},\boldsymbol{\xi} \succcurlyeq 0$ et $u,v \geq 0$) et que les variables duales sont admissibles ($\boldsymbol{\alpha},\boldsymbol{\tau},\boldsymbol{\beta} \succcurlyeq 0$ et $\delta,\epsilon \geq 0$) et complémentaires aux contraintes d'inégalité :

$$\begin{aligned}\forall i \in \mathbb{N}_n\colon \alpha_i = 0 \text{ ou } y_i\left(\left\langle \mathbf{w} \mid \psi_T^{\mathcal{L}}(x_{\sim}^{(i)})\right\rangle_{\ell_2} - u + v\right) &> 1-\xi_i \\ \forall i \in \mathbb{N}_n\colon \tau_i = 0 \text{ ou } \xi_i &= 0 \\ \forall j \in \mathbb{N}_r\colon \beta_i = 0 \text{ ou } \mathbf{w}_j &= 0 \\ \delta = 0 \text{ ou } u &= 0 \\ \epsilon = 0 \text{ ou } v &= 0.\end{aligned}$$

En utilisant (G.1) on obtient en particulier la complémentarité suivante : $(\boldsymbol{Q}^{\mathcal{L}}\boldsymbol{\alpha})_j = (1-\lambda)\mu_j$ ou $\mathbf{w}_j = 0$ ($j \in \mathbb{N}_r$).

Bibliographie

[Aharon *et coll.*, 2006] AHARON, M., ELAD, M. et BRUCKSTEIN, A. (2006). K-SVD : An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322. (Cité aux pages 52 et 53.)

[Amores, 2013] AMORES, J. (2013). Multiple instance classification : Review, taxonomy and comparative study. *Artificial Intelligence*, 201:81–105. (Cité à la page 27.)

[Andén et Mallat, 2011] ANDÉN, J. et MALLAT, S. (2011). Multiscale scattering for audio classification. *Dans les actes de International Society for Music Information Retrieval Conference*. (Cité à la page 51.)

[Andén et Mallat, 2014] ANDÉN, J. et MALLAT, S. (2014). Deep scattering spectrum. *IEEE Transactions on Signal Processing*, 62:4114–4128. (Cité aux pages 36, 50, 51, 87, 94 et 120.)

[Andrews et Hofmann, 2004] ANDREWS, S. et HOFMANN, T. (2004). Multiple instance learning via disjunctive programming boosting. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 29.)

[Andrews *et coll.*, 2003] ANDREWS, S., TSOCHANTARIDIS, I. et HOFMANN, T. (2003). Support vector machines for multiple-instance learning. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 28 et 115.)

[Ang *et coll.*, 2008] ANG, K., CHIN, Z., ZHANG, H. et GUAN, C. (2008). Filter bank common spatial pattern (fbcsp) in brain-computer interface. *Dans les actes de IEEE International Joint Conference on Neural Networks, 2008*. (Cité aux pages 42 et 43.)

[Argyriou *et coll.*, 2006] ARGYRIOU, A., HAUSER, R., MICCHELLI, C. et PONTIL, M. (2006). A DC-programming algorithm for kernel selection. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 18 et 25.)

[Argyriou *et coll.*, 2005] ARGYRIOU, A., MICCHELLI, C. et PONTIL, M. (2005). Learning convex combinations of continuously parameterized basic kernels. *Dans les actes de Conference on Learning Theory*, Berlin, Heidelberg. Springer-Verlag. (Cité à la page 25.)

[Aronszajn, 1950] ARONSZAJN, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404. (Cité aux pages 9, 13, 18 et 63.)

[Atlas *et coll.*, 1997] ATLAS, L., DROPPO, J. et MCLAUGHLIN, J. (1997). Optimizing time-frequency distributions for automatic classification. *Dans les actes de SPIE - The International Society for Optical Engineering*. (Cité aux pages 38 et 40.)

[Aucouturier *et coll.*, 2007] AUCOUTURIER, J.-J., DEFREVILLE, B. et PACHET, F. (2007). The bag-of-frames approach to audio pattern recognition : A sufficient model for urban soundscapes but not for polyphonic music. *Journal of the Acoustical Society of America*, 122:881–891. (Cité à la page 54.)

[Bach, 2009] BACH, F. (2009). Exploring large feature spaces with hierarchical multiple kernel learning. *Dans les actes de* KOLLER, D., SCHUURMANS, D., BENGIO, Y. et BOTTOU, L., éditeurs : *Advances in Neural Information Processing Systems*. (Cité aux pages 22 et 24.)

[Bach *et coll.*, 2004] BACH, F., LANCKRIET, G. et JORDAN, M. (2004). Multiple kernel learning, conic duality, and the SMO algorithm. *Dans les actes de International Conference on Machine Learning*, New York, NY, USA. (Cité aux pages 17, 21, 22, 23, 24, 63 et 135.)

[Balcan et Blum, 2006] BALCAN, M.-F. et BLUM, A. (2006). On a theory of learning with similarity functions. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 29, 96, 99 et 111.)

[Balcan *et coll.*, 2008] BALCAN, M.-F., BLUM, A. et SREBRO, N. (2008). A theory of learning with similarity functions. *Machine Learning*, 72:89–112. (Cité aux pages 128 et 131.)

[Baraniuk et Jones, 1993] BARANIUK, R. et JONES, D. (1993). Signal-dependent time-frequency analysis using a radially gaussian kernel. *Signal Processing*, 32(3):263–284. (Cité à la page 39.)

[Barthélemy *et coll.*, 2013] BARTHÉLEMY, Q., SANGNIER, M., LARUE, A. et MARS, J. (2013). Comparaison de descripteurs pour la classification de décompositions parcimonieuses invariantes par translation. *Dans les actes de XXIVème Colloque GRETSI*. (Cité à la page 83.)

[Bartlett et Mendelson, 2002] BARTLETT, P. et MENDELSON, S. (2002). Rademacher and gaussian complexities : Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482. (Cité aux pages 5, 6 et 114.)

[Bengio, 2009] BENGIO, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127. (Cité à la page 94.)

[Bennett *et coll.*, 2006] BENNETT, K., HU, J., JI, X., KUNAPULI, G. et PANG, J.-S. (2006). Model selection via bilevel optimization. *Dans les actes de International Joint Conference on Neural Networks*, pages 1922–1929. (Cité à la page 16.)

[Bennett *et coll.*, 2002] BENNETT, K., MOMMA, M. et EMBRECHTS, M. (2002). MARK : A boosting algorithm for heterogeneous kernel models. *Dans les actes de International Conference on Knowledge Discovery and Data Mining*. ACM. (Cité à la page 21.)

[Bentley *et coll.*, 2011] BENTLEY, P., NORDEHN, G., COIMBRA, M. et S., M. (2011). The pascal classifying heart sounds challenge (chsc). http ://www.peterjbentley.com/heartchallenge/. (Cité à la page 69.)

[Bergstra et Bengio, 2012] BERGSTRA, J. et BENGIO, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305. (Cité à la page 16.)

[Berkelaar *et coll.*, 2004] BERKELAAR, M., EIKLAND, K. et NOTEBAERT, P. (2004). lpsolve : Open source (mixed-integer) linear programming system. Eindhoven University of Technology. (Cité à la page 105.)

