TS-Relax : Interprétation des représentations apprises pour les séries temporelles

A M Jedidi, T Blanchard, A Bonnefoy

TS-Relax : Interprétation des représentations apprises pour les séries temporelles.

A. M. Jedidi[1], T. Blanchard[2], A. Bonnefoy[1]

[1] EURA NOVA
[2] Ecole Centrale Marseille

antoine.bonnefoy@euranova.eu, aziz.jedidi@euranova.eu, thomas.blanchard@centrale-marseille.fr

Résumé

Les modèles d'apprentissage de représentations sont de plus en plus utilisés, mais leur opacité peut être problématique dans des domaines tels que la santé ou la justice. Des modèles d'IA explicables et de confiance sont nécessaires. Ce travail présente l'adaptation aux séries temporelles, d'une méthode d'interprétation de représentation initialement conçue pour les images. Nous proposons un protocole quantitatif pour évaluer la pertinence de cette adaptation. Les résultats préliminaires encourageants nous amènent à envisager des perspectives de recherches sur la spécifité de l'interprétabilité des modèles d'apprentissage de représentations sur les séries temporelles.

Mots-clés

Interprétabilité, Time Series, Représentations, Explicabilité, encodeur.

Abstract

Representation learning models are increasingly used, but their opacity can be problematic in domains such as healthcare or justice. Explainable and trustworthy AI models are needed. This work presents the adaptation to time series of a representation learning explainability method initially designed for images. We propose a quantitative protocol to evaluate the relevance of this adaptation. The encouraging preliminary results lead us to consider research perspectives on the specificity of the interpretability of representation learning models for time series.

Keywords

Interpretability, Time Series, Representations, Explainability, encoder.

1 Introduction

L'analyse des séries temporelles (Time Series) a pris une place de choix dans l'analyse de données, notamment grâce à l'avancée des techniques de Machine Learning (ML) et de Deep Learning (DL). Ces données, capturées à intervalles réguliers ou non sur une période de temps, peuvent révéler des informations précieuses sur les tendances, les cycles, les événements et les modèles qui se produisent dans le temps. Leur exploitation permet de prédire des comportements ou des événements futurs dans des domaines variés tels que la météorologie, la physique, l'économie, la finance, la santé, entre autres. Cependant, l'analyse de ces séries temporelles peut s'avérer complexe, à cause de la dimensionnalité élevée, du bruit et de la complexité intrinsèque des données.

Et alors que le Deep Learning a montré son efficacité dans l'analyse des données séquentielles, l'interprétabilité des modèles qu'il produit demeure un défi majeur. En effet, les réseaux de neurones, malgré leur efficacité dans la résolution de problèmes complexes comme la reconnaissance d'images, la traduction automatique ou la prédiction de séries temporelles, sont souvent perçus comme des "boîtes noires". Il est donc crucial de développer des méthodes permettant de comprendre les caractéristiques des données qui ont été utilisées pour prendre une décision et ainsi assurer une meilleure adoption des modèles de ML et DL.

Par ailleurs, l'utilisation croissante de représentations apprises ajoute une couche d'opacité supplémentaire à ces modèles. Ces représentations, qui transforment les données complexes en formats plus exploitables pour le modèle tout en conservant l'essentiel des informations, peuvent toutefois devenir incompréhensibles pour l'humain. Bien que ces encodages efficaces aient démontré leur efficacité dans des domaines tels que le traitement du langage naturel et la vision par ordinateur, ils posent un défi pour notre compréhension des données.

Si nous parvenons à expliquer pourquoi certains encodeurs efficaces encodent de cette manière spécifique, nous pourrons mieux comprendre les données en tant qu'êtres humains. En comprenant les facteurs et les mécanismes qui influencent la représentation apprise des séries temporelles, nous pourrions obtenir des informations précieuses sur la nature des données elles-mêmes. Cette compréhension accrue pourrait nous aider à interpréter les résultats des modèles d'intelligence artificielle, à détecter les biais potentiels et à développer des méthodes d'explication plus robustes pour ces modèles de représentation apprise. Par conséquent, la recherche visant à élucider ces encodages opaques est essentielle pour progresser dans notre compréhension des modèles d'apprentissage automatique et de leurs applications.

