

Augmentation de données pour la classification de séries temporelles par réseaux de neurones profonds résiduels

Hassan Ismail Fawaz*, Germain Forestier*
Jonathan Weber*, Lhassane Idoumghar*, Pierre-Alain Muller*

*IRIMAS, Université de Haute-Alsace, 68100 Mulhouse, France
{first-name}.{last-name}@uha.fr

1 Introduction

L'augmentation de données consiste à générer des données artificielles permettant d'améliorer la diversité des données d'entraînement d'un classifieur dans le but d'améliorer ses performances. Les réseaux de neurones convolutionnels profonds (CNN) ont récemment prouvé leur efficacité dans le domaine de la classification de séries temporelles (CST). Cependant, les techniques d'augmentation de données n'ont pas encore été complètement explorées pour la CST. Dans cet article, nous proposons de tirer parti d'une technique d'augmentation des données basée sur DTW spécialement développée pour les séries temporelles, afin d'améliorer les performances d'un réseau de neurones profond résiduel (ResNet) pour la CST.

2 Méthode

Nous avons choisi d'améliorer la capacité de généralisation du réseau ResNet proposé dans Wang et al. (2017). En adoptant une architecture déjà validée, nous pouvons attribuer toute amélioration des performances du réseau uniquement à l'augmentation des données. Pour plus de détails concernant l'architecture, nous conseillons la lecture de Ismail Fawaz et al. (2018).

La méthode d'augmentation des données que nous avons choisie de tester avec cette architecture profonde a été proposée afin d'augmenter le jeu d'entraînement défini pour un NN-DTW dans un problème de simulation de démarrage à froid (Forestier et al., 2017). Pour plus de détails concernant la méthode de pondération, les lecteurs peuvent se référer à Forestier et al. (2017). En ce qui concerne le calcul de la séquence moyenne, nous avons adopté l'algorithme DBA (Petitjean et al., 2016) dans notre processus d'augmentation de données.

3 Résultats

Nous avons évalué notre méthode d'augmentation de données pour ResNet sur l'UCR archive (Chen et al., 2015). Pour former les modèles d'apprentissage profond, nous avons tiré parti de la puissance de calcul élevée de 60 GPU d'un cluster de calcul ¹.

1. Notre code source est disponible sur <https://github.com/hfawaz/aaltd18>