Classification des Séries Temporelles Incertaines Par Transformation "Shapelet"

Michael Franklin Mbouopda, Engelbert Mephu Nguifo

Classification des Séries Temporelles Incertaines Par Transformation Shapelet

Michael Franklin MBOUOPDA[1], Engelbert MEPHU NGUIFO[1]

[1] University Clermont Auvergne, CNRS, ENSMSE, LIMOS, F-63000 CLERMONT-FERRAND, FRANCE

{michael.mbouopda, engelbert.mephu_nguifo}@uca.fr

Abstract

Time series classification is a task that aims at classifying chronological data. It is used in a diverse range of domains such as meteorology, medicine and physics. In the last decade, many algorithms have been built to perform this task with very appreciable accuracy. However, the uncertainty in data is not explicitly taken into account by these methods. Using uncertainty propagation techniques, we propose a new uncertain dissimilarity measure based on euclidean distance. We also show how to classify uncertain time series using the proposed dissimilarity measure and shapelet transform, one of the best time series classification methods. An experimental assessment of our contribution is done on the well known UCR dataset repository.

Keywords

Time series, Classification, Uncertainty, Shapelet.

Résumé

La classification des séries temporelles est une tâche qui consiste à classifier les données chronologiques. Elle est utilisée dans divers domaines tels que la météorologie, la médecine et la physique. Plusieurs techniques performantes ont été proposées durant les dix dernières années pour accomplir cette tâche. Cependant, elles ne prennent pas explicitement en compte l'incertitude dans les données. En utilisant la propagation de l'incertitude, nous proposons une nouvelle mesure de dissimilarité incertaine basée sur la distance euclidienne. Nous montrons également comment faire la classification de séries temporelles incertaines en couplant cette mesure avec la méthode de transformation shapelet, l'une des méthodes les plus performantes pour cette tâche. Une évaluation expérimentale de notre contribution est faite sur le dépôt de données temporelles UCR.

Mots-clés

Série temporelle, Classification, Incertitude, Shapelet.

1 Introduction

La dernière décennie fut caractérisée par la disponibilité de données dans un large domaine d'application tels que la météorologie, l'astronomie et le suivi d'objets. Généralement, ces données sont représentées sous forme de séries temporelles [6], c'est-à-dire des données séquentielles ordonnées suivant le temps. Par ailleurs, plusieurs méthodes de classification automatique de ces types de données ont été développées durant la même décennie [3, 8, 11]. Pour autant que nous sachions cependant, toutes ces méthodes supposent que les données sont précises et fiables. Ainsi elles ne prennent pas en compte l'incertitude présente dans les mesures. Pourtant, toute mesure est sujette à l'incertitude qui pourrait provenir de l'environnement, de la précision de l'outil de mesure, des contraintes de confidentialité et bien d'autres facteurs. De plus, même si l'incertitude peut être réduite, elle ne peut être supprimée [18]. Dans certaines applications, l'incertitude ne peut être négligée et doit être traité avec rigueur [17]. Par exemple, [5] a montré l'importance de la prise en compte de l'incertitude dans l'interprétation des images médicales. De même, la nécessité de bien gérer l'incertitude dans les données spatiales a été mise en évidence par [7].

Les méthodes basées sur les shapelets sont parmi les meilleures approches qui ont été développées pour la classification des séries temporelles [3]. Elles sont spécialement appréciées pour leur caractère interprétable, leur robustesse et leur vitesse d'inférence [19]. Ces approches procèdent en trois étapes :

1. étape d'extraction des shapelets, qui peut être vue comme une extraction de caractéristiques,
2. étape de transformation, qui consiste à calculer les vecteurs caractéristiques de chaque séries temporelle dans le jeu de données,
3. étape d'apprentissage, qui consiste à entraîner un modèle de classification supervisée sur le jeu de données obtenu après transformation.

Dans ce papier, nous montrons comment cette approche peut être adaptée et appliquée dans le contexte de la classification des séries temporelles incertaines. Pour y parvenir, nous proposons premièrement une mesure de dissimilarité incertaine inspirée de la distance euclidienne. Ensuite, nous intégrons cette mesure dans l'approche de transformation shapelet afin de faire la classification des séries temporelles incertaines.

La suite de cet article est organisée comme suit : les travaux connexes sont présentés dans la section 2. Dans la section 3, nous présentons une nouvelle mesure de dissi-

milarité incertaine et dans la section 4, nous construisons le modèle de classification des séries temporelles incertaines. La section 5 présente les expérimentations que nous avons effectuées et les résultats obtenus. Finalement, la section 6 conclut ce papier.