Malgré cela, et suite à une revue de la littérature extensive que nous avons menée, nous constatons que les séries tem-

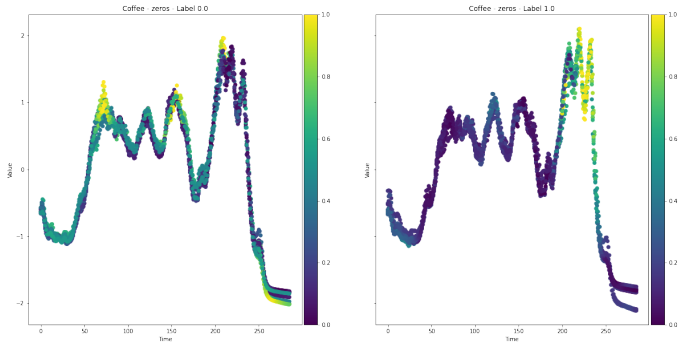


FIGURE 1 – Importances obtenues par TS-RELAX pour le dataset Coffee [2]. À gauche la classe 1 et à droite la classe 2. Bien que ces importances aient été obtenues de façon non supervisée on observe que la méthode a été capable d'identifier les parties discriminantes des séries temporelles pour une tache de classification aval.

porelles ne semblent pas être largement discutées dans la communauté académique en ce qui concerne l'interprétabilité des représentations apprises (Embeddings, encodages ...), plus spécifiquement sous l'angle de l'interprétation locale. L'adaptation de ces méthodes à l'explication d'un encodage, ou à une représentation agnostique de la tâche, apparaît donc comme un domaine qui reste relativement peu exploré dans les écrits académiques.

Dans ce contexte, notre travail vise à combler ce manque en proposant une approche pour l'interprétabilité des représentations de séries temporelles basée sur l'adaptation de la méthode RELAX, initialement développée pour l'interprétation de représentations d'images. Nous avons ainsi développé TS-Relax, une méthode qui permet d'attribuer des scores d'importance aux échantillons (timestamp) d'une série temporelle. Les résultats obtenus grâce à notre protocole expérimental sont prometteurs et contribuent à une meilleure compréhension des prédictions effectuées par les modèles de séries temporelles.

Dans cet article, nous commencerons par présenter les travaux existants qui ont servi de base à notre adaptation dans la section 2. Ensuite, nous détaillerons l'adaptation de la méthode RELAX pour les séries temporelles dans la Section 3. Nous discuterons également de notre protocole d'évaluation, conçu pour évaluer la méthode TS-RELAX en l'absence de méthodes de référence pour notre tâche, dans la Section 4. Enfin, nous partagerons l'analyse de nos résultats et discuterons des pistes de recherche futures dans la Section 5.

Notre travail contribue à une meilleure compréhension des prédictions des modèles de séries temporelles et ouvre la voie à de nouvelles recherches pour rendre ces modèles plus interprétables et, par conséquent, plus accessibles et acceptables pour un public plus large

2 Travaux connexes à l'interprétabilité sur séries temporelles

L'interprétation locale désigne la compréhension des prédictions d'un modèle de manière locale, en identifiant les caractéristiques des données d'entrée qui ont le plus d'impact sur la sortie du modèle. Et pour notre travail, nous nous sommes appuyés sur quelques articles précis, comme RISE [6], qui est considéré comme l'un des représentants emblématiques des approches par perturbations. La méthode RISE a pour but de donner une interprétation à des modèles boites noires, dont les réseaux de neurones. Cette méthode repose sur la perturbation aléatoire des entrées du modèle pour identifier les caractéristiques les plus importantes qui influencent les prédictions du modèle. Pour cela, la méthode s'axe sur la génération de masques aléatoire. L'image est occultée suivant ces masques, puisque la perturbation dans cet article est une multiplication terme à terme entre l'image et le masque M. Ne sont alors conservés à l'identique que les pixels qui correspondent aux éléments fixés à 1 dans le masque. L'importance relative de chacun des pixels est ensuite agrégée à partir des scores du modèle pour chaque image masquée issue des multiples masques générés.

La méthode RISE a inspiré d'autres travaux, notamment la méthode RELAX, expliquée dans l'article : *"RELAX : Representation Learning Explainability"* [7], qui étend l'approche de RISE aux modèles de représentations qui ne présupposent pas de tâches particulières, ajoutant ainsi la notion d'importance absolue dans l'image, indépendante de la tâche d'aval. En se basant sur le postulat que la représentation doit changer significativement si les zones les plus importantes sont masquées, la méthode RELAX calcule la similarité entre les représentations des images perturbées et initiales. Plus amplement détaillée dans la section suivante, cette méthode nous semblait intéressante pour plusieurs raisons, à savoir ses résultats, apportant une solution simple à un problème original, ainsi que son adaptabilité affichée à des données autres que des images, puisque basée sur l'usage de représentations pré-apprise.

