Une méthode de classification basée sur une stratégie de raffinement des données d'échantillon

A Classification method based on a refining strategy of the example data

Rosanna Verde[1]
in collaboration avec Mohammed Sabri et Antonio Balzanella

Dep. of Mathematics and Physics, UDCLV

CNAM Paris 25 octobre 2024

Motivation : L'impact des motifs cachés sur la performance des classificateurs

Patterns cachés sont des structures complexes et non évidentes au sein des données que les méthodes d'analyse standard laissent souvent de côté. Ces structures représentent des relations ou des phénomènes importants, dont la détection nécessite des techniques avancées, et peuvent fournir des informations cruciales qui remettent en question les hypothèses et améliorent notre compréhension du système.

Défis posés par les modèles cachés :

1. Précision : Peut réduire les performances de la classification
2. Complexité : Nécessite souvent des modèles plus sophistiqués
3. Sélection des caractéristiques : Rend plus difficile l'identification des variables pertinentes
4. Surparamétrage : Risque d'assimiler le bruit à des modèles

L'étude des modèles cachés dans les données est cruciale pour faire progresser l'apprentissage automatique et la science des données.
Améliore la généralisation des modèles, renforce l'interprétabilité et rendre les systèmes de classification plus robustes.

Introduction

Exemples concrets

- Patient subgroups
- Treatment responses
- Disease progression

- Fraud strategies
- Transaction anomalies
- Coordinated behavior

- Fine-grained classes
- Adversarial examples
- Early-stage diseases

Introduction

Exemples concrets

Un résumé concis de chaque exemple de modèle caché dans le monde réel :

- Santé : Des modèles cachés peuvent révéler des sous-groupes de patients ayant des réactions uniques au traitement, ce qui pourrait conduire à des approches de médecine personnalisée. Ces modèles peuvent indiquer que l'efficacité des médicaments varie d'un groupe à l'autre ou prédire la progression de la maladie, ce qui permet des traitements plus ciblés et plus efficaces.
- Finances : Dans le secteur financier, les modèles cachés sont souvent liés à l'évolution des activités frauduleuses. Ces modèles peuvent révéler de nouvelles stratégies de fraude, mettre en évidence des anomalies contextuelles dans les transactions et découvrir des réseaux de comportements frauduleux coordonnés. La détection de ces modèles est cruciale pour le maintien de la sécurité des systèmes financiers.
- Classification d'images : Des caractéristiques subtiles et non évidentes peuvent s'avérer essentielles pour distinguer des classes similaires dans les tâches de classification d'images. Ces modèles cachés peuvent être essentiels pour une classification fine (par exemple, l'identification d'espèces spécifiques), la compréhension d'exemples contradictoires, la résolution de problèmes d'apprentissage par transfert ou la détection de maladies à un stade précoce dans le domaine de l'imagerie médicale.

Plan de la proposition

- Motivation
- intégration de techniques d'apprentissage supervisé et non supervisé dans la construtution d'un classificateur
- Partition de classes à priori
- Utilisation d'une nouvelle fonction objective
- Une méthode innovante de classification des DF :
  Local Mean k-NN
- Application à la detection des fraudes (K-FUSE)
- Application à la classification des données fonctionnelles (DF)
- Validation de la stratégie proposée
- Conclusions

Introduction

Objectifs de la recherche et innovations

Nous nous concentrons sur le développement et l'amélioration des méthodes de classification supervisée en intégrant des techniques d'apprentissage non supervisé afin de découvrir et d'utiliser des modèles cachés dans les données.
Notre principale contribution concerne les défis posés par la détection des fraudes et la classification de l'analyse fonctionnelle des données.

Questions clés de recherche :

- Comment pouvons-nous découvrir et utiliser efficacement les modèles cachés dans les ètapes de classification ?
- Comment améliorer la détection des fraudes en relation à l'évolution de la structure des modèles ?
- Comment étendre cette approche à la classification des données fonctionnelles ?
- Quels sont les fondements théoriques du nouvelle approche d'apprentissage intégré ?

Innovations méthodologiques :

- **Clustering Dynamique (DC)** avec Distances Adaptatives
- Découvre des modèles (patterns) complexes
- Integration de DC et **KNN**
- Améliore la précision de la classification
- Nouvelle fonction objective de clustering

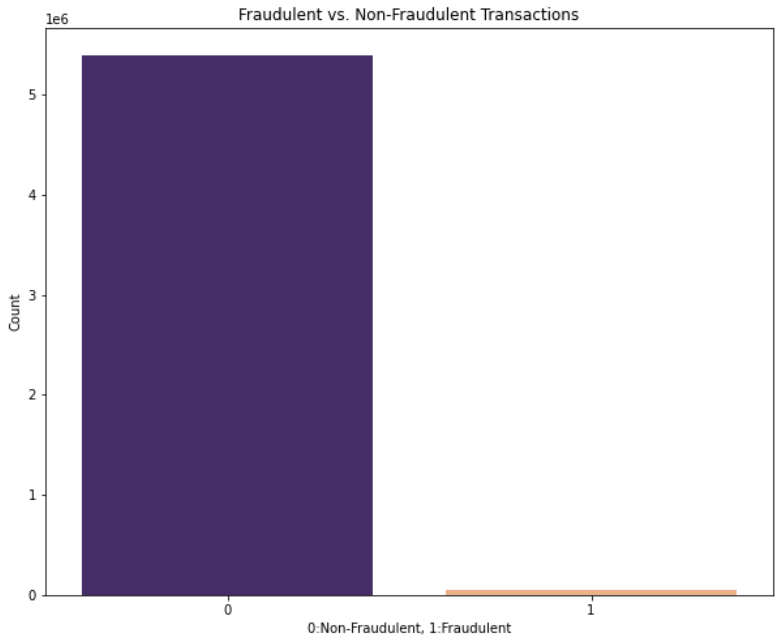
Défis en matière de détection de la fraude

- Le défi dans la détection des fraudes provient du déséquilibre entre la petite fraction de cas de fraude (généralement inférieure à 1 %) et la majorité des transactions authentiques. Ce **déséquilibre des données entrave les méthodes conventionnelles de détection des fraudes, conduisant à une détection inexacte**.
- Les fraudeurs s'adaptent et utilisent des méthodes sophistiquées pour **emuler un comportement légitime**, ce qui rend **difficile pour les modèles de détection de faire la distinction entre les transactions authentiques et frauduleuses**, ce qui entraîne un taux accru de faux négatifs.
- **Le comportement des clients est susceptible de changer** en raison de divers facteurs tels que les conditions économiques, les tendances ou les circonstances personnelles.
- Les fraudeurs peuvent tenter activement de manipuler le modèle de détection des cartes de crédit en étudiant ses faiblesses et en trouvant des moyens de le tromper. Les attaques adverses consistent à exploiter les vulnérabilités des algorithmes du modèle, à le tromper et à **misclassifier des transactions frauduleuses comme étant légitimes.**

Gestion des classes déséquilibrées

La plupart des recherches ne se concentrent généralement pas sur les données déséquilibrées, même si un **ensemble de données déséquilibré peut produire des résultats biaisés en matière d'apprentissage automatique**.

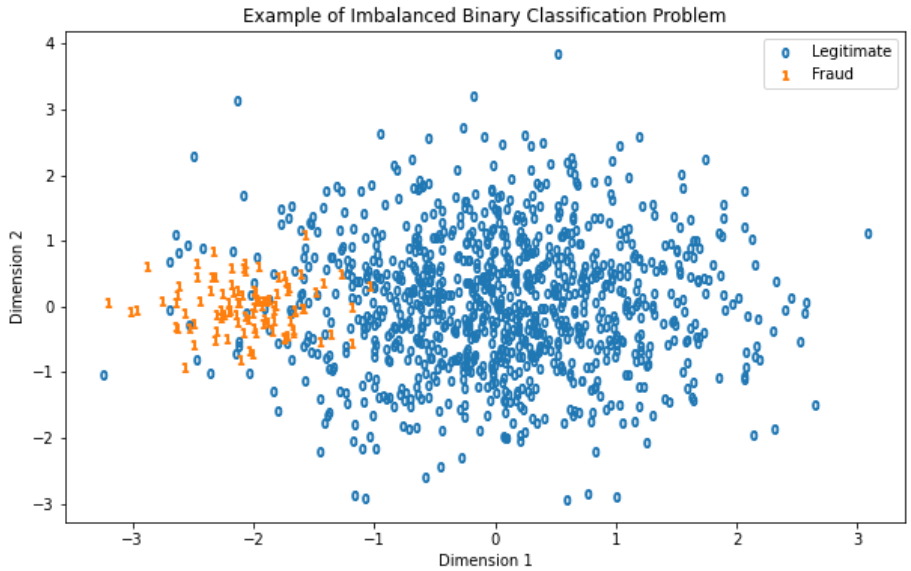
Toutefois, l'objectif est d'optimiser les performances sur l'ensemble des données d'apprendissage équilibrées.



