Résumé

La préservation et/ou la restauration du bon état des cours d'eau exigée par la Directive Cadre Européenne sur l'eau met en exergue la nécessité de disposer d'outils opérationnels pour aider à l'interprétation des informations complexes concernant les cours d'eau et leur fonctionnement. En effet, ces outils permettront l'évaluation de l'efficacité des programmes d'actions engagés, ce qui a conduit à la naissance du projet ANR Fresqueau. Le projet Fresqueau a conduit à la création d'une base de données à partir de diverses sources pourtant sur la qualité physico-chimique et biologique des cours d'eau.

Le projet ADQUEAU est un prolongement du projet ANR Fresqueau avec pour double objectifs s'étalant sur deux années académiques. Le premier objectif sur lequel nous avons travaillé consistait à implémenter des modèles d'apprentissage pour la construction de clusters pour une analyse rétrospective sur les données recueillies. Le deuxième objectif sert, lors de la deuxième année de mettre en oeuvre l'approche de clustering sous contrainte proposée par l'équipe SDC. Cette analyse étant sur des séries temporelles, il a été décidé de faire une analyse chronologique et non chronométrique pour l'évaluation de la qualité physico-chimique et biologique des cours d'eau.

Pour y parvenir lors de mon stage, nous avons procédé à une étude des travaux connexes sur l'apprentissage non supervisé des séries temporelles. Une première approche a été l'étude des algorithmes tels que DBSCAN, TDBSCAN, CHA, SWAP, Kmeans. Une seconde approche étudiée a été celle basée sur l'apprentissage non supervisé avec les réseaux de neurones tels que Deep Belief Network (DBN), les réseaux de neurones convolutionnels (RNC) et Deep Temporal Clustering (DTC).

A l'issue de l'étude réalisée sur les différentes approches relatives au sujet, notre choix est porté sur Kmeans pour sa rapidité et sa robustesse. Comme de nombreux algorithmes d'apprentissage, une distance de similarité est requise. Ainsi la distance DTW (Dynamic Time Warping) a été approuvée comme étant la mesure de similarité la mieux adaptée. Mais bien avant l'application de Kmeans, plusieurs méthodes d'imputation des valeurs manquantes, de normalisation et de réduction de dimension des données ont été implémentées pour avoir un jeu de données cohérent avec les objectifs du projet.

Nos travaux serviront de base à la seconde phase du projet pour la prochaine année académique.

Abstract

The preservation and/or restoration of the good condition of watercourses required by the European Water Framework Directive highlights the need for operational tools to help interpret complex information about watercourses and their functioning. These tools will allow the evaluation of the effectiveness of the action programs undertaken, which led to the birth of the ANRFresqueau project. The Fresqueau project led to the creation of a database from various sources on the physico-chemical and biological quality of watercourses.

The ADQUEAU project is an extension of the ANR Fresqueau project with two objectives spread over two academic years. The first objective we worked on was to implement learning models for the construction of clusters for retrospective analysis of the data collected. The second objective is to implement the constrained clustering approach proposed by the SDC team in the second year. This analysis being on time series, he has was decided to do a chronological analysis for the evaluation of the physico-chemical and biological quality of watercourses.

To achieve this during my internship, we conducted a study of related work on unsupervised time series learning. A first approach was to study algorithms such as DBSCAN, TDBSCAN, CHA, SWAP, Kmeans. A second approach studied was the one based on unsupervised learning with neural networks such as Deep Belief Network (DBN), convolutional neural networks (RNC) and Deep Temporary Clustering (DTC).

At the end of the study carried out on the different approaches relating to the subject, we chose Kmeans for its speed and robustness. Like many learning algorithms, a similar distance is required. Thus the DTW (DynamicTime Warping) distance was approved as the most appropriate similarity measure. But before the application of Kmeans, several methods of missing values imputation, normalization and data reduction were implemented to have a data set consistent with the project objectives.

Our work will serve as the basis for the second phase of the project for the next academic year.

Remerciements

La réalisation de ce mémoire a été possible grâce au concours de plusieurs personnes à qui je voudrais adresser ma profonde gratitude.

Je voudrais par ces mots, remercier tout d'abord mes encadrants ***Pierre Gançarski*** *le Directeur Adjoint d'ICube, et* ***Agnès Braud*** *de l'équipe* ***SDC*** *(Science des Données et Connaissances), pour leurs disponibilités, leurs conseils judicieux qui m'ont été très bénéfiques dans la réalisation de ce projet.*

J'adresse également ma reconnaissance à tous les membres des deux laboratoires : laboratoire ICube et laboratoire LIVE, qui lors de nos rencontres ont pu soulever des problèmes qui m'ont permis de mieux comprendre et d'appronfondir mes connaissances sur le projet.

Je saisis cette occasion également pour remercier tout le corps enseignant de l'IFI (Institut Francophone International) pour la formation que nous avons reçue durant ces deux années académiques.

Je ne saurais terminer sans témoigner ma reconnaissance à tous ceux ou celles qui de près ou de loin ont apporté leurs soutiens de différentes natures pour la réussite de ce mémoire. Je vais ici remercier toute la famille ZONGO, en particulier mes parents et la famille BOMBIRI.

Ce mémoire doit beaucoup son succès aux différentes rencontres que nous avons tenues pour la clarification de chaque point du sujet, aux critiques mais surtout aux différentes suggestions. De toutes ces expériences j'ai pu tirer le plus grand profit des connaissances qui me permettrons d'entrer dans ma vie professionnelle avec sérénité.

Dédicaces

A mes grands-parents Vourma, Yamba, Kolibié et Noaga pour votre amour inexprimable et toutes vos bénédictions qui continuent à me fortifier et à m'animer de force.

A mes parents Noaga et N'Gané pour vos encouragements et vos soutiens qui sont toujours une bouffée d'oxygène qui me ressource dans les moments pénibles, de solitude et de souffrance. Merci d'être toujours à mes côtés, par votre présence, vous qui n'avez jamais cessé de me soutenir tout au long de mes études, je ne saurai vous exprimer ma profonde gratitude et ma reconnaissance.

Table des matières

Table des figures

Introduction générale

Ces dernières décennies sont marquées par de nombreuses études centrées autour des données. Cette accélération des recherches autour des données dans ces dernières décennies est due à l'explosion des données à l'échelle mondiale.

L'exploration de ces données est rendue possible grâce à l'avancée de la technologie qui a conduit à l'apparition des serveurs, des micro-serveurs, des conteneurs pouvant stocker un volume encore jamais vu de données, ainsi de nombreuses techniques et approches ont été développées pour l'extraction de connaissances à partir de ces données. L'apprentissage supervisé connaît aujourd'hui un grand succès avec l'avènement de Deep learning. Cependant l'apprentissage non supervisé en particulier celui des données temporelles demeure un challenge en Machine learning. Ce challenge est dû à la complexité de la structure de certaines séries temporelles, mais aussi à la grande dimensionnalité de ces données temporelles qui nécessitent un filtrage de très bon qualité des attributs perninents. Ainsi cet apprentissage non supervisé s'effectue avec des données non étiquetées.

De telles données temporelles peuvent provenir de diverses sources telles que les objets connectés, les informations de santé, les données de réseaux sociaux, les transactions sur un marché, les données sur la météorologie, les données sur la production agricole, les données sur l'évolution de la population d'une région, les données sur la pollution de l'air, la pollution des sources d'eau.

Le projet sur lequel nous avons travaillé est relatif aux données collectées sur des cours d'eaux afin d'étudier les caractéristiques physico-chimique et biologique de ceux-ci.

Pour le déroulement de nos travaux, nous présentons tout d'abord la structure d'accueil en chapitre 1, l'analyse du sujet en chapitre 2, l'étude des travaux connexes et de l'outil d'analyse de données FoDoMuST en chapitre 3, les techniques et méthodes utilisées en chapitre 4, en chapitre 5 nous présentons la partie implémentation et expérimentation et en chapitre 6 Intégration à FoDoMuST de fonctions externes pour l'analyse de séries et enfin nous terminons par la conclusion et les perspectives.

Chapitre 1

Structure d'accueil

Dans ce chapitre nous présentons la structure d'accueil en particulier sa hiérarchie, l'équipe dans laquelle j'ai intégré et surtout les thèmes sur lesquels son activité de recherche s'articule.

1.1 Présentation générale du Laboratoire ICube

Créé en 2013, le laboratoire regroupe les forces de recherche du site universitaire de Strasbourg, dans le domaine des sciences de l'ingénieur et de l'informatique, avec l'imagerie comme thème fédérateur.

ICube a vocation à être un acteur majeur en ingénierie biomédicale, en étant implanté sur les sites des hôpitaux universitaires de Strasbourg, au sein de l'Institut Hospitalo-Universitaire (IHU), de l'Institut de Recherche contre les Cancers de l'Appareil Digestif (IRCAD), et de l'Institut de Physique Biologique(IPB). Les scientifiques du laboratoire développent également des recherches originales pour l'environnement et le développement durable, en œuvrant notamment dans la géothermie et le photovoltaïque.

Les principaux champs d'expertise couvrent la physique, la microélectronique et les nanosciences, l'automatique et la robotique, l'informatique et les réseaux (infrastructures numériques et objets communicants), le traitement des données ("data science / Big Data" qui comprend le traitement du signal et des images), l'optique et le laser ainsi que l'ingénierie pour la santé.

FIGURE 1.1 – Structure d'accueil

1.2 Organigramme du laboratoire

Les activités de recherche du laboratoire sont regroupées en 4 départements et 16 équipes de recherche. Le schéma ci-dessous est une représentation de l'organisation du laboratoire.

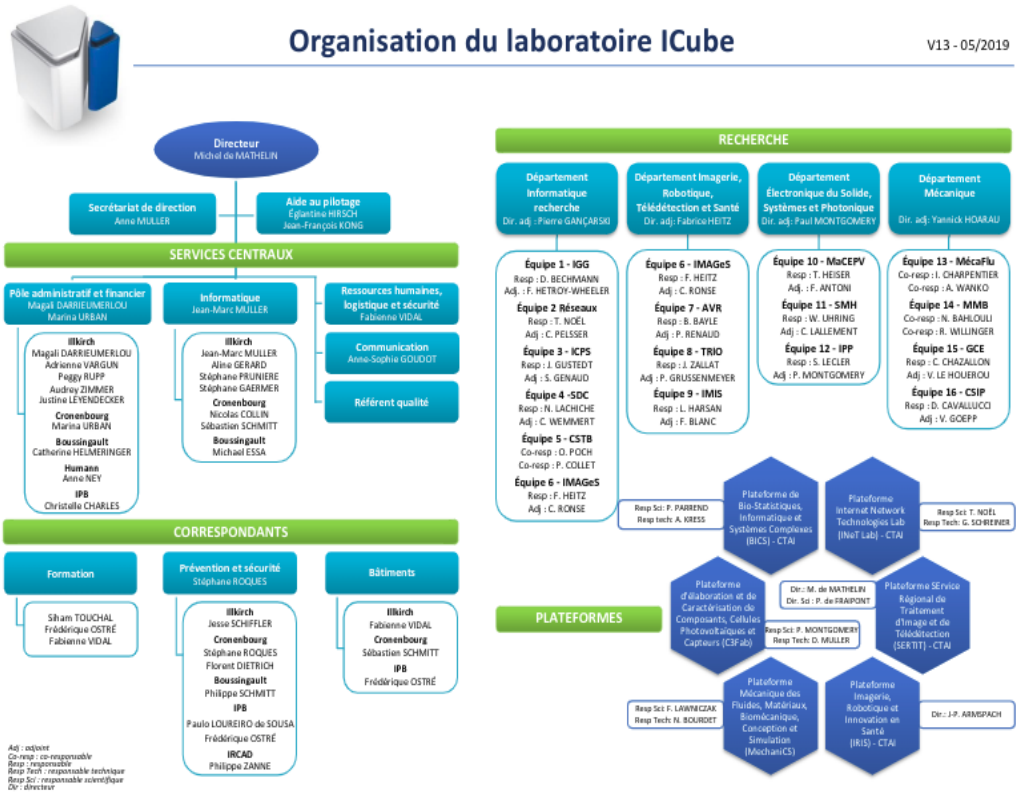


FIGURE 1.2 – Organigramme

1.3 Présentation de l'équipe SDC

L'équipe que j'ai intégrée est l'équipe SDC : **Science des Données et Connaissances** qui couvre un large spectre de recherches en informatique, plus précisément en fouille de données et en intelligence artificielle. C'est en début 2016 qu'est née l'équipe SDC, quand l'ancienne équipe BFO est scindée en deux pour devenir d'un côté l'équipe Systèmes Complexes, Bioinformatique Translationnelle (CSTB) et de l'autre l'équipe SDC. Cette dernière a pour thématique la Fouille de Données et l'Ingénierie des Connaissances. Son activité de recherche s'articule autour de deux thèmes de recherche théoriques et de quelques domaines d'application privilégiés à savoir :

— La science des données.
— Les connaissances et technologies sémantiques.
— avec des applications concrète telles que :
    •Modélisation des connaissances du domaine au moyen d'ontologies (web sémantique).
    •Raisonnement qualitatif spatial et temporel dans les ontologies.
    •Modélisation des connaissances imprécises au moyen de la logique floue.
    •Télédétection et analyse d'images.
    •Analyse de données temporelles massives

Les membres de cette équipe travaillent le plus souvent sur des projets de recherche en collaboration avec des laboratoires ou des entreprises.

1.4 Ressources Humaines et budgets

Le laboratoire compte 650 membres, dont 274 Permanents (professeurs, maîtres de conférences, techniciens, personnels administratifs, chercheurs, directeurs de recherche, ingénieurs et techniciens), 180 non permanents (doctorants, post doctorants, et personnels sous CDD et chargés de recherche). Le laboratoire ICube gère un budget de 28 Millions d'Euros de fonctionnement annuel. Enfin, le laboratoire est multisite sur six emplacements, Télécom Physique Strasbourg, IRCAD à l'hôpital civil, Institut de Physique Biologie, Campus de Chronenbourg, site de la rue Boussingault et l'INSA de Strasbourg.

Après une brève présentation de la structure d'accueil, nous faisons dans le chapitre suivant l'analyse du sujet qui a fait l'objet de nos travaux.

Chapitre 2

Analyse du sujet

L'objectif de préserver ou restaurer le bon état des masses d'eau, imposé par la Directive Cadre Européenne sur l'eau, met en exergue la nécessité de disposer d'outils opérationnels pour aider à l'interprétation des informations complexes concernant les cours d'eau et leur fonctionnement, ainsi que pour évaluer l'efficacité des programmes d'actions engagés ce qui conduit à la naissance du projet ANR Fresqueau. Mon stage s'inscrit dans le cadre du nouveau projet ADQUEAU qui fait suite à Freasqueau, projet que nous présentons dans les sections suivantes.

2.1 Contexte du projet ADQUEAU

Le projet ANR Fresqueau s'est déroulé de 2011 à 2015 et a réuni un consortium de quatre laboratoires de recherche et deux bureaux d'études, avec des équipes d'informaticiens spécialisés en structuration et extraction de connaissances à partir de données et des équipes d'hydrologues et d'écologues spécialistes de l'évaluation des écosystèmes aquatiques.

Cette collaboration a débouché entre autres sur la construction d'une base de données importante à partir de sources diverses telles que des agences de l'eau et l'ONEMA (Office National de l'Eau et des Milieux Aquatiques), mais également l'IGN (Institut Géographique et Forestier National) pour l'information géographique et différents services de l'État. Les données collectées portent sur deux grands bassins hydrographiques, correspondant aux districts Rhin-Meuse (33.000 km2) et Rhône-Méditerranée et Corse (130.000km2), pour une période de temps allant de 1995 à 2010. Ces données, couvrant les deux districts, ont été intégrées dans des bases de données contenant 80 tables, dont certaines ont un nombre de lignes important.

2.2 Objectif du projet ADQUEAU

Le projet ADQUEAU, financé par le Conseil Scientifique de l'ENGEES (École Nationale du Génie de l'Eau et de l'Environnement), regroupe des chercheurs des laboratoires ICube et LIVE (équipe composée des experts hydro-écologues). Il a pour objectifs de faire collaborer des thématiciens (hydro-écologues) et des informaticiens. L'objectif du projet ADQUEAU est de construire des clusters (groupes d'objets similaires) à partir de séquences de données numériques de qualité de l'eau issues de stations de mesures sur des rivières. Ces données sont stockées dans des bases de données Fresqueau. Les clusters obtenus seront utilisés comme base de construction des classes thématiques par une opération que nous pourrions qualifier de sémantisation. Le projet se déroule en deux phases correspondant chacune à une année universitaire.

