Compétition Kaggle

Mise en contexte :

Kaggle est une plateforme qui organise des compétitions en Data Science et qui récompense les meilleurs analystes internationaux.

La mission est donc de rechercher et participer à une compétition réelle et en cours sur la plateforme et de partager les résultats obtenus avec la communauté.

Sommaire :

I. ***Présentation de la compétition :***

a. Vue d'ensemble de la compétition **:**

Que ce soit au restaurant ou pour acheter des billets de concert, la vie moderne compte sur la commodité d'une carte de crédit pour effectuer les achats quotidiens. Elle nous évite de transporter de grandes quantités d'argent liquide et peut également avancer un achat complet qui peut être payé dans le temps. Comment les émetteurs de cartes savent-ils que nous rembourserons ce que nous facturons ? Il s'agit d'un problème complexe pour lequel il existe de nombreuses solutions, et encore plus d'améliorations potentielles.

La prévision de défaut de crédit est essentielle à la gestion du risque dans une entreprise de prêt à la consommation. Elle permet aux prêteurs d'optimiser leurs décisions de prêt, ce qui se traduit par une meilleure expérience client et une économie d'entreprise saine. Il existe des modèles pour aider à gérer le risque, mais il est possible d'en créer de meilleurs qui peuvent surpasser ceux qui sont actuellement utilisés.

American Express est une société de payement qui travaille à l'échelle mondiale. Premier émetteur de cartes de payement au monde, elle offre à ses clients l'accès à des produits, des connaissances et des expériences qui enrichissent leur vie et contribuent à leur succès commercial.

Dans cette compétition, il nous faut appliquer nos compétences de Machine Learning pour prédire les défauts de payement. Plus précisément, il faut exploiter un ensemble de données à l'échelle industrielle pour construire un modèle d'apprentissage automatique qui défie le modèle actuel en production. Les ensembles de données de formation, de validation et de test comprennent des données comportementales chronologiques et des informations anonymes sur le profil des clients.

b. Aperçue des données :

L'ensemble de données contient des caractéristiques de profil agrégées pour chaque client à chaque date de relevé. Les caractéristiques sont rendues anonymes et normalisées, et sont classées dans les catégories générales suivantes :

- D_* = variables de délinquance
- S_* = variables de dépenses
- P_* = variables de payement
- B_* = variables de solde
- R_* = variables de risque

Les caractéristiques suivantes étant catégorielles :

['B_30', 'B_38', 'D_114', 'D_116', 'D_117', 'D_120', 'D_126', 'D_63', 'D_64', 'D_66', 'D_68', 'S_2'].

c. Objectif :

L'objectif est de prédire la probabilité qu'un client ne rembourse pas le montant du solde de sa carte de crédit à l'avenir, sur la base de son profil client mensuel. La variable binaire **target** est calculée en observant la fenêtre de performance des 18 mois avant le dernier relevé de carte de crédit, et si le client ne paye pas le montant dû dans les 120 jours suivant la date de son dernier relevé, il est considéré comme étant en défaut de payement.

d. Métrique d'évaluation :

La métrique d'évaluation pour cette compétition, notée M, est la moyenne de deux mesures de classement : le coefficient de Gini normalisé, G, et le taux de défaut capturé à 4%, D.

$$M = 0,5(G + D)$$

Le taux de défaut capturé à 4% est le pourcentage d'étiquettes positives (défauts) capturées dans les 4% des prédictions les mieux classées, et représente une statistique de sensibilité/rappel. Pour les deux sous-métriques G et D , un poids de 20 est attribué aux étiquettes négatives pour tenir compte du sous-échantillonnage. Cette métrique a une valeur maximale de 1.

II. Importation et exploration des données :

Les DataFrames mis à disposition par American Express étant lourds, des utilisateurs Kaggle en ont proposé des versions compressées plus légères. Nous avons ainsi utilisé une de ces versions afin de ne pas saturer la mémoire RAM de notre machine.

a. Valeurs manquantes :

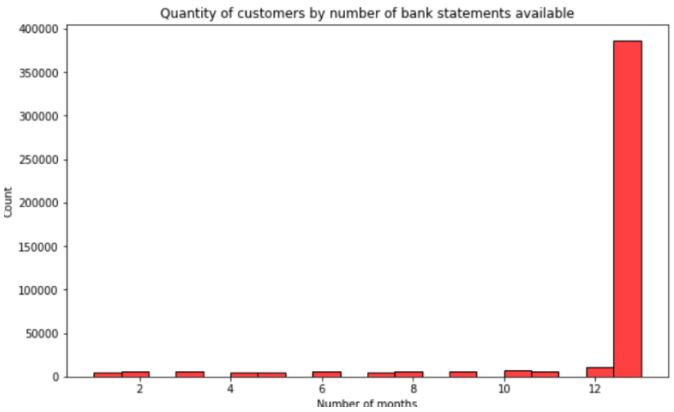
Une fois les données chargées et utilisables, une première étape a été de rechercher les éventuelles valeurs manquantes.