Les stratégies visant à résoudre les problèmes de déséquilibre des données sont des techniques bien établies, notamment de sur-échantillonnage ou de sous-échantillonnage. Ces méthodologies sont généralement mises en œuvre avant la formation du modèle d'apprentissage automatique.
Notre idée est d'utiliser une technique **sous-échantillonnage** pour équilibrer l'ensemble d'apprendissage.

Stratégie de découverte de nouveaux modèles de fraude

Différents modèles peuvent souvent caractériser certains types de fraude au sein d'une classe (étiquette), de sorte que l'algorithme de classification peut se révéler imprécis.



Notre principale contribution consiste à **exploiter** une méthode de **regroupement non supervisé** basée sur un nouveau critère de partitionnement/discrimination et sur une sélection appropriée des caractéristiques **pour améliorer les performances d'un classificateur** en découvrant **de nouveaux modèles de fraude dans les groupes d'origine** avant d'entraîner l'algorithme du classificateur.

Stratégie de détection de fraudes : K-Fuse

K-Fuse, intégrant quatre techniques :

1. algorithme de clustering dynamique, pour partitioner les classes antérieures de l'ensemble d'apprentissage en sous-groupes homogènes, utilisant :
   - distances adaptives
   - une nouvelle *intra-classe interclasse* critère
2. une nouvelle stratégie de sélection des caractéristiques et une technique d'échantillonnage pour obtenir un ensemble de données équilibré
3. une version révisée de l'algorithme de classification KNN qui utilise les résultats du regroupement pour améliorer les performances de l'algorithme de classification
4. évaluation de la précision et résultats explicatifs

L'objectif est de construire un classificateur qui prenne en compte la structure en sous-groupes des classes a priori afin d'améliorer la précision du modèle de classification.

Reference : Maturo, F., Verde, R. Comput Stat 39, (2024)

Algorithme de clustering dynamique

La classification non supervisée traite du problème de l'identification des groupes d'éléments qui sont les plus similaires à l'intérieur de chaque groupe et les plus différents des éléments des autres groupes. Un schéma général de regroupement est donné par l'algorithme de clustering dynamique.

Contrairement à la classification supervisée, il n'y a **aucun échantillon de formation** pour servir de guide, et nous n'avons **aucune connaissance préalable du nombre de groupes**.

L'algorithme k-means est un cas particulier de l'algorithm DC et c'est la méthode de **clustering** la plus couramment utilisée dans l'apprentissage automatique.

Étant donné un ensemble d'objets $(x_1, \ldots, x_n)$, la classification par k-moyennes vise à répartir les n individus en $K \leq n$ classes $P = \{C_1, \ldots, C_K\}$ de manière à minimiser, comme critère, la somme des carrés à l'intérieur d'un groupe :

$$\underset{P}{\operatorname{argmin}} \sum_{i=1}^{K} \sum_{i \in C_k} d^2(x_i, z_k)$$

où z_k est la moyenne des éléments x_i de la classe C_k.

Importance de la Distance Adaptive

Le **choix des variables** peut avoir un impact significatif sur la qualité des résultats du regroupement.

L'importance des variables peut être déterminée en introduisant une pondération automatique des variables.

Cela se fait par l'utilisation de distances d'adaptation.

Idea

En utilisant **distance adaptive** un système de pondération approprié est déterminé sur les variables en fonction de leur variabilité et de leur influence dans la recherche de la meilleure partition en classes.

Les poids sont calculés dynamiquement à chaque itération de l'algorithme.
Dans ce contexte, la distance d_k est une somme pondérée des distances $d_{w_{kj}}$

$$d_k(X_i, Z_k) = \sum_{j=1}^{m} d_{w_{kj}}(x_{ij}, z_{kj}) = \sum_{j=1}^{m} w_{kj} d(x_{ij}, z_{kj})$$

L'adaptabilité de la distance $d_{w_{kj}}$ est exprimée par le vecteur de poids W_k.

Étapes de la classification dynamique

Le processus de l'algorithme de clustering dynamique si déroule en deux étapes : la phase de représentation et la phase d'attribution des données aux classes, en optimisant la fonction de critère :

$$\Delta(X_i, Z_k, W_k) = \sum_{j=1}^{m} \sum_{i \in C_k} w_{kj} d(x_{ij}, z_{kj})$$

Lors de l'utilisation de distances adaptatives, l'étape de représentation est divisée en deux phases, qui peuvent être exprimées comme suit :

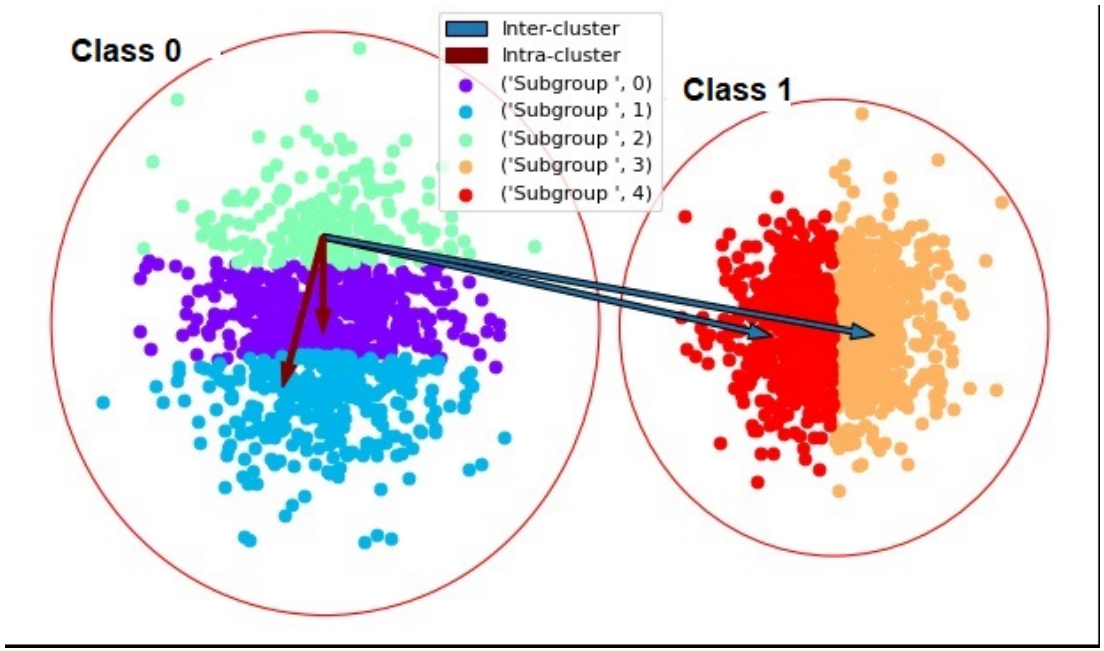
1. Étape de Representation

- **Étape 1 :** fixer la matrice $\hat{U}$ d'appartenance et le vecteur de poids $\hat{W}$
  Trouver la solution $Z_k = \{z_{k1}, \ldots, z_{km}\}$ du problème d'optimisation $\Delta(\hat{U}, \hat{W}, Z)$.
  0.1cm
- **Étape 2 :** fixés la matrice $\hat{U}$ d'appartenance et le vecteur des centroïdes $\hat{Z}$
  Trouve le vecteur de poids $W_k = \{w_{k1}, \ldots, w_{km}\}$ qui minimise le critère $\Delta(\hat{U}, W, \hat{Z})$.

2. Étape d'affectation (fixer l'ensemble des vecteurs de poids $\hat{W}$ et les vecteurs de centroïdes $\hat{Z}$) :
Trouver la matrice d'appartenance U qui correspond à l'ensemble des vecteurs de poids $\hat{W}$. Trouver la matrice d'appartenance U qui minimise le critère $\Delta(U, \hat{W}, \hat{Z})$

A nouvelle funtion objective

La **nouvelle fonction objective** vise à **minimiser les distances intra-groupes des sous-groupes des classes d'origine** (en utilisant une distance adaptative) et simultanément **maximiser la distance inter-groupes entre les sous-groupes des autres classes d'origine**.