Enfin, il est évident que la qualité de la représentation est également un facteur critique dans l'interprétation des données de séries temporelles. Une bonne représentation permettra de mieux comprendre les caractéristiques importantes des données et de capturer les motifs temporels importants. À l'inverse, une mauvaise représentation peut entraîner des erreurs de prévision et des modèles peu interprétables. Par conséquent, il est crucial pour nous de nous appuyer sur de bonnes représentations, qui n'abaisseront pas le niveau de prédiction du modèle. Nous avons alors choisi de nous baser sur les représentations obtenues dans [3], puisque l'article offre des représentations polyvalentes pour des séries temporelles de longueur variable et multivariées, dont la qualité est prouvée, et obtenues au travers d'une méthode non-supervisée qui se base sur l'utilisation d'un encodeur basé sur des convolutions dilatées causales avec une nouvelle triplet loss utilisant un échantillonnage

négatif basé sur le temps.

3 RELAX appliqué aux séries temporelles

Dans cette partie, nous allons présenter la méthode RELAX, qui a été initialement conçue pour les images, mais que nous avons cherché à adapter aux séries temporelles pour pouvoir apporter un approfondi des données sans a priori sur la tache aval. Ainsi, dans un premier temps, nous présenterons la méthode RELAX, telle qu'elle est décrite dans l'article original, avant de détailler l'adaptation que nous en avons faîte.

3.1 Présentation de la méthode RELAX

La méthode RELAX, conceptualisée pour des images, présente plusieurs fonctions majeures, que nous allons détailler ici, avant d'approfondir sur les résultats que nous avons obtenus.

Cette méthode consiste à masquer aléatoirement certaines parties de la donnée d'entrée afin d'évaluer l'importance des parties masquées en évaluant l'impact sur la représentation de l'image masquée. Pour cela, le masque M est généré suivant la procédure décrite dans [6] il prend des valeurs entre 0 et 1 pour masquer certaines régions de l'image d'entrée X. Si l'on prend une variable d'entrée de taille $X \in \mathbb{R}^{H \times W}$, on génère une matrice stochastique de taille $H \times W$. Puis, on calcule à travers une fonction de représentation f, les représentations $h = f(X)$, et $\bar{h} = f(X \odot M)$. On obtient ainsi la représentation de l'image masquée $\bar{h}$ en multipliant l'image originale X par M (produit de Hadamard). La méthode de perturbation choisie pour cette méthode est donc l'occultation. La similarité entre h et $\bar{h}$ est enfin mesurée par la fonction $s(h, \bar{h})$. Si les parties masquées sont non-informatives, la similarité entre h et $\bar{h}$ est élevée, car la représentation est peu modifiée, alors que si des parties informatives sont masquées, la similarité est faible.

L'importance de chacun des pixels X_{ij} de l'image pour le modèle de représentation f est ainsi obtenue via l'équation suivante : $\bar{R}_{ij} = \frac{1}{N} \sum_{n=1}^{N} s(h, \bar{h}_n) M_{ij}(n)$

3.1.1 Choix des masques

L'adaptation de la fonction de génération de masques de la méthode RELAX aux Time Series s'est avérée relativement aisée. En effet, la principale différence entre les images et les séries temporelles se situe dans leur dimensionnalité. Cependant, cette différence n'a pas posé de problème majeur puisque RELAX, dans sa procédure de génération de masques, s'appuyait sur la création d'une grille en 2 dimensions plus petite que l'entrée, qu'ils redimensionnaient par la suite au travers d'une interpolation bilinéaire. Il nous a alors suffit de modifier la dimension de la grille initiale, pour qu'elle corresponde à la dimension de la série temporelle en entrée, et d'adapter l'interpolation finale. En pratique, cela signifie que la méthode de masquage pour les images est globalement la même que celle utilisée pour les séries temporelles.

Le calcul de l'importance de l'échantillon X_i d'une série temporelle est donc simplement obtenu par :

$$\bar{R}_i = \frac{1}{N} \sum_{n=1}^{N} s(h, \bar{h}_n) M_i(n) \tag{1}$$

3.1.2 Méthode de perturbation

Dans l'adaptation de la méthode RELAX aux séries temporelles, nous avons décidé de ne pas modifier la méthode de perturbation utilisée dans la version originale. Cependant, il est important de noter que cette décision pourrait avoir un impact significatif sur les résultats obtenus. En effet, la méthode de perturbation consiste en une multiplication de Hadamard entre le masque généré et la représentation de la série temporelle. Il est possible que d'autres méthodes de perturbation soient plus efficaces pour les séries temporelles, et cela pourrait être un sujet de recherche futur à approfondir. Néanmoins, pour cette étude, nous avons décidé de conserver la méthode de perturbation de la version originale de RELAX pour une comparaison plus directe avec les résultats précédents.

