**NOTE METHODOLOGIQUE**

**Introduction**

Le risque de crédit se définit comme la probabilité de perte financière liée à un défaut de remboursement du crédit octroyé par une institution financière à un emprunteur. Il est aujourd’hui, l’une des priorités des sociétés financières qui accordent de plus en plus de prêts bancaires sous plusieurs formes à leurs clients. Ce risque peut entraîner des conséquences assez importantes pour les institutions financières dont la dette n’a été remboursée. Pour cette raison, un nombre important de sociétés financières usent de méthodes très efficaces afin d’évaluer la situation de leurs clients avant tout octroi de crédit. Parmi ces méthodes, figurent les modèles statistiques d’octroi de crédit par scoring dont fait l’objet notre analyse.

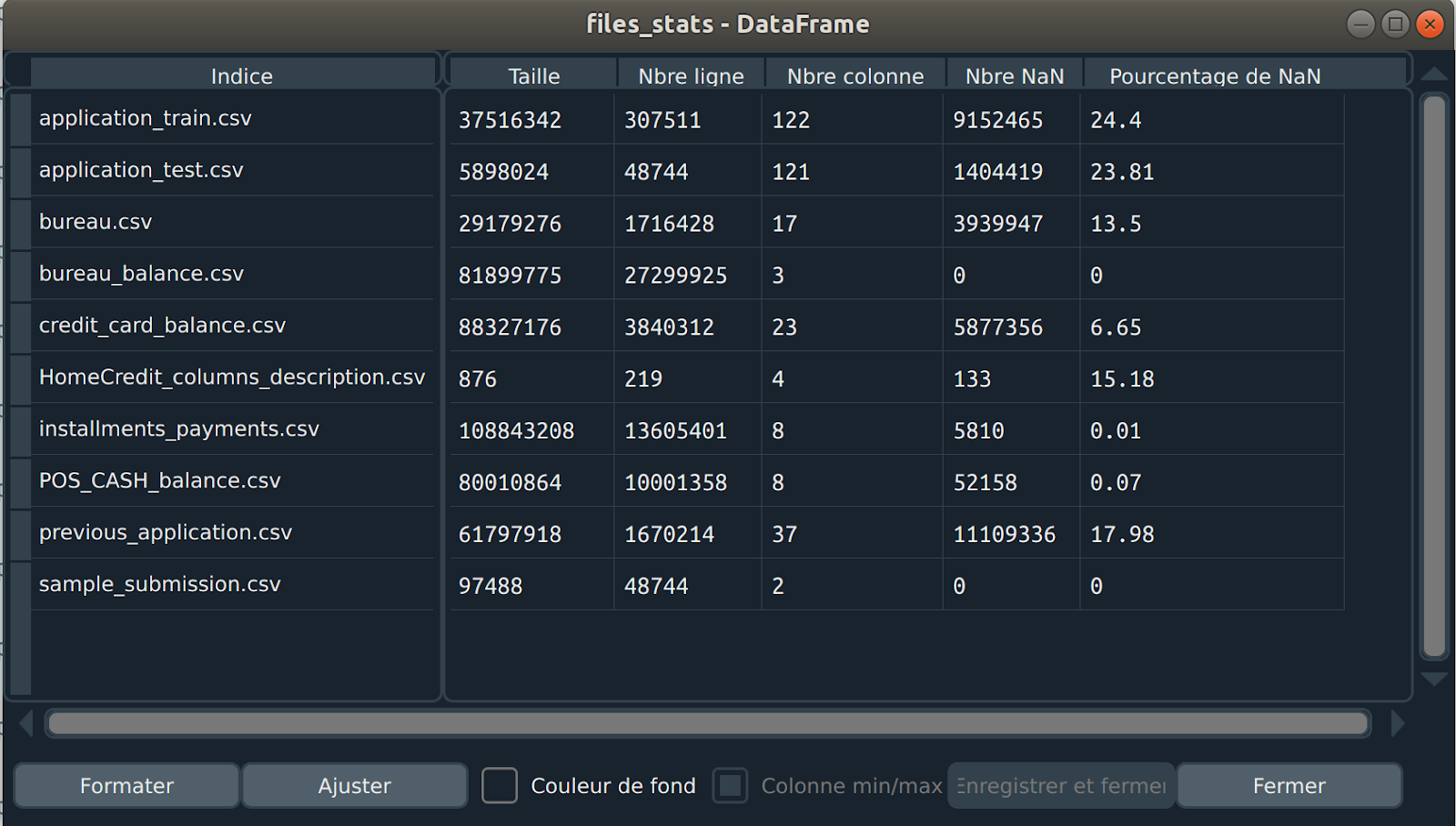
Ce travail s’inscrit dans le cadre du Projet 7 du parcours Data scientist intitulé : « Implémentez un modèle de scoring ». Il a pour objectif d’une part de concevoir un modèle de scoring de la probabilité de défaut de paiement d’un client et d’autre part, de développer un dashboard d’aide à la prise de décision des gestionnaires de crédit. Il s’articule autour de quatre point principaux :