Sous-groupes des classes d'origine

Une nouvelle fonction objective

La **nouvelle fonction objective** permet de minimiser les distances intra-groupes entre les sous-groupes de chaque classe d'origine *(numérateur)* et de maximiser la distance entre les centroids de sous-groupes d'une classe d'origine et le centroides globals des autres classes d'origine *(dénominateur)*.

$$P(U,W,Z) = \sum_{j=1}^{m} \left(\sum_{g=1}^{N} \sum_{p=1}^{c_g} \frac{w_{gpj}^{\beta} \sum_{i=1}^{n_g} u_{gip} (x_{ij} - z_{gpj})^2}{n_g (z_{gpj} - z_{gGj})^2} \right)$$

$$\text{s.t.} \quad u_{gip} \in \{0,1\}, \ \sum_{p=1}^{c_k} u_{gip} = 1 \quad \sum_{j=1}^{m} w_{gpj} = 1$$

où : u_{gip} sont les éléments d'une matrice U_g et représentent l'appartenance binaire de chaque élément de l'ensemble de données au groupe g (pour $= 1, \ldots, N$)

Solutions de fonctions objectives

- Initialiser les paramètres $\hat{U}$, $\hat{W}$, et $\hat{Z}$ de tous les groupes.
Ensuite, nous procédons aux étapes de partitionnement et d'allocation en minimisant l'équation suivante :

$$P(U,W,Z) = \sum_{p=1}^{c_k} \sum_{i=1}^{n_k} u_{kip} \sum_{j=1}^{m} w_{pj}^2 \frac{(x_{ij} - z_{kpj})^2}{n_k (z_{kpj} - z_{kGj})^2}$$

$$\text{s.t.} \quad u_{ip} \in \{0,1\}, \ \sum_{p=1}^{c_k} u_{kip} = 1 \quad \sum_{j=1}^{m} w_{kpj} = 1, \quad 1 \le p \le n_k$$

La minimisation de l'équation est réalisée en résolvant itérativement $P1$, $P2$, et $P3$:

1. l' **étape de représentation** nécessite la résolution de deux phases distinctes $P1$ et $P2$:
   - $P1$: fixer $U = \hat{U}$, $W = \hat{W}$ et résoudre le problème réduit $P(\hat{U}, Z, \hat{W})$ par Z
   - $P2$: fixer $U = \hat{U}$, $Z = \hat{Z}$ et résoudre le problème réduit $P(\hat{U}, \hat{Z}, W)$ par W
2. le **problème d'allocation**, nécessite la résolution du problème défini par $P3$:
   - $P3$: fixer $Z = \hat{Z}$, $W = \hat{W}$ et résoudre le problème réduit $P(U, \hat{Z}, \hat{W})$ par U.

Solutions de la fonction objective

Les résultats de la solution de la nouvelle fonction objective sont présentés ci-dessous :

$$z_{kpj} = \frac{\sum_{i=1}^{n_k} u_{kip} x_{ij} (x_{ij} - z_{kGj})}{\sum_{i=1}^{n_k} u_{kip} (x_{ij} - z_{kGj})}$$

Le minimiseur W_k du problème d'optimisation $P2$ est donné par :

$$w_{kpj} = \frac{1}{\sum_{l=1}^{m} \left(\frac{D_{kpj}}{D_{kpl}} \right)^{\frac{1}{\beta-1}}} \text{ with } D_{kpj} = \sum_{i=1}^{n_k} u_{gip} \frac{(x_{ij} - z_{kpj})^2}{n_k (z_{kpj} - z_{kGj})^2}$$

Le problème $P3$ est résolu par

$$u_{kip} = \begin{cases} 1 & \text{if } \sum_{j=1}^{m} w_{kpj}^2 \frac{(x_{ij} - z_{kpj})^2}{n_k (z_{kpj} - z_{kGj})^2} \leq \sum_{j=1}^{m} w_{kpj}^2 \frac{(x_{ij} - z_{krj})^2}{n_k (z_{krj} - z_{kGj})^2} \\ 0 & \text{otherwise} \end{cases}$$

où $1 \leq r \leq k,\ r \neq p$.

Solutions pour les centroïdes

Le représentant (par exemple le centroïde) de C_k peut être interprété comme une moyenne pondérée des éléments du sous-groupe avec des poids égaux à la distance par rapport au centroïde global z_{kG_j} de l'autre (ou des autres, s'il y en a plus de deux) classe(s) originale(s).

$$z_{kpj} = \frac{\sum_{i=1}^{n_k} u_{kip} x_{ij} (x_{ij} - z_{kGj})}{\sum_{i=1}^{n_k} u_{kip} (x_{ij} - z_{kGj})}$$

Plus la distance est grande, plus l'élément du sous-groupe contribue avec un poids élevé à la détermination du centroïde du sous-groupe.

Ce résultat est dû à l'optimisation de la composante discriminante du critère qui met l'accent sur la séparation entre les classes.

Algorithm 1 K-means avec une nouvelle function objective

1: **Input :** $X = \{X_1, X_2, \ldots, X_n\}$, $n_{c_1}, \ldots, n_{c_N}$
2: **Output :** U, Z, W
3: Diviser l'ensemble de données dans les groupes d'origine
4: initialisation aléatoire $Z^0 = \{Z_1, Z_2, \ldots, Z_{k1}\}$ and weights W^0
5: **repeat**
6: Fixed $\hat{W}, \hat{Z}$ résoudre la matrice U
7: Fixed $\hat{U}, \hat{W}$ résoudre la matrice Z
8: Fixed $\hat{U}, \hat{Z}$ résoudre la matrice W
9: **until** convergence

Sélection de caractéristiques basée sur les poids DC

- Le processus de regroupement implique une étape pondérée qui prend en compte **la pertinence des variables pour chaque sous-groupe des classes d'origine**.
  L'étape pondérée facilite la sélection des variables.
- La mesure de variation pondérée (CV) est utilisée comme critère pour déterminer l'importance des caractéristiques sur la base de la matrice de poids.

La mesure de la pertinence des variables est donnée par :

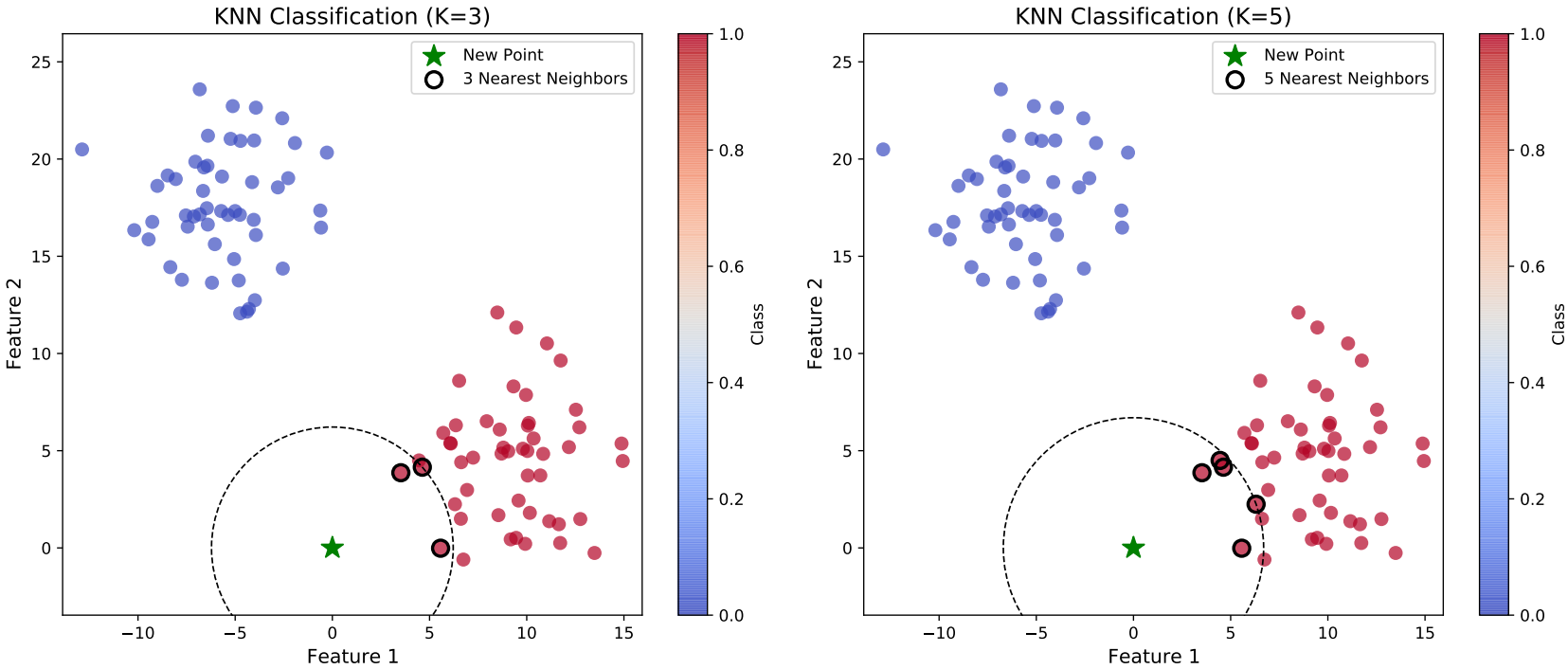
$$CV_{pj}(W) = \frac{Std(W^{pj})}{\mu(W^{pj})} \quad j = 1, \ldots, m \ ; \ p = 1, \ldots, c_k$$

Dove W^{pj} indica la colonna di variabili j per il gruppo p della matrice W.

Le CV facilite l'évaluation relative des poids des variables pour chaque sous-groupe.