La première année, il s'agit, grâce à deux stagiaires (dont un thématicien et un informaticien financé directement par l'équipe SDC d'ICube), de recenser et mettre à jour et en forme les données disponibles. J'ai donc été associé à ce projet en tant que stagiaire informaticien. Parallèlement, une adaptation de la plateforme FoDoMuST et de son interface MultiCube (permettant le clustering sous contraintes) développée par l'équipe SDC est faite. Les premières expériences de clustering sous contraintes doivent être menées.

En deuxième année, il s'agira de valider (grâce à un stagiaire thématicien) l'approche de clustering sous contraintes interactives dans le domaine concerné et de publier les résultats. Une comparaison pourra être menée avec les résultats obtenus par la recherche de motifs.

L'objectif de ce projet est double. Il s'agit, par analyse de cette masse de données collectées, de répondre à deux enjeux scientifiques :

- Mettre en évidence des liens entre différentes métriques permettant de caractériser la qualité des cours d'eau.
- Relier les sources de pressions sur le milieu à la qualité physico-chimique et biologique des cours d'eau.

Mes travaux s'inscrivent dans l'objectif de la première année du projet qui est de recenser et mettre à jour et en forme les données disponibles et d'adapter l'interface MultiCube pour permettre le clustering sous contraintes.

2.3 Dynamique dans la réalisation du projet

Le laboratoire LIVE, en particulier les membres de l'ENGEES, maintient les bases de données Fresqueau et possède toutes les expertises thématiques nécessaires à leur analyse. L'équpe SDC apporte son expertise en analyse de données à travers de nouvelles méthodes sous contraintes. Elle est composée d'experts en Data science.

Notre projet s'inscrit dans une collaboration des membres de ces deux structures en vue d'adapter l'outil d'analyse FoDoMuST développé par l'équipe SDC aux données de l'équipe LIVE ainsi que de faciliter l'exploitation des résultats pour mettre en évidence des liens entre différentes métriques permettant de caractériser la qualité des cours d'eau.

2.3.1 Échanges des flux de données entre les équipes

L'extraction de jeux de données à partir de bases de données existantes **(a)** nécessite une connaissance forte sur leurs potentialités. De ce fait cette tâche **(1)** indispensable et cruciale est faite par les thématiciens du LIVE. Les jeux de données **(b)** extraits décrivent des caractéristiques physico-chimiques et biologiques des cours d'eau et sont transmises **(2)** aux spécialistes de traitement de données d'ICube. Ceux-ci, en fonction de la demande, étudient le problème en vue de modifier **(3)** les algorithmes existants ou d'en développer de nouveaux pour répondre à la demande. Les résultats (c) des traitements **(4)** sont retournés aux experts du LIVE pour analyse thématique **(6)**.
La figure ci-dessous montre les différents flux de données et les traitements mis en oeuvre.

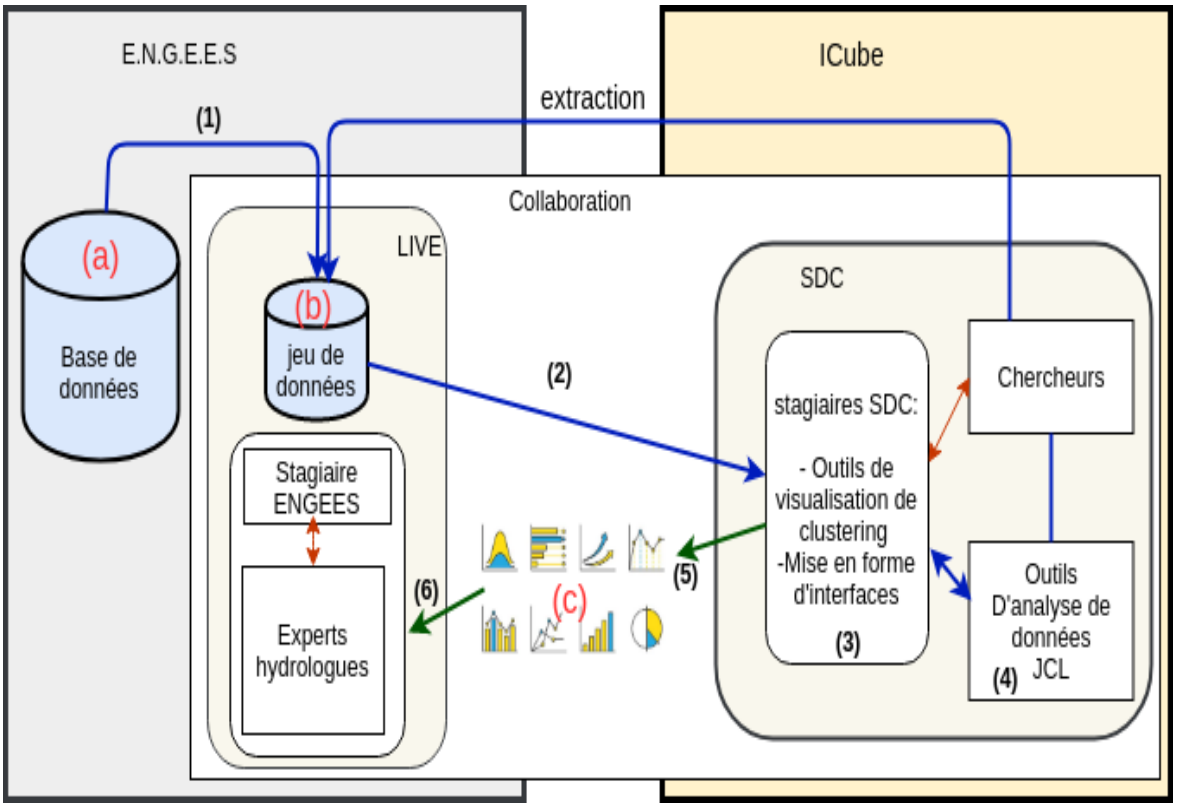


FIGURE 2.1 – Flux de données et les traitements

2.3.2 Organisation du projet

Afin de garantir une collaboration efficace nous avons organisé de nombreux points de rencontre.

Réunion plénière

Des réunions se tiennent régulièrement à propos du projet.

- La première rencontre a eu pour objectif, une présentation générale du projet, avec la définition des premiers besoins.
- Les rencontres qui ont suivi, ont eu pour objectif de suivre l'avancement du projet.
- Enfin une réunion finale a permis de faire le point sur le projet et de définir les axes de recherches pour l'année suivante.

Réunion secondaire Les rencontres secondaires étaient des rencontres avec un stagiaire de L'ENGEES (HONDA Gabriel), qui exprimait les besoins définis avec ses encadrants afin de mieux adapter les outils aux demandes pour fournir des résultats cohérents aux experts du domaine.

Propositions de solution aux encadrants

Toutes nos propositions de solution permettant d'améliorer la visualisation des résultats ou l'analyse des données ont été soumises à nos encadrants qui les valident selon leur efficacité.

2.4 Définitions

Une analyse temporelle consiste à étudier un phénomène au cours du temps (évolution des cours d'eau, variation des températures, suivi de la production agricole, suivi des ventes commerciales ...) à partir de données dites temporelles issues de capteurs sur ce phénomène et prises à des moments différents de la réalisation de celui-ci.

2.4.1 Les données temporelles

Les données temporelles, sont très souvent représentées sous forme de **séries temporelles** c'est-à-dire d'une suite de valeurs correspondant généralement chacune à l'évolution d'une donnée : par exemple, suite de valeurs prises par un capteur de pollution, série d'images prises par un satellite, etc. On parlera plus souvent de **séquences temporelles** lorsque les données considérées sont d'un type symbolique ou catégoriel (État de pollution, classe d'appartenance, ...). Partant de ces séries, l'analyse temporelle peut être vue suivant deux angles principaux.

2.4.2 Vision chronologique versus chronométrique

Chaque donnée d'une série temporelle est la captation d'un **évènement** produit par le phénomène. D'un point de vue **chronologique**, on s'intéresse uniquement à l'instant auquel l'événement a eu lieu. Ainsi,

- la durée et l'écart entre ces évènements ne sont pas pris en compte.
- seul l'ordre de réalisation de ces événements est considéré.

— Exemple :

- Etats pris par un bloc urbain, valeurs d'un pixel, etc.

Contrairement à la vision **chronologique**, la vision **chronométrique** consiste à étudier le phénomène en s'intéressant à la durée de chaque évènement de ce phénomène.

— Le temps est pris en compte :

- Dans la durée et l'écart entre des « évènements »
- L'ordre de réalisation de ces événements est considéré.

— Exemples :

- Suivi d'une culture.
- Suivi de la croissance de la population.

2.4.3 Analyse prospective versus rétrospective

L'objectif de l'expert peut être soit de s'intéresser au passé soit au futur. Dans le premier cas, l'objectif de son analyse est de comprendre les causes et conséquences d'un phénomène passé (ou en cours). Dans le second cas, il cherche à extraire des données et des modèles permettant de prévoir l'évolution du phénomène.

— L'analyse **rétrospective**

Elle consiste à tenter de comprendre ou caractériser le passé à partir des données disponibles : quelles ont été les grandes tendances dans l'évolution de la pollution ? Pourquoi cette rivière a eu une évolution différente des autres, etc.

— L'analyse **prospective**

Elle consiste à tenter d'extraire des données disponibles, des informations sur l'évolution future du phénomène étudié : quel sera le taux de pollution à venir ? quelle est l'évolution potentielle d'une rivière ? quelle sera la température des les jours à venir ? quels seront les clients à forte potentielle dans l'achat des nouveaux produits ? etc.

Il a été décidé de limiter dans ce projet, nos travaux à une approche rétrospective chronologique.

2.4.4 Les composantes des séries temporelles

Les séries temporelles présentent l'évolution d'un phénomène qui peut prendre plusieurs formes telles que :

La tendance à long terme (ou trend)

La tendance représente le mouvement profond de l'évolution à long terme du phénomène.

Les variations saisonnières

Les variations saisonnières ou la saisonnalité des fluctuations périodiques s'équilibrent autour de la tendance à court terme. Les variations saisonnières ont de multiples causes : cycle des saisons, dispositions réglementaires, dont les effets se produisent à date fixe.

Les ruptures temporelles

Les ruptures temporelles correspondent soit à des évènement irréversibles telles que les catastrophes soit à des évènements singuliers (coupures forestières, construction d'une route, ...). Elles font intervenir des composantes conjoncturelles ou accidentelles pour tenir compte des phénomènes particuliers, limités dans le temps (grèves, actions volontaristes ou publicitaires).

2.5 Les données Fresqueau

2.5.1 La description des données

Les données collectées portent sur deux grands bassins hydrographiques, correspondant aux districts Rhin-Meuse (33.000 km2) et Rhône-Méditerranée et Corse (130.000km2), pour une période de temps allant de 1995 à 2010. Ces données, couvrant les deux districts, ont été intégrées une base PostgreSQL/PostGIS. Cette base contient 80 tables, dont certaines ont un nombre de lignes important. On trouve notamment plus de cinq cent milliers de lignes correspondant à des mesures climatiques, plus de quatorze millions de mesures pour la physico-chimie, plus de neuf millions d'exploitations dans le registre parcellaire graphique, plus de huit millions de bâtiments et plus d'un million de tronçons hydrographiques. De plus vingt-deux des tables possèdent au moins un attribut représentant une géométrie. Des données physico-chimiques et biologiques couvrant la France entière pour la période 2007-2013, ont également été acquises dans le cadre d'un projet financé par l'ONEMA (2015-2016). L'association de dates aux données font d'elles des données chronologiques.

2.5.2 Les problèmes liés aux données

De nombreuses difficultés sont rencontrées dans le traitement des séries temporelles. En effet les données sont généralement bruitées, peuvent contenir des valeurs redondantes, et dans la majorité des cas elles sont de longueurs différentes (nombres d'événements différents) ou de durées différentes. En particulier dans le cas des données sur lesquelles nous travaillons, qui sont des données environnementales liées à la qualité biologique et physico-chimique des cours eau, nous avons pu identifier les problèmes suivants.

- Une forte hétérogénéité des données.
- Des données manquantes.
- Des séquences de longueurs variables qui engendrent un problème d'inadéquation de l'application de certaines métriques de calcul de distance telle que la distance euclidienne.

— Le manque de connaissance sur le type des phénomènes analysables pour ces données.
— Et pour terminer, nous avons le problème de distorsion (la distorsion ici est due à la variabilité des dimensions séquentielles) temporelle des phénomènes.

De nombreuses méthodes et techniques existent pour la résolution de ces problèmes.

Chapitre 3

Étude de l'existant et les travaux connexes

Ce chapitre fait l'objet de l'étude de l'existant (système d'analyse de données FoDoMuST) tant ces librairies que ces interfaces et des travaux connexes relatifs à l'analyse des données temporelles afin de mieux comprendre le sujet.

3.1 Étude de l'existant

FoDoMuST (Fouille de données Multi-Stratégie Multi-Temporelles) est un environnement d'analyse de données développé et maintenu par l'équipe SDC.

Librairies

La plateforme FoDoMuST est composée de deux librairies principales que nous présentons avec chacune son rôle.

— JCL qui est une librairie de clustering en java et développée par l'équipe SDC d'ICube. En résumé c'est une librairie composée de classifieurs.

— JSL qui est une bibliothèque d'algorithmes de segmentation de données soit propres à ICube soit proposés par l'Orfeo Tool Box (OTB).

Interfaces

FoDoMuST contient trois interfaces dédiées chacune à une famille d'applications différentes. Ces trois interfaces sont :

— Classifx qui est l'interface dédiée à l'analyse et à la classification de données temporelles de format ARFF.

— MultICube qui est l'interface dédiée à l'analyse et la classification de l'analyse de séries temporelles d'images.
— Ivisualize qui est l'interface dédiée à l'analyse et à la classification des séries temporelles géographiques.

Architecture de FoDoMuST

La plateforme d'analyse FoDoMuST est un outil qui permet le traitement des données de type image, des données de type shapes et celles de type arrf. La figure ci-dessus est une présentation de la structure de FoDoMuST.

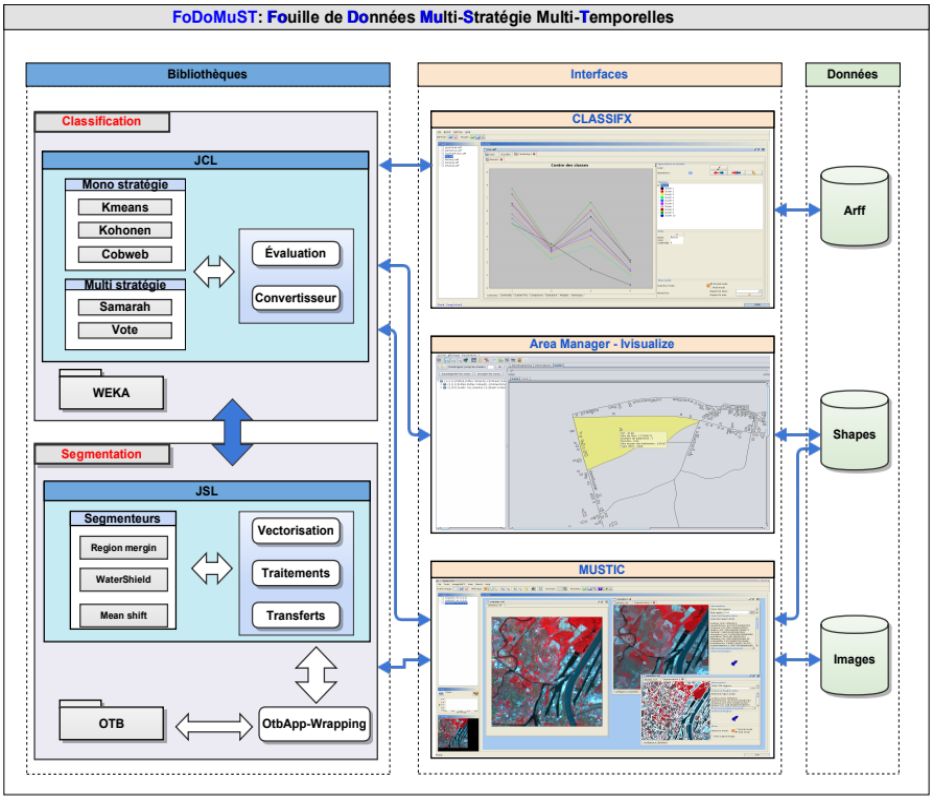


FIGURE 3.1 – Structure de FoDoMuST
[13]

3.2 Travaux connexes

Le clustering des données, en particulier des données temporelles reste un challenge dans le monde d'apprentissage. Ainsi nous nous intéressons aux travaux qui ont été effectués dans le domaine de l'analyse de séries temporelles afin de mieux comprendre comment de tels problèmes sont traités et quels ont été les résultats obtenus lors des expériences au cours de ces travaux. Dans une première approche nous allons étudier les algorithmes clas-

siques du machine learning permettant l'apprentissage non supervisé et dans une deuxième approche nous allons étudier les algorithmes du deep learning permettant l'apprentissage non supervisé.