2 Travaux connexes

L'analyse des séries temporelles incertaines est un problème bien connu et il existe dans la littérature des travaux sur le sujet. Ces travaux ont conduit au développement de mesures de similarité dites probabilistes. Étant données deux séries temporelles incertaines, une mesure de similarité incertaine calcule la probabilité que la distance entre ces deux séries soit inférieure à un seuil défini par l'utilisateur [6]. Ces mesures probabilistes ont été couplées avec le modèle de classification supervisée 1-PPV (pour 1-Plus Proche Voisin) afin d'effectuer la classification des séries temporelles incertaines. Par exemple, [17] a proposé la mesure de similarité probabiliste DUST et l'a couplée avec un 1-PPV. Les mesures de similarité probabilistes ne sont pas toujours applicables en pratique. En effet, elles sont basées sur des suppositions qui ne sont pas toujours satisfaites. C'est le cas de la mesure de similarité probabiliste PROUD [20] qui requiert que l'incertitude soit la même à chaque observation de la série temporelle. MUNICH [1], une autre mesure de similarité incertaine représente l'incertitude comme étant la non unicité des observations à chaque instant de la série. Ainsi à chaque instant, on n'a pas une unique observation, mais plutôt un ensemble d'observations possibles de cet instant. DUST a été proposé comme solution aux limitations de MUNICH et PROUD, mais fait aussi une supposition qui n'est pas toujours observable en pratique : DUST requiert que l'incertitude des observations du même instant dans toutes les séries temporelles du jeu de données soit la même. Plus récemment, la mesure de dissimilarité FOTS [9] a été proposée ; elle est robuste à l'incertitude, mais ladite incertitude n'est pas explicitement prise en compte dans le calcul de FOTS. Toutes ces mesures de similarité/dissimilarité pour les données incertaines partagent ensemble une même caractéristique : celle de représenter la similarité/dissimilarité entre deux données incertaines par une valeur certaine. Il est impossible de comparer des données contenant de l'incertitude et espérer que le résultat de la comparaison soit sans incertitude. Pour toutes ces raisons, nous proposons à la section 3, UED, une mesure de dissimilarité qui ne fait aucune supposition sur la distribution de l'incertitude et qui donne le résultat de la comparaison accompagné d'un intervalle de confiance.

Introduite par [19] en tant que arbre de décision shapelet, la classification des séries temporelles par shapelet a été généralisée par [10] sous l'appellation transformation shapelet. La généralisation permet d'utiliser les shapelets avec tout modèle de classification supervisée, et plus le modèle de classification utilisé est performant plus la classification des séries est meilleure[10]. Pour autant que nous sachions, le meilleur modèle de classification des séries temporelles tel que reporté dans la littérature est HIVE-COTE [13]. Il s'agit d'un méta-modèle composé de plusieurs modules dont l'un est basé sur les shapelets. Pour avoir un modèle comme HIVE-COTE pour les séries temporelles incertaines, il faut prendre en compte l'incertitude dans chacun des modules. Dans ce papier, nous présentons à la section 4 un moyen de prendre en compte l'incertitude dans le module basé sur les shapelets.

3 UED : Mesure de dissimilarité incertaine

L'incertitude est différente de l'erreur car l'erreur peut être évitée en faisant plus attention alors qu'on ne peut pas échapper à l'incertitude [18]. Cependant il existe des moyens pour réduire l'incertitude. Peu importe la méthode de mesure, il y a toujours une incertitude et les mesures incertaines ne peuvent pas être comparées avec 100% de confiance. Il existe plusieurs représentation de l'incertitude et dans ce papier, une mesure incertaine x est représentée par deux composantes qui sont l'estimation optimiste $\hat{x}$ et l'incertitude δx qui est l'écart maximal possible par rapport à l'estimation optimiste.

$$x = \hat{x} \pm \delta x, \delta x \geq 0 \tag{1}$$

Cette formule signifie que x est un élément de l'interval $[\hat{x} - \delta x, \hat{x} + \delta x]$ et qu'il est fort probable que sa valeur soit $\hat{x}$.

La distance euclidienne (ED) est très utilisée dans la littérature pour mesurer la dissimilarité entre deux séries temporelles. Elle est particulièrement utilisée dans les méthodes à base de shapelet [3, 10, 19]. Étant données deux séries temporelles $T_1 = \{t_{11}, t_{12}, ..., t_{1n}\}$ et $T_2 = \{t_{21}, t_{22}, ..., t_{2n}\}$, la distance euclidienne entre elles est définie comme suit :

$$ED(T_1, T_2) = \sum_{i=1}^{n} (t_{1i} - t_{2i})^2 \tag{2}$$

La racine carrée étant une fonction positive et croissante, il n'est pas obligatoire de l'appliquer dans le calcul de la dissimilarité. En effet, si on ordonne des séries temporelles selon leur similarité les unes par rapport aux autres, l'application ou pas de la racine carrée n'influence pas l'ordre. Lorsque chaque composante t_{ij} est une mesure incertaine, T_1 et T_2 sont appelées des séries temporelles incertaines et la similarité entre elles ne pas être calculée avec 100% de fiabilité. Nous utilisons les techniques de propagation de l'incertitude [18] afin de mesurer l'incertitude qu'il y a dans la dissimilarité entre les deux séries temporelles.

Soient $x = \hat{x} \pm \delta x$ et $y = \hat{y} \pm \delta y$ deux mesures incertaines, nous avons les propriétés suivantes :

- $z = x + y = \hat{z} \pm \delta z$, où $\hat{z} = \hat{x} + \hat{y}$ et $\delta z = \delta x + \delta y$
- $z = x - y = \hat{z} \pm \delta z$, où $\hat{z} = \hat{x} - \hat{y}$ et $\delta z = \delta x + \delta y$
- $z = x^n = \hat{z} \pm \delta z$, où $\hat{z} = (\hat{x})^n$ et $\delta z = |n\frac{\delta x}{x}(x_b)^n|$

En utilisant ces propriétés de propagation de l'incertitude, nous pouvons calculer l'incertitude qu'il y a sur la distance euclidienne entre deux séries temporelles incertaines T_1 et T_2 en propageant l'incertitude. Nous obtenons une mesure

de dissimilarité incertaine que nous appellons UED et elle est définie comme suit :

$$\begin{aligned}\mathrm{UED}(T_1, T_2) &= \sum_{i=1}^{n} (\hat{t_{1i}} - \hat{t_{2i}})^2 \pm 2 \sum_{i=1}^{n} |\hat{t_{1i}} - \hat{t_{2i}}| \times (\delta t_{1i} + \delta t_{2i}) \\ &= ED(\hat{T_1}, \hat{T_2}) \pm 2 \sum_{i=1}^{n} |\hat{t_{1i}} - \hat{t_{2i}}| \times (\delta t_{1i} + \delta t_{2i})\end{aligned} \tag{3}$$

où $\hat{T_i}$ en obtenue en ignorant l'incertitude dans T_i, c'est à dire en supposant que toutes les incertitudes sont à 0.
La sortie de UED correspond à la dissimilarité incertaine entre deux séries temporelles incertaines qu'on place en entrée. Afin d'utiliser UED pour la classification des séries temporelles incertaines, plus précisement avec les méthodes à shapelet, il est nécessaire de définir une relation d'ordre pour les mesures incertaines. Nous proposons deux façons de comparer deux mesures incertaines : la première est la plus simple et est basée sur la confiance et la seconde est un ordre stochastique.