[Bi *et coll.*, 2004] BI, J., ZHANG, T. et BENNETT, K. (2004). Column-generation boosting methods for mixture of kernels. *Dans les actes de International Conference on Knowledge Discovery and Data Mining*. ACM. (Cité à la page 22.)

[Biem et Katagiri, 1993] BIEM, A. et KATAGIRI, S. (1993). Feature extraction based on minimum classification error/generalized probabilistic descent method. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 275–278 vol.2. (Cité à la page 42.)

[Biem et Katagiri, 1994] BIEM, A. et KATAGIRI, S. (1994). Filter bank design based on discriminative feature extraction. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume i, pages I/485–I/488 vol.1. (Cité à la page 42.)

[Biem et Katagiri, 1997] BIEM, A. et KATAGIRI, S. (1997). Cepstrum-based filter-bank design using discriminative feature extraction training at various levels. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 1503–1506 vol.2. (Cité à la page 42.)

[Biem *et coll.*, 1993] BIEM, A., KATAGIRI, S. et JUANG, B.-H. (1993). Discriminative feature extraction for speech recognition. *Dans les actes de IEEE Workshop on Neural Networks for Signal Processing*, pages 392–401. (Cité à la page 42.)

[Biem *et coll.*, 1997] BIEM, A., KATAGIRI, S. et JUANG, B.-H. (1997). Pattern recognition using discriminative feature extraction. *IEEE Transactions on Signal Processing*, 45(2):500–504. (Cité à la page 42.)

[Biem *et coll.*, 2001] BIEM, A., KATAGIRI, S., MCDERMOTT, E. et JUANG, B.-H. (2001). An application of discriminative feature extraction to filter-bank-based speech recognition. *IEEE Transactions on Speech and Audio Processing*, 9(2):96–110. (Cité aux pages 42 et 140.)

[Biem *et coll.*, 1996] BIEM, A., MCDERMOTT, E. et KATAGIRI, S. (1996). Discriminative feature extraction application to filter bank design. *Dans les actes de IEEE Workshop on Neural Networks for Signal Processing*, pages 273–282. (Cité à la page 42.)

[Blum et Kalai, 1998] BLUM, A. et KALAI, A. (1998). A note on learning from multiple-instance examples. *Machine Learning*, 30(1):23–29. (Cité à la page 27.)

[Bonnans et Shapiro, 1998] BONNANS, F. et SHAPIRO, A. (1998). Optimization problems with perturbations, a guided tour. *SIAM Review*, 40:228–264. (Cité à la page 22.)

[Boser *et coll.*, 1992] BOSER, B., GUYON, I. et VAPNIK, V. (1992). A training algorithm for optimal margin classifiers. *Dans les actes de Conference on Learning Theory*. (Cité à la page 1.)

[Boureau *et coll.*, 2010a] BOUREAU, Y., BACH, F., Y., L. et PONCE, J. (2010a). Learning mid-level features for recognition. *Dans les actes de IEEE Conference on Computer Vision and Pattern Recognition*. (Cité à la page 83.)

[Boureau *et coll.*, 2010b] BOUREAU, Y., PONCE, J. et Y., L. (2010b). A theoretical analysis of feature pooling in visual recognition. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 35 et 83.)

[Boyd et Vandenberghe, 2004] BOYD, S. et VANDENBERGHE, L. (2004). *Convex optimization*. Cambridge University Press. (Cité aux pages 7 et 68.)

[Bruna et Mallat, 2011] BRUNA, J. et MALLAT, S. (2011). Classification with scattering operators. *Dans les actes de IEEE Conference on Computer Vision and Pattern Recognition*. (Cité à la page 51.)

[Bruna et Mallat, 2013] BRUNA, J. et MALLAT, S. (2013). Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886. (Cité aux pages 50, 51 et 94.)

[Buckheit et Donoho, 1995] BUCKHEIT, J. et DONOHO, D. (1995). WaveLab and reproducible research. *Dans les actes de Wavelets and Statistics*, numéro 103 de Lecture Notes in Statistics, pages 55–81. Springer New York. (Cité à la page 69.)

[Bui *et coll.*, 2002] BUI, H., VENKATESH, S. et WEST, G. (2002). Policy recognition in the abstract hidden markov model. *Journal of Artificial Intelligence Research*, 17(1):451–499. arXiv : 1106.0672. (Cité à la page 55.)

[Bunescu et Mooney, 2007] BUNESCU, R. et MOONEY, R. (2007). Multiple instance learning for sparse positive bags. *Dans les actes de International Conference on Machine Learning*. (Cité à la page 29.)

[Cao *et coll.*, 2013] CAO, Y., BARETT, D., BARBU, A., NARAYANASWAMY, S., YU, H., MICHAUX, A., LIN, Y., DICKINSON, S., SISKIND, J. et WANG, S. (2013). Recognize human

activities from partially observed videos. *Dans les actes de Conference on Computer Vision and Pattern Recognition*. (Cité à la page 55.)

[Cauwenberghs et Poggio, 2001] CAUWENBERGHS, G. et POGGIO, T. (2001). Incremental and decremental support vector machine learning. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 106 et 120.)

[Chachada et Kuo, 2013] CHACHADA, S. et KUO, C.-C. J. (2013). Environmental sound recognition : A survey. *Dans les actes de Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 1–9. IEEE. (Cité aux pages 34 et 35.)

[Chai *et coll.*, 2014] CHAI, J., DING, X., CHEN, H. et LI, T. (2014). Multiple-instance discriminant analysis. *Pattern Recognition*, 47:2517–2531. (Cité à la page 27.)

[Chang et Lin, 2011] CHANG, C.-C. et LIN, C.-J. (2011). LIBSVM : a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27. (Cité à la page 63.)

[Chapelle, 2007] CHAPELLE, O. (2007). Training a support vector machine in the primal. *Neural Computation*, 19:1155–1178. (Cité à la page 12.)

[Chapelle *et coll.*, 2002] CHAPELLE, O., VAPNIK, V., BOUSQUET, O. et MUKHERJEE, S. (2002). Choosing multiple parameters for support vector machines. *Machine Learning*, 46:131–159. (Cité aux pages 17, 18, 19, 23, 85 et 138.)

[Chen *et coll.*, 1998] CHEN, S., DONOHO, D. et SAUNDERS, M. (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61. (Cité à la page 52.)

[Chen *et coll.*, 2006] CHEN, Y., BI, J. et WANG, J. (2006). Miles : Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:1931–1947. (Cité aux pages 30, 103, 115, 120 et 122.)

[Chen et Wang, 2004] CHEN, Y. et WANG, J. (2004). Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, 5:913–939. (Cité à la page 29.)

[Coates *et coll.*, 2010] COATES, A., LEE, H. et NG, A. (2010). An analysis of single-layer networks in unsupervised feature learning. *Dans les actes de NIPS Workshop on Deep Learning and Unsupervised Feature Learning*. (Cité à la page 103.)

[Cohn *et coll.*, 2009] COHN, J., KRUEZ, T., MATTHEWS, I., YANG, Y., NGUYEN, M. H., PADILLA, M., ZHOU, F. et De la TORRE, F. (2009). Detecting depression from facial actions and vocal prosody. *Dans les actes de 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009*, pages 1–7. (Cité à la page 57.)

