Classification ensembliste des séries temporelles multivariées basée sur les M-histogrammes et une approche multi-vues

Angéline Plaud

École doctorale : Sciences pour l'ingénieur

Doctorat Université Clermont Auvergne

THÈSE

pour obtenir le grade de docteur délivré par

Université Clermont Auvergne

Spécialité doctorale "Informatique"

présentée et soutenue publiquement par

Angéline PLAUD

le 4 Décembre 2019

Classification ensembliste des séries temporelles multivariées basée sur les M-histogrammes et une approche multi-vues

Directeur de thèse : **Engelbert MEPHU NGUIFO**
Co-encadrant de thèse : **Jacques CHARREYRON**

Jury

M. Emilion Richard,	Professeur	Rapporteur
M. Benadeslem Khalid,	MCF, HDR	Rapporteur
M. De Macêdo José Antonio Fernandes,	Professeur	Rapporteur
Mme Antoine Violaine,	MCF	Examinateur

LIMOS
Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes
UMR CNRS 6158, Clermont-Ferrand, France

THESE

Table des matières

Liste des figures

Liste des tableaux

Liste des Algorithmes

Remerciements

Je tiens à remercier Monsieur Engelbert Mephu Nguifo, Professeur à l'Université Clermont-Auvergne qui m'a encadré tout au long de cette thèse, pour sa gentillesse, son soutien et sa disponibilité.

Je remercie également Monsieur Jacques Charrayron, co-encadrant de la thèse, ingénieur chez Michelin, pour nos discussions et ses conseils. Chez Michelin, je remercie aussi toutes les personnes qui ont apporté leurs points de vue sur cette thèse, Marc Duvernier, Vincent Dubourg et Jean Dejonghe. Enfin je remercie Frédéric Dombprost à l'origine du projet de thèse.

J'adresse tous mes remerciements à Monsieur Richard Emilion, Professeur à l'Université d'Orléans, ainsi qu'à Monsieur Khalid Benadeslem, Maitre de conférence à l'Université Lyon 1, et enfin à Monsieur José Antonio Fernandes de Macêdo, Professeur à L'Université de Ceará, de l'honneur qu'ils m'ont fait en acceptant d'être rapporteurs de cette thèse.

J'exprime ma gratitude à Madame Violaine Antoine, Maitre de conférence à l'Université Clermont-Auvergne, qui a bien voulu faire partie du jury de cette thèse.

Je tiens à remercier particulièrement Amina Chorfi et Emeline Gayrard pour toutes nos discussions et leurs conseils, qui m'ont accompagnée tout au long de ces trois années.

Un grand merci aussi à mon conjoint pour son soutien ainsi qu'à tous les membres de ma famille.

Résumé

La mesure des différents phénomènes terrestres et l'échange d'informations ont permis l'émergence d'un type de données appelé série temporelle. Celle-ci se caractérise par un grand nombre de points la composant et surtout par des interactions entre ces points. En outre, une série temporelle est dite multivariée lorsque plusieurs mesures sont captées à chaque instant de temps. Bien que l'analyse des séries temporelles univariées, une mesure par instant, soit très développée, l'analyse des séries multivariées reste un challenge ouvert. En effet, il n'est pas possible d'appliquer directement les méthodes univariées sur les données multivariées, car il faut tenir compte des interactions entre séries de mesures.

De plus, dans le cadre d'applications industrielles, les séries temporelles ne sont pas composées d'un même nombre de mesures, ce qui complique encore leur analyse. Or les méthodes mises à disposition, aujourd'hui, pour la classification supervisée de ces séries, ne permettent pas de répondre de manière satisfaisante à cette problématique en plus d'une gestion rapide et efficace des données. Cette approche emploie donc un nouvel outil, qui n'a jamais été utilisé dans le domaine de la classification de séries temporelles multivariées, qui est le M-histogramme pour répondre à cette question.

Un M-histogramme est à la base une méthode de visualisation sur M axes de la fonction de densité sous-jacente à un échantillon de données. Son utilisation ici permet de produire une nouvelle représentation de nos données afin de mettre en évidence les interactions entre dimensions.

Cette recherche de liens entre dimensions correspond aussi tout particulièrement à un sous-domaine d'apprentissage, appelé l'apprentissage multi-vues. Où une vue est une extraction de plusieurs dimensions d'un ensemble de données, de même nature ou type. L'objectif est alors d'exploiter le lien entre ces dimensions afin de mieux classifier les dites données, au travers d'un modèle ensembliste permettant d'agréger les prédictions émises à partir de chaque vue.

Dans cette thèse, nous proposons donc une méthode multi-vues ensembliste de M-histogrammes afin de classifier les Séries Temporelles Multivariées (STM). Cela signifie que plusieurs M-histogrammes sont créés à partir de plusieurs vues des STM exploitées. Une prédiction est ensuite réalisée grâce à chaque M-histogramme. Enfin ces prédictions sont ensuite agrégées afin de produire une prédiction finale.

Il est montré dans la suite que la méthode ainsi créée est capable de répondre au problème général de la classification supervisée de STM et son efficacité est exposée sur un cas applicatif Michelin.
Mots-clef : Série temporelle multivariée, M-histogramme, Classifieur ensembliste, Apprentissage multi-vues.

Abstact

Recording measurements about various phenomena and exchanging information about it, participate in the emergence of a type of data called time series. Today humongous quantities of those data are often collected. A time series is characterized by numerous points and interactions can be observed between those points. A time series is multivariate when multiple measures are recorded at each timestamp, meaning a point is, in fact, a vector of values. Even if univariate time series, one value at each timestamp, are well-studied and defined, it's not the case of multivariate one, for which the analysis is still challenging. Indeed, it is not possible to apply directly techniques of classification developed on univariate data to the case of multivariate one. In fact, for this latter, we have to take into consideration the interactions not only between points but also between dimensions. Moreover, in industrial cases, as in Michelin company, the data are big and also of different length in terms of points size composing the series. And this brings a new complexity to deal with during the analysis. None of the current techniques of classifying multivariate time series satisfies the following criteria, which are a low complexity of computation, dealing with variation in the number of points and good classification results.

In our approach, we explored a new tool, which has not been applied before for MTS classification, which is called M-histogram. A M-histogram is a visualization tool using M axis to project the density function underlying the data. We have employed it here to produce a new representation of the data, that allows us to bring out the interactions between dimensions.

Searching for links between dimensions correspond particularly to a part of learning techniques called multi-view learning. A view is an extraction of dimensions of a dataset, which are of same nature or type. Then the goal is to display the links between the dimensions inside each view in order to classify all the data, using an ensemble classifier.

So we propose a multi-view ensemble model to classify multivariate time series. The model creates multiple M-histograms from differents groups of dimensions. Then each view allows us to get a prediction which we can aggregate to get a final prediction.

In this thesis, we show that the proposed model allows a fast classification of multivariate time series of different sizes. In particular, we applied it on a Michelin use case.
Keywords : Multivariate time series, M-histogram, Ensemble classifier, Multi-view learning.

Chapitre 1

Introduction

Sommaire

Nouvelle révolution planétaire, l'acquisition de données innombrables et variées transforme notre société. La multiplication de capteurs permet en effet de suivre et de regarder évoluer à peu près tout phénomène terrestre voire extraterrestre. L'abondance de ces données rend néanmoins le travail d'analyse fastidieux, car l'échelle de données aujourd'hui acquise dépasse de très loin la puissance de calcul mondiale disponible. Il est, en effet, possible de suivre l'évolution de nombreux phénomènes de bout en bout mais cela donne naissance à de grandes **séries temporelles**.

Dans cette introduction, le contexte de notre étude sur les séries temporelles est détaillé, ainsi que la problématique et les définitions des concepts de base afin de bien comprendre les enjeux de cette thèse.

1.1 Contexte

Dans cette première section sont détaillés tous les concepts liés à la classification des séries temporelles multivariées. Le M-histogramme est aussi expliqué afin de réaliser l'introduction de l'outil comme élément de réponse à la problématique.

1.1.1 classification supervisée

La notion de classification supervisée au sens général est ici définie. Qu'est-ce que la tâche de la classification dans l'analyse de données ?

En général

La classification supervisée de données est un concept présent dans de très nombreux domaines d'application, aussi bien la médecine DEEKSHATULU et collab. [2013], que la finance NGAI et collab. [2011] ou l'astrophysique BORNE [2008]. En effet, l'action de **classification des données** est, en réalité, la recherche de la relation permettant le lien entre un ensemble de valeurs appelées **variables ou attributs** et une valeur cible dite **classe ou étiquette**. De manière formelle, la classification supervisée est définie comme suit :

Définition : classification supervisée

Soit un ensemble E de données d'apprentissage associées avec leurs classes $c \in$ C, où C est un ensemble de valeurs dans $\mathbb{Z}$ et soit la relation $f(.)$ tel que $f(\mathrm{E}) = c_i \in \mathrm{C}$. Nous avons alors, pour tout jeu de données non classifiées A, $f(\mathrm{A}) \in \mathrm{C}$. AGGARWAL [2014].

Pour réaliser cette classification, un modèle est mis en place et se déroule en deux phases. Tout d'abord, la phase dite d'**apprentissage** qui permet de déterminer la relation entre variables et classes. Ensuite, une phase de test qui permet de vérifier si la relation déterminée est la bonne, c'est-à-dire qu'elle fait bien le lien entre attributs et classes Fig. 1.1.

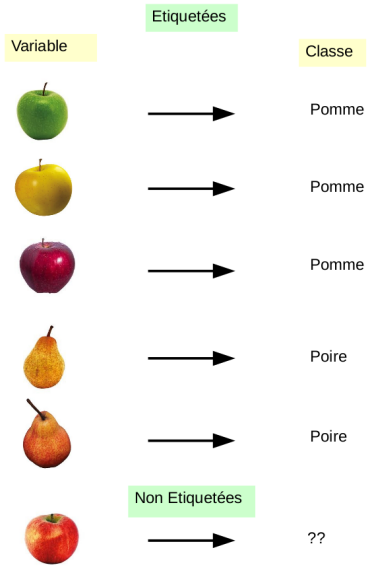


FIGURE 1.1 – Exemple de classification supervisée. La relation déterminée grâce au modèle est symbolisée par la flèche noire et est apprise sur les données étiquetées. Selon cette relation, quelle serait la classe de la nouvelle image ?

Apprentissage

Il existe trois façons différentes de gérer les données disponibles dans le cadre de l'apprentissage. Nous rappelons que le but ici est de trouver **la relation entre attributs et classes**. Cette relation peut être établie par différents modèles mathématiques qu'il va alors falloir paramétrer. Ce paramétrage du modèle est appelé la **phase d'apprentissage** afin d'estimer ladite relation.

La première manière de paramétrer le modèle est de couper un jeu de données étiquetées en deux. Puis les paramètres permettant d'approcher au mieux la relation sont recherchés sur une des deux parties. Cet ensemble de données est alors appelé jeu d'apprentissage. Ensuite, le reste des données sert à évaluer la fonction avec le paramétrage final. C'est le jeu de données test.

La deuxième manière est de séparer le jeu d'apprentissage précédemment créé en deux. Un jeu dit de **validation** est ainsi obtenu. Ce sous-jeu permet alors d'évaluer les résultats provenant du modèle paramétré en apprentissage, puis lorsque le paramétrage donne de bons résultats en validation, la prédiction finale est réalisée sur le jeu de test. Ce deuxième découpage permet d'éviter **l'overfitting**, c'est-à-dire, de créer un modèle capable uniquement de classifier les données d'apprentissage, mais qui ne fonctionnerait pas pour toute autre donnée.

Enfin, le processus qui peut être mis en place, est la validation **croisée**. Dans ce cas, les données d'apprentissage sont découpées en K sous-groupes. Durant K itérations un sous-ensemble k, où $k \in [1, \mathrm{K}]$, sert de jeu de validation. Tandis que le reste des données, soit K-1 sous-groupes servent à l'apprentissage. Pour chaque itération le score de performances sur le jeu de validation est calculé. À la suite de quoi, la moyenne des scores sur l'ensemble des sous-groupes est calculée. C'est un processus d'apprentissage plus coûteux en temps, mais qui permet d'obtenir le modèle le plus robuste aux données. C'est ce processus qui est appliqué dans cette thèse.

La classification supervisée vient d'être définie de manière générale, son application chez Michelin peut maintenant être abordée.

Chez Michelin

Dans le cadre du développement et de la recherche, Michelin innove avec la **mobilité connectée**. L'objectif est de mieux comprendre le pneu dans son environnement d'utilisation. Pour cela, de nombreux capteurs sont installés sur de multiples flottes de véhicules à la fois particuliers et professionnels. Un certain nombre de tâches de classification peuvent alors être explorées à travers les données enregistrées par ces capteurs.
Dans le cadre de cette thèse, la tâche de classification supervisée s'applique aux **performances pneumatiques**. Ces performances correspondent à différentes étiquettes, la résistance au roulement, en lien avec la consommation d'essence, le bruit du pneu ou encore l'usure. Tout particulièrement, l'objectif ici est de construire un modèle qui permet de mettre en relation **l'usage d'un véhicule et l'usure d'un pneumatique**. Pour cela, de très grandes quantités de données sont mises à disposition. Ces données sont des séries temporelles dites multivariées de longueurs variables.

1.1.2 Série temporelle

Dans le contexte de la thèse, les données exploitées sont donc appelées des **séries temporelles**. Une série temporelle est un ensemble de mesures relevées au cours du temps afin de suivre l'évolution d'un phénomène. Ces données sont présentes dans un très grand nombre d'applications, notamment grâce à l'expansion de la télématique, qui est constitué de tous les objets connectés. Par exemple en médecine KAMPOURAKI et collab. [2008], en finance LAWRENCE et GILES [1998] et en astrophysique WACHMAN et collab. [2009].

Tout d'abord, les séries temporelles dites **univariées** ne sont composées que d'une seule mesure par instant du temps. C'est le cas, par exemple, de la série bleue ou de la série orange sur l'illustration 1.2.

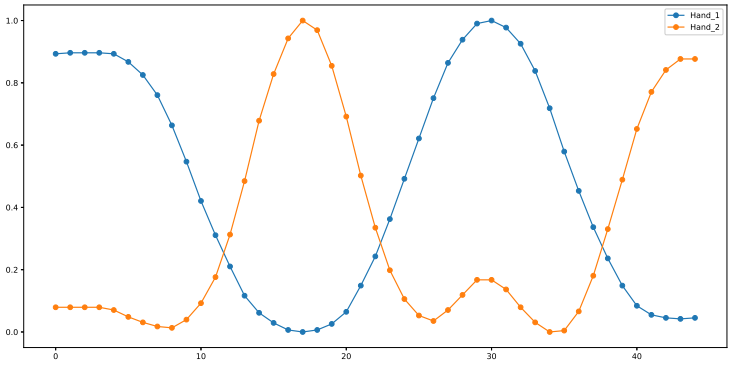


FIGURE 1.2 – Exemple d'une série temporelle multivariée avec 2 dimensions issue du jeu de données Libras BAYDOGAN et RUNGER [2015].

Une série temporelle se caractérise par des **interactions dans le temps** des points qui la composent. En effet, pour de petits ensembles de points consécutifs, il y a auto-corrélations. De la même manière, il peut y avoir des corrélations au long terme. En effet, une série temporelle peut être composée de motifs qui se répètent tout au long de la série. Un motif étant un ensemble de points, il y a donc des points qui interagissent entre eux au sein de la série. Ces interactions ainsi que le nombre de points par série à considérer influent grandement sur la capacité des modèles à classifier les données.

Définition : Série temporelle univariée

Soit X une série temporelle univariée, nous avons :

$$\mathrm{X} = [x(1), x(2), ..., x(t), ..., x(\mathrm{T})] \tag{1.1}$$

où t ∈ [1,T] et T est le nombre d'observations qui composent la série et $x(i) \in \mathbb{R}$.

Cela est d'autant plus vrai lorsque la série temporelle est **multivariée**, c'est-à-dire que plus d'une mesure sont relevées à chaque instant du temps. Dans ce cas, en plus des interactions temporelles, il y a aussi des **interactions entre séries de mesures**. Cela veut dire qu'il peut être observé des corrélations entre les différents attributs qui constituent la STM, ensemble des séries bleue et orange sur la Fig. 1.2.

Définition : Série temporelle multivariée

Soit X une série temporelle multivariée possédant M attributs à chaque instant du temps, nous avons :

$$\mathrm{X} = \begin{pmatrix} x_1(1) & ... & x_1(t) & ... & x_1(\mathrm{T}) \\ ... & ... & ... & ... & ... \\ x_m(1) & ... & x_m(t) & ... & x_m(\mathrm{T}) \\ ... & ... & ... & ... & ... \\ x_\mathrm{M}(1) & ... & x_\mathrm{M}(t) & ... & x_\mathrm{M}(\mathrm{T}) \end{pmatrix} \tag{1.2}$$

où m ∈ [1,M] et M est le nombre de dimensions, t ∈ [1,T] et T est le nombre d'observations qui composent la série.

Le cas de la série univariée est en réalité un cas particulier des séries temporelles multivariées où M = 1. Dans le cadre de cette thèse, le cas général est traité c'est-à-dire le cas des séries temporelles multivariées.

La notion de série temporelle est maintenant définie. La prochaine section aborde les caractéristiques particulières que ces données peuvent avoir et qui sont au coeur du travail de thèse.

1.1.3 Données réelles

Dans le contexte de la classification des séries temporelles multivariées, il peut être nécessaire de comparer des séries de **longueurs différentes et très variables**. En effet, un individu qui conduit sa voiture, comme dans les cas d'analyse Michelin, peut tout

aussi bien aller au travail en 15 min que partir en week-end à 3 heures de chez lui, voir Fig. 1.3.

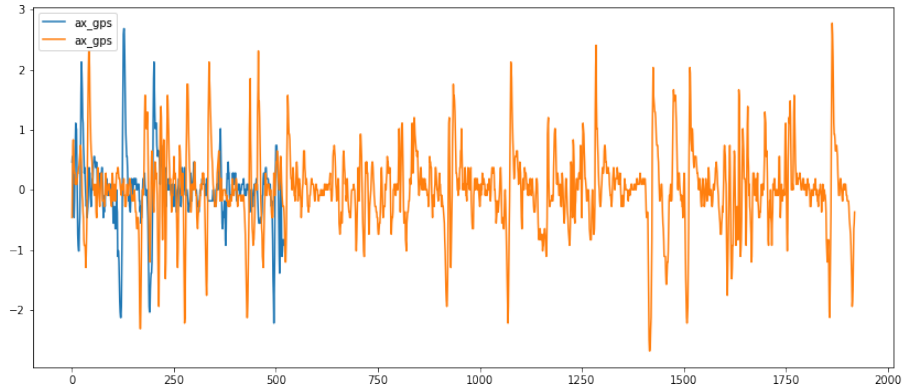


FIGURE 1.3 – Exemple de séries temporelles de tailles différentes. Le trajet bleu a duré 5 min tandis que le orange dure plus d'une heure.

Le modèle employé doit, tout de même, être capable de comparer l'information contenue dans ces deux séries. De manière formelle, il peut être défini trois cas de figures où les gestions des STM sont différentes :

Définition : Longueurs de STM

1. *STM avec la même longueur* pour toutes les dimensions et pour toutes les STM du jeu de données. C'est le cas le plus courant de la littérature.
2. *STM avec la même longueur pour chaque dimension et une longueur variable pour chaque STM.* C'est le cas lorsque le même phénomène est enregistré via des mesures de même fréquence.
3. *Dimensions et STM avec des longueurs variables.* C'est le cas lorsque le même phénomène est enregistré via des mesures de fréquences différentes et donc il y a plus ou moins de points par dimension, ou si une/plusieurs voies de mesures sont défectueuses.

Le premier cas est le plus répandu dans la recherche sur les STM, car c'est celui le plus simple. Il est, en effet, plus aisé, de par la multitude d'outils mathématiques disponibles, de comparer des ensembles de même taille. Dans la réalité, les cas (2) et (3) sont plus fréquents, il faut alors mettre en place de nouveaux outils pour les comparer. Le travail de cette thèse est consacré au deuxième cas (et implicitement au premier). L'objectif, comme décrit ci-avant, est d'être capable de **classer des données à grande variabilité de longueur**. Le troisième cas n'est pas pris en compte dans ce travail.

La spécificité des données maintenant abordée, le M-histogramme, qui permet de prendre en compte les variations de longueurs, est défini ci-après.

1.1.4 M-histogramme

Dans le contexte de ce travail, le **M-histogramme** a été utilisé de nombreuses fois afin de visualiser les données.

Le M-histogramme, ou histogramme multivarié est un **outil statistique** permettant la visualisation de la fonction de densité sous-jacente à un ensemble de données. Les formes les plus connues de l'outil sont l'*histogramme* et le *bi-histogramme*. Ce dernier est très utilisé chez Michelin, car il permet de déduire des profils accélérométriques. C'est-à-dire qu'à partir d'un bi-histogramme des accélérations longitudinales et latérales, il est possible de déduire le profil de conduite du conducteur, voir Fig. 1.4.

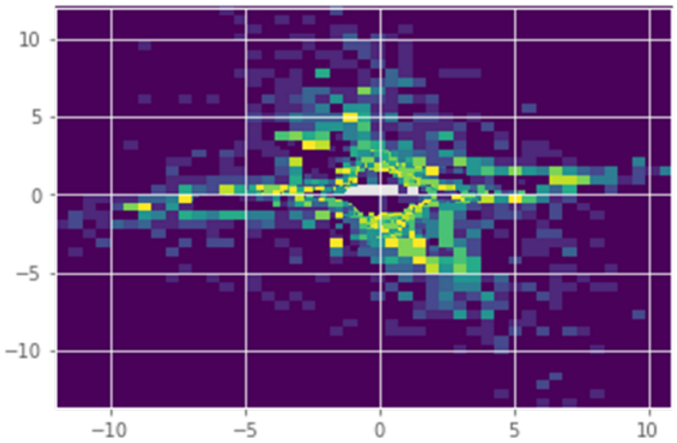


FIGURE 1.4 – Profil accélérométrique d'un pilote professionnel. L'accélération longitudinale (dans le sens de la marche du véhicule) en ordonnée est projetée en fonction de l'accélération latérale en abscisse (coté du véhicule, virage etc...).

Un M-histogramme, permet de compter la fréquence de points d'une STM qui tombent au sein de mêmes intervalles de valeurs. Les paramètres à régler sont donc M qui correspond au nombre de dimensions à projeter et le nombre d'intervalles voulu par dimension.

Définition : M-histogramme

Soit le M-histogramme S de la STM X, nous avons S une matrice de taille $\prod_i b_i$ où b_i est le nombre d'intervalles par dimension i $\in$ M et M est le nombre de dimensions. S est composée des fréquences des tuples $[x_1(t), ..., x_M(t)]$ contenus dans chaque intervalle avec $t \in$ T.

Le temps de calcul d'un M-histogramme est lié au nombre de points de la série à traiter par le nombre d'intervalles pour chaque dimension, soit $O(T(\sum_i(b_i))$.

Les M-histogrammes sont donc un appui visuel et statistique qui peuvent permettre de discriminer des profils de conduite à partir des données accélérométriques. Il est donc admis ici que les M-histogrammes peuvent s'inscrire dans le cadre d'application de classification de séries multivariées. C'est pourquoi il est intéressant d'aller plus loin dans la recherche autour de cet outil appliqué à la classification supervisée.

Précision linguistique

Il existe une version linguistique de cet outil appelé Mgramme. Celui-ci est alors appliqué sur un ensemble de mots discrets et non plus dans l'espace Euclidien continu. Nous avons par le passé utilisé ces deux notations de manière équivalente, car l'outil est fondamentalement le même. Mais ayant constaté des confusions parmi la communauté scientifique quant à l'usage de la notation, nous nous cantonnerons donc ici à l'appellation **M-histogramme**.

Le contexte de l'étude sur STM, en particulier de celles de tailles variables vient d'être exposé. Les données définies ci-avant sont donc aussi l'objet d'études Michelin qui par ailleurs utilisent le M-histogramme afin de les résumer statistiquement. La prochaine étape est la définition de la problématique.

1.2 Problématique

Le contexte vient d'être décrit, ainsi que les définitions majeures nécessaires pour que le lecteur puisse comprendre tous les aspects du travail effectué. Cette section aborde maintenant la question principale autour de laquelle la thèse porte. La question étant :

Question : Problématique

Comment classifier des STM obtenues à grande échelle, contenant d'importantes variations de longueurs ? Comment le M-histogramme peut-il apporter une partie de la réponse à cette question ?

1.2.1 Classification des séries temporelles multivariées

Lorsque les STM font l'objet d'étude de classification supervisée, plusieurs points durs sont à solutionner.

Premièrement, il faut gérer l'**aspect multi-dimensionnel** des données, c'est-à-dire qu'il faut fournir une méthode capable de relever les interactions temporelles propres aux STM, mais aussi les interactions entre dimensions.

Dans cette thèse, il faut, tout particulièrement, proposer une classification sur un **grand volume de séries** où ces dernières peuvent avoir de très importantes **variations dans le nombre de points** les composant.

Par ailleurs, la classification doit être effectuée dans un temps *relativement court*. La notion de temps étant subjective et dépendante en partie du matériel, c'est la **complexité du modèle** qui sert ici de référence. Cette dernière devrait rester en temps quadratique pour la borne maximale.

Enfin, la solution doit permettre une **interprétation visuelle humaine**. En effet, le modèle est appliqué dans un cadre industriel. L'interprétation des résultats émit par ce dernier sont donc du ressort d'experts du domaine d'application, qui doivent pouvoir les comprendre facilement.

Définition : Classification de grands volumes de STM à taille variables

Soit D un jeu de données composé de N STM avec N très grand. Nous avons X^n la n-ième série de D où $n \in N$. X^n est une STM qui possède M attributs à chaque instant t durant T observations, où $T \in [1, \tau]$, tel que :

$$X_m^n = [x_m^n(1), ..., x_m^n(T)] \tag{1.3}$$

où $m \in M$. Classifier les STM de D non étiquetées.

Enfin, si le modèle proposé peut correspondre à un applicatif industriel sur les pneumatiques, il doit aussi pouvoir fonctionner pour la classification de STM en générale, car cette problématique est commune à plusieurs domaines d'activités.

Variabilité des Séries temporelles

Il est important, pour la suite de la thèse, d'insister ici sur la variabilité des longueurs des séries temporelles. En effet, dans le cadre de l'étude de l'état de l'art sur les Séries Temporelles Univariées (STU) et les STM, la prise en compte de ces variations de longueurs est quasi-inexistante, voire nulle. La plupart des travaux sur le sujet se proposent d'interpoler le nombre de points ou de tronquer la série RATANAMAHATANA et KEOGH [2004]. Or, cela n'est pas possible dans cette étude.

En effet, si lors d'un cas d'usage comme la comparaison de profil de conducteur, ce dernier est caractérisé par plusieurs STM de taille **extrêmement** variable. Il peut conduire aussi bien 5 min pour un travail quotidien qu'une heure pour un trajet hebdomadaire. Et pourtant, ces deux types de séries le définissent et doivent pouvoir être comparés. Il n'existe à l'heure actuelle, aucun travail précisant explicitement pouvoir s'intéresser à ce genre de problème, voir chapitre État de l'art.

1.2.2 Exploration par le M-histogramme

La question de la classification supervisée des STM est une question récente et encore ouverte à l'exploration de réponses pertinentes. C'est pourquoi, l'emploi du M-histogramme, n'ayant jamais été exploré dans ce domaine, est suggéré ici.

L'utilisation du M-histogramme est innovant et répond à plusieurs points durs posés par la gestion des STM. Le M-histogramme est un outil multi-dimensionnel, il rend donc possible la gestion de toutes les dimensions d'une série. De même, il permet de se défaire de la question des variations du nombre de points, car un M-histogramme est une projection fréquentielle des données. Enfin, l'objet lui-même est à visée d'interprétation humaine et permet une visualisation simplifiée des séries.

Définition : Problématique de la classification des STM par l'usage de M-histogramme

Soit D un jeu de données de N STM et soit D′ l'ensemble des M-histogrammes les réprésentant. Nous avons S^n le n-ième M-histogramme de D′ où $n \in N$ et S^n est une projection de X^n, tel que :

$$S^n = [Freq(X^n(t) \in b_1, ..., Frequence(X^n(t) \in b] \tag{1.4}$$

où $t \in T$, b est un vecteur d'intervalles pour toutes les dimensions. Classifier les M-histogrammes de D′ non étiquetées.

La suite de la thèse s'attache donc à savoir quelle méthode utilisée pour classifier les STM à longueurs variables et si le M-histogramme est une voie de réponse à cette question.

1.2.3 Application Michelin

Bien que des données de référence de la communauté soient utilisées afin de proposer une réponse aux questions ci-avant, elles ne sont pas dans les ordres de grandeurs des données Michelin à traiter. C'est pourquoi ces dernières composent en réalité le test final que le modèle final doit passer.

En effet, les véhicules exploités génèrent plus de 4 Gigaoctets(Gb) de données par mois. Or la base est composée de plusieurs centaines de véhicules. De plus, certains capteurs de mesures peuvent remonter jusqu'à 300 mesures par seconde pour des trajets d'une heure en moyenne. Cela constitue des bases de données de données de tailles pharamineuses difficiles à exploiter dans leur globalité.
A la différence des jeux de données de référence pour lesquels la taille sera précisée dans le chapitre Résultat de référence.

Enfin, si les quantités de données sont très supérieures à celles de la littérature, la qualité de ces données est inférieure. En situation réelle de nombreuses incertitudes et biais se retrouvent dans les données. C'est pourquoi les données de références sont tout de même employées pour évaluer le travail.

Ces données Michelin représentent donc un challenge supplémentaire pour la robustesse des modèles en plus de leurs efficacités. L'objectif applicatif est donc de classifier la performance des pneumatiques par l'étude de STM obtenues sur véhicules en situations réelles.

1.3 Synthèse

Le travail de cette thèse exploite donc de grands volumes de données. Ces données sont des STM qui peuvent avoir des longueurs très variables que nous devons classifier. L'objectif est de montrer qu'une partie de la réponse peut être apportée par l'étude des M-histogrammes qui n'a encore jamais été faite pour l'analyse des STM.

Dans le Chapitre 2, un état de l'art sur la classification des séries temporelles est réalisé, ainsi que sur le M-histogramme. Dans le Chapitre 3, le modèle mis au point

est présenté. Dans le Chapitre 4, les résultats du modèle, sur les données de références issues de la littérature, sont donnés. Dans le Chapitre 5, les résultats pour un cas applicatif Michelin sont exposés. Enfin, le Chapitre 6 constitue une conclusion du travail et des perspectives.

Points clefs du chapitre Introduction

Problématique	Classification de grands volumes de STM de longueurs variables
Points durs	Nombre de dimensions Interactions entre dimensions Nombre de points variable Interactions entre points

Dans chaque chapitre, sont précisées les références bibliographiques employées, afin de faciliter la lisibilité de la thèse.

1.4 Références

AGGARWAL, C. C. 2014, *Data classification : algorithms and applications*, CRC press. 8

BAYDOGAN, M. et G. RUNGER. 2015, «Learning a symbolic representation for multivariate time series classification», *DMKD*, vol. 29, nº 2, p. 400–422. 10

BORNE, K. D. 2008, «Scientific data mining in astronomy», dans *Next Generation of Data Mining*, Chapman and Hall/CRC, p. 114–137. 8

DEEKSHATULU, B., P. CHANDRA et collab.. 2013, «Classification of heart disease using k-nearest neighbor and genetic algorithm», *Procedia Technology*, vol. 10, p. 85–94. 8

KAMPOURAKI, A., G. MANIS et C. NIKOU. 2008, «Heartbeat time series classification with support vector machines», *IEEE transactions on information technology in biomedicine*, vol. 13, nº 4, p. 512–518. 10

LAWRENCE, S. R. et C. L. GILES. 1998, «Method and apparatus for foreign exchange rate time series prediction and classification», US Patent 5,761,386. 10

NGAI, E. W., Y. HU, Y. H. WONG, Y. CHEN et X. SUN. 2011, «The application of data mining techniques in financial fraud detection : A classification framework and an academic review of literature», *Decision support systems*, vol. 50, nº 3, p. 559–569. 8

RATANAMAHATANA, C. A. et E. KEOGH. 2004, «Everything you know about dynamic time warping is wrong», dans *Third Workshop on Mining Temporal and Sequential Data*, Citeseer. 15

WACHMAN, G., R. KHARDON, P. PROTOPAPAS et C. R. ALCOCK. 2009, «Kernels for periodic time series arising in astronomy», dans *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, p. 489–505. 10

Chapitre 2

État de l'art

Sommaire

L'ensemble du contexte de la thèse est défini ainsi que la question scientifique à laquelle nous devons répondre, et qui est la classification de grands volumes de STM, en particulier celles avec des longueurs variables. Et quelle réponse les M-histogrammes peuvent apporter sur le sujet. Ce chapitre présente les méthodes qui tentent de répondre à la première partie de la question. Dans un premier temps, une courte introduction à la classification des STU est proposée car celle-ci constitue la base des travaux STM. Par la suite, les méthodes de classification supervisée des séries multivariées sont détaillées. Un état de l'art des techniques autour de la gestion de STM est aussi proposé. Enfin, les méthodes sur l'exploitation des M-histogrammes sont aussi expliquées.

2.1 Classification des séries temporelles univariées

Les travaux sur les STU sont les premiers travaux présentés portant sur les séries temporelles. Le premier papier identifié date de 1978 SAKOE et CHIBA [1978], et concerne la reconnaissance de discours vocaux. L'objectif est alors de créer une classification de mêmes mots prononcés par différentes personnes. Pour cela, les auteurs proposent une nouvelle distance capable de prendre en compte la notion de temps. Cette méthode n'est pas une distance au sens mathématique du terme, car elle ne respecte pas l'inégalité triangulaire, mais par abus de langage est considérée comme telle. Cette méthode s'appelle Dynamic Time Warping (DTW), voir Fig. 2.1.

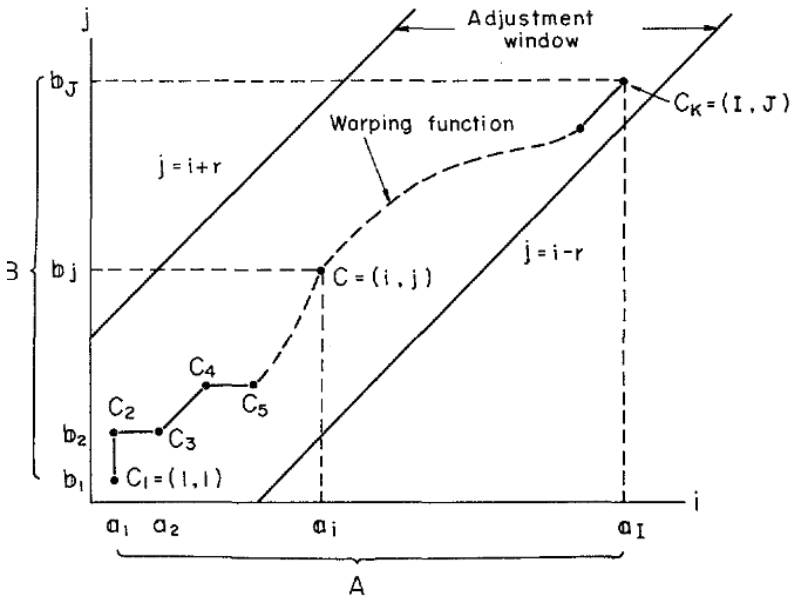


FIGURE 2.1 – Illustration de Dynamic Time Warping en 1978 qui décrit le fonctionnement de la méthode.

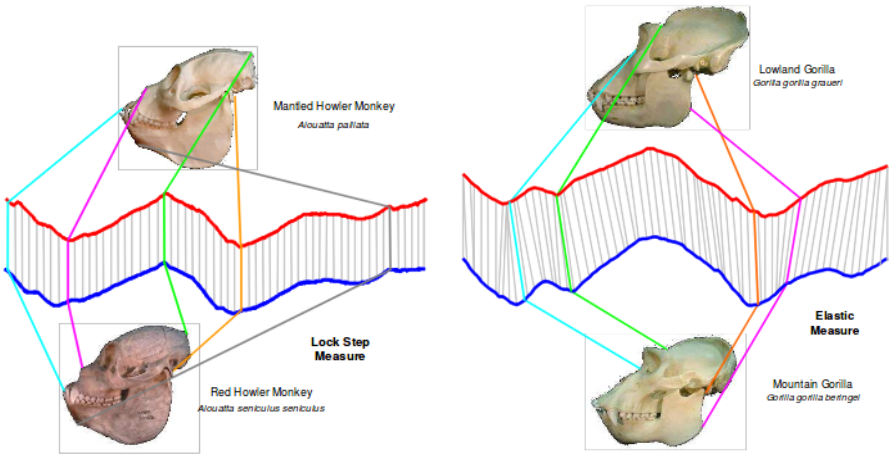
Dans la matrice des distances entre points des séries A et B, il faut extraire la distance cumulée qui ne s'éloigne pas trop de la diagonale (contrainte à ajuster) et qui donne la somme la plus petite.

Cette méthode introduit la première catégorie de recherches en matière de séries temporelles qui concerne **les mesures de similarités**. Depuis 1978, plusieurs autres

mesures de similarités, ou distance, ont été présentées, dont des variations de la méthode originale.

2.1.1 Mesure de similarités

La distance DTW est exploitée depuis 1978 et a été introduite pour la classification supervisée de STU en 1994 BERNDT et CLIFFORD [1994]. Ce papier décrit l'application la plus connue de DTW, car celle-ci donne encore aujourd'hui parmi les meilleurs résultats de classification des séries temporelles WANG et collab. [2013].



(a) Calcul de la distance euclidienne entre deux STU

(b) Calcul de DTW entre deux STU

FIGURE 2.2 – Comparaison entre la distance euclidienne et DTW issue de WANG et collab. [2013]

Là où une distance euclidienne est calculée entre points à la même position dans la série, DTW permet le calcul avec plusieurs points de positions avoisinantes et donc introduit la notion de **souplesse temporelle**, voir Fig. 2.2.

Il existe plusieurs variations de DTW comme Weighted Dynamic Time Warping (WDTW) JEONG et collab. [2011] qui utilise un système de pénalité par poids afin de pénaliser les points les plus éloignés les uns des autres. Il existe aussi Derivative Dynamic Time Warping (DDTW) GÓRECKI et ŁUCZAK [2013] qui utilise **les dérivées de la série initiale** afin de nuancer la mesure de similarités. Par ailleurs, DTW est aussi employé dans d'autre domaine que la classification tel que dans BENKABOU et collab. [2018], où les auteurs proposent d'utiliser cette mesure de similarités dans le cadre de la détection d'anomalies.

Enfin, il existe pléthore d'autres méthodes de similarités telles que Time Warp Edit Distance (TWED) MARTEAU [2009] qui mesure la similarité entre deux séries A et B en comptant le nombre de transformations à effectuer sur A et B afin de faire correspondre les deux séries. Mais aussi Move-Split-Merge (MSM) STEFAN et collab. [2013] qui comptent aussi le nombre de transformations, mais qui n'utilisent pas les mêmes opérations. Ce sont toutes des distances avec DTW regroupées sous l'appellation de **distances élastiques**, mais DTW reste la plus utilisée.

Cette catégorie de méthodes donne de bons résultats en classification supervisée mais elles ont en réalité de grandes complexités. C'est pourquoi, une autre catégorie de méthodes a émergé.

2.1.2 Représentation des séries temporelles

La deuxième catégorie de méthodes, introduite la première fois par FALOUTSOS et collab. [1994] concerne le changement de représentation. Ce dernier propose d'utiliser les coefficients de la transformée de Fourier afin d'extraire des sous-séries des STU. Cela crée une nouvelle représentation des données plus compacte et donc plus rapide.

En effet, les séries temporelles constituées de nombreux points sont longues à analyser. Certains proposent alors de changer la représentation des données pour accélérer cette analyse. Les plus connues aujourd'hui sont Symbolic Aggregate approXimation (SAX) LIN et collab. [2007] et Shapelet YE et KEOGH [2011].

SAX

Présentée en 2007, SAX est une nouvelle représentation des STU qui permet de transformer ces dernières en chaîne de caractères.

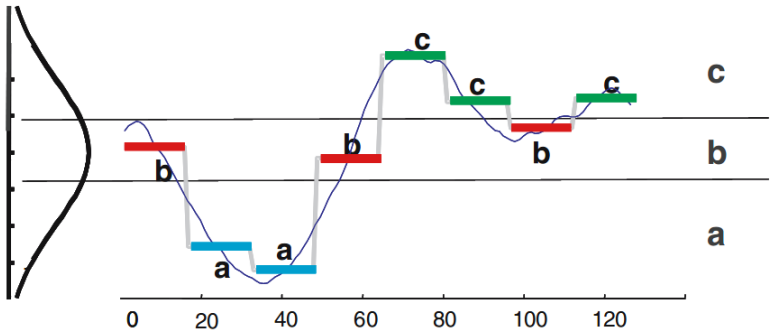


FIGURE 2.3 – Illustration issue de LIN et collab. [2007] du fonctionnement de SAX

Tout d'abord, la STU est transformée en moyenne par segment par une méthode appelée Piecewise Aggregation Approximation (PAA) YI et FALOUTSOS [2000] . Puis en étudiant la distribution et en attribuant un alphabet par tranche de distribution, la série est transformée en chaîne de caractère, voir Fig. 2.3.

Ce travail est à la base de plusieurs autres comme SAX and Vector Space Model (SAX-VSM) SENIN et MALINCHIK [2013] et Bag of SFA Symbol (BOSS) SCHÄFER [2015]. Le premier modèle transforme une STU en un ensemble ordonné de mots. L'ordre est déterminé par l'importance du mot dans la discrimination de la classe. Le deuxième emploie une autre méthode que PAA afin de segmenter la série mais construit aussi des ensembles de mots. Ce dernier modèle est l'un des meilleurs modèles de classification de STU à l'heure actuelle BAGNALL et collab. [2017].

Shapelet

Une autre représentation très connue est l'extraction d'une sous-série de la STU qui s'appelle un shapelet YE et KEOGH [2009]. Cette extraction est la partie de la série

qui permet au mieux de déterminer la classe, voir Fig. 2.4.

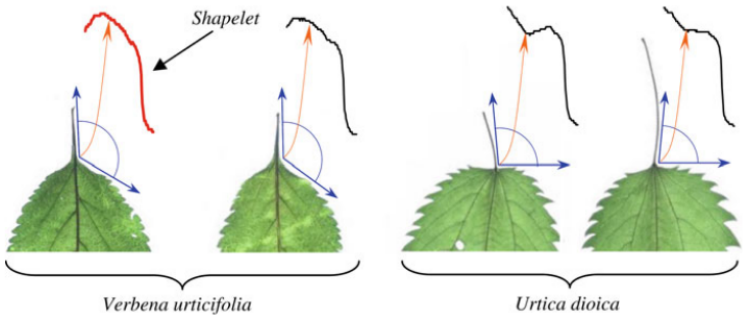


FIGURE 2.4 – Illustration de l'extraction d'un Shapelet issue de YE et KEOGH [2011]. Si le contour des feuilles est vu comme des STU, alors la partie de la queue de la feuille est la partie la plus déterminante pour trouver la classe.

De même que SAX, la méthode des Shapelets a entraîné la recherche de nouvelles solutions telles que BOSTROM et BAGNALL [2015] et RAKTHANMANON et KEOGH [2013]. Le premier améliore la performance du modèle shapelet de départ. Tandis que le deuxième travail permet d'accélérer le calcul, plutôt long, des shapelets.

En effet, il est à préciser ici que l'apprentissage d'un shapelet n'est pas rapide, car il nécessite d'extraire toutes les sous-séries possibles de la série principale. Mais une fois le shapelet identifié, la phase d'indexation est rapide.

