Projets "Advanced Supervised Learning"

M2 DM - Université Lyon 2 - 2021/2022

Responsable : Julien Ah-Pine

1 Objectif du projet

Dans le cadre du cours "Advanced Supervised Learning" du Master 2 Data-Mining, il vous est demandé de réaliser un projet afin de compléter vos connaissances, pratique de méthodes et concepts vus en cours dans le but de parfaire vos compétences. Il y a deux aspects associés à ce projet : (a) l'un relevant d'un cadre problématisé fixé et (b) l'autre relevant d'un cas pratique que vous choisirez. Les deux aspects ne sont pas indépendants. Le point (a) consiste à vous fournir une tâche spécifique qu'il vous faudra choisir parmi quatre (sujets [A], [B], [C] ou [D] décrits par la suite). Le point (b) consiste à choisir un jeu de données intéressant qui soit en adéquation avec le sujet que vous aurez choisi en (a). Par exemple le sujet [C] concerne l'apprentissage dans un cas déséquilibré il vous faudra donc trouver un jeu de données qui soit déséquilibré. Pour plus de précisions sur les autres sujets voir ci-après.

2 Aspects liés à l'organisation et à la remise du dossier

Les projets s'effectuent par groupe de 4. Dans l'esprit, vous êtes une team de data scientists devant répondre à plusieurs besoins exprimés par une entreprise. La bonne organisation de votre travail collectif, la répartition efficace des tâches, et le travail en groupe/sous-groupes pour la résolution de points difficiles, constitueront autant d'atouts pour la réussite de votre projet. Il est attendu que vous fournissiez :

1. un script R ou python dans lequel se trouvera toutes les fonctions développées,
2. un ou plusieurs fichiers comportant les données utilisées,
3. un rapport au format PDF accompagnant le projet.

Vous devrez créer une archive .zip contenant ces fichiers. Vous nommerez votre archive de la manière suivante NOM1_NOM2_NOM3_NOM4.zip où NOMi sont les noms des membres du groupe. Pour les non-alternants, **vous devrez envoyer par email cette archive au plus tard le samedi 23 octobre 2021 minuit à l'adresse mail : julien.ah-pine@univ-lyon2.fr**. Pour les alternants, la date de rendu sera précisée ultérieurement. **ATTENTION** : vous êtes responsables de votre envoi et donc toute absence de ressource ou tout problème conduisant à l'impossibilité d'accéder correctement à votre travail est de votre responsabilité.

3 Descriptif du rapport accompagnant votre projet

L'objectif du manuscrit est de présenter et de mettre en valeur votre travail. Vous devez le rédiger dans l'esprit d'un rapport scientifique. Vous devrez présenter de façon claire :

+ le sujet : les thématiques abordées et l'intérêt de celles-ci selon la nature du travail choisi (qu'est ce que vous allez faire et pourquoi il est intéressant de le faire en pratique ?) ;
+ les problématiques scientifiques : mise en perspective des méthodes et/ou des concepts que vous allez traiter (quelles problématiques théoriques/méthodologiques ce que vous allez faire permet de traiter ?) ;
+ votre implémentation : le langage utilisé, le workflow, les différentes fonctions et leur spécification (comment avez-vous réalise, organisé votre code et avec quels outils ?) ;

- le jeu de données/benchmark que vous avez choisi pour votre analyse et qui met en valeur le sujet choisi (présentez votre jeu de données et indiquez quelles sont l'originalité, les spécificités de celui-ci et en quoi il permet d'illustrer les problématiques de votre sujet) ;
- le protocole expérimental : les méthodes et/ou implémentations concurrentes à votre travail, la procédure d'évaluation, les ensembles de paramètres utilisés ;
- les résultats expérimentaux et leurs analyses : tables et/ou graphiques des benchmarks montrant des comparaisons entre différentes méthodes et/ou implémentations, commentaires sur les résultats obtenus et notamment montrez en quoi votre travail répond aux problématiques initialement posées ;
- une discussion scientifique/analyse critique : les avantages et les limites de la/les méthodes et/ou de votre implémentation, les points à améliorer, les extensions de la littérature qui seraient intéressantes de poursuivre... ;

= une conclusion : en indiquant notamment les apports et les difficultés rencontrées lors de la réalisation de ce projet.

Il est clair également que le manuscrit devra comporter une bibliographie montrant votre investissement lors de la réalisation de ce travail.

L'organisation du manuscrit donnée ci-dessus n'est pas restrictive et vous pouvez ainsi ajouter toute section que vous jugerez utile pour la mise en valeur de votre travail.

Concernant le code R ou Python, il est attendu qu'il soit soigné et bien commenté. Vous devez programmer dans l'esprit de la réutilisabilité de votre travail par une tierce personne. Même si cela est encouragé, il n'est pas requis que le code soit optimisé à condition que l'exécution du script se fasse dans des temps raisonnables (ceci est notamment valable pour les sujets "algorithmiques" [A] ou [B]).

4 Liste des sujets

La nature des sujets diffère selon deux axes : "algorithmique", "contexte particulier".