Ordre simple des mesures incertaines

Cet ordonnancement est basé sur deux propiétés basiques. Soient x et y deux mesures incertaines, la première propriété est celle de l'égalité et stipule que les mesures sont équivalentes si et seulement si leurs estimations optimistes et leurs incertitudes sont égales :

$$x = y \iff \hat{x} = \hat{y} \wedge \delta x = \delta y \tag{4}$$

La propriété d'infériorité est la seconde et stipule que x est inférieur y si et seulement si l'estimation optimiste de x est inférieure à celle de y. Lorsque leurs estimations optimistes sont égales, la plus petite est celle qui la plus petite incertitude.

$$x < y \iff (\hat{x} < \hat{y}) \vee ((\hat{x} = \hat{y}) \wedge (\delta x < \delta y)) \tag{5}$$

Contrairement à la propriété d'égalité qui est intuitive, la propriété d'infériorité est moins simple. Malheureusement, nous n'avons pas de justification mathématique de cette propriété. Cependant deux aspects ont guidé sa conception : premièrement nous faisons d'une certaine manière confiance à l'estimation optimiste donnée par l'expert du domaine. Deuxièmement, il est préferable de travailler avec les valeurs dont l'incertitude est réduite au maximum.
Il est à noter que ces deux propriétés ne permettent pas toujours d'avoir le bon ordre ; en effet, si $x = 2 \pm 0.5$ et $y = 2 \pm 0.1$ alors l'application de la propriété d'infériorité nous dit que $y < x$. Maintenant, s'il y avait un oracle capable de calculer la valeur exacte d'une mesure incertaine, il pourrait dire que $x = 1.8$ et $y = 2$. Et ainsi, l'ordre défini serait incorrect. Ce phénomène peut aussi s'observer avec la propriété d'égalité.

Ordre stochastique des mesures incertaines

Une mesure incertaine peut être considérée comme une variable aléatoire dont la moyenne est l'estimation optimiste et l'incertitude est l'écart-type. Partant de cette considération, l'ordre stochastique entre deux mesures incertaines peut être défini. Une variable aléatoire X est *stochastiquement plus petite ou égale* (noté $\leq_{st}$) à une autre variable aléatoire Y si et seulement si $P(X > t) \leq P(Y > t) \ \forall t \in \mathbb{R}$ [14]. Étant donnée que la valeur exacte d'une mesure incertaine x est dans l'intervalle $[\hat{x} - \delta x, \hat{x} + \delta x]$, le domaine de t peut être réduit à l'intervalle $\mathbb{I} = [\min(X,Y); \max(X,Y)]$; où $\min(X,Y)$ et $\max(X,Y)$ sont repectivement les valeurs minimale et maximale possibles de l'union des valeurs de X et Y. L'ordre stochastique peut être écrite et développée comme suit :

$$\begin{aligned}X \leq_{st} Y &\iff P(X > t) \leq P(Y > t) \ \forall t \in \mathbb{I} \\ &\iff 1 - P(X > t) > 1 - P(Y > t) \ \forall t \in \mathbb{I} \\ &\iff P(X \leq t) > P(Y \leq t) \ \forall t \in \mathbb{I} \\ &\iff CDF_X(t) > CDF_Y(t) \ \forall t \in \mathbb{I}\end{aligned} \tag{6}$$

$CDF_X(t)$ est la fonction de répartition de la variable aléatoire X évaluée à t. Puisque le nombre d'éléments de $\mathbb{I}$ est infini, nous discrétisons en divisant l'intervalle en k valeurs différentes :

$$\min(X,Y) + i \times \frac{\max(X,Y) - \min(X,Y)}{k} \tag{7}$$

$0 \leq i \leq k$ et k est un nombre entier à fixer.
Contrairement à l'ordre simple qui est total, l'ordre stochastique n'est qu'un ordre partiel. Ainsi, la relation *stochastiquement plus petite ou égale* n'est pas définie pour toute paire de mesures incertaines. L'ordre stochastique entre deux mesures incertaines n'est donc pas toujours défini. Il s'agit là clairement d'une limitation, cependant nous n'avons pas trouvé un ordre stochastique qui soit total dans la littérature.
Maintenant que nous pouvons ordonner des mesures incertaines, voyons comment UED est utilisée pour la classification de séries temporelles incertaines.