4 Expérimentations

Dans cette section, nous présentons les expériences que nous avons menées dans le but d'évaluer la qualité des interprétations produites par TS-Relax. Pour cela, nous avons choisi deux caractéristiques souhaitées pour nos explications.

Tout d'abord, nous avons besoin de nous assurer de la fidélité des scores d'importances générés par TS-Relax au modèle, et ce afin de pouvoir les utiliser en toute confiance. Ensuite, nous souhaitons également que ces explications soient accessibles et facilement compréhensibles pour l'œil humain. Le critère de simplicité nous apparaissait donc essentiel pour l'interprétabilité de ces attributions, et donc pour leur usage final. Ces caractéristiques désirables pour une explication peuvent être évaluées objectivement à l'aide des métriques existantes que nous avons sélectionnées et justifiées. Les expérimentations liées à l'évaluation de ces caractéristiques sont décrites dans la section suivante 4.1.

4.1 Évaluation quantitative

4.1.1 Fidélité (Monotonicity)

La première caractéristique souhaitée, la fidélité, ou "faithfulness" d'une explication, peut être évaluée par la monotonicité, comme définie par Nguyen et Al [5]. Pour cela, nous avons repris le postulat suivant des auteurs : "L'importance d'une caractéristique devrait être proportionnelle à l'imprécision de la prédiction si nous ne connaissions pas sa valeur". Autrement dit, l'imprécision causée par sa disparition ou sa mise à zéro. Nous détaillons cette métrique utilisée également pour évaluer la méthode RELAX originale [7] et qui est déjà implémentée par [4].

En considérant $\bar{R}_i$ le score d'importance de l'attribut i avec $i \in [1, N]$ pour une fonction prenant la valeur $y^* = f(\mathbf{x}^*)$ au point $\mathbf{x}^* = (x_1^*, \ldots, x_N^*)$, nous pouvons écrire le postulat suivant :

$$\left|\bar{R}_i\right| \propto \mathbb{E}\left(l\left(y^*, f_i\right) \mid \mathbf{x}^*_{-i}\right)$$

où f_i est la restriction de la fonction f à la caractéristique i obtenue en fixant toutes les autres caractéristiques aux valeurs $\mathbf{x}^*_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, x_N)$, et l est une mesure de performance (une fonction de perte (loss)).
La définition de la monotonicité pour les scores d'importances $\bar{R}_i$ correspond au coefficient de corrélation de Spearman $\rho_S(\bar{\mathbf{R}}, \mathbf{e})$, avec :
$\bar{\mathbf{R}} = \left(\ldots, \left|\bar{R}_i\right|, \ldots\right)$ un vecteur contenant les valeurs absolues des scores d'importances et $\mathbf{e} = \left(\ldots, \mathbb{E}\left(l\left(y^*, f_i\right); X_i \mid \mathbf{x}^*_{-i}\right), \ldots\right)$ les estimations des espérances des fonctions de pertes correspondantes.

4.1.2 Simplicité (Complexity)

Pour évaluer la simplicité et la minimalité des explications de notre modèle, nous reprenons le concept d'explication complexe, proposé par [1], basée sur une distribution de contribution fractionnelle définie comme suit, où $|\cdot|$ représente la valeur absolue :

$$\mathbb{P}_g(i) = \frac{|g(f,x)_i|}{\sum_{j\in[d]} |g(f,x)_j|}; \mathbb{P}_g = \{\mathbb{P}_g(1), \ldots, \mathbb{P}_g(d)\}$$

Avec $\mathbb{P}g$, une distribution de probabilité valide. Chaque caractéristique x_i de la série temporelle a une contribution fractionnelle $\mathbb{P}g(i)$ à la magnitude totale de l'attribution. À savoir que l'explication (bien que fidèle) est d'autant plus complexe que toutes les caractéristiques présentent une attribution égale. A contrario, l'explication la plus simple se concentre sur une seule caractéristique. On définit alors la complexité comme l'entropie de $\mathbb{P}g$.

