# Rapport Machine Learning

## Présentation Synthétique

Dans la première partie, après avoir importé le fichier (train\_projet) sur Python, nous avons appliqué un séparateur pour avoir un bon affichage. Ensuite, nous nous rendus compte qu’il y avait des colonnes sans données donc nous avons cherché ces valeurs manquantes. Dès lors qu’il y avait trop d’information manquant, nous avons supprimé la colonne pour éviter les aberrations. Après, on numérise les variables en changeant leur type (les variables ne sont plus des objets, mais des int32). Le but de cette partie était de garder les données qui étaient le plus viable possible et de les numériser pour pouvoir les réutiliser facilement.

Dans la deuxième partie Data Visualisation, on commence par afficher les graphiques pour chaque variable. Par la suite, on modélise la loi gamma puis on calcul la kurtosis et le skewness et on donne la matrice de corrélation. Enfin, on affiche de nouveaux tableaux à partir de la base de départ en enlevant les valeurs manquantes.

Dans la troisième partie Modélisation, on refait la même chose que précédemment en modélisant les variables dans un autre format puis on prend les variables qui sont déterministes. Ensuite, on cherche les meilleurs paramètres, nous avons décidé de prendre n\_estimator égal à 10 pour éviter que le programme tourne pendant trop longtemps. Après, on prend les variables qui ont le plus d’impact et on secoue les données pour avoir des données aléatoires. (La compréhension de la suite du code sera dans la partie II)

## Description technique utilisé pour étudier la base

On sépare la base en deux (une base d’apprentissage et une base de validation). Dans la base apprentissage, on évalue les prédictions, c’est-à-dire la précision, le recall, le f1-core et le support.

Dans la partie Optimisation des Paramètres, on prend les paramètres d’avant puis on fait tourner la machine un grand nombre de fois avec des paramètres différents. Le but est de trouver les meilleurs paramètres. Puis on refait les prédictions avec ces nouveaux paramètres. Donc on sait dorénavant, quelles sont les meilleurs paramètres va faire en sorte qu’un individu peut rembourser ou non le prêt.

On peut donc maintenant passer à la base test en important les données du fichier test\_projet. On rechange encore le type pour qu’il ne soit plus considéré comme des objets. Enfin, on applique avec nos hyperparamètres, puis on lui demande de créer un fichier datatest\_projet qui détermine si la personne a remboursé ou non.

## Synthèses des résultats et limites

Comme on peut le voir sur les deux graphiques provenant du dossier datatest\_projet, plus de 70% des individus qui sont capables de rembourser leur prêt avec les hyperparamètres déterminés précédemment. Cependant, cette méthode possède plusieurs limites. Tout d’abord, la première limite vient du fait qu’on ne puisse pas avoir un n\_estimator trop grand car sinon le programme prendrait trop de temps. La seconde, vient du fait qu’il est très difficile pour le programme de trouver les bons hyperparamètres et qu’il prend également beaucoup de temps.