Classification supervisée - Algorithme K-NN (K-Nearest Neighbors) et variantes

Rappel de l'algorithme K-NN (K=3 et K=5))



K-Nearest Neighbors (KNN) algorithme et variantes

Principaux défis de l'algorithme K-NN :

- Sensibilité à K : le choix de K a un impact considérable sur les performances
- Curse of Dimensionality (malédiction de la dimensionnalité) : La précision diminue dans les espaces à haute dimension.
- Coût de calcul : Peut être lent pour les grands ensembles de données
- Données déséquilibrées : Difficultés liées aux distributions inégales des classes
- Données bruitées : Sensibles aux valeurs aberrantes et aux caractéristiques non pertinentes

KNN Variantes :

- KNN pondéré : Donne plus de poids aux voisins les plus proches
- KNN basé sur la moyenne locale : Utilise la moyenne de K voisins au lieu du vote majoritaire
- KNN adaptatif : Ajuste K en fonction de la densité locale des points

KNN utilisant les centroïdes comme plus proches voisins

La procédure de classification proposée utilise une **nouvelle variante de l'algorithme KNN** qui se fonde sur la classification des instances dans la classe la plus proche en calculant la **distance entre la nouvelle instance et les centroïdes des sous-groupes**.

Une nouvelle instance X_i de l'ensemble de test est attribuée en fonction du minimum de distances adaptatives entre X_i et le barycentre Z_{gp} des sous-groupes y_p de la classe a priori g :

$$d_{W_g p}(X_i, Z_{gp})$$

La distance adaptative pour un point centroïde donné Z_{gp} et une requête X_i est définie comme suit :

$$d_{y_g}(X_i, Z_{gp}) = d_{W_{gp}}(X_i, Z_{gp}) = \sum_{j=1}^{m} w_{gpj}(x'_{ij} - z_{gpj})^2$$

Ici, W_{gp} est un vecteur de poids correspondant au p-ième sous-groupe de y_g, où g représente la classe d'origine.

Application à un ensemble de données fourni par un partenaire financier européen

L'ensemble de données contient **96 attributes** dont :

- l'heure des la transaction,
- le nombre de transactions,
- le type de transactions,
- le pays où les transactions ont lieu, et
- autres attributs ont été indiqués de V1 à V90 pour cacher les informations sensibles des détenteurs de cartes.

La variable cible de l'ensemble de données est binaire ; elle prend la valeur 0 pour une transaction légitime et 1 pour une transaction frauduleuse.

Application à l'ensemble de données fourni par notre partenaire financier européen

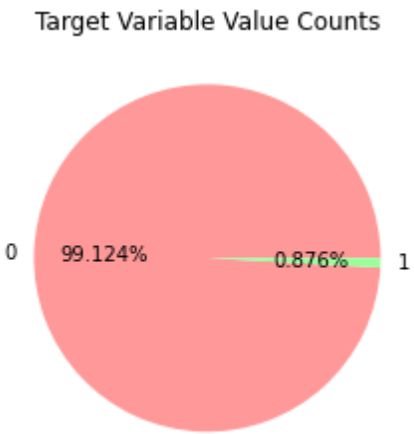


Figure – Diagramme circulaire du ratio de classes cibles

Figure – Histogramme des effectifs des deux classes après équilibrage des données

L'ensemble expérimental pour la détection des transactions frauduleuses comprend des données enregistrées entre le 1er mars et le 30 juin 2016 (trois mois), pour un total de 5 millions de transactions et 75 variables.

Figure – Nombre de fraudes par mois

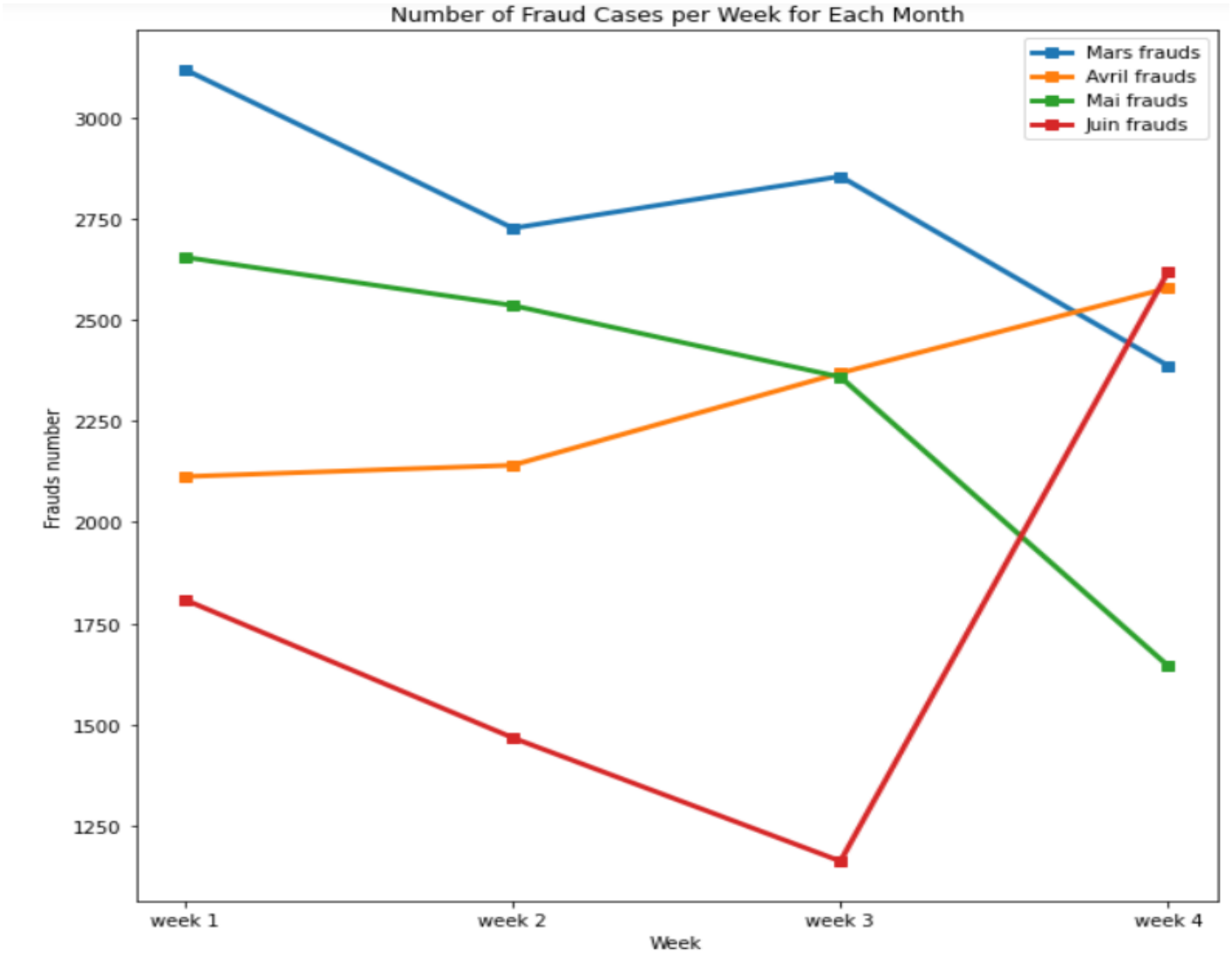
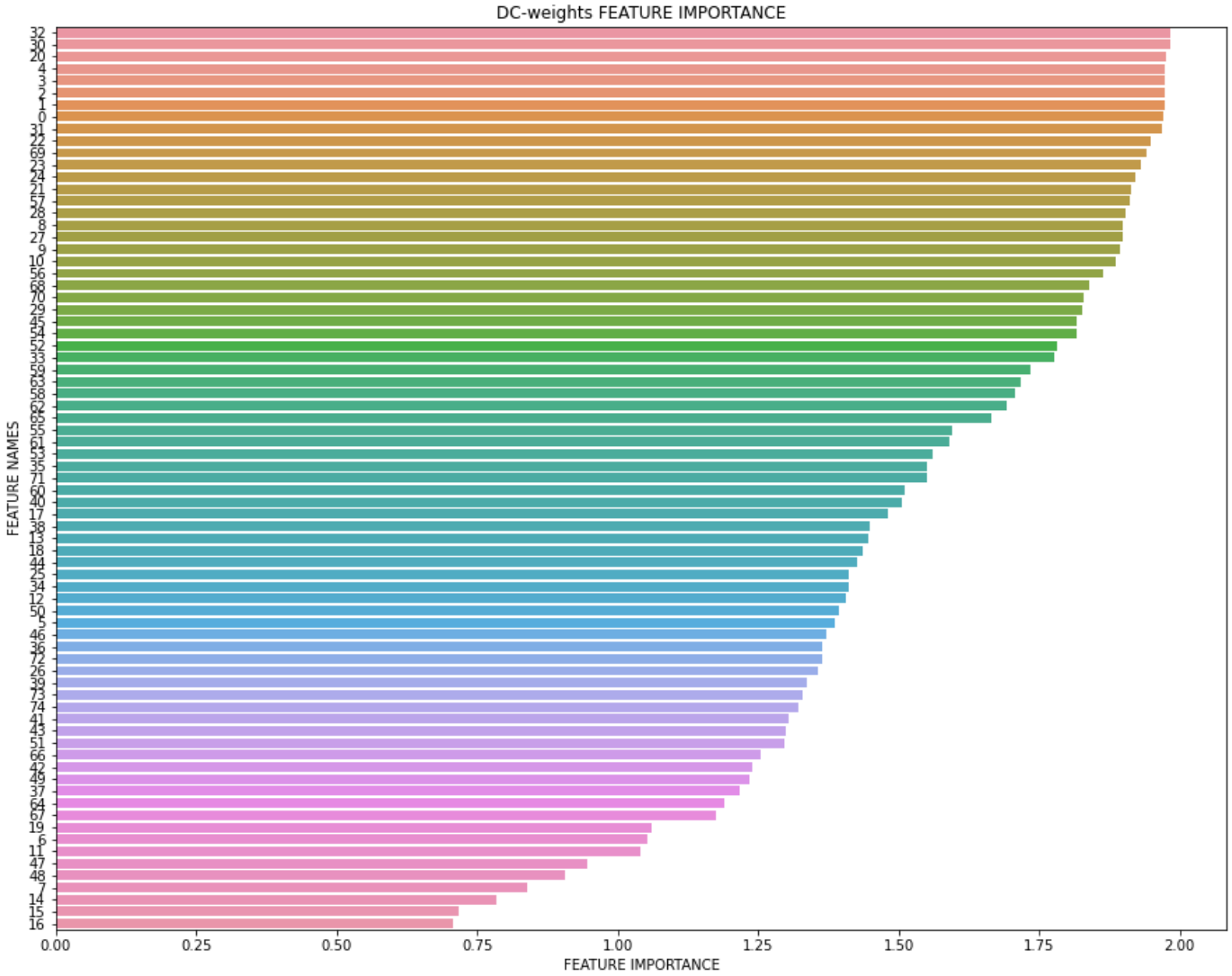