[Coifman et Wickerhauser, 1992] COIFMAN, R. et WICKERHAUSER, M. (1992). Entropy-based algorithms for best basis selection. *IEEE Transactions on Information Theory*, 38:713–718. (Cité à la page 49.)

[Collobert et Bengio, 2004] COLLOBERT, R. et BENGIO, S. (2004). A gentle hessian for efficient gradient descent. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04)*. (Cité à la page 46.)

[Cortes *et coll.*, 2013] CORTES, C., KLOFT, M. et MOHRI, M. (2013). Learning kernels using local rademacher complexity. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 23.)

[Cortes *et coll.*, 2009] CORTES, C., MOHRI, M. et ROSTAMIZADEH, A. (2009). Learning non-linear combinations of kernels. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 24.)

[Cortes *et coll.*, 2010a] CORTES, C., MOHRI, M. et ROSTAMIZADEH, A. (2010a). Generalization bounds for learning kernels. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 20 et 21.)

[Cortes *et coll.*, 2010b] CORTES, C., MOHRI, M. et ROSTAMIZADEH, A. (2010b). Two-stage learning kernel algorithms. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 17 et 20.)

[Cortes *et coll.*, 2012] CORTES, C., MOHRI, M. et ROSTAMIZADEH, A. (2012). Algorithms for learning kernels based on centered alignment. *Journal of Machine Learning Research*, 13:795–828. (Cité aux pages 17 et 85.)

[Crammer *et coll.*, 2003] CRAMMER, K., KESHET, J. et SINGER, Y. (2003). Kernel design using boosting. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 21.)

[Cristianini *et coll.*, 1999] CRISTIANINI, N., CAMPBELL, C. et SHAWE-TAYLOR, J. (1999). Dynamically adapting kernels in support vector machines. *Dans les actes de* KEARNS, M., SOLLA, S. et COHN, D., éditeurs : *Advances in Neural Information Processing Systems*. (Cité à la page 19.)

[Cristianini *et coll.*, 2002] CRISTIANINI, N., SHAWE-TAYLOR, J., ELISSEEFF, A. et KANDOLA, J. (2002). On kernel-target alignment. *Dans les actes de Advances in Neural Information Processing Systems*. MIT Press. (Cité aux pages 17, 21 et 39.)

[Davis et Tyagi, 2006] DAVIS, J. et TYAGI, A. (2006). Minimal-latency human action recognition using reliable-inference. *Image and Vision Computing*, 24:455–472. (Cité à la page 55.)

[Davy et Doncarli, 1998] DAVY, M. et DONCARLI, C. (1998). Optimal kernels of time-frequency representations for signal classification. *Dans les actes de IEEE International Symposium on Time-Frequency and Time-Scale Analysis*. (Cité aux pages 38, 39 et 140.)

[Davy *et coll.*, 2001] DAVY, M., DONCARLI, C. et BOUDREAUX-BARTELS, G. (2001). Improved optimization of time-frequency-based signal classifiers. *IEEE Signal Processing Letters*, 8:52–57. (Cité aux pages 38, 39 et 140.)

[Davy *et coll.*, 2002] DAVY, M., GRETTON, A., DOUCET, A. et RAYNER, P. J. W. (2002). Optimized support vector machines for nonstationary signal classification. *IEEE Signal Processing Letters*, 9:442–445. (Cité aux pages 38, 39 et 140.)

[Dietterich *et coll.*, 1997] DIETTERICH, T., LATHROP, R. et LOZANO-PÉREZ, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1-2):31–71. (Cité à la page 27.)

[Dong *et coll.*, 2008] DONG, Y., XIA, Z. et XIA, Z. (2008). A two-level approach to choose the cost parameter in support vector machines. *Expert Systems with Applications*, 34:1366 – 1370. (Cité à la page 16.)

[Dong *et coll.*, 2007] DONG, Y.-L., XIA, Z.-Q. et WANG, M.-Z. (2007). An mpec model for selecting optimal parameter in support vector machines. *Dans les actes de The First International Symposium on Optimization and Systems Biology (OSB'07)*. (Cité à la page 16.)

[Droppo et Atlas, 1998] DROPPO, J. et ATLAS, L. (1998). Applications of classifier-optimal time-frequency distributions to speech analysis. *Dans les actes de IEEE International Symposium on Time-Frequency and Time-Scale Analysis*. (Cité aux pages 40 et 140.)

[Duan *et coll.*, 2003] DUAN, K., KEERTHI, S. S. et POO, A. N. (2003). Evaluation of simple performance measures for tuning svm hyperparameters. *Neurocomputing*, 51:41–59. (Cité à la page 18.)

[Ellis *et coll.*, 2013] ELLIS, C., MASOOD, S., TAPPEN, M., LAVIOLA, J. J. et SUKTHANKAR, R. (2013). Exploring the trade-off between accuracy and observational latency in action recognition. *International Journal of Computer Vision*, 101:420–436. (Cité aux pages 56, 104 et 116.)

[Farina *et coll.*, 2007] FARINA, D., do NASCIMENTO, O., LUCAS, M. et DONCARLI, C. (2007). Optimization of wavelets for classification of movement-related cortical potentials generated by variation of force-related parameters. *Journal of Neuroscience Methods*, 162:357–363. (Cité aux pages 49, 50 et 140.)

[Flamary *et coll.*, 2012] FLAMARY, R., TUIA, D., LABBÉ, B., CAMPS-VALLS, G. et RAKOTOMAMONJY, A. (2012). Large margin filtering. *IEEE Transactions on Signal Processing*, 60:648–659. (Cité aux pages 43, 63, 68 et 140.)

[Flandrin, 1998] FLANDRIN, P. (1998). *Temps-Fréquence. 2ème édition revue et corrigée*. Hermès. (Cité aux pages 37 et 38.)

[Fukushima, 1980] FUKUSHIMA, K. (1980). Neocognitron : A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202. (Cité aux pages 35, 44 et 140.)

[Gai *et coll.*, 2010] GAI, K., CHEN, G. et ZHANG, C.-S. (2010). Learning kernels with radiuses of minimum enclosing balls. *Dans les actes de* LAFFERTY, J., WILLIAMS, C., SHAWE-TAYLOR, J., ZEMEL, R. et CULOTTA, A., éditeurs : *Advances in Neural Information Processing Systems*. (Cité aux pages 24, 137 et 138.)

[Gauthier *et coll.*, 2009] GAUTHIER, J., DUVAL, L. et PESQUET, J.-C. (2009). Optimization of synthesis oversampled complex filter banks. *IEEE Transactions on Signal Processing*, 57:3827–3843. (Cité à la page 41.)

[Gehler et Nowozin, 2008a] GEHLER, P. et NOWOZIN, S. (2008a). Infinite kernel learning. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 25, 30, 77, 82, 85, 94 et 137.)

[Gehler et Nowozin, 2008b] GEHLER, P. V. et NOWOZIN, S. (2008b). Infinite kernel learning. Rapport technique, Max Planck Institute for Biological Cybernetics. (Cité aux pages 75, 76, 77 et 106.)