Concernant le type "algorithmique", l'objectif principal du travail à effectuer est d'implémenter "from scratch" une ou plusieurs méthodes et d'illustrer sur un jeux de données choisis, les apports de la ou des méthodes étudiées. La difficulté principale est donc la maîtrise technique et l'implémentation de l'approche. Il est attendu du rapport que les étudiants présentent la ou les méthodes, le pseudo-code, le code et qu'ils s'attardent sur les points particulièrement ardus de l'implémentation. Dans l'esprit, le rapport doit permettre à une personne tierce et novice de comprendre aisément les fondements de la méthode et son implémentation.

Pour ce qui est du type "contexte particulier", il s'agit ici de situations où l'apprentissage nécessite des traitements supplémentaires et spécifiques. En effet, dans des études de cas réels, nous rencontrons des situations particulières, qui nécessitent des traitements ad-hoc en amont ou en aval ou au cours de la phase d'apprentissage. Les problèmes de données mixtes ou de fusion d'information ou de distributions de classes asymétriques en sont des illustrations. Dans ce cadre, il est attendu des étudiants qu'ils implémentent des techniques permettant de tenir compte de ces contextes particuliers. Dans l'esprit, le rapport doit présenter la problématique, les méthodes classiques et illustrer sur un exemple concret et à l'aide d'une ou plusieurs méthodes d'apprentissage, l'intérêt des techniques mises en oeuvre dans le cadre de la problématique étudiée.

Remarque : les références données ci-dessous sont toutes disponibles sur le net.

4.1 Sujet à finalité "algorithmique"

4.1.1 Sujet "random forest" [A]

Ce projet consiste en l'étude approfondie des arbres de décision et de leur extension par le biais des forêts aléatoires. Les objectifs sont :

— Implémenter le pseudo-code des arbres décisionnels donné en cours. On attend donc deux fonctions `ArbreGeneration` et `DivisionAttribut`. Une seul des deux problèmes,

régression ou catégorisation, pourra être considéré.
— Implémenter une fonction `RandomForest` qui utilise les fonctions précédentes et met en oeuvre les ré-échantillonage sur les individus et sur les variables tels que décrit dans le pseudo-code du cours.
— Présenter dans le rapport un texte synthétique sur les fondements et propriétés des forêts aléatoires. Vous pourrez vous inspirer du cours mais toute autre référence pourra être utilisée pour enrichir le propos.
— Montrer aux travers d'expériences la supériorité des forêts aléatoires sur les arbres de décision permettant ainsi de valider empiriquement les propriétés théoriques discutés en cours sur les méthodes de ré-échantillonage.

Quelques références à titre indicatif que vous compléterez à votre guise et selon les besoins de votre projet : [Breiman, 2001, Rokach, 2010, Breiman, 1996].

Concernant le type de jeux de données, des données mixtes avec variables quantitatives et qualitatives sont requises pour profiter de la capacité des arbres à traiter ce type de données. Vis à vis des méthodes concurrentes, vous comparerez votre implémentation à celle classiquement utilisée dans les librairies R ou Python. Vous comparez également vos résultats avec une autre méthode classique accessible au sein des librairies usuelles.

4.1.2 Sujet "adaboost" [B]

Ce projet consiste en l'étude approfondie des deux algorithmes de base d'adaboost pour la catégorisation. Les objectifs sont :

— Implémenter le pseudo-code des approches adaboost (problèmes binaires) et adaboost.M1 (problèmes multiclasses) tels que décrits dans les articles fondateurs [Freund et al., 1996, Freund and Schapire, 1997] :
— Présenter dans le rapport une synthèse expliquant les fondements et propriétés des méthodes adaboost. Il est attendu des étudiants un "bref" état de l'art sur le sujet. Les liens/comparaisons avec d'autres méthodes telles que les svm ou le bagging seraient appréciés.
— Montrer aux travers d'expériences les bonnes performances des méthodes adaboost vis à vis d'une ou plusieurs méthodes baseline que vous choisirez afin de montrer l'apport de "décider en comité".

Quelques références à titre indicatif que vous compléterez à votre guise et selon les besoins de votre projet : [Freund et al., 1999, Rokach, 2010].

Le jeu de données choisi ici est assez libre mais le but est de montrer que le boosting permet de faire mieux qu'une méthode classique. Ainsi le jeu de données doit être réputé comme étant difficile. Vous vous comparerez également à l'implémentation d'adaboost disponible au sein des librairies usuelles.