Figure – Histogramme des principales caractéristiques



Résultats comparatifs

Classifiers	Recall	Precision	Specificity	F1 Score	AUC
K-FUSE	83.68	89.39	99.98	86.44	91.83
KNN	82.97	72.22	99.94	77.22	91.46
Random Forest	79.43	78.87	99.96	79.15	89.69
Decision Tree	81.56	37.70	99.77	51.56	90.66
XGBoost	80.14	84.32	99.97	82.18	90.05
Logistic regression	64.53	79.34	99.98	73.38	82.26
SVM	70.21	97.05	99.99	81.48	85.10

$$\text{Recall} : 100 \times \frac{TP}{TP+FN}$$

$$\text{Precision} : 100 \times \frac{TP}{TP+FP}$$

$$\text{Specificity} : 100 \times \frac{TN}{TN+FP}$$

$$\text{F1 score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} = \frac{TP}{TP + 1/2(FP+FN)}$$

Évaluation de la précision et résultats explicatifs de la méthode K-Fuse

L'utilisation de K-Fuse avec un nouveau critère de sélection des caractéristiques permet d'obtenir les résultats les plus précis et les plus fiables en termes de **Recall**.

résultats explicatifs

Cela implique que **notre modèle améliore le TP - taux de vrais positifs et réduit le FP - taux de faux negative**, ce qui indique **une meilleure détection des cas réels de fraude**.

Cependant, il en résulte également une mauvaise classification de certaines **transactions légitimes comme frauduleuses**, **suggérant la présence de modèles supplémentaires au sein du groupe initial de transactions légitimes que le modèle doit identifier**.

Contexte et objectifs : classification des données à haute dimension

L'augmentation des données caractérisées par des séquences d'observations variant dans le temps et l'espace, telles que les courbes et les surfaces, est de plus en plus fréquente, ce qui souligne l'importance de l'analyse des données fonctionnelles (ADF).

Contrairement aux méthodes traditionnelles qui traitent les données comme des points discrets ou des vecteurs, L'ADF assume les courbes ou les fonctions entières en tant qu'entités d'analyse.

Notre proposition

Une stratégie innovante pour la classification de données fonctionnelles, menant à :

- Améliorer la précision de la classification grâce à une intégration de techniques d'apprentissage supervisé et non supervisé
- Introduire une nouvelle fonction objective pour le clustering afin de découvrir des modèles cachés
- Améliorer le critère de prédiction basé sur le FKNN qui prend en compte la variabilité des distributions locales d'échantillons dans les différentes classes.

Motivation

- Découverte de sous-groupes cachés :
  - **Défi :** Les méthodes traditionnelles de classification des données fonctionnelles peuvent manquer de considerer des structures sous-jacentes complexes ou des model subtils dans les données.
  - **Impact :** L'absence de détection de ces sous-groupes cachés peut conduire à une compréhension incomplète des données et à des classifications moins précises, en particulier dans des ensembles de données complexes ou de haute dimension.
- Influence des valeurs aberrantes :
  - **Défi :** Les valeurs aberrantes dans le k-voisinage peuvent affecter de manière disproportionnée le processus de vote majoritaire dans FKNN, conduisant à une mauvaise classification.
  - **Impact :** Ceci est particulièrement problématique dans les scénarios avec de petits échantillons où les valeurs aberrantes peuvent significativement fausser les résultats.
- Défauts du vote majoritaire :
  - **Défi :** Le mécanisme simpliste de vote majoritaire dans FKNN ne tient pas compte adéquatement de la variabilité des distributions d'échantillons locaux entre différentes classes.
  - **Impact :** Cela conduit souvent à une performance de classification réduite, en particulier dans les ensembles de données hétérogènes.

Représentation des données fonctionnelles

Les données enregistrées à des points temporels discrets $t_1, \ldots, t_n$ peuvent être représentées comme une fonction continue

$$X_i(t) \in \mathbb{R}, \qquad t \in [a, b], \qquad i = 1, \ldots, n$$

Les n courbes sont observées sur un intervalle commun $[a, b]$.

→ La représentation des séquences de données sous forme de fonctions permet de :

- faire face à la malédiction de la dimensionnalité ;
- évaluer l'enregistrement à n'importe quel point temporel ;
- évaluer les taux de changement ;
- réduire le bruit ;
- permettre l'enregistrement sur une échelle de temps commune.

Représentation des données fonctionnelles

Dans une representation par des fonctions, les valeurs $X(t)$ existent à n'importe quel point t ; alors que les données originales ne sont disponibles qu'à certains points spécifiques t.

Ansi, les objets en ADF sont représentés comme des courbes lisses :

$$\{X_i(t): \quad t \in [a,b], \quad i = 1,\ldots,n\}$$

- Typiquement exprimées en expansion de base :

$$X(t) = \sum_{j=1}^{K} C_j \phi_j(t)$$

où ϕ_j est la k-ième fonction de base et C_j est un coefficient de la j-ième fonction de base.

Ref. : Ramsay, Silverman Functional Data Analysis, Springer (2005)

L'ensemble d'apprentissage dans une nouvelle méthode de classification des données fonctionnelles

Considérons un ensemble **E** de séquences de valeurs réelles $\mathbf{x}_i(t) \in T$ sur une grille temporelle t.
L'ensemble d'apprentissage, représenté par une partition de **E** de n séquences de valeurs observées, est exprimé en fonctions par lissage :

$$X_i(t) = \sum_{j=1}^{K} C_{ij} B_j(t)$$

en tant que combinaisons de bases (B-splines) au moyen de coefficients C_{ij}.

On suppose qu'on connaît l'appartenance des fonctions $X_i(t)$ à les classes à priori $\{G_1, \ldots, G_g, \ldots, G_N\}$: $X_{ig}(t)$.

Une nouvelle méthode de classification des données fonctionnelles

1. L'objectif est de découvrir des modèles cachés dans l'ensemble de données fonctionnelles en cours d'apprentissage.
2. On recherche donc les sous-groupes S_{gp} (avec $p = 1, \ldots, n_g$) par la partition de chaque classe a priori G_g.
3. Le classificateur prendra alors en compte la structure en sous-groupes des classes a priori afin d'améliorer la précision du modèle de classification.

Étapes de partitionament de classes à priori au moyen d'un algorithme de type *Nuées dynamiques*

L'algorithme est basé sur l'optimisation d'un critère d'homogénéité intra-classe

$$\Delta(U, W, \boldsymbol{\mu}) = \sum_{k=1}^{K} \sum_{i \in C_k} d_w^2(x_i(t), \mu_k(t))$$

et se déroule en deux étapes : *représentation* et *d'affectation* ;

Lors de l'utilisation *de distances adaptatives* : $d_w^2(x_i(t), \mu_k(t))$, l'*étape de représentation* est divisée en deux étapes, afin de déterminer le meilleur système de points $W_k = \{w_{k1}, \ldots, w_{kn}\}$ pour chaque classe :

1. **Étape de représentation**

   - **Étape 1 :** fixer la matrice $\hat{U}$ d'appartenance et le vecteur de poids $\hat{W}$
     Trouver la solution $\boldsymbol{\mu}_k = \{\mu_{k1}, \ldots, \mu_{kn}\}$ du problème d'optimisation $\Delta(\hat{U}, \hat{W}, \boldsymbol{\mu})$.
   - **Étape 2 :** fixer la matrice $\hat{U}$ d'appartenance et le vecteur de centroïdes $\hat{\boldsymbol{\mu}}$
     Trouver le vecteur de poids $W_k = \{w_{k1}, \ldots, w_{kn}\}$ qui minimise le critère $\Delta(\hat{U}, W, \hat{\boldsymbol{\mu}})$.