3.2.1 Approche des algorithmes classiques du machine learning

Dans l'article [3] deux algorithmes ont été mixés pour le regroupement des thématiques sur le climat. Il s'agit de l'algorithme de CHA (Classification Hiérarchique Ascendante) pour la partie construction des premiers centres qui a permis dans ces travaux d'améliorer la sélection des centres et la qualité de la classification. Ainsi, de l'algorithme SWAP qui lui est lancé à partir des centres issus de CAH et toutes les permutations des centres possibles sont faites à ce niveau. Enfin on calcule leur effet sur la somme des dissimilarités entre les centres et les autres individus. Dans l'article [2] les travaux sont expérimentés sur des jeux de données personnelles de voyage basées sur le GPS et recueillies dans une base de données de Shanghai pour comparer les résultats des algorithmes T-DBSCAN et DBSCAN [6]. Les résultats lors de ces expériences ont indiqué que T-DBSCAN améliore de façon efficace à la fois la précision et la vitesse de calcul dans la segmentation de trajectoire. Les algorithmes classiques n'ayant pas tous la capacité d'extraire les motifs les plus importants des données temporelles pour le clustering, des algorithmes d'extraction sont utilisés avant de passer à la phase de clustering. Parmi ces algorithmes d'extraction, l'algorithme TSFRESH : Time Series FeatuRe Extraction Scalable Hypothesis [5] a été utilisé pour une extraction rapide d'un grand nombre de fonctions compatibles à l'apprentissage automatique. Parmi ces fonctions extraites nous pouvons citer le nombre de pics d'une série, la valeur moyenne d'une série, les maximums d'une série, les minimums d'une série, la statistique de symétrie par inversion du temps, ainsi que d'autres motifs importants et plus complexes. TSFRESH a eu son application sur terrain pour de nombreux avantages dont sa robustesse, son filtrage des motifs est correct du point de vue statistique et mathématique, sa riche documentation, mais surtout sa flexibilité et extensibilité. Ces avantages permettre de réduire le temps consacré à l'analyse des séries temporelles et d'avoir plus de temps de faire de l'apprentissage profond du problème afin de construire de meilleurs modèles.

La figure ci-dessous illustre la structure fonctionnelle du package TSFRESH.

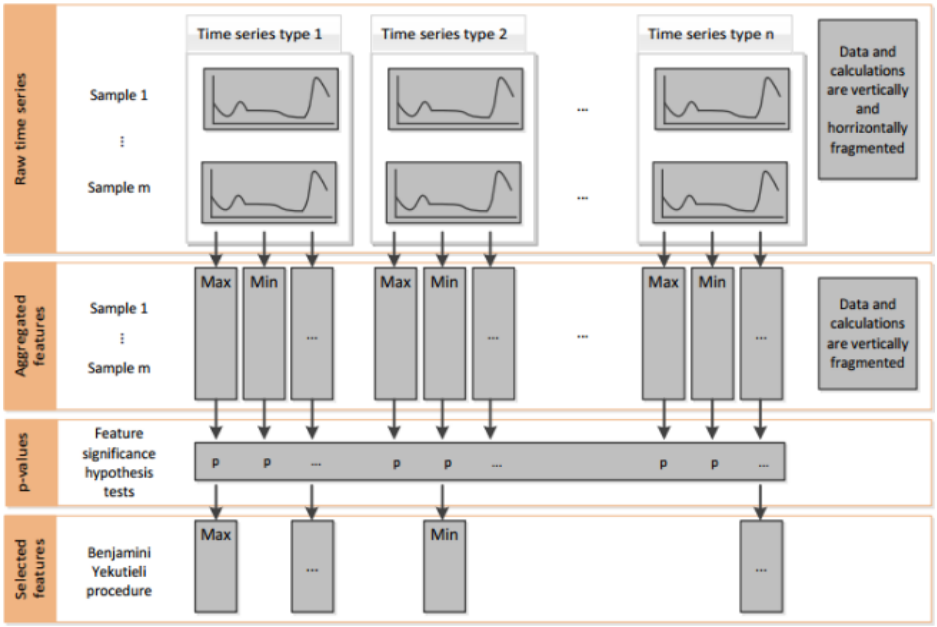


FIGURE 3.2 – Structure fonctionnelle de TSFRESH
[15]

Les séries chronologiques contiennent souvent du bruit, des redondances ou des informations non pertinentes. En conséquence, la plupart des fonctionnalités extraites ne seront pas utiles pour la tâche d'apprentissage automatique en cours. Ainsi il dispose d'une procédure de filtrage intégré. Il est basé sur la théorie bien développée du test d'hypothèse et utilise une procédure de test multiple. En conséquence, le processus de filtrage contrôle mathématiquement le pourcentage de caractéristiques extraites non pertinentes.

3.2.2 Approche des algorithmes du Deep Learning

Les progrès les plus passionnants proviennent de l'exploration d'algorithmes d'apprentissage non supervisés pour les modèles génératifs, tels que les Deep Belief Networks (DBN) et les Denoised Auto-encoders (DA) [7]. De nombreux modèles de génération profonde sont développés à partir de modèles basés sur l'énergie ou de codeurs automatiques. L'auto-encodage temporel est intégré aux machines de Boltzmann à restriction (RBM) pour améliorer les modèles génératifs [2]. Des machines de Boltzmann restreintes empilées (RBM) ou des réseaux de neurones convolutifs (CNN) ont été déployés pour les tâches de classification de séries temporelles de type images et DTC (Deep Temporal Clustering) [11] qui est une méthode très récente qui consiste à intégrer à la fois la réduction de la dimensionnalité et le clustering temporel dans un unique framework d'apprentissage de bout en bout.

Chapitre 4

Techniques et méthodes

Nous présentons dans ce chapitre l'étude des techniques et des méthodes pour l'analyse des données temporelles non étiquetées. Mais avant tout nous définissons les principales familles d'apprentissage.

4.1 Apprentissage automatique

L'apprentissage automatique implique la conception et le développement de programmes qui améliorent leurs modes de fonctionnement par acquisition de connaissances et aptitudes nouvelles. L'apprentissage automatique met au centre des algorithmes qui permettent aux machines d'apprendre d'elles-mêmes à partir de données. Les algorithmes apprennent de leurs erreurs pour développer par la suite des meilleurs résultats sans intervention humaine. L'apprentissage automatique a pour objectif de concevoir des programmes pouvant s'améliorer automatiquement avec l'expérience. Il existe principalement quatre grandes familles d'apprentissage qui sont :

— L'apprentissage supervisé.
— L'apprentissage non-supervisé.
— L'apprentissage semi-supervisé.
— L'apprentissage par renforcement.

4.1.1 L'apprentissage supervisé

Dans le cas de l'apprentissage supervisé, la machine s'appuie sur des classes prédéterminées et sur un certain nombre de paradigmes connus pour mettre en place un système de classement à partir de données déjà cataloguées. Dans ce cas, deux étapes sont nécessaires pour compléter le processus, à commencer par une phase d'apprentissage qui consiste en la modélisation des données cataloguées. Ensuite, il s'agira au second stade

de se baser sur les modèles ainsi définis pour attribuer des classes aux nouvelles données introduites dans le système, afin de les cataloguer elles aussi.

4.1.2 L'apprentissage non supervisé

C'est l'apprentissage sur des données non étiquetées. Dans ce type d'apprentissage il n'y a pas de données cibles comme dans le cas de l'apprentissage surpervisé.

4.1.3 L'apprentissage semi-supervisé

L'apprentissage semi-supervisé est l'apprentissage qui mixe à la fois l'apprentissage supervisé et non-supervisé.

4.1.4 L'apprentissage par renforcement

L'apprentissage par renforcement repose sur un système de récompenses et de pénalités pour permettre à l'ordinateur d'apprendre à résoudre un problème de manière autonome. Le programmeur humain se contente de modifier l'environnement d'apprentissage et d'effectuer des modifications sur le système de récompenses. Cette méthode est particulièrement pertinente lorsqu'il n'existe pas de façon unique d'accomplir la tâche demandée, mais que des règles doivent être respectées.

La figure ci-dessous est une représentation des principales familles d'apprentissage.

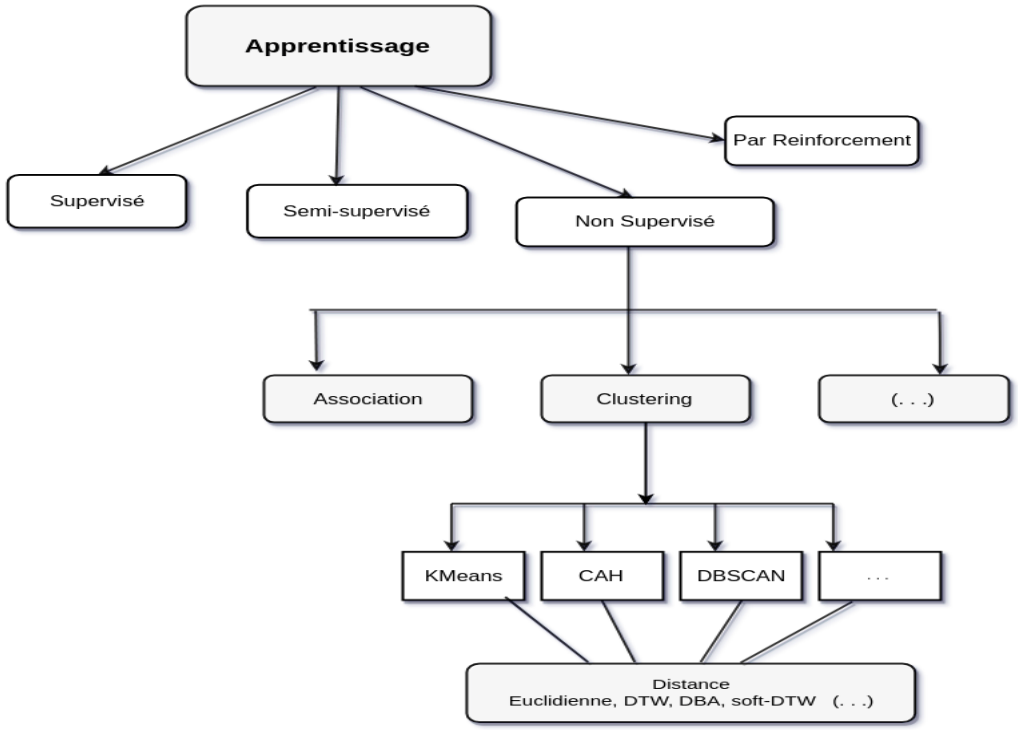


FIGURE 4.1 – Apprentissage

Dans nos travaux nous nous limitons à l'apprentissage non supervisé. En particulier le clustering, qui consiste à structurer automatiquement un ensemble de données en groupes (clusters) les plus homogènes possibles. Pour cela, il existe un grand nombre de méthodes dont les principales et plus utilisées sont basées sur l'utilisation d'une mesure de similarité entre les objets. Ainsi, par exemple, le plus connu des algorithme **Kmeans**, se base sur une mesure de distance mais aussi sur un mécanisme de moyennage. Il utilise donc très souvent une distance et une moyenne arithmétique.

4.2 Description de l'algorithme Kmeans

Kmeans comme tous les algorithmes de clustering de façon générale a pour objectif commun de regrouper dans des clusters des éléments en fonction de critères de similarités dans des clusters. Ces éléments peuvent être de tout type, du moment qu'ils sont encodés dans une matrice de données et sont munis d'une distance et d'une moyenne.

Kmeans a une capacité à traiter de très grandes bases de données. Et lors des différents calculs seuls les vecteurs des moyennes sont à conserver en mémoire centrale, ce qui explique sa rapidité. La complexité de **Kmeans** est d'ordre linéaire par rapport au nombre d'observations. Pour résumer **Kmeans** est simple d'utilisation, et robsute. Il est facile à comprendre et il permet d'avoir rapidement un premier résultat. L'algorithme **Kmeans** est basé sur un calcul d'inertie donné par la formule suivante :

$$\sum_{r=1}^{k} \sum_{x_i \in c_r} (x_i - g_r)^2$$

- Cr est la classe numéro r.
- Xi est un individu dans une classe.
- Gr est le centre de classe Cr.

4.2.1 Fonctionnement de Kmeans

L'algorithme **Kmeans** est un algorithme de regroupement d'objets qui peut être de toute sorte. C'est un algorithme qui cherche à maximiser la similarité intra-cluster et à minimiser l'inertie inter-classe en utilisant comme mesure de similarité une fonction de distance (distance euclidienne, Dynamic Time Warping, distance de Minkowski, distance de Manhattan), ceci en fonction de la nature des données. Pour l'utilisation de **Kmeans** nous avons besoin de :

— Transformer le jeu de données sous forme de matrices de données.
— Fixer le nombre de clusters (groupes) a priori : k.
— Initialiser aléatoirement k centres (k individus tirés au hasard comme représentants des clusters).
— Affecter chaque individu au cluster (groupe) qui minimise la distance entre le centre et l'individu (centre le plus proche).
— Recalculer les nouveaux centres (la moyenne).
— Itérer l'opération jusqu'à la stabilité.

Le but de **Kmeans** est de rechercher des structures naturelles dans les données pour construire des groupes automatiquement. À la sortie du traitement, nous obtenons un

ensemble de clusters compacts et clairement séparés, sous réserve de choisir le bon nombre de clusters à former. Ainsi l'utilisation de **Kmeans** peut s'avérer nécessaire dans la segmentation de la clientèle en fonction d'un certain nombre de critères définis, dans l'exploration de données en Data Minning pour la compréhension des structures de données, dans le clustering des documents, etc.

4.2.2 Limites de Kmeans

Optimums locaux :

L'application de **Kmeans** sur un même jeu de données peut donner des partitionnements différents, cela est dû au fait que les centroïdes initiaux sont généralement choisis aléatoires et l'algorithme trouve des clusters relatifs aux centroïdes initiaux. Ainsi nous pouvons avoir une configuration de clusters qui n'est pas optimale. De plus **Kmeans** est un algorithme qui est très sensible aux valeurs aberrantes.

Résolution du problème d'optimums locaux :

Pour résoudre ce problème nous allons utiliser la fonction de coude pour retrouver le nombre optimal de clusters. L'idée de la méthode du coude est d'exécuter le regroupement de **Kmeans** sur l'ensemble de données pour une plage de valeurs de k, et pour chaque valeur de k calculer la somme des erreurs quadratiques. Une représentation graphique de l'erreur quadratique en fonction du nombre de clusters sur la plage définie nous permet d'identifier la distorsion de la courbe que l'on appelle coude. En résumé nous pouvons construire la fonction de coude en suivant les étapes suivantes :

- Former un certain nombre de modèles **Kmeans** en utilisant differentes valeurs de k.
- Noter le coefficient de silhouette moyen à chaque entraînement.
- Représentation de la silhouette en fonction du nombre de clusters (k).

La figure ci-dessous représente une fonction de coude avec une valeur de **K=4** qui permet de faire un meilleur regroupement des données en 4 groupes bien distincts.

