NOTE METHODOLOGIQUE

Nature du projet :

Il s'agit d'un projet de scoring pour lequel nous disposons d'une base de données avec un label associé à chaque enregistrement, ce qui permet d'appliquer des techniques d'**apprentissage supervisé.** L'analyse consiste à prédire la classe d'appartenance de chaque observation, celle-ci pouvant prendre 2 valeurs : il s'agit donc d'un problème de **classification binaire.**

Séparation des données d'entrainement et des données test :

Les données ont été séparées en 2 datasets avec stratification : un jeu d'entrainement représentant 75 % du total et un jeu de test représentant 25 % du total. La stratification a permis de s'assurer que les classes à prédire soient présentes dans les mêmes proportions que dans le jeu de données initial.

Feature engeneering/Preprocessing

De nouvelles variables ont été créées à partir des tables application_test et bureau. D'autres, non pertinentes, ont été supprimées. Le nombre final de variables retenues pour la modélisation est de 74.

La moyenne de chaque variable a été imputée aux valeurs manquantes des variables numériques, la catégorie « non_renseignée » à celles des variables catégorielles.
Les variables numériques ont ensuite été standardisées (pour aboutir à une moyenne de 0 et un écart-type de 1) tandis que les variables catégorielles ont été binarisées (création d'une colonne par catégorie, remplie avec un 1 ou un 0 en fonction de la présence ou non de la catégorie chez l'observation)

Déséquilibre des classes :

On constate que le nombre de valeurs du label est déséquilibré : 92% de « 0 » et 8% de « 1 ».
Les « 1 » identifient les dossiers avec défaut de paiement.
Une première modélisation test a été effectuée à l'aide d'une régression logistique : 99.997 % de la classe 0 a été correctement classifiée contre seulement 0.08 % de la classe 1.
Un modèle simple sans aucun retraitement est donc très efficace pour détecter la classe majoritaire (« 0») mais ne parvient pas à détecter les « 1 ».
Or, malgré l'absence de précisions de la part de la société concernant les risques associés à chaque type d'erreur, on peut supposer que les conséquences de la non détection d'un « 1 » (validation d'une demande de crédit qui aboutit à un défaut de paiement) sont plus couteuses que le refus d'un dossier d'un client de classe « 0 » (refus d'une demande de crédit d'un dossier qui n'aurait pas posé de problème).
Il est donc nécessaire de veiller à bien détecter les « 1 »

Méthodes pour améliorer la qualité de prédiction de la classe minoritaire :

3 approches ont été testées pour tenter de corriger le déséquilibre des classes : 2 méthodes de sampling (oversampling, undersampling) et 1 méthode de modification des règles de pénalisation :

- L'oversampling est une méthode qui consiste à générer de nouvelles observations appartenant à la classe minoritaire. SMOTE (Synthetic Minority Over-sampling Technique) est une technique d'oversampling populaire. Celui-ci génère de nouvelles observations artificielles à l'aide de la méthode KNN : chaque nouvelle observation est créée à un endroit aléatoire situé sur le

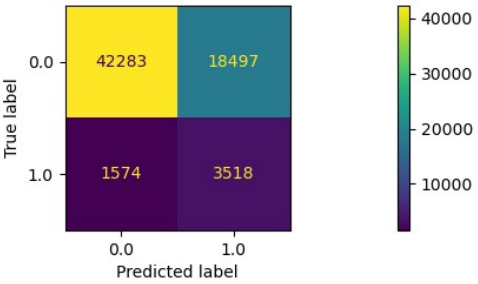
segment reliant 2 observations voisines de la classe minoritaire. Lors de ce projet leur nombre a été augmenté jusqu'à égaler celui de la classe majoritaire.

- L'undersampling consiste à supprimer des enregistrements de la classe majoritaire de manière aléatoire . Pour ce projet leur nombre a été diminué jusqu'à égaler celui de la classe majoritaire.

- La modification des régles de pénalisation permet d'ajuster les poids des observations de chaque classe dans la fonction de coût. Il a été ici fait en sorte que le poids de chaque classe devienne inversement proportionnel à sa fréquence.