Définition (Complexité). Étant donné une fonction de prédiction f, une fonction d'explication g (dans notre cas TS-Relax), et un point x, la complexité de g à x est :

$$\mu_C(f, g; x) = \mathbb{E}_i\left[-\ln\left(\mathbb{P}_g\right)\right] = -\sum_{i=1}^{d} \mathbb{P}_g(i) \ln\left(\mathbb{P}_g(i)\right)$$

4.1.3 Protocole d'évaluation de RELAX pour les séries temporelles

Dans cette section, nous comparons les scores des explications produites par notre méthode TS-Relax aux scores d'un grand nombre d'explications générées aléatoirement. En effet, il n'existe pas d'alternatives à notre modèle sur la tâche d'explicabilité des représentations de séries temporelles. Néanmoins, nous pouvons générer des scores d'importances de manière aléatoire à l'aide d'une loi uniforme.
Par la suite, nous comparons les scores de monotonie de l'explication produite par TS-Relax avec ceux de 10 explications "aléatoires" générées selon une loi normale. Cette expérience est réalisée 5 fois pour chacun des modèles et nous faisons la moyenne et le maximum des résultats. Nous avons utilisé deux modèles : (1) un Encodeur-Classifieur basé sur l'architecture de l'encodeur de [3], auquel ont été ajoutées trois couches denses pour effectuer de la classification et (2) un réseau dense de classification sans encodeur (FFN) pour évaluer l'universalité des attributions construites grâce à l'encodeur et à RELAX, afin de proposer une évaluation non biaisée par le modèle d'apprentissage de représentation. Les auteurs de [3] ont, par exemple, testé un encodeur appris sur un dataset "FordA" sur d'autres datasets et ont démontré qu'il restait assez performant. Nous avons voulu savoir si les caractéristiques temporelles les plus importantes pour notre encodeur l'étaient également pour un autre réseau. Les détails des architectures sont présentés en annexe de cet article.

Jeux de données Nous avons évalué notre méthode sur des jeux de données unidimensionnels, tous provenant du répertoire UCR [2]. Nous en avons choisi 10, de manière à avoir une variété de types (ECG, Image, Motion, Sensor, Simulated et Spectro). Le choix des jeux de données est très important pour rendre compte des résultats de notre méthode. Deux types de jeux de données ont été pris en compte : des jeux de données synthétiques(simulés), et des jeux de données réels. Pour les jeux de données synthétiques, nous les avons sélectionnés ceux pour lesquels nous connaissions leur formule de génération, et qui n'étaient pas bruités. Pour le choix des jeux de données réels, ce sont les résultats d'accuracy du classifieur (Encodeur+SVM) de [3] que nous avons pris en compte. Une liste exhaustive des jeux de données utilisés est disponible en annexe de cet article sous forme d'un tableau détaillant les résultats obtenus.
Néanmoins, nous n'avons pas pris de jeux de données pour lesquels le couple encodeur+SVM obtient de mauvaises performances de classification (voir table S1 de [3]). Cela fera l'objet d'une expérience future pour tenter de tirer profit de TS-Relax pour comprendre pourquoi le couple encodage+SVM n'est pas efficace pour certains jeux de données.
Pour chaque jeu de données, nous avons entraîné un classifieur sur un ensemble de données d'entraînement, puis évalué sa performance sur un ensemble de test distinct. À partir de cet ensemble de test, nous avons sélectionné aléatoirement 10 exemples de chaque classe et utilisé notre méthode pour produire des scores d'importance. Nous avons répété cette expérience 5 fois pour chaque classifieur. Nous avons ensuite comparé la moyenne des scores d'importance générés par TS-Relax à la moyenne, et au maximum, des 50 explications générées aléatoirement.
Il convient de souligner que nos explications sont totalement indépendantes du modèle de classification utilisé, car elles ne sont basées que sur l'encodeur.

Hyperparametres Pour cette expérience, nous avons défini une sous-grille unidimensionnelle (num_cells) de taille 1/10 la longueur des timeseries du jeux de données ou nous avons choisi une probabilité p=0.5 que la valeur d'un timestamp soit totalement masqué (et donc mise à 0), Cette sous grille sera par la suite interpolée pour créer un masque de même taille que la série temporelle. L'interpolation utilisée est la même que le papier relax original. Fixer le paramètre p à 0.5 implique que les valeurs de 50% des timestamps

seront conservés telles quelles. Quant aux 50% restantes, elles seront soit mise à 0, soit réduite par le facteur d'interpolation (entre 0 et 1 donc).

Enfin, nous utilisons 3000 masques en tout par exemple, cela se fait en 30 batches de 100 masques. La division en batches permet un gain en temps d'exécution d'après les auteurs de l'article Relax pour les images.