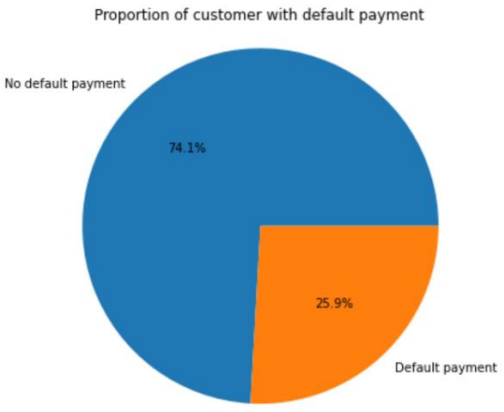
	Number of missing values	% of missing values
D_87	5527586	99.930000
D_88	5525447	99.890000
D_108	5502513	99.480000
D_110	5500117	99.430000
D_111	5500117	99.430000
B_39	5497819	99.390000
D_73	5475595	98.990000
B_42	5459973	98.710000
D_135	5336752	96.480000
D_138	5336752	96.480000
D_137	5336752	96.480000
D_134	5336752	96.480000
D_136	5336752	96.480000
R_9	5218918	94.350000
B_29	5150035	93.100000
D_106	4990102	90.210000
D_132	4988874	90.190000
D_49	4985917	90.140000

Il s'est avéré que le jeu de données contenait un très grand nombre de valeurs manquantes. Toutefois, en gardant en tête que certaines variables sont liées à des évènements de non payement ou de fraude, évènements relativement rares à l'échelle d'une banque, on peut considérer certaines valeurs manquantes comme des indicateurs d'évaluation des clients. De plus, les données étant agrégées, en regroupant les données après chaque période de relevés bancaires, certaines informations peuvent évoluer d'une période à l'autre, créant ainsi de nouvelles valeurs manquantes.

Se pose alors le problème du traitement des valeurs manquantes. Après quelques échanges avec d'autres membres Kaggle participant à la compétition et la lecture de discussions sur la question présentes dans les onglets de la compétition, le choix a été fait de tester des modèles de classification en opérant différents traitement des valeurs manquantes. Nous présenterons les différentes méthodes de traitement des valeurs manquantes un peu plus tard.

b. Attention portée aux clients :



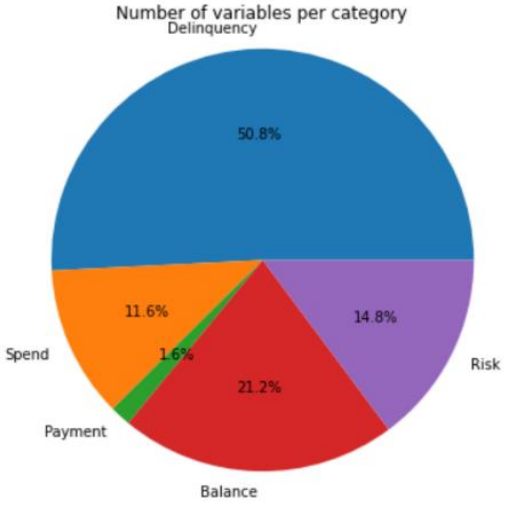


Les données concernent 458913 clients distincts, pour la quasi-totalité desquels on dispose des données bancaires des 13 derniers mois. En liant la valeur de la variable target, indiquant la présence ou non de défaut de payement, on se rend compte que plus de 25% des clients ont eu des

défaut de payement, bien plus que ce que l'on supposait plus haut. Cependant, cette proportion n'est peut-être pas révélatrice de la situation générale. On peut imaginer qu'une proportion plus importante de clients en défaut de payement a été incorporée dans le jeu de données dans le but d'obtenir de meilleurs résultats lors de l'entraînement des modèles de classification.

c. Distribution des variables :

Les variables présentes dans le jeu de données étant déjà normalisées, il était difficile de tirer de l'information à partir de leurs distributions. En revanche, il était toujours possible de rechercher d'éventuelles corrélations, entre les variables d'une même catégorie d'une part, et entre les variables et la target d'autre part. La conclusion générale de cette étude est qu'il ne semble pas exister de tendance particulière au niveau des corrélations, et qu'il serait donc préférable de garder un maximum de variables.



III. Modèles de prédiction :

Les données mises à disposition sont déjà séparées en jeu d'entraînement et jeu de test. Cependant, les données étant conséquentes, nous nous sommes limités à 10% des données pour tester et comparer les différents modèles dans le but d'accélérer les calculs et ne pas saturer la mémoire. En plus du temps de calcul et du score Accuracy, nous avons utilisé la métrique fournie par la compétition pour comparer les différents modèles.

a. Préparation des données :

Après avoir récupéré aléatoirement 10% des données et encoder les variables catégorielles, nous avons procédé à trois types de traitement des valeurs manquantes :