[Giannoulis *et coll.*, 2013] GIANNOULIS, D., BENETOS, E., STOWELL, D., ROSSIGNOL, M., LAGRANGE, M. et PLUMBLEY, M. (2013). Detection and classification of acoustic scenes and events. Rapport technique, Queen Mary University of London. An IEEE AASP Challenge. (Cité à la page 91.)

[Gillespie et Atlas, 1998] GILLESPIE, B. et ATLAS, L. (1998). Data-driven optimization of time and frequency resolution for radar transmitter identification. *Dans les actes de SPIE - The International Society for Optical Engineering*. (Cité à la page 40.)

[Gillespie et Atlas, 1999] GILLESPIE, B. et ATLAS, L. (1999). Optimization of time and frequency resolution for radar transmitter identification. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing*. (Cité à la page 40.)

[Gillespie et Atlas, 2001] GILLESPIE, B. et ATLAS, L. (2001). Optimizing time-frequency kernels for classification. *IEEE Transactions on Signal Processing*, 49:485–496. (Cité à la page 40.)

[Gorelick *et coll.*, 2007a] GORELICK, L., BLANK, M., SHECHTMAN, E., IRANI, M. et BASRI, R. (2007a). Actions as space-time shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:2247–2253. (Cité à la page 120.)

[Gorelick *et coll.*, 2007b] GORELICK, L., BLANK, M., SHECHTMAN, E., IRANI, M. et BASRI, R. (2007b). Actions as space-time shapes code. http ://www.csd.uwo.ca/ ygorelic/STP.zip. (Cité à la page 120.)

[Grandvalet et Canu, 2003] GRANDVALET, Y. et CANU, S. (2003). Adaptive scaling for feature selection in SVMs. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 23.)

[Gärtner *et coll.*, 2002] GÄRTNER, T., FLACH, P., KOWALCZYK, A. et SMOLA, A. (2002). Multi-instance kernels. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 26 et 27.)

[Gönen et Alpaydin, 2008] GÖNEN, M. et ALPAYDIN, E. (2008). Localized multiple kernel learning. *Dans les actes de International Conference on Machine Learning*. (Cité à la page 24.)

[Gönen et Alpaydin, 2011] GÖNEN, M. et ALPAYDIN, E. (2011). Multiple kernel learning algorithms. *Journal of Machine Learning Research*, 12:2211–2268. (Cité à la page 20.)

[Hastie *et coll.*, 2004] HASTIE, T., ROSSET, S., TIBSHIRANI, R. et ZHU, J. (2004). The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5:1391–1415. (Cité à la page 17.)

[Hastie *et coll.*, 2008] HASTIE, T., TIBSHIRANI, R. et FRIEDMAN, J. (2008). *The elements of statistical learning : data mining, inference, and prediction*. Springer. (Cité aux pages 11 et 16.)

[Heitz, 1995] HEITZ, C. (1995). Optimum time-frequency representations for the classification and detection of signals. *Applied Signal Processing*, 3(2):124–143. (Cité aux pages 38 et 140.)

[Heitz, 1996] HEITZ, C. (1996). Classification of time series with optimized time-frequency representations. *Dans les actes de Data Analysis and Information Systems*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 41–51. Springer Berlin Heidelberg. (Cité à la page 38.)

[Hoai et De la Torre, 2012] HOAI, M. et De la TORRE, F. (2012). Max-margin early event detectors. *Dans les actes de IEEE Conference on Computer Vision and Pattern Recognition*. (Cité aux pages 57 et 116.)

[Hoai et De la Torre, 2014] HOAI, M. et De la TORRE, F. (2014). Max-margin early event detectors. *International Journal of Computer Vision*, 107:191–202. (Cité aux pages 54, 57, 58, 96, 101, 104, 116, 117, 121, 122, 125 et 128.)

[Honeine, 2007] HONEINE, P. (2007). *Méthodes à noyau pour l'analyse et la d écision en environnement non-stationnaire*. Thèse de doctorat, Université de Technologie de Troyes. (Cité aux pages 39 et 140.)

[Honeine et Richard, 2007] HONEINE, P. et RICHARD, C. (2007). Signal-dependent time-frequency representations for classification using a radially gaussian kernel and the alignment criterion. *Dans les actes de IEEE Workshop on Statistical Signal Processing*. (Cité à la page 39.)

[Honeine *et coll.*, 2007] HONEINE, P., RICHARD, C. et FLANDRIN, P. (2007). Time-frequency learning machines. *IEEE Transactions on Signal Processing*, 55:3930–3936. (Cité à la page 39.)

[Honeine *et coll.*, 2006] HONEINE, P., RICHARD, C., FLANDRIN, P. et POTHIN, J.-B. (2006). Optimal selection of time-frequency representations for signal classification : a kernel-target alignment approach. *Dans les actes de IEEE International Conference on Acoustics, Speech and Signal Processing*. (Cité aux pages 39 et 140.)

[Huang et LeCun, 2006] HUANG, F. et LECUN, Y. (2006). Large-scale learning with SVM and convolutional nets for generic object categorization. *Dans les actes de IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. (Cité à la page 46.)

[Huang et Aviyente, 2006] HUANG, K. et AVIYENTE, S. (2006). Sparse representation for signal classification. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 52 et 53.)

[Hubel et Wiesel, 1962] HUBEL, D. et WIESEL, T. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106–154. (Cité à la page 35.)

[Jain *et coll.*, 2012] JAIN, A., VISHWANATHAN, S. et VARMA, M. (2012). Spg-gmkl : Generalized multiple kernel learning with a million kernels. *Dans les actes de International Conference on Knowledge Discovery and Data Mining*. (Cité aux pages 22 et 24.)

[Jones *et coll.*, 2001] JONES, E., RUNKLE, P., DASGUPTA, N., COUCHMAN, L. et CARIN, L. (2001). Genetic algorithm wavelet design for signal classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:890–895. (Cité à la page 49.)

[Kakade *et coll.*, 2009] KAKADE, S., SRIDHARAN, K. et TEWARI, A. (2009). On the complexity of linear prediction : Risk bounds, margin bounds, and regularization. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 5, 6, 30, 96, 104, 111, 112, 113 et 114.)

[Kar et Jain, 2012] KAR, P. et JAIN, P. (2012). Supervised learning with similarity functions. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 103, 111, 128 et 131.)

[Karasuyama et Takeuchi, 2009] KARASUYAMA, M. et TAKEUCHI, I. (2009). Multiple incremental decremental learning of support vector machines. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 106.)

[Karasuyama et Takeuchi, 2010] KARASUYAMA, M. et TAKEUCHI, I. (2010). Multiple incremental decremental learning of support vector machines. *IEEE Transactions on Neural Networks*, 21:1048–1059. (Cité aux pages 12, 106 et 110.)

[Keeler *et coll.*, 1991] KEELER, J., RUMELHART, D. et LEOW, W.-K. (1991). Integrated segmentation and recognition of hand-printed numerals. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 26.)

[Keerthi *et coll.*, 2006] KEERTHI, S., SINDHWANI, V. et CHAPELLE, O. (2006). An efficient method for gradient-based adaptation of hyperparameters in SVM models. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 16 et 17.)

[Kimeldorf et Wahba, 1971] KIMELDORF, G. et WAHBA, G. (1971). Some results on tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33(1):82–95. (Cité à la page 10.)

