nexialog CONSULTING

NEXIASEARCH

MÉTHODES DE RÉÉQUILIBRAGE DES CLASSES

en classification supervisée

Merwan CHELOUAH

THINK SMART ACT DIFFERENT

TABLE DES MATIÈRES

Introduction

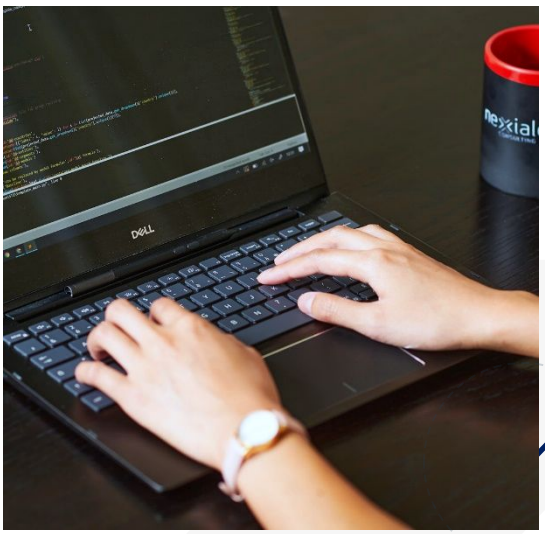
L'un des obstacles auxquels font face les institutions financières est de décider des candidats potentiels à qui accorder une ligne de crédit, en évaluant les risques de non-remboursement d'un prêt. Pour une décision aussi cruciale, les données démographiques et financières antérieures des débiteurs sont importantes afin de construire un modèle automatisé de prédiction du score de crédit basé sur un classificateur d'apprentissage automatique. Cependant, de nombreuses tâches de classification binaire ne disposent pas d'un nombre équilibré d'observations dans chaque classe, notamment lorsque la distribution des classes est asymétrique.

Un tel déséquilibre peut affecter notre modèle d'apprentissage, lorsqu'il a tendance à ignorer la classe minoritaire par exemple. Ce problème est particulièrement critique quand la classe minoritaire est celle qui nous intéresse le plus. En particulier, le déséquilibre peut affecter notre algorithme d'apprentissage lorsqu'il ignore complètement la classe minoritaire. En effet, l'algorithme dispose de peu d'exemples de la classe minoritaire pour apprendre.

Il est donc biaisé vers la population majoritaire et produit des prédictions potentiellement moins robustes qu'en l'absence de déséquilibre. En outre, dans des domaines d'applications sensibles comme pour la détection de pathologies dans le domaine médical, le scoring ou la détection de la fraude, il est extrêmement important de classer correctement les instances minoritaires. L'une des questions les plus importantes est de savoir si la prise en compte de ces interactions conduit à des gains prédictifs, c'est-à-dire à une meilleure évaluation des risques de crédit.

En tout état de cause, ces techniques pourraient potentiellement générer des gains de productivité importants dès lors qu'elles rendent obsolètes un certain nombre d'opérations de prétraitement des données, visant à capturer ces non-linéarités. L'objectif ici est de réaliser une revue des principales méthodes de rééquilibrage de classes ainsi que de proposer une étude de cas permettant d'illustrer ces méthodes ainsi que leur efficacité.

Dans un premier temps, nous présenterons les différentes solutions au problème de classes déséquilibrées, nous verrons ainsi dans quelle mesure la sélection des métriques de performance du modèle est essentielle dans le jugement de ces méthodes. Dans un deuxième temps, nous présenterons plusieurs cas d'usage de ces approches. Nous verrons ainsi qu'aucune méthode de prétraitement ne surpasse systématiquement les autres dans sa contribution à une meilleure performance lorsque le degré de déséquilibre des classes varie. Nous verrons également qu'une approche métier couplée à l'application de pratiques d'ajustement est essentielle dans la gestion de ces cas de figure.



