Projet de Statistique descriptive 2

31 mars 2022

1 Principes généraux

Les objectifs de ce projet sont détaillés ci-dessous.

1.1 Manipuler les modèles autour d'un cas pratique réaliste

On cherche ici à faire de la détection d'opérations financières frauduleuses. Concrètement, on vous demande de créer des classifieurs ayant la meilleure performance possible. La notation dépendra bien sûr de la qualité générale du travail et des explications qui lui sont liées, mais également des résultats d'une compétition de performances prédictives. Nous tenons à vous rassurer sur ce point : d'une part, toute la notation ne reposera pas sur la seule compétition ; d'autre part, il ne s'agit pas de pénaliser le dernier de la compétition, mais bel et bien de récompenser le premier. Aussi, ce n'est pas parce qu'un étudiant se situe en bas du classement qu'il aura une mauvaise note, à condition - bien évidemment - que son travail soit correct. Vous avez tout à gagner à participer en toute bonne foi.

1.2 Acquérir/approfondir votre connaissance du langage R

Tout au long de l'enseignement, il a été question d'implémenter les modèles vus en cours en langage R. Un certain nombre de notebooks est disponible ici : https://github.com/users/mpalenciaolivar/projects/4 . Le notebook portant sur le TD 3 (arbres de décision) [1] est un TD initiatique sur les problèmes de classification ; il est donc vivement recommandé de le consulter, mais il ne constitue qu'une iniatiation rudimentaire. Les instructions pour prendre en main les projets mis sur GitHub figurent en en-tête de chaque dépôt. Il faudra sans doute descendre d'un écran ou deux pour lire ces instructions, pas forcément visibles lors de votre arrivée sur la page. De même, vous disposez de références vers des ressources que vous pourrez consulter à l'envi sur internet ou à la BU.

2 Règles de la compétition, livrables attendus et délais

2.1 Règles attenantes à la compétition

1. Les compétiteurs formeront des binômes (voire des trinômes dans des cas dont le caractère exceptionnel est laissé à l'appréciation de votre chargé de TD). La liste des binômes est à communiquer rapidement à vos chargés de TD, selon des modalités qu'ils vous communiqueront en temps voulu ;
2. On n'a le droit d'utiliser que le langage R. Les interfaces vers Python ou d'autres langages (packages reposant sur *reticulate* par exemple) ne sont pas autorisées. Si le plagiat est interdit, vous avez toutefois le droit de vous inspirer de code issu d'internet [2] et des notebooks fournis en TD. Certains ont demandé s'ils pouvaient faire leurs recherches de modèles en Python (ou autre). La réponse est oui, mais il faudra faire un livrable en R, et il faudra bien prendre garde aux différences d'implémentation entre R et Python (choix d'algo d'inférence, etc.). En d'autres termes, c'est possible, mais pas forcément conseillé.
3. Le critère d'évaluation de la compétition est le F1-score (pas le β-F1 score, mais bien le F1-score au sens strict).

Au-delà de ces 3 trois règles que l'on réputera être "de bon sens", toutes les techniques sont utilisables, même si pas vues en cours. De même, créer des features, gérer des soucis propres aux variables, etc. est tout à fait autorisé et même encouragé. L'important est que les choses soient faites correctement.

1. https://github.com/mpalenciaolivar/L2MIASHS2022-StatDesc2-TD3
2. Ex : https://www.kaggle.com/sasakitetsuya/fraudulent-transactions-analysis-by-xgboost

2.2 Livrables attendus

Le mieux est de respecter le canvas mis à disposition sur GitHub [3]. Veillez à bien indiquer les packages que vous installez dans requirements.R (les instructions sont délivrées sous forme de commentaire dans le code). Il nous faudra aussi le csv issu de l'exécution du script "MyPackages.R" (1 seul par groupe, vérifiez que tout fonctionne sur un seul PC !). Ce csv contient vos packages installés dans le détail ; l'idée est de pouvoir répliquer le fonctionnement du projet que vous fournirez au plus près, et de faire du troubleshooting au besoin. Vous avez le droit le plus absolu d'anonymiser les informations que vous jugerez confidentielles (noms des chemins, etc.), nous n'avons besoin que de répliquer votre environnement. Dans l'absolu, les livrables attendus sont les suivants :

- TOUT votre code pour la manipulation des données (préparation, etc.), l'exploration, l'entraînement des modèles (**avec seed**), avec l'ordre dans lequel consulter/exécuter les scripts/notebooks (pensez à utiliser d'astucieuses règles de nommage) ;
- TOUS vos modèles **sérialisés** ;
- TOUTES vos métriques ;
- Les ID des lignes correspondant à l'entraînement, à la validation, etc.
- un pipeline permettant de charger nos données test, de les transformer à l'identique des données d'entraînement, de faire des prédictions et d'en tirer les métriques demandées/voulues.