Implémentation Le code du papier original "Relax" étant public, nous avons pu en reprendre l'essentiel pour notre adaptation. Quant aux métriques utilisées, nous avons utilisé le paquet python "Quantus" [4], qui implémente plusieurs métriques du domaine de l'explicabilité (XAI).

Model	Relax (Moy)	Random (Moy)	Random (Max)
ENC	17.20 ± 1.91	-0.041 ± 0.02	5.72 ± 0.07
FFN	18.08 ± 2.16	-0.042 ± 0.02	8.66 ± 0.24

TABLE 1 – Score de monotonicité moyen (en $\%$) des attributions générées par Relax et attributions générées aléatoirement (moyenne et maximum) pour deux types de modèles (Encodeur-Classifieur et Classifieur à réseau dense (sans Encodeur donc)), avec indication de leur précision moyenne.

Analyse des résultats et premières conclusions Les résultats obtenus sont présentés dans la Table 1. Les performances moyennes de classification des modèles appris sont données à titre indicatif : Encodeur-Classifieur a une accuracy de 74.21% ($\pm$ 3.37), Classifieur seul a une accuracy de $61.97\ \%(\pm$ 2.61).

Nous pouvons retenir plusieurs points de ces expériences. D'abord, la faible variance pour TS-Relax nous rassure quant à la robustesse de la méthode face à la stochasticité des masques. Le nombre élevé de masques permet de palier au caractère aléatoire de la génération de masques. Ensuite, nous remarquons que les scores des explications générées de manière aléatoires restent très faible. Ainsi, même la meilleure explication générée aléatoirement par modèle appris obtient un moins bon score de Monotonicité que celle produite pas TS-RELAX.

Les scores de monotonicity du modèle prenant en entrée les séries temporelles (FFN) et n'utilisant pas l'encodeur sont assez proches de ceux du modèle avec encodeur(ENC) qui utilise les représentations apprises. Sous réserve des résultats de nos prochaines expériences, nous pourrions arriver à la conclusion que les explications de TS-Relax pour les représentations produites par cet encodeur model-agnostique, sont pertinentes pour d'autres modèles.

Nous présentons dans un tableau en annexe 2 les résultats détaillés pour plusieurs datasets. Nous remarquons qu'il n'y a pas une grande différence de performance entre datasets synthétiques et datasets réels, néanmoins certains datasets présentent de moins bonnes performances en termes de monotonicité (Coffee, GunPoint et Plane par exemple).

5 Discussion

Les résultats précédents sont encourageants quant à la pertinence de l'utilisation de la méthode Relax pour l'interpretabilité des représentations de séries temporelles. Traitons maintenant des pistes de recherche à explorer.

Techniques de masquages et de perturbation Nous aimerions étudier l'impact du choix du masquage et de la perturbations sur les résultats des interprétations pour les séries temporelles. Nous envisageons par exemple pour les perturbations d'étudier les modèle autoregressif type ARMA.
Les auteurs de [3] n'ont pas expliqué la variabilité des scores d'accuracy selon le paramètre K qui représente le nombre d'exemples négatifs utilisés lors de l'entraînement de l'encodeur par contrastive learning. Grâce à TS-Relax, nous pourrions visualiser, sur un même exemple, les scores d'importances généré par TS-Relax pour différents encodeurs entrainé avec différentes valeurs de K. Et de manière plus générale, TS-Relax pourrait servir à comprendre les différences dans la manière d'encoder les séries temporelles entre différents encodeurs.
La métrique de complexité n'est pas compatible avec nos scores d'importances générés par la loi uniforme. En effet, l'entropie d'une distribution uniforme entre 0 et 1 est nulle, les scores de complexité seront donc autour de 0 bien que les importances aléatoires ne soient pas du tout "simple d'interprétation". Il faudra soit trouver une autre façon de générer les explications aléatoires, soit trouver une autre métrique comme la "effective complexity" [5] par exemple.
Early Stopping : L'hyper-paramètre du nombre de masques influence le temps de calcul des scores. Si le modèle converge rapidement, une procédure d'arrêt précoce pourrait être utilisée, basée sur l'étude de la variation de la pente des scores d'importance. Si cette variation est sous une valeur epsilon, la génération de masques supplémentaires serait superflue et l'algorithme se terminerait avec le vecteur de scores actuel.
L'encodeur non supervisé vise à maximiser la séparation des représentations qu'il génère, sans connaissance des classes, en se concentrant sur la séparation et le rapprochement. Une hypothèse intéressante est l'existence de sous-clusters au sein d'une même classe, que nous pourrions visualiser et expliquer grâce à TS-Relax. L'application d'une méthode de réduction de dimension pourrait révéler ces sous-clusters. En attribuant des explications à chaque cluster, nous pourrions identifier les caractéristiques ou combinaisons de caractéristiques que l'encodeur utilise pour séparer les données. Cette interprétation des représentations et le clustering pourraient offrir une meilleure compréhension d'un dataset, facilitant par exemple la description d'un cluster.