I. Les principales approches

Les méthodes classiques

Certaines approches classiques, basées sur le rééchantillonnage, cherchent à augmenter la fréquence de la classe minoritaire ou à diminuer celle de la classe majoritaire. Ceci est fait dans le but d'obtenir approximativement le même nombre d'instances pour les classes.

Undersampling (Sous-échantillonage) : Il s'agit de réduire le ratio de la variable catégorielle cible calculé précédemment afin de réduire la part de la catégorie prédominante dans le jeu de données. Généralement, la réduction des effectifs de la classe majoritaire est réalisée aléatoirement en supprimant une certaine quantité, choisie arbitrairement, d'échantillons de cet effectif. On qualifie généralement cette approche de Random Undersampling.

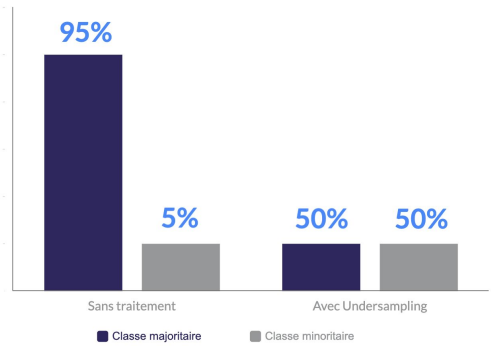


Figure 1. Schématisation de l'undersampling

Oversampling (Sur-échantillonnage) : De la même manière, l'objectif est de réduire la différence dans l'occurrence des modalités, mais cette fois, il s'agit d'augmenter la quantité de modalités en sous-nombre en créant des copies aléatoires d'échantillons de la classe minoritaire, on parle plus généralement de Random Oversampling.

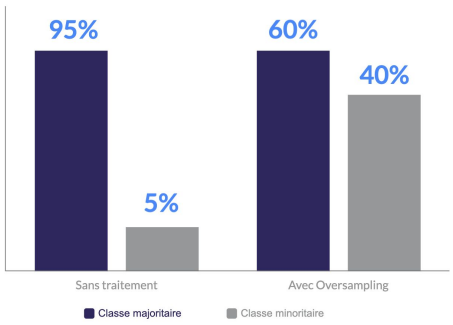


Figure 2 – Schématisation de l'oversampling

Ces deux méthodes de rééquilibrage sont plus communément appelées méthodes de "rééchantillonnage naïf" car elles ne supposent rien sur les données.

Elles sont donc simples à mettre en œuvre et rapides à exécuter, ce qui est souhaitable pour les ensembles de données très vastes et complexes.

L'article de Bee Wah Yap et al. (2013) [1] conclut que l'utilisation du Random Undersampling plutôt que du Random Oversampling est préférable si la volumétrie des données le permet, étant donné que l'oversampling, par la réplication des données présentes dans le dataset initial, augmente le risque de surapprentissage (overfitting).

A l'inverse, un des écueils du *Random Undersampling* selon Alberto Fernandez et al. (2018) [2] est que cette méthode peut écarter des données critiques pouvant représenter une source d'information importante pour le modèle, voire agir comme des données "pivots".

Néanmoins, il n'en reste pas moins que ces deux méthodes sont tout de même très répandues dans l'industrie.

Bien que ces deux méthodes restent assez classiques, d'autres méthodes plus avancées de génération d'échantillons existent.

Les méthodes avancées

Synthetic Minority Oversampling Technique

Cette méthode de suréchantillonnage plus communément dénommée SMOTE se concentre sur la classe minoritaire, qui est augmentée par la création d'instances "synthétiques". Il s'agit de l'un des algorithmes les plus utilisés pour améliorer les performances des classificateurs appliqués à des données non équilibrées. L'algorithme fournit un ensemble de règles simples pour générer de nouvelles données "synthétiques".

