Génération de données synthétiques à partir d'une forêt aléatoire

J. Gonzalez[1], F. Dama[1]

[1] Centre de Recherche et d'Innovation de Talan, France

Résumé

Dans le cadre de l'apprentissage incrémental avec les forêts NCMF, nous proposons une stratégie de génération de données synthétiques locale à la volée et sur demande. En outre, cette stratégie permet, contrairement à l'algorithme original, de continuer à incrémenter le modèle sans nécessiter l'accès aux anciennes données ; également, elle permet de synthétiser un dataset complet. Nos expériences ont été réalisées sur des ensembles de données de référence de UCI et offrent des résultats prometteurs.

Mots-clés

Apprentissage incrémental, Données synthétiques, NCMF.

Abstract

In the context of incremental learning with NCMF forests, we propose a strategy for generating local synthetic data on the fly and on demand. Moreover, this strategy allows, unlike the original algorithm, to continue to increment the model without requiring access to old data ; also, it allows to synthesize a complete dataset. Our experiments were performed on UCI benchmark datasets and offer promising results.

Keywords

Incremental Learning, Synthetic data, NCMF.

1 Introduction

Dans de nombreuses applications, les données sont disponibles que par petits lots au fil du temps [2]. La question se pose donc de savoir ce que l'on fait du modèle entraîné sur un sous-jeu de données. La stratégie qui consiste à réentraîner le modèle depuis le début (*from scratch*) n'est clairement pas efficace [4]. D'une part, cette méthode est très coûteuse en temps de calcul, d'autre part, elle empêche l'intégration de nouvelles données en temps réel et n'est pas réalisable lorsque les données initiales ne sont plus disponibles. Il apparaît ainsi nécessaire de mettre à jour un modèle existant de manière incrémentale.

Il a été démontré que les forêts aléatoires (RF) [1], en plus de leur nature multi-classes et leur capacité à généraliser, ont également la capacité de s'incrémenter en données et en classes [5]. Les forêts de type NCM (*Nearest Class Mean*) proposent une fonction de séparation différente des forêts classiques, ainsi que diverses stratégies d'incrémentation [6]. Cependant, les stratégies proposées pour continuer de faire croître les arbres pendant la phase incrémentale nécessitent l'accès aux données précédentes, ce qui peut être problématique. Nous proposons ici une stratégie qui permet aux NCMF de poursuivre leur incrémentation sans accéder aux anciennes données, mais plutôt en les générant sur demande de manière synthétique.

La structure de l'article est décrite comme suit. Dans la section 2, nous décrivons la stratégie d'incémentation proposée. La section 3 est dédiée à la génération de données synthétiques avec la NCMF et le modèle de mélange gaussien (utilisé pour comparaison). Ensuite en section 4, la capacité générative de la NCMF et la pertinence de la stratégie d'incrémentation proposée sont évaluées, en s'appuyant sur des datasets benchmarks de UCI. La dernière section est réservée pour la conclusion.

2 La stratégie proposée : IGTLGSS

2.1 Nearest Class Mean Forest (NCMF)

Nearest Class Mean Forest (NCMF) [6] est une forêt aléatoire (RF) [1] dont les nœuds sont des classifieurs NCM (Nearest Class Mean) [3]. Dans ces nœuds, des centroïdes conditionnels aux classes sont calculés et affectés à un noeud fils gauche ou droit. Une observation parcourant le noeud est alors dirigée vers le noeud fils associé au centroïde le plus proche. Une fois entraînées sur un premier jeu d'entraînement, les NCMFs peuvent être mises à jour avec de nouvelles données. La stratégie d'incrémentation IGT (*Incremental Growing Tree*) peut être utilisée à cet effet.