[Kinnunen, 2002] KINNUNEN, T. (2002). Designing a speaker-discriminative adaptive filter bank for speaker recognition. *Dans les actes de Annual Conference of the International Speech Association*. (Cité à la page 43.)

[Kloft *et coll.*, 2011] KLOFT, M., BREFELD, U., SONNENBURG, S. et ZIEN, A. (2011). ℓ_p-norm Multiple Kernel Learning. *Journal of Machine Learning Research*, 12:943–997. (Cité à la page 22.)

[Kunapuli *et coll.*, 2008] KUNAPULI, G., BENNETT, K., HU, J. et PANG, J.-S. (2008). Classification model selection via bilevel programming. *Optimization Methods and Software*, 23:475–489. (Cité à la page 16.)

[Lanckriet *et coll.*, 2002] LANCKRIET, G., CRISTIANINI, N., BARTLETT, P., GHAOUI, L. E. et JORDAN, M. (2002). Learning the kernel matrix with semidefinite programming. *Dans les actes de International Conference of Machine Learning*. (Cité aux pages 21, 22, 24 et 63.)

[Lanckriet *et coll.*, 2004] LANCKRIET, G., CRISTIANINI, N., BARTLETT, P., GHAOUI, L. E. et JORDAN, M. (2004). Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72. (Cité aux pages 17, 20, 21, 22, 24 et 135.)

[Lang *et coll.*, 1990] LANG, K., WAIBEL, A. et HINTON, G. (1990). A time-delay neural network architecture for isolated word recognition. *Neural Networks*, 3(1):23–43. (Cité à la page 44.)

[Laskov *et coll.*, 2006] LASKOV, P., GEHL, C., KRÜGER, S. et MÜLLER, K.-R. (2006). Incremental support vector learning : Analysis, implementation and applications. *Journal of Machine Learning Research*, 7:1909–1936. (Cité aux pages 106 et 120.)

[LeCun, 1989] LECUN, Y. (1989). Generalization and network design strategies. Rapport technique CRG-TR-89-4, Department of Computer Science, University of Toronto. (Cité aux pages 44 et 140.)

[LeCun *et coll.*, 1989] LECUN, Y., BOSER, B., DENKER, J., HENDERSON, D., HOWARD, R., HUBBARD, W. et JACKEL, L. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551. (Cité aux pages 35 et 44.)

[LeCun *et coll.*, 1998] LECUN, Y., BOTTOU, L., BENGIO, Y. et HAFFNER, P. (1998). Gradient-based learning applied to document recognition. *IEEE Proceedings*, 86(11):2278–2324. (Cité aux pages 36, 44, 45, 85 et 87.)

[Lee et Pottier, 2009] LEE, J.-S. et POTTIER, E. (2009). *Polarimetric Radar Imaging : From Basics to Applications*. CRC Press, Boca Raton. (Cité à la page 35.)

[Lemaréchal et Sagastizábal, 1997] LEMARÉCHAL, C. et SAGASTIZÁBAL, C. (1997). Practical aspects of the moreau–yosida regularization : Theoretical preliminaries. *SIAM Journal on Optimization*, 7(2):367–385. (Cité aux pages 21 et 135.)

[Lewicki et Sejnowski, 2000] LEWICKI, M. et SEJNOWSKI, T. (2000). Learning overcomplete representations. *Neural Comput.*, 12(2):337–365. (Cité à la page 52.)

[Li et Fu, 2012] LI, K. et FU, Y. (2012). Arma-hmm : A new approach for early recognition of human activity. *Dans les actes de International Conference on Pattern Recognition*. (Cité à la page 55.)

[Li et Fu, 2014] LI, K. et FU, Y. (2014). Prediction of human activity by discovering temporal sequence patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1. (Cité à la page 55.)

[Liao *et coll.*, 2005] LIAO, L., FOX, D. et KAUTZ, H. (2005). Location-based activity recognition using relational markov networks. *Dans les actes de International Joint Conference on Artificial Intelligence*. (Cité à la page 55.)

[Long et Leow, 2002] LONG, H. et LEOW, W. (2002). A hybrid model for invariant and perceptual texture mapping. *Dans les actes de International Conference on Pattern Recognition*. (Cité à la page 46.)

[Luenberger, 1984] LUENBERGER, D. (1984). *Linear and nonlinear programming*. Addison-Wesley. (Cité aux pages 77 et 78.)

[Luo *et coll.*, 2010] LUO, Z.-Q., MA, W.-K., SO, A.-C., YE, Y. et ZHANG, S. (2010). Semidefinite relaxation of quadratic optimization problems. *IEEE Signal Processing Magazine*, 27:20–34. (Cité aux pages 66 et 67.)

[Mairal *et coll.*, 2012] MAIRAL, J., BACH, F. et PONCE, J. (2012). Task-driven dictionary learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(4):791–804. (Cité à la page 54.)

[Mairal *et coll.*, 2009] MAIRAL, J., BACH, F., PONCE, J., SAPIRO, G. et ZISSERMAN (2009). Supervised dictionary learning. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 53 et 140.)

[Mairal *et coll.*, 2008] MAIRAL, J., BACH, F., PONCE, J., SAPIRO, G., ZISSERMAN, A., COG, T., Équipes-projets WILLOW et SUPÉRIEURE, E. N. (2008). Supervised dictionary learning. (Cité à la page 53.)

[Mallat, 1989] MALLAT, S. (1989). A theory for multiresolution signal decomposition : the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693. (Cité à la page 46.)

[Mallat, 1999] MALLAT, S. (1999). *A wavelet tour of signal processing*. Elsevier/Academic Press. (Cité aux pages 47, 48, 147 et 148.)

[Mallat, 2012] MALLAT, S. (2012). Group invariant scattering. *Communications in Pure and Applied Mathematics*, 65:1331–1398. (Cité aux pages 36 et 50.)

[Mallat et Zhang, 1993] MALLAT, S. et ZHANG, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41:3397–3415. (Cité aux pages 51 et 52.)

[Mandel et Ellis, 2008] MANDEL, M. I. et ELLIS, D. P. W. (2008). Multiple-instance learning for music information retrieval. *Dans les actes de International Society for Music Information Retrieval Conference*. (Cité aux pages 27 et 115.)

[Mangasarian et Musicant, 2001] MANGASARIAN, O. et MUSICANT, D. (2001). Lagrangian support vector machines. *Journal of Machine Learning Research*, 1:161–177. (Cité à la page 105.)

[Maron et Lozano-Pérez, 1998] MARON, O. et LOZANO-PÉREZ, T. (1998). A framework for multiple-instance learning. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 27, 29, 30 et 115.)

[McLaughlin, 1997] MCLAUGHLIN, J. (1997). *Applications of operator theory to time-frequency analysis and classification*. Thèse de doctorat, University of Washington. (Cité à la page 40.)

[McLaughlin *et coll.*, 1997] MCLAUGHLIN, J., DROPPO, J. et ATLAS, L. (1997). Class-dependent, discrete time-frequency distributions via operator theory. *Dans les actes de IEEE International Conference on Acoustics, Speech, and Signal Processing*. (Cité à la page 40.)