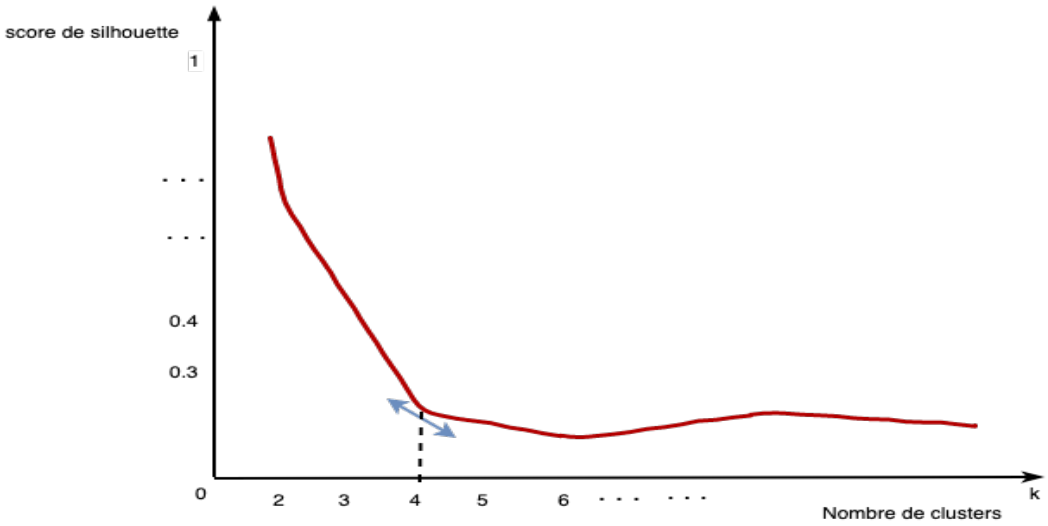


FIGURE 4.2 – Fonction de coude

La formule du coefficient de silhouette est la suivante :

$$S(obj) = \frac{B(obj) - A(obj)}{max\{A(obj), B(obj)\}} \text{ avec S(obj)} \in [-1,1]$$

où

- S(obj) : le coefficient de silhouette du point de donnée **obj**.
- A(obj) : la distance moyenne entre **obj** et tous les autres points de données du cluster auquel il appartient.
- B(obj) : la distance moyenne minimale de **obj** à tous les clusters auxquels **obj** il n'appartient pas.

Nous pouvons également nous servir de la valeur de coefficient de silhouette pour évaluer le modèle **Kmeans** sur la base de la variation de celui-ci.

- Lorsque $S(obj) = 1$, on dit que **obj** est vraiment compact avec le cluster auquel il appartient.
- Lorsque $S(obj) = -1$, on dit que **obj** n'est pas compatible avec le cluster auquel il appartient.
- Lorsque $S(obj) \approx 0$, cela correspond à la zone de chevauchement entre les clusters.

La comparaison entre individus nécessite une mesure de similarité et Kmeans utilise comme mesure de similarité une fonction de distance. Ainsi nous abordons dans la section qui suit les fonctions de distances.

4.3 Distances temporelles

De nombreuses formules de calcul de distance entre séries temporelles existent, cependant elles ne fournissent pas les mêmes résultats en fonction des problèmes traités. Chacune des formules peut donc être mieux adaptée à un problème type.

Distance Euclidienne

Cette formule de calcul de la distance est faite de façon linéaire. Elle ne permet pas la prise en compte des distorsions, des variations des dimensions.

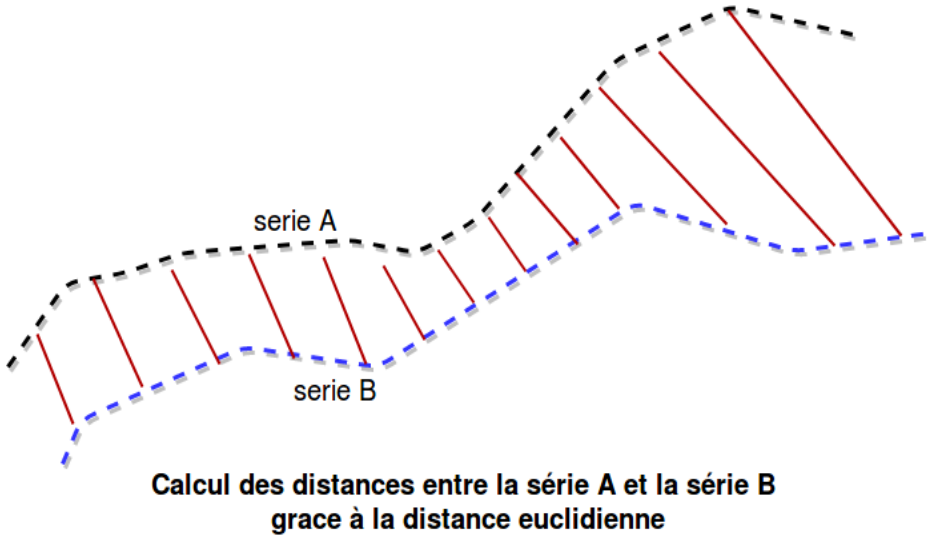


FIGURE 4.3 – Représentation de calcul avec la distance euclidienne

DTW

DTW qui signifie en anglais Dynamic Time Warping, est un algorithme qui permet de trouver l'alignement global optimal entre deux séquences temporelles en minimisant les coûts d'association. DTW associe les points des courbes non-linéaires de façon à respecter certains critères. Le but de l'algorithme est de trouver l'association des points qui minimise la distance entre les deux séries temporelles.

Cette méthode prend en compte la distorsion c'est-à-dire la non-linéarité des courbes représentant l'évolution du phénomène, la variation de la dimension des séries mais aussi des valeurs manquantes. Elle nous permet de déterminer la distance optimale entre deux séries chronologiques. Elle parvient à extraire des similarités que la distance euclidienne n'arrive pas à extraire. Pour ce faire, elle se base sur un calcul récursif des distances entre les deux séquences considérées et à partir de sa matrice de coûts obtenue, le chemin de

coût minimal est estimé [4][12]. Pour ce faire, une matrice de coût de taille **nxm** avec **n** et **m** respectivement les longueurs des deux séries est construites. À partir de cette matrice de coût le chemin de coût minimum d'alignement entre ces séries est déterminé. Elle est définie récursivement par la formule suivante :

$$D(A_i, B_j) = \delta(a_i, b_j) + \min \begin{cases} D(A_{i-1}, B_{j-1}), \\ D(A_i, B_{j-1}), \\ D(A_{i-1}, B_j) \end{cases}$$

FIGURE 4.4 – Formule de DTW

où

- Ai représente la sous-séquence $\boldsymbol{a1, \ldots, ai}$.
- $\delta(ai, bj)$ représente la distance entre deux séries par rapport à une même variable (attribut).
- Le coût de l'alignement optimal est alors donné par $\boldsymbol{D(}A_{|A|}\boldsymbol{,}\ B_{|B|}\boldsymbol{)}$ qui représente la distance totale des $\delta(ai, bj)$.

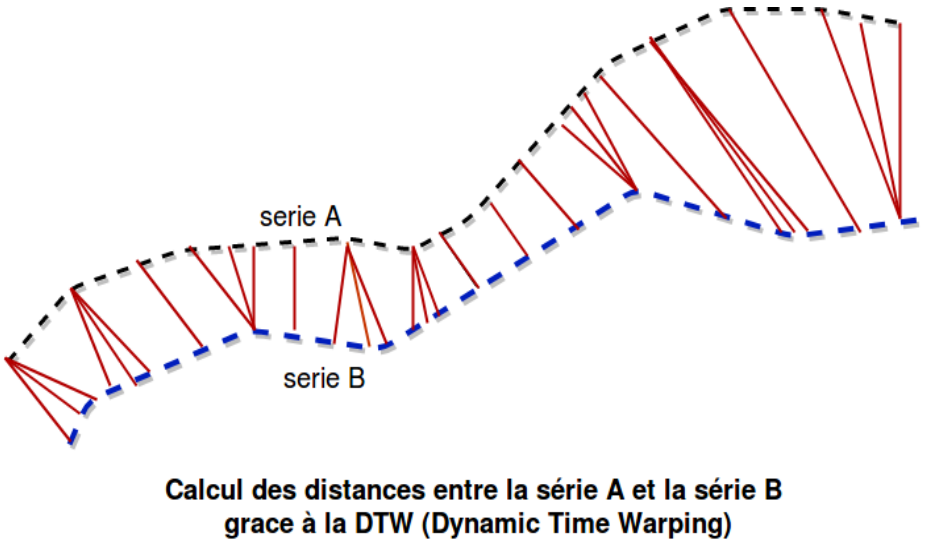


FIGURE 4.5 – Représentation de calcul de distance avec DTW

Soft-DTW

DTW définit l'écart entre deux séries temporelles pouvant être de longueurs variables. Les séries de l'article [8] montrent que DTW ne peut être calculé que dans un temps quadratique. Ainsi pour pallier les limites de DTW pour le calcul dans le temps non quadratique soft-DTW a été développé. Le soft-DTW est une forme lissée de DTW, qui calcule le minimum souple de tous les coûts d'alignement. [9]

4.3.1 Moyenne DBA

DBA (DTW barycenter averaging) est une méthode de calcul de la moyenne qui consiste à affiner de façon itérative une séquence moyenne afin de minimiser sa distance (DTW) au carré par rapport aux séquences moyennes. L'objectif est de réduire au minimum la somme des distances DTW quadratiques par rapport à la séquence moyenne à l'ensemble des séquences. Cette somme est formée par distances simples entre chaque coordonnée de la séquence moyenne et les coordonnées des séquences qui lui sont associées. Ainsi, la contribution d'une coordonnée de la séquence moyenne à la somme totale de la distance au carré est en fait la somme des distances euclidiennes entre cette coordonnée et les coordonnées des séquences qui lui sont associées lors du calcul du DTW. Pour chaque coordonnée de la séquence la séquence moyenne est obtenue en calculant le barycentre de l'ensemble de coordonnées. Pour chaque itération DBA fonctionne en deux étapes.

— la première étape consiste à calculer DTW entre chaque séquence individuelle et la séquence moyenne temporaire à affiner, afin de trouver des associations entre les coordonnées de la séquence moyenne et les coordonnées de l'ensemble des séquences.

— La seonde consiste à faire la mise à jour de chaque coordonnée de la séquence moyenne comme barycentre des coordonnées qui lui sont associées lors de la première étape.

La figure suivante est une représentation du calcul de la moyenne de 2 séries temporelles.

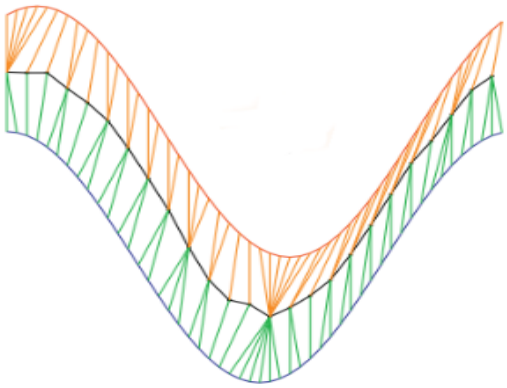


FIGURE 4.6 – Représentation du calcul de la moyenne avec DBA

Les courbes à l'extrémité de la figure représentent celles des deux séries et celle de la moyenne au centre de la figure.

4.3.2 Limites de DTW et soft-DTW

La conséquence de ces méthodes est qu'elles ne tiennent pas compte du temps séparant deux éléments des séquences. Elles conduisent ainsi à une analyse chronologique et non une analyse chronométrique. [10]

4.4 Autres approches

DTC (Deep Temporal Clustering) [11] est une méthode très récente qui consiste à intégrer à la fois la réduction de la dimensionnalité et le clustering temporel dans un unique framework d'apprentissage de bout en bout. DTC a une structure à 3 niveaux dont le premier est une implémention de CNN qui permet la réduction de la dimensionnalité des données et l'apprentissage sur la forme ondulatoire des séries ; le second niveau est l'implémentation de BI-LSTM pour réduire davantage la dimensionnalité des données et l'apprentissage sur la connexion entre les formes ondulatoires à travers toutes les échelles temporelles ; le troisième niveau permet d'amélioration le clustering non paramétrique obtenu à partir des BI-LSTM. DTC permet d'inclure une visualisation dans la formation des clusters obtenus. La figure ci-dessous représente une l'architecture de DTC.

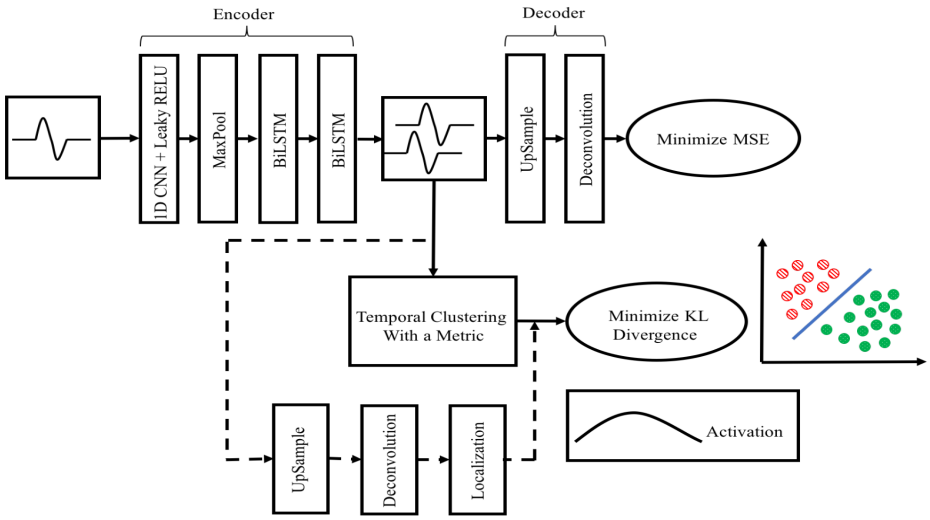


FIGURE 4.7 – Architecture de DTC [11]