Transformée de Fourrier

Il existe un certain nombre de travaux utilisant la transformée de Fourrier afin de représenter les séries temporelles, comme la méthode Symbolic Fourrier Approximation (SFA), SCHÄFER et HÖGQVIST [2012]. La transformée de fourrier discrète est alors appliquée permettant d'obtenir les coefficients sous-jacents à la série. Puis la valeur des coefficients est associée à une lettre par un découpage en intervalles des valeurs des coefficients, voir Fig. 2.5.

Ce travail a lui-même donné lieu a une autre méthode appelée Word ExtrAction for time SEries cLassification (WEASEL) SCHÄFER et LESER [2017]. Cette méthode transforme non plus une série en un ensemble de caractères, mais en un ensemble de mots, voir Fig. 2.6.

Cette dernière méthode donne de très bons résultats pour la classification des séries temporelles univariées.

2.1.3 Modèles mathématiques

Une dernière catégorie de modèles correspond à ceux qui proposent de réduire les séries temporelles par l'usage de modèles mathématiques, où la relation entre attributs et classes est estimée par une fonction dont il faut déterminer les paramètres.

Auto Regressive Moving Average (ARMA)

Le modèle ARMA GRANGER et ANDERSEN [1978], propose une estimation de la fonction sous-jacente à une série temporelle. L'hypothèse derrière l'utilisation de ce mo-

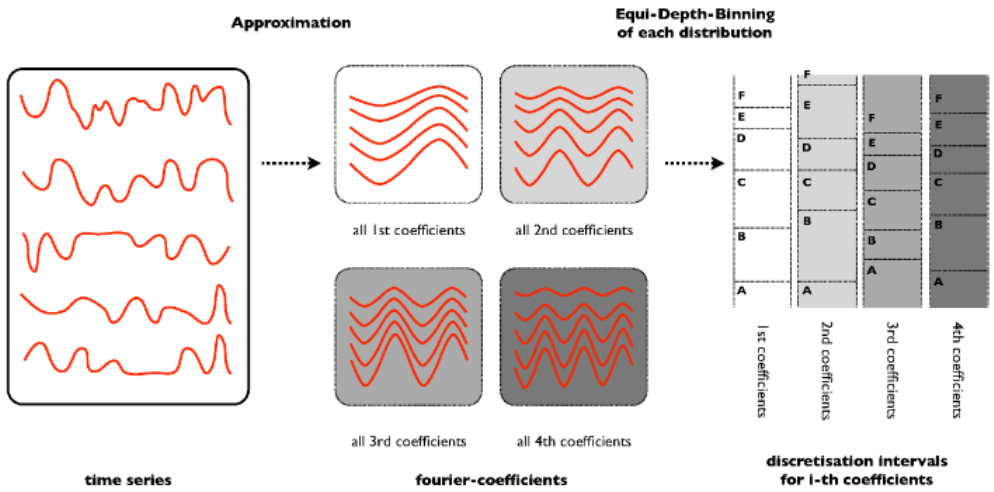


FIGURE 2.5 – Illustration de l'application de SFA issue de SCHÄFER et HÖGQVIST [2012].

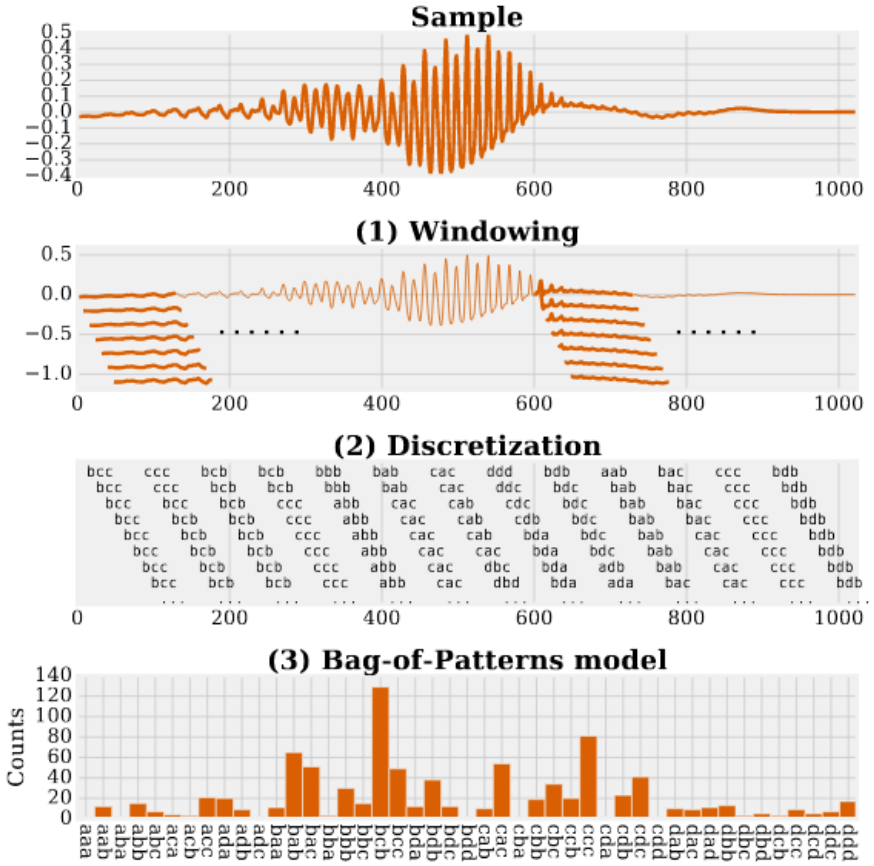


FIGURE 2.6 – Illustration de l'application de WEASEL issue de SCHÄFER et LESER [2017].

dèle est que la valeur de la série à l'instant t dépend des valeurs de la série aux instants précédents $t-1, t-2, t-3, ...$ Il est alors possible d'approcher cette valeur en mettant en place une fonction polynomiale tel que $x_t = \alpha_p * x_p + ... + \alpha_{t-1} * x_{t-1} + \beta_q * \epsilon_q + \beta_{t-1} * \epsilon_{t_1}$, où ϵ est le bruit contenu au sein de la série. La méthode KAN DENG et collab. [1997] utilise alors les deux vecteurs α et β comme nouvelle représentation de la série temporelle dans le cadre de la classification des STU.

Auto Regressive Moving Average (ARIMA)

Le modèle ARIMA BOX et collab. [1994] est très proche du modèle ARMA. La seule différence est que le modèle ARIMA prend en compte un nouveau terme dans la fonction qui est celui des différences de degré d où d est à paramétrer. Ce terme permet de prendre en compte la non-stationnarité des séries. Une série est considérée comme stationnaire si la loi statistique sous-jacente aux données ne change pas avec le temps. Concrètement, si la moyenne et l'écart-type évoluent sur l'ensemble de la série alors la série est non stationnaire. Dans ces cas-là, le modèle ARMA n'est plus suffisant pour estimer la série. Il faut alors utiliser le modèle ARIMA.

Ces modèles anciens sont principalement utilisés dans le domaine des séries temporelles issues de la finance.

Tous les travaux présentés ci-avant sont très performants pour les STU et ont donc constitué une base de travail sérieuse pour l'étude des STM. Nous allons voir dans quelles mesures ils ont inspiré les travaux de classification des STM.

2.2 Classification des séries temporelles multivariées

Il existe relativement peu de travaux sur les STM comparativement à ceux sur les STU et la plupart de ceux-ci sont basés sur les travaux cités ci-avant.

2.2.1 DTW

Dans le travail de 2017 SHOKOOHI-YEKTA et collab. [2017] présentent plusieurs adaptations de DTW au cas du multivariée. Cette publication démontre pourquoi l'adaptation des méthodes de classification des STU au cas des STM est **non-triviale**. En effet, lorsque il faut adapter DTW aux STM, deux cas de figures se présentent. Il est nécessaire de rappeler que DTW n'est pas une distance et donc aucune des adaptations non plus.

Question : Adaptation de DTW aux STM

Soit Q et C deux STM telles que Q(x,y) et C(x,y) les dimensions x et y sont des séries temporelles composées de T observations. Comment adapter DTW ?

La Fig. 2.7 permet de voir que DTW peut s'appliquer de deux manières sur les STM. Soit DTW est calculée :

— sur chaque point. Il faut alors sommer la distance pour chacun des points, où un point est un vecteur de plusieurs dimensions. Cas (a)

— sur chaque dimension. Il faut alors sommer les distances obtenues pour chaque dimension. Cas (b)

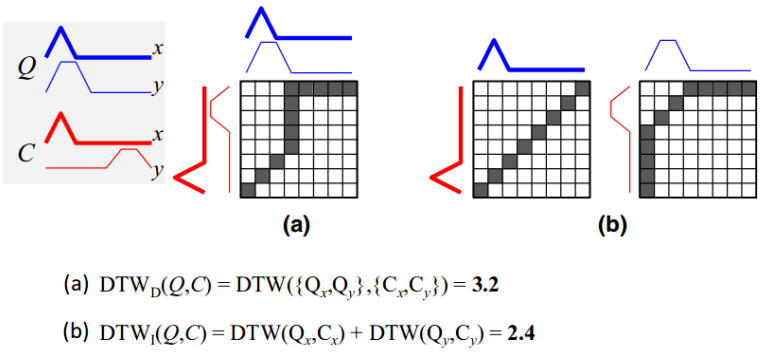


FIGURE 2.7 – Illustration des multiples applications de DTW aux STM issue de SHOKOOHI-YEKTA et collab. [2017]

Il est aussi visible sur la figure que ces deux adaptations ne donnent pas la même distance. C'est pourquoi, la méthode proposée est une troisième alternative qui est un mélange des deux cas de figures ci-avant. L'adaptation à choisir dépend alors du cas d'application.

La complexité de ce modèle n'est pas explicitement donnée, mais se base sur une optimisation minutieuse de DTW qui serait de $O(n)$ RATANAMAHATANA et KEOGH [2004]. Néanmoins, aucune preuve ne démontre qu'après adaptation la complexité soit toujours la même. Si la méthode peut être vue comme assez rapide et efficace pour gérer de petites séries temporelles, il n'en est pas de même pour les longues séries et encore moins à tailles variables.

En effet, l'application de DTW sur des séries de différentes tailles nécessite **l'interpolation** de points pour obtenir des séries de tailles identiques RATANAMAHATANA et KEOGH [2004]. Cela est loin d'être judicieux lorsque de très grosses différences de taille sont constatées entre les séries comme c'est le cas dans les données exploitées ici. En effet, les points interpolés augmentent la taille d'échantillonnage et donc la fréquence à laquelle les données ont supposément été acquises. Des séries avec des fréquences d'échantillonnage différentes sont alors comparées.

L'autre solution est de tronquer la série, ce qui veut dire ici, la perte de beaucoup d'information, ce qui n'est pas judicieux.

Cela fait de cette méthode une application très limitée dans la réalité et qui n'est pas réellement applicable dans le cas de figure présent.

2.2.2 SMTS

La méthode SMTS BAYDOGAN et RUNGER [2015] présentée en 2015, est une technique de classification des STM inspirée de SAX. L'idée est de transformer une STM en un ensemble de mots par le biais d'une forêt aléatoire BREIMAN [2001], voir Fig. 2.8.

La forêt aléatoire est composée d'arbres via un algorithme classique de construction d'arbre. Chaque arbre est créé par le tirage aléatoire d'une ou plusieurs dimensions de la STM originale ainsi que de ses dérivées. En effet, l'auteur introduit le fait que la dérivée d'une STM qui est donc composée des dérivées de chaque dimension,

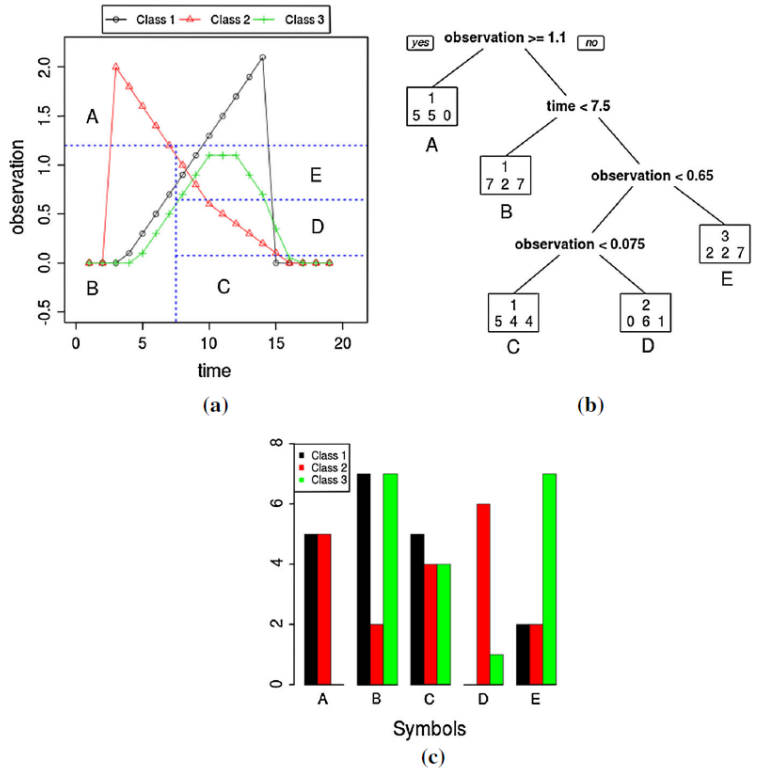


FIGURE 2.8 – Illustration issue de BAYDOGAN et RUNGER [2015] explicitant la création d'un arbre de la random Forest (b) pour une série (a) et qui montre le lien entre les mots et les classes (c).

est porteuse d'un sens nouveau qui peut aider le processus de classification supervisée.

Ces arbres permettent grâce aux feuilles d'associer aux STM des ensembles de mots qui vont chacune les caractériser. L'ambition ici est que des mots communs décrivent des classes communes. Lorsqu'une STM non étiquetée est analysée par le modèle, un ensemble de mots lui est associé grâce aux arbres. Ces ensembles de mots servent ensuite à nourrir une nouvelle forêt aléatoire permettant de lier des mots à des classes.

Cette méthode fait figure de précurseur dans le domaine de la classification des STM et donne de bons résultats. Par ailleurs, elle est capable de gérer des séries de tailles variables. Cependant, la complexité associée à ce modèle est liée à deux forêts aléatoires. La première, qui transforme les séries en mots, est de l'ordre de $O(J_{ins}\sqrt{2M+1}NT(R-1))$ où J_{ins} est le nombre d'arbre , M le nombre de dimensions, N le nombre de séries, T la longueur de la série, R la profondeur d'un arbre. La deuxième forêt qui classifie les nouvelles représentations est d'une complexité de l'ordre de $O(J_{ts}\sqrt{RJ_{ins}}NlogN)$, où J_{ts} est le nombre d'arbres. De fait, ce modèle dépend d'un grand nombre de paramètres et il nécessite un long apprentissage.

2.2.3 Shapelet

Il existe un certain nombre de papiers portant sur l'adaptation de shapelet aux séries multivariées.

Dans GRABOCKA et collab. [2016], les auteurs proposent une nouvelle formulation

de la méthode des shapelets afin d'accélérer le temps de calcul. Il note toutefois que l'algorithme n'est pas capable de traiter de très gros jeux de données STM en un temps raisonnable. Dans le travail He et collab. [2015], la méthode proposée est une classification anticipée. Toutefois, nous pouvons déjà voir que les résultats obtenus sur le peu de jeux de données sont très inférieurs à ceux des autres modèles, SMTS présenté ci-avant et WEASEL+MUSE présenté ci-après. Enfin dans Bostrom et Bagnall [2017] présenté dans un workshop, les auteurs avouent tout simplement que cette idée n'est pas la bonne.

Aucun de ces papiers ne donne de réponse satisfaisante aux problèmes rencontrés lors de la classification de STM. Tout d'abord, calculer un shapelet est extrêmement coûteux comme décrit dans Grabocka et collab. [2016] et est difficilement applicable même sur de petits jeux de données. De plus, les shapelets ici décrits ne prennent pas en compte les différentiations de longueurs. Cela étant, les adaptations peuvent encore être améliorées et c'est pourquoi ils sont présentés ici Bostrom et Bagnall [2017].

2.2.4 WEASEL+MUSE

WEASEL+MUSE Schäfer et Leser [2017] est un papier non publié officiellement, mais qui néanmoins présente des résultats prometteurs. Ceci est une généralisation du modèle WEASEL Schäfer et Leser [2017] présenté ci-avant. La différence majeure pour le traitement des STM se trouve dans le comptage des mots qui se fait aussi par couple de mots.

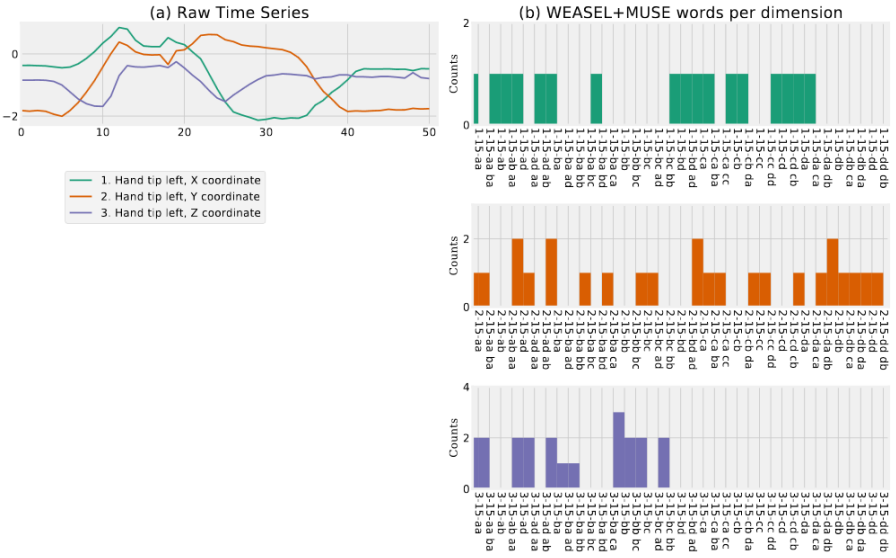


Figure 2.9 – Illustration issue de Schäfer et Leser [2017] montrant la transformation d'une STM en bi-histogramme de mots.

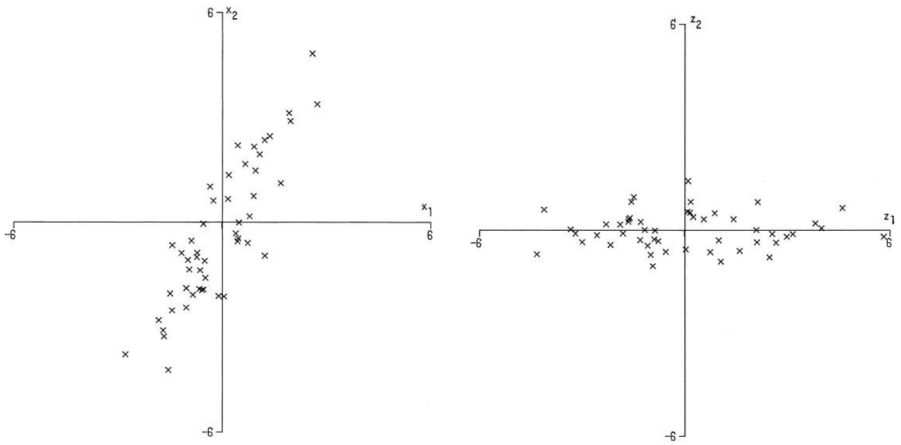
C'est ainsi la première fois que la notion de *bi-histogramme* est abordée, voir Fig. 2.9, puisque le comptage des couples est appelé un bigramme comme expliqué dans le chapitre introduction. Néanmoins, l'espace des données exploitées ici est différent de l'espace que nous comptons utiliser. Dans un cas, c'est l'espace discret composé par un ensemble de mots. Dans l'autre, son application se fait sur les STM brutes et donc

dans l'espace des valeurs réelles. Par ailleurs, dans ce papier, s'il n'est pas explicité clairement que la méthode est capable de gérer des STM à longueur variable, les jeux de données testés en contiennent. Cependant, il semble compliqué de mettre à l'échelle cette méthode dans le cas de grands volumes de données. En effet, la méthode emploie de multiples extractions itératives afin de créer les mots représentatifs de la série temporelle. De fait, la complexité n'est pas clairement établie, mais estimée l'ordre de $O(min[NT^3, N^2T^2] * m)$ où N est le nombre de séries, T la longueur et M le nombre d'attributs en se basant sur les travaux publiés sur les STU et en prenant en compte le caractère ici multidimensionnel des données. C'est une complexité très importante qui rend extrêmement compliquée l'utilisation du modèle sur le cas applicatif Michelin.

2.2.5 Autres techniques

Parmi les autres méthodes rencontrées, certaines sont basées sur les techniques de **manifold learning** HUO et collab. [2007]. Celles-ci consistent à réduire le nombre initial de dimensions d'un jeu de données. Sont alors obtenues de nouvelles dimensions, moins nombreuses, mais contenant suffisamment de l'information principale des données de base.

La technique la plus connue et utilisée est appelée Principle Component Analysis (PCA) JOLLIFFE et CADIMA [2016]. Cette méthode de factorisation matricielle permet d'obtenir les dimensions vectorielles qui contiennent les informations principales ainsi que le vecteur des coefficients représentant l'importance des dites dimensions. Ces dimensions sont appelées *composantes principales*. Il est possible de ne garder qu'une sous-partie des nouvelles dimensions et de limiter alors la perte d'information.



(a) Projection des données selon leurs dimensions x_1 et x_2.

(b) Projections des données selon leurs composantes principales z_1 et z_2.

FIGURE 2.10 – Application de PCA 2.10b sur les données 2.10a issue de JOLLIFFE et CADIMA [2016].

Sur la Fig. 2.10, il est visible que seule la dimension z_1 contient l'information principale du jeu de données. En effet, c'est la seule dimension qui possède une grande

amplitude de valeurs. Ainsi, pour réduire le temps de classification, seule la dimension z_1 peut être gardée.

Cette technique est néanmoins coûteuse en temps de calcul. La complexité de PCA est de $O(min[n^2m, nm^2])$ où n est le nombre de données et m le nombre de dimensions. En outre, le danger de cette technique est de perdre trop d'informations, si trop de dimension sont écartées, pour ne garder que l'information dite principale. Il n'est pourtant pas possible de déterminer à l'avance le nombre de dimensions à garder, ni même si la base de données peut être orthogonalisée.

Dans le cas de l'application de ces méthodes aux séries temporelles, sur l'ensemble des travaux utilisant PCA YANG et SHAHABI [2004] et LI [2016], aucun n'est reproductible et ils ne sont pas comparés aux autres via des données de références. Le premier travail propose d'utiliser PCA puis met en place une nouvelle distance afin de classifier les nouvelles séries. Le deuxième travail propose une nouvelle implémentation de PCA plus adaptée à l'étude des STM afin de diminuer les temps de calcul et de mieux considérer les caractéristiques de ces données particulières. La complexité est alors de l'ordre de $O(Tm^3)$ cela signifie que le temps de calcul grandit de manière cubique avec le nombre de dimensions.

Enfin, il peut être cité, un autre type de travail sur les STM, la réduction en une structure de **meta-features**. Réalisé en 2004, KADOUS et SAMMUT [2005], l'auteur propose une nouvelle structure de représentation où une meta-feature est un attribut calculé à partir de toutes les STM. Ces meta-features peuvent être des attributs d'agrégation ou d'extraction d'informations qui permettent de rendre compte de l'information temporelle des séries, comme la moyenne, le point du milieu de la série, le gradient, la durée, etc. Une meta-feature peut aussi être une suite de valeurs extraite de la série similaire à ce que réalise l'algorithme PAA par exemple. Ce travail permet la prise en charge de STM à taille variable, mais les résultats présentés sont très en dessous des capacités actuelles des techniques plus récentes, alors même que ce travail a été évalué sur peu de jeux de données.

2.2.6 Résumé

Finalement, les techniques présentées ne semblent pas répondre réellement à notre cas d'application qui est **la classification de grands volumes de STM de tailles très variables**. Néanmoins, deux techniques se rapprochent des besoins exprimés ci-avant. Ce sont SMTS et WEASEL+MUSE, qui sont donc considérés comme éléments comparatifs dans la suite du travail. Elles permettront, sur les données de références, de situer l'emploi des M-histogrammes, dans l'état de l'art. Par ailleurs, bien que la distance DTW ne puisse être exploitée sur les séries de tailles variables, cet algorithme est toujours considéré comme une référence dans le domaine donc il est ajouté comme méthode de référence de base, voir Tab. 2.1.

Un ensemble de méthodes de classification supervisée des STM vient d'être présenté, bien qu'il ne réponde que partiellement aux besoins de classification rapide de

Algorithme	Longueurs	Complexité	Gestion de gros volume	Classifieur utilisé
DTW_d et DTW_i	Fixe	$O(NT^2M)$	non	1NN
SMTS	Fixe et peu variable	$O(J_{ins}\sqrt{2M+1}NT(R-1))$ suivi de $O(J_{ts}\sqrt{RJ_{ins}}NlogN)$	non	Random Forest
WEASEL+MUSE	Fixe et peu variable	$O(min[NT^3, N^2T^2] * m$	non	BOP

TABLEAU 2.1 – Synthèse des algorithmes de l'état de l'art

grands volumes de STM à tailles variables. C'est pourquoi, la suite introduit une sous-partie des méthodes de classification qui peuvent être employées pour y répondre. Ce sont les modèles ensemblistes et multi-vues.

2.3 Classification par méthodes multi-vues ensemblistes

Dans la suite de la thèse, il est intéressant d'expliquer les concepts de classifieur ensembliste et d'apprentissage multi-vues en particulier. Une courte introduction de chacun des concepts est donc réalisée. Dans un premier temps, le concept d'apprentissage ensembliste est décrit, avec ses origines ainsi que son apport dans le cadre de la classification de séries temporelles. Dans un second temps, le cas particulier de l'apprentissage multi-vues est présenté, ainsi que son intérêt et la manière de l'appliquer aux séries temporelles.

2.3.1 Classification ensembliste

La classification ensembliste se définit comme l'apprentissage par plusieurs classifieurs. Ce concept, introduit en 1979 par DASARATHY et SHEELA [1979] présente les *systèmes ensemblistes*. L'idée est qu'il vaut mieux utiliser plusieurs petits classifieurs simples plutôt qu'un seul classifieur complexe, voir Fig. 2.11.

Bien qu'aujourd'hui la recherche sur l'apprentissage ensembliste ait explosée et ait proposée de multiples modèles, deux sont plus connus, le *boosting* et le *bagging*.

Le Boosting SCHAPIRE [1990] consiste à entraîner plusieurs classifieurs tel que le ième classifieur s'entraîne sur les données malclassifiées du classifieur i-1. Le bagging BREIMAN [1996] quant à lui entraîne les classifieurs sur différents sous-ensembles de données extraits du jeu d'apprentissage de base. L'autre différence majeure se situe dans la manière d'agréger les classifications des différents classifieurs. Dans le cadre du boosting, un vote avec poids est mis en place, là où le bagging réalise simplement un vote majoritaire. Enfin, l' algorithme du bagging à l'avantage d'être facilement parallélisable.

Dans le cadre des ensembles d'apprentissage appliqués aux séries temporelles, c'est le bagging qui est privilégié. Ce dernier répond mieux aux besoins de rapidité de traitement nécessaire à la tâche. Ainsi, c'est ce qui est employé dans le modèle Collective of

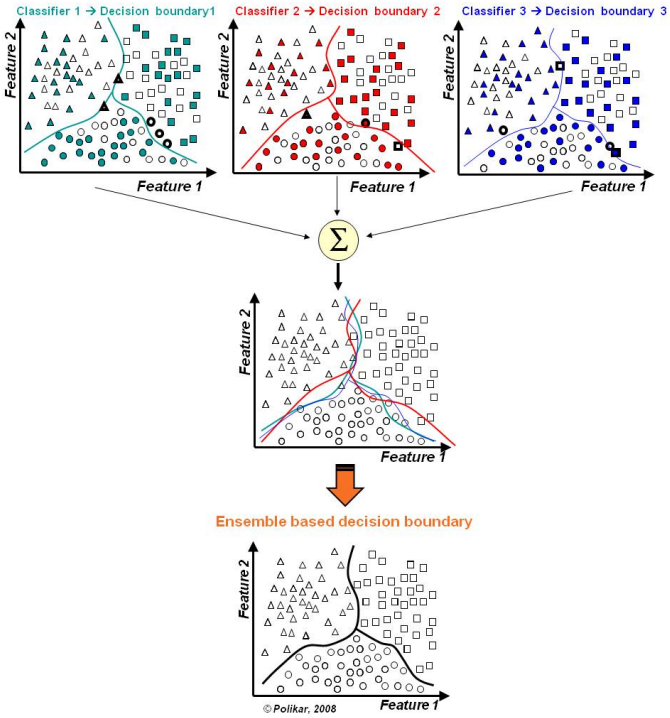


FIGURE 2.11 – Illustration issue de POLIKAR [2012] montrant comment trois classifieurs ensemble permettent d'obtenir une meilleure classification que par un seul.

Transformation-Based Ensembles (COTE) BAGNALL et collab. [2015] qui est le meilleur modèle de classification de STU BAGNALL et collab. [2017], à l'heure actuelle.

COTE

COTE est un modèle de classification supervisée STU appartenant à la catégorie des modèles ensemblistes. Ce modèle propose de créer des classifieurs issus de l'univers des séries temporelles puis d'agréger leurs prédictions. Ce modèle regroupe donc Transformée de Fourier, shapelet, DTW, etc. En combinant les meilleurs méthodes de l'existant, le modèle donne aujourd'hui de très bons résultats. Néanmoins, comme ce modèle intègre des algorithmes lents comme shapelet son temps d'exécution s'en trouve dégradé. Par ailleurs, de la même manière, il n'intègre pas la notion de variations de longueurs. Enfin, aucune adaptation multivariée n'a été proposée pour le moment.

2.3.2 Apprentissage multi-vues

L'apprentissage multi-vues est une branche de l'apprentissage ensembliste créée spécifiquement afin de prendre en compte les différences de natures que peuvent avoir les dimensions d'un jeux de données XU et collab. [2013]. Ces dimensions peuvent être groupées par type et ce nouveau groupe est appelé une vue.

L'objectif de l'apprentissage multi-vues est donc de créer des sous-groupes d'apprentissages, appelés **vues** afin de **mettre en avant la relation** contenue au sein de ces dimensions.

LI et collab. [2016] exploite cette notion de multi-vues pour la classification de séries temporelles. Dans ce travail, les dimensions des séries temporelles multivariées **sont séparées en groupes de même nature** sur lesquels est appliqué un modèle de projection discriminante. C'est-à-dire que l'objectif est de minimiser une fonction de perte globale appliquée sur les vues telle que :

$$min \mathrm{F}(\mathrm{X}_1, .., \mathrm{X}_i, ..., \mathrm{X}_v) \tag{2.1}$$

où X_i est une vue composée de dimension de la STM et $i \in v$ et v est le nombre de vues.

L'intérêt de ce travail est la meilleure prise en compte des relations entre dimensions d'une série temporelle et propose une vraie nouveauté via l'intégration de la notion de vue. Néanmoins, ce travail ne s'est pas situé par rapport à l'existant et ne traite pas le problème des longueurs variables, même au sein d'une vue.

La notion de multi-vues reste peu exploitée dans le domaine des séries temporelles et il semble intéressant de donner une meilleure exploitation de cette notion qui s'applique particulièrement bien aux STM. C'est pourquoi, elle est présentée ici et mise en place dans la suite de la thèse.

Un ensemble de techniques de classification applicables aux séries temporelles vient d'être présenté. Il reste donc à aborder la dernière partie qui présente le changement de représentation des séries temporelles au profit des M-histogrammes.

2.4 Utilisation des M-histogrammes

Une autre notion importante entrevue dans l'introduction est le concept de M-histogramme. Celui-ci reste peu exploité en dehors de son utilité purement statistique, il semble donc pertinent de montrer son utilisation dans la littérature actuelle.

2.4.1 Cas particulier : Le bi-histogramme et l'histogramme

Tout d'abord, le M-histogramme est en réalité connu dans ses formes les plus basiques, c'est-à-dire sous forme d'histogramme à une dimension et de bi-histogramme à deux dimensions. Il est exploité dans de nombreux domaines d'applications, tels que l'imagerie, SHAH et collab. [2015] où les histogrammes sont utilisés afin d'améliorer le contraste d'une image, ou encore la classification de flux avec SOULE et collab. [2004], où les auteurs utilisent les histogrammes afin de classifier les flux de trafic internet.

Il est aussi présent dans le domaine de la mobilité et des capteurs où les bi-histogrammes sont utilisés afin de réduire l'information. Nous pouvons citer ce travail VAICIUKYNAS et collab. [2017] voir Fig. 2.12.

Dans ce travail, l'objectif est de réduire les informations d'un véhicule en limitant la perte d'information. Deux dimensions sont alors projetées sous forme de bi-histogramme telles que la vitesse de rotation du moteur ou encore la pression appliquée sur la pédale.

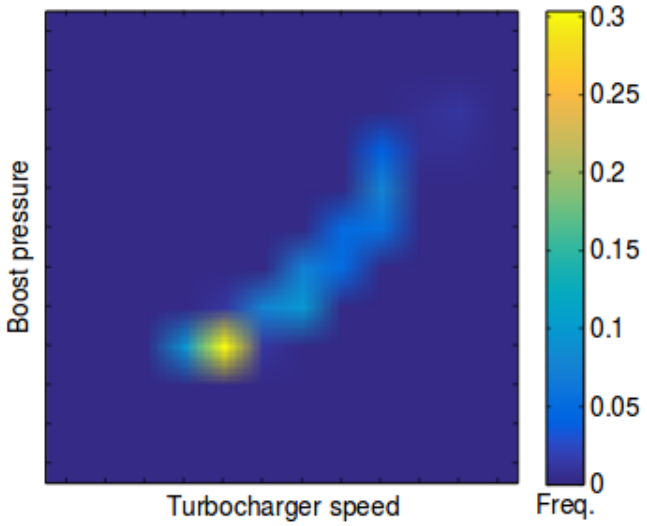


FIGURE 2.12 – Illustration issue de VAICIUKYNAS et collab. [2017] montrant le bi-histogramme à deux dimensions de capteurs posés dans un camion

2.4.2 M-histogramme

La notion de M-gramme se trouve quant à elle très régulièrement dans la reconnaissance de mots et de langages BROWN et collab. [1992]. C'est cette définition qui est utilisée dans SCHÄFER et LESER [2017]. Cette définition diffère légèrement de la définition générale, car l'outil s'applique ici sur un ensemble de mots. Un 1-gramme est le comptage par mot, bi-gramme compte les couples de deux mots, etc... Par ailleurs, le bi-histogramme dans sa définition statistique est souvent présenté sous forme de matrice, voir Fig. 2.12. Tandis qu'en modèle de langage, les bi-grammes sont toujours représentés *à plat* comme les histogrammes, voir Fig. 2.9.

Ainsi bien que la notion de bi-gramme soit déjà exploitée dans la classification de STM, le M-histogramme quant à lui n'exploite pas le même espace de projection. Cela représente un aspect de la nouveauté exploitée dans cette thèse. Par ailleurs, il est aussi employé, comme dans VAICIUKYNAS et collab. [2017], pour réduire les données dans le but d'accélérer la tâche de classification.

Les méthodes les plus performantes dans le cadre de la classification de STM viennent d'être présentées. Leurs particularités ainsi que leurs limites dans le cadre applicatif de cette thèse ont aussi été données. Par ailleurs, de nouveaux concepts permettant de répondre à la question scientifique posée ont aussi été introduits. L'ensemble des informations importantes abordées dans ce chapitre va maintenant être synthétisé.

2.5 Résumé

La recherche sur la classification de STU est aujourd'hui bien avancée, mais celle sur les STM est balbutiante. Par ailleurs, si des modèles efficients sont disponibles sur

le sujet, aucun ne s'intéresse **explicitement** à la problématique des variations non-négligeables de longueurs de séries. De la même manière, peu de travaux présentés exploitent réellement de grands volumes de données. De ce fait, les complexités des modèles proposés freinent leur exploitation sur de grands volumes de séries temporelles. De plus, si le concept de relation inter-dimensions est introduit dans chaque publication, seul un travail exploite réellement la notion de vues pourtant faite pour explorer ce sujet. Enfin, il en est de même pour la notion de M-histogramme qui permet l'exploitation de vues de données tout en réduisant l'information. Dans la suite de la thèse est donc défini un modèle permettant **la classification de grands volumes de STM de longueurs variables par l'exploitation de M-histogrammes multi-vues**.

Points clefs du chapitre Etat de l'art

Méthodes principales issues de l'état de l'art tentant de répondre à la question scientifique de la classification des STM de tailles variables :

Méthodes de classification des STM de références	— DTW — WEASEL+MUSE — SMTS
Limites de ces modèles	— Gestion de grands volumes — Complexités importantes — Prise en compte des variations de longueur

Techniques de l'état de l'art qui peuvent répondre à la question scientifique que nous voulons exploiter :

Classifieurs ensemblistes	— Exploités sur les STU par COTE avec succès — A exploiter sur les STM
Apprentissage multi-vues	— Exploité sur les STM brutes — A exploiter sur des représentation de STM
M-histogrammes	— Exploité partiellemnt sur les STM par comptage de mots avec WEASEL+MUSE — A exploiter sur les STM brutes

2.6 Références

BAGNALL, A., J. LINES, A. BOSTROM, J. LARGE et E. KEOGH. 2017, «The great time series classification bake off : a review and experimental evaluation of recent algorithmic advances», *Data Mining and Knowledge Discovery*, vol. 31, nº 3, doi :

10.1007/s10618-016-0483-9, p. 606–660, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-016-0483-9. 22, 32

BAGNALL, A., J. LINES, J. HILLS et A. BOSTROM. 2015, «Time-series classification with cote : The collective of transformation-based ensembles», *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, nº 9, doi :10.1109/TKDE.2015.2416723, p. 2522–2535, ISSN 1041-4347. 32

BAYDOGAN, M. G. et G. RUNGER. 2015, «Learning a symbolic representation for multivariate time series classification», *Data Mining and Knowledge Discovery*, vol. 29, nº 2, doi :10.1007/s10618-014-0349-y, p. 400–422, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-014-0349-y. 26, 27

BENKABOU, S.-E., K. BENABDESLEM et B. CANITIA. 2018, «Unsupervised outlier detection for time series by entropy and dynamic time warping», *Knowledge and Information Systems*, vol. 54, nº 2, doi :10.1007/s10115-017-1067-8, p. 463–486, ISSN 0219-3116. 21

BERNDT, D. J. et J. CLIFFORD. 1994, «Using dynamic time warping to find patterns in time series», dans *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, AAAIWS'94, AAAI Press, p. 359–370. URL http://dl.acm.org/citation.cfm?id=3000850.3000887. 21

BOSTROM, A. et A. BAGNALL. 2015, «Binary shapelet transform for multiclass time series classification», dans *Big Data Analytics and Knowledge Discovery*, édité par S. Madria et T. Hara, Springer International Publishing, Cham, ISBN 978-3-319-22729-0, p. 257–269. 23

BOSTROM, A. et A. J. BAGNALL. 2017, «A shapelet transform for multivariate time series classification», *CoRR*, vol. abs/1712.06428. URL http://arxiv.org/abs/1712.06428. 28

BOX, G. E., G. M. JENKINS et REINSEL. 1994, «Time series analysis : forecasting and control», . 25

BREIMAN, L. 1996, «Bagging predictors», *Machine Learning*, vol. 24, nº 2, p. 123–140. 31

BREIMAN, L. 2001, «Random forests», *Machine learning*, vol. 45, nº 1, p. 5–32. 26

BROWN, P. F., P. V. DESOUZA, R. L. MERCER, V. J. D. PIETRA et J. C. LAI. 1992, «Class-based n-gram models of natural language», *Computational linguistics*, vol. 18, nº 4, p. 467–479. 34

DASARATHY, B. V. et B. V. SHEELA. 1979, «A composite classifier system design : Concepts and methodology», *Proceedings of the IEEE*, vol. 67, nº 5, doi :10.1109/PROC.1979.11321, p. 708–713. 31

FALOUTSOS, C., M. RANGANATHAN et Y. MANOLOPOULOS. 1994, «Fast subsequence matching in time-series databases», *SIGMOD Rec.*, vol. 23, nº 2, doi :10.1145/191843.