4 Classification shapelet incertaine

Dans cette partie, nous décrivons comment effectuer la classification de séries temporelles incertaines avec l'approche par shapelet. Nous nous basons sur l'algorithme de classification par transformation shapelets [10]. Tout d'abord, il est nécessaire de définir quelques concepts fondamentaux.
Une *série temporelle incertaine* T est une séquence de m (sa taille) valeurs incertaines ordonnées suivant une dimension temporelle.

$$T = \hat{T} \pm \delta T = \{t_1 \pm \delta t_1, t_2 \pm \delta t_2, ..., t_m \pm \delta t_m\} \tag{8}$$

Une *sous séquence incertaine* S d'une série temporelle incertaine T est une séquence de l (sa taille) valeurs consecutives dans T.

$$S = \hat{S} \pm \delta S = \{t_{i+1} \pm \delta t_{i+1}, ..., t_{i+l} \pm \delta t_{i+l}\} \tag{9}$$

, où $1 \leq i \leq m - l, \ 1 \leq l \leq m$ et m la longueur de T

La *dissimilarité entre deux sous séquences incertaines* S et R est donnée par UED

$$d = \mathrm{UED}(S, R) = \mathrm{UED}(R, S). \tag{10}$$

Et la dissimilarité entre une série temporelle incertaine T et une sous séquence quelconque S de longueur inférieure ou égale à la longueur de T est définie comme suit :

$$\mathrm{UED}(T, S) = \min\{\mathrm{UED}(S, R) \mid \forall R \subset T, |S| = |R|\} \tag{11}$$

Un *séparateur incertain* sp pour un jeu de données D de séries temporelles incertaines est une sous séquence incertaine qui divise D en deux sous ensembles D_1 et D_2 tels que :

$$\begin{aligned} D_1 &= \{T \mid \mathrm{UED}(T, sp) \leq \epsilon\,, \forall T \in D\} \\ D_2 &= \{T \mid \mathrm{UED}(T, sp) > \epsilon\,, \forall T \in D\} \end{aligned} \tag{12}$$

La qualité d'un séparateur est mesurée en utilisant le gain d'information (IG). Étant données les définitions précédentes, nous pouvons définir ce qu'est un *shapelet incertain*. Pour un jeu de données D, un shapelet incertain S est un séparateur incertain qui maximise le gain d'information

$$S = \underset{sp}{\mathrm{argmax}}(IG(D, sp)) \tag{13}$$

L'algorithme de tranformation shapelet est décrite en détail dans [10]. Nous donnons ici un résumé de cet algorithme tout en mettant l'accent sur les changements lorsqu'on est dans le contexte des séries temporelles incertaines.
Étant donné un jeu de données D de séries temporelles incertaines, la première étape consiste à sélectionner parmi toutes les sous séquences de D les k shapelets de gain d'information maximal. Cette étape est effectuée par l'algorithme 1 qui prend en entrée le jeu de données, le nombre de shapelets incertains à extraire, les longueurs minimale et maximale d'un shapelet incertain. La longueur minimale est au moins égale à 3 et la longueur maximale est celle de la plus longue série dans le jeu de données. Cependant, une connaissance du domaine d'application peut aider à mieux fixer ces paramètres. L'algorithme utilise trois sous procédures :

- $GenCand(T, MIN, MAX)$ qui génère toutes les sous séquences incertaines contenues dans la série temporelle T. Ici, seules les sous séquences de longueur comprise entre MIN et MAX sont générées.
- $EvaluerCand(cands, D)$ qui évalue la qualité de séparation de chacune des sous séquences générées. Pour chaque sous séquence dans $cands$, le gain d'information obtenu en l'utilisant comme séparateur pour D est calculé.
- $ExtraireTop(C, Q, k)$ prend la liste des shapelets candidats C, leur gain d'information ou qualité Q et retourne les k shapelets incertains dont les qualités sont les plus élevées.

En résumé, l'algorithme 1 génère toutes les sous séquences de longueur au moins égale à MIN et au plus égale à MAX à partir du jeu de données, mesure la qualité de chacune de ces sous séquences en tant que séparateur en utilisant le gain d'information et finalement renvoie les k sous séquences ayant les meilleures qualités.

Algorithme 1 : Sélection des top-k Shapelets Incertains

Données : D, k, MIN, MAX
Résultat : k meilleurs shapelets incertains
début
  $C \leftarrow \emptyset$; $Q \leftarrow \emptyset$;
  pour $i \leftarrow 1, n$ **faire**
    $cands \leftarrow GenCand(T_i, MIN, MAX)$;
    $qualities \leftarrow EvaluerCand(cands, D)$;
    $C \leftarrow C \cup cands$;
    $Q \leftarrow Q \cup qualities$;
  fin
  $S \leftarrow ExtraireTop(C, Q, k)$;
  retourner S
fin

La prochaine étape après la sélection des k shapelets incertains est la transformation shapelet incertain. Il s'agit de la transformation shapelet telle que décrite par [10], mais à la différence que l'incertitude est propagée tout au long du processus de transformation. Cette étape est faite par l'algorithme 2 qui a pour entrée le jeu de données D, l'ensemble des k meilleurs shapelets S ainsi que sa taille k. Pour chaque série temporelle incertaine dans le jeu de donnée, son vecteur caractéristique de longeur k est calculé en utilisant UED. Le i-ème élément de ce vecteur est la distance euclidienne incertaine entre la série temporelle incertaine et le i-ème shapelet incertain dans S. L'ensemble des vecteurs caractéristiques forme le jeu de données transformé et est renvoyé par la procédure.