Bien que chaque nouvelle donnée synthétique soit construite à partir de ses parents (d'une donnée existante et l'un de ses plus proches voisins), la donnée générée n'est jamais une copie exacte de l'un de ses parents. SMOTE est un algorithme qui s'appuie sur la technique du K-voisin le plus proche.

L'idée est de créer une relation linéaire entre les modalités représentant la modalité du sous-ensemble et à construire des points aléatoires sur la base de cette représentation linéaire.

Etape 1 : Dans l'ensemble de classes minoritaires A, pour chaque $x \in A$, les k plus proches voisins de x sont obtenus en calculant la distance euclidienne entre x et chaque autre échantillon de l'ensemble A.

Etape 2 : Le taux d'échantillonnage N est fixé en fonction du ratio de déséquilibre. Pour chaque x appartenant à l'ensemble minoritaire A, N observations (c'est-à-dire x_1, x_2 ...x_n) sont sélectionnés aléatoirement parmi ses k plus proches voisins, et ils construisent l'ensemble A_1.

Etape 3 : Pour chaque $x_k \in A_1$ avec $k \in N$, la formule suivante est utilisée pour générer une nouvelle observation :

$$x' = x + rand(0, 1) * |x - x_k| \quad (1)$$

Où rand(0, 1) représente un nombre aléatoire entre 0 et 1.

La principale force de SMOTE réside dans le fait qu'il permet de créer de nouvelles données synthétiques à partir de nos données observées, alors que les méthodes "classiques" dupliquent les données existantes dans l'ensemble de données.

L'inconvénient des méthodes classiques est qu'elles peuvent suggérer que certaines règles (interactions) capturent la majeure partie de l'appartenance de la modalité minoritaire. Cependant, l'un des inconvénients de l'utilisation de l'algorithme SMOTE est qu'il nécessite des temps de calcul plus longs lorsque la taille de la base d'apprentissage augmente.

Une alternative serait d'utiliser une combinaison de sous-échantillonnage classique et d'algorithmes de génération plus avancés. Par ailleurs, l'un des problèmes liés à l'algorithme SMOTE qui apparaît de manière récurrente dans la littérature des méthodes d'ajustement est le problème lié aux points aberrants.

En effet, si l'un des points représentant la classe en sous-effectif apparaît comme aberrant en étant situé à proximité d'un point de la classe majoritaire, la synthèse va alors utiliser un point de cette classe majoritaire afin de construire d'autres points synthétiques.

Une des solutions à ce problème est d'ignorer ces points dans le cas où les K plus proches voisins sont effectivement identifiés comme des points de la classe majoritaire.

Certaines variantes, plus adaptés à des variables explicatives de types "qualitatives" existent, on peut citer par exemple SMOTE NC ou Borderline SMOTE (Hui Han, Wen-Yuan Wang, and Bing-Huan Mao, 2005 [3])

Adaptive synthetic sampling

L'Adaptive synthetic sampling (ou ADASYN) est un algorithme qui a été proposé par Haibo He, Yang Bai, Edwardo A. Garcia, Shutao Li. (2008) [4]

L'idée principale est d'utiliser une distribution pondérée pour différents groupes de la classe minoritaire en fonction de leur niveau de difficulté d'apprentissage. Plus les données sont difficiles à apprendre, plus le nombre de données synthétiques générées est élevé.

L'approche améliore l'apprentissage par rapport aux distributions de données de deux manières : elle réduit le biais introduit par le déséquilibre des classes et déplace de manière adaptative la frontière de classification par rapport aux données en trop faible effectif.

L'algorithme ADASYN est basé sur l'algorithme SMOTE, à la différence qu'il prend en compte une certaine variance entre les points plutôt que d'utiliser les corrélations linéaires entre les observations de sous-effets.

En effet, ADASYN utilise la densité de nos observations, ce qui signifie que dans l'espace de nos observations, plus de données seront générées dans le sous-espace où la densité de nos données de sous-effets est la plus faible. La densité est calculée à partir du nombre d'observations appartenant à la classe majoritaire autour du point duquel on veut générer de nouveaux échantillons synthétiques.