191925, p. 419–429, ISSN 0163-5808. URL http://doi.acm.org/10.1145/191843.191925. 22

GÓRECKI, T. et M. ŁUCZAK. 2013, «Using derivatives in time series classification», *Data Mining and Knowledge Discovery*, vol. 26, nº 2, doi :10.1007/s10618-012-0251-4, p. 310–331, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-012-0251-4. 21

GRABOCKA, J., M. WISTUBA et L. SCHMIDT-THIEME. 2016, «Fast classification of univariate and multivariate time series through shapelet discovery», *Knowledge and Information Systems*, vol. 49, nº 2, doi :10.1007/s10115-015-0905-9, p. 429–454, ISSN 0219-3116. URL https://doi.org/10.1007/s10115-015-0905-9. 27, 28

GRANGER, C. et A. ANDERSEN. 1978, *An introduction to bilinear time series models*, nº 8 dans Angewandte Statistik und Ökonometrie, Vandenhoeck und Ruprecht, Göttingen, ISBN 3525112394. 23

HE, G., Y. DUAN, R. PENG, X. JING, T. QIAN et L. WANG. 2015, «Early classification on multivariate time series», *Neurocomputing*, vol. 149, doi :https://doi.org/10.1016/j.neucom.2014.07.056, p. 777 – 787, ISSN 0925-2312. URL http://www.sciencedirect.com/science/article/pii/S092523121401008X. 28

HUO, X., X. S. NI et A. K. SMITH. 2007, «A survey of manifold-based learning methods», *Recent advances in data mining of enterprise data*, p. 691–745. 29

JEONG, Y.-S., M. K. JEONG et O. A. OMITAOMU. 2011, «Weighted dynamic time warping for time series classification», *Pattern Recognition*, vol. 44, nº 9, doi :https://doi.org/10.1016/j.patcog.2010.09.022, p. 2231 – 2240, ISSN 0031-3203. URL http://www.sciencedirect.com/science/article/pii/S003132031000484X, computer Analysis of Images and Patterns. 21

JOLLIFFE, I. et J. CADIMA. 2016, *Philosophical Transactions of the Royal Society of London Series A*, vol. 374, p. 20150 202. 29

KADOUS, M. W. et C. SAMMUT. 2005, «Classification of multivariate time series and structured data using constructive induction», *Machine Learning*, vol. 58, nº 2, doi :10.1007/s10994-005-5826-5, p. 179–216, ISSN 1573-0565. URL https://doi.org/10.1007/s10994-005-5826-5. 30

KAN DENG, A. W. MOORE et M. C. NECHYBA. 1997, «Learning to recognize time series : combining arma models with memory-based learning», dans *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 'Towards New Computational Principles for Robotics and Automation'*, p. 246–251. 25

LI, H. 2016, «Accurate and efficient classification based on common principal components analysis for multivariate time series», *Neurocomputing*, vol. 171, doi :https://doi.org/10.1016/j.neucom.2015.07.010, p. 744 – 753, ISSN 0925-2312. URL http://www.sciencedirect.com/science/article/pii/S0925231215009844. 30

LI, S., Y. LI et Y. FU. 2016, «Multi-view time series classification : A discriminative bilinear projection approach», dans *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ACM, p. 989–998. 32

LIN, J., E. KEOGH, L. WEI et S. LONARDI. 2007, «Experiencing sax : a novel symbolic representation of time series», *Data Mining and Knowledge Discovery*, vol. 15, nº 2, doi :10.1007/s10618-007-0064-z, p. 107–144, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-007-0064-z. 22

MARTEAU, P. 2009, «Time warp edit distance with stiffness adjustment for time series matching», *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, nº 2, doi :10.1109/TPAMI.2008.76, p. 306–318, ISSN 0162-8828. 21

POLIKAR, R. 2012, *Ensemble Learning*, Springer US, Boston, MA, ISBN 978-1-4419-9326-7, p. 1–34, doi :10.1007/978-1-4419-9326-7_1. URL https://doi.org/10.1007/978-1-4419-9326-7_1. 32

RAKTHANMANON, T. et E. KEOGH. 2013, «Fast shapelets : A scalable algorithm for discovering time series shapelets», dans *proceedings of the 2013 SIAM International Conference on Data Mining*, SIAM, p. 668–676. 23

RATANAMAHATANA, C. A. et E. KEOGH. 2004, «Everything you know about dynamic time warping is wrong», dans *Third Workshop on Mining Temporal and Sequential Data*, Citeseer. 26

SAKOE, H. et S. CHIBA. 1978, «Dynamic programming algorithm optimization for spoken word recognition», *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, nº 1, p. 43–49. 20

SCHÄFER, P. 2015, «The boss is concerned with time series classification in the presence of noise», *Data Mining and Knowledge Discovery*, vol. 29, nº 6, doi :10.1007/s10618-014-0377-7, p. 1505–1530, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-014-0377-7. 22

SCHÄFER, P. et M. HÖGQVIST. 2012, «Sfa : a symbolic fourier approximation and index for similarity search in high dimensional datasets», dans *Proceedings of the 15th International Conference on Extending Database Technology*, ACM, p. 516–527. 23, 24

SCHÄFER, P. et U. LESER. 2017, «Fast and accurate time series classification with weasel», dans *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, ACM, New York, NY, USA, ISBN 978-1-4503-4918-5, p. 637–646, doi :10.1145/3132847.3132980. URL http://doi.acm.org/10.1145/3132847.3132980. 23, 24, 28

SCHÄFER, P. et U. LESER. 2017, «Multivariate time series classification with WEASEL+MUSE», *CoRR*, vol. abs/1711.11343. URL http://arxiv.org/abs/1711.11343. 28, 34

SCHAPIRE, R. E. 1990, «The strength of weak learnability», *Machine Learning*, vol. 5, nº 2, doi :10.1007/BF00116037, p. 197–227, ISSN 1573-0565. 31

SENIN, P. et S. MALINCHIK. 2013, «Sax-vsm : Interpretable time series classification using sax and vector space model», dans *2013 IEEE 13th International Conference on Data Mining*, ISSN 1550-4786, p. 1175–1180, doi :10.1109/ICDM.2013.52. 22

SHAH, G., A. KHAN, A. SHAH, M. RAZA et M. SHARIF. 2015, «A review on image contrast enhancement techniques using histogram equalization», *Science International*, vol. 27, p. 1297–1302. 33

SHOKOOHI-YEKTA, M., B. HU, H. JIN, J. WANG et E. KEOGH. 2017, «Generalizing dtw to the multi-dimensional case requires an adaptive approach», *Data Mining and Knowledge Discovery*, vol. 31, nº 1, doi :10.1007/s10618-016-0455-0, p. 1–31, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-016-0455-0. 25, 26

SOULE, A., K. SALAMATIA, N. TAFT, R. EMILION et K. PAPAGIANNAKI. 2004, «Flow classification by histograms : Or how to go on safari in the internet», *SIGMETRICS Perform. Eval. Rev.*, vol. 32, nº 1, doi :10.1145/1012888.1005696, p. 49–60, ISSN 0163-5999. 33

STEFAN, A., V. ATHITSOS et G. DAS. 2013, «The move-split-merge metric for time series», *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, nº 6, doi : 10.1109/TKDE.2012.88, p. 1425–1438, ISSN 1041-4347. 21

VAICIUKYNAS, E., M. ULICNY, S. PASHAMI et S. NOWACZYK. 2017, «Evaluating dimensionality reduction of 2d histogram data from truck on-board sensors», . 33, 34

WANG, X., A. MUEEN, H. DING, G. TRAJCEVSKI, P. SCHEUERMANN et E. KEOGH. 2013, «Experimental comparison of representation methods and distance measures for time series data», *Data Mining and Knowledge Discovery*, vol. 26, nº 2, doi :10.1007/s10618-012-0250-5, p. 275–309, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-012-0250-5. v, 21

XU, C., D. TAO et C. XU. 2013, «A survey on multi-view learning», *arXiv preprint arXiv :1304.5634*. 32

YANG, K. et C. SHAHABI. 2004, «A pca-based similarity measure for multivariate time series», dans *2Nd ACM International Workshop on Multimedia Databases*, ACM, p. 65–74. 30

YE, L. et E. KEOGH. 2009, «Time series shapelets : a new primitive for data mining», dans *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, p. 947–956. 22

YE, L. et E. KEOGH. 2011, «Time series shapelets : a novel technique that allows accurate, interpretable and fast classification», *Data Mining and Knowledge Discovery*, vol. 22, nº 1, doi :10.1007/s10618-010-0179-5, p. 149–182, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-010-0179-5. 22, 23

YI, B.-K. et C. FALOUTSOS. 2000, «Fast time sequence indexing for arbitrary lp norms», dans *Proceedings of the 26th International Conference on Very Large Data Bases*, VLDB '00, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN 1-55860-715-3, p. 385–394. URL http://dl.acm.org/citation.cfm?id=645926.671689. 22

Chapitre 3

Ensemble de M-histogrammes multi-vues

Sommaire

Dans le chapitre précédent, ont été présentés des modèles tentant de répondre à la question de la classification des séries temporelles multivariées. Mais ceux-ci ne répondent que partiellement aux points durs exposés que sont la gestion de grands volumes de séries de longueurs variables avec une complexité acceptable. C'est pourquoi, plusieurs autres concepts d'apprentissage, techniques de classification, représentation de données, qui peuvent aider à répondre à la question, ont été présentés. En particulier, quelle réponse peut apporter le M-histogramme au problème ?

Ce chapitre explique comment ces concepts sont combinés afin de donner naissance à une nouvelle méthode permettant de répondre à la problématique de la classification des séries temporelles de tailles variables disponibles en grands volumes. La théorie derrière le méthode est développée, les algorithmes sont démontrés et un exemple illustratif détaillé permet de comprendre le méthode.

3.1 Principe

Le problème de la classification de grandes volumes de STM de longueurs variables, est donc abordé par l'exploitation d'un ensemble de M-histogrammes multi-vues. Cette méthode se découpe en trois étapes principales détaillées dans les sections ci-après. Ces étapes sont la création des M-histogrammes, l'apprentissage multi-vues, puis la classification ensembliste. La figure 3.1 un diagramme de la méthode permet de comprendre l'imbrication des étapes avec la prédiction finale.

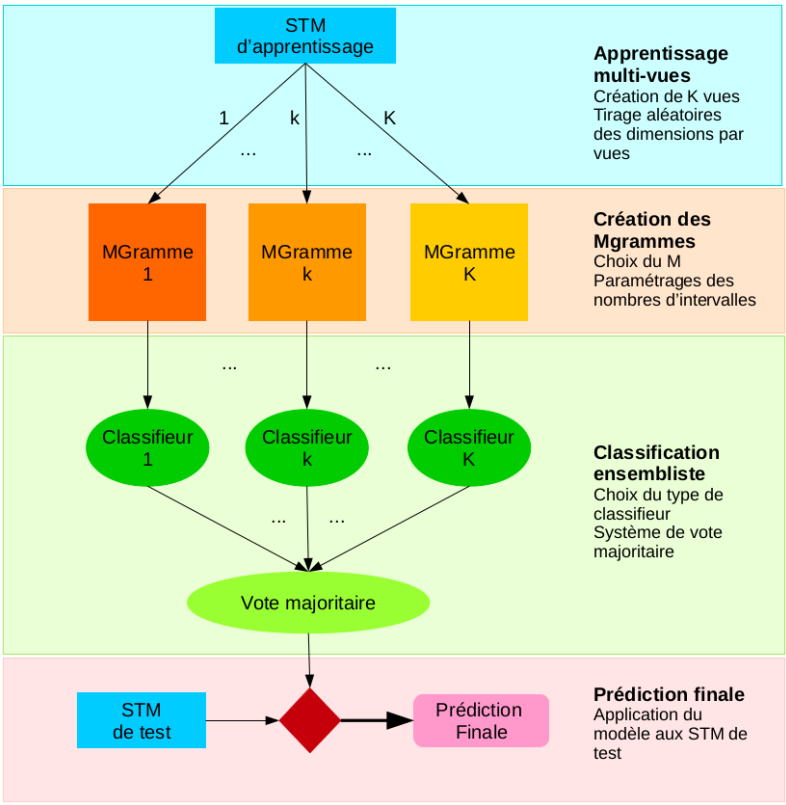


FIGURE 3.1 – Ensemble de M-histogramme Multi-Vues

Dans la suite de cette thèse, la méthode est appelée Ensemble de Mgramme Multi-Vues (EMMV).

La première étape décrite est celle de la création des M-histogrammes et non celle de l'apprentissage de vues. Ce choix est fait, car c'est de la représentation par les M-histogrammes que découle les choix d'apprentissage et non l'inverse.

Il est à rappeler la définition formelle de la problématique à laquelle doit répondre la méthode EMMV :

Définition : Classification de grands volumes de STM à taille variables par la méthode EMMV

Soit D un jeu de données composé de N STM avec N très grand. Nous avons X^n la n-ième série de D où $n \in N$. X^n est une STM qui possède M attributs à chaque instant t durant T observations, où $T \in [1, \tau]$, tel que :

$$X^n_m = [x^n_m(1), ..., x^n_m(T)] \tag{3.1}$$

où $m \in M$. Soit C l'attribut *classe* recherchée tel que $C = c_1, c_j$. Classifier les STM de D non étiquetées, tel que $EMMV(X^n) \in C$, où EMMV est la relation estimée par la méthode.

3.2 Description des données par M-histogrammes

Afin de traiter les STM rapidement tout en exploitant les relations intra et inter dimensions, la méthode est basée sur l'utilisation de M-histogramme. Le M-histogramme permet, en effet, de projeter les dimensions d'une série dans une représentation de taille potentiellement inférieure.
Suivant la définition du M-histogramme donnée page 16, la taille de l'objet finale peut être plus petite à condition que $M' << M$ où M' est le nombre de dimensions du M-histogramme et M le nombre de dimensions de la STM. De la même manière $B << T$ où B est le nombre d'intervalles par dimension du M-histogramme et T la longueur de la série.

Théorème : M-histogramme comme technique de réduction

$\forall$ STM X de M dimensions et de longueur $T \in \tau$ où τ est la longueur maximale atteinte par les séries.
$\exists$ S un M-histogramme de dimension M' et de taille $\prod B$ où B est le nombre d'intervalles par dimension, tel que
$M' <= M$ et $B <= T$ et que $|S| <= |X|$

Preuve : M-histogramme comme technique de réduction

La preuve est naturelle.
En effet, soit un raisonnement par l'absurde, nous avons : $\exists$ X, $\exists!$ S tel que M′ <= M et B <= T et que |S| <= |X|. Ceci est impossible car $M, M', B, T \in \mathbb{N}$ et donc les inégalités sont toujours vraies. De la même manière, $\forall$ S, $\exists!$ X tel que M′ <= M et B <= T et que |S| <= |X| n'a pas de sens pour les mêmes raisons.

Les M-histogrammes sont donc exploités afin de représenter de manière plus compacte les données tout en exploitant les relations contenues entre dimensions et points. La complexité est définie comme $o(TB)$, soit le nombre d'itérations maximum possibles, voir Alg. 1. En effet, il faut itérer sur tous les points de la série afin de savoir dans quel intervalle du M-histogramme tombe le point considéré. Dans le pire des cas, il faut atteindre le dernier intervalle de valeurs pour incrémenter la valeur de la matrice du M-histogramme, ligne 8. Cela n'arrive en réalité jamais. L'exécution réelle est donc plus courte.

Algorithme : Création d'un M-histogramme

Data : X une STM, b_i les intervalles pour chaque dimension i
Result : S un M-histogramme

```
1 S ← Zéro(b);
2 for t ∈ T do
3     for j ∈ b_i do
4         if x_i(t) ∈ b_i(j) then
5             STOP;
6             Sauvegarder b_i(j);
7     Répéter pour tous les b_i;
8     S(b_i(j)) += 1 ;
```

Preuve d'Algorithme : Création d'un M-histogramme

Terminaison
L'algorithme termine forcément, car l'ensemble est contenu dans une boucle for et une STM est de taille T ∈ τ finie.

Correction
Par rapport au déroulement de l'algorithme et à la déclaration de S, la sortie sera forcément une matrice de taille B et donc un M-histogramme.

Complétude
De la même manière, si l'entrée attendue est bien une STM alors nous aurons un M-histogramme en sortie.

Donc l'algorithme permet la transformation d'une STM en un M-histogramme de taille plus réduite. Néanmoins, comme toute technique de réduction, le M-histogramme

amène une **perte d'information** contenue dans les données, notamment sur l'ordre des événements, leurs durées, les tendances contenues.

C'est pourquoi le choix de compléter la STM de base avec sa **dérivée** et sa **somme cumulée** est fait ici.

3.2.1 Dérivée de la STM

Il est facile de trouver des cas où le M-histogramme des séries de base n'est pas suffisant pour discriminer deux objets faisant partie de deux classes différentes.

Question : Perte de la durée et des tendances ?

Soit X et Y deux STM, tel que :

$$\mathrm{X} = (x(1), ...x(t), x(\mathrm{T})$$

avec x un vecteur de dimension M et

$$\mathrm{Y} = (x(t), ..., x(\mathrm{T}), ..., x(1))$$

Nous construisons S(X) et S(Y), respectivement les M-histogrammes de chaque série. Est -ce que $\mathrm{S(X)} \neq \mathrm{S(Y)}$?

La réponse à cette question est non. Les deux M-histogrammes sont les mêmes. Cela revient à perdre les notions de durées des événements et des tendances elles-mêmes qui sont pourtant primordiales dans certains cas d'usage. C'est pourquoi nous ajoutons la dérivée. Cette opération mathématique permet de récupérer les tendances au sein de données. En outre, c'est une notion qui est aussi exploitée dans SMTS Baydogan et Runger [2015].

3.2.2 Somme cumulée

De la même manière, il existe aussi des cas où le M-histogramme fait perdre la notion d'ordre des événements contenus dans une série.

Question : Perte de l'ordre des évènements ?

Soit X et Y deux STM, tel que :

$$\mathrm{X} = (x(1), ...x(t), x(\mathrm{T})$$

avec x un vecteur de dimension M et

$$\mathrm{Y} = (x(t), x(t+1), x(t+2), ..., x(\mathrm{T}-2), x(\mathrm{T}-1), x(\mathrm{T})...x(1), x(2))$$

où $x(t+2) - x(t+1) = x(3) - x(1) = x(\mathrm{T}) - x(\mathrm{T}-1)$ Nous construisons S(X) et S(Y), respectivement les M-histogrammes de chaque série. Est -ce que $\mathrm{S(X)} \neq \mathrm{S(Y)}$?

Encore une fois, la réponse est non et la condition de similitude des bornes de blocs fait que les M-histogrammes des dérivées sont aussi identiques. Donc la notion de

somme cumulée est aussi ajoutée ici. En effet, la somme cumulée se rapproche de par sa nature au concept d'intégration mathématique. Cette intégration quant à elle, permet d'approcher l'idée de la position des points au sein de la STM et donc le concept d'ordre temporel.

Cette nouvelle définition des STM est appelée une STM **renforcée**. Cette proposition n'a , à notre connaissance, jamais été faite.

Définition : STM renforcée

Soit X une STM renforcée, tel que 3M est le nombre de dimension et $T \in \tau$ la longueur de la série, nous avons :

$$X = \begin{pmatrix} \dots & \dots & \dots & \dots & \dots \\ x_m(1) & \dots & x_m(t) & \dots & x_m(T) \\ \dots & \dots & \dots & \dots & \dots \\ x_m(2) - x_m(1) & \dots & x_m(t+1) - x_m(t) & \dots & x_m(T) - x_m(T-1) \\ \dots & \dots & \dots & \dots & \dots \\ x_m(1) & \dots & \sum_{i=1}^{t} x_m(i) & \dots & \sum_{i=1}^{T} x_m(i) \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

Dans la littérature, il peut être constaté l'usage régulier de la dérivée comme dans les méthodes SMTS et WEASEL+MUSE. Bien que non théorisé son usage est admis au sein de la communauté afin de relever tendances et évolution. Au contraire, la somme cumulée reste peu déployée. A. Bondu et son équipe en font référence dans leur travail BONDU et collab. [2019]. C'est un papier très récent sur la classification des STU par l'usage de nouvelles représentations dont la dérivée et la somme cumulée. Cette méthode permet de montrer elle aussi l'intérêt de ces deux transformations, au travers de résultats expérimentaux.

Dans le chapitre suivant, les résultats expérimentaux de tests réalisés en fonction de la nature de la série : Dérivée, Originale et Somme cumulée, confirment l'intérêt de cette définition renforcée des STM.

Les M-histogrammes permettent donc de réduire efficacement les séries temporelles à condition d'utiliser cette représentation sur des STM renforcées. Il faut maintenant déterminer comment exploiter la représentation, c'est-à-dire quelle est la valeur du M dans le M-histogramme ?

3.2.3 Bi-histogramme : M-histogramme pour M=2

Le paramètre majeur à régler dans l'application des M-histogrammes est le choix du nombre M de dimensions. Plusieurs versions du M-histogramme applicatives ont été réalisées et les résultats sont présents au prochain chapitre. Néanmoins, la théorie qui est présentée ici, montre qu'il faut privilégier la transformation des séries en plusieurs bi-histogrammes et histogramme, plutôt qu'en un seul et unique M-histogramme de toutes les dimensions.

L'argument principal est celui de **complexité spatiale de la méthode**. En effet, la taille d'un M-histogramme va dépendre en grande partie de la valeur de M le décrivant. Comme définie dans le chapitre d'introduction, la taille de l'objet final est égale à $|\prod_i^M b_i|$. Soit le produit du nombre d'intervalles pour chaque dimension du M-histogramme. Afin de réduire de manière conséquente la taille de la série, il vaut mieux plusieurs petits M-histogrammes, qu'un seul contenant toutes les dimensions.

Exemple de taille

Soit X une STM de 4 dimensions de taille 100 points, et soit le M-histogramme qui la représente S de 4 dimensions de taille 10 et l'ensemble E de 3 bi-histogrammes et 1 histogramme de 2 et 1 dimensions de taille 10, nous avons :

$$|X| = 4 * 100 = 400$$

$$|S| = 10^4 = 10000$$

$$|E| = 3 * (10^2) + 10 = 310$$

L'exemple ci-dessus montre que prendre S avec le même nombre de dimensions que X peut créer rapidement de très gros objets, car la taille est exponentielle par rapport au nombre de dimensions. Cet objet peut donc rapidement être impossible à stocker. Néanmoins, il montre aussi qu'il est toujours possible d'extraire des sous-ensembles de données afin d'obtenir des bi-histogrammes ou histogrammes qui ensemble diminuent toujours la taille originale de la série.

La méthode proposée exploite donc plusieurs transformations en bi-histogramme et histogramme sur les données. Cela se rapproche en réalité d'un concept existant qui s'appelle **l'apprentissage multi-vues**.

L'étape de la transformation des dimensions des séries temporelles en M-histogramme vient d'être définie, tout comme la nécessité de calculer les dérivées et les sommes cumulées des données. Cette nouvelle définition des données nécessite l'usage d'un apprentissage multi-vues.

3.3 Apprentissage multi-vues

Le concept de vue s'intègre parfaitement et à différents degrés d'analyse, à l'exploitation des séries temporelles. Il est nécessaire de commencer par définir une vue.

Une vue est un ensemble de dimensions de nature homogène. Cependant, cette définition n'est pas mathématique au sens propre et la nature d'une donnée désigne un groupe d'appartenance **partageant des propriétés en commun** comme un capteur de mesure par exemple, ou **une transformation mathématique**.

Trois ensembles de nature homogène peuvent être définis ici et pour lesquels les algorithmes des bi-histogrammes où M=2 et de l'histogramme où M=1 peuvent être ensuite appliqués. Ces trois groupes sont donc trois vues. En effet, la nouvelle définition des STM renforcées donne trois vues des données que sont les données originales, leurs dérivées et sommes cumulées. Si le diagramme de la méthode 3.1 est détaillé, il

montre l'imbrication des vues et des représentations par les M-histogrammes, Fig. 3.2. Les STM d'origine, leurs dérivées et leurs sommes cumulées sont donc traitées chacune à part.

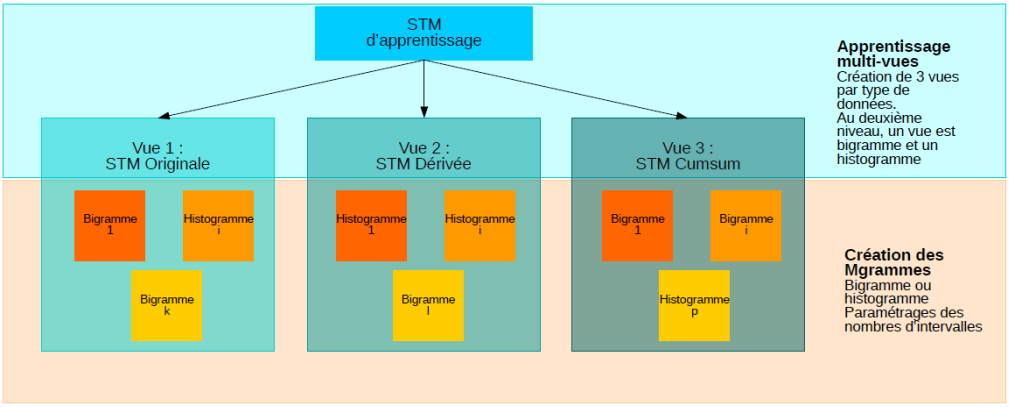


FIGURE 3.2 – Ensemble d'apprentissage comptant minimum trois vues

Il est à noter que la définition d'une vue n'est pas formelle, elle sous-entend néanmoins l'extraction de plusieurs sous-groupes de données afin de mieux en exploiter la relation sous-jacente. C'est pourquoi il peut aussi être considéré que chaque bi-histogramme et histogramme sont de nouvelles représentations de vues plus petites, permettant de mieux appréhender les **interactions inter dimensions** pour l'un et **intra-dimension** pour l'autre. La définition est donnée comme suit :

Définition : Vue d'une série multivariée

Soit X une STM de M dimensions, V est une vue à M' dimensions de X si :

— $V \in X$ et $M' <= M$

— Les M' dimensions de V sont de nature homogène

— V exploite une relation inter ou intra dimension

La méthode exploite donc trois vues principales des données correspondant aux opérations effectuées décrites avant. Seulement un deuxième niveau de vues peut aussi être défini. Celui-ci est constitué des bi-histogrammes et des histogrammes qui mettent eux aussi en avant des relations contenues dans les dimensions.

3.3.1 Relation intra et inter temporelle

Tout d'abord, l'**histogramme** qui permet de **mettre en avant la relation inter temporelle des points**. En effet, une série temporelle est elle-même composée d'un ensemble d'attributs (points) homogènes. Chaque dimension temporelle peut donc être considérée comme une **vue d'un phénomène mesuré**. Et l'adaptation en histogramme est simplement le changement de représentation de cette vue. De même le bi-histogramme permet de **mettre en exergue les interactions entre dimensions** et donc est aussi la représentation d'une vue constituée de deux dimensions homogènes.

La méthode se base donc sur **l'imbrication de deux niveaux de vues**. Seulement, l'ensemble de celles-ci peut alourdir le temps de traitement de la méthode c'est pourquoi une étape de nettoyage est réalisée.

3.3.2 Analyse de corrélation

Pour améliorer le traitement de ces vues, c'est-à-dire le temps de calcul et de stockage, et éviter toutes redondances dans les vues, une analyse des corrélations entre dimensions est réalisée.

Définition : Corrélation intra STM

Soit X une STM de M dimensions et de longueur T $\in \tau$, telle que $X = (X_1, ..., X_M)$ avec X_m une STU de longueur T où $m \in M$. Nous avons C(X), la matrice des corrélations entre dimensions de X telle que :

$$C(X) = \begin{pmatrix} Corr(X_1,X_1) & ... & Corr(X_1,X_m) & ... & Corr(X_1,X_M) \\ ... & ... & ... & ... & ... \\ Corr(X_m,X_1) & ... & Corr(X_m,X_m) & ... & Corr(X_m,X_M) \\ ... & ... & ... & ... & ... \\ Corr(X_M,X_1) & ... & Corr(X_M,X_m) & ... & Corr(X_M,X_M) \end{pmatrix} \tag{3.2}$$

Dans la mesure où la prédiction finale est basée sur un vote majoritaire, voir Fig. 3.1, tous les M-histogrammes qui représentent une information discriminante pour les classes, doivent avoir le même poids. En effet, si deux M-histogrammes contiennent une information similaire, alors celle-ci a deux fois plus de poids dans la méthode que les autres informations. L'étude de la corrélation entre séries permet de supprimer ces doublons.

L'utilisation d'un seuil déterminé par l'utilisateur permet de supprimer l'une des deux séries dont la corrélation dépasse le dit seuil, voir l'algorithme 2.

Algorithme : Réduction des dimensions d'une STM par corrélation

Data : X une STM de M dimensions, a : un seuil déterminé par l'utilisateur tel que $a \in [0,1]$
Result : X' une STM de M' dimensions tel que $M' \leq M$

```
1 X' ← X ;
2 corr_X ← Corrélation(X) ;
3 for row ∈ Triangle(corr_X) do
4     if abs(corr_X(row)) contains upper a then
5         X' ← X'.drop(X_row);
```

Preuve d'Algorithme : Réduction des dimensions d'une STM par corrélation

Terminaison
L'algorithme termine forcément car l'opération C(X) est finie ainsi que la boucle for.

Correction
De même par la déclaration X', X' est une STM non nulle de taille identique ou inférieure.

Complétude
De même si X est un STM et $a \in [0,1]$ alors il aura une sortie de taille identique ou inférieure.

Cette analyse des corrélations permet donc l'obtention d'une nouvelle série temporelle pour laquelle les dimensions ont toutes le même poids. Il est à clarifier ici que dans l'exécution de la méthode, cette étape est réalisée avant le calcul des dérivées et sommes cumulées afin d'éviter la redondance de calculs inutiles.

La méthode commence donc par supprimer les dimensions contenant des informations similaires. Les dimensions restantes donnent lieu aux calculs de dérivées et de sommes cumulées. Une STM qui est découpée en trois vues est donc obtenue. Au sein de ces vues se trouvent encore plusieurs dimensions. Un choix s'impose alors sur les dimensions à représenter et leur forme : bi-histogramme ou histogramme.

3.3.3 Choix des dimensions dans la représentation

À cette étape, la méthode dispose de séries temporelles de M' dimensions où M' est au moins égale à 3 : Une dimension par type (Originale, Dérivée, Somme cumulée). Le cas où il n'y a effectivement que trois dimensions est rare : soit l'auteur a choisi un seuil de corrélation trop bas, supprimant au passage toutes les dimensions de la STM, soit les dimensions de la série contenaient en réalité toute la même information. Dans ce cas, aucun choix n'est nécessaire, la méthode ne peut construire que des histogrammes.

En général, la méthode doit gérer des séries de plus de 6 dimensions, soit au moins deux par vue. Dans ce cas, plusieurs choix se posent entre construction d'histogrammes et de bi-histogrammes, et pour quelles dimensions.

Dans cette méthode, le choix fait est, de sélectionner les dimensions à représenter **de manière aléatoire** comme pour les forêts aléatoires BREIMAN [2001]. Originalement, les attributs utilisés pour construire les arbres peuvent être tirés de manière aléatoire comme lors du bagging. C'est une stratégie qui donne de très bons résultats, qui est aussi privilégiée par BAYDOGAN et RUNGER [2015] et cela s'explique facilement.

Les auteurs de BERGSTRA et BENGIO [2012] ont montré que l'hyperparamétrisation est plus efficiente si elle est aléatoire que faite à la main ou par recherche sur grille. En effet, il est visible sur la Fig. 3.3 que si la méthode se contente d'énumérer une partie des possibilités alors la méthode passe toujours à coté de l'optimum (pique vert sur la

figure), mais une recherche exhaustive est particulièrement longue à exécuter. De plus, une sélection des dimensions à la main exige la mise à disposition d'un expert capable de dire quelles sont les dimensions à garder et celles à écarter.

Ce choix est aussi privilégié dans d'autres domaines de la classification supervisée comme la classification multi-label READ et collab. [2009]. Dans ce domaine de classification, de nombreux sous-problèmes sont extraits à partir des combinaisons possibles entre les labels multiples. Ce nombre de sous-problèmes peut rapidement exploser et c'est pourquoi le choix de l'aléatoire a aussi été effectué, afin de combiner les labels et cela donne de très bons résultats.

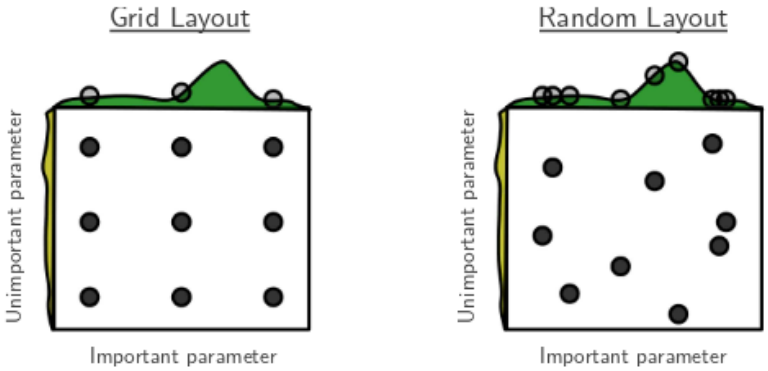


FIGURE 3.3 – Illustration issue de BERGSTRA et BENGIO [2012] qui montre l'intérêt d'employer une recherche aléatoire plutôt qu'une recherche sur grille afin de trouver l'optimum sur la courbe en vert de g(x) où f(x) est en réalité une fonction de bruit négligeable.

Une méthode de sélection aléatoire est donc appliquée ici pour la recherche des dimensions à extraire et à représenter dans la méthode finale. Pour le bi-histogramme, le nombre de combinaisons possible est de 2 parmi les M'' dimensions de chaque vue. Ce M'' étant parfois très grand, un nouveau paramètre appelé K est défini tel qu'il limite le nombre de couples de dimensions à étudier. K est donc le pourcentage de M-histogrammes que nous voulons calculer par rapport à l'ensemble des possibles. La même politique est appliquée pour l'histogramme, voir Algorithme 3.

Définition : Tirage aléatoire des dimensions à l'usage des M-histogrammes

Soit X une STM renforcée de M'' dimensions non corrélées pour chacune des 3 vues, soumis à deux représentations de M-histogramme à M=2 et M=1. Le tirage aléatoire est représenté par le nombre de possibilités suivantes :

$$\binom{M''}{2} \times 2 \times 3 \times K$$

où $\binom{M''}{2}$ est le nombre de choix possibles de 2 dimensions à représenter, 2 le choix du M=1 ou 2, 3 les vues possibles et $K \in [0,1]$ est ici le pourcentage de l'ensemble des possibilités à garder.

La répétition sur les données d'apprentissage, pour régler les hyperparamètres ci-dessus permet de déterminer la configuration à garder.

Algorithme : Création d'une vue

Data : X une STM composée de trois vues de M'' dimensions tel que $M'' \geq 2$, N_v, un noyau aléatoire pour choisir les vues, N_r : un noyau pour la représentation, N_d : un noyau pour les dimensions, B_I : le nombre d'intervalles de la représentation
Result : S un M-histogramme

```
1 v ← N_v(0,2) ;
2 r ← N_r(0,1) ;
3 D ← (N_d(0,M''), N_d(0,M'')) ;
4 while D[1] == D[2] do
5 |   D ← [N_d(0,M''), N_d(0,M'')] ;
6 if r==0 then
7 |   S ←Création d'un Histogramme(X(v,D),B_I);
8 else
9 |   S ← Création d'un bi-histogramme(X(v,D),B_I);
```

Preuve d'Algorithme : Création d'une vue

Terminaison
Si la pré-condition de $M'' \geq 2$ est respectée alors il y a toujours deux dimensions différentes par vue et donc la boucle while finie.

Correction
Nous avons montré avant que l'algorithme de création de M-histogramme est correct. Donc celui-ci, qui l'emploie pour le résultat, l'est aussi.

Complétude
Si la condition initiale sur M'' est respectée, alors il y a aura toujours une réponse de l'algorithme.

Avec ces deux étapes, la méthode crée donc plusieurs M-histogrammes qui vont permettre de classifier les données. Ceci est la prochaine étape abordée et réalisée par un classifieur ensembliste.

3.4 Classification ensembliste

Lorsque la méthode crée plusieurs M-histogrammes, il faut mettre en place un système afin de les classifier et d'agréger leurs prédictions.

3.4.1 Vote majoritaire

L'ensemble de M-histogrammes est utilisé afin d'entraîner des classifieurs spécialisés pour chacune de ces représentations. Tel que décrit, la méthode mise en place s'approche finalement du *bagging*. C'est pourquoi, un vote majoritaire est utilisé pour agréger les prédictions. En effet, c'est le vote mis en place dans ce type de modèle.

Définition : Vote majoritaire

Soit P l'ensemble des prédictions émises grâce aux M-histogrammes pour une série temporelle testée. On a $P = [p_1, ...p_p, ..., p_P]$ avec $p_p \in C$ où $C = [c_1, ..., c_c]$ est l'ensemble des classes. Nous avons p_f la prédiction finale telle que :

$$p_f = Argmax(P)$$

et $p_f \in C$

De plus, un vote pondéré aurait supposé l'accès à des experts des domaines d'activités des données afin de définir avec eux les vues méritant une meilleure prise en compte et celles considérées comme négligeables.

De plus, c'est aussi le vote à la complexité la plus faible $O(|P| \times |C|)$ avec $|P|$ le nombre de prédictions et $|C|$ le nombre de classes, voir Algorithme 4 et qui donne de très bons résultats. Ce choix a d'ailleurs été fait dans l'algorithme COTE BAGNALL et collab. [2015].

Algorithme : Vote majoritaire

Data : $P=(p_1,....p_P)$: Vecteur de prédictions avec $p_i \in C$, C l'ensemble des classes
Result : p_f une prédiction finale$\in C$

```
count_C ← Zero(P);
for p_i ∈ P do
    j ← 0;
    while p_i ∉ c_j do
        j ← j + 1;
    count_C[j] ← count_C[j] + 1;
p_f ← C[Index(max(Count_c))] ;
```

Preuve d'Algorithme : Vote majoritaire

Terminaison
La terminaison dépend de la boucle while. Or, si P est bien constitué d'éléments appartenant à l'ensemble des classes alors il termine.

Correction
P_f sera bien une classe appartenant à C donc une solution du problème.

Complétude
Si la pré-condition est respectée alors il aura toujours une solution. S'il y a des égalités dans $count_C$, max retourne la première classe majoritaire.

Il est à préciser que la comparaison entre vote majoritaire et vote pondéré par hyperparamétrisation aléatoire est réalisée dans le chapitre 4 sur des données de référence.

Le choix d'un vote majoritaire est donc fait ici. Il faut maintenant définir le type de classifieur employé.

3.4.2 Classifieur 1NN

Les classifieurs choisis pour être entraînés dans la méthode sont des classifieurs 1NN. Ce choix est fait pour plusieurs raisons.

Algorithme : 1NN

Data : D un ensemble de M-histogramme d'apprentissage tel que
D = $((S_1, c_i), ..., (S_s, c_1))$ où $c_i \in C$ avec C l'ensemble des classes,
S_test, le M-histogramme à tester
Result : p une prédiction $\in$ C

```
dist_min ← Inf ;
for S_s ∈ D do
    dist ← Euclid(S_s, S_test);
    if dist ≤ dist_min then
        p = c_s ;
```

Preuve d'Algorithme : 1NN

Terminaison
L'ensemble de l'algorithme est encapsulé dans une boucle for donc fini.

Correction
P est obligatoirement dans l'espace de C donc est une solution du problème.

Complétude
Il y aura une réponse à condition que la distance calculée à chaque étape soit en dessus de l'initialisation de la distance minimum, au moins une fois. C'est pourquoi l'initialisation est volontairement très grande, ici infinie.

Tout d'abord, l'algorithme 1NN, voir Alg. 5 dispose d'une complexité plus faible que les autres algorithmes de classification supervisée, car celui-ci ne demande pas d'apprentissage. En effet, cet algorithme n'a pas de paramètres à régler et consiste en réalité en un simple test de similarité. Dans la méthode, la distance Euclidienne est employée, car la représentation en M-histogramme s'affranchit du temps. Il n'y a donc pas besoin d'appliquer DTW qui est, par ailleurs, un algorithme glouton.

De plus, ce classifieur qui ne nécessite pas d'apprentissage, n'influence pas les résultats de la méthode qui dépendent déjà d'autres paramètres. Il permet alors de déterminer l'impact réel des paramètres de la méthode sur les performances de celle-ci.

Enfin, il est largement admis par la communauté que la distance DTW combinée avec 1NN donne de très bons résultats sur les STU, PETITJEAN et collab. [2016]. Le choix du classifieur 1NN est de ce fait aussi employé dans plusieurs autres algorithmes du domaine comme SAX présenté chapitre 2 ou COTE BAGNALL et collab. [2015]. En réalité, une équipe a comparé plusieurs classifieurs entre eux et plusieurs versions de K plus proches voisins (KNN) pour différentes valeurs de K BAGNALL et LINES [2014]. La réponse est que k=1 est la meilleure solution. Pour les mêmes raisons que citées ci-avant. De même, dans le cadre de cette thèse, des tests similaires ont été réalisé, mais en utilisant des STM. Les mêmes conclusions sont obtenues, voir Chapitre 4.

Pour toutes ces raisons, des classifieurs 1NN sont appliqués dans la méthode finale.

Pour résumé, une analyse de corrélation des dimensions des STM est réalisée afin de supprimer les dimensions redondantes. Les dérivées et sommes cumulées des dimensions restantes sont ensuite calculées donnant naissance à trois vues de données. Ensuite, sont tirées aléatoirement plusieurs dimensions par vue afin de les représenter sous forme de M-histogrammes avec M=1 ou 2. Enfin, des classifieurs 1NN sont appliqués sur les M-histogrammes afin d'émettre des prédictions pour chaque représentation. Ces prédictions sont ensuite agrégées via un vote majoritaire. Il reste maintenant à aborder le système d'apprentissage, conduisant à la prédiction finale.

3.5 Prédiction finale

Afin de régler les différents paramètres dans le but d'émettre la meilleure prédiction possible, deux étapes sont réalisées, l'apprentissage puis le test. Il est à expliciter que l'apprentissage se fait par validation croisée des paramètres comme abordé dans le chapitre d'introduction.

3.5.1 Apprentissage du M-histogramme

Tout d'abord, le premier paramètre à régler est le **nombre d'intervalles** à prendre en considération par dimension du M-histogramme. **Ce paramétrage se fait par M-histogramme et non pour l'ensemble des M-histogrammes**. Le nombre d'intervalles maximum à tester est choisi par l'utilisateur. Toutefois, les tests ont montré que construire des objets de 10 intervalles par dimension pour un bi-histogramme et de 50 intervalles pour un histogramme permettent l'obtention de bonnes performances, voir Chapitre 4.
En outre, rien n'indique que la même **granularité** doit être respectée pour chaque dimension du M-histogramme. Comme dans l'illustration 3.3, une dimension peut contenir l'information discriminante et l'autre non. Il peut alors être intéressant d'avoir une granularité plus fine pour la première dimension que pour m-ième. C'est pourquoi le M-histogramme n'est pas obligatoirement carré, c'est-à-dire n'a pas le même nombre d'intervalles par dimensions, voir Alg. 6.

Algorithme : Apprentissage d'un M-histogramme

Data : D = Ensemble(X,C), un jeu de donnée composée de STM associée avec C l'ensemble des classes, B : l'ensemble des nombres d'intervalles en réglages possibles, Metric : la fonction d'évaluation de l'apprentissage, cross : Le nombre de validation croisées désirées

Result : S_f : un ensemble de M-histogramme, b_f : le nombre d'intervalles par dimension

```
1  eval_best ← 0;
2  for b_i ∈ B do
3      S ← [];
4      for X_j in D do
5          s_j ← Création d'un M-histogramme(X_j, b_i);
6          S.append(s_j);
7      P ← [];
8      for crossval ∈ cross do
9          S' ← N_s(crossval);
10         p_i ← 1NN(([S − S'], C], S') ;
11         P.append(p_i);
12     eval ← Evaluation(P, C, Metric) ;
13     if eval ≥ eval_best then
14         eval_best ← eval;
15         b_f ← b_i ;
16         S_f ← S ;
```

Pour chaque vecteur b_i d'intervalles, il faut transformer l'ensemble D de STM d'apprentissage. Ensuite, S′ M-histogrammes sont tirés aléatoirement parmi S, de manière à réaliser $cross$ validations. Chaque apprentissage sur S − S′ donne lieu à une prédiction pour S′. La qualité des prédictions est ensuite évaluée par une fonction choisie par l'utilisateur. Les paramètres et les M-histogrammes stockés sont ceux qui ont reçu la meilleure évaluation. Par défaut, ici, l'évaluation est le taux de bonnes prédictions réalisées par l'algorithme.

Preuve d'Algorithme : Apprentissage d'un M-histogramme

Terminaison
L'algorithme finit, car il est encapsulé dans une boucle finie.

Correction
La solution sera toujours correcte, car l'algorithme *Création de M-histogramme* l'est et, car b_i appartient à l'ensemble de départ définit.

Complétude
L'apprentissage donnera une solution à condition que :

— b est bien un ensemble de vecteurs d'intervalles.

— Metric est une fonction d'évaluation cohérente, c'est-à-dire permettant l'évaluation d'algorithmes de classification.

— cross est un nombre valide de découpage, c'est-à-dire qu'il reste au moins deux exemples de chaque classe dans $S - S'$.

La complexité de la méthode dépend donc en partie du type de M-histogrammes mis en place, car celui-ci fait varier la taille de la boucle $b_i \in B$. La complexité de transformation d'un jeu de données D en M-histogramme est, d'après l'algorithme précédent, de $O(|D|T\sum(b_i))$. Tandis que la complexité de l'apprentissage en validation croisée est de $O(cross|D|b_i)$. Donc, finalement, la complexité de l'algorithme est définie par $O(B(|D|T\sum(b_i) + cross|D|b_i))$. Le terme influent ici est B et toute la complexité de l'apprentissage dépend en réalité de la taille de B et de son nombre de dimensions $|b_i|$, comme expliqué dans la première partie de ce chapitre.

Définition : Complexité simplifiée de l'apprentissage de bi-histogramme et histogramme

— Complexité d'apprentissage d'un bi-histogramme rectangle : $O(|D|T \times 2B)$

— Complexité d'apprentissage d'un bi-histogramme carré : $O(|D|T \times B)$

— Complexité d'apprentissage d'un histogramme : $O(|D|T \times B)$

La définition 16 montre qu'il peut y avoir un réel intérêt à définir le même nombre d'intervalles pour un bi-histogramme au regard de la complexité. En effet, cela permet de s'affranchir de B opérations.