Algorithme 2 : Transformation Shapelet Incertain

Données : D, S, k
Résultat : Le jeu de données transformées
début
  pour $i \leftarrow 1, n$ **faire**
    $temp \leftarrow \emptyset$;
    pour $j \leftarrow 1, k$ **faire**
      $temp_j \leftarrow UED(T_i, S_j)$
    fin
    $D_i \leftarrow temp_j$
  fin
  retourner D
fin

La troisième et dernière étape est la classification proprement dite. Un algorithme de classification supervisée est entraîné sur le jeu de données transformé, de sorte qu'étant donné le vecteur caractéristique d'une nouvelle série temporelle, sa classe puisse être prédite. Vu que l'incertitude a été propagée, elle doit être prise en compte dans le processus d'apprentissage. L'algorithme d'apprentissage a donc comme entrées les meilleurs estimations (les caractéristiques) et l'incertitude sur ces estimations (métacaractéristiques). Si au lieu de UED, les algorithmes 1 et 2 utilisent l'une des mesures de dissimilarité/similarité de la littérature (DUST, MUNICH, PROUD ou FOTS), le modèle de classification ne pourrait être avisé de l'incertitude dans les données car aucune de ces métriques ne donne l'incertitude qu'il y a sur son résultat, ce qui est problématique sachant que nos données ont de l'incertitude.

La figure 1 donne un aperçu global du processus de classification de séries temporelles incertaines. Pendant la phase d'apprentissage, les k meilleurs shapelets sont sélectionnés puis un modèle de classification supervisée (illustré ici par un arbre de décision pour sa simplicité) est entraîné sur le jeu de données transformées. À chaque noeud de l'arbre de décision une fois entraîné est associé un shapelet incertain et la branche suivie par une série temporelle dans l'arbre est fonction de la similarité (calculée avec UED) entre elle et les shapelets associés à chacun des noeuds de ladite branche. Durant la phase d'inférence, les shapelets incertains extraits lors de la phase d'apprentissage sont utilisés pour transformer les données de test, et l'arbre de décision précédemment entraîné est utilisé pour prédire les différentes classes. Appelons ce processus de classification par transformation shapelet incertain *UST*.

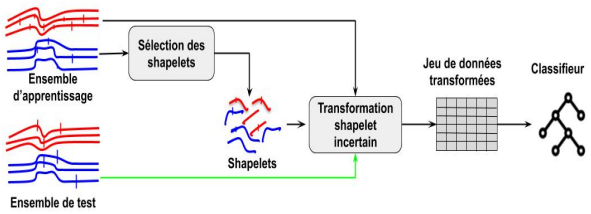


FIGURE 1 – Processus de classification des séries temporelles incertaines

5 Expérimentations

Dans cette section, nous évaluons expérimentalement notre approche de classification de séries temporelles incertaines et la comparons avec ce qui existe dans la littérature. Tout comme il est généralement le cas dans la littérature, le critère de comparaison pour les différents modèles est le taux d'exactitude (ou de généralisation) qui mesure le taux de bonnes prédictions sur l'ensemble de test [3, 6, 9, 10]. Étant donné que les modèles produisent en sortie la distribution de probabilité des instances en entrée par rapport aux différentes classes, nous utilisons la classe la plus probable comme la classe prédite, et c'est elle qui est prise en compte dans l'évaluation des modèles. Nous avons comparé les 4 modèles suivants :

- **UST_FLAT** : il s'agit de l'algorithme décrit dans la section 4. Les vecteurs caractéristiques sont représentés comme des vecteurs plats dont la première moitié contient les estimations optimistes, et la deuxième moitié contient les incertitudes sur les estimations. Ce modèle utilise l'ordre simple des mesures incertaines.
- **UST_FLAT_ST** : ce modèle est pareil que UST_FLAT, mais à la seule différence qu'il utilise l'ordre stochastique pour trier les mesures incertaines. Dans ce modèle, une mesure incertaine $x = \hat{x} \pm \delta x$ est considérée comme une variable aléatoire distribuée suivant une loi normale de moyenne $\hat{x}$ et d'écart-type δx. La fonction de répartition d'une telle variable aléatoire est :

$$CDF_X(t) = \frac{1}{2}(1 + erf(\frac{t - \hat{x}}{\delta x\sqrt{2}}))$$

  $erf(\cdot)$ est la fonction d'erreur de Gauss et nous avons utilisé l'implémentation de Apache Commons Math [1]. Afin de discrétiser $\mathbb{I}$ (en utilisant Eq. 7), nous avons fixé la valeur de k à 100. Plus k est grand, mieux l'intervalle $\mathbb{I}$ est approché, mais une valeur de k trop grande ralentit le processus de classification. Nous avons essayé plusieurs valeurs de k, et les meilleurs résultats pour nos données sont obtenus avec $k = 100$. Nous avons aussi utilisé une version relaxée de l'ordre stochastique : étant données deux variables aléatoires X et Y, nous considérons $X \leq Y$ si le nombre de valeurs t dans $\mathbb{I}$ pour lesquelles $CDF_X(t) > CDF_Y(t)$ est plus grand que le nombre de valeurs t' dans $\mathbb{I}$ pour lesquelles $CDF_X(t') \leq CDF_Y(t')$
- **DUST_UNIFORM** : Il s'agit ici de l'algorithme UST dans lequel UED a été remplacée par la version uniforme de DUST. Chaque série temporelle est supposée suivre une distribution uniforme. Ainsi il n'y a pas de propagation de l'incertitude et le modèle de classification n'est pas au courant de l'incertitude dans les données. Pour comparer deux mesures incertaines, DUST requiert qu'elles aient la même incertitude. Nous avons utilisé l'incertitude de la valeur la plus incertaine comme incertitude dans le calcul de DUST. Ceci permet d'inclure les deux incertitudes, bien que cela rend encore plus incertaine la valeur qui a moins d'incertitude.
- **DUST_NORMAL** : tout comme DUST_UNIFORM, mais suppose que chaque série temporelle est distribuée suivant une loi normale.