4.2 Sujet à finalité "contexte particulier"

4.2.1 Sujet "apprentissage déséquilibré" [C]

En catégorisation, dans de très nombreux problèmes réels, la distribution des classes est fortement déséquilibrée. Dans le cas binaire cela se traduit par une classe très représentée par rapport à l'autre. On rencontre cette situation pour les problèmes de détection d'anomalies, de fraudes, de maladies rares... Si on ignore cette asymétrie, toute méthode serait implicitement biaisée et aurait plus de difficultés à détecter la classe minoritaire qui est souvent celle d'intérêt. Les objectifs de ce projet sont les suivants :

— Présenter la problématique et expliquer en quoi celle-ci nécessite un traitement particulier. Dans cette perspective, il faut expliquer également en quoi les mesures d'erreurs vues en cours doivent être adaptées.
— Faire un état de l'art sur les différentes méthodes classiques basées sur l'échantillonnage. Les techniques discutées et implémentées doivent contenir les approches suivantes : sous-

échantillonage, sur-échantillonage et échantillonage synthétique (SMOTE, Borderline-SMOTE et ADASYN).
- Implémenter en R ces différentes méthodes classiques. Pour les tester, vous emploierez au moins deux méthodes de catégorisation vues en cours parmi les suivantes : SVM ou régression logistique pénalisée ou arbres de décision ou analyse discriminante.
- Analyser les résultats de classification selon une (ou plusieurs) mesure d'erreur adéquate. Comparer les méthodes d'échantillonage entre elles.

Quelques références à titre indicatif que vous compléterez à votre guise et selon les besoins de votre projet : [Chawla et al., 2002, He et al., 2008, Han et al., 2005].

Naturellement, vous devrez ici utiliser un jeu de données déséquilibré.

4.2.2 Sujet "apprentissage multiclasse" [D]

Nous avons vu en cours plusieurs méthodes de catégorisation binaire et évoqué leurs extensions au cas multiclasses. En particulier, les approches "un contre tous" et "un contre un" ont été évoquées. Dans le cadre de ce sujet, l'ojectif est de se focaliser sur la méthode svm et d'implémenter pour des problèmes multiclasses, les deux précédentes méthodes ainsi que deux autres techniques non discutées en cours que sont : DAGSVM ("Directed Acyclic Graph SVM") [Platt et al., 2000] et ECOC ("Error Coding Outputs Codes") [Dietterich and Bakiri, 1995]. Plus particulièrement, il faudra :

- Présenter la problématique et rappeler que l'extension d'un classifieur binaire au cas multiclasses n'est pas trivial.
- Faire un bref état de l'art sur les quatre méthodes évoquées précédemment en vous focalisant en particulier sur leurs utilisations avec la méthode svm. Dans cette perspective, il est attendu une présentation détaillée des techniques et de leur fonctionnement (pseudo-code) ainsi qu'un rappel sur les principes de la méthode svm.
- Implémenter en R ces quatre approches multiclasses. En revanche, il ne vous est pas demandé de réimplémenter le svm binaire et une librairie R ou Python de votre choix pourra être employée.
- Tester vos différentes implémentations. Comparer les performances des méthodes et pour plusieurs noyaux (au moins deux).
- Comparer de façon critique les résultats que vous obtenez sur votre étude de cas.

Quelques références à titre indicatif que vous compléterez à votre guise et selon les besoins de votre projet : [Allwein et al., 2000, Hsu and Lin, 2002].

Naturellement, vous devrez ici utiliser un jeu de données avec plusieurs classes à prédire (au moins 5 classes).

Références

[Allwein et al., 2000] Allwein, E. L., Schapire, R. E., and Singer, Y. (2000). Reducing multiclass to binary : A unifying approach for margin classifiers. Journal of machine learning research, 1(Dec) :113–141.

[Breiman, 1996] Breiman, L. (1996). Bagging predictors. Machine learning, 24(2) :123–140.

[Breiman, 2001] Breiman, L. (2001). Random forests. Machine learning, 45(1) :5–32.

[Chawla et al., 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote : Synthetic minority over-sampling technique. J. Artif. Int. Res., 16(1) :321–357.

[Dietterich and Bakiri, 1995] Dietterich, T. G. and Bakiri, G. (1995). Solving multiclass learning problems via error-correcting output codes. Journal of artificial intelligence research, 2 :263–286.

[Freund et al., 1999] Freund, Y., Schapire, R., and Abe, N. (1999). A short introduction to boosting. Journal-Japanese Society For Artificial Intelligence, 14(771-780) :1612.

[Freund and Schapire, 1997] Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci., 55(1) :119–139.

[Freund et al., 1996] Freund, Y., Schapire, R. E., et al. (1996). Experiments with a new boosting algorithm. In Icml, volume 96, pages 148–156.

[Han et al., 2005] Han, H., Wang, W.-Y., and Mao, B.-H. (2005). Borderline-smote : A new over-sampling method in imbalanced data sets learning. In Huang, D.-S., Zhang, X.-P., and Huang, G.-B., editors, Advances in Intelligent Computing, volume 3644 of Lecture Notes in Computer Science, pages 878–887. Springer Berlin Heidelberg.

[He et al., 2008] He, H., Bai, Y., Garcia, E., and Li, S. (2008). Adasyn : Adaptive synthetic sampling approach for imbalanced learning. In Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on, pages 1322–1328.

[Hsu and Lin, 2002] Hsu, C.-W. and Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. IEEE transactions on Neural Networks, 13(2) :415–425.

[Platt et al., 2000] Platt, J. C., Cristianini, N., and Shawe-Taylor, J. (2000). Large margin dags for multiclass classification. In Advances in neural information processing systems, pages 547–553.

[Rokach, 2010] Rokach, L. (2010). Ensemble-based classifiers. Artificial Intelligence Review, 33(1-2) :1–39.