3.5.2 Apprentissage des vues

Il reste à expliciter l'apprentissage des vues. Sur le même principe que la création de M-histogramme, le processus de création des vues est réitéré plusieurs fois, afin de déterminer quelles sont les vues qui permettent la meilleure classification. Cet apprentissage est toujours réalisé en validation croisée, voir Alg. 7.

Il est rappelé que la définition d'une vue a été donnée sur deux niveaux. Les trois vues principales ne nécessitent pas d'apprentissage seulement leurs calculs. Puis les vues qui permettent soit de mettre en avant une relation inter-dimension par l'usage

de bi-histogramme, soit une relation intra-dimension par l'usage des histogrammes. C'est ce qu'il faut apprendre ici. **Faut-il faire l'une ou l'autre représentation et pour combien et quelles dimensions ?**

Algorithme : Apprentissage des vues

Data : D = Ensemble(X,C), un jeu de donnée composée de STM associée avec C l'ensemble des classes, K, le nombre de vues voulues, Metric : la fonction d'évaluation de l'apprentissage, cross : Le nombre de validations croisées désirées, App, le nombre de fois où il faut réitérer l'apprentissage des vues
Result : V_f : un ensemble de bi-histogramme et d'histogramme

```
1  eval_best ← 0;
2  for app ∈ App do
3      V ← [];
4      for k ∈ K do
5          v_k ← Création des vues pour D;
6          V.append(v_k);
7      P ← [];
8      for crossval ∈ cross do
9          V' ← N_v(crossval);
10         p_i ← 1NN(([V − V'], C], V') ;
11         P_i ← Vote Majoritaire(p_i);
12         P.append(P_i);
13     eval ← Evaluation(P, C, Metric) ;
14     if eval ≥ eval_best then
15         eval_best ← eval;
16         V_f ← V ;
```

Il est aussi rappelé que K est un pourcentage du nombre de combinaisons possibles défini précédemment dans la section *Choix des dimensions dans la représentation*. La méthode ne diffère pas réellement de l'apprentissage des M-histogrammes, car tout simplement la procédure est la même. La complexité ici est donc définie comme O(AppK|D|T2B)

Preuve d'Algorithme : Apprentissage d'une vue

Terminaison
L'algorithme finit, car il est encapsulé dans une boucle finie.

Correction
La solution sera toujours correcte, car l'algorithme *Création de vues* l'est.

Complétude
L'apprentissage donnera une solution à condition que :

— K et App soient des entiers

— Metric est une fonction d'évaluation cohérente, c'est-à-dire permettant l'évaluation d'algorithmes de classification

— cross est un nombre viable de découpage, c'est-à-dire qu'il reste au moins deux exemples de chaque classe dans $V - V'$

Dans l'exécution, l'apprentissage des M-histogrammes est donc encapsulé dans l'apprentissage des vues. Tout d'abord, k vues sont tirées. Puis pour chaque vue, le paramétrage de la représentation M-histogramme associée est appris. Ensuite, chaque M-histogramme permet de réaliser une prédiction. Finalement, l'ensemble des paramètres qui obtient la meilleure évaluation est gardé. Les paramètres stockés sont donc les noms des dimensions exploitées, leurs formes et le nombre d'intervalles dans la représentation.

3.5.3 Classification Finale

Finalement, la liste des paramètres appris est appliquée sur l'ensemble test afin d'appliquer les mêmes transformations. Puis, l'ensemble d'apprentissage est utilisé afin de réaliser une prédiction en test, voir Alg. 8.

Algorithme : Prédiction finale

Data : D_{app}, D_{test} : Ensemble(X,C), jeu de donnée composée de STM associées avec C l'ensemble des classes, K, le nombre de vues voulues, Metric : la fonction d'évaluation de l'apprentissage, cross : le nombre de validations croisées désirées, App, le nombre de fois où il faut réitérer, B : l'ensemble des nombres d'intervalles, s : le seuil d'études des corrélations
Result : P_{test} : les prédictions sur le jeu de test

1 $D' \leftarrow$ Réduction par étude la corrélation (D_{app}, s);
2 $D' \leftarrow$ Calcul des dérivées et sommes cumulées ;
3 param, $S_{app} \leftarrow$ Apprentissage des vues (D') ;
4 $S_{test} \leftarrow$ Création des M-histogrammes$(D_{test}, param)$;
5 $p_{test} \leftarrow$ 1NN (S_{app}, S_{test}) ;
6 $P_{test} \leftarrow$ Vote majoritaire(p_{test}) ;

Il est à noter que les performances en test peuvent changer suivant les vues captées en apprentissage. Sans le noyau permettant de ré-exécuter les mêmes tirages, lors de la prochaine exécution le résultat sera différent. C'est la limite de la paramétrisation par l'aléatoire. Ou alors il faut que l'utilisateur stocke la liste finale des paramètres de transformations. Dans ce cas, le résultat sera toujours identique.

Preuve d'Algorithme : Prédiction finale

Terminaison,Correction,Complétude
L'algorithme finale termine, est correct et complet, car il est composé de blocs algorithmiques qui terminent sont corrects et complets.

La complexité finale est définie dans le pire des cas comme : O((AppK|D$_{app}$|T2B) + |D$_{test}$|KT2B) c'est-à-dire qu'aucune dimension n'a été réduite lors de l'étude de corrélation et que toutes les vues sont des bi-histogrammes rectangles.

La méthode EMMV est donc un ensemble de M-histogrammes ou M=1 ou 2 représentant des dimensions non corrélées tirées de manière aléatoire. Ces représentations donnent lieu à des prédictions, par des classifieurs 1NN, ensuite agrégées par un vote majoritaire. Le paramétrage des M-histogrammes, c'est-à-dire le nombre d'intervalles par dimensions, ainsi que le nombre de vues à garder sont appris en validation croisée. Une prédiction en test est finalement réalisée. Afin de faciliter la compréhension de la méthode, un exemple illustratif est déroulé dans la prochaine section.

3.6 Illustration applicative

L'exemple présenté ici est relativement court et a pour objectif de visualiser simplement le fonctionnement de la méthode. Le lecteur pourra lui-même recalculer toutes les étapes intermédiaires amenant aux résultats. L'exemple suivant, une pure construction ne sert pas de références à l'évaluation de la méthode.

3.6.1 STM

Jeux de données

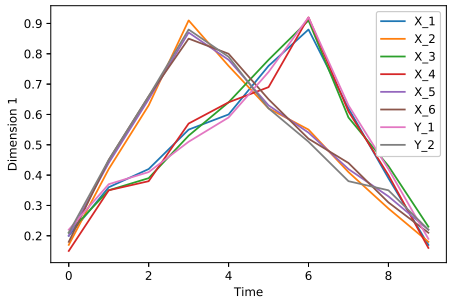
Soit D$_{app}$ un jeu de donnée d'apprentissage et D$_{test}$ un jeu de test, composés de STM tel que

$$D_{app} = [X_1;0], [X_2;1], [X_3;0], [X_4;0], [X_5;1], [X_6;1]$$

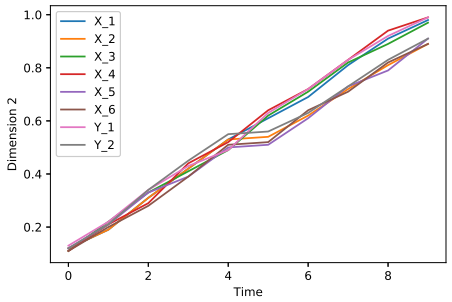
$$D_{test} = [Y_1;0], [Y_2;1]$$

où le dernier élement est la classe de la série définit dans l'ensemble C = 0, 1.

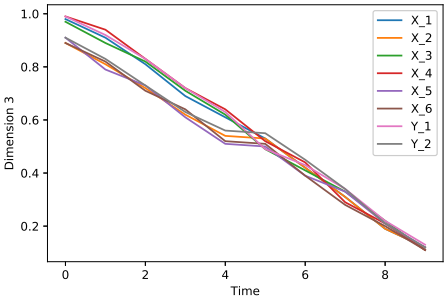
Soit un jeu d'apprentissage composé de six séries avec trois séries représentant chaque classe, et un jeu de test composé de deux séries, un dans chaque classe, tel que sur la Fig. 3.4. Les données à l'origine des illustrations peuvent être trouvées en annexe de cette thèse.



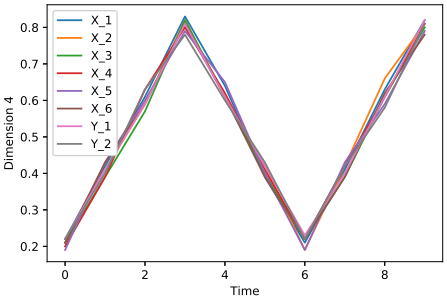
(a) Dimension n°1 pour toutes les STM



(b) Dimension n°2 pour toutes les STM



(c) Dimension n°3 pour toutes les STM



(d) Dimension n°4 pour toutes les STM

FIGURE 3.4 – Affichage de X_1 à X_6, Y_1, Y_2 par dimension

3.6.2 Réduction des dimensions par étude des corrélations

Tout d'abord, l'algorithme de réduction des dimensions, avec un seuil de corrélation fixé à 0.7, est appliqué et la matrice des corrélations suivante est obtenue :

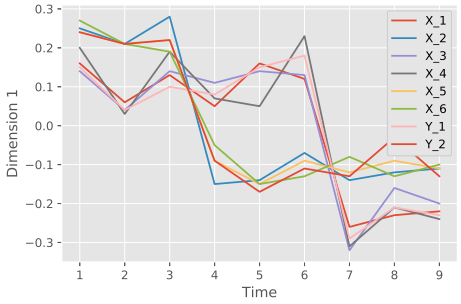
Dimensions	Dim_1	Dim_2	Dim_3	Dim_4
Dim_1	1.0	0.014	0.021	0.05
Dim_2	0.014	1.0	0.997	0.345
Dim_3	0.021	0.997	1.00	0.345
Dim_4	0.005	0.345	0.345	1.0

TABLEAU 3.1 – Moyennes des corrélations pour toutes les séries de chaque dimension deux à deux

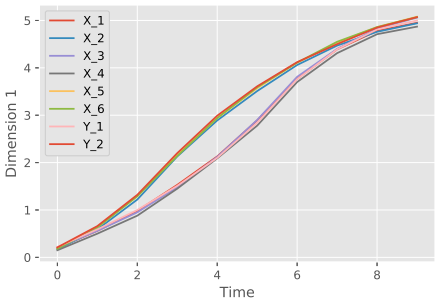
La corrélation a été construite par la méthode de Pearson LEE RODGERS et NICEWANDER [1988]. Par ailleurs, la valeur absolue est ici affichée dans le tableau Tab. 3.1. Pour un seuil de 0.7, deux attributs correspondent à la définition de forte corrélation. Seul le premier, par ordre d'apparition dans le tableau, est gardé.

3.6.3 Calcul des dérivées et sommes cumulées

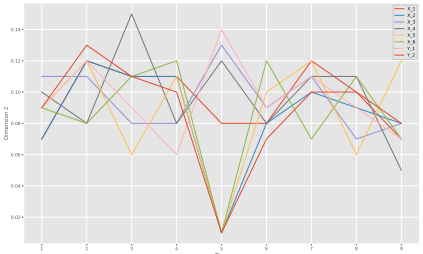
Les dérivées et sommes cumulées sont ensuite calculées sur les dimensions restantes, soit Dim_1, Dim_2, Dim_4. Les résultats sont observables dans la Fig. 3.5



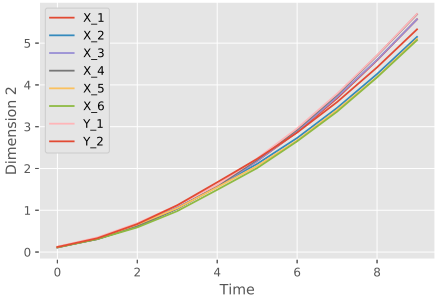
(a) Dérivée de la dimension n°1 pour toutes les STM



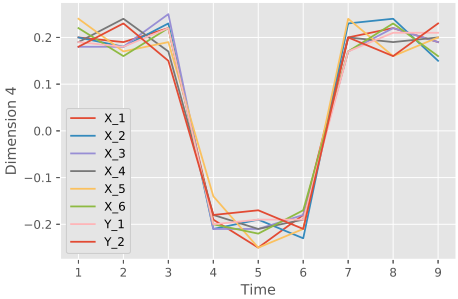
(b) Somme cumulée de la dimension n°1 pour toutes les STM



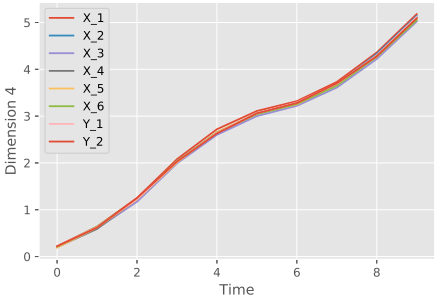
(c) Dérivée de la dimension n°2 pour toutes les STM



(d) Somme cumulée de la dimension n°2 pour toutes les STM



(e) Dérivée de la dimension n°4 pour toutes les STM



(f) Somme cumulée de la dimension n°4 pour toutes les STM

FIGURE 3.5 – Affichage des dérivées et sommes cumulées de X_1 à X_6, Y_1, Y_2 par dimension

Finalement, il y a donc 9 dimensions par série séparées en trois vues principales : origine, dérivée et sommes cumulée.

3.6.4 Apprentissage

Soit le processus d'apprentissage des vues de la méthode qui est réalisé deux fois. Lors des deux répétitions, un nombre K de M-histogrammes sont calculés pour réaliser la prédiction. K est généralement un pourcentage du nombre de combinaisons possibles des paramètres. La définition est donnée ci-avant, il y a ici 18 combinaisons possibles. K est choisi pour calculer uniquement 25 % de ces possibilités. Il y a donc 4 vues par itération.

N°1 : Initialisation

Lors du premier apprentissage, les vues suivantes sont obtenues :

— Vue N°1 : Un bi-histogramme des sommes cumulées des dimensions 1 et 4 de taille 2×2 Fig. 3.6 et Fig. 3.7

— Vue N°2 : Deux histogrammes des sommes cumulées des dimensions 2 et 4 de tailles 2 et 15

— Vue N°3 : Deux histogrammes des dimensions 2 et 4 de tailles 2 et 3

— Vue N°4 : Deux histogrammes des dérivées des dimensions 2 et 4 de tailles 2 et 8

Il est à rappeler que la taille des M-histogrammes est apprise ici en validation croisée.

Ensuite, La distance pour chaque M-histogramme entre ceux en apprentissage et ceux en validation est calculée. Puis la classe déterminée est celle dont l'individu d'apprentissage à la distance la plus courte de l'individu de validation. Le taux de bonnes classifications est ensuite calculé pour chaque vue, les résultats de classification sont les suivants :

— Vue N°1 : 100%

— Vue N°2 : 100% et 100%

— Vue N°3 : 100% et 100%

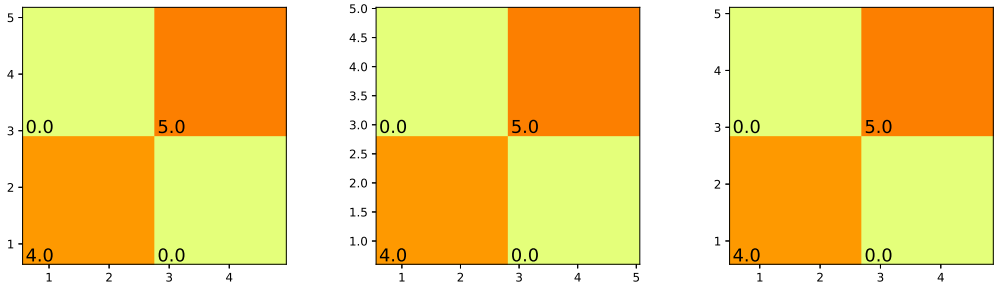
— Vue N°4 : 100% et 100%

Pour une prédiction par vote majoritaire de : 100 %

Les représentations des séries d'apprentissages selon la **Vue N°1** par classe sont affichées sur la figure 3.6 :

Les bi-histogrammes sont donc des matrices ici carrées de taille 2×2. Grâce à l'algorithme du plus proche voisin, la série qui à la représentation la plus proche et par extension sa classe est trouvée. De plus, le fait que les bi-histogrammes soient parfaitement identiques voir Figs. 3.6 et 3.7 explique pourquoi le score de réussite est ici de 100%.

En annexe sont joints les visuels de chaque vue pour chaque série.

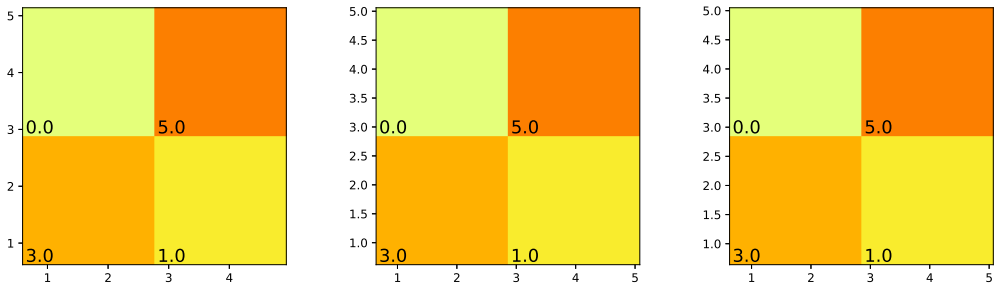


(a) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série N°1

(b) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série N°3

(c) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série N°4

FIGURE 3.6 – Affichage de la vue N°1 pour chaque STM appartenant à la classe 0



(a) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série N°2

(b) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série N°5

(c) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série N°6

FIGURE 3.7 – Affichage de la vue N°1 pour chaque STM appartenant à la classe 1

N°5 : Final

Lors du deuxième apprentissage, les vues obtenues sont les suivantes :

— Vue N°1 : Deux histogrammes des dérivées des dimension 2 et 4 de tailles 8 et 2
— Vue N°2 : Deux histogrammes des sommes cumulées des dimensions 1 et 4 de tailles 2 et 15
— Vue N°3 : Un bi-histogramme des sommes cumulées des dimensions 1 et 2 de taille 2×2
— Vue N°4 : Un bi-histogramme des sommes cumulées des dimensions 2 et 4 de taille 2×2

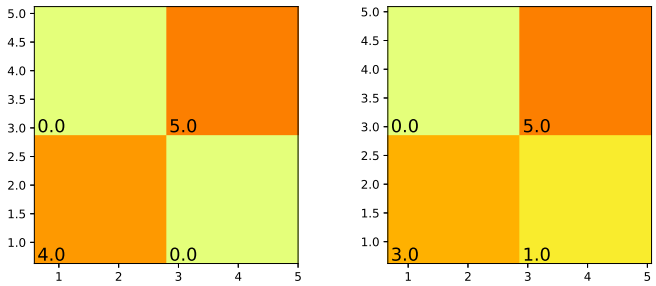
Avec ces vues, les taux de classifications sont :

— Vue N°1 : 100% et 100%
— Vue N°2 : 100% et 100%
— Vue N°3 : 100%
— Vue N°4 : 100%

Pour une prédiction par vote majoritaire de : 100 %

3.6.5 Prédiction finale

Finalement, l'exemple ici permet de garder les vues et les paramètres obtenus qu'importe l'itération. Les paramètres de transformation appliqués sont donc, par exemple, ceux de la première itération aux données tests. Fig. 3.8, illustre la vue N°1 des STM test par classe. Encore une fois les bigrammes sont identiques par classe, le taux de classification obtenu est donc de **100 %**.



(a) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série test N°1

(b) bi-histogramme des sommes cumulées des dimensions 1 et 2 de la Série test N°2

FIGURE 3.8 – Affichage de la vue N°1 pour chaque STM test

Une illustration de la méthode et de son déroulement viennent donc d'être présentés. Il reste à aborder les derniers points induits par la construction de la méthode.

3.7 Construction finale

Avant de récapituler le fonctionnement de la méthode, il faut aborder la problématique de la normalisation des données et des M-histogrammes. La normalisation est souvent une étape automatique et triviale. Mais elle revête ici une importance particulière, c'est pourquoi elle est détaillée.

3.7.1 Normalisation

Tout d'abord la normalisation des données est mise en place comme dans les modèles SMTS et WEASEL+MUSE.

Normalisation des STM

Cette étape n'a pas été discutée précédemment, car elle est considérée comme triviale et allant de soi. Les jeux de données de références sont très souvent normalisés, mais il faut préciser ici que l'exécution de la méthode nécessite l'utilisation d'une norme min-max afin que les M-histogrammes soient bornés. En effet, si une z-normalisation est réalisée alors les min et max de chaque série ne sont pas identiques et donc il faut en tenir compte dans la construction des intervalles des bi-histogrammes. Le simple fait de choisir une normalisation min-max permet de réduire ces considérations sans entacher les performances de la méthode car toutes les séries sont bornées par les valeurs 0 et 1.

Normalisation des M-histogrammes

Enfin, la normalisation des M-histogrammes permet de comparer des séries de tailles variables. En effet, sans normalisation du M-histogramme alors la matrice contient des fréquences de comptage d'apparition de points par intervalles de valeurs, soit des valeurs entre 0 si aucun point n'est tombé dans l'intervalle et T si tous les points sont tombés dans le même. Donc la normalisation des M-histogrammes permet leurs transformations en matrices des fonctions de densité où chaque cellule contient un pourcentage d'apparition de points par intervalle, soit la probabilité qu'un point tombe dans cette cellule.

Le point sur la normalisation étant fait, la méthode, créée et présentée dans ce chapitre, peut être résumé.

3.7.2 Récapitulatif

Cette section rappelle les caractéristiques de la méthode, ses avantages comme ses limites, mais aussi les paramètres à régler. Finalement, il est aussi comparé d'un point de vue théorique avec l'état de l'art.

Caractéristiques et capacités

En résumé, la méthode proposée permet le traitement rapide de STM à tailles variables. Pour se faire, il propose de réduire les dimensions à celles non-corrélées, afin que les poids des dimensions soient équilibrés, ce qui est indispensable à l'application d'un vote majoritaire.

Ensuite, il ajoute aux dimensions présentes leurs dérivées et sommes cumulées afin d'apporter des informations complémentaires sur les événements, les tendances et l'ordre des événements, non portés par les séries de base.

Ensuite, par groupe de nature homogène, il extraie aléatoirement deux dimensions afin de construire un bi-histogramme ou deux histogrammes. Cette étape permet de réduire la taille de très grandes séries et de mettre en exergue les relations intra et inter dimensions. Chaque nouvelle représentation permet alors une classification des séries par l'usage d'un classifieur 1NN puis via un système de vote, une prédiction finale est fournie.

Comme l'illustre l'exemple précédent, l'aspect visuel de la solution est très intuitive et permet humainement d'appréhender la solution très rapidement.

Paramètres à régler

Les paramètres principaux à régler sont donc le nombre d'itérations d'apprentissage, le nombre de M-histogrammes, les tailles des bi-histogrammes et histogrammes. Les paramètres secondaires qui peuvent être laissés par défaut sont les classifieurs, la méthode de normalisation, la fonction d'évaluation de la classification.

3.7.3 Comparaison avec l'état de l'art

Par rapport à l'état de l'art proposé Chapitre 2, la méthode proposée peut gérer les STM de tailles très variables. Elle permet aussi le traitement rapide de grands volumes de séries temporelles. Cela étant, il peut aussi appréhender de petits volumes de données de tailles fixes. De plus, cette méthode permet une représentation des STM compréhensible rapidement pour l'utilisateur et introduit les M-histogrammes dans le domaine des séries temporelles ainsi que leurs emplois dans le cadre de l'apprentissage multi-vues. Enfin, il exploite pour la première fois la complémentarité apportée par la somme cumulée.

Pour les points communs, il utilise le même concept de dérivée que la méthode SMTS et l'emploi de l'aléatoire afin d'extraire des dimensions de la série. De plus, l'emploi de bigramme se trouve aussi dans WEASEL+MUSE bien que l'espace d'application ne soit pas le même. Enfin, le concept de multi-vues est introduit aussi dans l'état de l'art, bien qu'ici, les vues homogènes soient créées après calcul de dimensions supplémentaires. Le concept est aussi exploité à deux niveaux puisque que le tirage aléatoire des dimensions crée de nouvelles vues homogène dans le but d'extraire la relation intra ou inter dimensions qui les lie.

Points clefs du chapitre Modèle

Nouveautés de la méthode	— Gestion des STM avec grandes variabilités de longueurs — Complexité plus faible que l'existant — Représentation visuelle sous forme d'image — Usage des M-histogramme — Apprentissage multi-vues : Originale/Dérivée/Somme cumulée
Points communs avec l'état de l'art	— Hyperparamétrisage aléatoire — Utilisation de la dérivée

Dans la suite, les performances de la méthode sont présentées en utilisant ce dernier sur des données de références. Ces données vont permettre d'établir les limites de la méthode en terme d'influences des paramètres qui le composent, mais aussi par l'influence des données mêmes.

3.8 Références

BAGNALL, A. et J. LINES. 2014, «An experimental evaluation of nearest neighbour time series classification», *arXiv preprint arXiv :1406.4757*. 57

BAGNALL, A., J. LINES, J. HILLS et A. BOSTROM. 2015, «Time-series classification with cote : The collective of transformation-based ensembles», *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, nº 9, doi :10.1109/TKDE.2015.2416723, p. 2522–2535, ISSN 1041-4347. 55, 57

BAYDOGAN, M. G. et G. RUNGER. 2015, «Learning a symbolic representation for multivariate time series classification», *Data Mining and Knowledge Discovery*, vol. 29, nº 2, doi :10.1007/s10618-014-0349-y, p. 400–422, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-014-0349-y. 47, 52

BERGSTRA, J. et Y. BENGIO. 2012, «Random search for hyper-parameter optimization», *Journal of Machine Learning Research*, vol. 13, nº Feb, p. 281–305. 52, 53

BONDU, A., D. GAY, V. LEMAIRE, M. BOULLÉ et E. CERVENKA. 2019, «Fears : a feature and representation selection approach for time series classification», . 48

BREIMAN, L. 2001, «Random forests», *Machine learning*, vol. 45, nº 1, p. 5–32. 52

LEE RODGERS, J. et W. A. NICEWANDER. 1988, «Thirteen ways to look at the correlation coefficient», *The American Statistician*, vol. 42, nº 1, p. 59–66. 64

PETITJEAN, F., G. FORESTIER, G. I. WEBB, A. E. NICHOLSON, Y. CHEN et E. KEOGH. 2016, «Faster and more accurate classification of time series by exploiting a novel dynamic time warping averaging algorithm», *Knowledge and Information Systems*, vol. 47, nº 1, doi :10.1007/s10115-015-0878-8, p. 1–26, ISSN 0219-3116. URL https://doi.org/10.1007/s10115-015-0878-8. 57

READ, J., B. PFAHRINGER, G. HOLMES et E. FRANK. 2009, «Classifier chains for multi-label classification», dans *Machine Learning and Knowledge Discovery in Databases*, édité par W. Buntine, M. Grobelnik, D. Mladenić et J. Shawe-Taylor, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-642-04174-7, p. 254–269. 53

Chapitre 4

Résultats de Références

Sommaire

Dans le chapitre précédent, une nouvelle méthode EMMV - Ensemble de M-histogrammes Multi-Vues - permettant de répondre à la question scientifique de la classification supervisée de grands volumes de STM de tailles variables, a été présenté. Elle a été théorisée et les algorithmes qui permettent de répondre aux points durs soulevés ont été donnés.

Dans ce chapitre, la méthode est expérimentée et les résultats obtenus par celle-ci sont détaillés. Afin de tester les différents paramètres de la méthode ainsi que certaines variations de celle-ci, des jeux de données de références sont exploités. Ces mêmes données servent aussi à évaluer leurs impacts potentiels sur la méthode que ce soit en terme de taille de jeux de données ou encore de domaine d'activité.

Ces données sont aussi employées afin d'expérimenter la combinaison de la méthode avec une autre représentation issue de l'état de l'art qui est SAX. Enfin, elles permettent de positionner la méthode par rapport aux méthodes présentées dans l'état de l'art.

4.1 Les données

Dans ce chapitre, les données utilisées sont connues de la littérature et permettent de se situer par rapport à l'état de l'art. Il existe pour le moment peu d'archives, et celles-ci sont peu fournies car le sujet d'étude est finalement plutôt récent - WEASEL+MUSE et SMTS datent respectivement de 2017 et de 2015. Il n'y a donc actuellement deux archives qui sont BAYDOGAN et RUNGER [2015] et BAGNALL et collab. [2018], voir Tab. 4.1.

Une quarantaine de jeux de données issues des deux sites sont disponibles. La majorité sont des jeux de STM à longueur fixe qui sont très courtes, c'est-à-dire dont la longueur est, en moyenne, inférieure à 100 points. De manière générale, les jeux contiennent peu de séries de tailles supérieures à 500 mesures par série.

Pour rappel, en comparaison les instruments de mesure mis en place par Michelin peuvent enregistrés des séries temporelles jusqu'à *300 points par seconde*.

Seule l'archive du site BAYDOGAN et RUNGER [2015] propose quelques jeux à longueurs variables. Il y en a une dizaine en tout et pour tout. Le reste des jeux de données proposé par BAGNALL et collab. [2018] sont extrêmement récents - fin 2018 - et n'ont à l'heure actuelle fait l'objet que de très peu d'études.

Ces jeux de données vont servir à situer la méthode par rapport à l'existant. De fait, bien que la méthode soit pensée pour des jeux de données conséquents de STM à taille variable, elle est aussi capable de fonctionner sur de petits jeux de données à taille fixe. C'est pourquoi, les jeux de données présentés servent également à évaluer les limites de la méthode, ainsi que l'influence des différents paramètres sur ses performances.

Les données utilisées dans ce chapitre afin de tester notre méthode, viennent d'être définies. Leur influence sur les performances de la méthode est maintenant abordée.

	# classes	# attributes	length	NTrain	NTest
ArabicDigits	10	13	4-93	6600	2200
AUSLAN	95	22	45-136	1140	1425
CharTrajectories	20	3	109-205	300	2558
CMUsubject16	2	62	127-580	29	29
ECG	2	2	39-152	100	100
JapaneseVowels	9	12	7-29	270	370
KickvsPunch	2	62	274-841	16	10
Libras	15	2	45	180	180
RobotFailureLP1	4	6	15	17	30
RobotFailureLP2	5	6	15	17	30
RobotFailureLP3	4	6	15	17	30
RobotFailureLP4	3	6	15	42	75
RobotFailureLP1	5	6	15	64	100
NetFlow	2	4	50-997	803	534
PenDigits	10	2	8	300	10692
UWave	8	3	315	200	4278
Wafer	2	6	104-198	298	896
WalkvsRun	2	62	128-1918	28	16
ArticularyWordRecognition	25	9	144	275	300
AtrialFibrillation	3	2	640	15	15
BasicMotions	4	6	100	40	40
Cricket	12	6	1197	108	72
DuckDuckGeese	5	1345	270	60	40
EigenWorms	5	6	17984	128	131
Epilepsy	4	3	206	137	138
EthanolConcentration	4	3	1751	261	263
Ering	6	4	65	30	30
FaceDetection	2	144	62	5890	3524
FingerMovements	2	28	50	316	100
HandMovementDirection	4	10	400	320	147
HandWriting	26	3	152	150	850
Heartbeat	2	61	405	204	205
InsectWingbeat	10	200	78	30000	30000
LSST	14	6	36	2459	2466
MotorImagery	2	64	3000	278	100
NATOPS	6	24	51	180	180
PEMS	7	963	144	267	173
Phoneme	39	11	217	3315	3353
RacketSports	4	6	30	151	152
SelfRegulationSCP1	2	6	896	268	293
SelfRegulationSCP2	2	7	1152	200	180
StandWalkJump	3	4	2500	12	15

TABLEAU 4.1 – Ensemble des jeux de données de référence issus de http://www.mustafabaydogan.com/files/viewcategory/20-data-sets.html et de http://www.timeseriesclassification.com/

4.2 Influence des données

Dans un premier temps, il convient d'évaluer l'influence des données sur la méthode. Les résultats sont donc présentés dans cette section. Il est nécessaire d'évaluer, en particulier, si la méthode donne de meilleurs résultats par rapport à un type d'activités auquel appartiennent les données, ainsi que l'influence du nombre de points dans les séries, le nombre de dimensions, le nombre de classes et enfin le pourcentage de données en apprentissage/test.

4.2.1 Protocole d'expérimentations

Pour rappel, la méthode employée ici, utilise des classifieurs 1NN sur trois vues principales sur lesquelles sont extraites des vues plus petites afin de le représenter au choix sous forme de bi-histogramme ou d'histogramme. Les prédictions ensuite émises sont combinées avec un système de vote majoritaire. Cette méthode ne varie pas ici. Plusieurs boucles d'apprentissage, toujours en validation croisée, sont réalisées sur les vues afin d'obtenir le meilleur taux de classification. Nous rappelons que le taux de classification est le ratio de bonne prédiction, *i.e.* la prédiction de classe correspond à la classe réelle, sur l'ensemble des prédictions.

Définition :Taux de classification

Soit P l'ensemble des prédictions émises par la méthode. On a $\mathrm{P} = [p_1, ...p_p, ..., p_\mathrm{P}]$ avec $p_p \in \mathrm{C}$ où $\mathrm{C} = [c_1, ..., c_c]$ est l'ensemble des classes. Et soit $\mathrm{V} = [v_1, ..., \mathrm{V}_v, ..., \mathrm{V}_\mathrm{P}]$ l'ensemble des vraies classes des séries de test. Nous avons :

$$\mathrm{T}_c = \frac{\#(\mathrm{P} == \mathrm{V})}{\#\mathrm{P}}$$

où # est le nombre de.

Il est à préciser que le choix de l'évaluation s'est porté sur le taux de classification car c'est le moyen mis en oeuvre dans les autres méthodes de l'état de l'art et que cela permet donc de positionner la méthode.

4.2.2 Gestion des domaines d'activités

Dans un premier temps, il est nécessaire d'évaluer si la méthode obtient de meilleures performances pour des données au sein d'un certain type d'activité. Les jeux de données sont donc réunis en groupes d'activités comme dans SCHÄFER et LESER [2017] : Écriture, Mouvement, Capteur et Reconnaissance de paroles, voir Fig. 4.1. Les performances en classification supervisée de la méthode sont ensuite projetées en fonction du domaine d'activité du jeu de données. Puis la moyenne des taux de classification par groupe est calculée, en couleur sur la figure 4.1.

La figure montre qu'aucune tendance ne se dégage des sous-groupes, et les performances les plus basses, pour la reconnaissance d'écriture et de langage, semblent plus être dues au petit nombre d'exemples dans ces groupes d'activités, qu'aux performances de la méthode.

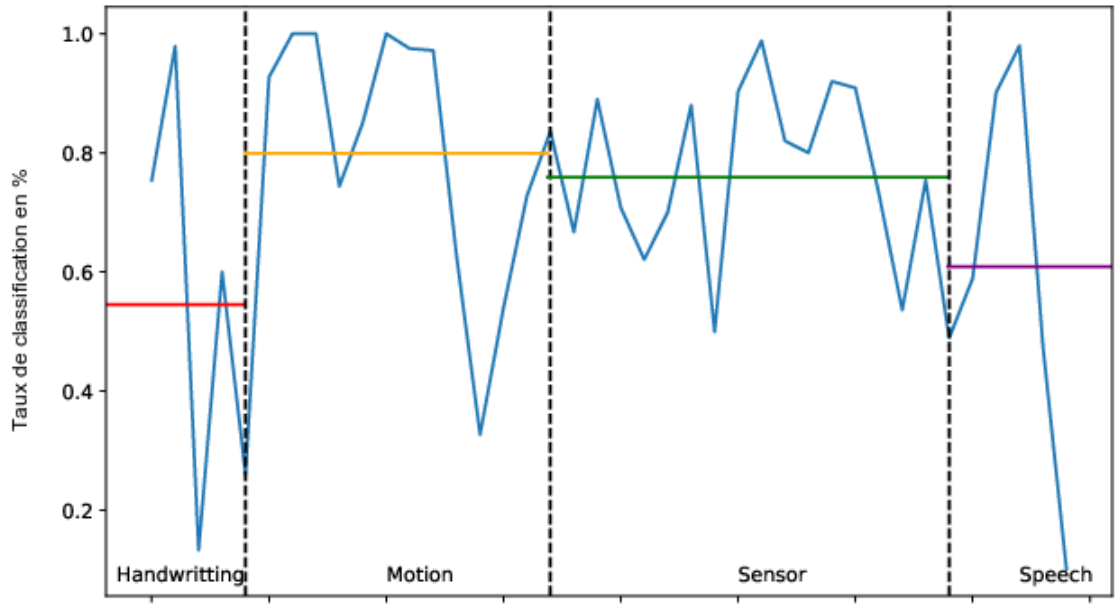


FIGURE 4.1 – Prédiction par domaines d'activité avec moyennes par domaines symbolisées par les segments de couleur.

Ainsi le M-histogramme peut être utile pour extraire, réduire, classifier les données sans influence particulière liée aux types d'activité. Une grande qualité de la méthode est donc qu'il peut être exploité pour toute application industrielle.

4.2.3 Gestion du nombre de points

Il est ensuite nécessaire de tester l'influence du nombre de points sur nos données. Les taux de classification de la méthode sont donc projetés en fonction du nombre de points par taille croissante. Puis une régression linéaire est calculée afin de voir si une tendance se dégage des données. Les résultats sont visibles dans la Fig. 4.2.

Il semble se dégager ici une tendance ascendante des performances en fonction du nombre de points. Cette conclusion corrobore la définition de l'outil M-histogramme qui est un outil statistique dont la performance dépend en effet de la taille de l'échantillon étudié. La conclusion est donc que **la méthode EMMV fonctionne mieux sur des séries de grande taille**.

4.2.4 Gestion du nombre de dimensions

Dans cette section, le test porte sur le nombre de dimensions à traiter. En effet, l'apprentissage dépend grandement de la boucle d'aléatoire de tirage des dimensions. En effet, le nombre de choix de tirages augmente avec le nombre de dimensions disponibles. Sur la Fig. 4.3, les taux de classification sont projetés en fonction du nombre de dimensions. Une droite de régression linéaire est aussi calculée.

Il ne semble pas y avoir de tendance claire, les taux de classification restent stables en fonction du nombre de dimensions. En effet, la pente de la droite de régression est très proche de 0. Cette conclusion est aussi abordée en approfondissant le choix du

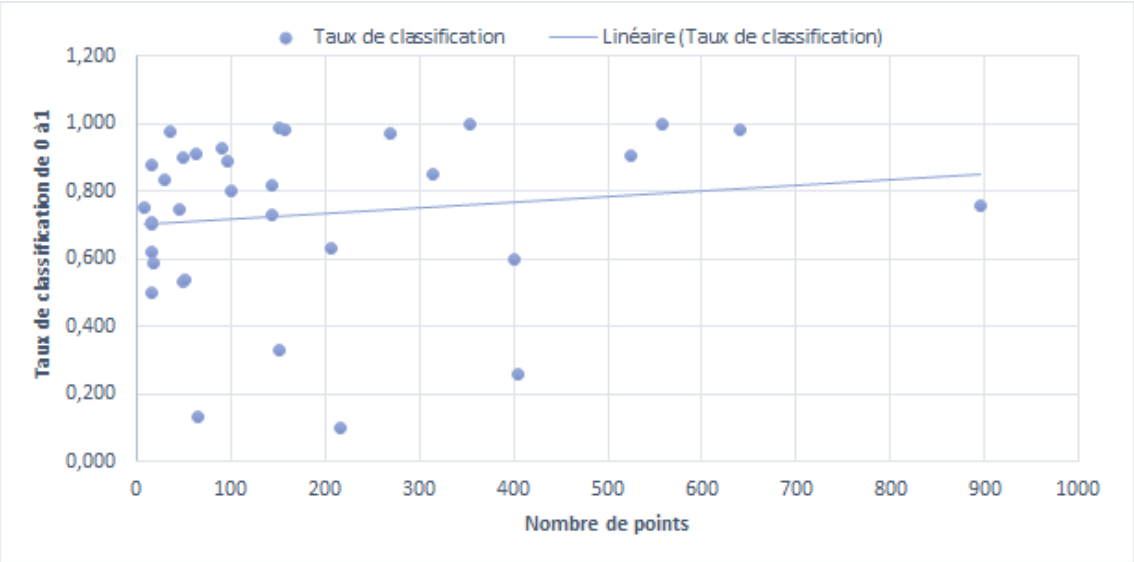


FIGURE 4.2 – Nuage des points des performances en classification de la méthode avec projection d'une droite de régression linéaire en fonction du nombre de points. Dans un souci de lisibilité de la figure, nous affichons uniquement ici les séries de moins de 1000 points.

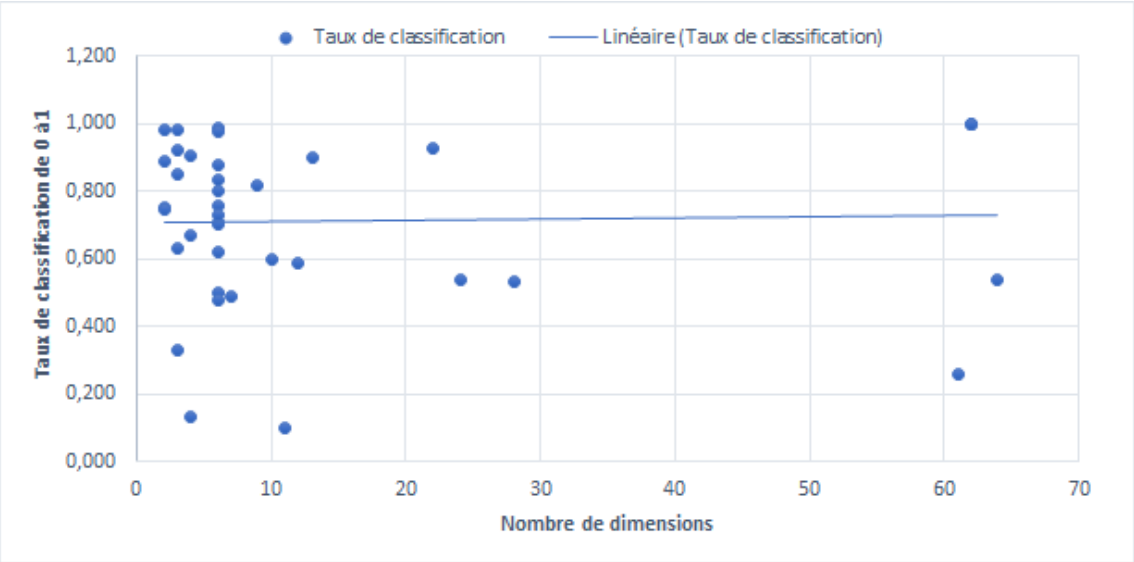


FIGURE 4.3 – Nuage des points des performances en classification de la méthode avec projection d'une droite de régression linéaire en fonction du nombre de dimensions. Dans un souci de lisibilité de la figure, nous affichons ici uniquement les séries de moins de 70 dimensions.

nombre de vues extraites à mettre en place dans la méthode car le choix de projection des dimensions dépend aussi du nombre de vues projetées.

4.2.5 Gestion du nombre de classes

Cette section permet de s'assurer que la méthode est robuste au nombre de classes. Pour cela, le taux de classification de la méthode est affiché en fonction du nombre de classes sur un nombre de séries disponibles similaires. Encore une fois, une droite de régression linéaire est calculée. Les résultats sont visibles sur la Fig. 4.4.