* La description des données et de la méthodologie de travail ;
* Le choix de la métrique d’évaluation ;
* L’interprétabilité du modèle ;
* Les limites et améliorations possibles.

1. **Données utilisées et méthodologie d’entrainement des modèles**
   1. **Données utilisées**

Les données utilisées pour ce travail proviennent de la base de données Kaggle (<https://www.kaggle.com/c/home-credit-default-risk/data>). Elles sont constituées d’un ensemble de dix (10) fichiers de données contenant l’historique des demandes de prêt de la société financière « Prêt à dépenser ». On peut citer en autres les informations générales des clients (âge, sexe, situation familiale, revenus, etc…) et des informations relatives au crédit des clients (montant du crédit, type de crédit, date de demande du crédit, etc…). Dans cette base de données, deux (02) fichiers principaux contiennent les données d’entraînement et les données de test. Le fichier de données d’entraînement contient 307 511 clients et 122 variables dont la variable cible « TARGET » avec la valeur 0 pour client en règle et 1 pour client non solvable.

Quelques statistiques effectuées sur l’ensemble de ces fichiers de données sont présentées sur la Figure 1. Elles détaillent les informations sur le nombre de ligne, le nombre de colonne et le pourcentage de valeurs manquantes dans chaque fichier.

Figure 1 : Synthèse des statistiques de l’ensemble de la base de données

* 1. **Méthodologie d’entraînement des modèles**

Nous abordons dans cette section, notre approche méthodologique. Cette démarche constitue l’ensemble des techniques et moyens mise en œuvre afin de répondre au problème posé. Pour le Data scienstist, cette approche se doit d’être rigoureuse et doit respecter toutes les étapes entrant dans le processus de traitement d’un projet de data science. Pour ce faire, nous présentons dans cette section, les différentes étapes de traitement d’un projet de data science allant du prétraitement des données jusqu’à la modélisation de celles-ci.

* + 1. *Le prétraitement des données*

Le prétraitement des données est l’étape pendant laquelle les données brutes collectées sont nettoyées et structurées avant tout utilisation dans un modèle de Machine Learning. Elle a pour objectif d’éliminer les impuretés (doublons, valeurs manquantes, valeurs aberrantes, etc…) dans la base de données. Elle s’accompagne d’une étape de « Feature Engineering » qui consiste à partir des variables disponibles de déterminer celles qui sont pertinentes pour la modélisation.

Pour notre analyse, notre choix s’est porté sur le kernel Kaggle suivant : <https://www.kaggle.com/willkoehrsen/start-here-a-gentle-introduction>. Quelques statistiques sur les données numériques et catégorielles ont été calculées afin d’apporter de nouvelles variables. Comme certaines variables ne disposent pas de valeurs pour certains clients, une méthode de traitement des valeurs manquantes a été appliquée. Elle consiste à remplacer ces valeurs par la médiane de la distribution de chaque variable à l’aide de la méthode **SimpleImputer(strategy=”median”)**. Ensuite, nous avons appliqué un encodage des variables via **LabelEncoder()**. Ces différentes étapes ont été à la fois appliquée sur les deux fichiers principaux de données (les données d’entraînement et les données test). Enfin, ces fichiers ont été fusionnés pour les besoins du dashboard. Ce fichier final est constitué de 356 255 clients et de 122 variables.