2.2 Incrémentation avec IGT et limite

La stratégie d'incrémentation IGT, introduite dans [6], procède comme suit : la nouvelle observation est propagée dans chaque arbre de la forêt pré-entraînée ; ensuite, les distributions de classes dans chacune des feuilles qu'elle atteint sont mises à jour. Lorsque l'incrément engendre un changement de classe majoritaire dans une feuille, alors cette feuille est transformée en nœud et la construction récursive d'un sous-arbre local débute. Une limite inhérente à la stratégie IGT est qu'elle nécessite l'accès aux anciennes données d'entraînement. Pour pallier cette limite, nous proposons la méthode IGTLGSS (*Incremental Growing Tree with Local Generation of Synthetic Samples*).

2.3 La stratégie IGTLGSS

La méthode IGTLGSS consiste en la génération de données synthétiques locales au moyen de la loi normale multivariée. Pour rendre cela possible, nous avons stocké les matrices de covariance conditionnelles aux classes au ni-

veau des noeuds, en plus des centroïdes. Dans la suite, nous notons $\mathcal{K} = \{k_1, ..., k_l\}$ l'ensemble de l labels et $(X^{[INCR]}, y^{[INCR]})$ l'ensemble à partir duquel la forêt pré-entraînée doit s'incrémenter.
Pour chaque observation (x, y) dans $(X^{[INCR]}, y^{[INCR]})$:

1. Chaque arbre de la forêt propage l'observation jusqu'à une feuille pour prédire un label $k_i \in \mathcal{K}$.
2. Chaque arbre t, pour lequel la prédiction k_i ne correspond pas au label y, est incrémenté. L'idée ici est de mettre à jour seulement les arbres en difficulté en exécutant les étapes suivantes :
   (i) (x, y) est propagée jusque dans une feuille, puis la répartition $S^t(x) = [S^t(k_1|x), ..., S^t(k_l|x)]$ stockée dans celle-ci est extraite, où $S^t(k_i|x)$ correspond au nombre d'occurrences de la classe k_i dans la feuille.
   (ii) Pour chaque classe k_i, nous utilisons la loi normale multivariée locale correspondant à cette classe (dont les paramètres sont stockés dans le noeud parent, *i.e.*, l'ancienne feuille). De cette manière, nous générons un ensemble synthétique local $(X^{[SYNTH]}, y^{[SYNTH]})$.
   (iii) L'observation (x, y) est ajoutée à $(X^{[SYNTH]}, y^{[SYNTH]})$ et, depuis cette position, la construction récursive du sous-arbre se poursuit comme dans une phase d'apprentissage classique.

Le modèle $NCMF$ devient alors capable de s'incrémenter à partir de nouvelles données sans avoir besoin d'accéder aux anciennes données.

3 Génération de données synthétiques

3.1 *Nearest Class Mean Forest* (NCMF)

Pour générer des données synthétiques à partir d'une NCMF entraînée, nous proposons la procédure suivante. Pour chaque classe k_i, une observation est générée en trois étapes : (i) un arbre t est tiré aléatoirement dans la forêt ; (ii) ensuite, une feuille est tirée aléatoirement, de telle sorte que plus une feuille contient d'occurrences de la classe k_i, plus elle a de chance d'être choisie ; (iii) enfin, une observation est générée à partir de la loi normale multivariée locale correspond à la classe k_i (dont les paramètres sont stockés dans le noeud parent). Les étapes précédentes sont répétées pour obtenir le bon nombre de données.

3.2 Gaussian Mixture Model (GMM)

Le GMM est un modèle probabiliste exprimé sous forme de mélange de lois Gaussiennes (lois normales multivariées) [8]. Il permet d'estimer de façon paramétrique la distribution d'un ensemble de variables aléatoires interdépendantes en la modélisant comme la somme de K Gaussiennes. De plus, chaque loi Gaussienne est associée à une classe spécifique.
Les paramètres du modèle GMM sont généralement estimés de façon non-supervisée au moyen de l'algorithme Espérance-Maximisation (EM) [9]. Cependant, lorsque les données sont labélisées, comme c'est le cas dans notre étude, une loi Gaussienne peut être ajustée à chaque classe de façon supervisée.
Le GMM est un modèle probabiliste qui permet de générer des données synthétiques. La génération de données est faite en deux étapes : (i) dans un premier temps, la classe de l'observation est générée ; (ii) ensuite, l'observation est générée à partir de la loi Gaussienne correspondante. Les étapes (i) et (ii) sont répétées jusqu'à l'obtention d'un dataset synthétique.