6 Conclusion

En conclusion, nous avons tenté de répondre au défi de l'interprétabilité des séries temporelles en adaptant la méthode RELAX à des représentations apprises de séries temporelles. Les résultats obtenus ont montré une amélioration significative en termes de monotonicité par rapport

à des scores d'importances aléatoires. Bien que prometteurs, ces résultats nécessitent néanmoins encore des expériences supplémentaires pour arriver à une méthode universelle d'interprétabilité des modèles d'apprentissage automatique de séries temporelles. En particulier, les choix des techniques de masquage et de perturbation méritent d'être explorées pour mieux prendre en compte les spécificités des séries temporelles.

7 Remerciements

Ce travail a été réalisé avec le soutien de l'Agence nationale de la recherche française (ANR), dans le cadre du projet TAUDoS (ANR-20-CE23-0020).

Références

[1] Umang Bhatt, Adrian Weller, and José M. F. Moura. Evaluating and aggregating feature-based model explanations. *CoRR*, abs/2005.00631, 2020.

[2] Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6) :1293–1305, 2019.

[3] Jean-Yves Franceschi, Aymeric Dieuleveut, and Martin Jaggi. Unsupervised Scalable Representation Learning for Multivariate Time Series. In *Advances in neural information processing systems*, volume 32. arXiv, 2019. arXiv :1901.10738 [cs, stat].

[4] Anna Hedström, Leander Weber, Dilyara Bareeva, Franz Motzkus, Wojciech Samek, Sebastian Lapuschkin, and Marina M.-C. Höhne. Quantus : An Explainable AI Toolkit for Responsible Evaluation of Neural Network Explanations, February 2022. arXiv :2202.06861 [cs].

[5] An-phi Nguyen and María Rodríguez Martínez. On quantitative aspects of model interpretability, July 2020. arXiv :2007.07584 [cs, stat].

[6] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE : Randomized Input Sampling for Explanation of Black-box Models, September 2018. arXiv :1806.07421 [cs].

[7] Kristoffer K. Wickstrøm, Daniel J. Trosten, Sigurd Løkse, Ahcène Boubekki, Karl Øyvind Mikalsen, Michael C. Kampffmeyer, and Robert Jenssen. RELAX : Representation Learning Explainability, February 2022. arXiv :2112.10161 [cs, stat].

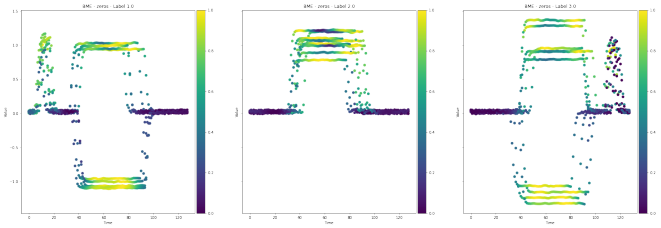
Annexes

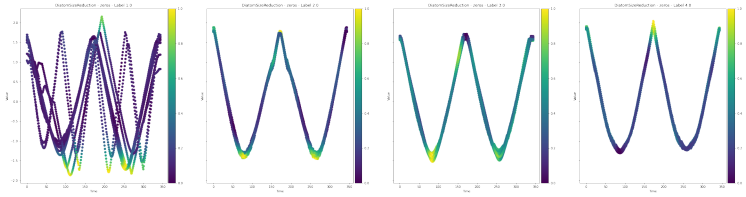
Pour une transparence et une compréhension complètes de notre travail, nous présentons ici, à titre informatif, les résultats exhaustifs de nos expériences sur tous les jeux de données sélectionnés. Vous trouverez ces résultats détaillés (moyennes et variances sur 5 entraînements) dans le tableau présenté à la page suivante de ce document.