Chapitre 5

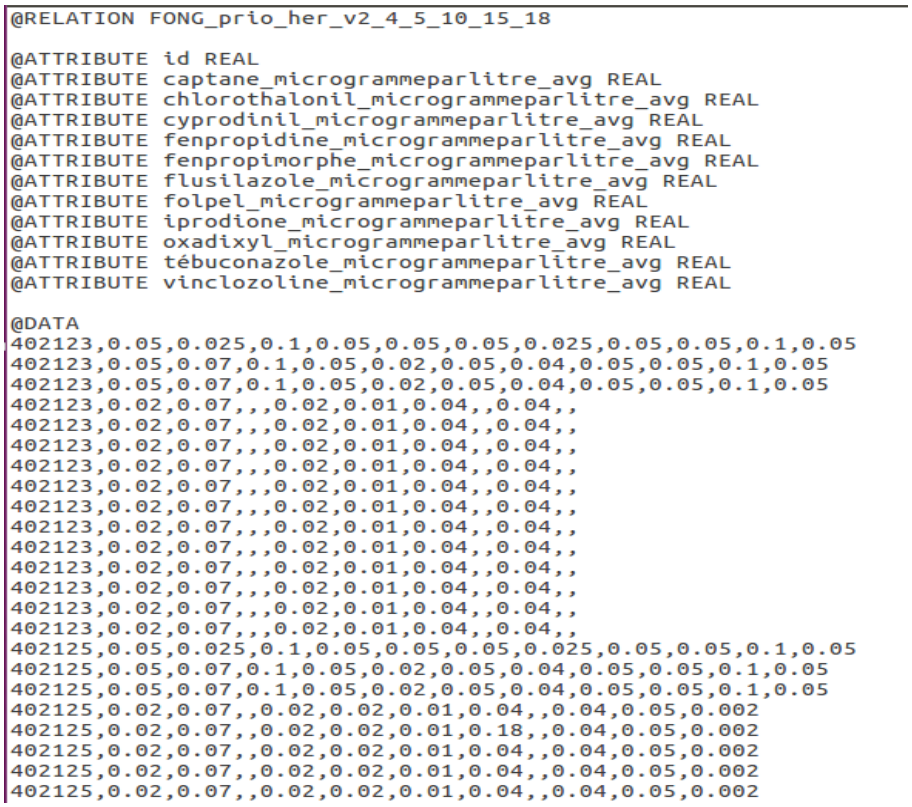
Implémentations et expérimentations

5.1 Implémentation

5.1.1 Problème du format des données extraites

Explication du problème

Le système d'analyse de données FoDoMuST utilise à la base un fichier de type **tiff** pour les données de type image et **arff** pour les données numériques ou catégorielles. Il y a eu au préalable un rendez-vous afin d'expliquer comment structurer ces fichiers. Malheureusement l'extraction de fichier de type **arff** à partir de base de données existantes était difficile, et laissait passer de nombreuses incohérences, telles que des caractères spéciaux, des espaces et bien d'autres dans les noms des attributs. Toutes ces incohérences observées dans les données rendaient difficiles leur compréhension. Malgré des efforts consentis pour la correction de ces problèmes observés dans les données, les incohérences demeuraient. Sur la figure ci-dessous est représenté un exemplaire de données au format **arff**.



```
@RELATION FONG_prio_her_v2_4_5_10_15_18

@ATTRIBUTE id REAL
@ATTRIBUTE captane_microgrammeparlitre_avg REAL
@ATTRIBUTE chlorothalonil_microgrammeparlitre_avg REAL
@ATTRIBUTE cyprodinil_microgrammeparlitre_avg REAL
@ATTRIBUTE fenpropidine_microgrammeparlitre_avg REAL
@ATTRIBUTE fenpropimorphe_microgrammeparlitre_avg REAL
@ATTRIBUTE flusilazole_microgrammeparlitre_avg REAL
@ATTRIBUTE folpel_microgrammeparlitre_avg REAL
@ATTRIBUTE iprodione_microgrammeparlitre_avg REAL
@ATTRIBUTE oxadixyl_microgrammeparlitre_avg REAL
@ATTRIBUTE tébuconazole_microgrammeparlitre_avg REAL
@ATTRIBUTE vinclozoline_microgrammeparlitre_avg REAL

@DATA
402123,0.05,0.025,0.1,0.05,0.05,0.05,0.025,0.05,0.05,0.1,0.05
402123,0.05,0.07,0.1,0.05,0.02,0.05,0.04,0.05,0.05,0.1,0.05
402123,0.05,0.07,0.1,0.05,0.02,0.05,0.04,0.05,0.05,0.1,0.05
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402123,0.02,0.07,,,0.02,0.01,0.04,,0.04,,
402125,0.05,0.025,0.1,0.05,0.05,0.05,0.025,0.05,0.05,0.1,0.05
402125,0.05,0.07,0.1,0.05,0.02,0.05,0.04,0.05,0.05,0.1,0.05
402125,0.05,0.07,0.1,0.05,0.02,0.05,0.04,0.05,0.05,0.1,0.05
402125,0.02,0.07,,0.02,0.02,0.01,0.04,,0.04,0.05,0.002
402125,0.02,0.07,,0.02,0.02,0.01,0.18,,0.04,0.05,0.002
402125,0.02,0.07,,0.02,0.02,0.01,0.04,,0.04,0.05,0.002
402125,0.02,0.07,,0.02,0.02,0.01,0.04,,0.04,0.05,0.002
402125,0.02,0.07,,0.02,0.02,0.01,0.04,,0.04,0.05,0.002
```

Figure 5.1 – Données au format arff

Résolution du problème de format arff

Suite à de multiples tentatives pour le fonctionnement du système avec des formats **arff**, nous avons fait des tests avec des fichiers **csv**. Ainsi après plusieurs tests sur des fichiers au format **csv**, nous avons constaté que ce type de format de données est le mieux adapté aux traitement de données texte par le système FoDoMuST. À l'issue de ces tests, nous avons proposé le format **csv** pour répondre à ces problèmes. La figure ci-dessous représente des données au format **csv**. L'utilisation du format **csv** au sein du système a nécessité le développement de fonctionnalités supplémentaires au système.

La figure ci-dessous représente un exemplaire de données au format **csv**.

id	Captane_microgr	Captane_microgr	Chlorothalonil_m	Chlorothalonil_micr	Cyprodinil_microg	Cyprodinil_microgramm	Fenpropidine
402123	0.05	10	0.025	10	0.1	10	0.05
402123	0.05	10	0.07	10	0.1	10	0.05
402123	0.05	10	0.07	10	0.1	10	0.05
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402123	0.02	10	0.07	10			
402125	0.05	10	0.025	10	0.1	10	0.05
402125	0.05	10	0.07	10	0.1	10	0.05
402125	0.05	10	0.07	10	0.1	10	0.05
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			0.02
402125	0.02	10	0.07	10			
402125	0.02	10	0.07	10			
402125	0.02	10	0.07	10			
402125	0.02	10	0.07	10			
402125	0.02	10	0.07	10			
402125	0.02	10	0.07	10			

Figure 5.2 – Données au format csv

Un fichier **csv** a comme premier avantage d'être lu par un très grand nombre d'outils gratuits ou payants, ce qui rend le format quasiment universel. Le fichier **csv** est majoritairement employé pour élaborer des bases de données. MySQL, PostgresSQL ou Oracle par exemple peuvent aussi bien importer qu'exporter des fichiers **csv**. L'autre avantage d'un fichier **csv** réside dans sa taille. Le fait qu'il n'y ait aucune mise en forme ni utilisation d'une police spéciale rend le fichier léger, plus petit, et pourtant tout aussi complet que n'importe quel autre fichier créé sur Microsoft Excel.

5.1.2 Première solution en ligne de commandes

Une première solution a été implémentée en ligne de commandes, et a servi de tests initiaux pour s'assurer que les traitements pouvaient être effectués à partir du système FoDoMuST grâce à la librairie JCL. Pour la mise en place de cette solution, il a fallu trouver un moyen d'interaction entre le format de données de Python et celui de JCL. En effet, il fallait effectuer un parsing entre le DataFrame en Python (DataFrame de Pandas) et le DataObject en Java. La communication entre les deux langages a donc nécessité

l'utilisation de package d'interaction. Nous avons donc utilisé JPYPE pour la résolution de l'interconnectivité entre les deux langages, car des tests antérieurs ont été faits avec cette bibliothèque et ont prouvé sa viabilité. Ces premiers tests ont soulevé les deux gros problèmes suivants.

La qualité des données

Le premier problème étant la qualité des données fournies par l'ENGEES. Les données qui nous sont transmises ne sont pas propres. Il y avait des valeurs manquantes, des colonnes vides, des colonnes sans nom, des valeurs dupliquées. Ceci a mis en avant un problème de la plateforme FoDoMuST, qui était l'absence de fonctionnalités de prétraitement sur les données fournies, et donc il a fallu ajuster le cahier des charges et ajouter cette tâche à la liste des fonctionnalités à implanter.

L'adaptation de DTW (Dynamic Time Warping)

Le second problème fut révélé pendant les tests. Pour nos premiers tests, nous avons effectué une imputation des valeurs manquantes par la valeur 0. Malgré cette imputation, nous avons constaté une erreur pendant l'exécution de DTW : la distance globale résultant de DTW était nulle et donc le clustering avait échoué. Le problème identifié était dû au fait que la matrice de coût était initialisée à partir de la taille de la première série ce qui faisait que si la taille de la seconde série était plus grande alors toute la longueur de celle-ci n'était pas prise en compte et il fournissait une valeur nulle, ce qui conduisait à des résultats non conformes. Ce problème a donc été résolu en prenant en compte la taille maximale des séries à comparer et en générant la matrice modèle à la dimension de la plus grande série.

Visualisation des résultats

Une fois le clustering effectué, il reste la problématique liée à l'interprétation des résultats par les experts. Il a fallu trouver une sortie adaptée à leurs demandes. Pour ce faire, des réunions avec un stagiaire de l'ENGEES ont été organisées afin de pouvoir lui présenter diverses solutions. La solution acceptée était une génération des tendances des clusters sous forme de courbe avec comme aide visuelle un affichage de seuil de qualité que l'on charge à partir d'un **csv** fournie par l'expert, ainsi que des générations de **csv** des tendances de chaque cluster et un **csv** qui fait le lien entre les identifiants des stations et leurs clusters respectifs.

Afin de faciliter l'utilisation de cette solution en ligne de commandes, nous avons proposé la seconde solution avec l'interface.

5.1.3 Solution avec interface graphique

Pour la continuité, il s'agissait de permettre l'interprétation des résultats directement au sein de la plateforme FoDoMuST. Malheureusement l'interface de visualisation existante était insuffisante à la visualisation des résultats. Il fut posé comme problématique de trouver une solution pour améliorer la visualisation des résultats sans surcharger l'interface afin de ménager les coûts d'entretien de cette plateforme. La solution proposée fut de transférer les fonctions d'affichage à un langage plus adapté à cet effet : Python.

L'utilisation d'un second langage interprété est différent d'un langage compilé au sein d'un programme. Lors de l'utilisation d'un langage compilé il suffit d'instancier les classes souhaitées afin de pouvoir les utiliser ou de faire un lien vers ces mêmes classes pour utiliser leur méthode static. Pour un langage interprété cela est différent, du fait que l'on utilise directement le langage au sein du programme. Nous allons prendre pour exemple l'utilisation du Python dans JAVA, il y a quelques problèmes :

- La compatibilité (seuls les types primaires le sont : int , double, float, String), il faut donc décomposer les classes créées afin de les transférer à Python.
- La récupération des données est bridée à des tableaux simples de type primaire.

Dans notre cas notre choix de bibliothèque pour l'interprétation Python au sein de JAVA s'est porté sur JEP et cela pour plusieurs raisons. JEP est une bibliothèque Java pouvant être utilisée pour interagir entre Java et Python.
Elle permet d'analyser et d'évaluer des expressions mathématiques. Jep intègre CPython dans Java via JNI et cette intégration a plusieurs avantages. Nous citons entre autres :

- L'utilisation de l'interpréteur Python natif qui peut s'avérer beaucoup plus rapide que les alternatives.
- L'accessibilité aux modules Python de haute qualité, à la fois les extensions natives CPython et celles basées sur Python.
- Les compilateurs et les outils Python assortis sont aussi avancés que le langage.
- Python étant un langage interprété, il permet d'activer des scripts de code Java établis sans nécessiter de recompilation, ceci augmentant l'efficacité.
- Java et Python sont tous deux multi-plateformes, ce qui permet un déploiement sur différents systèmes d'exploitation.
- La prise en charge de Numpy pour les tableaux primitifs Java.
- Son code est open source.

L'ouverture des sources et la possibilité de personnaliser la bibliothèque était importante dû au fait que l'on va l'utiliser pour l'affichage. Il fallait donc pouvoir y inclure des modules à cet effet.

L'implémentation de cette solution devait être simple et intuitive, et limiter les interactions non-pertinentes de l'utilisateur, par exemple demander un affichage de résultats avant même d'avoir effectué une classification. La solution la plus logique était donc une fenêtre contextuelle, qui changerait de fonctionnalité selon où l'on se situe. En clair, si on a chargé des données, on aura les options de prétraitement disponibles, ou si on a effectué une classification on aura les options d'affichage de résultats.

5.2 Prétraitement des données

Le prétraitement est un processus qui nous permet d'avoir un jeux de données propre en fonction des objectifs visés par notre étude. Il permet de faire une transformation de données non structurées vers des données structurées. Il rend enfin les données mieux interprétables par les machines.

5.2.1 Pourquoi est-il important d'avoir des données propres ?

De nombreuses raisons nous ont conduit à traiter les données non propres (présences de valeurs manquantes, variation des ordres de grandeur,...) pour les rendre propres avant tout apprentissage automatique. La première des raisons est que les données soient mieux opérables par les algorithmes d'apprentissage automatique, puisque nombreux sont les algorithmes d'apprentissage automatique qui ne supportent pas les données qui contiennent des anomalies. Un modèle d'apprentissage ne peut fournir de résultats cohérents qu'avec des données cohérentes par rapport aux objectifs visés. Avec des données non propres, la performance des modèles d'apprentissage automatique est limitée. L'impureté des données est source de la production des résultats fallacieux. Vue l'impact de l'impureté des données, il est donc évident pour les data scientists de consacrer le maximum de temps sur cette partie du processus d'apprentissage afin d'avoir des données répondant aux objectifs visés car c'est une étape cruciale. Le prétraitement est composé d'un ensemble d'étapes.

5.2.2 Nettoyage des données

Le nettoyage des données consiste à traiter les données manquantes, aberrantes, ou inconsistantes du jeu de données pour qu'il puisse être utilisable par les algorithmes de Machine Learning. Le nettoyage d'un jeu de données peut prendre plusieurs formes :

supprimer les données manquantes ou les remplacer les valeurs manquantes par des valeurs artificielles, remplacer les valeurs aberrantes par des valeurs artificielles (moyenne, médiane, interpolation de valeur), supprimer des valeurs dupliquées.

5.2.3 Imputation (remplacement) des valeurs manquantes

Plusieurs moyens d'imputation des données manquantes existent, mais avant tout, il est important de connaître les causes (Est ce que ces valeurs apparaissent de façon naturelle ou elles apparaissent de façon dépendante par rapport à d'autres valeurs ?) de valeurs manquantes pour mieux les traiter.
En supposant $Dataset = (X, Y)$ avec X la partie des données observées et Y celle des données manquantes Il y en a trois types de données manquantes :

- MCAR (Missing Complet At Random)

Ce sont les valeurs manquantes qui apparaissent complètement de façon aléatoire. Cela signifie que la probabilité que Y soit une valeur manquante pour un individu ne dépend pas des valeurs X ou Y.

- MAR (Missing At Random)

Cela signifie que la probabilité que Y soit une valeur manquante pour un individu peut dépendre des valeurs X, mais pas celles de Y.

- MNAR (Missing Not At Random)

Cela signifie que la probabilité que Y soit une valeur manquante pour un individu dépend de Y.[1]

Ainsi pour imputer les données manquantes, nous avons implémenté un ensemble de fonctions qui permettront aux experts, qui utiliseront le système, de faire cette imputation en fonction de la compréhension et des objectifs de leurs données :

- *Imputation des valeurs manquantes par la moyenne.*
- *Imputation des valeurs manquantes par la médiane.*
- *Imputation des valeurs manquantes par une valeur quelconque.*
- *Imputation des valeurs manquantes par la moyenne dans chaque station.*
- *Imputation des valeurs manquantes par la médiane dans chaque station.*
- *Imputation par Interpolation temporelle.*
- *Suppression des lignes vides.*
- *Suppression de ligne à partir d'un seuil de pourcentage de valeurs manquantes.*
- *Suppression d'une colonne vide.*
- *Suppression des doublons.*
- *Suppression d'une colonne spécifique à partir d'un seuil de pourcentage de valeurs manquantes.*

5.2.4 Visualisation des données

Pour que l'utilisateur puisse prendre conscience des traitements à effectuer sur les données, nous avons implémenté un ensemble de fonctionnalités d'affichage des caractéristiques des données avant de procéder à une imputation adéquate et bien d'autres fonctions qui permettent de visualiser les graphes obtenus après le clustering des données :

- *Afficher les statistiques de chaque série temporelle par attribut (max, min, moyenne, médiane).*
- *Identification du nombre total des valeurs manquantes par station en pourcentage.*
- *Nombre de valeurs manquantes par ligne en pourcentage.*
- *Courbe de coude.*
- *Profil temporel avec seuil.*
- *Profil temporel sans seuil.*
- *Génération de fichier* ***csv*** *avec le numéro de cluster pour chaque observation (station pour le cas de nos données) dans le jeu de données.*

5.2.5 Normalisation

La normalisation est un processus de mise à la même échelle de toutes les variables. La normalisation est nécessaire lorsque les ordres de grandeur des variables sont différents. La normalisation est requise dans notre cas, puisque la mesure de similarité utilisée est une fonction de distance, ainsi si les ordres de grandeur sont différents alors la variable ayant le plus grand ordre de grandeur aura tendance à déterminer la distance totale car les plus petites seront presque négligeables. Ce qui pourrait conduire à des modèles d'apprentissage qui ne reflètent pas la réalité. En plus lorsque les données ne sont pas normalisées les algorithmes d'apprentissage automatique mettent plus de temps de calcul pour trouver un modèle prédictif optimal. Plusieurs techniques de normalisation existent, chacune est utilisée en fonction des objectifs de l'analyse.