Par ailleurs, l’analyse de la variable cible dans le jeu de données d’entrainement révèle un déséquilibre des classes avec 92% de clients en règle contre 8% de clients en défaut de paiement. Ce déséquilibre peut avoir des conséquences considérables sur les algorithmes de Machine Learning se basant sur le taux de précision (ou l’accuracy) pour construire leur modèle. Etant donné que la majorité des clients appartient à une seule classe (clients en règle), l’utilisation d’un modèle naïf peut conduire à une surestimation de la classe majoritaire dans sa prédiction. Pourtant, dans ce type de cas la classe minoritaire est celle qui est la plus intéressante à étudier. C’est pourquoi, il nécessaire de réduire ce déséquilibre des classes avant le passage à l’étape de la modélisation.

Plusieurs techniques existent. Nous pouvons citer entre autres, les méthodes de ré-échantillonnage (sous-échantillonnage et sur-échantillonnage), la pondération des classes, l’ensemble Learning, etc… En termes de technologies existantes, nous pouvons citer la méthode SMOTE, ADASYN, etc… Pour notre analyse, nous avons opté pour la méthode SMOTE (en anglais Synthetic Minority Over-Sampling Technique). Cette méthode consiste à sur-échantillonner les données associées à une classe en se basant sur les plus proches voisins.

* + 1. *Sélection et entraînement des algorithmes de Machine Learning*

Différents modèles de Machine Learning existent de nos jours et sont de plus en plus complexes. Dans notre cas d’étude, nous somme avons un problème de classification supervisée binaire. C’est pourquoi notre choix s’est porté sur trois modèles : La régression logistique, le modèle XGBoost et le modèle Random Forest Classifier.

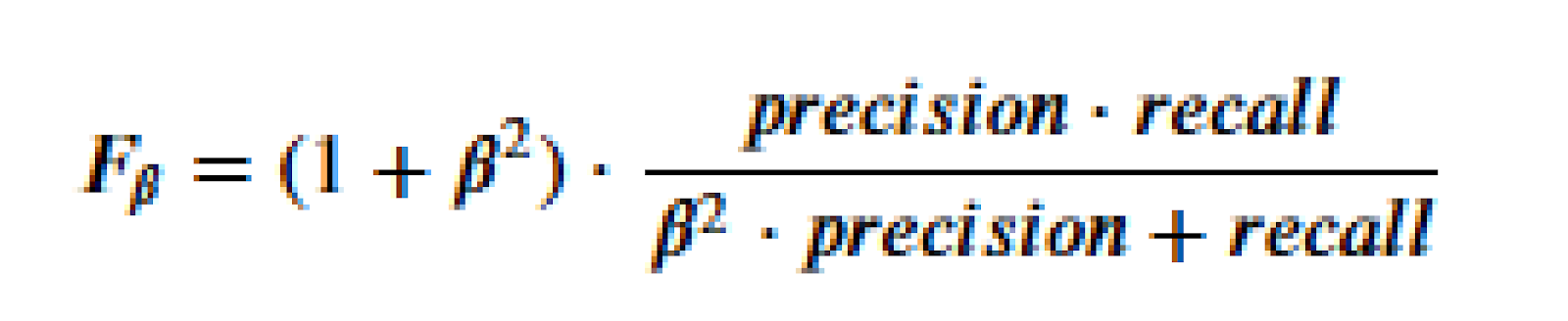
Notre démarche de modélisation a consisté dans un premier temps à réduire le déséquilibre des classes issues du jeu de données d’entraînement à travers l’algorithme SMOTE dans sa version par défaut. Le jeu données obtenu contient 50% de clients solvables et 50% de clients non solvables. Ensuite, ces données ont été séparées en partie d’entraînement (80% de l’échantillon) et en données de validation soit 20% de l’échantillon d’entraînement.