2. **Étape d'affectation** : fixer les vecteurs de poids $\hat{W}$ et des centroïdes $\hat{\boldsymbol{\mu}}$) :
   Trouver la matrice d'appartenance U qui minimise le critère $\Delta(U, \hat{W}, \hat{\boldsymbol{\mu}})$

La nouvelle fonction de partition de l'ensemble d'apprentissage

Une nouvelle fonction objective est proposée pour découvrir des modèles cachés dans l'ensamble d'apprentissage de données fonctionnelles.

Cet objectif est atteint en intégrant deux éléments clés : l'homogénéité (compactness) intra-groupe des courbes de sous-groupes au sein du même groupe a priori et la séparation inter-groupes entre les courbes de sous-groupes et celles de classes a priori différentes.

De plus, cette fonction objective utilise une distance euclidienne pondérée pour évaluer l'importance des caractéristiques dans l'optimisation de la fonction critère nouvellement introduite.

La séparation inter-groupes devient essentielle pour distinguer la pertinence de divers modèles, en reconnaissant l'hétérogénéité présente parmi les sous-groupes au sein de chaque groupe d'origine.

Une nouvelle fonction objective

Nous définissons une nouvelle fonction objective pour calculer la similarité des données, qui vise à minimiser les distances intra-cluster entre les sous-groupes d'un groupe original en utilisant **une distance adaptative**, tout en maximisant la distance inter-cluster avec les sous-groupes des autres classes originales.

Une nouvelle fonction objective est définie comme :

$$P(\mathbf{U},\mathbf{W},\boldsymbol{\mu}) = \sum_{g=1}^{N} \sum_{G_g:p=1}^{n_g} \sum_{X_i \in S_{gp}} \Delta_{w_p(t)}(X_i(t), \mu_{gp}(t))$$

où G_g est la classe a priori partitionnée en n_g sous-groupes S_{gp} (with $p = 1,\ldots,n_g$), et S_{gp} est le sous-groupe auquel est assignée la i-ième courbe en fonction de sa distance au barycentre μ_{gp}.

Si l'on considère la distance fonctionnelle ou semi-métrique, la distance $\Delta_{w_p(t)}$ devient :

$$\Delta_{w_p(t)}(X_i(t), \mu_{gp}(t)) = \frac{d_{w_p}(X_i(t), \mu_{gp}(t))}{d(\mu_{gp}(t), \mu_{Gg}(t))}$$

où : $\mu_{G_g}(t)$ est le centroïde global fonctionnel d'une classe à priori, excluée la classe G_g.

Distances dans la nouvelle fonction objective

Suite au lissage des données fonctionnelles, nous supposons que la distance $\Delta_{w_p(t)}$ dans la fonction objective est approximée de la manière suivante :

$$P(\mathbf{U},\mathbf{W},\boldsymbol{\mu}) = \sum_{g=1}^{N} \sum_{G_g:p=1}^{n_g} \sum_{i \in S_{gp}} \sum_{j=1}^{K} (C_{p,j}^{w})^{\beta} \frac{(C_{i,j}^{X} - C_{p,j}^{\mu})^2}{(C_{p,j}^{\mu} - C_{gG,j})^2}$$

où :
- $C_{p,j}^{w}$ sont les coefficients de l'expansion des poids,
- $C_{i,j}^{X}$, $C_{p,j}^{\mu}$ sont les coefficients de l'expansion de la courbe $X_i(t)$ et $\mu_{gp}(t)$ respectivement,
- et $C_{gG,j}$ sont les coefficients de l'expansion du centroïde global, à l'exclusion de la classe a priori G_g.

Solutions de la fonction objective - Étape de répresentation

Pour initier le processus de résolution de la fonction objective, il est nécessaire :
- d'initialiser les partitions $\mathbf{U}=\{\mathbf{U}_1,\ldots,\mathbf{U}_g,\ldots,\mathbf{U}_N\}$ des classes à priori G_g in n_g sous-groupes.
- Par l'étape de representation on determine les paramètres $\hat{\mathbf{W}}$ and $\hat{\boldsymbol{\mu}}$ de tous les groupes réduisant le problème de minimisation de l'équation suivant :

$$P(\mathbf{W},\boldsymbol{\mu}) = \sum_{G_g:p=1}^{n_g} \sum_{i:S_{gp}} \sum_{j=1}^{K} (C_{p,j}^{w})^{\beta} \frac{(C_{i,j}^{X} - C_{p,j}^{\mu})^2}{(C_{p,j}^{\mu} - C_{gG,j})^2}$$

$$\text{s.c.} \quad \sum_{j=1}^{K} C_{p,j}^{w} = 1, \quad 1 \leq p \leq n_g$$

Pour minimiser la fonction objective, il est impératif de résoudre les problèmes $P1$ et $P2$ dans l'étape de représentation :

1. Problème $P1$: Fixer $\mathbf{W} = \hat{\mathbf{W}}$ et résoudre le problème simplifié $P(\hat{\mathbf{W}},\boldsymbol{\mu})$.
2. Problème $P2$: Fixer $\boldsymbol{\mu} = \hat{\boldsymbol{\mu}}$, le problème est davantage contraint pour résoudre le problème réduit $P(\mathbf{W},\hat{\boldsymbol{\mu}})$.

- Par la suite, considérons la situation d'*affectation* dans le groupe p qui est évaluée.

Solutions de la fonction objective - Étape de répresentation (P1)

Pour résoudre le problème $P1$, nous dérivons la function P par rapport au coefficient $C^{\mu}_{p,j}$ du sous-groupe S_p de la classe à priori G_g comme suit :

$$\frac{\partial P(\hat{\mathbf{W}}, \boldsymbol{\mu})}{\partial C^{\mu}_{p,j}} = -2(C^{w}_{p,j})^{\beta} \sum_{X_i \in S_p} [(C^{X}_{i,j} - C^{\mu}_{p,j})^2 - \eta(C^{\mu}_{p,j} - C_{gG,j})^2]$$

En égalisant la dérivée à zéro, on obtient :

$$C^{\mu}_{p,j} = \frac{\sum_{X_i \in S_p} C^{X}_{i,j}(C^{X}_{i,j} - C_{gG,j})}{2 \sum_{X_i \in S_p} (C^{X}_{i,j} - C_{gG,j})}$$

Solutions de la fonction objective - Étape de répresentation (P2)

Pour résoudre le problème $P2$, nous considerons une équation lagrangienne $P(C^W, \alpha)$ correspondant à $P(\mathbf{W}, \boldsymbol{\mu})$. Dans cette équation, nous utilisons α comme multiplicateur de Lagrange :

$$P(\mathbf{W}, \alpha) = \sum_{p=1} n_g \sum_{j=1}^{K} (C_{p,j}^W)^\beta \Delta_{pj} - \alpha(\sum_{j=1}^{K} C_{p,j}^W - 1)$$

où :

$$\Delta_{pj} = \sum_{i=1}^{n_p} \frac{(C_{i,j}^X - C_{p,j}^\mu)^2}{(C_{p,j}^\mu - C_{gG,j})^2}$$

En prenant la dérivée par rapport à $C_{p,j}^W$ et α et en l'égalant à zéro, nous obtenons :

$$\frac{\partial L(\mathbf{W}, \alpha)}{\partial C_{p,j}^W} = \beta (C_{p,j}^W)^{\beta-1} \Delta_{pj} - \alpha = 0$$

$$C_{pj}^W = \frac{1}{\sum_{l=1}^{K} \left(\frac{\Delta_{pj}}{\Delta_{pl}} \right)^{\frac{1}{\beta-1}}}$$

Solutions de la fonction objective - Étape d'affectation

Initialisés les valeurs en $\mathbf{W}$ et $\boldsymbol{\mu}$ pour les poids et les centroïdes, respectivement.