La technique de Min-Max

Le but de cette technique est d'avoir un intervalle restreint entre [0,1] pour toutes les variables du jeu de données. Cette technique préserve l'effet des valeurs aberrantes. Elle est donc utilisée lorsque l'on veut faire une mise à l'échelle tout en préservant l'effet des valeur aberrantes et elle est donnée par la formule ci-dessous :

$$X_{normalise} = \frac{X_i - X_{min}}{X_{max} - X_{min}}$$

La technique de MaxAbsScaler

Cette technique est recommandée pour la normalisation lorsque la distribution des valeurs est éparse avec assez de valeurs aberrantes. C'est une technique robuste et elle préserve les entrées nulles sur une répartition. Elle est vraiment adaptée pour une répartition anormale des données.

La technique RobustScaler

Le principe de cette technique est le même que celui de la technique de Min-Max, mais à la place de min et max c'est plutôt les intervalles interquatiles qui sont utilisés ce qui rend plus fiables les valeurs aberrantes. Elle est déterminée grâce à la formule suivante :

$$X_{normalise} = \frac{X_i - Q_1}{Q_3 - Q_1}$$

La technique StandardScaler

Lorsque les données sont réparties de façon normale, cette technique est la mieux adaptée, car elle permet de recalculer chaque caractéristique afin que les données soient centrées autour de 0 avec un écart-type de 1.

$$X_{normalise} = \frac{X_i - means(X)}{stdev(X)}$$

5.2.6 Fonctionnement du traitement des données au sein de FoDoMuST

Après intégration des fonctionnalités de prétraitement et d'analyse des données au sein de FoDoMuST nous décrivons dans cette section le cheminement des traitements de données comme suit :
Tout d'abord les données sont chargées à travers Java sous format d'objet SimpleData,

ensuite elles sont transformées en DataObject. S'il est nécessaire de faire un prétraitement ou une simple visualisation des des données et/ou de leurs statistiques , alors elles transformées en DataFrame sous Python à travers un mapping l'aide de Jep. Après le prétraitement les données sont retransformées en DataObject avec Jep pour être utilisées par les algorithmes de clustering (Kmeans par exemple comme indiqué sur la figure ci-dessous), et les résultats obtenus sont mappés de nouveau à l'aide de Jep en DataFrame sous Python pour une visualisation de bonne qualité avec PySimpleGUI et Malplotlib ou Seaborn. Sur cette figure ce sont les flèches en vert qui illustrent le cheminement des traitements effectués.

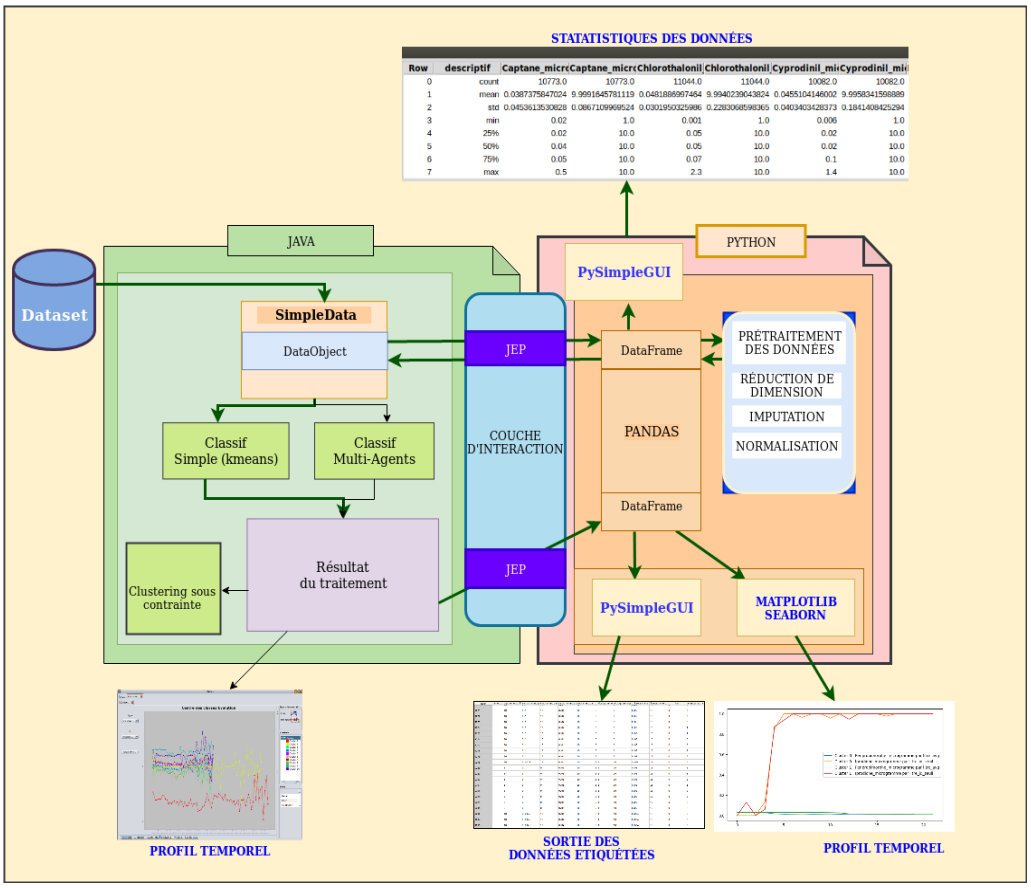


FIGURE 5.3 – Processus du cheminement des traitements des données

5.3 Expérimentations avec quelques jeux de données

Nous présentons dans cette section l'interface graphique de la plateforme d'analyse FoDoMuST avec les fonctionnalités d'analyse de données intégrées et les profils tomporels issus du traitement d'analyse de données temporelles.

5.3.1 Interface MultICube

Comme présenté sur la figure ci-dessous, à partir de l'onglet **preprocessing** nous aboutissons à 3 sous onglets que sont les onglets **Exclude** qui regroupe un ensemble de fonctionnalités pour éliminer les attributs non pertinents pour l'analyse, **Imputation** qui est regroupe un ensemble de fonctionnalités pour remplacer les valeurs manquantes et enfin **Normalisation** qui regroupe un ensemble de fonctionnalités pour la mise à l'échelle des données, elle n'est active que si les données n'ont plus de valeurs manquantes.

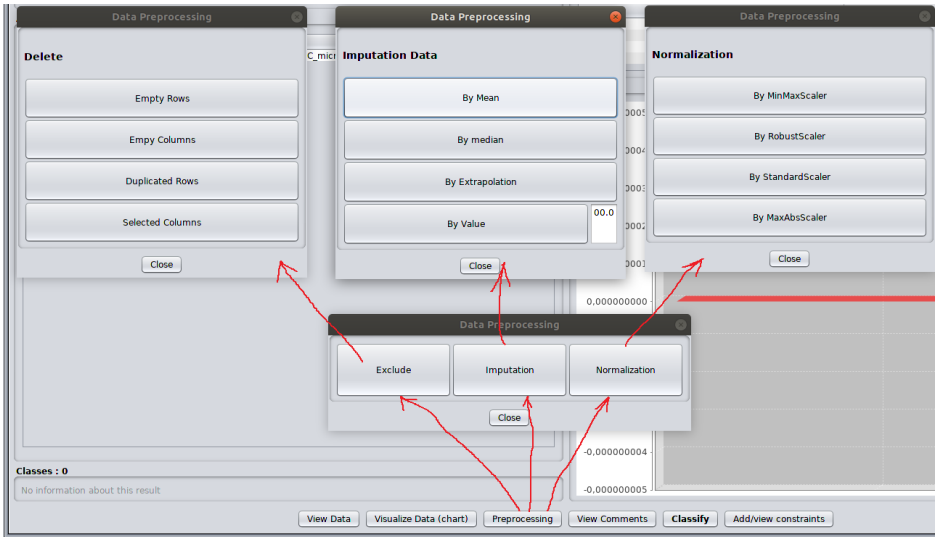


FIGURE 5.4 – Module preprocessing avec ses sous modules et leurs fonctionnalités

Cette interface met en relief les fonctionnalités de l'onglet Exclude en particulier sa fonctionnalité Selected Columns qui permet de sélectionner des attributs spécifiques à ignorer lors de l'analyse de données.

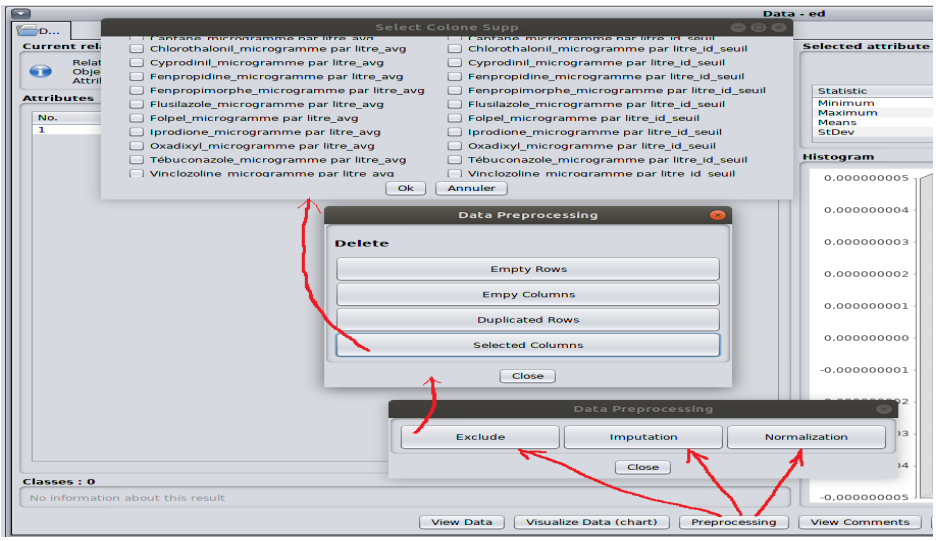


FIGURE 5.5 – Module Exclude avec ses fonctionnalités

Après une présentation des fonctionnalités intégrées au système d'analyse FoDoMuST nous déroulons les étapes et l'ordre des traitement à effectuer.

5.3.2 Étapes de l'expérimentation

Étape 1 : La réduction de dimension des données

Les premiers jeux de données étaient à plus de 80 attributs et à plus de 11.000 observations recueillies dans 304 stations à de différentes périodes. Ainsi, les experts en hydro-écologie ont procédé à une réduction de dimension qui est conduit à des jeux de données à des dimensions plus petites.

Étape 2 : Imputation des valeurs manquantes

Avant toute imputation une phase d'analyse de valeurs manquantes est faite afin de mieux choisir la méthode à utiliser pour l'imputation de ces valeurs manquantes. Les valeurs manquantes ont été remplacées dans le jeu de données **FONG_prio..15_18.csv** par la méthode de l'interpolation temporelle linéaire qui consiste à remplacer la ou les valeur(s) manquantes d'un attributs par la moyenne des deux valeurs immédiatement proches de celle-ci. Exemple : En considérant X la valeur manquante à la date T_i, celle-ci sera remplacée par la valeur $\frac{(T_i-1)+(T_i+1))}{2}$. Cependant si la variable manque se trouve en début ou en fin du jeu de données alors une interpolation inverse est mise en jeu, c'est à dire une considération d'un parcours dans le sens inverse est pris en compte pour l'estimation de cette valeur manque.

Étape 3 : Normalisation

Après la réduction de la dimensionnalité et l'imputation des données manquantes, l'objectif n'étant pas de perdre l'effet des valeurs extremums alors nous procédons à la normalisation par la méthode de MinMax.
La figure ci-dessous représente les données normalisées.

Row	Id	Capt	Chlorothalonil_micro	Chlo	Cyprodinil_microgr	Cypr	Fenp	Fenp	Fenpropimorphe_mi
0	402123.0	1.0	0.010869565217391306	1.0	0.0714285714285714	1.0	0.5	1.0	0.038461538461538464
1	402123.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
2	402123.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
3	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
4	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
5	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
6	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
7	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
8	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
9	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
10	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
11	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
12	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
13	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
14	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
15	402125.0	1.0	0.010869565217391306	1.0	0.0714285714285714	1.0	0.5	1.0	0.038461538461538464
16	402125.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
17	402125.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
18	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
19	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
20	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
21	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
22	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
23	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384

FIGURE 5.6 – Normalisation du jeu de données FONG_prio_her_v2_4_5_10_15_18.csv avec la méthode MinMax avec la phase d'imputation par interpolation temporelle linéaire.

Étape 4 : Courbe de coude

A partir des données prétraitées, nous construisons le graphique de la fonction de de coude pour estimer au mieux le nombre de clusters à former.

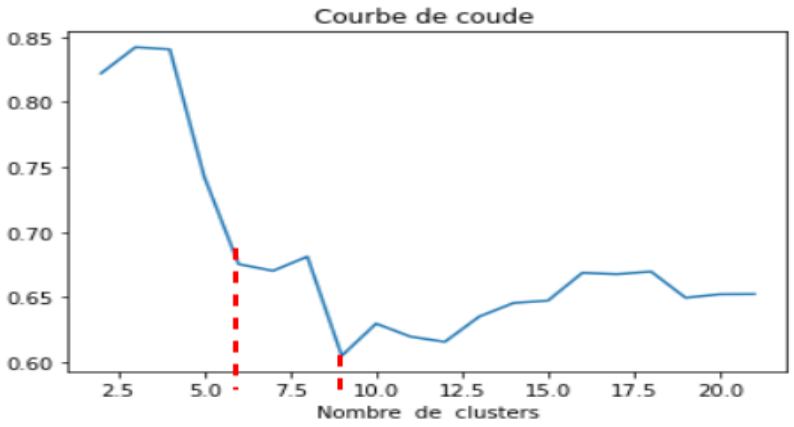


FIGURE 5.7 – Nombre de Cluster à former : Comme marquées en rouge sur la figure les valeurs approximatives 6 et 9 sont les mieux représentatifs en terme de nombre de clusters bien distingué à construire. Ainsi nous choisissons le nombre 9 pour l'expérimentation.

La figure ci-dessous représente les données avec leur groupe d'appartenance après le clustering.

Index											cluster_id
5271		10	0.1	10	0.04	10	0	0	0.05	10	0	3
5272		10	0.1	10	0.04	10	0	0	0.05	10	0	3
5273		10	0.1	10	0.04	10	0	0	0.05	10	0	0
5274		10	0.1	10	0.04	10	0	0	0.05	10	0	1
5275		10	0.1	10	0.04	10	0	0	0.05	10	0	1
5276		10	0.1	10	0.04	10	0	0	0.05	10	0	3
5277		10	0.1	10	0.04	10	0	0	0.05	10	0	3
5278		10	0.1	10	0.04	10	0	0	0.05	10	0	2
5279		10	0.1	10	0.04	10	0	0	0.05	10	0	3
5293		10	0.0025	10	0.04	10	0.1	10	0.004	10	0	3
5280		0	0	0	0.05	10	0.1	10	0.05	10	0	2
5281		0	0	0	0.05	10	0.1	10	0.05	10	0	1
5282		0	0	0	0.05	10	0.1	10	0.05	10	0	2
5283		0	0	0	0.05	10	0.1	10	0.05	10	0	1
5284		0	0	0	0.05	10	0.1	10	0.05	10	0	2
5288		0	0	0	0.05	10	0.1	10	0.05	10	0	0
5289		0	0	0	0.05	10	0.1	10	0.05	10	0	2
5290		0	0	0	0.05	10	0.1	10	0.05	10	0	2
5292		10	0.0025	10	0.04	10	0.1	10	0.004	10	0	0
5294		10	0.0025	10	0.04	10	0.1	10	0.004	10	0	2
5295		10	0.0025	10	0.04	10	0.1	10	0.004	10	0	0

FIGURE 5.8 – Données étiquetées (cluster_id) à la dernière colonne.

À partir des clusters obtenus ci-dessus nous construisons les profils temporels de chaque cluster en fonction du ou des attributs choisis.