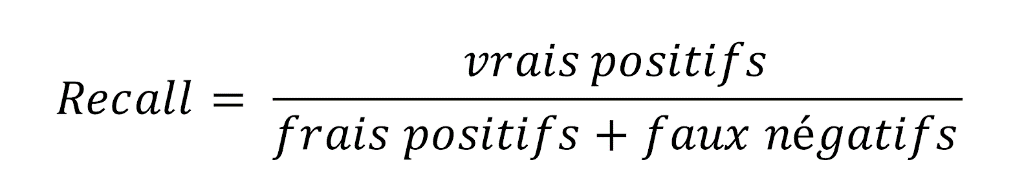
Dans un second temps, les modèles choisis ont été entraîné sur une grille de recherche suivant une combinaison des paramètres que nous avons définis. Les modèles ont finalement été évaluer à partir une métrique d’évaluation judicieusement choisie afin d’en retenir que le meilleur modèle pour la prédiction.

1. **Choix de la métrique d’évaluation**

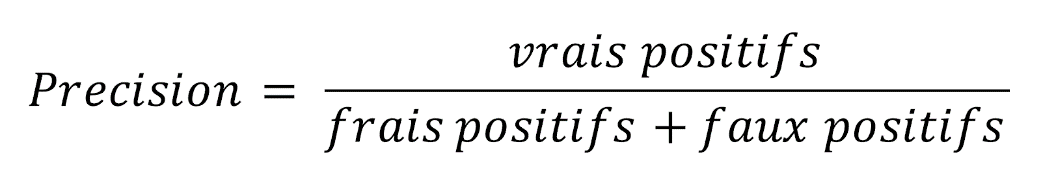
L’accuracy n’étant pas adapté pour les cas où l’on a un déséquilibre des classes, il devient indispensable d’explorer d’autres métriques d’évaluation des modèles. Cependant, cette métrique doit être cohérente avec les objectifs du métier et pertinente pour répondre à sa problématique. Dans ce cas d’étude, il s’agit d’un modèle de prédiction d’un défaut de paiement ou pas. Il est clair que pour une société financière les clients à risque constituent une source de perte d’argent et que la société ne doit pas se priver de clients potentiellement solvables. Quelle solution adopter afin d’obtenir une rentabilité ?

La réponse à cette question consiste à limiter la prédiction de client non solvables prédits comme solvables (faux négatifs) et maximiser la prédiction des clients solvables prédits comme solvables (vrais positifs). Pour notre étude, il s’agit de trouver une fonction qui optimise ces deux critères en privilégiant la prédiction des vrais positifs. Cette fonction est le F1 score. Sa formulation est donnée par la relation suivante :



Où le recall est donnée par la relation :

Et la précision par la relation :



1. **L’interprétabilité du modèle**

Les modèles de Machine Learning deviennent de plus en plus complexes. Plus leur complexité est élevée, plus ils sont précis qu’un modèle peu intelligent. L’interprétabilité est la capacité pour un humain à comprendre les raisons de la décision d’un modèle. Ce critère est devenu importants pour diverses raisons. Nous pouvons citer les raisons scientifiques, éthiques, et législatifs. Sur le plan scientifique, nous avons la nécessité pour le Data scientist de connaitre les variables qui influencent sont modèles et de vérifier la cohérence de ses résultats. Sur le plan éthique et législatif, personne ne doit faire l’objet d’une décision uniquement fondée sur le résultat d’un traitement automatisé. Plusieurs méthodes sont utilisées : LIME, ADASYN, SHAP, …

Pour ce travail, nous avons utilisé la méthode LIME pour interpréter localement les modèles. L’algorithme LIME (en anglais, Local Interpretable Model-agnostic Explanations) est un modèle agnostique, interprétable et simple localement. Son implémentation se fait en deux étapes : une première étape où l’algorithme génère des nouvelles données en se basant sur le voisinage proche de l’individu à expliquer. Dans la deuxième phase, le modèle LIME entraîne un modèle plus simple sur les prédictions du modèle complexe qu’on cherche à interpréter.

1. **Les limites et les améliorations possibles**

La modélisation effectuée ici dans notre analyse ce base sur les données issues des deux princioaux fichiers de la base de données et donc limites les nombre d’informations pouvant permettre à l’algorithme de mieux apprendre les caractéristiques des clients. Il serait plus judicieux d’utiliser le reste des fichiers disponibles dans la base de données afin de créer de nouvelles variables pertinentes pour la modélisation.