Chaque point de données X_i d'une classe a priori G_g est affecté au sous-groupe S_p qui minimise la distance euclidienne pondérée, déterminée par :

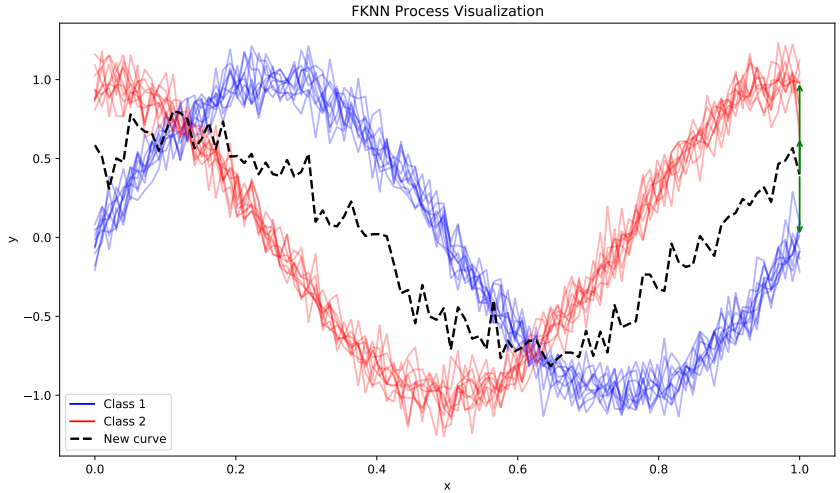
$$l = \underset{p}{\operatorname{argmin}}\, d_{w(t)}(X_i(t), \mu_p(t)) = \underset{p}{\operatorname{argmin}} \sum_{j=1}^{K} C_{pj}^{w}(C_{ij}^{X} - C_{pj}^{\mu})^2$$

où $d_{w(t)}$ est la distance pondérée, $X_i(t)$ est la function à affecter, $\mu_p(t)$ est le centroïde fonctionnel pour le cluster p, et C_{pj}^{w} sont les coefficients de pondération correspondants.

Les étapes de représentation et d'affectation sont réitérées par étapes alternées jusqu'à ce que l'algorithme converge vers des partitions stables $\mathbf{U}_g$ des classes a priori G_g et vers des valeurs déterminées des poids et centroïdes des sous-classes : $\mathbf{W}_g$ et $\boldsymbol{\mu}_g$

Étape de Classification – Critere du KNN Fonctionnelles - FKNN

Les K plus proches voisins fonctionnels (FKNN) sont une extension de l'algorithme classique des K **plus proches voisins (KNN)** conçue spécifiquement pour les **données fonctionnelles**.



Distance :

Norme L2 (Distance euclidienne) : Mesure la distance entre deux fonctions $x_i(t)$ and $x_j(t)$ comme :

L_2-distance

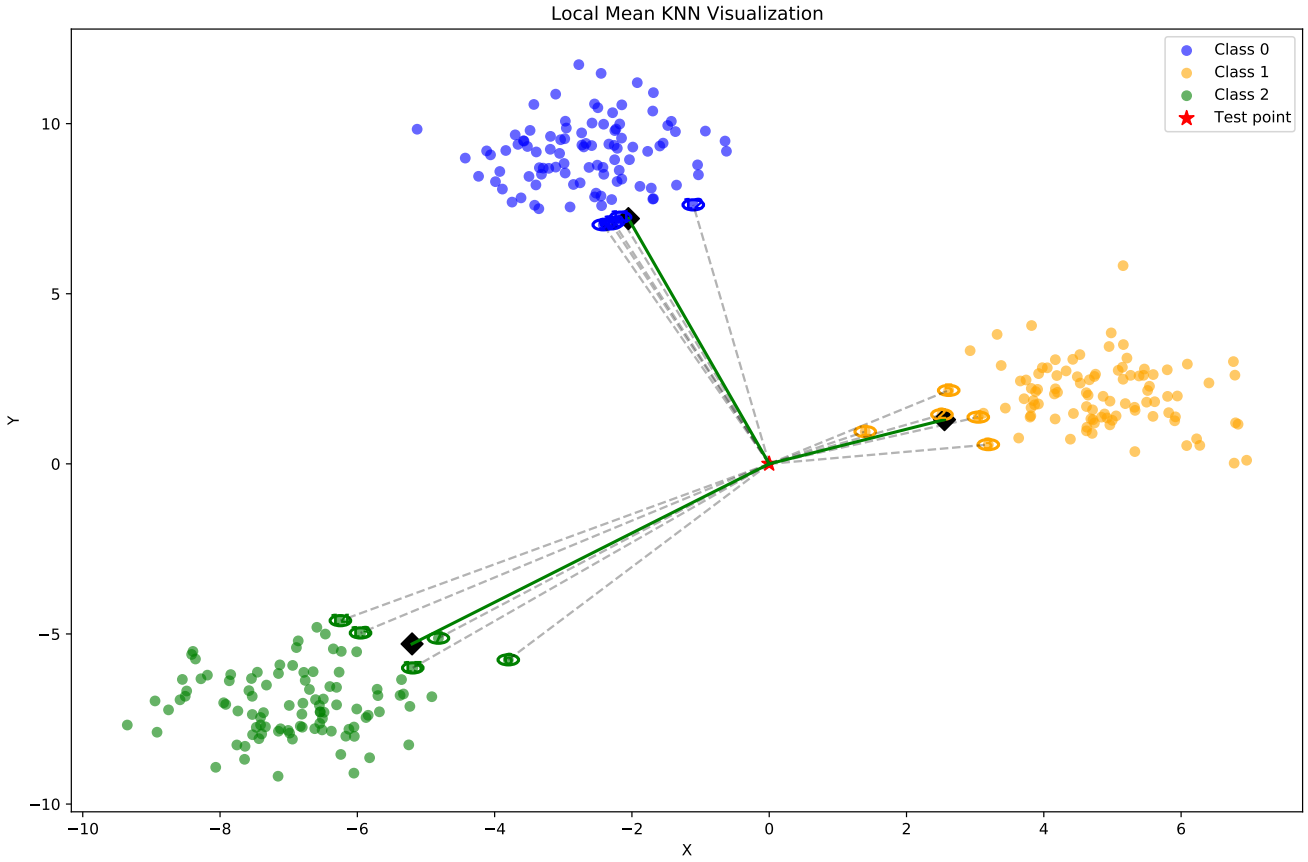
$$\left\| x_i(t) - x_j(t) \right\|_2 = \left\{ \frac{1}{\int_a^b w(t)dt} \int_a^b \left| x_i(t) - x_j(t) \right|^2 w(t)dt \right\}^{1/2} \quad (1)$$

où w sont les poids et les points observés sur chaque courbe sont espacés également (Febrero-Bande and de la Fuente, 2012).

Local Mean KNN - LMKNN

Pour un nouvel élément,
- on calcule les K points les plus proches de chaque classe
- on calcule ensuite les moyennes locales des distances les plus proches pour chaque classe.

→ L'élément est attribué à la classe dont la distance moyenne est la plus petite.



Approche proposée : Weighted Functional Local Mean KNN - FWLMKNN

FWLMKNN (Functional Weighted Local Mean K-Nearest Neighbor) est une méthode innovante de classification des données fonctionnelles qui :

1. Abordant les limitations inhérentes des méthodes traditionnelles des K plus proches voisins fonctionnels (FKNN)
2. Exploitant les riches informations de sous-groupes obtenues à partir de notre nouvelle phase de clustering
3. Incorporant des distances pondérées adaptatives pour mettre l'accent sur les caractéristiques pertinentes
4. Utilisant des vecteurs moyens locaux pour mieux capturer les structures de données complexes au sein des sous-groupes

Cette approche crée un pipeline synergique du clustering à la classification, optimisant le traitement des modèles de données fonctionnelles complexes.

Au lieu de s'appuyer sur un vote majoritaire, FWLMKNN utilise des vecteurs moyens locaux des k plus proches voisins au sein de chaque classe.

FWLMKNN : Étapes de l'Algorithme

1. **Identifier les k plus proches voisins :** Fixe k
   Soit $C_{gp}^{NN} = \{C_{ip}^{X} \in \mathbb{R}^{K}\}_{i=1}^{k}$ les coefficients du développement des k courbes les plus proches de X du sous-groupe S_{gp} de la classe d'apriori G_g.