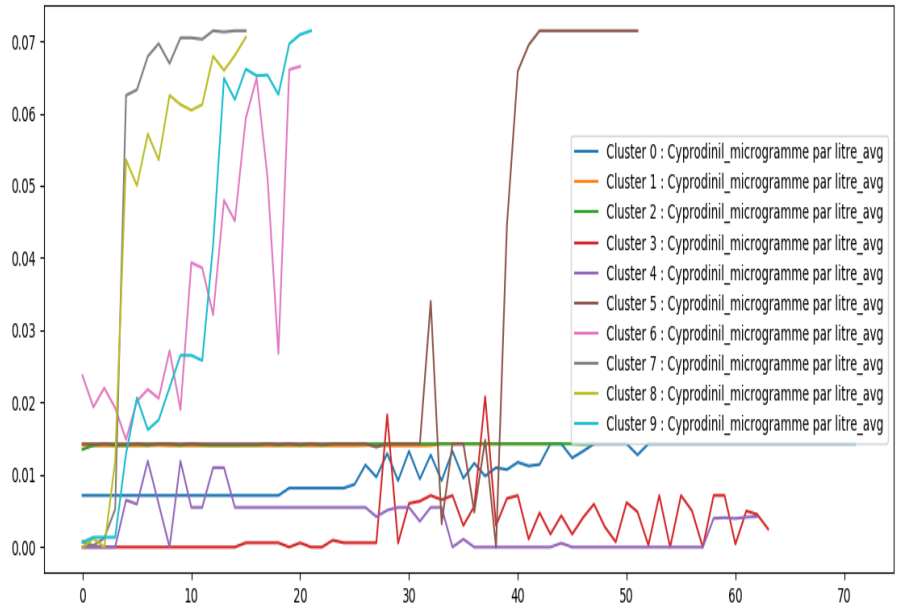


FIGURE 5.9 – Profil temporel de chaque cluster en fonction de l'attribut Cyprodinil_microgramme par litre_avg sans seuil. Cette représentation permet la mise en évidence de l'évolution de l'attribut Cyprodinil_microgramme par litre_avg dans chaque cluster.

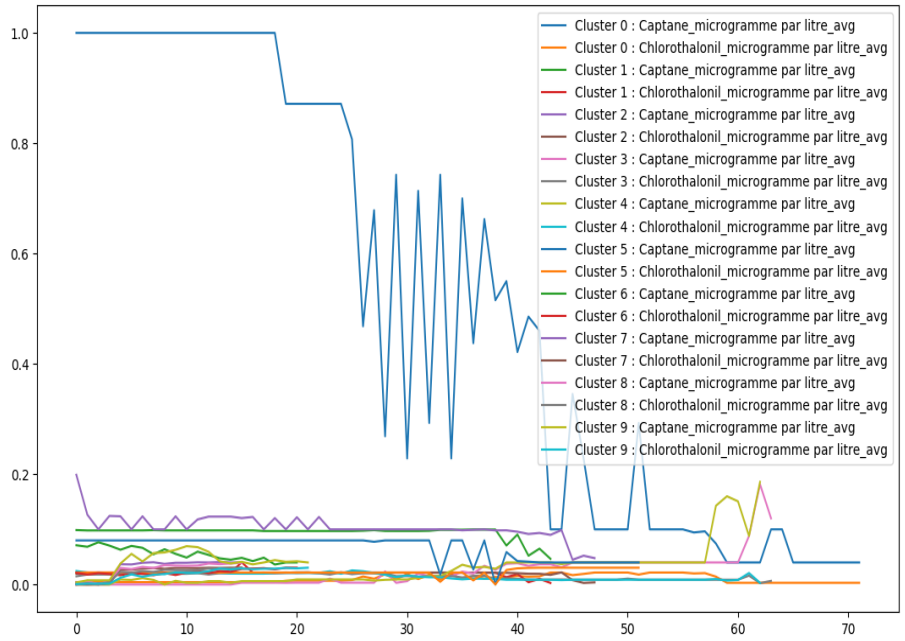


FIGURE 5.10 – Profil temporel de chaque cluster en fonction de l'attribut Captane_microgramme par litre_avg sans seuil.

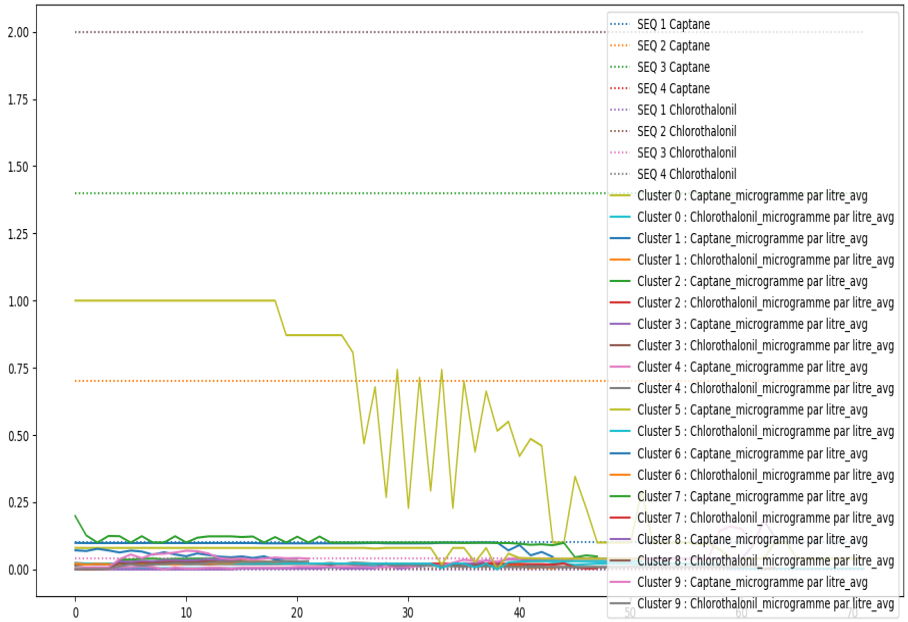


FIGURE 5.11 – Profil temporel de chaque cluster en fonction de l'attribut Captane_microgramme par litre_avg et chlothalonil_microgramme par litre_avg avec seuil.

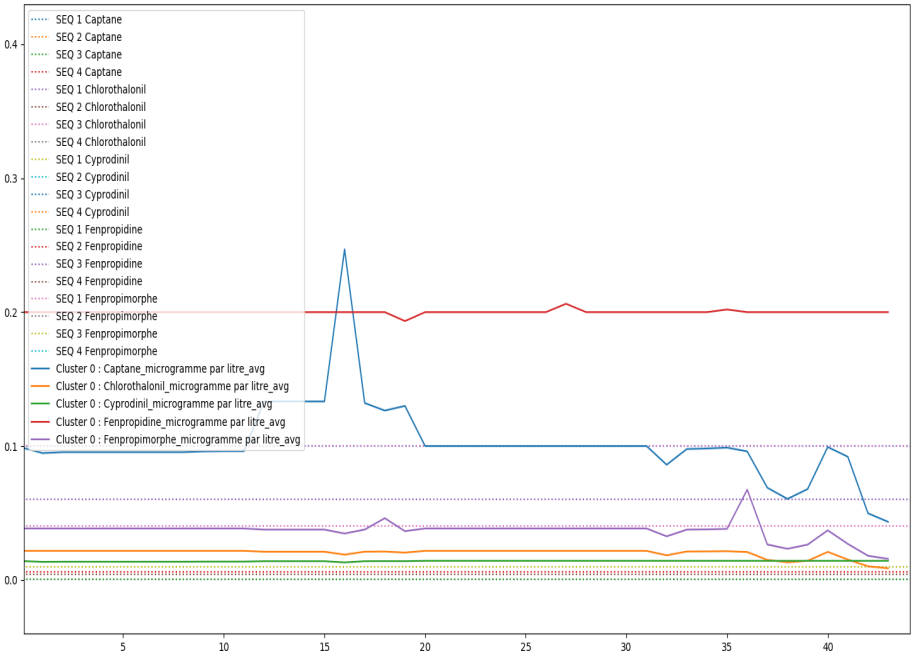


FIGURE 5.12 – Profil temporel du cluster 0 en fonction de 4 attributs du jeu de données avec seuil.

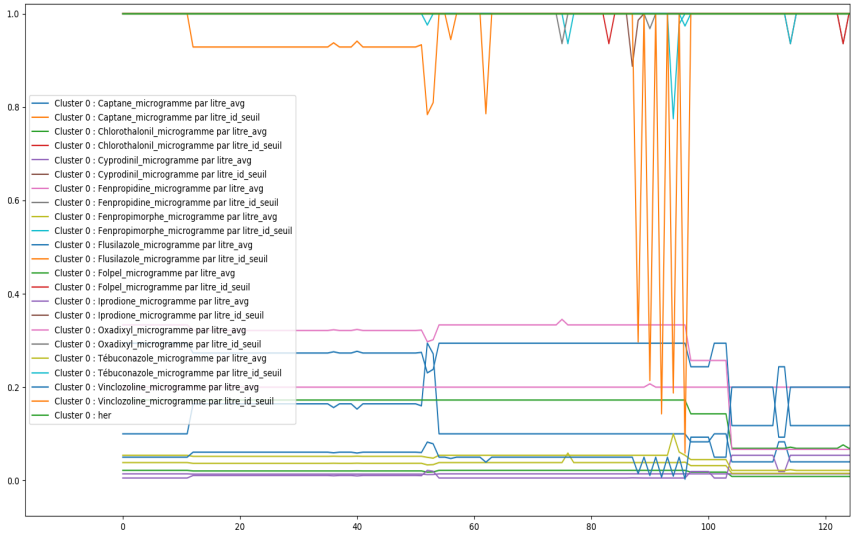


FIGURE 5.13 – Profil temporel du cluster 0 en fonction de tous les attributs du jeu de données sans seuil.

Chapitre 6

Intégration à FoDoMuST de fonctions externes pour l'analyse de séries

Nous présentons dans cette section une autre approche du traitement des séries temporelles autre que celle présenter précédemment. Cette solution est indépendante de la librairie JCL, qui est une librairie composée de classifieurs développées par l'équipe SDC. Elle est basée sur des algorithmes d'extraction et d'apprentissage en python que nous détaillons dans les sections suivantes. Cette solution a pour avantage principal, sa simplicité et sa rapidité et sa robustesse. Elle permet à l'aide de Scikit-learn, Tslearn de faire le clustering en utilisation KMeans. Chacun de ces deux frameworks Scikit-learn, Tslearn ont été utilisés pour plusieurs raisons.

6.1 Scikit-learn

En matière de frameworks de machine learning en python, Scikit-learn est devenu une référence mondiale, confirme Bouzid Ait Amir [14], team lead data science au sein du cabinet français Keyrus. Scikit-learn nous offre les avantages suivants.

- Il dispose d'une excellente documentation fournissant de nombreux exemples. Cette documentation riche de Scikit-learn permet à ses utilisateurs de comprendre les différents packages pour vite prendre la main pour leur utilisation.
- Il dispose d'une API uniforme entre tous les algorithmes, ce qui fait qu'il est facile de basculer de l'un à l'autre. Il a une construction qui permet à ses utilisateurs de passer facilement d'un algorithme à un autre sans assez de difficultés.
- Il est très bien intégrée avec les Librairies Pandas, Matplotlib et Seaborn pour la manipulation des données et la visualisation.

• Scikit-learn dispose d'une grande communauté et de plus de 800 contributeurs référencés sur GitHub, cette riche communauté permet à tout utilisateur de traiter des problèmes complexes en s'inspirant des exemples déjà abordés par cette communauté.
• C'est un projet open source et toute personne peut contribuer à son amélioration.
• Son code est rapide, certaines parties sont implémentées en Cpython.
• Il facilite la lecture et la compréhension de votre flux de travail.

Il dispose des librairies plus optimisées ou spécialisées dans l'apprentissage automatique. Cependant avant l'utilisation de Scikit-learn les données sont lues par Pandas. Le DataFrame obtenu est transféré à Scikit-learn où il subit un prétraitement à l'aide de **sklearn_preprocessing** et ensuite il est transformé en un tableau de données compréhensible par Tslearn.

6.2 Tslearn : Time series learning

TSLEARN est un package python dont l'objectif est de permettre l'apprentissage automatique sur les séries temporelles [16]. Ses principaux avantages sont :

• La rapidité, en effet contrairement aux autres librairies Tslearn a une mémoire cache qui lui permet de charger les données une et une seule fois. La conservation des données dans sa mémoire cache permet d'éviter les appels répétés à chaque action nécessitant l'utilisation des données.
• L'efficacité de cette librairie est tirée à partir de cette rapidité.
• Le traitement des séries temporelles à l'aide de tslearn.preprocessing.
• Tslearn est construit pour les données temporelles, ainsi il est bien spécifique pour le traitement de ce domaine.

Les données reçues sous forme de tableau par Tslearn sont transformées conformément au fonctionnement de Tslearn grâce à **to_time_series_dataset()** et ensuite **TimeSeriesKMeans()** est utilié pour le clustering. Après le clustering l'affichage des graphiques est fait par MatplotLib.

6.2.1 Architecture globale

La figure suivante illustre le cheminement du jeu de données jusqu'à la sortie des résultats après traitement. Le jeu de données est lu sous de dataframes par pandans. Ensuite ces données sont envoyées à scikit-learn où elles peuvent subir des opérations de prétraitement si nécessaire avant d'être transformées sous formes de matrices pour

être compréhensible par les algorithmes. Après cette étape les données sont transférées à tslearn pour le clustering avec une distance choisie. A la fin du clustering la visualisation des résultats est faite par matplotlib ou seaborn.

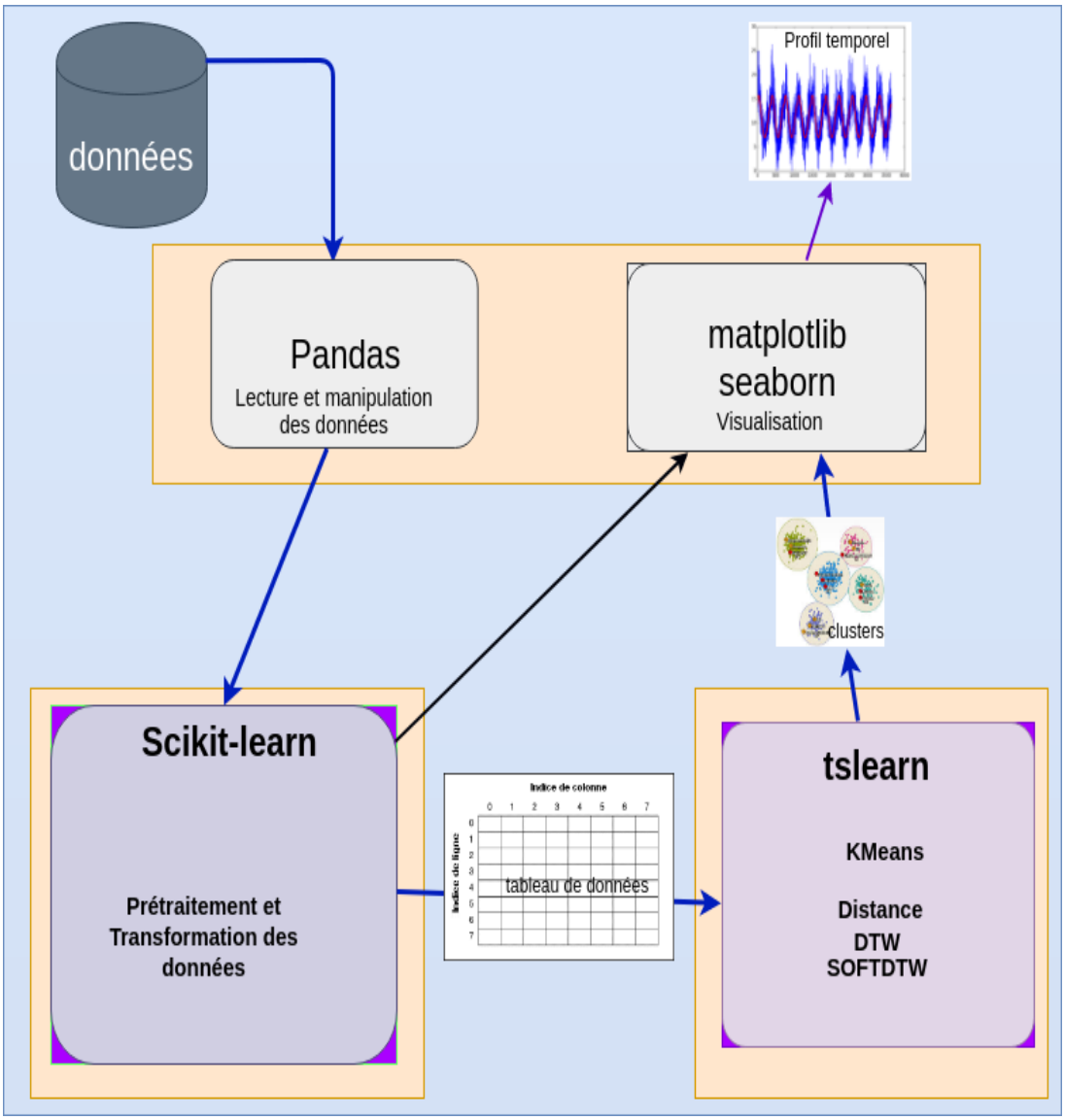


FIGURE 6.1 – Architecture de la solution

Conclusions et perspectives

Dans le cadre de notre stage, nous avons étudié des techniques et des approches qui convergent autour de l'analyse des séries temporelles massives. Nous avons suivi trois grandes étapes dans la réalisation de ce projet : l'étude et l'analyse du projet, l'étude des techniques et des approches en relation avec l'analyse des séries temporelles et enfin l'implémentation des techniques les plus adaptées et expérimentations.