Bien que des modèles d'apprentissage supervisé tels que les machines à vecteur de support, les forêts aléatoires et les réseaux de neurones peuvent être utilisés pour augmenter le taux d'exactitude (TE), nous avons choisi d'utiliser un arbre de décision (implémentation J48) comme modèle de classification dans chacun de ces quatre modèles. Nous souhaitons que le TE soit corrélé à la mesure de dissimilarité utilisée et non au modèle de classification. Ceci permet de mettre en évidence l'importance de la prise en compte de l'incertitude.

5.1 Datasets

Nous avons expérimenté sur 29 datasets provenant de UCR [4]. La première colonne du tableau 1 est la liste de nos jeux de données. Bien que UCR contient des séries temporelles univariées et multivariées, nous nous limitons aux datasets univariés. Cependant l'approche par transformation shapelet est aussi applicable sur des datasets multivariés, et aussi sur des datasets avec des séries temporelles de longueur

1. https://commons.apache.org/proper/commons-math/

variable. Chaque dataset du dépôt UCR est déjà divisé en ensemble de test et d'apprentissage.

Les datasets qui sont sur UCR ne contiennent par d'incertitude. Nous avons manuellement ajouté l'incertitude dans nos datasets. Pour chaque dataset, l'incertitude ajoutée suit une distribution normale de moyenne 0 et de déviation standard $c \times \sigma$, où σ est l'écart-type du jeu de données et c est un paramètre qui nous permet de contrôler la grandeur de l'incertitude ajoutée. Nous avons utilisé deux valeurs de c qui sont 0.1 et 0.2. L'ajout de l'incertitude dans le dataset CBF est illustré par la figure 2 pour une instance. La ligne orange est la série temporelle originale ne contenant pas d'incertitude. La ligne bleue est la série temporelle obtenue après ajout de l'incertitude. Pendant la phase d'apprentissage, la série temporelle originale n'est pas utilisée ; seule la série temporelle incertaine est utilisée. Chaque valeur dans un jeu de données a sa propre incertitude qui est l'écart-type de la loi normale qui a généré cette incertitude.

5.2 Code source

Nous avons utilisé un code existant comme base pour notre implémentation. Il s'agit d'un projet open source contenant l'implémentation d'une panoplie de méthodes de classification de séries temporelles parmi lesquelles la transformation shapelet. Ce projet est développé par [3] et est hébergé sur la plateforme Github [2]. Il est écrit en langage Java. Nous y avons ajouté notre implémentation de UED et UST et avons distribué le code via Github [3]. Le script Python utilisé pour ajouter l'incertitude est également disponible sur sur le même dépôt [4].

5.3 Résultats et discussion

Le taux d'exactitude (TE) de chacun de nos 4 modèles a été mesuré et est donné dans le tableau 1. Pour chaque jeu de données dans le tableau, nous avons le TE obtenu par chaque modèle pour chacun des deux niveaux d'incertitude.

Focalisons nous premièrement sur les trois premiers modèles du tableau 1, il s'agit de UST_FLAT, DUST_UNIFORM et DUST_NORMAL. Nous résumons leur performance en utilisant les boites à moustaches (Boxplot) comme on peut le voir sur la figure 3. Le triangle vert représente la moyenne des TE sur l'ensemble des jeux de données. Les méthodes basées sur DUST (DUST_UNIFORM and DUST_NORMAL) ne sont pas significativement différentes l'une de l'autre. Cependant, DUST_UNIFORM est légèrement meilleur que DUST_NORMAL. UST_FLAT est meilleur que les modèles utilisant DUST, surtout lorsque l'incertitude est élevée (figure 3b). Le troisième quantile des methodes basées sur DUST est seulement à 0.6, tandis qu'il est à environ 0.8 pour la méthode UST. Ce qui signifie que UST obtient 80% de TE sur 25% des jeux de données. UST est meilleur sur plus de datasets que les méthodes utilisant DUST. En particulier, pour $c = 0.2$, le meilleur TE pour les modèles DUST est de 55% et 54% pour le dataset DodgerLoopGame et BME respectivement ; alors que UST donne 77% de TE pour DodgerLoopGame, et 82% pour BME.

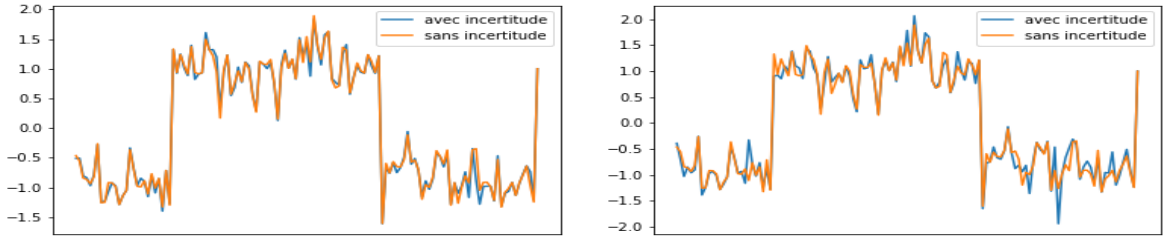
Examinons ensuite l'impact de la relation d'ordre utilisée par UST en comparant UST_FLAT et UST_FLAT_ST. Étant donné que cet ordre n'est pas total, UST_FLAT_ST ne s'est pas exécuté jusqu'au bout sur les 23 datasets ; Il y a eu des valeurs non comparables sur ces jeux de données soit lors de la phase d'apprentissage, soit lors de la phase de test. Les résultats obtenus sur les 6 jeux de données restant sont résumés dans la figure 4. L'ordre stochastique a amélioré les résultats pour $c = 0.1$ (figure 4a) ; En particulier, le TE est passé de 61% à 84% (une augmentation de 23%) pour le dataset Chinatown, et de 14% à 35% (une augmentation de 21%, mais pas assez pour passer les 50% de TE) sur le dataset SonyAIBORobotSurface1. Lorsque $c = 0.2$ (figure 4b), nous n'observons aucune amélioration, par contre il y a une baisse du TE sur le dataset BME qui passe de 82% à 69% (une baisse de 13%) lorsqu'on utilise l'ordre stochastique.