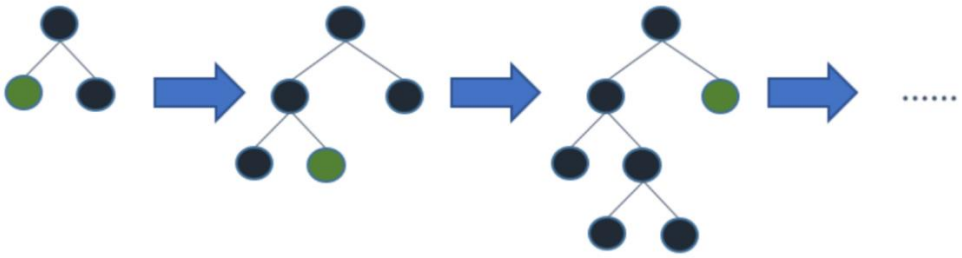
1. Conserver toutes les valeurs manquantes et laisser les algorithmes effectuer les traitements
2. Supprimer toutes les variables contenant des valeurs manquantes
3. Supprimer les variables contenant plus de 30% de valeurs manquantes, et remplacer les autres valeurs manquantes par la valeur moyenne dans le cas d'une variable numérique, ou la valeur la plus fréquente dans le cas d'une variable catégorielle.

b. Modèles utilisés :

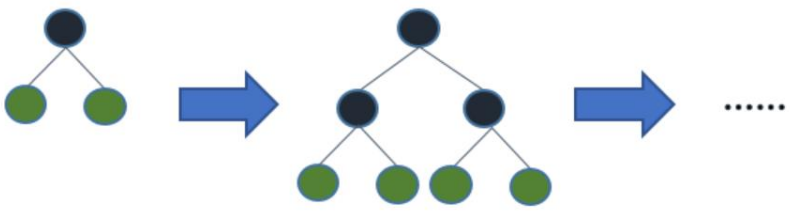
Nous avons cherché des algorithmes de classification binaire capables de traiter automatiquement les valeurs manquantes. Nous avons retenu les deux modèles suivants :

1. LightGBM (LGBM) est un framework de gradient boosting basé sur des arbres de décision. Il est conçu pour être distribué et efficace avec les avantages suivants :
    - Vitesse d'entraînement plus rapide et efficacité accrue
    - Utilisation réduite de la mémoire
    - Meilleure précision
    - Prise en charge de l'apprentissage parallèle et par le GPU
    - Capacité à traiter des données à grande échelle

    Le LightGBM fait croître l'arbre de décision verticalement tandis que les autres algorithmes d'apprentissage basés sur les arbres le font généralement horizontalement. Cela signifie que le LGBM fait croître l'arbre en fonction des feuilles (Leaf-wise tree growth), alors que les autres algorithmes le font croître en fonction des niveaux (Level-wise tree growth).



Leaf-wise tree growth



Level-wise tree growth

2. CatBoost a été développé par l'entreprise russe Yandex. CatBoost signifie Categorical Boosting parce qu'il est conçu pour fonctionner parfaitement sur des données catégorielles.

Voici quelques caractéristiques de CatBoost, qui le distinguent de tous les autres algorithmes de boosting :

- Haute qualité sans réglage des paramètres
- Prise en charge des caractéristiques catégorielles
- Version GPU rapide et évolutive
- Amélioration de la précision en réduisant l'overfitting
- Prédictions rapides
- Fonctionne bien avec moins de données

L'algorithme CatBoost est intéressant à utiliser lorsque les données sont très hétérogènes, c'est-à-dire avec beaucoup de variabilité, des types différents ou encore une quantité importante de valeurs manquantes.

c. Résultats :

1) En conservant toutes les valeurs manquantes :

	LGBM	CatBoost
computation time	80.144679	139.162659
metric score	0.504866	0.506867
accuracy score	0.880013	0.880465

2) En supprimant toutes les variables contenant des valeurs manquantes :

	LGBM	CatBoost
computation time	35.913448	73.578138
metric score	0.460894	0.466395
accuracy score	0.863951	0.865388

3) En complétant une partie des valeurs manquantes :

	LGBM	CatBoost
computation time	67.846078	122.341453
metric score	0.495818	0.502689
accuracy score	0.877356	0.878947

Dans les trois cas, les deux modèles donnent des résultats similaires, mais le LGBM est presque deux fois plus rapide.

En ce qui concerne les données, la suppression de toutes les variables avec des valeurs manquantes ne semble pas être la meilleure idée car elle réduit la qualité de la prédiction. Certes, la différence est faible, mais nous n'avons utilisé que 10% des données disponibles. Pour les deux autres méthodes, nous obtenons des résultats proches, un peu plus longs pour les données non traitées.

Nous restons toutefois convaincus que les variables de délinquance, qui contiennent de nombreuses valeurs manquantes, peuvent révéler de petits détails qui sont très importants pour indiquer un défaut de payement. Même si cela n'améliore pas la qualité générale du modèle, la prise en compte de toutes les variables permettrait peut-être d'éviter certains non-remboursements ou fraudes et donc d'éviter des conséquences dramatiques compte tenu du rôle central des banques dans nos sociétés.

Par conséquent, le modèle final retenu pour la mise en production est le LGBM entraîné sur le jeu de données original dans lequel on a conservé toutes les variables et toutes les valeurs manquantes.