[Micchelli et Pontil, 2005] MICCHELLI, C. et PONTIL, M. (2005). Learning the kernel function via regularization. *Journal of Machine Learning Research*, 6:1099–1125. (Cité aux pages 24 et 25.)

[Mitrović *et coll.*, 2010] MITROVIĆ, D., ZEPPELZAUER, M. et BREITENEDER, C. (2010). Features for content-based audio retrieval. *Dans les actes de Advances in Computers*, volume Volume 78 de *Advances in Computers : Improving the Web*, pages 71–150. Elsevier. (Cité aux pages 33, 34 et 36.)

[Nagi *et coll.*, 2012] NAGI, J., DI CARO, G., GIUSTI, A., NAGI, F. et GAMBARDELLA, L. (2012). Convolutional neural support vector machines : Hybrid visual pattern classifiers for multi-robot systems. *Dans les actes de International Conference on Machine Learning and Applications (ICMLA)*. (Cité à la page 46.)

[Narayanan *et coll.*, 1996] NARAYANAN, S., MCLAUGHLIN, J. et DROPPO, J. (1996). Operator theory approach to discrete time-frequency representations. *Dans les actes de IEEE International Symposium on Time-Frequency and Time-Scale Analysis*. (Cité à la page 40.)

[Neill *et coll.*, 2005] NEILL, D., MOORE, A. et COOPER, G. (2005). A bayesian spatial scan statistic. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 54 et 57.)

[Neumann *et coll.*, 2005] NEUMANN, J., SCHNÖRR, C. et STEIDL, G. (2005). Efficient wavelet adaptation for hybrid wavelet-large margin classifiers. *Pattern Recognition*, 38:1815–1830. (Cité aux pages 17, 18, 50, 133 et 140.)

[Nocedal et Wright, 2000] NOCEDAL, J. et WRIGHT, S. (2000). *Numerical optimization*. Springer. (Cité aux pages 78 et 105.)

[Nowozin et Shotton, 2012] NOWOZIN, S. et SHOTTON, J. (2012). Action points : A representation for low-latency online human action recognition. Rapport technique, Microsoft Research. (Cité à la page 54.)

[Olshausen et Field, 1997] OLSHAUSEN, B. et FIELD, D. (1997). Sparse coding with an overcomplete basis set : a strategy employed by v1 ? *Vision Research*, 37(23):3311–3325. (Cité à la page 52.)

[Olshausen et Field, 1996] OLSHAUSEN, B. A. et FIELD, D. J. (1996). Natural image statistics and efficient coding. *Network (Bristol, England)*, 7(2):333–339. (Cité à la page 52.)

[Ong *et coll.*, 2003] ONG, C., SMOLA, A. et WILLIAMSON, R. (2003). Hyperkernels. *Dans les actes de Advances in Neural Information Processing Systems*. MIT Press. (Cité à la page 24.)

[Ong *et coll.*, 2005] ONG, C., SMOLA, A. et WILLIAMSON, R. (2005). Learning the kernel with hyperkernels. *Journal of Machine Learning Research*, 6:1043–1071. (Cité à la page 24.)

[Parrish *et coll.*, 2013] PARRISH, N., ANDERSON, H., GUPTA, M. et HSIAO, D. (2013). Classifying with confidence from incomplete information. *Journal of Machine Learning Research*, 14:3561–3589. (Cité aux pages 56 et 101.)

[Pati *et coll.*, 1993] PATI, Y., REZAIIFAR, R. et KRISHNAPRASAD, P. (1993). Orthogonal matching pursuit : recursive function approximation with applications to wavelet decomposition. *Dans les actes de IEEE Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, 1993*. (Cité à la page 52.)

[Pavlidis *et coll.*, 2001] PAVLIDIS, P., WESTON, J., CAI, J. et GRUNDY, W. (2001). Gene functional classification from heterogeneous data. *Dans les actes de International Conference on Computational Biology*. (Cité à la page 21.)

[Peeman *et coll.*, 2011] PEEMAN, M., MESMAN, B. et CORPORAAL, H. (2011). Speed sign detection and recognition by convolutional neural networks. *Dans les actes de International Automotive Congress*. (Cité à la page 45.)

[Pekalska et Duin, 2008] PEKALSKA, E. et DUIN, R. (2008). Beyond traditional kernels : Classification in two dissimilarity-based representation spaces. *IEEE Transactions on Systems, Man, and Cybernetics, Part C : Applications and Reviews*, 38:729–744. (Cité aux pages 29, 96 et 99.)

[Ping *et coll.*, 2010] PING, W., XU, Y., REN, K., CHI, C. et SHEN, F. (2010). Non-i.i.d. multi-instance dimensionality reduction by learning a maximum bag margin subspace. *Dans les actes de Conference on Artificial Intelligence*. (Cité à la page 27.)

[Platt, 1999] PLATT, J. (1999). *Fast training of support vector machines using sequential minimal optimization*. Advances in Kernel Methods. MIT Press, Cambridge, MA, USA. (Cité aux pages 12, 21 et 63.)

[Pothin et Richard, 2005] POTHIN, J.-B. et RICHARD, C. (2005). Kernel machines : une nouvelle méthode pour l'optimisation de l'alignement des noyaux et l'amélioration des performances. *Dans les actes de XXème Colloque GRETSI sur le Traitement du Signal et des Images*. (Cité à la page 39.)

[Qi *et coll.*, 2013] QI, J., WANG, D., JIANG, Y. et LIU, R. (2013). Auditory features based on gammatone filters for robust speech recognition. *Dans les actes de IEEE International Symposium on Circuits and Systems*. (Cité à la page 41.)

[Rakotomamonjy *et coll.*, 2008] RAKOTOMAMONJY, A., BACH, F., CANU, S. et GRANDVALET, Y. (2008). SimpleMKL. *Journal of Machine Learning Research*, 9:2491–2521. (Cité aux pages 17, 22, 24, 25, 63, 68, 136 et 137.)

[Rakotomamonjy *et coll.*, 2013] RAKOTOMAMONJY, A., FLAMARY, R. et YGER, F. (2013). Learning with infinitely many feature. *Machine Learning*, 91:43–66. (Cité aux pages 81 et 82.)

[Rakotomamonjy et Gasso, 2014] RAKOTOMAMONJY, A. et GASSO, G. (2014). Histogram of gradients of time-frequency representations for audio scene detection. Rapport technique, University of Rouen. HAL-00951990. (Cité aux pages 36 et 120.)

[Ralaivola et d'Alché Buc, 2001] RALAIVOLA, L. et d'Alché BUC, F. (2001). Incremental support vector machine learning : A local approach. *Dans les actes de International Conference on Artificial Neural Networks*. (Cité à la page 106.)

[Ranzato *et coll.*, 2008] RANZATO, M., BOUREAU, Y.-L. et LECUN, Y. (2008). Sparse feature learning for deep belief networks. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 35.)

[Ravishankar *et coll.*, 2009] RAVISHANKAR, S., JAIN, A. et MITTAL, A. (2009). Automated feature extraction for early detection of diabetic retinopathy in fundus images. *Dans les actes de IEEE Conference on Computer Vision and Pattern Recognition*. (Cité à la page 57.)