4 Expérimentations

4.1 Datasets

Dans nos expérimentations, nous avons considéré six datasets benchmarks de UCI dont la description est fournie dans le tableau 1.

Dataset	$\lvert\mathcal{K}\rvert$	n_{obs}	n_{feat}
Raisin Dataset ($R.$)	2	900	4
Breast Cancer Coimbra ($B.C.$)	2	116	10
Banknote authentication ($B.K.$)	2	1372	5
Avila ($A.V.$)	12	20867	10
Speaker Accent Recognition ($A.R.$)	6	329	12
Optical Recognition of Handwritten Digits ($H.G.$)	10	5620	64

TABLE 1 – Description des datasets. De gauche à droite : nombre de classes, d'observations et de features (https ://archive.ics.uci.edu/ml/datasets/).

4.2 Protocole expérimental

4.2.1 Évaluation de la capacité générative de la NCMF

Les cinq premiers datasets (c.f. Table 1) ont été considérés dans cette expérimentation. Chaque dataset a été découpé en deux parties : un jeu d'entraînement (80%) et un jeu de test (20%), à l'exception de $A.V.$ pour lequel un découpage est déjà fourni. Ensuite, une NCMF et un GMM ont été entraînés sur chaque jeu d'entraînement. Enfin, les modèles obtenus ont été utilisés pour générer des datasets synthétiques $(X_{GMM}^{[SYNTH]}, y_{GMM}^{[SYNTH]})$ et $(X_{NCMF}^{[SYNTH]}, y_{NCMF}^{[SYNTH]})$, de taille identique aux datasets d'entraînement et qui respectent la répartition des classes de ces derniers.
Pour évaluer la qualité des données synthétiques précédemment générées, nous avons suivi la méthode proposée dans [7]. Plusieurs modèles de classification sont entraînés sur les données réelles et synthétiques. Ensuite, la performance des modèles obtenus est évaluée sur un jeu d'évaluation (composé uniquement de données réelles). Lorsque la perte de performance résultant de l'utilisation des données synthétiques est faible, les données synthétiques sont considérées suffisamment similaires aux données réelles.
Nous avons considéré les modèles de classification suivants : forêt aléatoire standard (RF), NCMF, *Naive Bayes* (NB) et SVM. La mesure de performance utilisée est la

moyenne des métriques d'*accuracy* obtenues pour chacun des modèles. À noter que l'optimisation des hyperparamètres des différents modèles n'entre pas dans le cadre de cet article.

4.2.2 Apprentissage incrémental

Pour comparer les stratégies d'incrémentation IGTLGSS et IGT, nous avons considéré le dataset H.G. (c.f. Table 1). Le protocole utilisé s'apparente à celui décrit dans [5]. Les données d'entraînement (80%) sont découpées aléatoirement sous forme de 50 sous-ensembles (*batchs*) de taille et de distributions de classes identiques. La forêt s'entraîne sur le premier batch, puis s'incrémente sur les autres. L'*accuracy* du modèle est mesurée à la fin de chaque incrémentation.

4.3 Résultats et Analyse

Le tableau 2 compare les performances des modèles entraînés sur les jeux de données réelles et synthétiques. Comme attendu, les modèles entraînés sur les données réelles obtiennent de meilleures performances en comparaison à ceux entraînés sur les données synthétiques (à l'exception du *Raisin Dataset*). Par ailleurs, les résultats montrent que les modèles entraînés sur $(X_{NCMF}^{[SYNTH]}, y_{NCMF}^{[SYNTH]})$ surpassent, dans la majorité des cas, ceux entraînés sur $(X_{GMM}^{[SYNTH]}, y_{GMM}^{[SYNTH]})$. De plus, la perte de performance obtenue suite à l'utilisation des données synthétiques générées par le modèle NCMF est seulement de 0.038 ± 0.029. Ce résultat est prometteur car, à notre connaissance, nous n'avons pas vu de forêts aléatoires être en capacité de générer des données synthétiques.