Les métriques d'évaluation

La matrice de confusion



La matrice de confusion récapitule les résultats des prévisions par rapports aux classes réelles, dont découlent plusieurs indicateurs. Les 4 parties sont :

True Positives TP (True label =1 & predicted label = 1) : Observations de la classe 1 correctement prédites

True Negatives TN (True label =0 & predicted label = 0) : Observations de la classe 0 correctement prédites

False Positives FP (True label=0 & predicted label=1) : Observations de la classes 0 incorrectement prédites. Appelés aussi erreur de type I

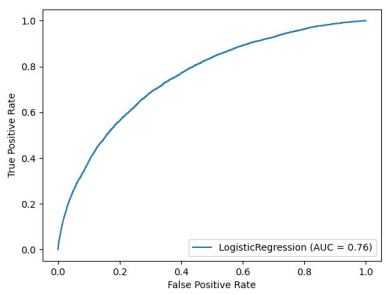
False Negatives FN (True label = 1 & predited label=0) :
Observations de la classe 1 incorrectement prédites. Appelés aussi erreur de type II

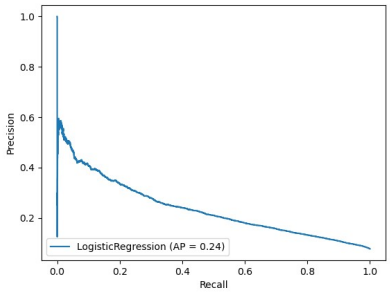
Le choix des indicateurs

Un choix par élimination a été fait. Les métriques d'évaluation qui accordent la même importance aux différents types d'erreur, comme l'accuracy(TP+TN/(TP+TN+FP+FN)), semblent contre indiquées. En effet, étant donné l'important déséquilibre des classes, un classifieur qui prédirait en permanence des 0 (qui accorderait toutes les demandes de prêts) obtiendrait un excellent score d'accuracy (92%).

La précision (TP/(TP+FP)) ne remplit absolument pas l'objectif visé étant donné qu'elle valorise davantage les modèles qui se trompent peu au détriment de ceux qui détectent tout.
Le recall(TP/(TP+FN)) peut sembler plus adapté au premier bord car il permet d'évaluer la capacité à détecter les défauts de paiement. Cependant un modèle qui refuserait l'ensemble des prêts aurait un score de recall parfait mais serait tout aussi inutile que le modèle précédent. La précision de la détection, bien que secondaire doit tout de même rentrer en ligne de compte pour évaluer la qualité de prédiction.

La courbe ROC, qui compare le taux de TP et celui de FP et la courbe precision- recall, qui compare la precision et le recall, présentent l'avantage de faire une comparaison pour chaque seuil plutôt qu'à une seule valeur de seuil donnée, ce qui les rend plus complet.





Cependant dans les deux cas un modèle qui détecte très mal la classe minoritaire peut obtenir un très bon AUC score en excellent dans la détection de la classe majoritaire.

Le F1 score, moyenne harmonique entre la précision et le recall, ne se focalise que sur les performances de prédiction de la classe « 1 » et semble être adaptée à la situation. Un modèle ne peut pas obtenir de F1 score élevé en compensant une prédiction médiocre des 1 par une excellente prédiction des 0. Le Fbeta score avec beta=2 est une variante qui accorde un poids plus important au recall qu'à la précision (détecter un maximum de 1 est plus important que de louper des 0). Il semble être un choix encore plus pertinent. C'est la métrique qui a été retenue.

La regression logistique

La régression logistique est un algorithme utilisé pour la classification, qui applique une fonction logistique à une combinaison linéaire de coefficients pour déterminer la probabilité d'appartenance d'une observation à une classe, les coefficients étant déterminés en maximisant la vraisemblance des observations d'un jeu d'entrainement. Un de ses avantages est l'interprétabilité de ses résultats via l'analyse des coefficients.
La fonction à optimiser est la suivante :

$$\max_{\beta \in \mathbb{R}^{p+1}} \quad \sum_{i=1}^{n} \left(y^i \log \left(\frac{1}{1+e^{-(\beta^\top x^i)}} \right) + (1-y^i) \log \left(1 - \frac{1}{1+e^{-(\beta^\top x^i)}} \right) \right)$$

Elle n'a pas de solution explicite mais celle-ci étant concave une approximation peut être trouvée à l'aide de la méthode du gradient.