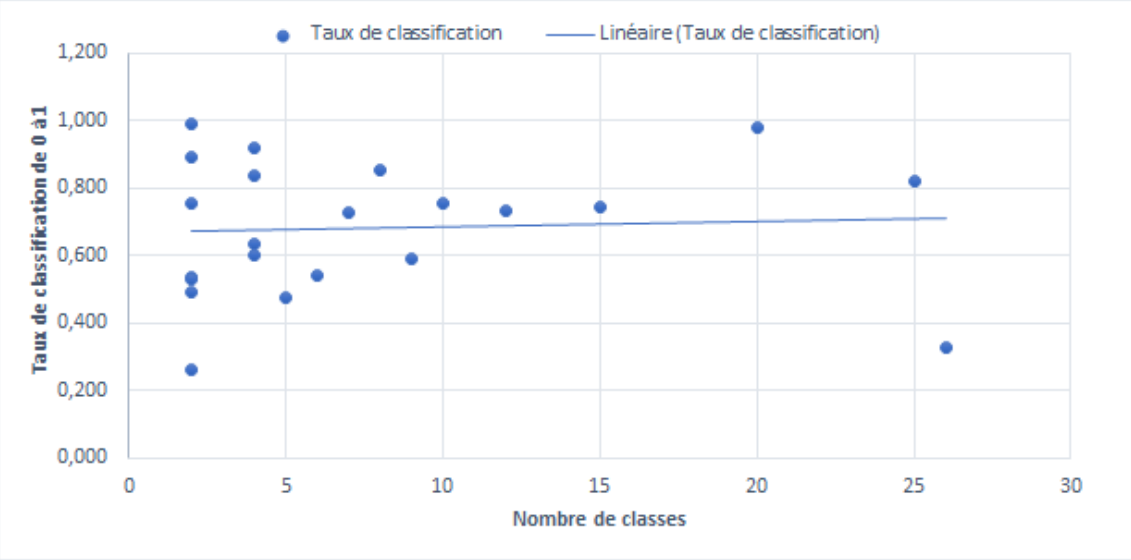


FIGURE 4.4 – Nuage des points des performances en classification de la méthode avec projection d'une droite de régression linéaire en fonction du nombre de classes.

En conclusion, le nombre de classes ne semble pas avoir d'impact réel sur les performances de la méthode.

4.2.6 Gestion du rapport pourcentage Apprentissage/Test

Finalement, il faut tester l'impact du ratio d'information disponible en apprentissage et en test. En effet, 1NN a pour caractéristique d'être très sensible à ce ratio, particulièrement lors d'un déséquilibre de classes. Moins il y a de données en apprentissage, moins il y a de chances que le tuple de test puisse être rencontré.

Une majorité des jeux de données utilisés ici sont équilibrés à 50/50, voir Fig ; 4.5. Les performances varient énormément mais pas en fonction du ratio. La même projection a été réalisée en fonction du nombre réel de séries et non plus un pourcentage mais les conclusions sont les mêmes.

4.2.7 Résumé

En conclusion, **l'influence des données sur la méthode semble être liée au nombre de points par séries temporelles**. La méthode donne de meilleurs taux de classification pour des séries de grandes tailles. Cela rejoint la définition du M-histogramme

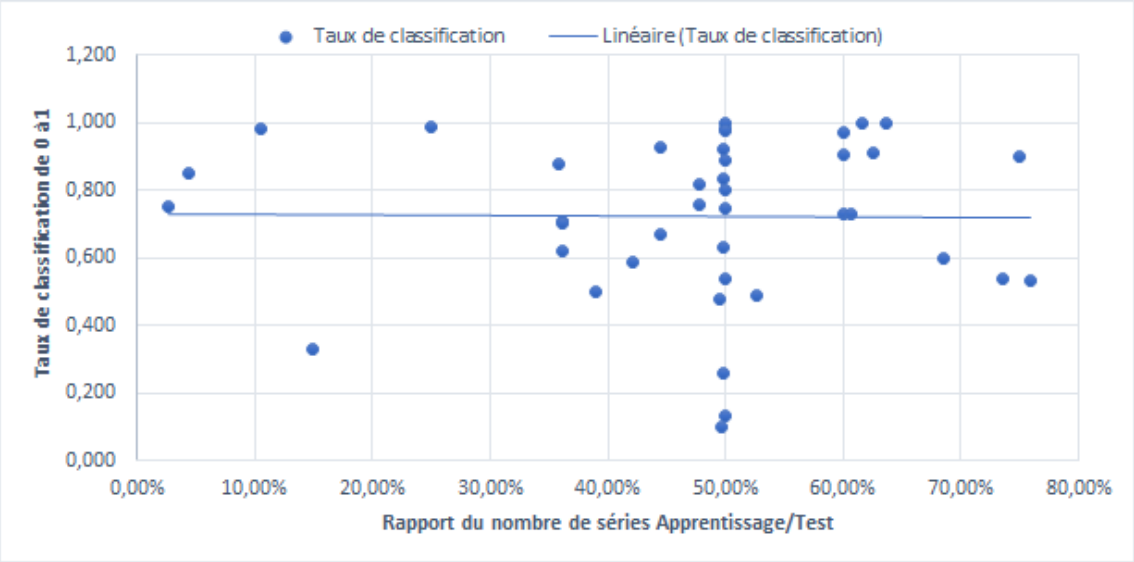


FIGURE 4.5 – Nuage des points des performances en classification de la méthode avec projection d'une droite de régression linéaire en fonction du nombre de classes.

qui est un outil statistique qui s'utilise normalement sur des échantillons de tailles conséquentes. Les autres paramètres des données, comme la classe, le nombre de dimensions et le ratio Apprentissage/Test ne semble pas influencer les résultats de la méthode. Quelques réserves sont quand même émises quant au dernier point. En effet, les jeux de données dans cette partie sont équilibrés et à ce titre les performances d'un classifieur 1NN sont bonnes. Néanmoins, dans un contexte de déséquilibre, il est possible que le ratio Apprentissage/Test influence les performances de la méthode.

Il vient d'être défini dans quelles mesures, les données, sur lesquelles est appliquée la méthode, peuvent influencer la méthode. Il reste maintenant à décrire l'influence des paramètres de ce dernier.

4.3 Influence des paramètres de la méthode

Cette partie résume l'influence des choix à paramétrer. Combien de vues faut-il explorer ? Est-il raisonnable d'augmenter le nombre de dimensions projetées du M-histogramme ? Qu'en est il du nombre d'intervalles ? Pourquoi rester sur un classifieur 1NN ? Le vote majoritaire est-il vraiment le bon choix ? Et enfin doit-on réellement prendre en compte dérivées et sommes cumulées ?

4.3.1 Protocole expérimental

Dans cette section, les tests font varier un par un les paramètres afin d'estimer l'influence de chacun sur les performances globales de la méthode. De manière générale, la méthode est donc toujours une méthode de plusieurs vues avec deux représentations possibles bi-histogrammes ou histogrammes qui permettent ensuite d'émettre

des prédictions par l'usage de 1NN. Puis nous agrégeons les prédictions par un système de vote majoritaire, pour obtenir une prédiction finale. La métrique d'évaluation reste ici aussi le taux de classification de la méthode.

4.3.2 Tirage aléatoire des dimensions

Comme expliqué dans le chapitre précédent, le paramètre k représente le nombre de vues à exploiter. Pour cela différentes valeurs de k sont testées comme pourcentages des combinaisons possibles avec $P \in [25\%, 50\%, 75\%, 95\%]$.

Pour cette expérience, les résultats en test après dix répétitions d'apprentissage de 25% des vues sont conservés. La moyenne et l'écart-type sont ensuite calculés. Le résultat est projeté dans la boite à moustache Fig. 4.6.

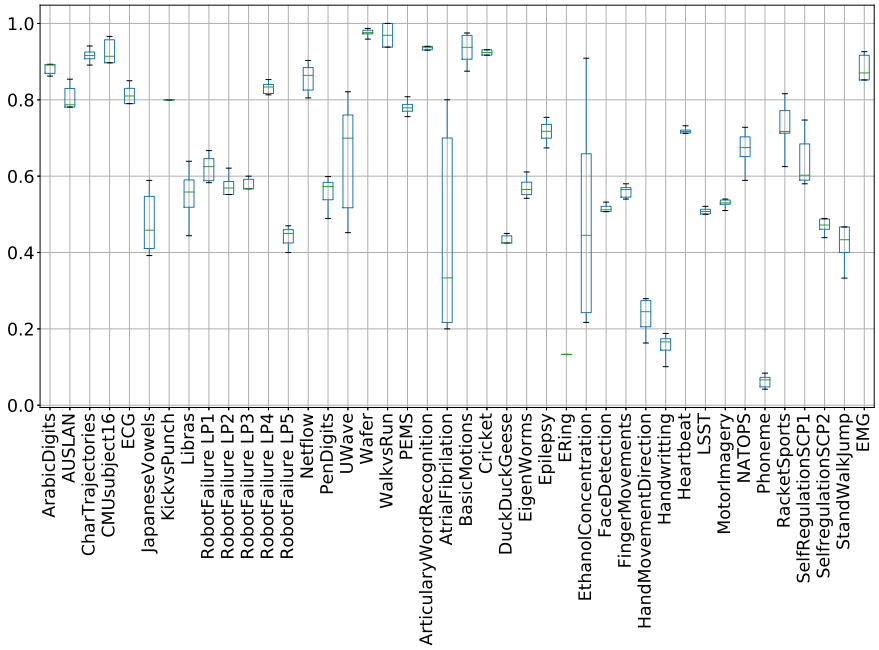


FIGURE 4.6 – Boîte à moustache des moyennes et des écart-types pour 25% des combinaisons.

La Fig.4.6 montre que l'écart-type diverge assez peu pour la majorité des jeux de données. Cela signifie que bien qu'il y ait un tirage aléatoire d'une petite sous-partie des vues possibles et que donc il y a une grande réduction des données, la méthode dispose toujours d'assez d'information pour réaliser la même classification. Il peut donc être déduit que dans le cadre de l'étude de STM **la plupart des dimensions d'une STM contiennent l'information nécessaire pour classifier les données**. Et la méthode avec 25% de combinaisons est suffisante pour émettre une classification robuste. C'est une conclusion plutôt inattendue qui tend à démontrer que dans le cadre des jeux de données de références, la multiplicité des dimensions est redondante et nécessite bien **un pré-traitement de nettoyage et de réduction comme le propose notre méthode**.

Les autres jeux de données *AtrailFibrilation*, *EthanolConcentration*, dont l'écart-type varie beaucoup, sont finalement ici des cas spéciaux où seule une infime partie des dimensions contient réellement les informations nécessaires à la classification. De

ce fait, lorsque sont extraits les bi-histogrammes/histogrammes des mauvaises dimensions, ceux-ci ne permettent pas d'établir une bonne prédiction. Cela prouve **l'utilité de la réitération de l'apprentissage des vues afin d'obtenir un meilleur taux de classification**.

Néanmoins, dans un cas comme dans l'autre la méthode est capable de produire un bon taux de classification. Le tirage aléatoire des dimensions ne réduit donc pas les performances de la méthode et à l'avantage de diminuer le temps d'apprentissage de la méthode par rapport à une recherche exhaustive des paramètres.

4.3.3 Nombre de vues

Afin de tester l'impact du nombre de vues, la moyenne des taux de classification de la méthode est calculée sur l'ensemble des jeux de données après de multiples exécutions, en faisant varier k comme expliqué ci-avant avec $P \in [25\%, 50\%, 75\%, 95\%]$. Les résultats sont visibles sur Fig. 4.7

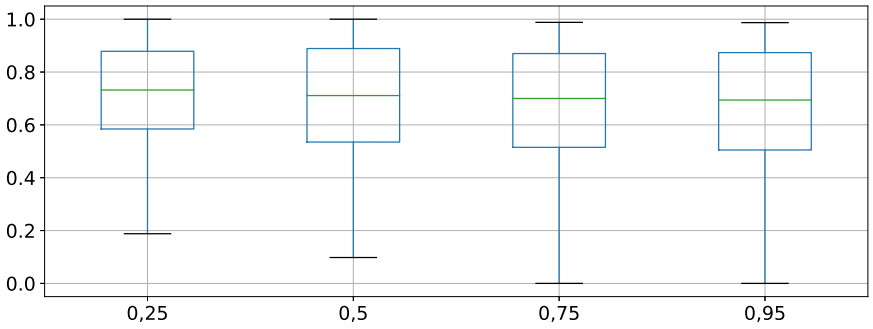


FIGURE 4.7 – Boîte à moustache des moyennes et écart-types pour 25%, 50%, 75%, 95% des combinaisons.

La figure montre que la méthode ne nécessite pas le calcul de tous les bi-histogrammes / histogrammes pour avoir de bonnes performances. En effet, les moyennes sont meilleures et les écart-types sont plus petits pour 25% que pour 95%. Cela signifie qu'ajouter trop d'informations à la méthode revient peut-être à ajouter une forme d'erreur et d'incertitude dans le processus de prédiction. En effet, il est envisageable d'imaginer que certaines dimensions, au lieu de contenir une information de classification claire, seraient en réalité très bruitées. De ce fait exploiter toutes les dimensions de manière exhaustive ne permet pas de faire le tri entre les dimensions utiles et non utiles à la classification supervisée. De plus, il est préférable, en temps de calcul aussi, de choisir de manière aléatoire de petits sous-ensembles de combinaisons plutôt que de tout calculer.

Cette conclusion rejoint la première ci-dessus qui montre l'intérêt d'extraire les dimensions plutôt que d'être exhaustif.

4.3.4 Choix du M d'un M-histogramme

Dans cette section est exploré le choix du M-histogramme. Le chapitre précédent montre que le choix d'un M très grand n'est pas idéal pour le stockage de la représenta-

tion. Cette section sert à appuyer le propos en affichant les performances du système en fonction de la taille du M.

Pour cela il faut comparer les performances entre la méthode de référence et celle avec un M égal au nombre de dimensions de la série dans les cas où cela est possible, c'est à dire que le M-histogramme n'est pas de taille trop conséquente. Il convient aussi de respecter la règle de réduction de l'information. Pour cela les jeux de données exploités sont ceux de moins de 8 dimensions. Ce chiffre n'est pas choisi au hasard. Si une STM à 8 dimensions est transformée en un M-histogramme de même taille et en choisissant de ne prendre que deux intervalles par dimensions, la matrice est de taille 2^8 soit 256 éléments. Cela représente déjà un objet très grand pour une très petite précision de représentation. En effet n'avoir que deux intervalles de projection ne permet pas une grande granularité de représentation. Par ailleurs, comme expliqué ci-avant les jeux de données ici contiennent finalement assez peu de points (une centaine). Choisir de représenter plus de dimensions ne réduit pas la taille des données.

Pour ce qui est des performances, celles-ci sont synthétisées dans la table des gagnants/perdants entre la méthode de base et la méthode avec M-histogramme de toutes les dimensions, suivante Tab. 4.2. Dans ce tableau sont comptés les nombres de fois où le taux de classification est meilleur pour chaque modèle.

Un taux de victoire de 70% est obtenu pour la méthode de base avec bi-histogramme et histogramme et une seule égalité. Encore une fois l'étude d'une sous partie des dimensions en lieu et place d'une étude exhaustive de toutes les relations semble être une meilleure option. De plus, malgré les quelques victoires de la méthode avec M-histogramme, celle-ci est plus gourmand en place de stockage.

4.3.5 Nombre d'intervalles

Une fois décidé la forme de M-histogramme voulue, il faut tester le paramètre du nombre d'intervalles par dimension. Ce paramètre est évalué sur un ensemble borné de nombres d'intervalles : de 2 à 10 intervalles inclus pour le bi-histogramme et de 2 à 50 intervalles pour l'histogramme. Le résultat présenté ici est que malgré cette borne max, qui peut paraître basse et arbitraire, celle-ci n'est que très rarement atteinte au cours de l'apprentissage. Par ailleurs, augmenter le nombre d'intervalles possibles dans la représentation revient à ne plus respecter l'hypothèse de réduction des données voulue.

Ces résultats sont résumés dans la Fig. 4.8 par les moyennes d'une dizaine d'apprentissage de bi-histogrammes et d'histogrammes.

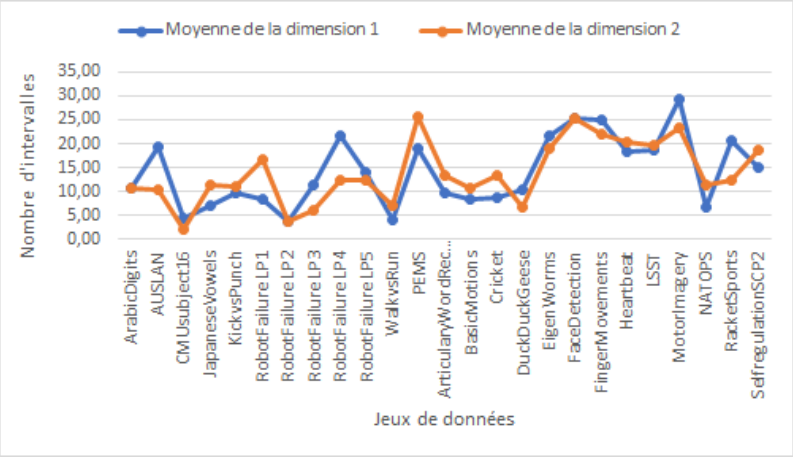
La figure montre que, pour tout jeu de données, le nombre d'intervalles moyen se situe loin de la borne max. La méthode ne rencontre pas ici de jeux de données qui nécessitent d'augmenter le nombre d'intervalles.

Cas particulier des bi-histogrammes carrés

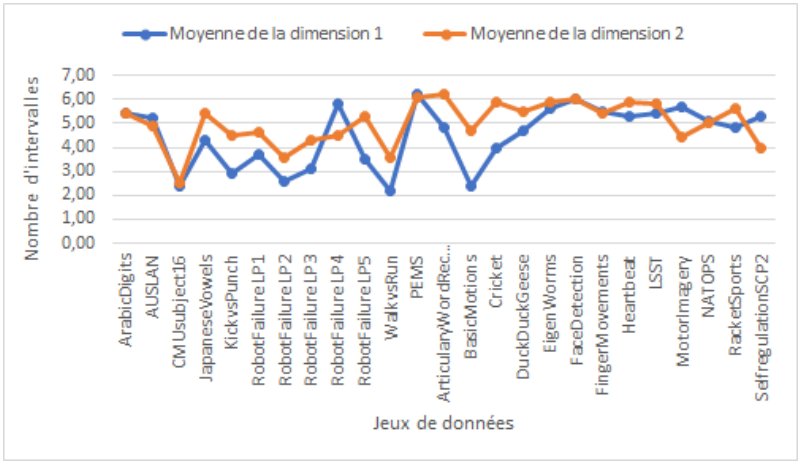
Afin de construire plus rapidement la méthode, il est possible de réduire le temps d'entraînement des bi-histogrammes en choisissant le même nombre d'intervalles pour les deux dimensions du bi-histogramme. Soit $b_1 = b_2$, la complexité est de taille B pour les nombres d'intervalles testés plutôt que $B_1 \times B_2$. Cela permet donc une grande réduction du temps de calcul. Cela peut être un compromis pour les grands jeux de données

	Modèle de base	Modèle avec M-histogrammes de toutes les dimensions
CharTrajectories	**0.979**	0.919
ECG	**0.89**	0.87
Libras	**0.744**	0.633
RobotFailureLP1	**0.708**	0.667
RobotFailureLP2	0.621	**0.655**
RobotFailureLP3	**0.744**	0.567
RobotFailureLP4	**0.88**	0.0.853
RobotFailureLP5	**0.5**	0.44
NetFlow	0.903	**0.919**
PenDigits	**0.754**	0.707
UWave	**0.851**	0.708
Wafer	**0.988**	0.987
AtrialFibrillation	**0.8**	0.533
BasicMotions	**0.975**	0.9
Cricket	**0.972**	0.875
EigenWorms	0.634	**0.718**
Epilepsy	**0.92**	0.819
EthanolConcentration	**0.909**	0.3
Ering	**0.133**	**0.133**
HandWriting	**0.26**	0.209
LSST	**0.536**	0.44
RacketSports	**0.836**	0.697
SelfRegulationSCP1	0.754	**0.771**
SelfRegulationSCP2	0.489	**0.556**
StandWalkJump	0.667	**0.733**
Nombre de victoires	**18**	**7**

TABLEAU 4.2 – Nombre de victoires en taux de classification pour les deux méthodes



(a) Nombre moyen d'intervalles par jeux de données pour les histogrammes



(b) Nombre moyen d'intervalles par jeux de données pour les bi-histogrammes

FIGURE 4.8 – Courbe des moyennes d'intervalles pour histogrammes et bi-histogrammes par jeux de données

permettant de calculer plus vite en perdant peu de points de classification, voir Fig. 4.9 et Tab. 4.3

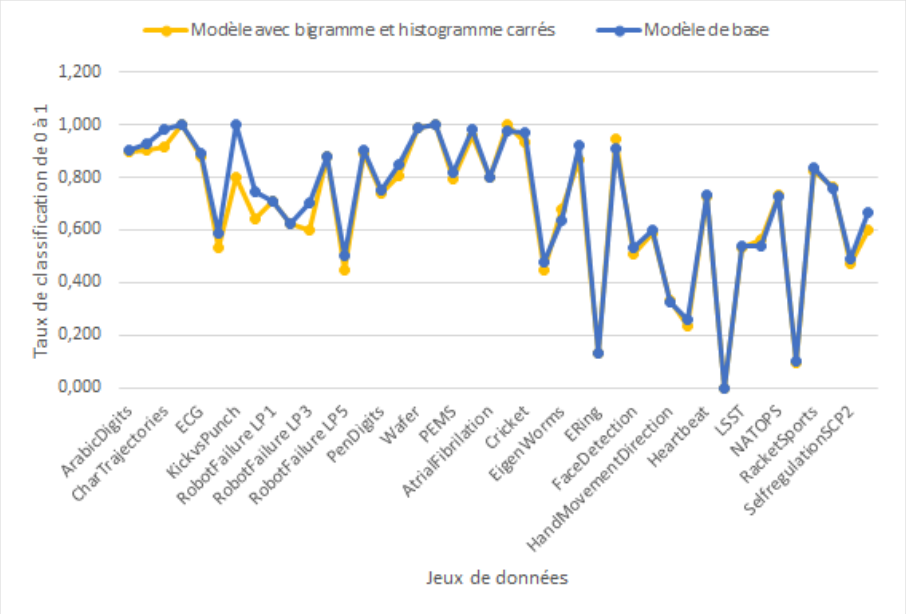


FIGURE 4.9 – Taux de classification pour la méthode de bi-histogrammes et histogrammes carré, face à la méthode de base

Le tableau Tab. 4.3 montre que la méthode de base obtient de meilleurs résultats dans une grande majorité des cas. Néanmoins, comme projeté sur Fig. 4.9, la différence entre les performances des deux méthodes est plutôt mince. Il est donc possible de faire un compromis judicieux entre performances de classification et de calcul.

4.3.6 Classifieurs

Il faut ensuite réaliser deux tests majeurs dans le cadre du choix du classifieur, pour évaluer l'impact du nombre de voisins dans le choix d'un classifieur KNN, ainsi que le choix d'un classifieur tout à fait différent.

Nombre de voisins

Bien qu'ayant vu ci-avant, que 1NN était certainement le meilleur choix, au moins du point de vue de la complexité, il faut quand même tester les performances de celle-ci en les comparant aux augmentations du nombres de voisins. Il est donc nécessaire de recalculer les performances pour chaque jeu de données avec respectivement 2 et 4 voisins de plus, c'est-à-dire avec **1, 3 et 5 voisins**.

Naive Bayes

Par ailleurs, il convient aussi de tester les performances de la méthode avec un autre type de classifieur. Le choix s'est porté sur Naive Bayes. En effet, ce dernier de la même manière que 1NN ne nécessite pas de paramétrage supplémentaire. Ce classifieur est un classifieur probabiliste extrêmement simple qui calcule les probabilités

	Modèle de base	Modèle avec bi-histogrammes et histogrammes carrés
AUSLAN	**0.927**	0.902
CharTrajectories	**0.979**	0.915
CMUsubject16	**1.0**	**1.0**
ECG	**0.89**	0.88
JapaneseVowels	**0.589**	0.535
KickvsPunch	**1.0**	0.8
Libras	**0.744**	0.644
RobotFailureLP1	**0.708**	**0.708**
RobotFailureLP2	**0.621**	**0.621**
RobotFailureLP3	**0.7**	0.60
RobotFailureLP4	**0.88**	**0.88**
RobotFailureLP5	**0.5**	0.45
NetFlow	**0.903**	0.891
PenDigits	**0.754**	0.742
UWave	**0.851**	0.808
Wafer	**0.988**	**0.988**
WalkvsRun	**1.0**	**1.0**
ArabicDigits	**0.901**	0.90
ArticularyWordRecognition	**0.98**	0.96
AtrialFibrilation	**0.8**	**0.8**
BasicMotions	**1.0**	**1.0**
Cricket	**0.972**	0.931
DuckDuckGeese	**0.475**	0.45
EigenWorms	0.634	**0.679**
Epilepsy	**0.92**	0.87
EthanolConcentration	**0.943**	**0.943**
Ering	**0.133**	**0.133**
FaceDetection	**0.532**	0.511
FingerMovements	**0.6**	0.59
HandMovementDirection	**0.33**	**0.33**
HandWriting	**0.26**	0.235
Heartbeat	**0.732**	0.727
LSST	**0.536**	0.535
MotorImagery	**0.56**	**0.56**
NATOPS	0.728	**0.733**
PEMS	**0.82**	0.792
Phoneme	**0.101**	0.096
RacketSports	**0.836**	0.822
SelfRegulationSCP1	**0.761**	**0.761**
SelfRegulationSCP2	**0.489**	0.472
StandWalkJump	**0.667**	0.6
Nombre de victoires	**26**	2

TABLEAU 4.3 – Nombre de victoire en taux de classification pour les deux méthodes de base et avec bi-histogrammes et histogrammes carrés

d'appartenir à une classe sachant les valeurs que prennent les différents attributs caractérisant le tuple. Ici son usage est adapté telle que la probabilité calculée est celle d'appartenir à une classe connaissant les valeurs de fréquences dans les différentes intervalles des M-histogrammes.

Résultats

Les résultats sont contenues dans la Fig. 4.10 et le Tab. 4.4.

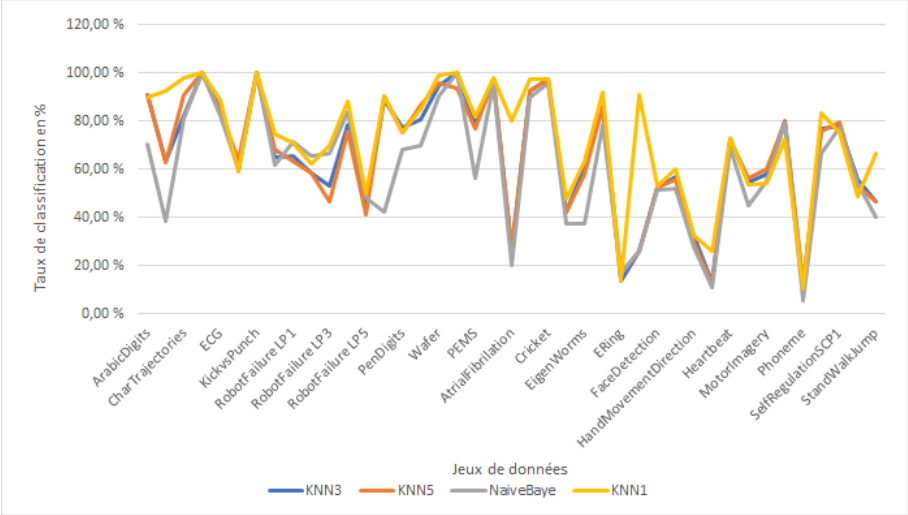


FIGURE 4.10 – Taux de classification en pourcentage pour tous les jeux de données par classifieur : plus proches voisins avec 1, 3 et 5 voisins, et Naive Bayes

	1NN	3NN	5NN	NB
Nombre de victoires	**26**	6	10	4

TABLEAU 4.4 – Nombre de victoire en taux de classification pour les différents classifieurs, plus proches voisins avec 1, 3 ou 5 voisins et Naive Bayes

La figure comme le tableau montre que dans la plus grande majorité des cas, 65%, le classifieur 1NN est bien meilleur que les autres. En plus, d'être plus rapide en temps de calcul, ce classifieur est aussi le plus performant.

4.3.7 Vote final

Ensuite, le test se porte sur les implications du système de vote final. Il faut alors comparer un vote majoritaire et un vote pondéré. Comme expliqué ci-avant, la méthode n'a pas d'experts à disposition afin de déterminer le choix des poids. Il convient donc de réaliser une paramétrisation aléatoire identique à celle qui était mise en place dans le reste de la méthode, pour déterminer les poids.

Ici les poids sont normés par le nombre de M-histogrammes pour donner une somme totale de poids égale à 1.

Définition : Classification par vote pondéré

Soit P l'ensemble des prédictions émises par les M-histogrammes par une série temporelle. On a P = $[p_1, ... p_p, ..., p_P]$ avec $p_p \in C$ où C = $[c_1, ..., c_c]$ est l'ensemble des classes, et W = $[w_1, ..., w_p, ..., w_P]$, l'ensemble des poids pour chaque prédiction tel que $\sum W = 1$. Nous avons P_f la prédiction finale telle que :

$$P_f = Argmax(WP)$$

et $P_f \in C$

Les poids sont donc tirés de manière aléatoire avec réitération sur apprentissage. Les poids permettant la meilleure prédiction sont conservés et appliqués dans la méthode final.

Les résultats obtenus sont visibles dans la Fig. 4.11.

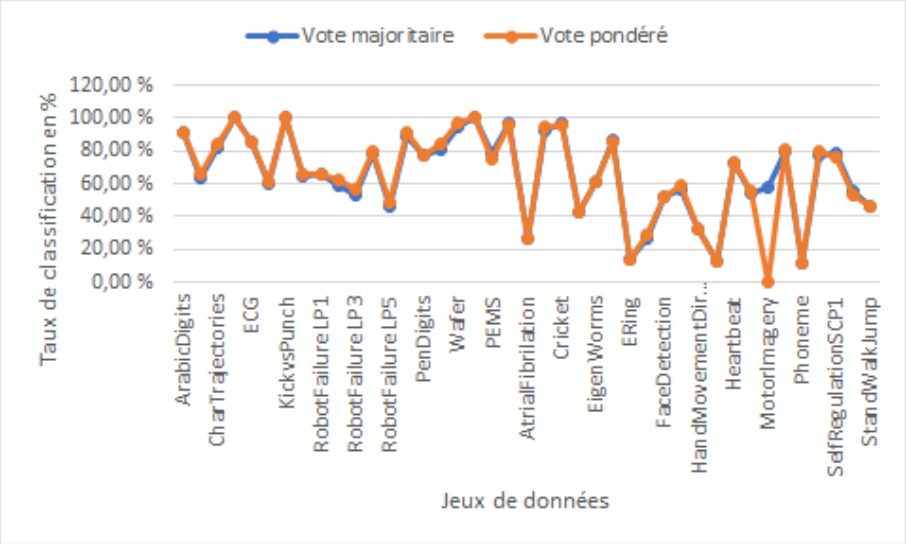


FIGURE 4.11 – Taux de classification en pourcentage pour tous les jeux de données par système de vote majoritaire et pondéré

Cette figure montre la superposition quasi-parfaite des deux courbes de résultats. Cela signifie que les deux méthodes obtiennent les mêmes taux de classification. Seulement, le vote pondéré implique la paramétrisation des poids et donc un temps de calcul allongé. Il ne semble donc pas judicieux de le mettre en place dans la méthode. En effet, la méthode majoritaire est plus rapide et donne des résultats identiques. Cela rejoint la conclusion amenée dans la présentation de la méthode. L'utilisation d'une méthode pondérée est conditionnée par la mise à disposition d'un expert des données.

4.3.8 Calculs complémentaires

Finalement, il est nécessaire d'évaluer l'impact des dimensions calculées et ajoutées sur les performances de la méthode. Est-ce que cela ajoute de la valeur ? Faut-il ne garder que les attributs calculés ? Il est à rappeler ici que la méthode contient une phase de calculs des dimensions dérivées et intégrées des dimensions de base. En effet, l'em-

ploi de la représentation fait perdre les notions de tendance et d'ordre d'événements. L'emploi de la dérivée et de la somme cumulée sert à limiter ces pertes.

Il convient donc comparer les performances de chaque attribut seul dans la méthode. Les résultats obtenus sont visibles dans la Fig.4.12 et dans le Tab. 4.5.

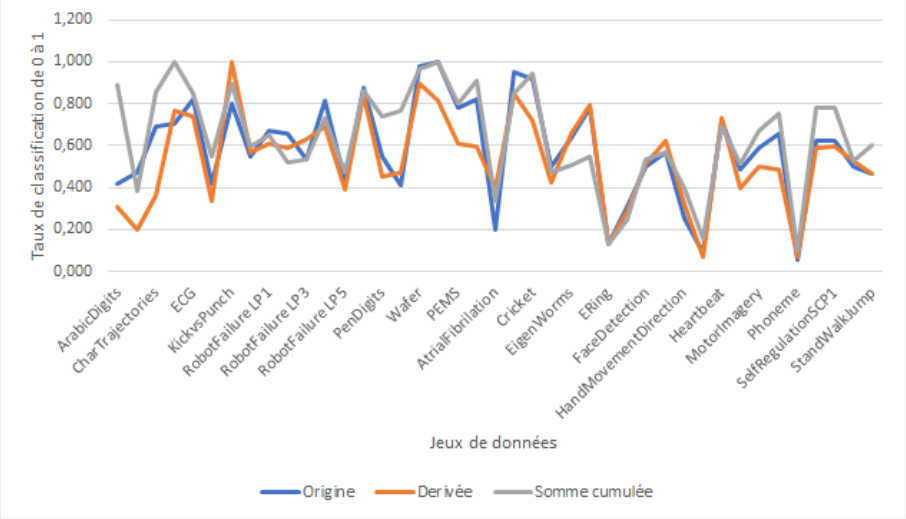


FIGURE 4.12 – Taux de classification en pourcentage pour tous les jeux de données par vue principale : Donnée originale, dérivée, somme cumulée

	Originale	Dérivée	Somme cumulée
Nombre de victoire	10	8	**24**

TABLEAU 4.5 – Nombre de victoire en taux de classification pour les séries originales, dérivées et sommes cumulées.

La figure montre que les séries originales, dérivées et sommes cumulées peuvent, à tour de rôle, présenter un avantage de classification dans les études de certains jeux de données avec un net avantage pour la somme cumulée. Bien qu'il existe dans la littérature l'usage de la dérivée, celle ci n'est jamais caractérisée comme une vue et à ce titre peut être combinée avec les autres dimensions. Ce n'est pas le cas ici. De la même manière, l'usage de la somme cumulée n'a jamais été mentionnée dans la littérature, pourtant il est clair ici que celle-ci représente un réel intérêt dans le cas de classification de STM, voir Tab. 4.5.

L'ensemble des paramètres et plusieurs variations de la méthode de base viennent d'être testés. Un résumé des conclusions principales est présenté dans la section suivante.

4.3.9 Résumé

En résumé, la méthode avec M-histogrammes où M=1 et 2 fonctionne mieux que la méthode des M-histogrammes complets. Par ailleurs, il est aussi plus intéressant en temps de sélectionner une petite sous-partie de toutes les combinaisons possibles

plutôt que d'effectuer toutes les transformations. Enfin, les bi-histogrammes carrés donnent des résultats très similaires aux bi-histogrammes rectangles, il est donc possible de réduire le temps d'exécution de la méthode en choisissant cette première alternative.
Pour ce qui est du système de classification, il est confirmé qu'un vote majoritaire sur un ensemble de classifieurs 1NN est à la fois plus performant et plus rapide. Finalement, la dernière section montre l'importance de la définition de la STM renforcée pour l'application de la méthode. En particulier, la somme cumulée permet l'obtention de très bons résultats de classification supervisée.

Il vient d'être établi les différentes implications entre les réglages des paramètres de la méthodes et les résultats. Il reste maintenant un autre point de la méthode à présenter, sa capacité à être combiné avec d'autres techniques de classification.

4.4 Résultats en combinatoire

La méthode proposée se base sur le changement de représentation en plusieurs M-histogrammes des données. Il est facilement envisageable d'utiliser une nouvelle représentation pour les différentes vues de données et de maintenir celle-ci dans la méthode global. C'est ce qui est exploré ici. C'est-à-dire que sont évaluées les possibilités de combinaisons et si celles-ci donnent de bons résultats.

En effet, l'histogramme qui permet ici d'exploiter ici les relation intra dimensions peut être remplacé dans la méthode par une autre représentation, par exemple SAX. Pour rappel, cette méthode est présentée dans le chapitre Etat de l'art.

4.4.1 SAX

Comme expliqué dans l'état de l'art la transformation par SAX d'une série en une chaîne de caractères se fait sur des séries univariées. C'est donc une représentation permettant elle aussi de mettre en exergue une relation intra-dimensionnelle. La méthode tire toujours les dimensions de manière aléatoire, mais il applique SAX à la place de l'histogramme.

L'application de SAX ici transforme la série en une chaîne de chiffres. De cette manière, la représentation est toujours dans le domaine euclidien et la distance, précédemment mise en place, peut toujours être appliquée ici. Cela permet de minimiser les modifications de la méthode et donc de voir l'influence réelle de cette transformation, sans endommager les performances de la méthode avec SAX.

4.4.2 Combinaison

Dans la méthode, le même système de vues est toujours utilisé. La différence majeure est qu'au vu des capacités de SAX à traiter les séries temporelles, les tests sont réduits aux jeux de données où les STM ont les mêmes longueurs de séries. Les résultats sont présentés dans la Fig. 4.13

Diagramme des différences critiques

Afin d'évaluer les performances d'un modèle plutôt qu'un autre, la représentation du *Diagramme des différences critiques* est ici employée. Cette représentation présentée dans DEMŠAR [2006], permet de ranger les modèles du plus au moins performants en taux de classification. Puis, il calcule s'il existe une différence significative entre les performances via un test statistique afin de savoir si ces modèles sont réellement différents. S'il n'y a pas de réelle différence, alors les modèles sont connectés par une barre noire sur la représentation. Nous obtenons ici la figure suivant Fig. 4.13

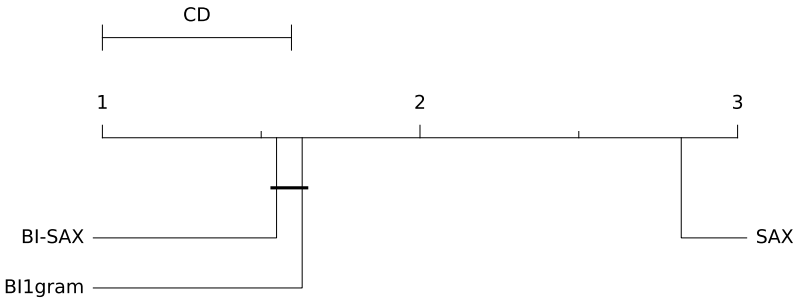


FIGURE 4.13 – Diagramme des différences critiques pour la méthode de base, appelée **Bi1gram**, de SAX, appelé **SAX** et combiné avec SAX, appelé **BI-SAX**

Il faut souligner ici que SAX ne permet pas une implémentation de la méthode sur des séries à tailles variables. Néanmoins, il est déjà observable que la méthode avec SAX seul n'obtient pas d'aussi bons résultats que la méthode appliquant la représentation bi-histogramme. Il peut donc être conclu que les relations inter-dimensions, que ne peut pas capter seule la méthode SAX, sont réellement déterminantes pour la classification des STM. Par ailleurs, il peut aussi être vu que la méthode de base se combine très bien avec d'autres représentations et obtient ici de très bons résultats.

La méthode permet donc la combinaison de plusieurs représentations différentes et notamment de représentation ayant fait leurs preuves sur les données univariées. Il reste maintenant à aborder la comparaison de la méthode avec les méthodes issues de l'état de l'art.

4.5 Comparaison globale

Finalement, il faut donc réaliser une comparaison de la méthode avec l'état de l'art, c'est-à-dire en comparaison avec WEASEL+MUSE, SMTS, DTW pour les STM et 1NN avec une simple distance euclidienne. Toutes ces méthodes ont déjà été présentées dans l'état de l'art. Les résultats obtenus sont présentés dans le tableau suivant Tab. 4.6

Les noms des modèles par ordre d'apparition dans le tableau sont :

— ED1NN :Classifieur 1NN avec distance euclidienne BAGNALL et collab. [2018]

— DTW1NN$_D$: Première adaptation de DTW avec distances inter-dimensions BAGNALL et collab. [2018]

— DTW1NN$_I$: Deuxième adaptation de DTW avec distances intra-dimensions BAGNALL et collab. [2018]

— SMTS : Modèle proposé par BAYDOGAN et RUNGER [2015] basé sur une forêt aléatoire

— WEASEL_MUSE : Modèle présenté dans SCHÄFER et LESER [2017] basé sur les bigrammes de mots

— B1gr1NN : Modèle de base que nous proposons avec vues bi-histogrammes et histogrammes avec classifieur 1NN

— S$quar$B1gr1NN : Modèle avec les bi-histogrammes carrés et classifieur 1NN

— M1gr1NN : Modèle avec les M-histogrammes et classifieur 1NN

— SAX1NN : Application de la représentation SAX seule avec classifieur 1NN

— BSAX1NN : Application de notre méthode avec la représentation SAX et classifieur 1NN

Les résultats obtenus par les multiples méthodes proposées ici, sur les jeux de données de référence, sont comparés avec les résultats obtenus par les modèles de références. Les résultats de ces modèles sont ceux **présentés par les auteurs dans leurs publications**. Ils sont donc donnés à titre indicatif.

Dans ce travail de thèse, les résultats des modèles de l'état de l'art n'ont pas été recréés en raison du nombre de paramètres à régler dont nous n'avions pas les valeurs optimales. En effet, la complexité des méthodes de références a rendu la faisabilité de la ré-exécution des deux méthodes sur tous les jeux de données avec paramétrages non réalisable dans le temps imparti. Il est rappelé ici que les complexités des méthodes sont de respectivement : O(J$_{ins}\sqrt{2M+1}$NT(R$-$1)) suivi de O(J$_{ts}\sqrt{RJ_{ins}}$NlogN) pour SMTS et O(min[NT^3,N^{2}T^2]$*m$ pour WEASEL+MUSE. Par ailleurs, cette dernière méthode n'a pas encore été publiée officiellement.

Il est à noter, toutefois, que quelques exécutions de la méthode STM ont été faites pour quelques jeux de données afin de pouvoir comparer les temps d'exécution des modèles. Les résultats sont visibles en annexe. Ils ne sont pas donnés de manière explicite ici, car trop peu de jeux de données ont été testés pour en conclure un résultat significatif. Et par ailleurs, l'exécution de WEASEL+MUSE n'a pas abouti, car la méthode a tourné plusieurs heures sans donner de résultats.

Il peut déjà être vu sur ce tableau en gras, que chacune des méthodes créées durant cette thèse est capable de compétiter avec les modèles de références. En effet, en fonction des jeux de données ils peuvent obtenir des résultats équivalents voir meilleurs. Encore une fois, il faut rappeler ici que la méthode EMMV en plus d'être compétitif, a principalement été pensé en terme de rapidité d'exécution et pour sa capacité à traiter dans un même ensemble des séries de tailles très variables.