Etape 1 : Dans l'ensemble de classes minoritaires A, pour chaque exemple $x_i \in A$, on trouve les K plus proches voisins à base de la distance euclidienne dans un espace à n dimensions. On calcule ensuite le ratio r_i défini comme suit :

$$r_i = \Delta_i / K \quad (2)$$

où Δ_i est le nombre d'exemples dans les K voisins les plus proches de xi qui appartiennent à la classe majoritaire

Etape 2 : Normaliser r_i afin qu'il puisse être considéré comme une densité de distribution.

Etape 3 : Calculer le nombre d'exemples de données synthétiques qui doivent être générés pour chaque x_i en fonction de la répartition cible et de la densité r_i. Soit,

$$N = r_i * G$$

où N représente le nombre de points à générer autour du point x_i de densité r_i et G représente le nombre de points total à générer.

Etape 4 : A partir du nombre d'exemple de données synthétiques qui doivent être générés autour de chaque x_i, appliquer le même traitement que SMOTE.

Tout comme pour SMOTE, un des principaux inconvénients de ce type d'algorithmes génératifs basés sur des représentations linéaires des données est que cette linéarité peut amener à des situations où les données synthétisées de manière générique sont assez loin dans l'espace des données originelles.

Majority Weighted Minority Oversampling Techniques

Les algorithmes de Majority Weighted Minority Oversampling Techniques (appelés MWMOT) localisent les échantillons minoritaires ayant seulement comme k_1-voisins les plus proches les échantillons de la classe majoritaire. Cela permet de trouver les échantillons minoritaires qui sont à la limite entre les vecteurs de l'espace de classe. Cet ensemble d'échantillons est appelé : ensemble minoritaire filtré. Ensuite, pour chaque échantillon de l'ensemble minoritaire filtré, cet algorithme trouve les k_2 plus proches voisins de la classe majoritaire. Ensuite, il agrège tous ces échantillons dans un ensemble d'échantillons uniques. Cet ensemble est nommé : ensemble majoritaire limite. A partir de chaque échantillon de cet ensemble, il suffit de trouver les k_3 plus proches voisins de la classe minoritaire.

Ce nouvel ensemble d'échantillons uniques de la classe minoritaire est appelé : ensemble minoritaire informatif. En utilisant cet ensemble minoritaire informatif, nous devons calculer la probabilité de sélection pour chaque échantillon.

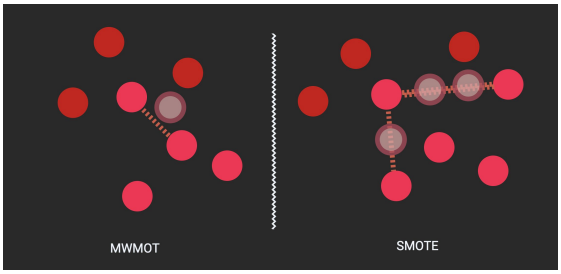
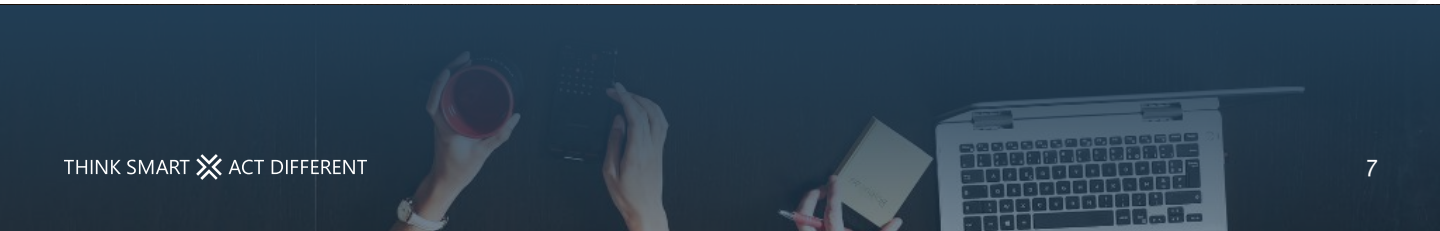