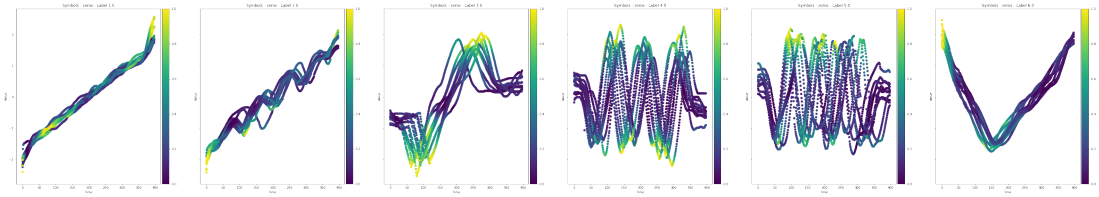
Dataset	NumClasses	Model	Accuracy	Relax (C)	Relax (M)	Random mean (M)	Random max (M)
BME	3	EncoderClassifier	44.1	4.852	23.5	-2.6	10.5
BME	3	FFNClassifier	49.7	4.852	24.4	-0.6	9.1
CBF	3	EncoderClassifier	65.3	4.848	21.2	0.4	5.4
CBF	3	FFNClassifier	72.6	4.848	13.6	0.1	10.7
Coffee	2	EncoderClassifier	97.9	5.643	5.7	-0.5	3.0
Coffee	2	FFNClassifier	49.3	5.643	4.6	-2.1	5.5
GunPoint	2	EncoderClassifier	66.3	5.007	13.7	-0.2	6.8
GunPoint	2	FFNClassifier	65.3	5.007	9.2	-0.8	8.9
Plane	7	EncoderClassifier	73.5	4.964	11.3	-0.4	5.4
Plane	7	FFNClassifier	64.2	4.964	7.4	0.8	6.6
Trace	4	EncoderClassifier	67.6	5.613	NaN	0.7	6.4
Trace	4	FFNClassifier	43.0	5.613	NaN	NaN	NaN
TwoLeadECG	2	EncoderClassifier	81.1	4.352	25.6	0.1	5.8
TwoLeadECG	2	FFNClassifier	58.9	4.352	25.7	-1.0	9.6
TwoPatterns	4	EncoderClassifier	90.7	4.845	23.0	-0.5	5.4
TwoPatterns	4	FFNClassifier	67.1	4.845	21.6	-1.0	3.6
UMD	3	EncoderClassifier	66.4	5.010	18.3	-2.3	8.4
UMD	3	FFNClassifier	51.8	5.010	43.9	-1.6	15.3
Wafer	2	EncoderClassifier	89.2	5.019	NaN	NaN	NaN
Wafer	2	FFNClassifier	97.7	5.019	13.6	0.6	11.1

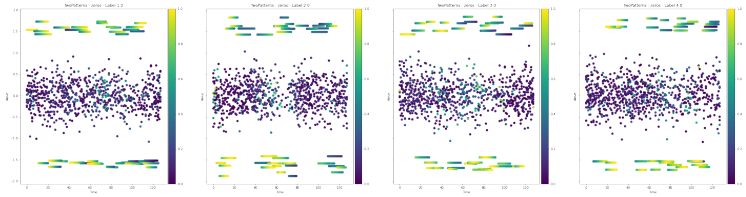
TABLE 2 – Les scores moyens (sur 5 apprentissages) d'accuracy (en %) et de monotonicity (en %) et de complexity. "Nan" signifie que l'évaluation n'a pas aboutit (Timeout)

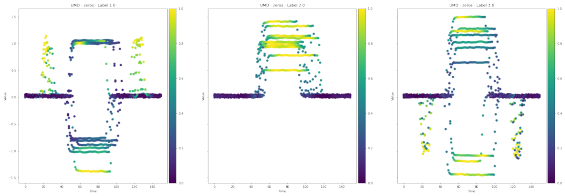
.


BME - Jorge - Label 1.0
BME - Jorge - Label 2.0
BME - Jorge - Label 3.0


DistortionWaveReduction - Jorge - Label 1.0
DistortionWaveReduction - Jorge - Label 2.0
DistortionWaveReduction - Jorge - Label 3.0
DistortionWaveReduction - Jorge - Label 4.0


Symbols - Jorge - Label 1.0
Symbols - Jorge - Label 2.0
Symbols - Jorge - Label 3.0
Symbols - Jorge - Label 4.0
Symbols - Jorge - Label 5.0
Symbols - Jorge - Label 6.0


TwoPatterns - Jorge - Label 1.0
TwoPatterns - Jorge - Label 2.0
TwoPatterns - Jorge - Label 3.0
TwoPatterns - Jorge - Label 4.0


UMD - Jorge - Label 1.0
UMD - Jorge - Label 2.0
UMD - Jorge - Label 3.0