Avec UED, nous obtenons des résultats similaires, et même meilleurs qu'avec les mesures incertaines de la littérature. Cependant, notre approche a des limites. En effet, en utilisant une représentation applatie, le modèle de classification n'est pas réellement avisé de l'incertitude qui a été propagée. Bien que l'incertude soit une méta-caractéristique, elle est considérée par le modèle de classification supervisée comme une caractéristique à part entière. Nous pensons qu'une meilleure façon de prendre en compte l'incertitude propagée mènerait à de meilleures classifications. En particulier, un arbre de décision flou [16] pourrait être un bon candidat. Il permet à une instance de traverser chaque branche de l'arbre avec une probabilité. À cause de l'incertitude, il est intuitif qu'une instance puisse passer par plusieurs branches de l'arbre avec des probabilités non nulles. Ainsi, nous comptons explorer la possibilité de modéliser le problème de classification des séries temporelles incertaines en utilisant la logique floue [2, 12, 15].

6 Conclusion

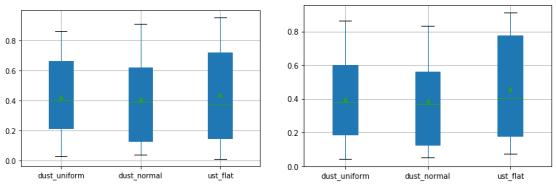
Le but de ce papier était de faire la classification des séries temporelles incertaines en utilisant l'approche par transformation shapelet. Pour le faire, nous avons utilisé les techniques de propagation de l'incertitude afin de dériver une mesure de dissimilarité incertaine appelée UED. Ensuite, nous avons adapté la méthode de classification par transformation shapelet au contexte des séries temporelles incertaines en utilisant UED et avons finalement proposé l'algorithme de classification par transformation shapelet incertain (UST). Nous avons effectué des expérimentations sur des jeux de données de la littérature et les résultats montrent l'efficacité de notre approche. En perspective, nous avons l'intention d'évaluer notre approche sur un vrai jeu de données contenant l'incertitude. Nous visons par

2. https://github.com/uea-machine-learning/tsml
3. https://github.com/frankl1/Uncertain-Shapelet-Transform
4. https://github.com/frankl1/Uncertain-Shapelet-Transform/blob/master/add-noise.ipynb



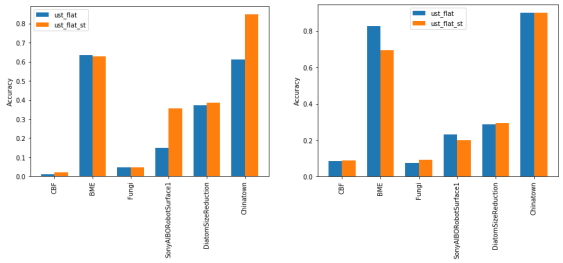
(a) $c = 0.1$ (b) $c = 0.2$

FIGURE 2 – Illustration de l'incertitude pour une instance du dataset CBF



(a) $c = 0.1$ (b) $c = 0.2$

FIGURE 3 – DUST vs UST



(a) $c = 0.1$ (b) $c = 0.2$

FIGURE 4 – UST_FLAT vs UST_FLAT_ST

exemple le dataset du challenge Plasticc [5]. Nous explorerons également la possibilité d'utiliser un modèle de classification flou afin de prendre mieux en compte l'incertitude après la propagation.

Remerciements

Ce travail a été soutenu par le Ministère français de l'Enseignement Supérieur, de la Recherche et de l'Innovation (MESRI), par le LabEx IMobS3 et le projet CNRS PEPS TransiXplore. Nous remercions également les relecteurs annonymes pour leurs remarques constructives.

Références

[1] Johannes Aßfalg, Hans-Peter Kriegel, Peer Kröger, and Matthias Renz. Probabilistic similarity search for uncertain time series. In *International Conference on Scientific and Statistical Database Management*, pages 435–443. Springer, 2009.

[2] Jamal Atif, Isabelle Bloch, and Céline Hudelot. Some relationships between fuzzy sets, mathematical morphology, rough sets, f-transforms, and formal concept analysis. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 24(Supplement-2) :1–32, 2016.

[3] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off : a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31(3) :606–660, 2017.

[4] Anthony Bagnall, Jason Lines, William Vickers, and Eamonn Keogh. The uea & ucr time series classification repository. Accessed on 14/01/2020.

[5] Isabelle Bloch. Model-based image interpretation under uncertainty and fuzziness. In Francesco Masulli, Gabriella Pasi, and Ronald Yager, editors, *Fuzzy Logic and Applications*, pages 171–183, Cham, 2013. Springer International Publishing.

[6] Michele Dallachiesa, Besmira Nushi, Katsiaryna Mirylenka, and Themis Palpanas. Uncertain time-series similarity : Return to the basics. *Proceedings of the VLDB Endowment*, 5(11) :1662–1673, 2012.

[7] Florence Dupin de Saint-Cyr and Henri Prade. Logical handling of uncertain, ontology-based, spatial information. *Fuzzy Sets Syst.*, 159(12) :1515–1534, 2008.

[8] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Deep learning for time series classification : a review. *Data Mining and Knowledge Discovery*, 33(4) :917–963, 2019.