	ED1NN	DTW1NN$_I$	DTW1NN$_D$	SMTS	WEASEL_MUSE	B1gr1NN	S$quar$B1gr1NN	M1gr1NN	SAX1NN	BSAX1NN
AUSLAN				0.947	**0.97**	0.927	0.902			
CharTrajectories	0.964	0.969	0.989	**0.992**	0.973	0.979	0.915	0.919		
CMUsubject16				**1.0**	**1.0**	**1.0**	**1.0**			
ECG				0.818	0.88	**0.89**	0.88	0.87		
JapaneseVowels	0.924	0.959	0.949	0.969	**0.976**	0.589	0.535			
KickvsPunch				0.82	**1.0**	**1.0**	0.8			
Libras	0.833	0.894	0.87	**0.909**	0.894	0.744	0.644	0.633	0.656	0.794
RobotFailureLP1				0.856	**0.94**	0.708	0.708	0.667	0.50	0.688
RobotFailureLP2				**0.76**	0.733	0.621	0.621	0.655	0.517	0.69
RobotFailureLP3				0.76	**0.9**	0.7	0.60	0.567	0.5	0.6
RobotFailureLP4				0.895	**0.96**	0.88	0.88	0.853	0.453	0.8
RobotFailureLP5				0.65	**0.69**	0.5	0.45	0.44	0.44	0.52
NetFlow				**0.977**	0.961	0.903	0.891	0.919		
PenDigits	0.973	0.939	**0.977**	0.917	0.912	0.754	0.742	0.707	0.785	0.818
UWave	0.881	0.868	0.903	**0.941**	0.916	0.851	0.808	0.708	0.791	0.878
Wafer				0.965	**0.997**	0.988	0.988	0.987		
WalkvsRun				**1.0**	**1.0**	**1.0**	**1.0**			
ArabicDigits	0.967	0.959	0.963	0.964	**0.992**	0.901	0.90			
ArticularyWordRecognition	0.97	0.98	**0.987**			0.98	0.96		0.677	0.98
AtrialFibrilation	0.267	0.267	0.22			**0.8**	**0.8**	0.533	0.467	**0.8**
BasicMotions	0.676	**1.0**	0.975			**1.0**	**1.0**	0.9	0.425	0.95
Cricket	0.944	0.986	**1.0**			0.972	0.931	0.875	0.472	**1.0**
DuckDuckGeese	0.275	0.55	**0.6**			0.475	**0.45**		0.2	0.2
EigenWorms	0.549	-	0.618			0.626	0	**0.718**	0.291	0.58
Epilepsy	0.666	**0.978**	0.964			0.92	0.87	0.819	0.623	0.90
EthanolConcentration	0.293	0.304	0.323			**0.943**	**0.943**	0.304	0.285	0.654
Ering	0.133	0.133	0.133			0.133	0.133	0.133	**0.3**	0.133
FaceDetection	0.519	-	0.529			**0.532**	0.511		0.503	0.517
FingerMovements	0.55	0.52	0.53			**0.6**	0.59		0.49	**0.6**
HandMovementDirection	0.278	0.306	0.231			0.33	0.33		0.365	**0.381**
HandWriting	0.2	**0.316**	0.286			0.26	0.235	0.209	0.126	0.228
Heartbeat	0.619	0.658	0.717			0.732	0.727		0.722	0.727
LSST	0.456	0.575	0.551			0.536	0.535		0.441	0.539
MotorImagery	0.51		0.5			0.56	0.56	0.440	0.5	**0.56**
NATOPS	0.85	0.85	0.883			0.728	0.733		0.594	0.767
PEMS	0.705	0.734	0.711			0.82	0.792		0.792	**0.827**
Phoneme	0.104	0.151	0.151			0.101	0.96		0.094	0.094
RacketSports	**0.868**	0.842	0.803			0.836	0.822	0.697	0.487	0.783
SelfRegulationSCP1	0.771	0.765	0.775			0.761	0.761	0.771	0.744	0.812
SelfRegulationSCP2	0.483	0.533	0.5339			0.489	0.472	0.556	544	**0.583**
StandWalkJump	0.2	0.333	0.2			**0.667**	0.6	0.733	0.467	**0.667**

TABLEAU 4.6 – Tableau des taux de classification de tous les modèles.
Le *gris clair* correspond aux résultats que nous n'avons pas car ils n'ont pas encore été publiés par leurs auteurs respectifs. Quant au *gris foncé*, cette couleur correspond aux résultats que nous n'avons pas car nous ne pouvions pas les calculer. Pour la méthode *Bi-Sax* et la méthode *SAX*, c'est à cause de l'incompatibilité des modèles avec les STM à longueur variable. Pour la méthode *Mgrams*, c'est à cause du gros grand nombre de dimensions des jeux de données correspondants.

4.5.1 Longueurs fixes

Les performances sont résumées dans deux diagrammes critiques en fonction des propriétés de longueurs des séries. Il est à préciser, en effet, que les méthodes créées ne sont pas comparés aux mêmes méthodes dans chaque diagramme. Cela est tout simplement dû aux faits que les résultats des méthodes sur l'un ou l'autre des ensembles de jeux de données ne sont pas disponibles. Dans certain cas, ils n'ont pas pu être calculés en raison des spécificités de la méthodes. Dans l'autre, les auteurs ne les ont pas communiqués, voir Tab. 4.6.
Tout d'abord, sur le diagramme des différences critiques pour les jeux de données contenant des séries de longueurs fixes, des performances similaires aux autres modèles de la littérature sont obtenues, voir Fig. 4.14

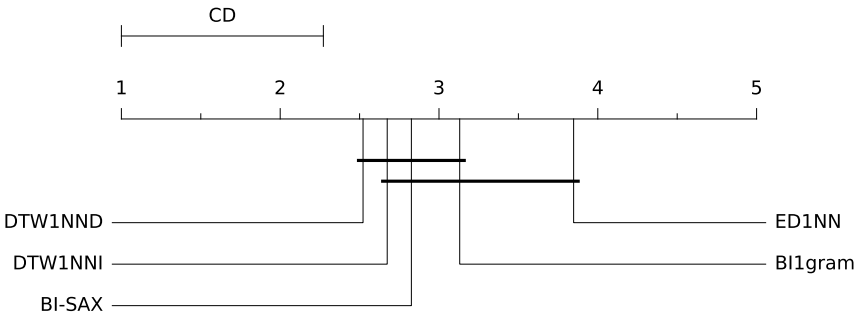


FIGURE 4.14 – Diagramme des différences critiques pour la méthode de base, appelée **Bi1gram**, de la distance euclienne avec 1NN, appelé **ED1NN**, des distances DTW avec 1NN, appelés **DTW1NNI** et **DTW1NND** et la méthode combinée appelée **BI-SAX** sur les longueurs fixes

4.5.2 Longueurs variables

Des résultats semblables sont obtenus pour les modèles de références sur les séries temporelles à tailles variables Fig 4.15

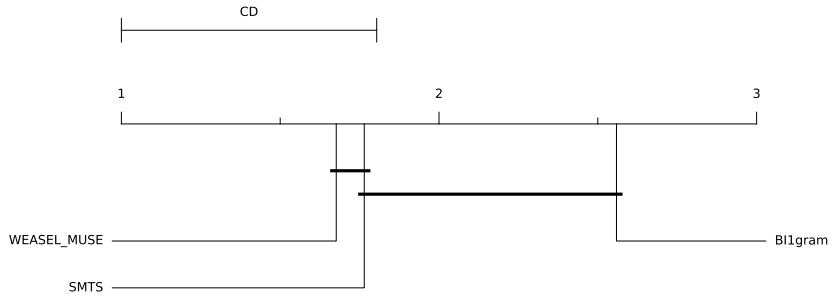


FIGURE 4.15 – Diagramme des différences critiques pour la méthode de base, appelée **Bi1gram** avec les modèles de références WEASEL+MUSE et SMTS sur les longueurs variables

Encore une fois le modèle WEASEL+MUSE, bien que présenté dans un workshop, n'a pas encore été officiellement publié. De la même manière les résultats présentés

pour les modèles références sont ceux obtenus par les auteurs et qui n'ont pas été reproduits.

L'ensemble des résultats obtenus sur les données a été détaillé, avec les paramètres et ceux issus de combinaisons et de comparaisons avec l'état de l'art. Il faut maintenant synthétiser l'ensemble des informations importantes.

4.6 Résumé

Pour résumer, la méthode EMMV est robuste, pour tout type de jeux de données. Elle est en plus capable de gérer de grosses variations de longueurs. La méthode est aussi très flexible et permet à l'utilisateur une grande liberté de paramétrage, il peut même combiner plusieurs représentations de différentes natures. Par ailleurs, bien qu'il obtienne des résultats similaires en terme de classification aux modèles de référence, EMMV est de complexité inférieure, tout en prenant en charge de manière explicite les variations de longueurs. EMMV est une méthode généraliste qui n'est pas influencé par la nature des données ni par le biais de certains domaines de données.

Par ailleurs, la méthode est aussi robuste au nombre de dimensions, au nombre de classes, ainsi qu'au ratio Apprentissage/Test. Elle a été testé par rapport à l'influence de différents paramètres, et la méthode telle que fournie permet d'obtenir les meilleurs résultats.

Pour conclure, l'introduction de l'usage d'une méthode M-histogramme multi-vues est proposée ici avec succès. De plus, le concept de l'usage de l'intégration des données par la somme cumulée est aussi exploré et montre que cette transformation peut faire la différence dans la classification d'un grand nombre de jeux de données PLAUD et collab. [2019].

Points clefs du chapitre Résultats de références

Sur les données de références, nous avons montré :

Points forts de la méthode	— Robuste avec peu de dimensions représentées — Classification rapide notamment grâce à l'hyperparamétrisation aléatoire — Compromis possible entre efficacité et rapidité grâces aux bi-histogrammes carrés — Insensible à la majorité des propriétés des données — Flexible dans le paramétrage pour l'utilisateur — Utilisation de la somme cumulée améliore la classification — Modèle combinable avec d'autres représentations de relation inter dimensionelle
Limites	— Nombre de points dans une série influence le résultat — M-histogrammes de toutes les dimensions sont trop grands à stocker — Classifieurs 1NN sensibles aux déséquilibres de classes

La méthode vient d'être appliquée sur des données de références, nous allons maintenant dérouler un cas applicatif de la méthode rencontré chez Michelin. Le challenge est alors de gérer de grands volumes de données de tailles variables qui par ailleurs contiennent de nombreuses incertitudes.

4.7 Références

BAGNALL, A., H. DAU, J. LINES, M. FLYNN, J. LARGE, A. BOSTROM, P. SOUTHAM et E. KEOGH. 2018, «The UEA multivariate time series classification archive, 2018», *CoRR*, vol. abs/1811.00075. URL http://arxiv.org/abs/1811.00075. 74, 92, 93

BAYDOGAN, M. G. et G. RUNGER. 2015, «Learning a symbolic representation for multivariate time series classification», *Data Mining and Knowledge Discovery*, vol. 29, nº 2, doi :10.1007/s10618-014-0349-y, p. 400–422, ISSN 1573-756X. URL https://doi.org/10.1007/s10618-014-0349-y. 74, 93

DEMŠAR, J. 2006, «Statistical comparisons of classifiers over multiple data sets», *Journal of Machine learning research*, vol. 7, nº Jan, p. 1–30. 92

PLAUD, A., E. MEPHU NGUIFO et J. CHARREYRON. 2019, «Classification des séries temporelles multivariées par l'usage de mgrams», *CAp 2019*. 96

SCHÄFER, P. et U. LESER. 2017, «Multivariate time series classification with WEASEL+MUSE», *CoRR*, vol. abs/1711.11343. URL http://arxiv.org/abs/1711.11343. 76, 93

Chapitre 5

Application Michelin

Sommaire

Le problème soulevé dans cette thèse est celui de la classification de grands volumes de données de tailles variables. L'état de l'art sur le sujet a été étudié et une nouvelle méthode pour palier aux manques des méthodes existantes a été introduit. Le chapitre précédent montre que la méthode EMMV - Ensemble de M-histogrammes Multi-Vues - propose une réponse efficace et robuste au problème. Il faut maintenant détailler la méthode sur un cas applicatif rencontré chez Michelin.

Dans ce chapitre, un exemple d'application de la méthode issu de Michelin R&D est exposé. Tout d'abord sont présentées les données utilisées, puis leurs caractéristiques et enfin les challenges qu'elles soulèvent. Pour finir l'application de la méthode sur ces données avec les résultats est présentée.

5.1 Données véhicules

Dans cette section sont définies les données sur les véhicules étudiés, ainsi que la raison pour laquelle cette étude se concentre sur ce type de véhicules. Enfin les données utilisées sont présentées.

5.1.1 Véhicules

Comme expliqué dans le chapitre *Introduction*, les données mises à disposition sont des données télématiques issues de capteurs installés sur divers véhicules. Dans la base de données, il y a plusieurs types de véhicules poids lourds, véhicules de tourisme, etc. Dans cette étude, seuls les véhicules de tourisme aussi appelés véhicules personnels sont considérés. C'est la voiture de Monsieur Toutlemonde. Cela pour deux raisons. Tout d'abord, les poids lourds ou véhicules de chantier sont soumis à un facteur poids très important. En effet, le poids d'une voiture varie globalement peu contrairement à un poids lourd où les variations s'expriment en tonnage. La deuxième raison est la distance parcourue par ces mêmes poids lourds est beaucoup plus importante et donc ils sont soumis à plus d'aléas routiers.

La voiture permet donc de limiter l'influence du poids et partiellement des aléas routiers.

5.1.2 Caractéristiques véhicules

Pour les 400 véhicules étudiés, nous disposons de la marque, du modèle et de la gamme de pneu installée.

La grande variété de véhicules de tourisme exploitée permet de s'intéresser à de nombreuses gammes de pneumatiques. En effet, l'objet de la classification ici est lié aux performances des pneus au sein de chaque gamme. Les véhicules de la base considérée possèdent donc tous 4 pneus, et entre véhicules, les dimensions de pneus sont variables et les propriétés différentes. Cette variété permet une vue globale de plusieurs performances pneumatiques dont l'usure.

Nous disposons donc de nombreux véhicules aux pneumatiques eux aussi variés. Nous allons maintenir définir les capteurs qui peuvent permettre d'obtenir de l'information sur l'usage de ces véhicules.

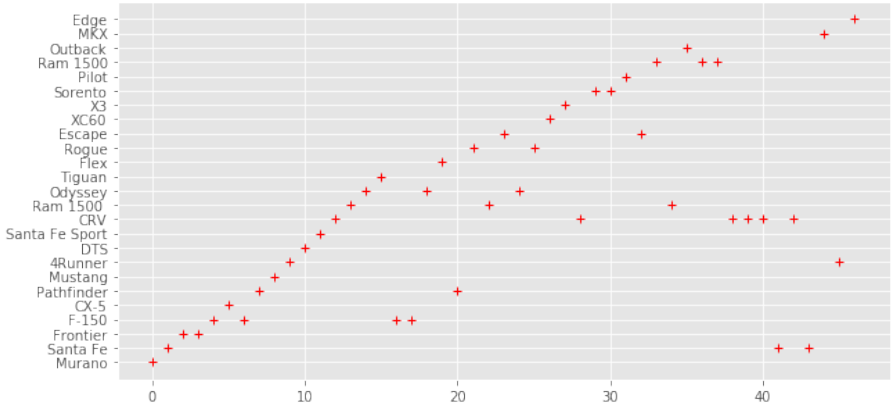


FIGURE 5.1 – Sous ensemble de voitures disponibles par modèle

5.1.3 Capteurs

Les données d'application peuvent être captées par différentes voies de mesures qui sont ensuite transférer via un boîtier centralisé, voir Fig. 5.2.

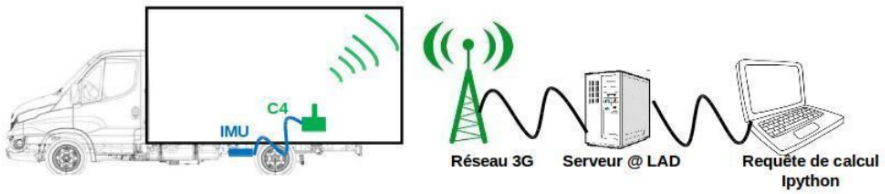


FIGURE 5.2 – Installation des capteurs et boîtiers sur véhicule.

Le boîtier télématique qui transmet l'information au serveur, est relié à plusieurs périphériques dont une centrale inertielle, un altimètre et le système Controller Area Network (CAN), voir Table 5.1. Le boîtier en lui-même contient le système Global Positioning System (GPS).

GPS

Le GPS est un système permettant d'identifier des positions géographiques grâce aux satellites. Dans ce cas d'application, il permet de connaître la latitude et la longitude d'un véhicule, mais aussi sa vitesse de croisière. La précision de ces données peut être corrigée en prenant en compte la qualité du signal transmis ainsi que le nombre de satellites utilisés pour établir le positionnement du véhicule. Ces données sont acquises à raison d'un point par seconde.

CAN

Le CAN est un réseau qui relie de petits calculateurs électroniques situés au sein du véhicule. Il permet de transmettre des signaux électroniques partout dans le véhicule, tels que l'allumage de la radio ou le rapport de vitesse engagé. Ce sont ses signaux qui sont remontés et analysés. À l'heure actuelle, où les véhicules contiennent

beaucoup de calculateurs - en moyenne 50 -, des milliers de signaux circulent dans le véhicule. Le boîtier ayant une capacité de transmission limitée, il permet la récupération d'une dizaine de signaux sur l'ensemble disponible. Ces signaux sont sélectionnés par ailleurs, et peuvent varier d'un véhicule à l'autre. Pour les véhicules de tourisme, chaque marque propose son propre système électronique et un codage des données qui circulent sur le réseau différent. Mais par souci de simplicité nous faisons un abus de langage et appelons les données acquises par ces bais, des données CAN.

Inertial Measurement Unit (IMU)

L' IMU ou centrale inertielle, permet la captation des accélérations en trois dimensions, ainsi que des vitesses angulaires et l'intensité du champ électromagnétique. Ce capteur est relié au reste des capteurs et au boîtier et peut être positionné indépendamment sur le véhicule. Ce capteur a d'ailleurs fait l'objet d'un travail annexe, car les données contiennent de très nombreuses incertitudes rendant leur exploitation délicate.

Capteur	Voies de mesures	Fréquence d'acquisition
GPS	longitude, latitude,vitesse	1Hz
CAN	14 signaux sélectionnés	jusqu'à 300Hz
IMU	accélérations, vitesses angulaires, intensités du champ magnétique	50 Hz
altimètre	altitude, température, pression	50 Hz

TABLEAU 5.1 – Synthèse des capteurs installés sur les véhicules.

Du fait de leur captation en situation courante, contrairement aux données qui servent de références dans plusieurs papiers BAGNALL et collab. [2018], les données contiennent un certain nombre d'anomalies supplémentaires, qu'il faut prendre en compte.

Premièrement, les données sont **non reproductibles**. En effet, chaque trajet est unique, car le trafic routier, l'environnement climatique ou encore l'humeur du conducteur influencent les données captées. De la même manière, les séries posséderont un **bruit de mesure** inhérent au contexte d'acquisition. De plus, les données peuvent aussi connaître des **anomalies** voire des **trous de mesure**. Le boîtier peut ne pas être dans une zone couverte par la téléphonie, et malgré la mémoire tampon, certains points de mesures peuvent être perdus.

Le contexte dans lequel sont récupérées les données utilisées en classification est défini. Il faut maintenant aborder ce qu'il nécessaire de classifier, ici l'usure.

5.2 Usure

Tout d'abord, est définie l'usure du pneumatique, qui est la **classe** des données que nous cherchons à discriminer par notre méthode, ainsi que les paramètres qui peuvent influencer celle-ci.

5.2.1 Définition

L'usure se définit comme la diminution de hauteur de gomme sur la totalité du pneu par intervalles de mesures dans le temps par rapport à la hauteur mesurée précédemment.

En particulier, l'usure est la moyenne de mesures faites en trois sillons sur le pneu comme le suggère la Fig. 5.3.

FIGURE 5.3 – Pneu Michelin Energy. Les mesures d'usure se font dans les trois sillons visibles sur le pneu.

Définition : Usure de pneumatique

Soit u_t un vecteur d'usures mesurées à un instant t sur un véhicule V. L'usure globale du véhicule est définie comme :

$$\mathrm{U} = [\mathrm{E}(u(1)), ..., \mathrm{E}(u(t), ..., \mathrm{E}(u(\mathrm{T}))]$$

où E(.) est la moyenne statistique sur le vecteur, et T est l'instant de fin de vie du pneu caractérisé par un changement de pneumatique.

Cette usure, qui est une valeur continue sera ensuite discriminée selon les besoins en classification des applications proposées ; c'est-à-dire que des intervalles de valeurs créant des groupes équilibrés peuvent être calculés. Et pour chaque groupe, une nouvelle étiquette peut être associée. Cette nouvelle étiquette devient alors la classe à discriminer, par exemple neuf/usé.

5.2.2 Paramètres influents

L'usure est soumise à plusieurs paramètres extérieurs, qui vont provoquer une usure plus importante, ou au contraire l'atténuer. L'ensemble de ces paramètres sont résumés dans la Fig. 5.4. Il y a cinq paramètres principaux, dont l'influence varie en

fonction du poids associé, qui sont le **véhicule**, la **saison**, la **route**, le **pneu** et le **circuit/conducteur**. Ces paramètres ont déjà fait l'objet d'une étude Michelin, présentée dans un rapport technique LE MAÎTRE et collab. [1998].

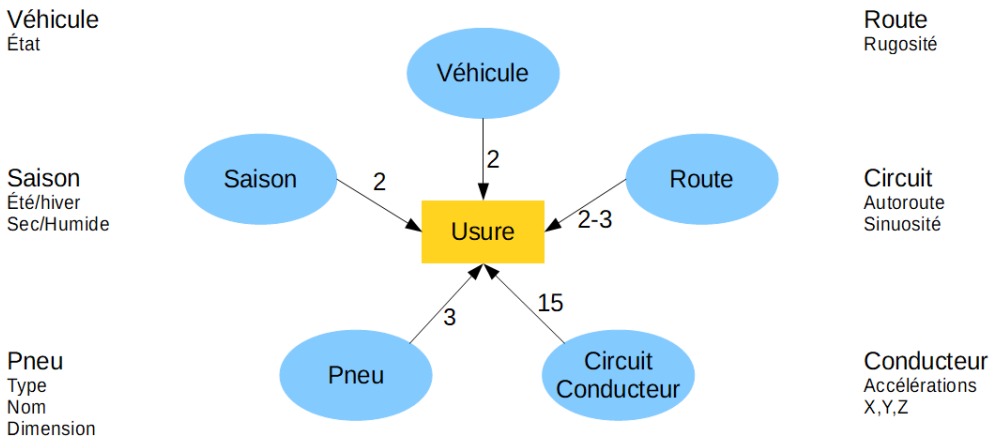


FIGURE 5.4 – Ensemble des paramètres d'influence de l'usure et leurs poids associés issu de LE MAÎTRE et collab. [1998]

Les saisons

Les saisons font simplement la différence ici entre été et hiver et notamment entre routes **sèches ou mouillées**. La différence est importante, car rouler sur une route mouillée n'use pas les pneumatiques contrairement à une route sèche. En effet, la pellicule d'eau représente une interface de protection entre le pneu et la route. De plus, la manière de conduire évolue avec la météo. Les profils de conduite ont tendance à être plus modérés sous la pluie.

La route

L'état de la route influence aussi l'usure. En effet, une route dite *rugueuse* peut arracher plus de gomme qu'une route lisse. De plus, un état de dureté du sol peut aussi être associé à un type de route. Les autoroutes et les routes pavés n'impliquent pas la même usure.

Le véhicule

Connaître le modèle de véhicule et donc le poids, le freinage, la puissance, etc. peut aussi permettre de mieux comprendre l'usure. Par ailleurs, ce paramètre prend aussi en compte l'état de celui-ci. Il est connu qu'un défaut de parallélisme peut provoquer, par exemple, une usure prématurée et anormale des pneumatiques.

Le pneumatique

Le type de pneus influence l'usure en elle-même. Cela peut être lié au type de gomme. Les gommes tendres s'usent plus vite que les gommes plus dures. Différents

pneus correspondent aussi à différents poids et usages de véhicules. Et enfin l'état du pneu en lui-même compte aussi. Un pneu sous ou surgonflé peut provoquer des usures prématurées aussi.

Circuit et conducteur

Finalement, le paramètre le plus influent est en fait une combinaison de deux paramètres indissociables : le circuit réalisé et le conducteur au volant. Le circuit représente à la fois le type de route, mais aussi les courbes ou virages rencontrés. Le conducteur représente en réalité la manière de conduire de ce dernier, une conduite économe et une conduite brutale sont par exemple deux styles de conduite différents. Ces deux paramètres évoluent ensemble, car par exemple, conduire de manière brutale sur autoroute provoque la même usure qu'une conduite économe sur route sinueuse. Ces deux paramètres sont captés et caractérisés par **les accélérations du véhicule**.

L'impact de chacun de ces paramètres est dans la réalité, très difficile à évaluer de manière dissociée. En effet, en réalité d'usage, l'intégralité de ces paramètres agissent ensemble et il n'est pas possible de dissocier leurs actions. De plus, dans le cadre de cette étude, aucune information permettant de les caractériser n'est disponible. C'est d'ailleurs l'un des enjeux de cette application. Peut-on définir l'usure à partir de données tronquées ?

La classe recherchée est définie, c'est-à-dire l'usure, tout comme les limites liées à celle-ci, c'est-à-dire les paramètres d'influence implicites. Il reste à aborder quelles sont les séries temporelles mises à disposition pour le faire.

5.3 Données disponibles

Les données disponibles pour l'analyse sont des données accélérométriques. Ces données sont calculées et non mesurées. En effet, les véhicules étudiés ici sont équipés exclusivement de GPS. Il faut donc utiliser des équations physiques afin de calculer les accélérations.

5.3.1 Accélérations longitudinale, latérale et verticale

Le GPS permet de remonter ici la vitesse du véhicule, sa position longitudinale et latérale, ainsi que l'altitude du véhicule. Les données sont remontées à raison d'un point par seconde. Ce sont ces données qui permettent de calculer les accélérations, de la manière suivante :

Définition : Accélérations calculées

Soit V(t), la vitesse du véhicule, P(t)=(Lon(t), Lat(t), Alt(t)), la position longitudinale et latérale et l'altitude du véhicule, où $t \in$ T le temps de mesure, ainsi que Cap(t) l'orientation relative par rapport au Nord, nous avons A_x, A_y, A_z, respectivement les accélérations longitudinale, latérale et verticales telles que :

$$A_x = \frac{\delta V}{\delta T}$$

$$A_y = \frac{V * \frac{\delta Cap}{\delta T}}{\delta T}$$

$$A_z = \frac{\delta Alt}{\delta T}$$

où δ est l'opération de première différence finie.

Ces formules font partie des formules d'approximation des accélérations tri-axiales, d'autres peuvent aussi être mises en place.

5.3.2 Incertitudes des données

Ces données ne sont pas de grande précision pour plusieurs raisons. La première est liée aux imprécisions du GPS. En effet, ce dernier ne remonte pas de mesures à la précision fine et ne remonte qu'un point par seconde. Cette granularité n'est pas suffisante pour mesurer un certain nombre de phénomènes, comme la présence de trous sur la chaussée.

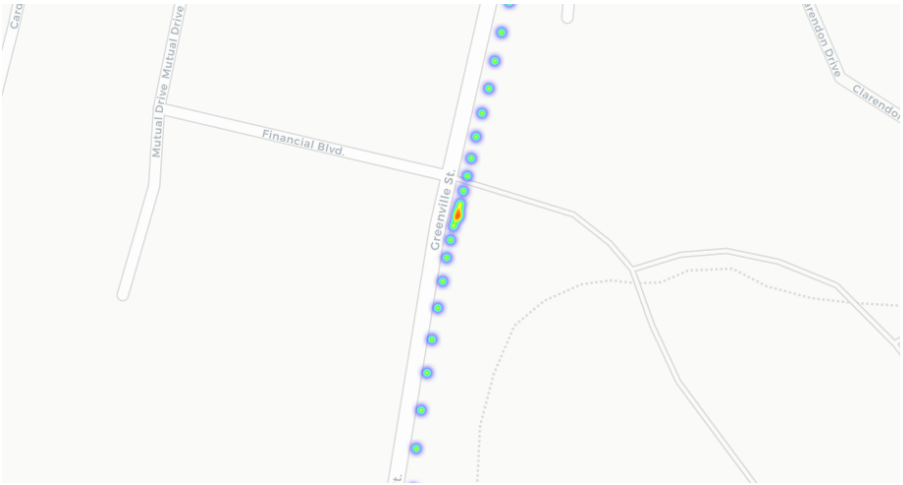


FIGURE 5.5 – Projection de la trace des positions longitudinales et latérales acquises par GPS sur carte routière. Nous pouvons voir que la trace projetée se trouve à coté de la route.

De plus, la précision de la position est de l'ordre du mètre, ce qui peut conduire à des erreurs de positionnement du véhicule sur la chaussée comme sur la Fig 5.5. Ces erreurs vont donc se retrouver dans les calculs effectués.

La deuxième part d'incertitude provient des calculs de l'accélération qui se font via des formules et calculs approximés. En effet, il existe plusieurs manières d'estimer les

accélérations impliquant des temps de calculs et des précisions finales différents. À cela, s'ajoutent les arrondis de stockage des valeurs calculées.

Finalement, des erreurs peuvent être liées aux valeurs manquantes dans les séries. Il faut réaliser dans ces cas une interpolation des valeurs qui peuvent en réalité différer de celles de l'événement réellement passé. Néanmoins, les données permettent de remonter des tendances routières suffisantes pour caractériser un événement routier. C'est-à-dire qu'il est possible d'extraire les virages, les freinages, les accélérations, les arrêts, etc.

5.3.3 Variations et nombre de points

Les données employées sont par ailleurs de tailles très variables. En effet, les séries sont enregistrées entre deux mesures d'usure réalisées. Or, l'usure n'est pas relevée à intervalle de temps régulier. Pas plus que le même nombre de kilomètres ou d'heures ne sont effectués dans les mêmes proportions, entre chaque relevé, voir Fig. 5.6.

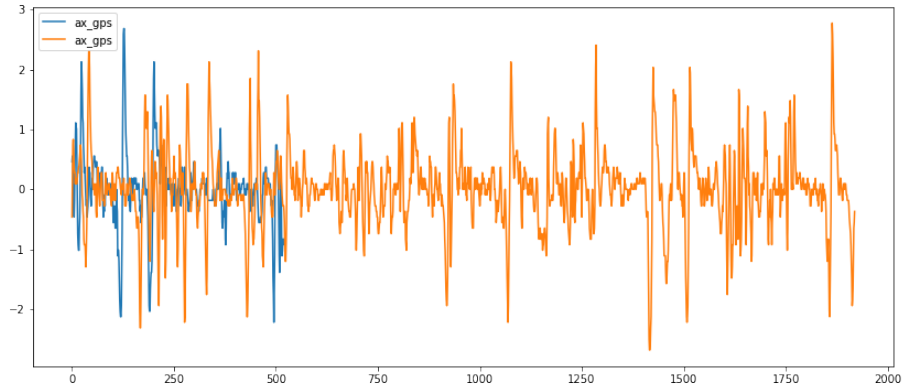


FIGURE 5.6 – Exemple d'accélérations enregistrées. Le voyage en bleu sur la figure a duré 5 minutes tandis que le voyage en orange a duré 33 minutes.

Une moyenne de 150000 points par série temporelle est enregistrée avec un maximum de 500000 points qui est une borne liée à la structure de la base de données.

Le contexte applicatif vient d'être défini ainsi que la classe recherchée et les séries temporelles à utiliser. Cette application somme les incertitudes, il faut donc maintenant définir les hypothèses et limites que ces incertitudes impliquent sur l'exécution de notre méthode.

5.4 Hypothèses

Au vu des données mises à disposition de nombreuses hypothèses s'appliquent ici. Ces hypothèses sont liées aux données, à la méthode, à l'application ainsi qu'à l'interprétation des résultats. Des postulats sont avancés, il faudra les corroborer ou non par les résultats.

5.4.1 Les accélérations seules sont suffisantes pour caractériser l'usure

La première hypothèse concerne l'apport des séries temporelles. Il est émis en effet ici l'hypothèse qu'**une STM des accélérations seule suffit à caractériser l'usure**. Le postulat sous-jacent est que la multitude de cas mesurés permet de rencontrer tous les cas d'influence des autres paramètres et donc n'ont pas besoin d'être mesurés explicitement. Un exemple est dans la figure suivante 5.7 dans le cas de la météo.

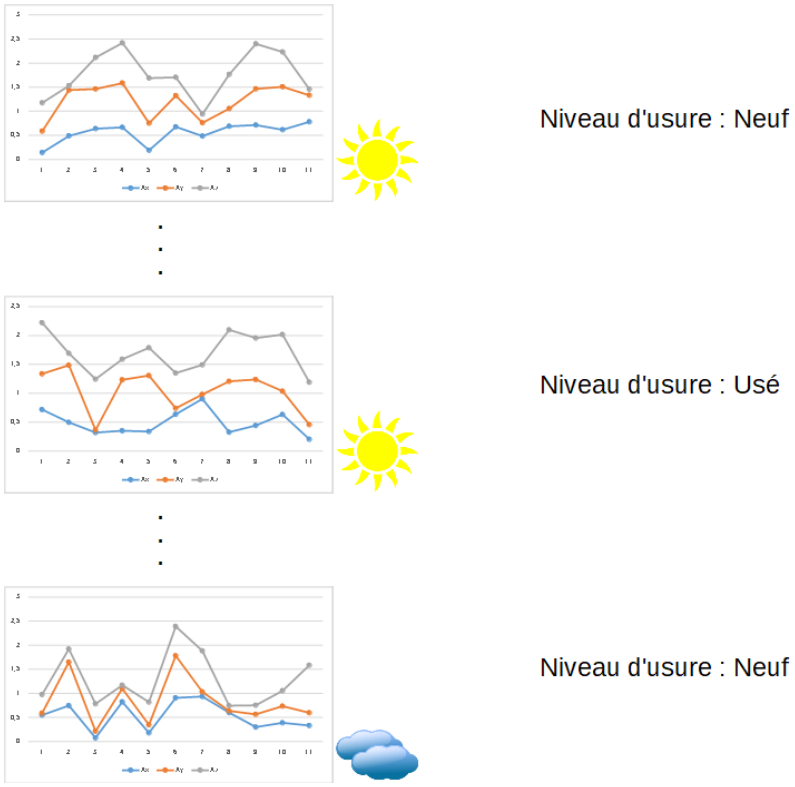


FIGURE 5.7 – Mesures d'accélérations et d'usures, l'influence sous-jacente de la météo

Par ailleurs, comme avec le paramètre d'influence *Circuit / Conducteur*, les limites d'effet des différents paramètres ne sont pas clairement établies. Plusieurs combinaisons de paramètres peuvent amener aux mêmes effets d'usure. L'hypothèse est donc que la multitude de trajets et de mesures d'usure peut compenser les défauts d'information sur l'ensemble des paramètres en apprentissage.

5.4.2 L'imprécision des données n'est pas significative

La deuxième hypothèse est liée à l'imprécision des accélérations exploitées comme expliquées ci-avant. Il est possible que **l'usure puisse se caractériser par l'extraction des phénomènes macro-routiers seuls**. Cela signifie qu'il est supposé que l'usure pneumatique se réalise au travers de grandes tendances routières plutôt que d'effet micro-routiers. Dans ce cas, il est estimé qu'une variation de l'accélération au centième près n'est pas significative pour l'usure, mais qu'une variation de plusieurs dixièmes si. C'est pourquoi la précision ne revêt pas un caractère indispensable ici au contraire d'autres applications comme la détection de trou dans la route.

5.4.3 Bien que l'usure réelle soit difficile à prévoir, des niveaux d'usure peuvent être discriminés

Au vu des données à disposition et du savoir Michelin, il parait peu probable de définir une usure fine pour notre problème. Il est tout de même probable de pouvoir **caractériser des niveaux d'usure**. Premièrement, l'usure d'un pneumatique n'est pas homogène, au contraire des différences peuvent être constatées en différents points du pneu. L'estimation ne peut donc pas être de granularité trop importante. Néanmoins, il est visible sur la Fig. 5.8 une nette tendance dans l'usure qui réalise une pente décroissante.

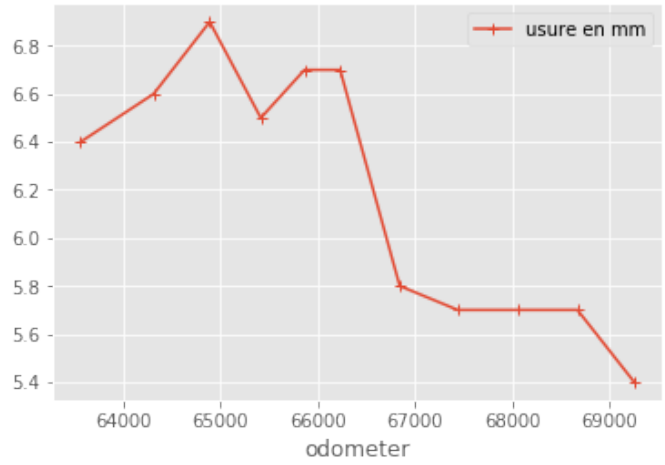


FIGURE 5.8 – Exemple de l'usure, par la hauteur de gomme en millimètre, d'un pneu au cours de son utilisation en fonction du nombre de kilomètres parcourus

Il est alors possible de définir des niveaux d'usure globales. Ceux-ci vont l'être avec plusieurs granularités de niveaux comme *neuf* / *usé* , ou encore, *neuf* / *mi-usé* / *usé*. Ces niveaux sont définis de manière à obtenir un système de classes équilibré.

5.4.4 Les bi-histogrammes et histogrammes seront suffisants pour discriminer les niveaux d'usure

Finalement, la dernière hypothèse est liée à l'application de la méthode. **Les bi-histogrammes et histogrammes étant déjà employés dans l'établissement de profils de conduite, ils représentent la meilleure alternative pour caractériser l'usure dans cette application**. En effet, comme vu ci-avant le paramètre circuit/conducteur est le paramètre le plus influent sur l'usure. Or, ce paramètre est déjà représenté sous forme de bi-histogrammes et histogrammes chez Michelin. Il semble donc légitime de les employer ici plutôt qu'une représentation M-histogramme globale par exemple.

La méthode EMMV est donc employée ici, c'est-à-dire un ensemble multi-vues de bi-histogrammes et histogrammes sur les données originales, dérivées et intégrées. Cet ensemble permet ensuite l'apprentissage de classifieurs 1NN avec un vote final majoritaire afin de définir le niveau ou classe d'usure.

L'ensemble de l'application ainsi que les hypothèses à prendre en compte dans l'exécution de la méthode viennent d'être définis. Il reste maintenant à aborder les résultats obtenus.

5.5 Résultats

Avant de présenter les résultats, quelques précisions sur l'expérimentation vont être apportées.

5.5.1 Précisions sur les résultats présentés

Il est important de synthétiser quelques points abordés précédemment tels que les données, les classes et la méthode appliquée afin de bien comprendre la portée des résultats présentés ci-après.

Les données

Les STM employées ici sont donc les accélérations tri-axiales d'un véhicule calculées à partir de données GPS. Ces données ont été captées aux États-Unis lors de l'usage réelle de véhicules personnels par des boîtiers télématiques low cost. Les conducteurs des véhicules étaient ensuite amenés à mesurer l'usure de leurs pneumatiques via le système de mesure *Hunter*. Ce système nécessite le passage du véhicule dans un tunnel dans lequel des lasers sont déployés au sol. Cette mesure laser est fiable et reproductible. Les STM étudiées sont donc constituées de l'ensemble des accélérations enregistrées par véhicule entre deux mesures d'usure.

Les classes

Dans l'ensemble des sections résultats ci-après, sauf indication contraire, les classes testées sont au nombre de deux : neuf/usé. Nous rappelons que la frontière entre les deux - c'est-à-dire le seuil qui fait basculé l'usure du pneu dans la catégorie neuf ou usé - est obtenue via une méthode statistique afin d'équilibrer les classes.

La méthode

Les résultats présentés ne résultent que de l'application de la méthode EMMV, c'est-à-dire que les deux méthodes de référence SMTS, ni WEASEL+MUSE n'ont pas été appliquées sur ces données. En effet, les complexités des méthodes qui sont de respectivement $O(J_{ins}\sqrt{2M+1}NT(R-1))$ suivi de $O(J_{ts}\sqrt{RJ_{ins}}NlogN)$ pour SMTS et de $O(min[NT^3, N^2T^2]*m)$ pour WEASEL+MUSE, ne permettent pas leurs applications sur le volume de données exploité ici.
En effet, SMTS est une méthode où de trop nombreux paramètres sont à régler et aucune indication sur les initialisations n'est précisée. De fait, cette méthode est trop longue à apprendre pour cette application. WEASEL+MUSE a quant à elle une complexité bien trop grande pour être appliquée sur les données en raison des ordres de grandeur des données disponibles. C'était déjà le cas, dans le chapitre sur les données de références, c'est toujours un problème ici.

5.5.2 Usure par modèle de véhicules

Dans cette section sont donc définis les résultats obtenus en prenant en compte le modèle de véhicule. Il peut y avoir ici plusieurs gammes de pneus différentes pour un même modèle de véhicules. Les modèles de voitures pour lesquels trop peu d'indications d'usure ont été remontées, ont été supprimés, c'est-à-dire que les modèles pour lesquels moins de 20 mesures d'usure sont disponibles.

Résultats en fonction du nombre de séries

Les résultats ici sont les projections des classes de niveaux d'usure en fonction du nombre de séries disponibles dans chaque jeu de données testé, voir Fig. 5.9.

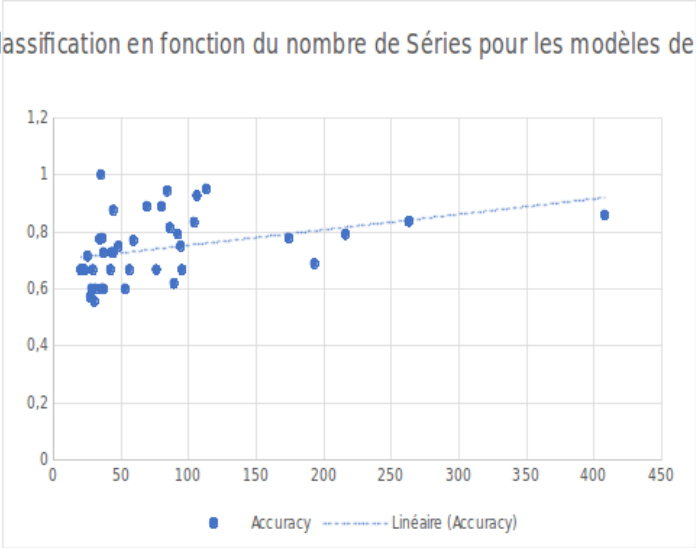


FIGURE 5.9 – Impacte du nombre de séries sur la classification de l'usure.

Il est visible sur la figure que le nombre de séries influe sur les résultats obtenus en taux de classification. **Plus il y a de séries disponibles par modèle de véhicule plus le taux de classification est grand**. Ce résultat permet de confirmer que la méthode fonctionne mieux dans des cas de grands jeux de données. Par ailleurs, cela rejoint la première hypothèse liée à cette application. Il faut avoir accès à de nombreux individus pour classifier l'usure sans exprimer explicitement tous les paramètres qui influent dessus.

Pour ce qui est de l'interprétation, cette figure montre aussi que le taux de prédiction de la classe neuf/usé est plus élevé pour un modèle de véhicule bien représenté dans la base. C'est-à-dire que beaucoup de mesures ont été remontées à son propos.