Pour les explorations et explications, on favorisera le format notebook (fournir une version pdf ou html en sus de la version interactive), mais un simple rapport Word peut suffire. Pour le pipeline, un simple script suffira, mais il ne faudra pas omettre de commenter. Attention aux noms des chemins, c'est pour cela que l'on impose l'usage du projet R (fichier .Rproj, à ouvrir dans RStudio), et des chemins relatifs. Pour tenir compte du fait que les chemins soient différents selon le système d'exploitation, on utilisera la fonction "file.path()". Ainsi les chemins relatifs, "Mon\Chemin\Quelconque" (Windows) ou "/Mon/Chemin/Quelconque" (Mac, GNU/Linux, FreeBSD, Unix ou Unix-like quelconque) deviendraient "file.path('Mon', 'Chemin', 'Quelconque')" [4].

Il est inutile de fournir les jeux de données (train, validation, etc.), mais il faudra nous fournir les identifiants des lignes que vous utilisez pour chaque section du jeu de données, sous forme sérialisée ou un simple listing (csv par ex.). Veillez donc à ne pas supprimer cette pseudo-variable id, qui ne **doit pas** entrer en ligne de compte dans vos traitements, encore une fois.

Nous attirons votre attention sur le fait de vérifier que tout fonctionne avant envoi. Dans le cas où nous nous retrouverions avec un projet non fonctionnel, nous essayerons de faire en sorte que cela marche (y compris en vous contactant), mais trop de temps perdu implique une pénalisation. S'il faut faire des manipulations particulières pour faire fonctionner le projet, merci de nous l'indiquer.

2.3 Date de rendu

Le 25 mai 2022 à minuit. Il vous faudra envoyer votre projet à votre chargé de TD sous forme d'archive .zip. Si votre projet ne passe pas par mail, déposer l'archive sur un Drive quelconque et envoyer le lien à la place. Rappel des adresses :

1. Mickaël LALLOUCHE : mickael.lallouche@univ-lyon2.fr
2. Miguel PALENCIA-OLIVAR : miguel.palencia-olivar@pm.me

3 Critères de notation

1. L'ensemble de la démarche d'analyse du problème (analyses exploratoires, graphiques, etc.) ;
2. L'ensemble des explications fournies autour des diverses transformations de données (centrer-réduire, etc.) ;
3. La fourniture de nouvelles métriques, à condition que celle-ci soient pertinentes au regard de ce qui est fait des données ;
4. Le fait de battre les baselines fournies (*cf* ci-dessous) ;
5. Le rang dans le classement final.

Certaines choses n'auront pas forcément été vues en cours ; vous l'aurez compris, l'idée est ici de valoriser le fait d'approfondir ce qui vous aura été enseigné. Vous disposez pour cela de toutes les ressources externes référencées dans les notebooks, et plus encore. De même, vous êtes invités à soigner votre rédaction, à expliciter vos choix et à discuter

3. https://github.com/mpalenciaolivar/L2MIASHS2022-StatDesc2-Projet
4. Ne pas copier l'instruction, il elle ne marcherait pas telle quelle dans le code du fait de LaTeX. Les "apostrophes" autour des mots sont des simples cotes, remplaçables par des guillemets. La fonction file.path est aussi utilisée dans le TD 3.

vos résultats : nous ne pouvons pas deviner vos motivations sans votre concours ! Une analyse sans fondement, et qui serait fournie au-delà du stade exploratoire ne sera pas valorisée [5]. Enfin, et même si le code n'est pas le principal, il reste important de respecter les bonnes pratiques. Prenez d'ailleurs garde au nommage des variables : R est parfois très (trop) permissif, au risque de masquer certaines fonctions/variables existantes.

Quelques exemples de problématiques auxquelles vous pouvez tenter d'apporter réponse :
- le traitement des valeurs manquantes, des outliers, etc. ;
- le déséquilibre dans les données ;
- création de nouvelles features ;
- dire comment les variables ont été sélectionnées ;
- dire quelles sont les variables-clés pour prédire si une opération est frauduleuse ou non, et si elles font sens ;
- etc.

4 Le jeu de données

4.1 Descriptif

Le jeu de données [6] est un fichier .csv de 6362620 lignes (individus) pour 12 colonnes. Nous ne vous en fournissons qu'une (grande) partie ; le reste (environ 33%) sera utilisé pour tester les performances des classifieurs et ainsi vous mettre une note. Des indications pour charger un .csv dans R sont disponibles dans le corrigé du TD 1 de Statistique descriptive [7].