[9] Vanel Steve Siyou Fotso, Engelbert Mephu Nguifo, and Philippe Vaslin. Frobenius correlation based u-shapelets discovery for time series clustering. *Pattern Recognition*, 103 :107301, 2020.

[10] Jon Hills, Jason Lines, Edgaras Baranauskas, James Mapp, and Anthony Bagnall. Classification of time series by shapelet transformation. *Data Mining and Knowledge Discovery*, 28(4) :851–881, 2014.

5. https://www.kaggle.com/c/PLAsTiCC-2018

TABLE 1 – Précision de chaque modèle sur chaque dataset et pour chaque niveau d'incertitude

Dataset	**DUST_UNIFORM**		**DUST_NORMAL**		**UST_FLAT**		**UST_FLAT_ST**	
	c = 0.1	**c = 0.2**	**c = 0.1**	**c = 0.2**	**c = 0.1**	**c = 0.2**	**c = 0.1**	**c = 0.2**
ArrowHead	0.58	0.57	0.58	0.56	0.51	0.61	-	-
BME	0.62	0.54	0.62	0.54	0.63	0.82	0.62	0.69
CBF	0.06	0.11	0.06	0.11	0.01	0.08	0.02	0.08
Chinatown	0.84	0.76	0.89	0.78	0.61	0.90	0.84	0.90
DiatomSizeReduction	0.26	0.27	0.26	0.27	0.37	0.28	0.38	0.29
DistalPhalanxTW	0.10	0.11	0.10	0.12	0.04	0.10	-	-
DodgerLoopGame	0.61	0.55	0.52	0.53	0.71	0.77	-	-
DodgerLoopWeekend	0.75	0.60	0.90	0.83	0.72	0.63	-	-
ECGFiveDays	0.12	0.18	0.12	0.10	0.17	0.22	-	-
ECG200	0.23	0.20	0.23	0.24	0.19	0.18	-	-
Fungi	0.03	0.04	0.03	0.05	0.04	0.07	0.04	0.09
GunPoint	0.12	0.08	0.12	0.08	0.08	0.12	-	-
GunPointOldVersusYoung	0.86	086	0.52	0.52	0.94	0.91	-	-
InsectEPGSmallTrain	0.41	0.30	0.37	0.45	0.31	0.37	-	-
ItalyPowerDemand	0.21	0.12	0.21	0.12	0.08	0.10	-	-
MedicalImages	0.38	0.38	0.40	0.35	0.42	0.40	-	-
MelbournePedestrian	0.40	0.28	0.12	0.10	0.70	0.64	-	-
MiddlePhalanxOutlineAgeGroup	0.40	0.40	0.42	0.40	0.34	0.45	-	-
MiddlePhalanxOutlineCorrect	0.41	0.38	0.41	0.43	0.34	0.36	-	-
MiddlePhalanxTW	0.38	0.40	0.38	0.37	0.40	0.41	-	-
MoteStrain	0.83	0.69	0.83	0.62	0.80	0.79	-	-
Plane	0.29	0.31	0.29	0.28	0.28	0.26	-	-
ProximalPhalanxOutlineAgeGroup	0.12	0.12	0.12	0.13	0.12	0.16	-	-
ProximalPhalanxTW	0.07	0.08	0.06	0.07	0.08	0.08	-	-
SmoothSubspace	0.72	0.71	0.73	0.70	0.84	0.73	-	-
SonyAIBORobotSurface1	0.21	0.22	0.21	0.23	0.14	0.23	0.35	0.19
SyntheticControl	0.76	0.78	0.78	0.81	0.90	0.90	-	-
TwoLeadECG	0.75	0.71	0.78	0.70	0.86	0.79	-	-
UMD	0.65	0.72	0.65	0.72	0.90	0.85	-	-

[11] Vassilis G Kaburlasos, Eleni Vrochidou, Fotios Panagiotopoulos, Charalampos Aitsidis, and Alexander Jaki. Time series classification in cyber-physical system applications by intervals' numbers techniques. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6. IEEE, 2019.

[12] Le Xu, Mo-Yuen Chow, and L. S. Taylor. Data mining based fuzzy classification algorithm for imbalanced data. In *2006 IEEE International Conference on Fuzzy Systems*, pages 825–830, July 2006.

[13] Jason Lines, Sarah Taylor, and Anthony Bagnall. Time series classification with hive-cote : The hierarchical vote collective of transformation-based ensembles. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(5) :52, 2018.

[14] Albert W. Marshall, Ingram Olkin, and Barry C. Arnold. *Stochastic Ordering*, chapter 17, pages 693–756. Springer New York, New York, NY, 2010.

[15] P. Martín-Muñoz and F. J. Moreno-Velo. Fuzzycn2 : An algorithm for extracting fuzzy classification rule lists. In *International Conference on Fuzzy Systems*, pages 1–7, July 2010.

[16] Cristina Olaru and Louis Wehenkel. A complete fuzzy decision tree technique. *Fuzzy Sets and Systems*, 138(2) :221 – 254, 2003.

[17] Smruti R Sarangi and Karin Murthy. Dust : a generalized notion of similarity between uncertain time series. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 383–392. ACM, 2010.

[18] John R. Taylor. *An Introduction to Error Analysis : The Study of Uncertainties in Physical Measurements*. University Science Books, 2 sub edition, 1996.

[19] Lexiang Ye and Eamonn Keogh. Time series shapelets : a new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 947–956. ACM, 2009.

[20] Mi-Yen Yeh, Kun-Lung Wu, Philip S Yu, and Ming-Syan Chen. Proud : a probabilistic approach to processing similarity queries over uncertain data streams. In *Proceedings of the 12th International Conference on Extending Database Technology : Advances in Database Technology*, pages 684–695. ACM, 2009.