Algorithme du gradient :

L'algorithme du gradient s'applique aux fonctions dérivables. On part d'un point initial aléatoire x0 et on fixe un seuil de tolérance. L'algorithme du gradient définit une suite d'itérés x1,x2… jusqu'à ce qu'un test d'arrêt soit satisfait. Il passe de x(k) à x(k+1) par les étapes suivantes :
1-Calcul du gradient de la fonction f au point x(k)
2-Test d'arrêt : arrêt du processus si le gradient est inférieur au seuil de tolérence
3-Calcul du pas par une recherche linéaire sur f en x(k) le long de la direction du gradiant (pour les recherches de maximum) ou inverse du gradiant (pour les recherches de minimum)
4-Ajout du pas à x(k) pour obtenir x(k+1)

Le choix des hyper-paramètres :

Un GridsearchCV 3CV a été utilisé pour déterminer les meilleurs hyper-paramètres de régularisation : la force et le type de régularisation, avec le score Fbeta comme critère de sélection

Le modèle avec un coefficient C de 0.01 et une régularisation Ridge a produit le meilleur résultat, avec un score Fbeta de 0.415. C'est également celui qui a produit le meilleur ROC AUC (0.758) et le meilleur precision-recall AUC (0.235)

Interprétabilité du modèle

A chaque variable est associé un coefficient

La somme des (coefficients*valeurs) et de l'intercept donne un score qui correspond au ln de la cote de voir un défaut de paiement (label « 1 ») se produire.
exp(score) donne la cote et cote/(1+cote) donne une prédiction de la probabilité qui correspond à la contribution au ln de la cote.
Plus le produit coef * valeur est négatif plus il diminue la probabilité de l'évènement 1
Plus le produit coef * valeur est positif plus il augmente la probabilité de l'évènement 1

Exemple à 3 variables (dossier 412932) :

Variable	Coefficient	Valeur	Impact
Montant du crédit	0.276	-0.659	-0.182
Montant du bien	-0.377	-0.554	0.209
Durée du crédit	-0.208	-0.227	0.047

Y Intercept = -0.742

Impact total = -0.742 -0.182 + 0.209 + 0.047 = -0.668
Exp(-0.668) = 0.513
0.513/(1+0.513)=0.339

Le dossier a une probabilité de défaut de paiement de 33.9 % et prendra le label « 0 »

Les limites et les améliorations possibles

La régression logistique a été utilisée pour ce projet car il était nécessaire d'avoir un modèle interprétable mais d'autres modèles plus opaques pourraient produire de meilleurs résultats.

Nous sommes partis du principe que les faux positifs étaient plus tolérables que les faux négatifs mais il ne s'agit que d'une hypothèse qu'il serait nécessaire de confirmer auprès de la société. Peut-être que les assurances couvrent les risques de défauts de paiement de manière satisfaisante.

Il est ici question d'accepter ou de refuser une demande de prêt en se basant uniquement sur le risque de défaut de paiement mais d'autres critères stratégiques rentrent probablement en compte (profil client recherché, prêt très ou peu rentable...) Ainsi face à une demande de prêt à haut potentiel une société pourrait être tentée d'accepter un risque de défaut de paiement allant jusqu'à 60 %. Inversement un dossier qui présente peu d'interêt pourrait se voir refusé dès 40 %.

La liste des variables brutes utiles n'est peut-être pas exhaustive. De la même manière il est peut-être possible d'aller plus loin dans le feature engineering et ainsi créér de nouvelles variables. Des variables trop complexes pourraient toutefois nuire à l'interprétabilité.

Certains critères basés sur le relationnel ou sur des caractéristiques très spécifiques de certains clients peuvent rentrer en ligne de compte pour l'acceptation d'un prêt tout en étant impossibles à retranscrire sous forme de variables.

Contrairement aux prêts acceptés pour lesquels on peut dénombrer à posteriori les cas de défauts de paiement, il est impossible en cas de refus de prêt de savoir avec certitude si un incident se serait produit ou non. La distinction faux postitifs / vrais positifs reste donc une suppostition.