Pour télécharger les données, préférez une connexion rapide et en illimité. Il faudra ouvrir l'archive avec un logiciel compatible avec 7zip [8] : https://drive.protonmail.com/urls/0NPGBXTX64#nKGYASYWMmJQ

4.2 Métadonnées

- id : un simple identifiant. Attention à ne pas le compter en tant que variable, et à ne pas le retirer, car c'est quelque chose de demandé dans les livrables ;
- step : correspond à une unité de temps dans le monde réel. Dans ce cas, 1 étape correspond à 1 heure de temps. Total des étapes 744 ;
- type : type d'opération, il y en a 5 au total ;
- amount : montant de l'opération ;
- nameOrig : nom du client ayant initié la transaction ;
- oldbalanceOrig : balance du compte **avant** la transaction ;
- newbalanceOrig : balance du compte **après** la transaction ;
- nameDest : client destinataire de la transaction ;
- oldbalanceDest : balance du compte du destinataire **avant** la transaction ;
- newbalanceDest : balance du compte du destinataire **après** la transaction ;
- isFraud : **votre variable cible** ; Il s'agit des transactions effectuées par les agents frauduleux au sein de la simulation. Dans ce jeu de données spécifique, le comportement frauduleux des agents vise à faire du profit en prenant le contrôle des comptes des clients et en essayant de vider les fonds en les transférant vers un autre compte, puis en sortant du système ;
- isFlaggedFraud : le modèle commercial vise à contrôler les transferts massifs d'un compte à l'autre et signale les tentatives illégales. Dans cet ensemble de données, une tentative illégale est une tentative de transfert de plus de 200.000 en une seule transaction.

Il est attendu de vous que vous fassiez un travail de qualification de ces variables (catégorielle, continue, etc.) et que vous en fassiez un traitement adéquat dans R. Vous n'êtes pas obligés d'utiliser tout le jeu de données pour réaliser votre entraînement de modèle. Sur le papier, "plus les modèles en voient, mieux c'est", mais vous pouvez être limités par vos machines. Dans ce cas, merci de l'indiquer [9], et surtout, d'indiquer les id des éléments retenus (cf livrables). Évidemment, s'il est fait usage de techniques pour éviter ces limitations matérielles (indexation des données plutôt que chargement in-memory, etc.), cela sera valorisé. Dans l'absolu, vous ne pourrez pas gérer autant de données avec Excel. Cela se passera donc sur R (ici encore, Python, etc. pas autorisés).

5. Au passage, ce n'est pas parce qu'une analyse ne donne rien qu'il n'est pas intéressant de savoir que vous l'avez conduite. Nous mettons ici l'accent sur la démarche et sur les conclusions.
6. https://www.kaggle.com/chitwanmanchanda/fraudulent-transactions-data
7. https://github.com/mpalenciaolivar/L2MIASHS2022-StatDesc2-TD1/blob/main/TD.md
8. Préférez la version 64 bits de 7zip : https://www.7-zip.org/download.html
9. "Je n'ai pas pu installer R et RStudio" ne compte pas.

5 Baselines

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
xgboost	Extreme Gradient Boosting	0.9995	0.9526	0.6002	0.9551	0.7357	0.7355	0.7561	28.4960
catboost	CatBoost Classifier	0.9995	0.9531	0.6032	0.9366	0.7319	0.7316	0.7504	43.6540
rf	Random Forest Classifier	0.9994	0.8249	0.5940	0.8343	0.6918	0.6915	0.7025	12.3810
et	Extra Trees Classifier	0.9994	0.8176	0.6033	0.7938	0.6836	0.6832	0.6907	4.1330
dt	Decision Tree Classifier	0.9991	0.7983	0.5971	0.6124	0.6014	0.6010	0.6026	0.7610
knn	K Neighbors Classifier	0.9990	0.6985	0.2739	0.7234	0.3921	0.3917	0.4410	7.7120
gbc	Gradient Boosting Classifier	0.9990	0.6660	0.3828	0.5338	0.4314	0.4311	0.4442	21.5810
lr	Logistic Regression	0.9989	0.7040	0.0281	0.1667	0.0481	0.0480	0.0683	1.6220
ridge	Ridge Classifier	0.9989	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1150
ada	Ada Boost Classifier	0.9989	0.9513	0.1727	0.5074	0.2533	0.2529	0.2922	5.0780
dummy	Dummy Classifier	0.9989	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0910
lightgbm	Light Gradient Boosting Machine	0.9941	0.5528	0.1476	0.0944	0.1055	0.1042	0.1097	1.2270
lda	Linear Discriminant Analysis	0.9935	0.8756	0.2359	0.0479	0.0794	0.0776	0.1037	0.2540
svm	SVM - Linear Kernel	0.9803	0.0000	0.0548	0.0003	0.0006	0.0004	0.0031	1.0390
nb	Naive Bayes	0.4523	0.8298	0.9777	0.0020	0.0040	0.0018	0.0291	0.1100
qda	Quadratic Discriminant Analysis	0.0011	0.0000	1.0000	0.0011	0.0023	0.0000	0.0000	0.1430

Source : https://www.kaggle.com/code/sasakitetsuya/fraudulent-transactions-analysis-by-xgboost