Résultats en fonction du niveau d'usure

Les résultats présentés ici sont les projections des classes de niveaux d'usure en fonction de la frontière entre les deux dites classes, voir Fig. 5.10. Il est à rappeler ici que l'usure mesurée est une valeur discrète qui est ensuite catégorisée à l'aide d'un

outil de découpage statistique. Celui ci détermine le seuil entre neuf/usé permettant d'avoir le même nombre d'individus dans chaque classe.

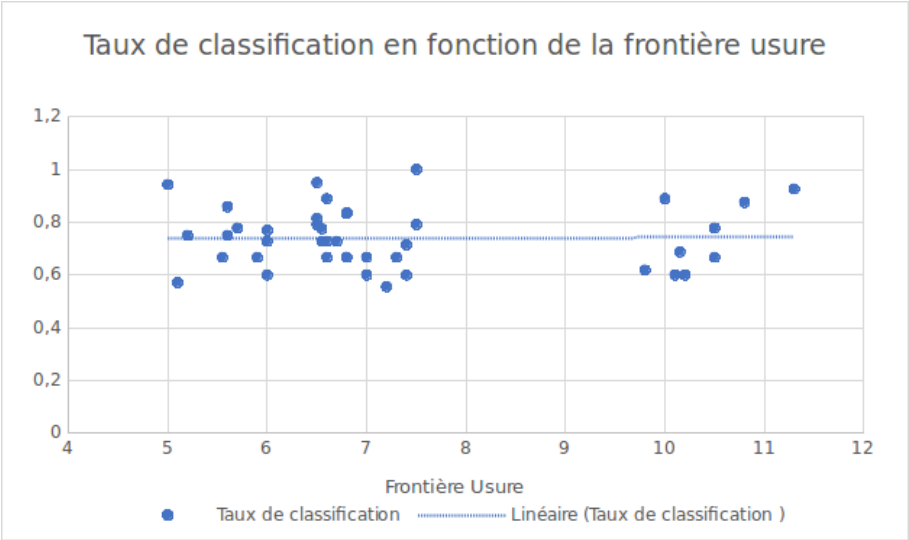


FIGURE 5.10 – Impacte de la frontière entre les deux classes d'usure sur la classification de celles-ci.

La figure montre que l'établissement de cette frontière qui varie en fonction des gammes de pneus n'influe pas sur les performances de la méthode. Le modèle est donc robuste à ce genre de considération tant que les jeux de données sont équilibrés en éléments par classe.

Résultats en fonction du nombre de véhicules et du nombre de gammes de pneus

Les classes de niveaux d'usure sont ensuite projetées en fonction du nombre de véhicules disponibles dans chaque jeu de données testé, voir Fig. 5.11.

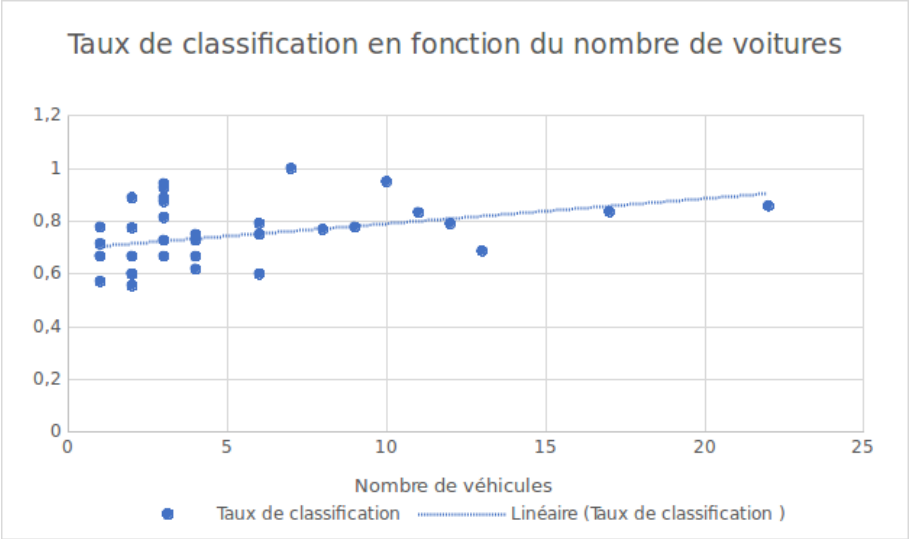


FIGURE 5.11 – Impacte du nombre de véhicules sur la classification de l'usure.

Il semble ici que le nombre de véhicules disponibles influence les résultats. **Plus il y a de véhicules disponibles plus le taux de classification est grand**. Ce résultat permet de confirmer lui aussi l'hypothèse 1 décrite ci-avant. La variété des données disponibles permet malgré la définition absente des différents paramètres d'usure de les décrire implicitement.

Pour conclure, la méthode est sensible à ce genre de configuration et fonctionne mieux dans des cas de grand jeu de données.

Des résultats identiques sont trouvés dans l'étude en fonction du nombre de gammes de pneus, voir Fig. 5.12.

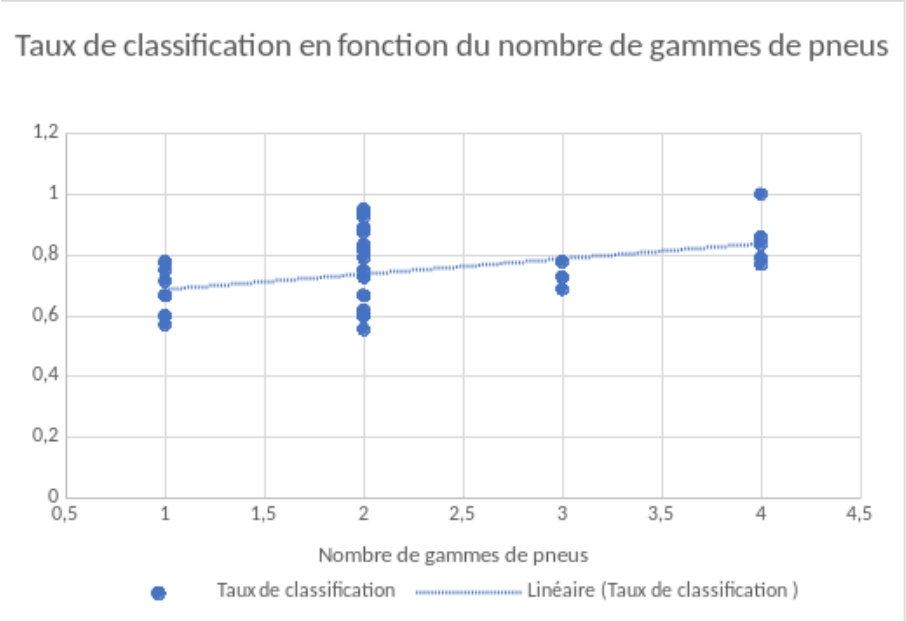


FIGURE 5.12 – Impacte du nombre de gammes de pneus sur la classification de l'usure.

Les résultats de la méthode dépendent donc pour beaucoup de la quantité de données disponibles. Il est aussi souhaitable de voir si le nombre de classes augmente comment varient les performances.

5.5.3 Augmentation du nombre de classes

L'impact de l'augmentation du nombre de classes est donc testés sur plusieurs modèles de véhicules, voir Table 5.2

	Modèle	Pneu	2 classes	3 classes	5 classes
1	F-150	Latitude Tour HP	0.7	0.7778	0.6667
2	Frontier	Latitude Tour HP	0.875	0.8667	0.8571
3	Ram 1500	Latitude Tour HP	1.0	1.0	0.8
4	Sienna FWD	Premier LTX	0.8889	0.875	0.875

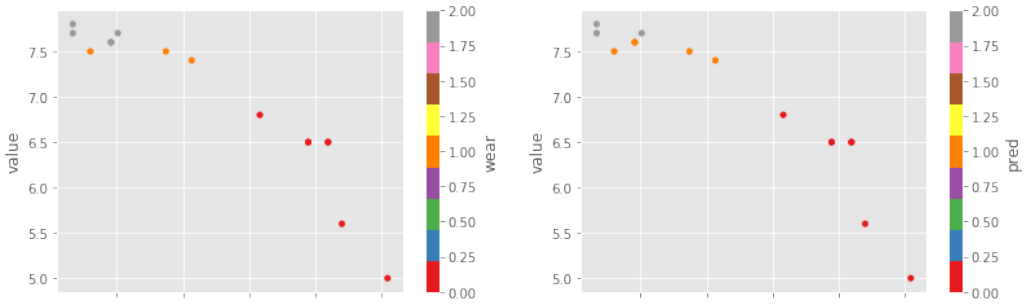
TABLEAU 5.2 – Taux de classification par rapport aux nombres de classes

Sur des jeux de données ayant donnés les meilleurs taux de classification avec deux classes, voir colonne *2 classes* de la Tab. 5.2, les résultats pour deux augmentations de classes ont été compilés. Il est difficile de tirer des conclusions claires de ce tableau. Cependant, le taux de classification semble décroître avec le raffinement des classes d'usure. Cela rejoint le postulat émis dans la section *hypothèses*. L'incertitude contenue dans les mesures d'usure ne semble pas permettre une prédiction fine de cette dernière. D'autres expérimentations devront être conduites afin de tester un plus grand nombre de jeux de données.

Les résultats de la méthode dépendent des données disponibles mais aussi de leurs qualités. En effet, à l'heure actuelle, il semble difficile de classifier finement l'usure des pneumatiques. Il faut maintenant étudier les sources d'erreurs de classification possibles afin d'améliorer les performances futures.

5.5.4 Classification erronée

Deux raisons principales semblent expliquer ici les erreurs de classifications de la méthode. La première concerne les cas limites. Il est rappelé ici que l'usure mesurée est une variable discrète. Les mesures sont découpées artificiellement en catégories ou classes, usé/neuf par exemple. Avec ce découpage toute classe possède le même nombre d'individus, mais la gestion des valeurs à la limite entre les deux classes n'est pas claire. Donc deux tuples ayant une valeur d'usure proche peuvent appartenir à deux classes différentes. Il y a alors un risque de mauvaise classification, voir Fig. 5.13.



(a) Classes d'usure réelles (b) Classes d'usure prédites

FIGURE 5.13 – Comparaison entre les classes réelles et prédites des données tests

La deuxième raison est contextuelle et est liée au pneu lui-même. En effet, les pneus sur les véhicules peuvent être changés pour diverses raisons comme un changement été/hiver, l'usure ou encore des défauts sur le pneu. Ces changements ne sont à l'heure actuelle pas signalés dans la base. De la même manière, un changement de pneu ne conduit pas automatiquement à une mesure d'usure. Ce cas est observable sur la Fig. 5.14, où le point très bas est classé comme un comportement de l'autre classe. Il est possible qu'il y ait eu un changement de pneu, cela ne peut être vérifié que dans le futur.

Ces deux cas limites conduisent à un ensemble de faux positifs et faux négatifs par la méthode dont il est difficile d'imputer les erreurs à la méthode.

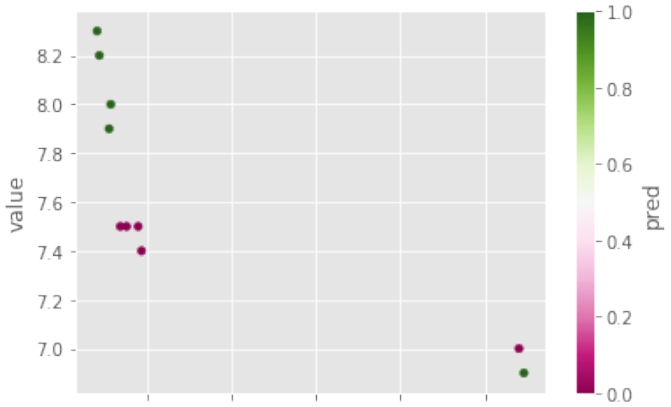


FIGURE 5.14 – Exemple de faux positif qui pourrait être lié à un changement de pneumatique

La méthode vient d'être appliquée dans le cadre de la recherche de l'usure chez Michelin. Malgré de nombreuses incertitudes, la méthode donne des résultats non négligeables de classification de l'usure. Il faut maintenant synthétiser cette application.

5.6 Résumé

Ce type de jeux de données nécessite une grande part d'interprétation liée à la disponibilité d'un expert pour tirer des conclusions claires. En effet, les résultats proposés par la méthode semblent faire sens. La méthode permet ainsi de mettre en relation usage et usure de manière rapide et efficace, tout en gérant les différences de tailles entre les séries. Néanmoins, des erreurs subsistent et il est difficile d'émettre des conclusions claires. Les erreurs de classification peuvent à l'heure actuelle être liées à des cas limites comme expliqué ci-avant. Tout comme liées aux paramètres d'influence de l'usure non définis explicitement (la rugosité de la route ou la météo). Les résultats montrent que la majorité des hypothèses liées à l'application de la méthode se vérifient. En effet la méthode, composée de bi-histogrammes et histogrammes, semble capable de discriminer des niveaux d'usure bien que nos données contiennent un certain nombre d'incertitudes. Tout comme il faut avoir accès à de nombreux cas pour pouvoir limiter les effets non explicites des paramètres d'influence de l'usure.

Du point de vue de l'interprétation, des expérimentations supplémentaires doivent être réalisées. Néanmoins, les résultats obtenus, de part leurs valeurs non-négligeables montrent que la méthode peut être efficace et rapide pour la classification de séries temporelles multivariées de longueurs variables.

Points clefs du chapitre Applicatif Michelin

Conclusions	— La méthode trouve un lien Usage/Usure — Les M-histogrammes réduisent les séries temporelles sans perdre l'information nécessaire à la classification. — Les M-histogrammes permettent de gérer l'incertitude contenue dans cette application — La méthode peut gérer des séries avec très grandes variations de points
Question ouverte	— Dans quelles mesures des paramètres contextuels non exprimés influence la méthode ?

L'ensemble du travail effectué vient d'être exposé, il faut maintenant résumer cet ensemble et en tirer les grandes conclusions de la thèse.

5.7 Références

BAGNALL, A., H. DAU, J. LINES, M. FLYNN, J. LARGE, A. BOSTROM, P. SOUTHAM et E. KEOGH. 2018, «The UEA multivariate time series classification archive, 2018», *CoRR*, vol. abs/1811.00075. URL http://arxiv.org/abs/1811.00075. 102

LE MAÎTRE, O., M. SÜSSNER et C. ZARAK. 1998, «Evaluation of tire wear performance», dans *SAE Technical Paper*, SAE International. URL https://doi.org/10.4271/980256. 104

Chapitre 6

Conclusion

Sommaire

Le premier chapitre de cette thèse expose la problématique qui est **la classification de grands volumes des STM de longueurs variables** et pose la question de savoir si le M-histogramme peut y apporter une réponse. Le deuxième chapitre présente l'état de l'art autour de la question, cependant les modèles proposés ne permettent pas la gestion rapide de grands volumes de données. Par ailleurs, le traitement des variations de longueurs n'est que succinctement abordé. C'est pourquoi, dans le troisième chapitre, une nouvelle méthode EMMV est proposée. Cette méthode combine deux concepts nouveaux dans le domaine de la classification des STM que sont l'apprentissage multi-vues et la représentation par M-histogramme. Le quatrième chapitre montre alors que la méthode proposée est robuste et efficace sur les données de références. Tandis que le cinquième chapitre permet de montrer la capacité de la méthode à extraire l'information sur un grand volume de séries de tailles variables et données incertaines.

Cette conclusion a pour but de synthétiser toutes les informations apportées précédemment. Tout d'abord, l'ensemble du travail effectué est résumé et les notions au coeur de la thèse sont rappelées. Puis, les publications et les soumissions qui ont eu lieu au cours de cette thèse sont exposées. Finalement, des perspectives sont aussi abordées comme suite possible de l'ensemble des travaux mis en place.

6.1 Travail effectué

Cette thèse introduit une nouvelle méthode EMMV - Ensemble de M-histogrammes Multi-Vues - qui permet la classification rapide et efficace de STM à longueurs variables. Les concepts mis en œuvres pour y arriver sont résumés ici.

M-histogramme

Pour la première fois, l'outil M-histogramme statistique est exploité afin de classifier les STM. Comme déjà précisé, un outil similaire appelé bigramme, mais qui compte uniquement des couples de mots a déjà été utilisé. Néanmoins, l'outil statistique général ne l'a jamais été. Cela représente notre idée de recherche principale. Cette représentation permet de réduire les séries temporelles tout en mettant en exergue les relations entre les dimensions et les relations temporelles. Pour cela et afin de contrôler l'ordre temporel, les notions de dérivées et de sommes cumulées sont introduites.

Multi-vues

Par ailleurs, l'emploi d'un modèle multi-vues associé à la représentation par M-histogrammes n'avait, à notre connaissance, jamais été exploité. Il était donc opportun de le faire ici.

En outre, la définition de trois vues à partir des séries originales, dérivées et sommes cumulées n'a elle non plus jamais été utilisée. Pourtant les expérimentations effectuées ici montrent que chacune de ses vues peut s'avérer déterminante pour la classification de certains jeux de données.

Solution aux points durs

Enfin, la méthode EMMV est capable de gérer de très grandes séries. En effet, contrairement aux jeux de données et modèles de références, les séries exploitées dans l'applicatif Michelin sont composées de centaines de milliers de points. La méthode est capable de traiter ces données rapidement et efficacement. Par ailleurs, pour les séries atteignant de plus grandes tailles, les variations entre ces dernières sont aussi plus importantes que celles constatées dans l'état de l'art. La méthode montre donc aussi sa capacité à comparer entre elles des séries qui sont pourtant de taille très différente.

Pour résumer, est proposée dans cette thèse, **une méthode rapide et efficace** de classification des STM adapté aux séries de tailles très longues et variables. Cette méthode repose sur des principes de multi-vues et de transformation des séries par l'usage des M-histogrammes. L'ensemble des publications reposant sur ces travaux de recherche sont présentées ci-après.

6.2 Publications et présentations

Plusieurs publications et présentations ont été réalisées au cours de cette thèse.

La première s'est déroulée dans le cadre de la conférence Société d'Ingénieurie Automobile française (SIA) qui a eu lieu à Paris en 2017. L'objectif était de présenter le début des travaux sur la classification de séries temporelles multivariées afin de plaider pour le libre-échange des données au sein de la communauté automobile.

La deuxième présentation des travaux, a été réalisée lors d'un atelier European Conference of Machine Learning (ECML), réservé aux doctorants afin de présenter leurs travaux de thèse. Une première version du travail a été présentée sur l'utilisation des M-histogrammes.

Ensuite, le travail a été publié et présenté lors de la conférence Conférence de l'Apprentissage Automatique (CAp) 2019. La méthode finale avec les résultats sur des jeux de données de références a été publié.

Finalement, l'ensemble du travail a été présenté dans le cadre de l'atelier Flux de Données et Séries Temporelles (FDST) 2019. Celui-ci est un nouvel atelier spécialisé, dont la première édition s'est tenue en septembre, sur *l'apprentissage à partir de flux de données et séries temporelles.*

Soumissions

Nous avons aussi soumis une version étendue du travail dans le journal Journal of Machine Learning and Cybernetics (JMLC).

6.3 Perspectives

De cette proposition de méthode, de nouvelles questions découlent, il reste donc à aborder ici des éléments de réponses et des initiatives à explorer.

6.3.1 Ordre et série temporelle

L'une des premières questions soulevées est celle de l'importance de l'ordre des points au sein d'une série temporelle. En effet, la méthode de transformation des séries temporelles par l'usage des M-histogrammes, ne tient pas compte de cette notion. C'est pourquoi, dans la méthode, est mis en place le système à trois vues composé des séries initiales, mais aussi des séries dérivées et intégrées. Il a été montré que l'intégration par l'usage de la somme cumulée permet de limiter la perte de l'ordre lors de la classification, permettant d'améliorer les taux de classifications dans de nombreux cas. Il pourrait donc être intéressant d'étudier la notion d'ordre et son impact dans sa globalité.

Granularité

Tout d'abord, en évaluant l'ordre des points. Quels résultats obtenons-nous en retirant toute notion d'ordre ? Que se passe-t-il si nous échangeons des séquences de points ? Existe-t-il des cas nécessitant l'ordre complet des points ? Ces questions pourront faire l'objet d'une étude approfondie. Bien qu'il ait été montré au travers des méthodes de l'état de l'art que l'ordre pouvait avoir de l'importance, une étude approfondie du sujet reste à être effectuée. Des conclusions claires à ce sujet et d'autant plus pour le cas des STM restent à être établies.

Technique de réduction

Enfin, existe-t-il une autre technique que celle de la somme cumulée permettant de récupérer l'ordre des points afin de compléter les modèles où cette notion est perdue ?

En effet, la somme cumulée possède une limite liée au nombre de points par série. Dans le cas où la taille d'une série tend vers l'infinie alors l'impact d'une variation de valeurs de la série est infime par rapport à la somme de tous les points de la série. Il peut être alors judicieux de réfléchir à une autre alternative afin d'extraire la notion d'ordre contenue dans les données.

6.3.2 Dépendance des dimensions

Une autre perspective concerne le choix fait, dans la méthode, de traiter la sélection des dimensions de manière aléatoire en tenant compte du nombre de combinaisons possibles. Dans la suite de cette thèse, il peut être pertinent, comme dans le cas de séries avec de très nombreuses dimensions, de développer en amont de la sélection, un algorithme d'analyse plus poussé des dépendances entre dimensions. L'objectif est alors d'éliminer les redondances, mais surtout de réaliser une pré-sélection des dimensions qui permettraient une meilleure classification.
L'idée peut être mise en place, par exemple, à travers l'étude d'un réseau bayésien Naïm et collab. [1999]. Ce dernier permet l'étude des liens entre des variables aléatoires. Il est donc possible de l'appliquer ici dans le but de mieux comprendre les imbrications entre dimensions.

6.3.3 Théorie de l'apprentissage multi-vues

Au cours de cette thèse, l'efficacité de l'apprentissage multi-vues a été montrée du point de vue expérimental. La continuité du travail présenté ici pourrait donc aussi être de réaliser une étude théorique sur cet apprentissage. Notamment, en se basant sur des travaux comme GOYAL et collab. [2019] qui montre les avantages de l'apprentissage multi-vues dans les systèmes ensemblistes de type boosting. De manière générale, l'apprentissage multi-vues reste peu exploité et méconnu, une étude formelle de la théorie ainsi que les preuves expérimentales pourraient amener un plus large consensus de son utilisation au sein de la communauté scientifique.

6.3.4 Emploi du deep learning

Un autre point qui n'est pas abordé précédemment est la projection des M-histogrammes qui peut ensuite être visualisée sous forme d'image. Il existe aujourd'hui de nombreuses méthodes de classification d'images qui ont fait leurs preuves et notamment celles issues du *deep learning*.

Deep learning

Le deep learning fait partie des méthodes d'apprentissage supervisées qui permettent la classification des données LECUN et collab. [2015]. Celui-ci a connu un essor retentissant en 2012, en permettant une large amélioration de la classification d'images. Le deep learning est en réalité une catégorie de réseaux de neurones qui sont aujourd'hui très appréciés dans leur globalité et utilisés dans de nombreux domaines de recherches et applicatifs.

Cette catégorie se différencie des autres, littéralement par son *apprentissage profond*. Cela veut dire qu'un réseau de neurones en apprentissage profond est constitué de très nombreuses couches de neurones permettant une grande abstraction des données initiales.

Ces modèles plus évolués qu'un simple classifieur des plus proches voisins nécessitent aussi une phase d'entraînement importante. Néanmoins, dans un souci de compromis entre temps de calcul et taux de performances, il peut être intéressant d'explorer cette piste.

Application

Il est donc possible d'appliquer un réseau de neurones pour chaque image créée à partir des séries. Le même schéma de méthode que celui présenté dans le document peut alors être employé. Des réseaux de neurones pourraient alors remplacer les classifieurs 1NN. Cela représente néanmoins l'inconvénient d'être très long à entraîner.

Une autre possibilité envisageable est de créer une image globale à partir des différents M-histogrammes assemblés. Il ne faut alors mettre en place plus qu'un seul classifieur neuronal qui peut analyser l'image dans son entièreté tout comme chaque M-histogramme sous-jacent. Cette application est facilement envisageable comme suite de cette thèse.

6.4 Références

GOYAL, A., E. MORVANT, P. GERMAIN et M.-R. AMINI. 2019, «Multiview boosting by controlling the diversity and the accuracy of view-specific voters», *Neurocomputing*, vol. 358, doi :https://doi.org/10.1016/j.neucom.2019.04.072, p. 81 – 92, ISSN 0925-2312. 121

LECUN, Y., Y. BENGIO et G. HINTON. 2015, «Deep learning», *nature*, vol. 521, nº 7553, p. 436. 121

NAÏM, P., P.-H. WUILLEMIN, P. LERAY, O. POURRET et A. BECKER. 1999, «Réseaux bayésiens», *Eyrolles, Paris*, vol. 3. 120

Annexe A

Séries temporelles

Exemple de l'utilité de la dérivée est de la somme cumulée

Les exemples illustrent l'intérêt de calculer la dérivée et la somme cumulée de la série initiale.

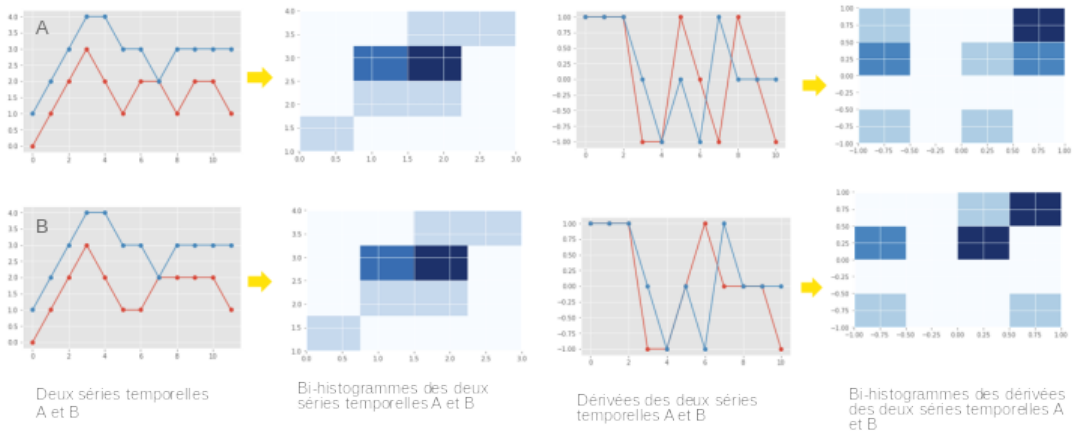


FIGURE A.1 – Exemple du bi-histogramme de dérivée. Le bihistogramme des séries sur la gauche, n'est pas suffisant pour différencier les deux séries. Par contre le bi-histogramme de la dérivée l'est.

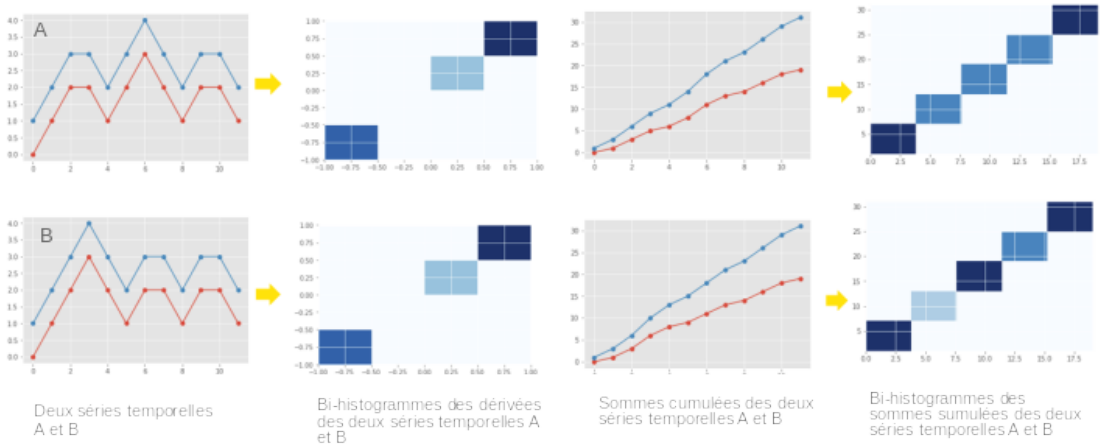


FIGURE A.2 – Exemple du bi-histogramme de somme cumulée. Le bihistogramme des dérivées des séries sur la gauche, n'est pas suffisant pour différencier les deux séries. Par contre le bi-histogramme de la somme cumulée l'est.

Exemple de séries temporelles

Cette annexe représente les données qui sont derrière l'exemple illustratif du modèle présentées Chapitre 3.

Séries temporelles

Les séries temporelles composant D_{app} etD_{test} sont :

$$X_1 = \begin{pmatrix} 0.20 & 0.36 & 0.42 & 0.55 & 0.60 & 0.76 & 0.88 & 0.62 & 0.39 & 0.17 \\ 0.12 & 0.19 & 0.31 & 0.42 & 0.53 & 0.61 & 0.69 & 0.81 & 0.91 & 0.98 \\ 0.98 & 0.91 & 0.81 & 0.69 & 0.61 & 0.53 & 0.42 & 0.31 & 0.19 & 0.12 \\ 0.22 & 0.42 & 0.61 & 0.83 & 0.64 & 0.39 & 0.21 & 0.41 & 0.63 & 0.82 \end{pmatrix}$$

$$X_2 = \begin{pmatrix} 0.17 & 0.42 & 0.63 & 0.91 & 0.76 & 0.62 & 0.55 & 0.41 & 0.29 & 0.18 \\ 0.12 & 0.19 & 0.31 & 0.42 & 0.53 & 0.54 & 0.62 & 0.72 & 0.81 & 0.89 \\ 0.89 & 0.81 & 0.72 & 0.62 & 0.54 & 0.53 & 0.42 & 0.31 & 0.19 & 0.12 \\ 0.21 & 0.41 & 0.59 & 0.82 & 0.61 & 0.42 & 0.19 & 0.42 & 0.66 & 0.81 \end{pmatrix}$$

$$X_3 = \begin{pmatrix} 0.21 & 0.35 & 0.39 & 0.53 & 0.64 & 0.78 & 0.91 & 0.59 & 0.43 & 0.23 \\ 0.11 & 0.22 & 0.33 & 0.41 & 0.49 & 0.62 & 0.71 & 0.82 & 0.89 & 0.97 \\ 0.97 & 0.89 & 0.82 & 0.71 & 0.62 & 0.49 & 0.41 & 0.33 & 0.22 & 0.11 \\ 0.21 & 0.39 & 0.57 & 0.82 & 0.61 & 0.40 & 0.22 & 0.39 & 0.61 & 0.80 \end{pmatrix}$$

$$X_4 = \begin{pmatrix} 0.15 & 0.35 & 0.38 & 0.57 & 0.64 & 0.69 & 0.92 & 0.61 & 0.40 & 0.16 \\ 0.11 & 0.21 & 0.29 & 0.44 & 0.52 & 0.64 & 0.72 & 0.83 & 0.94 & 0.99 \\ 0.99 & 0.94 & 0.83 & 0.72 & 0.64 & 0.52 & 0.44 & 0.29 & 0.21 & 0.11 \\ 0.20 & 0.39 & 0.63 & 0.80 & 0.62 & 0.41 & 0.22 & 0.42 & 0.61 & 0.81 \end{pmatrix}$$

$$X_5 = \begin{pmatrix} 0.20 & 0.44 & 0.65 & 0.87 & 0.78 & 0.63 & 0.54 & 0.42 & 0.33 & 0.22 \\ 0.12 & 0.21 & 0.33 & 0.39 & 0.50 & 0.51 & 0.61 & 0.73 & 0.79 & 0.91 \\ 0.91 & 0.79 & 0.73 & 0.61 & 0.51 & 0.50 & 0.39 & 0.33 & 0.21 & 0.12 \\ 0.19 & 0.43 & 0.60 & 0.79 & 0.65 & 0.40 & 0.19 & 0.43 & 0.59 & 0.79 \end{pmatrix}$$

$$X_6 = \begin{pmatrix} 0.18 & 0.45 & 0.66 & 0.85 & 0.80 & 0.65 & 0.52 & 0.44 & 0.31 & 0.21 \\ 0.11 & 0.20 & 0.28 & 0.39 & 0.51 & 0.52 & 0.64 & 0.71 & 0.82 & 0.89 \\ 0.89 & 0.82 & 0.71 & 0.64 & 0.52 & 0.51 & 0.39 & 0.28 & 0.20 & 0.11 \\ 0.21 & 0.43 & 0.59 & 0.81 & 0.61 & 0.39 & 0.22 & 0.39 & 0.62 & 0.78 \end{pmatrix}$$

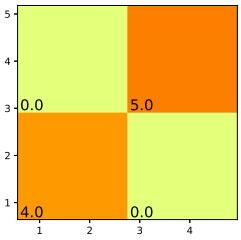
$$Y_1 = \begin{pmatrix} 0.22 & 0.37 & 0.41 & 0.51 & 0.59 & 0.74 & 0.92 & 0.63 & 0.42 & 0.19 \\ 0.13 & 0.22 & 0.34 & 0.43 & 0.49 & 0.63 & 0.72 & 0.83 & 0.92 & 0.99 \\ 0.99 & 0.92 & 0.83 & 0.72 & 0.63 & 0.49 & 0.43 & 0.34 & 0.22 & 0.13 \\ 0.22 & 0.41 & 0.59 & 0.81 & 0.61 & 0.42 & 0.23 & 0.40 & 0.61 & 0.82 \end{pmatrix}$$

$$Y_2 = \begin{pmatrix} 0.21 & 0.45 & 0.66 & 0.88 & 0.79 & 0.62 & 0.51 & 0.38 & 0.35 & 0.22 \\ 0.12 & 0.21 & 0.34 & 0.45 & 0.55 & 0.56 & 0.63 & 0.73 & 0.83 & 0.91 \\ 0.91 & 0.83 & 0.73 & 0.63 & 0.56 & 0.55 & 0.45 & 0.34 & 0.21 & 0.12 \\ 0.22 & 0.40 & 0.63 & 0.78 & 0.60 & 0.43 & 0.22 & 0.42 & 0.58 & 0.81 \end{pmatrix}$$

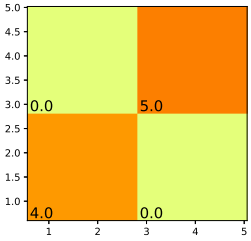
Illustrations des vues

Vous trouverez ici les visuels de toutes les vues pour toutes les séries de l'exemple illustratif du modèle présenté Chapitre 3.

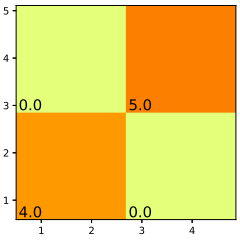
A.0.1 Vue N°1 : Bigramme



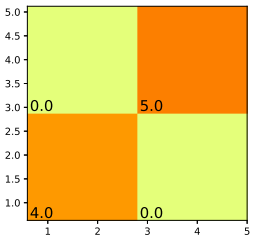
(a) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série N°1



(b) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série N°3



(c) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série N°4



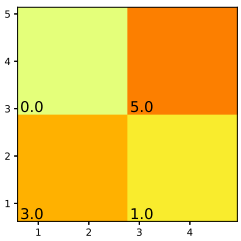
(d) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série Test N°1

FIGURE A.3 – Affichage de la vue N°1 pour chaque STM appartenant à la classe 0

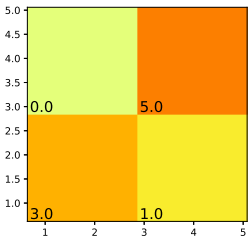
A.0.2 Vue N°2 : Histogrammes

A.0.3 Vue N°3 : Histogramme

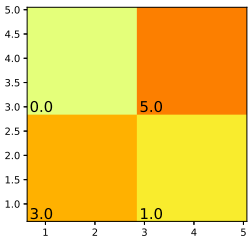
A.0.4 Vue N°4 :Histogrammes



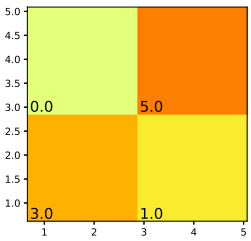
(a) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série N°2



(b) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série N°5

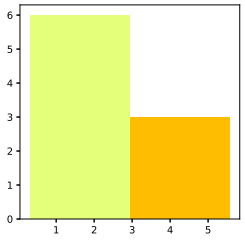


(c) Bigramme des sommes cumulées des dimensions 1 et 2 de la Série N°6

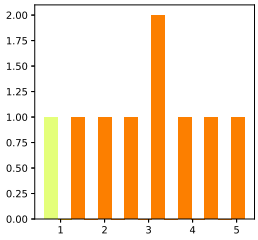


(d) Bigramme de dérivées des dimensions 1 et 2 de la Série Test N°2

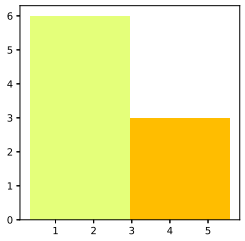
FIGURE A.4 – Affichage de la vue N°1 pour chaque STM appartenant à la classe 1



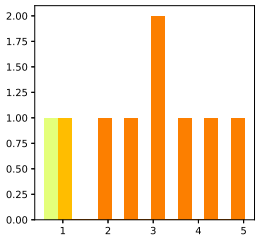
(a) Histogramme des sommes cumulées de la dimension 2 de la Série N°1



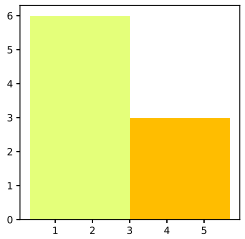
(b) Histogramme des sommes cumulées de la dimension 4 de la Série N°1



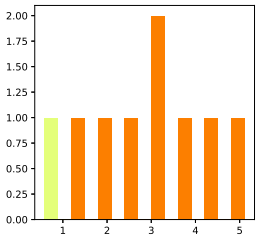
(c) Histogramme des sommes cumulées de la dimension 2 de la Série N°3



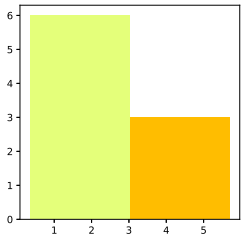
(d) Histogramme des sommes cumulées de la dimension 4 de la Série N°3



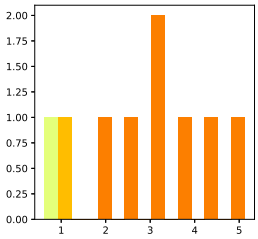
(e) Histogramme des sommes cumulées de la dimension 2 de la Série N°4



(f) Histogramme des sommes cumulées de la dimension 4 de la Série N°4

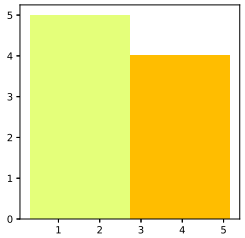


(g) Histogramme des sommes cumulées de la dimension 2 de la Série Test N°1

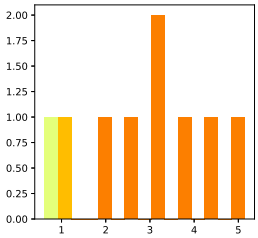


(h) Histogramme des sommes cumulées de la dimension 4 de la Série Test N°1

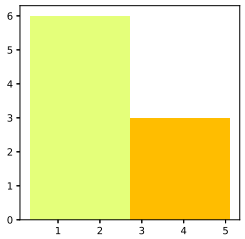
FIGURE A.5 – Affichage de la vue N°2 pour chaque STM appartenant à la classe 0



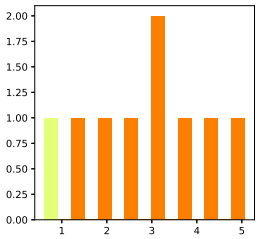
(a) Histogramme des sommes cumulées de la dimension 2 de la Série N°2



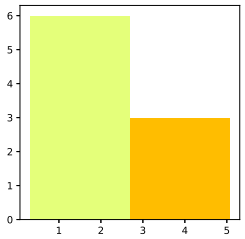
(b) Histogramme des sommes cumulées de la dimension 4 de la Série N°2



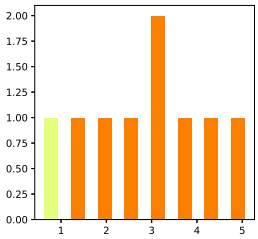
(c) Histogramme des sommes cumulées de la dimension 2 de la Série N°5



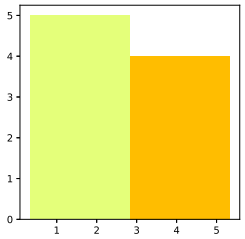
(d) Histogramme des sommes cumulées de la dimension 4 de la Série N°5



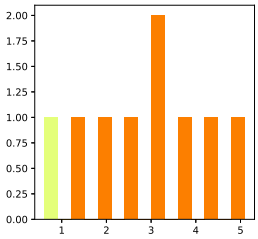
(e) Histogramme des sommes cumulées de la dimension 2 de la Série N°64



(f) Histogramme des sommes cumulées de la dimension 4 de la Série N°6

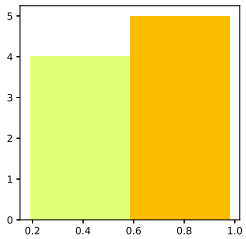


(g) Histogramme des sommes cumulées de la dimension 2 de la Série Test N°2

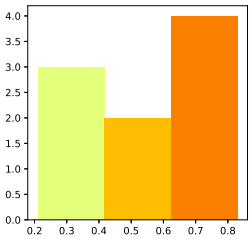


(h) Histogramme des sommes cumulées de la dimension 4 de la Série Test N°2

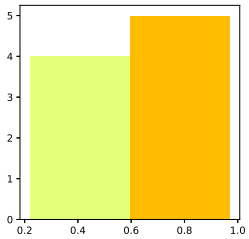
FIGURE A.6 – Affichage de la vue N°2 pour chaque STM appartenant à la classe 1



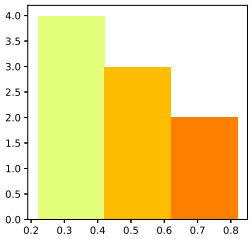
(a) Histogramme de la dimension 2 de la Série N°1



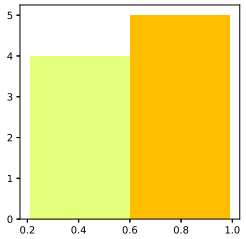
(b) Histogramme de la dimension 4 de la Série N°1



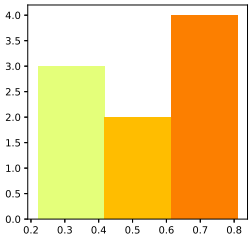
(c) Histogramme de la dimension 2 de la Série N°3



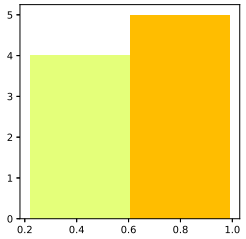
(d) Histogramme de la dimension 4 de la Série N°3



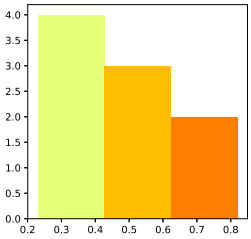
(e) Histogramme de la dimension 2 de la Série N°4



(f) Histogramme de la dimension 4 de la Série N°4

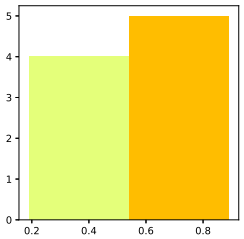


(g) Histogramme de la dimension 2 de la Série Test N°1

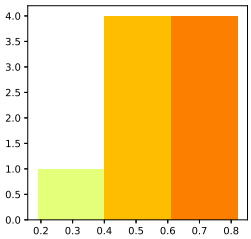


(h) Histogramme de la dimension 4 de la Série Test N°1

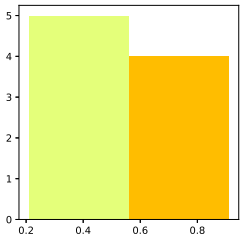
FIGURE A.7 – Affichage de la vue N°3 pour chaque STM appartenant à la classe 0