2. **Calculer le vecteur moyen local :**
   Le vecteur moyen local de k voisins les plus proches dans chaque sous-groupe S_{gp} est calculé comme suit :
   $$\overline{C}_{gp}^{NN} = \frac{1}{k} \sum_{i=1}^{k} C_{ip}^{X}$$

3. **Attribuer la classe :**
   Enfin, la fonction de requête X est attribué à la classe à priori G_g qui présente la distance euclidienne pondérée minimale entre X' et le vecteur moyen local d'un sous-groupe spécifique $\overline{C}_{gp}^{NN}$ parmi toutes les classes :
   $$G = \underset{G_g}{\operatorname{argmin}}\, d_{w_p}(X, \overline{C}_{gp}^{NN}) \quad g = 1, \ldots, N \quad p = 1, \ldots, n_g$$

Approche proposée : FWLMKNN (suite)

Avantages

- Réduit la sensibilité aux valeurs aberrantes $\rightarrow$ Résultats plus robustes
- Capture les structures de données locales $\rightarrow$ Compréhension nuancée des modèles complexes
- Améliore la précision de classification pour les données fonctionnelles complexes domaines

Applications sur trois jeux de données

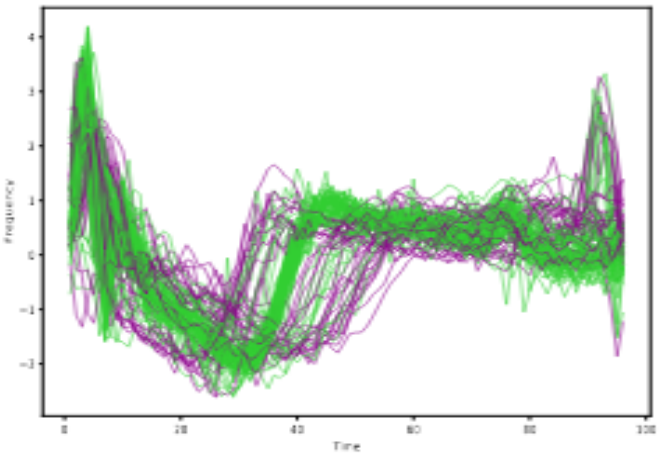
L'analyse comparative a été réalisée sur la base de trois ensembles de données distincts provenant du répertoire de classification des séries temporelles de l'UCR

Table – Description des jeux de données fonctionnelles

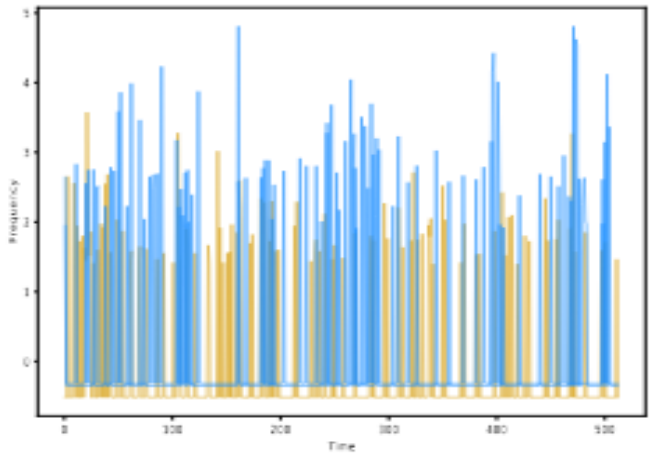
Dataset	**Samples**	**Features**	**Classes**	**Training**
ECG200	200	96	2	100
Earthquakes	461	512	2	322
Gun	451	150	2	115

Chaque expérience a été répétée dix fois, l'ensemble de données étant divisé de manière aléatoire en sous-ensembles d'apprendissage et de test de la taille indiquée dans le tableau pour chaque itération afin de garantir de meilleurs résultats.

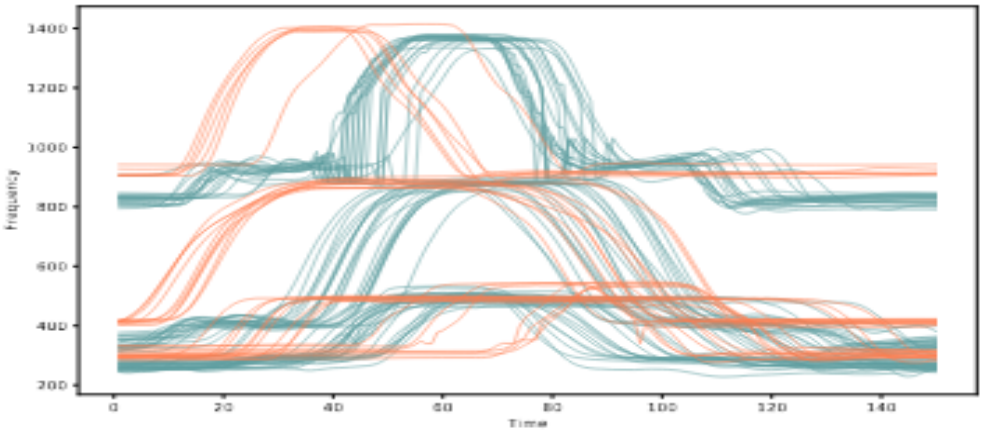
Visualization des trois jeux de données



(a) ECG200 dataset



(b) Earthquakes dataset

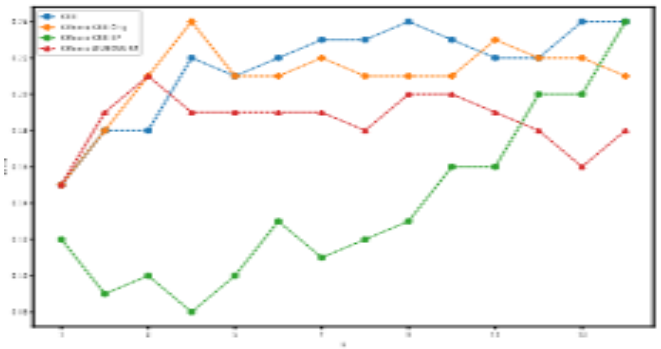


(c) Gun dataset

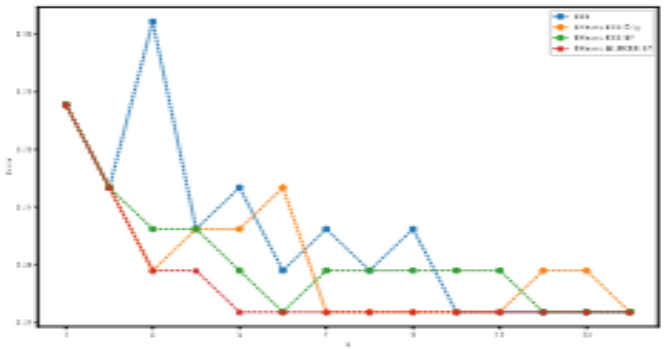
Figure – Visualization of signal profiles from real datasets.

Analyse comparative de la précision pour différentes valeurs de k sur trois ensembles de données

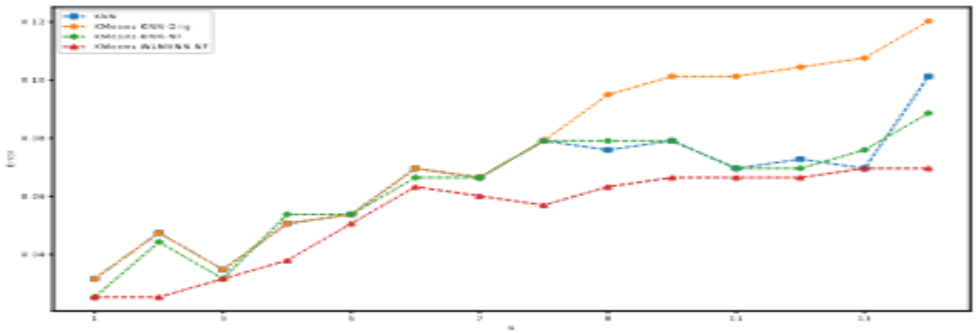
La taille du voisinage, k, a été systématiquement modifiée de 1 à 15 par incréments de 1 afin de déterminer le k optimal pour chaque ensemble de données en sélectionnant le k qui a donné lieu au taux d'erreur le plus faible dans cette échelle.



(a) ECG200 dataset



(b) Earthquakes dataset



(c) Gun dataset

Résultats de l'approche proposée

L'efficacité du nouveau classificateur est déterminée en le comparant aux algorithmes KNN et KM-KNN, ce dernier utilisant la méthode classique K-means. Cette évaluation se concentre sur l'effet de la modification de la fonction objective de K-means et de l'utilisation des étiquettes augmentées produites par K-means

Table – Taux d'erreur minimaux, écarts-types pour chaque classificateur sur trois jeux de données réels.

Dataset	FKNN	LMKNN	KM-KNN	FWLMKNN
ECG200	21.50 ± 0.0154	20.92 ± 0.0124	13.85 ± 0.027	18.57 ± 0.0089
Earthquakes	26.51 ± 0.87	26.11 ± 0.63	26.10 ± 0.57	25.69 ± 0.60
Gun	6.44 ± 1.08	7.59 ± 0.0169	6.30 ± 1.09	5.37 ± 0.95

Ce tableau compare les taux d'erreur minimaux et les valeurs k optimales pour différents classificateurs sur trois jeux de données réels.

Conclusion

Notre proposition

- L'espace d'étiquettes augmenté a significativement amélioré la précision de la classification.
- La métrique de distance pondérée a amélioré la pertinence des caractéristiques.
- Utilise des vecteurs moyens locaux, minimisant l'impact du paramètre de taille du voisinage k sur la précision de la classification.
- A conduit des expériences rigoureuses sur trois ensembles de données fonctionnelles du monde réel, et a démontré une performance supérieure de FWLMKNN par rapport aux algorithmes traditionnels.

Travaux futurs

- Investiguer les techniques de calcul parallèle pour réduire le temps de calcul.
- Explorer d'autres métriques fonctionnelles et explorer des algorithmes de clustering plus efficaces pour la phase initiale.
- Développer des techniques de visualisation pour l'espace d'étiquettes augmenté et les vecteurs moyens locaux.

Merci !