[Ray et Chan, 2001] RAY, S. et CHAN, A. (2001). Automatic feature extraction from wavelet coefficients using genetic algorithms. *Dans les actes de IEEE Proceedings of the Neural Networks for Signal Processing XI, 2001*. (Cité à la page 49.)

[Richard *et coll.*, 2013] RICHARD, G., SUNDARAM, S. et NARAYANAN, S. (2013). An overview on perceptually motivated audio indexing and classification. *Proceedings of the IEEE*, 101(9):1939–1954. (Cité à la page xiv.)

[Rodriguez et Sapiro, 2008] RODRIGUEZ, F. et SAPIRO, G. (2008). Sparse representation for image classification : Learning discriminative and reconstructive non-parametric dictionaries. Rapport technique Institute for Mathematics and its Applications Preprint 2213, University of Minnesota. (Cité aux pages 53 et 140.)

[Rodriguez et Alonso, 2002] RODRIGUEZ, J. et ALONSO, C. (2002). Boosting interval-based literals : Variable length and early classification. *Dans les actes de Workshop on Knowledge Discovery from (Spatio-) Temporal Data*. (Cité à la page 55.)

[Rumelhart *et coll.*, 1986] RUMELHART, D., HINTON, G. et WILLIAMS, R. (1986). Parallel distributed processing : Explorations in the microstructure of cognition, vol. 1. pages 318–362. MIT Press, Cambridge, MA, USA. (Cité à la page 45.)

[Ryoo, 2011] RYOO, M. (2011). Human activity prediction : Early recognition of ongoing activities from streaming videos. *Dans les actes de IEEE International Conference on Computer Vision*, pages 1036–1043. (Cité à la page 55.)

[Saito et Coifman, 1995] SAITO, N. et COIFMAN, R. (1995). Local discriminant bases and their applications. *Journal of Mathematical Imaging and Vision*, 5:337–358. (Cité aux pages 48, 49 et 140.)

[Saito et Coifman, 1997] SAITO, N. et COIFMAN, R. (1997). Improved discriminant bases using empirical probability density estimation. *Computing Section of American Statistical Association*. (Cité à la page 49.)

[Sangnier *et coll.*, 2014] SANGNIER, M., GAUTHIER, J. et RAKOTOMAMONJY, A. (2014). Kernel learning as minimization of the single validation estimate. *Dans les actes de IEEE Workshop on Machine Learning for Signal Processing*. (Cité aux pages 94 et 130.)

[Schafer, 1977] SCHAFER, R. (1977). *The Tuning of the World*. Knopf, New York, NY, USA. (Cité à la page 91.)

[Schölkopf *et coll.*, 2001] SCHÖLKOPF, B., HERBRICH, R. et SMOLA, A. (2001). A generalized representer theorem. *Dans les actes de Computational Learning Theory*. (Cité à la page 10.)

[Seeger, 2008] SEEGER, M. (2008). Cross-validation optimization for large scale structured classification kernel methods. *Journal of Machine Learning Research*, 9:1147–1178. (Cité à la page 16.)

[Shawe-Taylor et Cristianini, 2004] SHAWE-TAYLOR, J. et CRISTIANINI, N. (2004). *Kernel methods for pattern analysis*. Cambridge University Press. (Cité aux pages 8 et 19.)

[Shi et Manduchi, 2004] SHI, X. et MANDUCHI, R. (2004). Invariant operators, small samples, and the bias-variance dilemma. *Dans les actes de IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. (Cité à la page 35.)

[Skretting et Husøy, 2006] SKRETTING, K. et HUSØY, J. (2006). Texture classification using sparse frame-based representations. *EURASIP Journal on Advances in Signal Processing*, 2006(1):052561. (Cité à la page 53.)

[Slaney, 1993] SLANEY, M. (1993). An efficient implementation of the patterson-holdsworth auditory filter bank. Rapport technique 35, Apple Computer, Inc. (Cité à la page 41.)

[Sonnenburg *et coll.*, 2006a] SONNENBURG, S., RÄTSCH, G. et SCHÄFER, C. (2006a). A general and efficient multiple kernel learning algorithm. *Dans les actes de Advances in Neural Information Processing Systems*. MIT Press. (Cité aux pages 21, 22, 23, 25, 135 et 136.)

[Sonnenburg *et coll.*, 2006b] SONNENBURG, S., RÄTSCH, G., SCHÄFER, C. et SCHÖLKOPF, B. (2006b). Large scale multiple kernel learning. *Journal of Machine Learning Research*, 7:1531–1565. (Cité aux pages 21, 22, 26, 135 et 136.)

[Sriperumbudur et Lanckriet, 2009] SRIPERUMBUDUR, B. et LANCKRIET, G. (2009). On the convergence of the concave-convex procedure. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 29.)

[Stevens *et coll.*, 1937] STEVENS, S., VOLKMANN, J. et NEWMAN, E. (1937). A scale for the measurement of the psychological magnitude pitch. *The Journal of the Acoustical Society of America*, 8(3):185–190. (Cité aux pages 41 et 143.)

[Strang et Nguyen, 1996] STRANG, G. et NGUYEN, T. (1996). *Wavelets and filter banks*. Wellesley-Cambridge Press. (Cité aux pages 40, 41, 48 et 147.)

[Strauss et Steidl, 2002] STRAUSS, D. J. et STEIDL, G. (2002). Hybrid wavelet-support vector classification of waveforms. *Journal of Computational and Applied Mathematics*, 148:375–400. (Cité aux pages 49, 50 et 140.)

[Strauss *et coll.*, 2003] STRAUSS, D. J., STEIDL, G. et DELB, W. (2003). Feature extraction by shape-adapted local discriminant bases. *Signal Processing*, 83:359–376. (Cité aux pages 49 et 140.)

[Sturm, 1999] STURM, J. (1999). Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12:625–653. (Cité aux pages 67, 70 et 71.)

[Su et Sato, 2013] SU, L. et SATO, Y. (2013). Early facial expression recognition using early rankboost. *Dans les actes de IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*. (Cité à la page 57.)

[Sugiyama *et coll.*, 1991] SUGIYAMA, M., SAWAI, H. et WAIBEL, A. (1991). Review of TDNN (time delay neural network) architectures for speech recognition. *Dans les actes de IEEE International Sympoisum on Circuits and Systems*. (Cité à la page 44.)

[Suk et Lee, 2013] SUK, H.-I. et LEE, S.-W. (2013). A novel bayesian framework for discriminative feature extraction in brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):286–299. (Cité aux pages 43 et 140.)

[Suriani *et coll.*, 2013] SURIANI, N., HUSSAIN, A. et ZULKIFLEY, M. (2013). Sudden event recognition : A survey. *Sensors*, 13:9966–9998. (Cité aux pages 54 et 57.)

[Szafranski, 2008] SZAFRANSKI, M. (2008). *Pénalités hiérarchiques pour l'intégration de connaissances dans les modèles statistiques*. Thèse de doctorat, Université de Technologie de Compiègne. (Cité à la page 11.)

[Szafranski *et coll.*, 2008] SZAFRANSKI, M., GRANDVALET, Y. et RAKOTOMAMONJY, A. (2008). Composite kernel learning. *Dans les actes de International Conference of Machine Learning*. (Cité à la page 24.)