Tout d'abord nous avons fait l'étude et l'analyse du projet et une exploration de l'outil d'analyse FoDoMuST développée par l'équipe SDC. Cette exploration de l'existant, l'étude des travaux connexes et les différentes rencontres avec les membres des deux laboratoires m'ont permis de mieux comprendre les objectifs du projet.

Les solutions proposées et acceptées par les deux laboratoires ont été implémentées et intégrées au sein de FoDoMuST. L'implémentation de ces solutions nous a permis de fournir des résultats (génération automatique de jeux de données étiquetées, génération de profil temporel avec et sans seuil, visualisation de la répartition des clusters ...) , qui ont fait l'objet d'une étude d'analyse et d'interprétation par les experts.

Les résultats obtenus seront exploités pour la deuxième phase du projet qui sera une prédiction de la qualité de l'eau en fonction des caractéristiques physico-biologiques. Les données étant en perpétuelles augmentation, nous proposons une étude avec les méthodes de **clustering sous contraintes interactives**. Étant donné que les techniques de deep learning sont mieux adaptées aux grands volumes de données, nous pensons également qu'il est nécessaire de stocker les séries temporelles sur des **Time series Data Bases** (InfluxDB, Kdb+, Graphite, Prometheus, RRDTool, OpenTSDB, TimescaleDB, KairosDB, eXtremeDB) qui permettent une meilleure gestion de ce type de données que les gestionnaires de bases de données classiques comme MySQL, Postgres etc.

Bibliographie

[1] Eva Carmina Serrano Balderas. *Preprocessing and analysis of environmental data : Application to the water quality assessment of Mexican rivers*. PhD thesis, Université de Montpellier, 31 Janvier 2017.

[2] Filippo Maria Bianchi, Simone Scardapane, Sigurd Løkse, and Robert Jenssen. Reservoir computing approaches for representation and classification of multivariate time series. *arXiv preprint arXiv :1803.07870*, 2018.

[3] Franck Boizard. Méthodes d'analyse de sensibilité de modèles pour entrées climatiques. *Mémoire de stage d'Agrocampus Ouest, INRA : Toulouse*, 2015.

[4] Gustavo Camps-Valls, Devis Tuia, Lorenzo Bruzzone, and Jon Atli Benediktsson. Advances in hyperspectral image classification : Earth monitoring with statistical learning methods. *IEEE signal processing magazine*, 31(1) :45–54, 2013.

[5] Maximilian Christ, Andreas W. Kempa-Liehr, and Michael Feindt. Distributed and parallel time series feature extraction for industrial big data applications. *CoRR*, abs/1610.07717, 2016.

[6] Michael Hahsler, Matthew Piekenbrock, Derek Doran, Michael Hahsler, Michael Hahsler, Kurt Hornik, Michael Hahsler, Kurt Hornik, Michael Hahsler, Kurt Hornik, et al. dbscan : Fast density-based clustering with r. *Journal of Statistical Software*, 25 :409–416.

[7] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7) :1527–1554, 2006.

[8] Seibi Chiba Hiroaki Sakoe. Dynamic programming algorithm optimization for spoken word recognition. in : Ieee trans. on acoustics, speech, and sig. proc, 1978.

[9] Mathieu Blondel Marco Cuturi. oft-dtw : a differentiable loss function for time-series. in : Proc. of icml. 2017.

[10] Marion Morel, Catherine Achard, Richard Kulpa, and Séverine Dubuisson. Time-series averaging using constrained dynamic time warping with tolerance. *Pattern Recognition*, 74, 08 2017.

[11] Dimitry Fisher Homa Karimabad Naveen Sai Madiraju, Seid M. Sadat. Deeptemporalclustering :fully unsuper-vised learning of time-domain features. 4 Feb 2018.

[12] François Petitjean. Description des alignements formés par dtw. 2011.

[13] SDC_ICube. Fodomust. http ://icube-sdc.unistra.fr/en/index.php/FODOMUST.

[14] SKLEARN. Sklearn. https://blog.floydhub.com/introduction-to-k-means-clustering-in-python-with-scikit-learn/. Accessed : 20 avril 2019.

[15] TSFRESH. Tsfresh. https://tsfresh.readthedocs.io/en/latest/. Accessed : 15 juin 2019.

[16] TSLEARN. Tslearn. https://tslearn.readthedocs.io/en/latest/index.html#. Accessed : 15 juin 2019.

6.3 Annexe

Les jeux de données traités

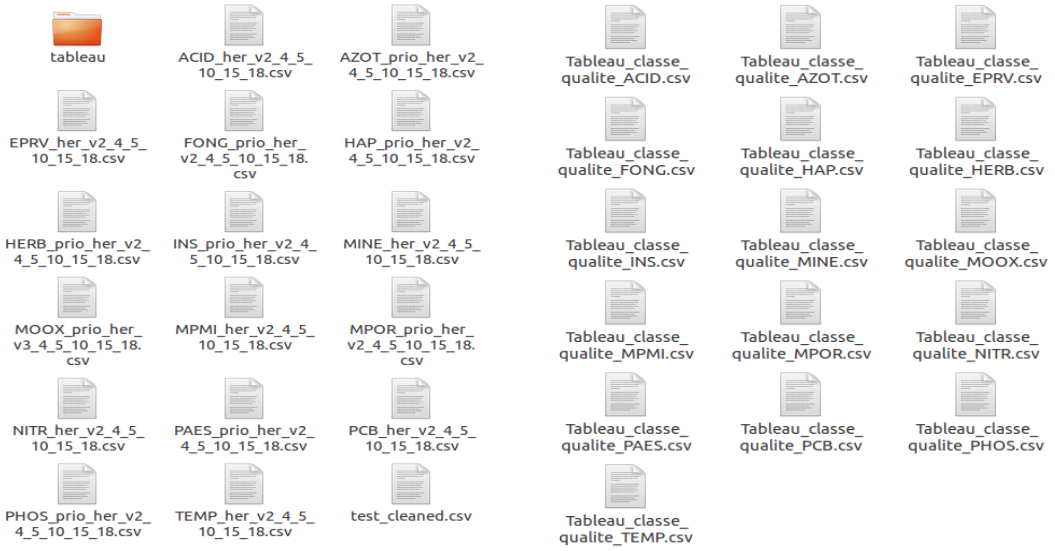


FIGURE 6.2 – Jeux de données

Différents affichages des données

Row	Id	Captane_mi	Captane_micr	Chlorothalonil	Chlorothalonil	Cyprodinil_mi	Cyprodinil_mi	Fenpropidine_	Fenpropidine_	Fenpropimorp
0	402123.0	0.05	10.0	0.025	10.0	0.1	10.0	0.05	10.0	0.05
1	402123.0	0.05	10.0	0.07	10.0	0.1	10.0	0.05	10.0	0.02
2	402123.0	0.05	10.0	0.07	10.0	0.1	10.0	0.05	10.0	0.02
3	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
4	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
5	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
6	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
7	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
8	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
9	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
10	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
11	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
12	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
13	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
14	402123.0	0.02	10.0	0.07	10.0	nan	nan	nan	nan	0.02
15	402125.0	0.05	10.0	0.025	10.0	0.1	10.0	0.05	10.0	0.05
16	402125.0	0.05	10.0	0.07	10.0	0.1	10.0	0.05	10.0	0.02
17	402125.0	0.05	10.0	0.07	10.0	0.1	10.0	0.05	10.0	0.02
18	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02
19	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02
20	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02
21	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02
22	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02
23	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02
24	402125.0	0.02	10.0	0.07	10.0	nan	nan	0.02	10.0	0.02

FIGURE 6.3 – Affichage des données

Row	descriptif	Captane_micrc	Captane_micrc	Chlorothalonil	Chlorothalonil	Cyprodinil_mi	Cyprodinil_mi
0	count	10773.0	10773.0	11044.0	11044.0	10082.0	10082.0
1	mean	0.0387375847024	9.9991645781119	0.0481886997464	9.9940239043824	0.0455104146002	9.9958341598889
2	std	0.0453613530828	0.0867109969524	0.0301950325986	0.2283068598365	0.0403403428373	0.1841408425294
3	min	0.02	1.0	0.001	1.0	0.006	1.0
4	25%	0.02	10.0	0.05	10.0	0.02	10.0
5	50%	0.04	10.0	0.05	10.0	0.02	10.0
6	75%	0.05	10.0	0.07	10.0	0.1	10.0
7	max	0.5	10.0	2.3	10.0	1.4	10.0

FIGURE 6.4 – Affichage des statistiques des données

Row	{C	{Ca	{Captane_micrograi	{Captane_n	{Ca	{Captane_micrc	{Captane_mi	{Captane_mi	{Chlorothalonil
0	0.	0.05	0.026000000000000006	0.02	10.0	10.0	10.0	10.0	0.025
1	0.	0.05	0.02290322580645162	0.02	10.0	10.0	10.0	10.0	0.025
2	0.	0.05	0.02290322580645162	0.02	10.0	10.0	10.0	10.0	0.025
3	0.	0.05	0.044374999999999984	0.05	10.0	10.0	10.0	10.0	0.02
4	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
5	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
6	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
7	0.	0.05	0.044651162790697516	0.05	10.0	10.0	10.0	10.0	0.02
8	0.	0.05	0.044651162790697516	0.05	10.0	10.0	10.0	10.0	0.02
9	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
10	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
11	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
12	0.	0.05	0.043125000000000024	0.05	10.0	10.0	10.0	10.0	0.02
13	0.	0.05	0.043125000000000024	0.05	10.0	10.0	10.0	10.0	0.02
14	0.	0.05	0.04232558139534886	0.05	10.0	10.0	10.0	10.0	0.02
15	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
16	0.	0.05	0.04386363636363639	0.05	10.0	10.0	10.0	10.0	0.02
17	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02
18	0.	0.5	0.1307499999999998	0.05	10.0	10.0	10.0	10.0	0.01
19	0.	0.05	0.043125000000000024	0.05	10.0	10.0	10.0	10.0	0.02
20	0.	0.05	0.043125000000000024	0.05	10.0	10.0	10.0	10.0	0.02
21	0.	0.05	0.042500000000000024	0.05	1.0	10.0	9.795454545454	10.0	0.02
22	0.	0.05	0.04232558139534886	0.05	10.0	10.0	10.0	10.0	0.02
23	0.	0.05	0.042500000000000024	0.05	10.0	10.0	10.0	10.0	0.02

FIGURE 6.5 – Affichage des statistiques des données par station

[a]

Table

Row	index	0
0	Captane_microgr	1028
1	Captane_microgr	1028
2	Chlorothalonil_mi	757
3	Chlorothalonil_mi	757
4	Cyprodinil_microg	1719
5	Cyprodinil_microg	1719
6	Fenpropidine_mic	3126
7	Fenpropidine_mic	3126
8	Fenpropimorphe_	337
9	Fenpropimorphe_	337
10	Flusilazole_micro	846
11	Flusilazole_micro	846
12	Folpel_microgram	337
13	Folpel_microgram	337
14	Iprodione_microgr	1712
15	Iprodione_microgr	1712
16	Oxadixyl_microgr	2
17	Oxadixyl_microgr	2

[b]

Table

Row	Id	Missing value
0	402123.0	0.0
1	402123.0	0.0
2	402123.0	0.0
3	402123.0	43.478260869565
4	402123.0	43.478260869565
5	402123.0	43.478260869565
6	402123.0	43.478260869565
7	402123.0	43.478260869565
8	402123.0	43.478260869565
9	402123.0	43.478260869565
10	402123.0	43.478260869565
11	402123.0	43.478260869565
12	402123.0	43.478260869565
13	402123.0	43.478260869565
14	402123.0	43.478260869565
15	402125.0	0.0
16	402125.0	0.0
17	402125.0	0.0
18	402125.0	17.391304347826
19	402125.0	17.391304347826
20	402125.0	17.391304347826
21	402125.0	17.391304347826
22	402125.0	17.391304347826
23	402125.0	17.391304347826
24	402125.0	17.391304347826

FIGURE 6.6 – Nombre de valeurs manquantes par colonne en [a] et Pourcentage des valeurs manquantes par ligne en [b]

Row	Id	Capt	Chlorothalonil_micro	Chlo	Cyprodinil_microgr	Cypr	Fenp	Fenp	Fenpropimorphe_mi
0	402123.0	1.0	0.010869565217391306	1.0	0.0714285714285714	1.0	0.5	1.0	0.038461538461538464
1	402123.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
2	402123.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
3	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
4	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
5	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
6	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
7	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
8	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
9	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
10	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
11	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
12	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
13	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
14	402123.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.0	0.0	0.015384615384615384
15	402125.0	1.0	0.010869565217391306	1.0	0.0714285714285714	1.0	0.5	1.0	0.038461538461538464
16	402125.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
17	402125.0	1.0	0.030434782608695657	1.0	0.0714285714285714	1.0	0.5	1.0	0.015384615384615384
18	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
19	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
20	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
21	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
22	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384
23	402125.0	1.0	0.030434782608695657	1.0	0.0	0.0	0.2	1.0	0.015384615384615384

FIGURE 6.7 – Test du fichier FONG_prio_her_v2_4_5_10_15_18.csv
Les valeurs manquantes sont remplacées par 0 et après ce remplacement, une normalisation de type MinMax est faite sur toutes les variables hormis l'identifiant (Id) du fichier FONG_prio_her_v2_4_5_10_15_18.csv.

Visualisation des profils temporels complets

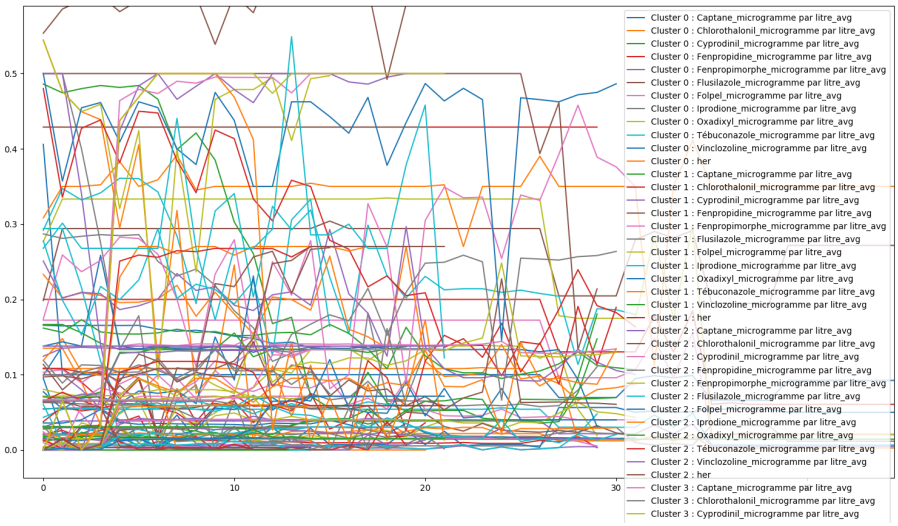


FIGURE 6.8 – Profil temporel de tous les attributs sans seuils