Dataset	Réel	GMM	NCMF
$R.$	$\underline{0.825} \pm 0.026$	0.829 ± 0.014	$\mathbf{0.846} \pm 0.021$
$B.C.$	$\mathbf{0.635} \pm 0.133$	0.583 ± 0.076	$\underline{0.615} \pm 0.205$
$B.K.$	$\mathbf{0.945} \pm 0.093$	0.924 ± 0.080	$\underline{0.928} \pm 0.102$
$A.V.$	$\mathbf{0.677} \pm 0.280$	0.348 ± 0.041	$\underline{0.642} \pm 0.201$
$A.R.$	$\mathbf{0.686} \pm 0.090$	$\underline{0.629} \pm 0.045$	0.591 ± 0.113

TABLE 2 – Moyenne des métriques d'*accuracy* obtenues pour différents modèles de classification en utilisant les datasets réels et synthétiques.

La figure 1 décrit l'évolution des performances des NCMF incrémentées suivant les stratégies IGT ou IGTLGSS. Nous pouvons observer que les performances sont très satisfaisantes, 0.01 point de moins pour les données synthétiques. La courbe montre que malgré le fait que les données soient synthétiques, la NCMF est capable d'améliorer ses performances jusqu'au *batch* numéro 35. Au delà, la stratégie IGTLGSS semble atteindre un plafond, là où la méthode IGT semble permettre un gain de performance. Il serait intéressant d'observer le comportement de la stratégie IGTLGSS sur des datasets plus larges permettant de considérer par exemple une centaine de *batchs*.

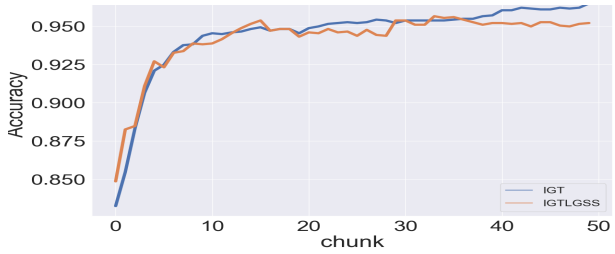

FIGURE 1 – Dataset H.G. : évolution de l'*accuracy* de la $NCMF$ en fonction du nombre d'incrémentations.

5 Conclusion

Notre proposition permet, contrairement à l'algorithme original, de continuer à incrémenter la NCMF sans nécessiter l'accès aux anciennes données. Nos premiers résultats sont encourageants et ouvrent des pistes pour de futures expérimentations.

Références

[1] L. Breiman. Random Forests. *Machine Learning*, 45(1) :5–32, October 2001.

[2] A. Gepperth and B. Hammer. Incremental learning algorithms and applications. In *European symposium on artificial neural networks (ESANN)*, 2016.

[3] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning : data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[4] J. He, R. Mao, Z. Shao, and F. Zhu. Incremental learning in online scenario. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13926–13935, 2020.

[5] R. Pecori, P. Ducange, and F. Marcelloni. Incremental learning of fuzzy decision trees for streaming data classification. In *11th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2019)*, pages 748–755, 2019.

[6] M. Ristin, M. Guillaumin, J. Gall, and L. Van Gool. Incremental learning of ncm forests for large-scale image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, page 3654–3661, 2014.

[7] A. Torfi and E. A. Fox. Corgan : Correlation-capturing convolutional generative adversarial networks for generating synthetic healthcare records. *arXiv preprint arXiv :2001.09346*, 2020.

[8] H. Wan, H. Wang, B. Scotney, and J. Liu. A novel gaussian mixture model for classification. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 3298–3303. IEEE, 2019.

[9] M.S. Yang, C.Y. Lai, and C.Y. Lin. A robust em clustering algorithm for gaussian mixture models. *Pattern Recognition*, 45(11) :3950–3961, 2012.