[Szafranski *et coll.*, 2010] SZAFRANSKI, M., GRANDVALET, Y. et RAKOTOMAMONJY, A. (2010). Composite kernel learning. *Machine Learning*, 79:73–103. (Cité aux pages 77 et 78.)

[Tang, 2013] TANG, Y. (2013). Deep learning using linear support vector machines. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 46 et 140.)

[Thomas *et coll.*, 2009] THOMAS, K., GUAN, C., LAU, C., VINOD, A. et ANG, K. (2009). A new discriminative common spatial pattern method for motor imagery brain #x2013 ;computer interfaces. *IEEE Transactions on Biomedical Engineering*, 56(11):2730–2733. (Cité à la page 43.)

[Tibshirani, 1996] TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288. (Cité à la page 52.)

[Tikhonov et Arsenin, 1977] TIKHONOV, A. et ARSENIN, V. (1977). *Solutions of ill-posed problems*. Winston, Washington, DC. (Cité aux pages 10 et 111.)

[Tsang et Kwok, 2006] TSANG, I. et KWOK, J.-Y. (2006). Efficient hyperkernel learning using second-order cone programming. *IEEE Transactions on Neural Networks*, 17(1):48–58. (Cité à la page 24.)

[Tsochantaridis *et coll.*, 2005] TSOCHANTARIDIS, I., JOACHIMS, T., HOFMANN, T. et ALTUN, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484. (Cité à la page 58.)

[Vaidyanathan, 1993] VAIDYANATHAN, P. (1993). *Multirate systems and filter banks*. Prentice Hall. (Cité à la page 41.)

[Vapnik, 1995] VAPNIK, V. (1995). *The Nature of Statistical Learning Theory*. Springer New York. (Cité à la page 1.)

[Vapnik, 1998] VAPNIK, V. (1998). *Statistical learning theory*. Wiley. (Cité aux pages 4, 5, 15 et 17.)

[Varma et Babu, 2009] VARMA, M. et BABU, B. (2009). More generality in efficient multiple kernel learning. *Dans les actes de International Conference on Machine Learning*. (Cité aux pages 22, 24, 30, 63, 77, 78, 86 et 91.)

[Varma et Ray, 2007] VARMA, M. et RAY, D. (2007). Learning the discriminative power-invariance trade-off. *Dans les actes de International Conference on Computer Vision*. (Cité à la page 35.)

[Vautrin *et coll.*, 2009] VAUTRIN, D., ARTUSI, X., LUCAS, M.-F. et FARINA, D. (2009). A novel criterion of wavelet packet best basis selection for signal classification with application to brain-computer interfaces. *IEEE Transactions on Biomedical Engineering*, 56:2734–2738. (Cité à la page 49.)

[Vignolo *et coll.*, 2011a] VIGNOLO, L. D., RUFINER, H. L., MILONE, D. H. et GODDARD, J. (2011a). Evolutionary cepstral coefficients. *Applied Soft Computing*, 11:3419–3428. (Cité aux pages 43, 140 et 143.)

[Vignolo *et coll.*, 2011b] VIGNOLO, L. D., RUFINER, H. L., MILONE, D. H. et GODDARD, J. (2011b). Evolutionary splines for cepstral filterbank optimization in phoneme classification. *EURASIP Journal on Advances in Signal Processing*, 2011:1–14. (Cité à la page 43.)

[Viola *et coll.*, 2006] VIOLA, P., PLATT, J. et ZHANG, C. (2006). Multiple instance boosting for object detection. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 27.)

[Vishwanathan *et coll.*, 2010] VISHWANATHAN, S., SUN, Z., THEERA-AMPORNPUNT, N. et VARMA, M. (2010). Multiple kernel learning and the SMO algorithm. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 21 et 22.)

[Wang et Zucker, 2000] WANG, J. et ZUCKER, J.-D. (2000). Solving the multiple-instance problem : A lazy learning approach. *Dans les actes de International Conference on Machine Learning*. (Cité à la page 26.)

[Weston *et coll.*, 2001] WESTON, J., MUKHERJEE, S., CHAPELLE, O., PONTIL, M., POGGIO, T. et VAPNIK, V. (2001). Feature selection for SVMs. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité aux pages 19 et 23.)

[Xing *et coll.*, 2009] XING, Z., PEI, J. et YU, P. (2009). Early prediction on time series : A nearest neighbor approach. *Dans les actes de International Joint Conferences on Artificial Intelligence*. (Cité à la page 55.)

[Xing *et coll.*, 2012] XING, Z., PEI, J. et YU, P. (2012). Early classification on time series. *Knowledge and Information Systems*, 31:105–127. (Cité à la page 56.)

[Yang *et coll.*, 2010] YANG, M., ZHANG, L., YANG, J. et ZHANG, D. (2010). Metaface learning for sparse representation based face recognition. *Dans les actes de IEEE International Conference on Image Processing*. (Cité à la page 53.)

[Yger et Rakotomamonjy, 2011] YGER, F. et RAKOTOMAMONJY, A. (2011). Wavelet kernel learning. *Pattern Recognition*, 44:2614–2629. (Cité aux pages 22, 50, 78, 79, 85, 87, 89, 90 et 140.)

[Yuille et Rangarajan, 2002] YUILLE, A. et RANGARAJAN, A. (2002). The concave-convex procedure (CCCP). *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 29.)

[Zhang *et coll.*, 2011] ZHANG, H., CHIN, Z., ANG, K., GUAN, C. et WANG, C. (2011). Optimum spatio-spectral filtering network for brain-computer interface. *IEEE Transactions on Neural Networks*, 22(1):52–63. (Cité à la page 43.)

[Zhong et Ghosh, 2000] ZHONG, S. et GHOSH, J. (2000). Decision boundary focused neural network classifier. *Dans les actes de Intelligent Engineering Systems Through Artificial Neural Networks*. (Cité aux pages 46 et 140.)

[Zhu *et coll.*, 2004] ZHU, J., ROSSET, S., HASTIE, T. et TIBSHIRANI, R. (2004). 1-norm Support Vector Machines. *Dans les actes de Advances in Neural Information Processing Systems*. (Cité à la page 103.)

[Zien et Ong, 2007] ZIEN, A. et ONG, C. (2007). Multiclass multiple kernel learning. *Dans les actes de Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 1191–1198, New York, NY, USA. ACM. (Cité aux pages 22 et 136.)

[Özöğür Akyüz et Weber, 2008] Özöğür AKYÜZ, S. et WEBER, G.-W. (2008). Learning with infinitely many kernels via semi-infinite programming. *Dans les actes de EURO Mini Conference on Continuous Optimization and Knowledge-Based Technologies*. (Cité à la page 26.)

[Özöğür Akyüz et Weber, 2010a] Özöğür AKYÜZ, S. et WEBER, G.-W. (2010a). Infinite kernel learning via infinite and semi-infinite programming. *Optimization Methods and Software*, 25(6):937–970. (Cité à la page 26.)

[Özöğür Akyüz et Weber, 2010b] Özöğür AKYÜZ, S. et WEBER, G.-W. (2010b). On numerical optimization theory of infinite kernel learning. *Journal of Global Optimization*, 48(2):215–239. (Cité à la page 26.)