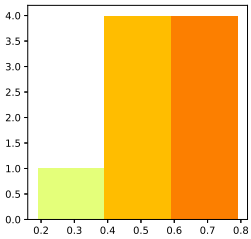
(a) Histogramme de la dimension 2 de la Série N°2



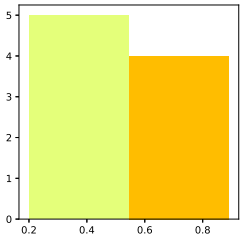
(b) Histogramme de la dimension 4 de la Série N°2



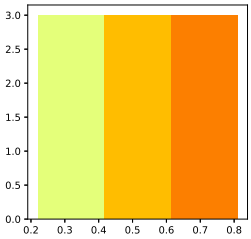
(c) Histogramme de la dimension 2 de la Série N°5



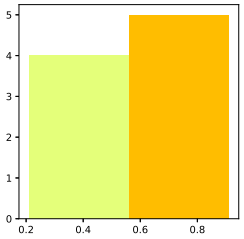
(d) Histogramme de la dimension 4 de la Série N°5



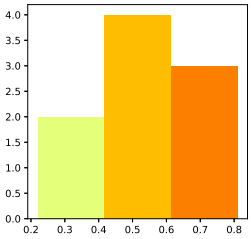
(e) Histogramme de la dimension 2 de la Série N°6



(f) Histogramme de la dimension 4 de la Série N°6

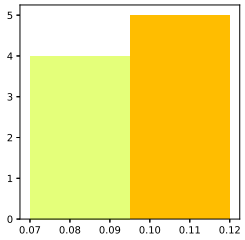


(g) Histogramme de la dimension 2 de la Série Test N°2

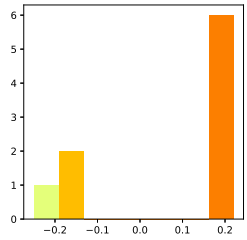


(h) Histogramme de la dimension 4 de la Série Test N°2

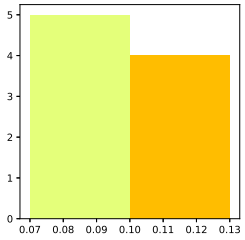
FIGURE A.8 – Affichage de la vue N°3 pour chaque STM appartenant à la classe 1



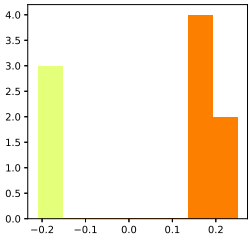
(a) Histogramme des dérivées de la dimension 2 de la Série N°1



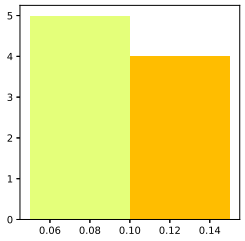
(b) Histogramme des dérivées de la dimension 4 de la Série N°1



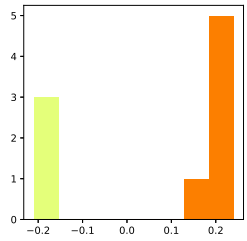
(c) Histogramme des dérivées de la dimension 2 de la Série N°3



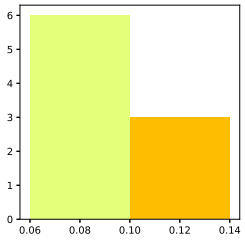
(d) Histogramme des dérivées de la dimension 4 de la Série N°3



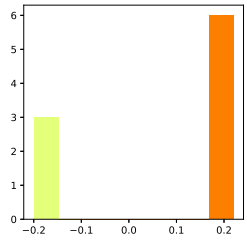
(e) Histogramme des dérivées de la dimension 2 de la Série N°4



(f) Histogramme des dérivées de la dimension 4 de la Série N°4

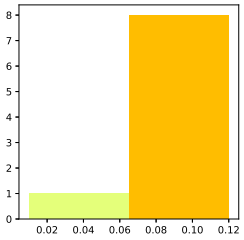


(g) Histogramme des dérivées de la dimension 2 de la Série Test N°1

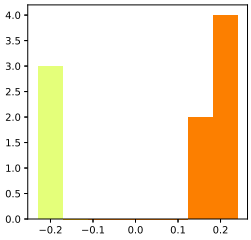


(h) Histogramme des dérivées de la dimension 4 de la Série Test N°1

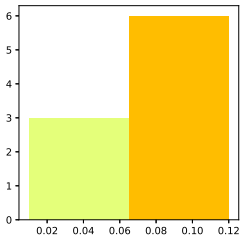
FIGURE A.9 – Affichage de la vue N°4 pour chaque STM appartenant à la classe 0



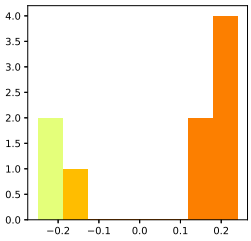
(a) Histogramme des dérivées de la dimension 2 de la Série N°2



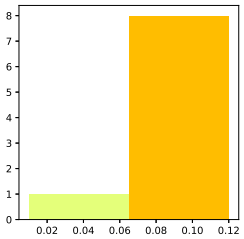
(b) Histogramme des sommes cumulées de la dimension 4 de la Série N°2



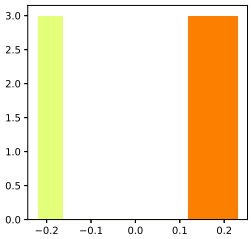
(c) Histogramme des dérivées de la dimension 2 de la Série N°5



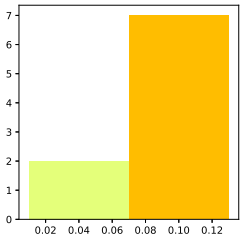
(d) Histogramme des dérivées de la dimension 4 de la Série N°5



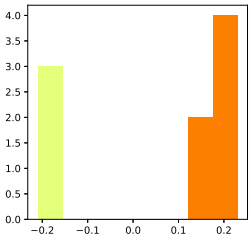
(e) Histogramme des dérivées de la dimension 2 de la Série N°64



(f) Histogramme des sommes cumulées de la dimension 4 de la Série N°6



(g) Histogramme des dérivées de la dimension 2 de la Série Test N°2



(h) Histogramme des dérivées de la dimension 4 de la Série Test N°2

FIGURE A.10 – Affichage de la vue N°4 pour chaque STM appartenant à la classe 1

Temps de calcul

Nous avons mesuré le temps d'exécution de chaque modèle après chargement des données et jusqu'à la prédiction finale. Nous avons donc laissé les paramétrages par défaut. Nous obtenons le tableau des exécutions suivant, Tab A.1 :

	B1gr1NN	SMTS
Wafer	69.82	82.22
ECG	16.05	22.05
CharacterTrajectories	59.51	157.78

TABLEAU A.1 – Temps d'exécution de deux modèles SMTS et de notre modèle en secondes

WEASEL+MUSE ne figure pas dans le tableau car les calculs n'ont jamais abouti à une réponse. C'est-à-dire, que la complexité de la méthode n'a pas permis son exécution dans notre environnement en un temps raisonnable, soit là où notre méthode met quelques secondes, WEASEL+MUSE a tourné plusieurs heures.

Annexe B

Calibration des séries temporelles issues de centrale inertielle

Dans le cadre de la thèse et du travail avec les accélérations, un certain nombre d'étapes de nettoyage ont été nécessaires pour utiliser les données. Ces étapes ont conduit à l'élaboration d'une méthode sur la calibration de séries temporelles issues de centrales inertielles (IMU).

Calibration des séries temporelles multivariées issues d'accéléromètres

Angéline Plaud[1,2], Engelbert Mephu Nguifo[1], Jacques Charreyron[2], and Vincent Dubourg[2]

[1] LIMOS, CNRS, Université Clermont Auvergne, Clermont-Ferrand, France
[2] Michelin, Clermont-Ferrand, France
angeline.plaud@isima.fr

Résumé L'utilisation de capteurs afin de mesurer des phénomènes physiques est aujourd'hui généralisée et massive. Que ce soit via les smartphones ou plus généralement, l'Internet of Things (IoT), les capteurs sont utilisés partout et pour tous domaines d'activités. Le point commun de tous ces capteurs est la nécessité de calibrer la mesure donnée. En effet, toute information captée possède un bruit, qui pollue la valeur que nous cherchons à mesurer et que nous devons quantifier. Lorsque nous nous intéressons à l'accéléromètre, nous nous rendons compte qu'il existe peu de méthodes de calibration. Pourtant, il peut être constaté rapidement que les résultats issus de ce capteur, contiennent beaucoup d'incertitudes, et doivent être traités.

Dans cette étude, nous faisons le tour des méthodes de calibration d'accéléromètres existantes et nous proposons notre propre protocole. Nous comparons ensuite les résultats sur un nouveau jeu de données fourni par Michelin.

Keywords: IMU · accéléromètre · magnétomètre · GPS · calibration · erreur · bruit

1 Introduction

L'utilisation de capteurs afin de mesurer des phénomènes physiques est aujourd'hui généralisée et massive, que ce soit via les smartphones ou plus généralement, l'Internet of Things (IoT), qui rassemble tous les objets connectés du quotidien. Les capteurs sont utilisés partout et pour tous domaines d'activités afin de récupérer de l'information. Le point commun de tous ces capteurs est la nécessité de calibration de la mesure donnée. En effet, toute information captée est sujette à un bruit de mesure, qui pollue la valeur que nous cherchons réellement à mesurer. Nous devons estimer ces incertitudes afin d'approcher la *vraie valeur* du phénomène physique. Lorsque nous nous intéressons à l'accéléromètre, nous nous rendons compte qu'il existe peu de méthodes de calibration capable de corriger les incertitudes très spécifiques de ce capteur. L'accéléromètre permet de mesurer ici l'accélération de véhicule terrestre soit la variation de vitesse. Dans cette étude, nous faisons le tour des méthodes existantes et montrons que

les méthodes proposent une correction très limitée et imparfaite. C'est pourquoi nous proposons notre propre protocole de calibration de l'accéléromètre. La calibration tri-axiale de l'accéléromètre est un problème toujours ouvert qui a ce jour n'a pas reçu de réponse satisfaisante.

Dans la partie 2, nous décrivons l'accéléromètre et son contexte d'utilisation dans notre étude. Dans la partie 3, nous réalisons une veille des méthodes existantes, issus principalement du milieu universitaire, mais aussi du brevet. Dans la partie 4, nous proposons un nouveau protocole de calibration. Dans la partie 5, nous présentons nos données et les résultats des différents algorithmes sur celles-ci. La partie 6 est une conclusion de notre travail.

2 Définitions

Dans cette partie, nous allons décrire le capteur que nous calibrons, c'est-à-dire l'accéléromètre. Nous décrirons son contexte d'utilisation, ses faiblesses que nous voulons corriger et celles que nous sont en dehors de notre travail.

2.1 Le capteur

Le capteur accéléromètre, que nous utilisons, se trouve en réalité au sein d'un ensemble de capteurs, appelé centrale inertielle (IMU). L'accéléromètre permet d'acquérir de la donnée à 22 Hz ou 50 Hz sur tout type de véhicule, voitures, poids lourds, etc. Il est fixé à un endroit définitif dans le véhicule, mais aucune garantie n'est donnée sur sa position exacte de fixation. Ce capteur permet d'obtenir l'accélération du véhicule sur les trois axes du repère spatiale du véhicule en m/s2. Il est fixé à un endroit définitif dans le véhicule, mais aucune garantie n'est donnée sur sa position exacte de fixation. Par ailleurs, l'accéléromètre comme tout capteur n'est pas capable de fournir une mesure exacte d'un phénomène. La mesure renvoyée est la somme de cette valeur avec plusieurs incertitudes.

2.2 Erreurs internes

Le capteur en lui-même est sujet à plusieurs erreurs internes, liées aux composants. La chaleur dégagée par ceux-ci ainsi que les vibrations créent du bruit qui s'ajoutent à la mesure. La valeur mesurée à partir du capteur est donc la somme de la *vraie valeur* et du bruit, soit $M(t) = m(t) + \epsilon(t)$.

Ce bruit ϵ peut être considéré comme négligeable, lissé par des filtres ou approché via des estimateurs statistiques [9]. Ce dernier travail montre que ce bruit est négligeable dans la majorité des travaux de classification et que dans ce cadre, ne pas faire d'opérations est encore la meilleure option. Il n'est donc pas l'objet de notre calibration corrective.

Par ailleurs, le capteur peut parfois émettre une mesure qui dérive à cause des opérations de calcul. Une dérive se produit lorsqu'un calcul doit être effectué de manière interne au capteur et qu'une approximation du résultat est effectuée

car il n'y pas assez de bit pour encoder ce dernier parfois très grand. La série de mesures se met alors *à dériver*, cela veut dire que la moyenne peut varier au cours du temps. Nous abordons donc dans cette étude, ce point de correction qui nous semble important, mais souvent négligé.

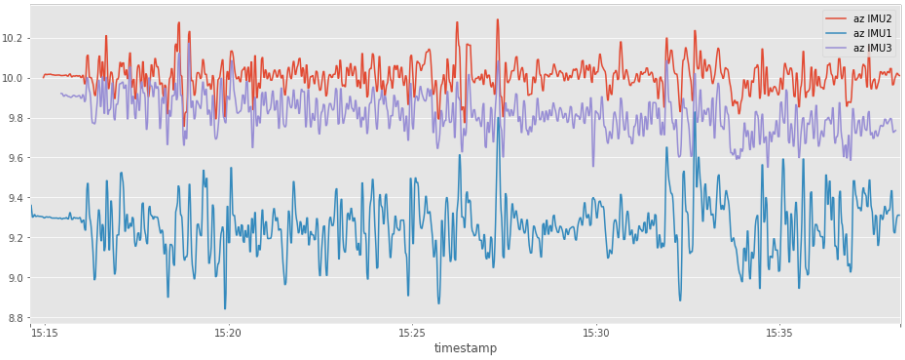


FIGURE 1: Trois axes z de trois boîtiers posés dans la même position, dans un même véhicule, au cours d'un même trajet. Nous pouvons voir des problèmes d'amplitudes et de dérives

Il reste une dernière erreur que nous n'avons pas abordée, qui est l'erreur d'amplitude. Dans certains cas, l'amplitude des signaux transmis par les capteurs diffère 1, c'est-à-dire que l'écart-type peut varier au cours du temps. Cela peut être dû à la somme des erreurs précédentes, comme la chaleur et les erreurs de calcul. Il est primordial de corriger cette erreur, car aucune comparaison n'est possible entre variables de différentes échelles de valeurs.

La dérive et l'erreur d'amplitude sont liées. Nous parlons alors de signal non-stationnaire. Hors, d'un point de vue physique, l'accélération d'un véhicule terrestre est stationnaire.

2.3 Erreurs externes

Le capteur est aussi victime d'erreurs contextuelles. Le capteur peut ainsi être mal installé initialement dans le véhicule. Être mal installé signifie que le repère du capteur n'est pas aligné avec le repère du véhicule (Voir illustration Figure 1.b). On appelle cela la désorientation initiale. Cette désorientation est dite dynamique, si le capteur change d'orientation au cours de l'acquisition de mesures (Illustré par le passage des figures 1.b à 1.c). [3] Nous retrouvons cette configuration dynamique principalement dans le cas d'utilisation de smartphones, qui ne

3. Nous tenons à préciser que nous ne cautionnons pas le comportement induit dans l'illustration 1.b et 1.c.

seraient pas fixés sur un support mais qui se trouveraient sur le siège passager par exemple.

Dans le cas d'acquisition de données par accéléromètre, chez Michelin, le capteur est fixé dans le véhicule. Il peut donc être soumis à une désorientation initiale, mais pas à la désorientation dynamique, hors cas de mauvaises fixations. C'est pourquoi nous ne traiterons que le premier des deux cas.

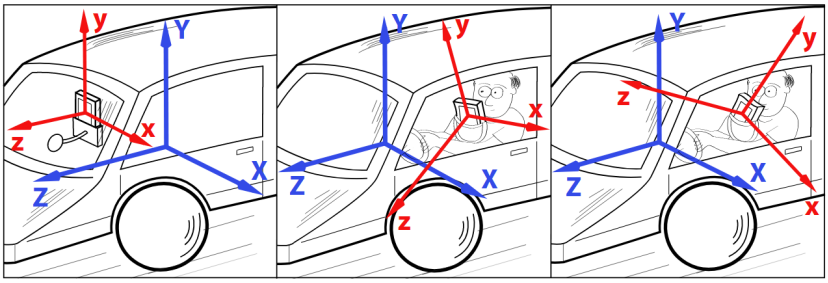


FIGURE 2: L'accéléromètre peut (a) ou ne pas (b) et (c) être dans la bonne position. C'est-à-dire que les référentiels véhicules et capteurs sont identiques. [10]

Finalement, nous nous intéresserons à une IMU fixée dans un véhicule de manière définitive mais non-précise. Nous devons donc corriger la désorientation initiale ainsi que la dérive de calcul et erreur d'amplitudes. Nous ne nous intéresserons pas au bruit capteur, ni à la désorientation dynamique.

3 État de l'art

Nous proposons ici une veille non-exhaustive des méthodes de calibration disponibles. Nous avons analysé aussi bien des publications universitaires, que des brevets et nous proposons ici une description des méthodes présentées comme les plus pertinentes. Par ailleurs, en 2015, [4] ont déjà publiés une veille de la calibration d'IMU. Néanmoins, celle-ci ne s'intéresse qu'aux smartphones et n'a pas exploré les brevets. Par ailleurs, il n'est pas possible de trouver toutes les publications situées de même que nous n'avons pas accès aux données utilisées.

Afin de synthétiser notre travail de veille, un tableau regroupant les principales caractéristiques de chaque méthode est proposé en fin de chapitre.

Méthode de nettoyage des séries temporelles Tout d'abord, les méthodes de nettoyage dites classiques ne sont pas applicables ici [8]. En effet, les méthodes de *smoothing* qui consistent à lisser l'ensemble des données, mais encore les filtres fréquentielles, ne permettent pas de corriger le bruit physique ici. En effet, ce bruit est lié principalement à un mauvais référentiel capteur.

Nericell La méthode de Nericell [11] est basée sur un changement de repère mécanique via l'utilisation des angles d'Euler [5]. Ici, la méthode proposée se base sur une rotation *Z-Y-Z*, aussi appelé *pre-rotation, tilt, post-rotation.* Les deuxièmes premières sont estimées via les équations suivantes :

$$\theta_{tilt} = cos^{-1}(a_z) \tag{1}$$

$$\phi_{pre} = tan^{-1}(a_y/a_z) \tag{2}$$

La fenêtre de temps choisie afin d'appliquer ces équations ne doit pas être prise alors que le véhicule tournerait à haute vitesse. Cette hypothèse est toute fois imprécise et il est libre à l'utilisateur de la définir. La post-rotation, quant à elle, doit être estimée lors des freinages du véhicule. Dans ce cas, nous avons :

$$\psi_{post} = tan^{-1}\frac{-a_x sin(\phi_{pre}) + a_y cos(\phi_{pre})}{(a_x cos(\phi_{pre}) + a_y sin(\phi_{pre}))cos(\theta_{tilt}) - a_z sin(\theta_{tilt})} \tag{3}$$

Nericell++ Cette méthode [10] est basée sur Nericell, seule l'estimation de la post-rotation est revue, d'où notre choix de la nommer *Nericell++*. Ici, le travail nécessite l'extraction d'informations sur la période d'accélération après stationnarité. Dans ce cas, l'estimation de la post-rotation est égale à :

$$\psi_{post} = tan^{-1}\frac{a'_x}{a'_z} \tag{4}$$

où a'_x et $a'z$ sont les accélérations après application des deux premières rotations.

Patent Cette méthode [1] est aussi basée sur un changement de repère via trois rotations. Le deux premières sont estimées alors que le véhicule est à l'arrêt sur un sol plat.

$$\theta = tan^{-1}\frac{a_x}{\sqrt{a_y^2 + a^2 - z}} \tag{5}$$

$$\phi = tan^{-1}(a_y/a_z) \tag{6}$$

La dernière est estimée alors que le véhicule avance en ligne droite et sur une route plate. Dans ce cas, après application des deux autres rotations, il faut réaliser une Analyse des composantes principales (ACP) [7] sur a_x et a_y. L'autre point important de cette méthode est la correction de la dérive du capteur. Ici, la pente du signal est estimée via une régression linéaire et annulée. De même, l'amplitude des signaux est normée en tenant compte de la constante de gravité universelle.

TiltCompensated Cette méthode [12] de même que les précédentes utilise aussi certaines plages de données particulières afin de déterminer les trois rotations nécessaires à un changement de repère mécanique. Seulement, ici, l'application des rotations se fait par X-Y-Z. Il faut d'abord déterminer la rotation permettant de redresser l'axe Z puis les deux autres rotations. Nous avons finalement les angles de rotations suivants :

$$\theta = cos^{-1}(\frac{az}{\sqrt{ax^2 + ay^2 + az^2}}) \tag{7}$$

$$\phi = tan^{-1}(\frac{a'_y}{a'_z}) \tag{8}$$

$$\psi = tan^{-1}(\frac{-a'_x}{\sqrt{a'^2_y + a'^2_z}}) \tag{9}$$

où a'_x, a'_y, a'_z sont les accélérations après application la première rotation.

Le principal point dur des méthodes ci-avant est la nécessité de travailler dans des sous-ensembles de données qui sont loin d'être triviaux à extraire comme les arrêts sur routes plates. Toutes ces méthodes multiplient les hypothèses de départ et donc les sources d'erreurs dans la calibration.

Wolverine Cette méthode [3] de correction de l'accélération utilise le magnétomètre. Le protocole se base sur un double changement de repère mécanique. Tout d'abord, les accélérations sont placées dans un repère dit géométrique grâce à la matrice de rotation formée par le vecteur gravité, le Nord-Sud magnétique et l'Est-Ouest magnétique. Puis il y a un deuxième changement de repère basé sur la matrice de rotation composée par la déclinaison magnétique et le *bearing* qui est l'angle de cap d'un véhicule. La méthode est très peu décrite et semble en réalité basée sur une API fournie sur Android. Aucune hypothèse n'est fournie sur quelles données utilisées. La reproduction de ce travail est donc compliqué et nous avons dû nous-même combler le manque d'explication. Nous sommes par ailleurs circonspects vis-à-vis du magnétomètre puisqu'il est connu que ce capteur n'est pas fiable[6].

Résumé Nous disposons donc de 4 méthodes basées sur un changement de repère. Nous remarquerons que dans tous les cas la notion de translation de repère mécanique est ignorée et que les formules afin de trouver les angles de la matrice de rotations ne sont pas les mêmes. Nous disposons aussi d'une méthode basée sur la fusion avec magnétomètre bien que ce capteur ne soit pas fiable.

4 Protocole de calibration

Nous présentons ici notre protocole de calibration palliant aux déficiences des méthodes proposées ci-avant. C'est à dire, que nous n'imposons aucune condition

	Désorientation initiale	Compensation de la dérive	Hypothèses nécessaires	Autres capteurs
Nericell	Rotations		Freinage	
Nericell++	Rotations		Phase d'accélération	
Patent	Rotations PCA	Régression linéaire	Arrêt Ligne droite	
Tilt Compensated	Rotations		Arrêt	
Wolverine	Rotations			Magnétomètre GPS

TABLE 1: Tableau synthétisant les informations principales sur les différentes méthodes de calibration de l'état de l'art

d'application, de même nous utiliserons un capteur de référence, le GPS, mais celui-ci ne sera en aucun cas fusionné avec les données de l'accéléromètre.

4.1 Correction de la pente

La première étape de notre protocole consiste à corriger la dérive visible sur certains jeux de données comme sur Fig.1 Nous avons choisi ici de faire une simple régression linéaire. Dans le cadre de son application, l'utilisateur est libre de choisir tout autre technique.

Nous avons donc les accélérations brutes définies par :

$$a = \alpha a' + \beta \tag{10}$$

où α et β sont les paramètres à estimer et supprimer. Nous précisions ici que a est une écriture générique qui représente aussi bien l'accélération suivant l'axe x que y et z. De même qu'il est possible que la dérive ne soit pas identique sur chaque axe, nous devons donc estimer trois α et β

4.2 Normalisation

Une fois la dérive des calculs supprimée des données, nous pouvons les normaliser et ainsi corriger les erreurs d'amplitude.

Nous avons donc

$$a''_i = \frac{a'_i}{||a||_2} \tag{11}$$

où i peut être x, y, z.

Nous réalisons ici une simple norme euclidienne, il est libre à l'utilisateur de choisir la norme min-max ou Zéro.

4.3 Calculs des accélérations de références

Pour la recalibration en elle-même de notre protocole, nous aurons besoin d'un capteur de référence permet d'obtenir des accélérations que nous savons

bien orienté, mais qui peuvent être beaucoup plus pauvres en information, *i.e.* fréquence d'acquisition moins élevée - où soumis à trop d'erreurs d'approximations *i.e.* accélérations calculées et non mesurées. Nous avons choisi de nous baser sur le GPS.

Le GPS Le GPS rentre dans les deux critères cités ci-avant. C'est généralement un capteur à la fréquence d'acquisition basse, qui n'est donc pas suffisante si nous voulons détecter des trous ou des bosses sur la route [11] mais qui est toujours disponible. De plus, c'est un capteur dont l'objectif premier n'est pas la captation de l'accélération et donc celle-ci doit être calculée à partir de plusieurs données fournies par le GPS. Cela en fait une référence fiable, mais une mesure pas assez précise pour être utilisée seule. Enfin, l'avantage du GPS est que nous sommes sûrs d'être dans **le référentiel véhicule**.

Formulation des accélérations Nous calculons les accélérations à partir du GPS via les formules physiques de base suivantes :

$$A_x = \frac{\delta T}{\delta T} \tag{12}$$

$$A_y = V \times \frac{\delta B}{\delta T} \tag{13}$$

$$A_z = \frac{\delta Al}{(\delta T)^2} \tag{14}$$

Les erreurs qui pourraient s'accumuler dans le calcul ne sont pas un obstacle pour les prochaines étapes. En effet, le but n'est pas la précision, mais les tendances d'évolutions d'accélérations du véhicule.

4.4 Rotation et translation

Dans notre protocole, en se basant sur la mécanique du solide, pour changer de repère, il faut réaliser une rotation et une translation [2].

La méthode présentée dans [13] permet d'estimer la matrice de rotation et le vecteur de translation, permettant le passage d'une matrice vers une autre, au moyen d'une réduction de dimension par une technique de décomposition de type *Singular value decomposition* (SVD). La SVD est un cas particulier de ACP déjà abordé ci-avant. C'est un concept de factorisation matricielle qui permet de décomposer des matrices réelles. Nous avons décidons de l'appliquer ici sur nos séries afin de trouver la matrice de rotation et le vecteur de translation permettant à l'accéléromètre d'être dans le même référentiel (véhicule) que le GPS.

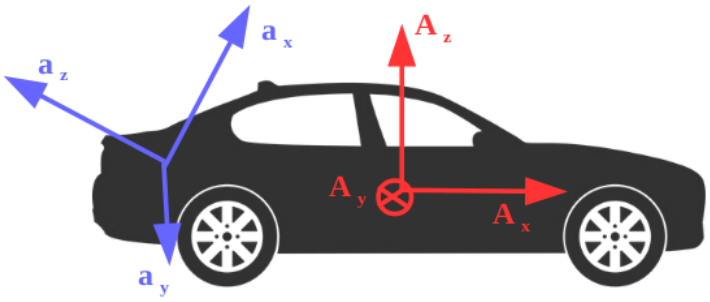


FIGURE 3: Changement repère

4.5 Protocole final

Notre protocole à la différence des autres ne fait donc pas d'hypothèse sur les données à utiliser pour déterminer le changement de repère. De même, les angles de rotations sont déterminés via une procédure mathématique et non sur des formules physiques. Par ailleurs, nous ne voyons pas de justification au fait que le changement de référentiel doit se limiter à la matrice de rotation et donc nous cherchons le vecteur de translation.

De plus, nous jugeons obligatoires de corriger les erreurs de dérives et d'amplitudes et donc les avons intégrer à la méthode. Enfin, nous utilisons un capteur de référence, le GPS, mais ne réalisons en aucun cas de fusion d'informations.

L'ensemble du protocole est résumé dans le diagramme algorithmique ci-après Fig. 4. Les calculs déterminants du GPS ainsi que les premières étapes du protocole peuvent être réalisées en parallèle.

Les opérations opérées sur le GPS sont de simples opérations arithmétiques réalisées sur l'ensemble des valeurs acquises soit $O(3N)$ une N est le nombre de points d'accélérations captées, 3 pour chaque axe. Pour l'étapes de normalisation, la même complexité est atteinte soit $O(3N)$ pour l'imu et pour le gps. Nous réalisons aussi une régression linéaire $O(N^2)$ et une SVD $O(3N^2)$.

5 Évaluations

Dans cette partie, nous détaillons l'évaluation des protocoles présentés ci-avant. Les données utilisées sont la propriété de Michelin. Nous détaillons aussi les métriques employées pour évaluer la validité de la recalibration. Enfin, nous présentons les résultats.

5.1 Les données

Nous avons installé trois boîtiers télématiques contenant un GPS et une centrale inertielle (IMU) sur un même véhicule. Les boîtiers étaient disposés

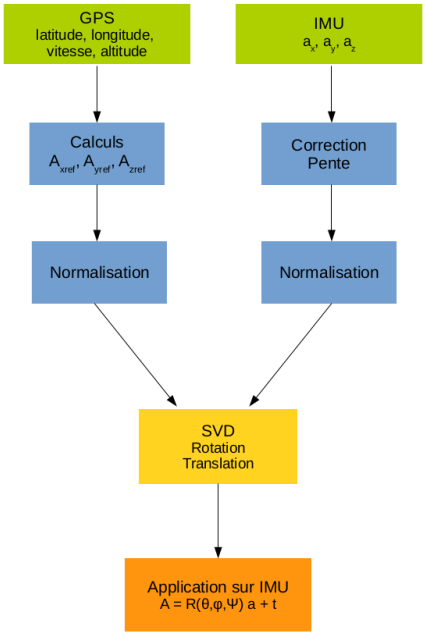


FIGURE 4: Diagramme algorithmique du protocole de calibration mis en place

à chaque trajet dans différentes positions et les trois étaient déclenchés simultanément.

Nous avons réalisé un découpage des données captées qui nous donne cinq voyages. Il est possible de découper plus finement les données en faisant évoluer la définition d'un trajet. Un trajet est ici défini comme nouveau lorsqu'aucun mouvement du véhicule n'est enregistré **au cours d'une minute (vitesse = 0)**. Par ailleurs, un trajet est aussi nouveau à chaque redémarrage de boîtiers. Le GPS a une fréquence d'acquisition de 1Hz, quant à l'IMU, la fréquence d'acquisition est de 50Hz lors de l'expérimentation.

Avant d'appliquer les protocoles, nous avons ré-échantillonné les données par seconde pour le GPS afin de combler les sauts et donc interpolé les données si nécessaire. De même, l'IMU est passée par un filtre passe-bas afin de supprimer une partie du bruit parasite.

5.2 Paramètre d'évaluation

Pour évaluer le résultat de la calibration nous utilisons deux métriques. La moyenne statistique et la corrélation de Pearson.

La moyenne Le premier métrique employé la moyenne des trois accélérations pour chaque axe. Du point de vue statistique, les données accélérométriques

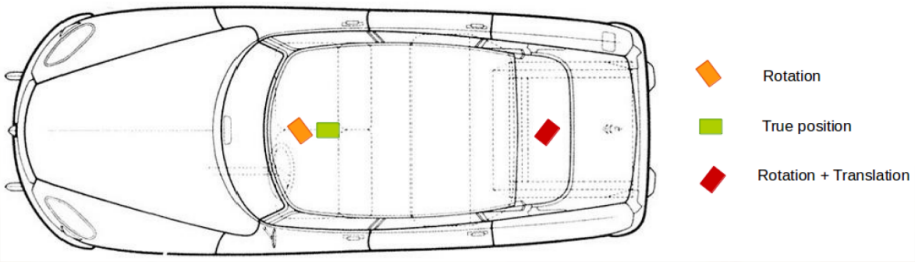


FIGURE 5: Exemple de positionnements des boîtiers dans le véhicule.

d'un véhicule terrestre suivent une loi Normale centrée sur 0 pour l'accélération longitudinale (dans le sens de marche du véhicule) et latérale (sur les côtés du véhicule). Pour l'axe Z, la moyenne est centrée sur 9.81 qui est la constante de gravité terrestre.

La valeur 0 s'explique par la physique, car l'état principal d'un véhicule est la stationnarité et la vitesse constante où l'accélération est donc nulle.

Donc, dans le cas où le boîtier n'est pas recalibré, la constante de gravité peut influencer plusieurs axes de même que les erreurs de calcul. Les moyennes ne sont alors plus centrées sur zéro ou 9.81.

La corrélation La moyenne seule n'est pas suffisante pour définir la qualité de la procédure. La moyenne peut être centrée sur zéro avec des événements déformés ne correspondant pas à la réalité. Nous avons donc décidé d'ajouter la mesure de corrélation. Cette mesure nous assure que les tendances contenues au sein de chaque série sont similaires. La corrélation c est égale à :

— 1 : si les deux séries évoluent de manière identique.
— 0 : si les deux séries n'ont aucune tendance communes et sont donc complètement différentes.
— -1 : si les deux séries évoluent de manière opposée.

Le cas -1 peut représenter les cas où les axes du référentiel boîtiers seraient dans des sens opposés à ceux du référentiel véhicule.

Nous calculons la corrélation entre boîtiers recalibrés, mais aussi avec le GPS.

Si la correction a fonctionné alors nous devrions avoir une moyenne proche de 0 pour les accélérations X et Y, et proche de 9.81 pour Z. De même les corrélations devrait toutes être proches de 1.

Le choix de ces métriques est limité, mais suffit à évaluer les recalibrations. Par ailleurs, ce sont des métriques rapides à calculer.

5.3 Résultats

Moyennes Nous pouvons voir ici que la méthode issue du brevet donne les meilleurs résultats. Néanmoins, notre méthode le talonne de peu et est dans le

MOYENNE									
	Ax A	Ax B	Ax C	Ay A	Ay B	Ay C	Az A	Az B	Az C
Tilt compensated	-0,979	0,872	-0,461	0,393	-1,159	-0,820	8,301	9,219	9,459
Nericell	3,813	0,583	0,455	1,211	0,722	1,076	6,849	9,558	9,216
Nericell++	3,627	0,639	0,717	1,059	-0,337	-1,551	6,893	9,713	8,155
Wolverine	-0,558	**0,000**	-0,093	0,402	-0,412	-0,345	6,938	8,482	8,685
Patent	-0,049	-0,132	**0,000**	**-0,016**	**-0,053**	**0,063**	**9,810**	**9,810**	**9,810**
SVD	**0,033**	-0,002	0,021	0,056	0,072	0,069	9,817	9,807	9,808
GPS Référence	0,007			0,074			9,808		

Moyenne par axes et par boîtiers des accélérations recalibrées.

même ordre de grandeur que ses moyennes calculées sur les données GPS. Nous pouvons conclure que ces méthodes sont ex-aequo du point de vue de la moyenne. Nous pouvons aussi voir que toutes les méthodes ne réalisant pas de corrections de pentes ni d'amplitude donnent de mauvais résultats. Cela prouve la nécessité de réaliser ces deux corrections dans tout protocole de recalibration.

Corrélations Les corrélations vertes et rouges foncées sur la fig.6 signifient des corrélations fortes positives et négatives respectivement. Tandis que le jaune pale indique des corrélations faibles. Les tableaux les plus à gauches représentent les axes Z, au milieu Y et à droite X. De même les lignes et colonnes de chaque tableau sont organisées comme suit : GPS, boîtier A, boîtier B et puis C.

Nous pouvons conclure plusieurs choses. La première est que dans tous les cas les corrélations sur l'axe Z sont les plus faibles. Cela s'explique, car c'est un axe très bruité. C'est en effet sur cet axe que nous captons les effets de routes.

La deuxième chose est que contrairement à la moyenne, le brevet ne donne pas de bonnes corrélations. La moins bonne méthode est Wolverine. Cela s'explique par le magnétomètre qui n'est pas un capteur fiable. Finalement, nous pouvons voir que c'est la correction par notre méthode qui donne les corrélations les plus élevées.

Par ailleurs, les corrélations des autres méthodes donnent des corrélations non-négligeables. Nous allons nous intéresser à ces dernières méthodes pour voir si intégrer à notre protocole de correction, nous pouvons améliorer leurs résultats, c'est-à-dire si nous réalisons les étapes de corrections de la dérive et de l'amplitude.

Corrections de la dérive et de l'amplitude Nous appliquons les premières étapes du protocole, en bleu dans le diagramme 4, puis réalisons les différentes méthodes, Nericell, Nericell++ et Tilt compensated, afin de voir si nous pouvons améliorer leurs résultats.

Nous voyons que les moyennes ne sont pas améliorées. Nous supposons donc que les erreurs de dérivées et d'amplitudes n'expliquent pas à elle seules les erreurs de moyennes. Et donc ces méthodes ne permettent pas de déterminer

avec exactement les angles de rotations puisque la gravité continue d'influencer tous les axes des boîters.

Nous pouvons conclure que seule notre méthode permet une recalibration de l'accéléromètre.

6 Conclusion

Nous avons analysé l'accéléromètre expliquant les différents paramètres pouvant provoquer des erreurs de captations. Nous avons réalisé un état de l'art et montré qu'aucune de ces méthodes ne permet objectivement de réaliser une recalibration tri-axiales. Nous avons montré que notre méthode peut le faire, néanmoins l'axe Z contient toujours beaucoup de bruits.

Le problème de la recalibration est un problème ouvert. Comme étape future de notre travail, nous devons incorporer le bruit de route et espérer approcher une meilleure détermination de la recalibration.

Références

1. Basir, O.A., Jamali, S.H., Miners, W.B., Toonstra, J. : Method of correcting the orientation of a freely installed accelerometer in a vehicle (04 2013)
2. Berthelot, J.M. : Mécanique des solides rigides. Technique et documentation (1999)
3. Bhoraskar, R., Vankadhara, N., Raman, B., Kulkarni, P. : Wolverine : Traffic and road condition estimation using smartphone sensors. In : 2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012). pp. 1–6 (2012)
4. Carlos, M.R., González, L.C., Martínez, F., Cornejo, R. : Evaluating reorientation strategies for accelerometer data from smartphones for its applications. In : García, C.R., Caballero-Gil, P., Burmester, M., Quesada-Arencibia, A. (eds.) Ubiquitous Computing and Ambient Intelligence. pp. 407–418. Springer International Publishing (2016)
5. Euler, L. : Du mouvement de rotation des corps solides autour d'un axe variable. Mém. Berlin (1765)
6. Hemanth, K., Talasila, V., Rao, S. : Calibration of 3-axis magnetometers. IFAC Proceedings Volumes **45**(1), 175 – 178 (2012)
7. Jolliffe, I., Cadima, J. : Principal component analysis : a review and recent developments. Philosophical Transactions of the Royal Society of London Series A **374**, 20150202 (2016)
8. Kostelich, E., Schreiber, T. : Noise reduction in chaotic time-series data : A survey of common methods. Physical Review E - Statistical, Nonlinear, and Soft Matter Physics **48**(3), 1752–1763 (1993). https://doi.org/10.1103/PhysRevE.48.1752
9. Large, J., Southam, P., Bagnall, A.J. : Can automated smoothing significantly improve benchmark time series classification algorithms ? ECML 2018
10. Li, K., Lu, M., Lu, F., Lv, Q., Shang, L., Maksimovic, D. : Personalized driving behavior monitoring and analysis for emerging hybrid vehicles. In : Kay, J., Lukowicz, P., Tokuda, H., Olivier, P., Krüger, A. (eds.) Pervasive Computing. pp. 1–19. Springer Berlin Heidelberg (2012)

11. Mohan, P., Padmanabhan, V., Ramjee, R. : Nericell : Rich monitoring of road and traffic conditions using mobile smartphones. In : ACM Sensys. Association for Computing Machinery, Inc. (November 2008)
12. Pedley, M. : Tilt sensing using a three-axis accelerometer. Freescale semiconductor application note **1**, 2012–2013 (01 2013)
13. Sorkine-Hornung, O., Rabinovich, M. : Least-squares rigid motionusing svd. Tech. rep. (2016)

FIGURE 6: Corrélations entre les trois boîtiers et le GPS pour toutes les méthodes

MOYENNE									
	Ax A	Ax B	Ax C	Ay A	Ay B	Ay C	Az A	Az B	Az C
Tilt compensated	-0,892	0,354	-0,438	-1,032	-1,051	-0,772	7,031	9,112	9,236
Nericell	2,726	0,999	0,880	-1,995	0,475	1,166	8,769	9,386	9,050
Nericell++	2,366	0,976	0,984	-1,854	-0,794	-1,376	7,844	9,553	8,003
GPS Référence	0,007			0,074			9,808		

Moyenne par axes et par boitiers des accélérations recalibrés avec notre protocole

Annexe C

Liste des acronymes

1NN Plus proche voisin. 43, 56, 57, 63, 70, 76, 79, 80, 86, 88, 90, 92, 93, 95, 96, 109, 121

ARIMA Auto Regressive Moving Average. 25

ARMA Auto Regressive Moving Average. 23, 25

BOSS Bag of SFA Symbol. 22

CAN Controller Area Network. 101, 102

CAp Conférence de l'Apprentissage Automatique. 119

COTE Collective of Transformation-Based Ensembles. 31, 32, 35, 55, 57

DDTW Derivative Dynamic Time Warping. 21

DTW Dynamic Time Warping. 20, 21, 25, 26, 30, 32, 35, 57, 92, 93, 95

ECML European Conference of Machine Learning. 119

EMMV Ensemble de Mgramme Multi-Vues. 44, 45, 63, 74, 77, 93, 96, 100, 109, 110, 118

FDST Flux de Données et Séries Temporelles. 119

GPS Global Positioning System. 101, 102, 105, 106, 110

IMU Inertial Measurement Unit. 102, XIII

JMLC Journal of Machine Learning and Cybernetics. 119

KNN K plus proches voisins. 57, 86

MSM Move-Split-Merge. 21

PAA Piecewise Aggregation Approximation. 22, 30

PCA Principle Component Analysis. 29, 30

SAX Symbolic Aggregate approXimation. 22, 23, 26, 57, 74, 91, 92, 93

SAX-VSM SAX and Vector Space Model. 22

SFA Symbolic Fourrier Approximation. 23

SIA Société d'Ingénieurie Automobile française. 119

SMTS Symbolic representation for MTS. 19, 26, 27, 30, 35, 47, 48, 69, 70, 74, 92, 93, 95, 110, XII

STM Séries Temporelles Multivariées. 3, 11, 12, 13, 14, 15, 16, 17, 20, 25, 26, 27, 28, 30, 33, 34, 35, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 57, 58, 59, 61, 62, 63, 64, 66, 68, 69, 70, 74, 81, 83, 90, 91, 92, 93, 107, 110, 118, 119, 120, IV

STU Séries Temporelles Univariées. 15, 20, 21, 22, 23, 25, 28, 31, 32, 34, 35, 48, 51, 57

TWED Time Warp Edit Distance. 21

WDTW Weighted Dynamic Time Warping. 21

WEASEL Word ExtrAction for time SEries cLassification. 23, 28

WEASEL+MUSE WEASEL plus Multivariate Unsupervised Symbols and dErivatives. 19, 27, 28, 30, 35, 48, 69, 70, 74, 92, 93, 95, 110, XII