Figure 3. *Ici, les points rouges foncés représentent les échantillons de la classe majoritaire, les roses ceux de la classe minoritaire. Les points avec bordures sont les nouveaux échantillons générés.*

Un des inconvénients des algorithmes basés sur le principe MWMOT est qu'ils sont bien souvent très conservateurs dans la génération de nouveaux échantillons par rapport à SMOTE notamment, et d'autres techniques de suréchantillonnage en générant des données assez proches dans l'espace des données initiales.



II. Les métriques

Choisir une métrique de performance est bien souvent un exercice délicat, mais ce choix est d'autant plus déterminant dans le cas d'une classification supervisée avec des classes déséquilibrées. Retenir une mauvaise métrique pour évaluer nos modèles peut conduire au choix d'un mauvais modèle ou, dans le pire des cas, de mal estimer la robustesse ainsi que les performances attendues de notre modèle.

Précision

La Précision s'avère être une métrique très utile dans le cas de classes présentant un déséquilibre, notamment lorsqu'on attache davantage d'importance à l'erreur de type I qu'à l'erreur de type II. En effet, elle reflète extrêmement bien le taux de vrais positifs.

$$Precision = \frac{TruePositif}{TruePositif + FalsePositive} \quad (4)$$

Recall

Le Recall (ou sensibilité) indique la capacité du modèle à prédire correctement tous les positifs et la précision indique la capacité du modèle à prédire uniquement les positifs. Ce sont des métriques pertinentes à utiliser lorsque l'on s'intéresse principalement à la classe des positifs.

Cependant, elles sont asymétriques. Autrement, si on inverse la classe positive et la classe négative, elles n'auront pas la même valeur.

$$Recall = \frac{TruePositive}{TruePositive + TrueNegative} \quad (5)$$

ROC Curve

La métrique ROC-AUC correspond à l'aire sous la courbe obtenue par la courbe ROC.

Pour obtenir la courbe ROC nous tenons compte des valeurs réelles dans l'intervalle [0, 1] produites par la modélisation d'un modèle de classification binaire, par exemple, pouvant être interprétées comme une probabilité que l'échantillon soit positif. Nous pouvons dire que si la sortie du modèle est supérieure à 0,5, alors l'échantillon est positif et négatif sinon, mais 0,5 n'est pas le seul choix pour le seuil de classe, il s'agit du seuil arbitrairement choisi par tout prédicteur.

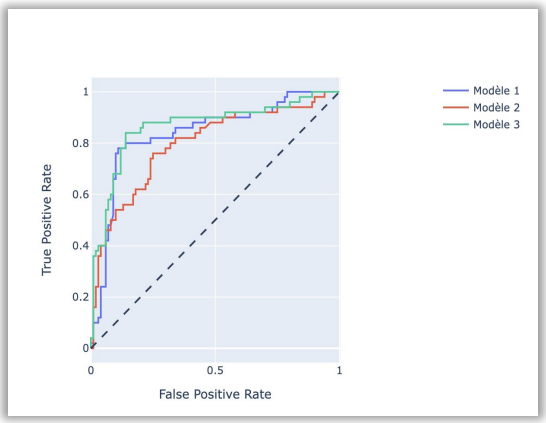


Figure 4. Une représentation de plusieurs courbes ROC

Sur cette figure, la courbe ROC montre le compromis entre la sensibilité (ou TPR) et la spécificité (1 - FPR). Les classificateurs qui donnent des courbes plus proches du coin supérieur gauche indiquent une meilleure performance.

En principe, un classificateur aléatoire devrait donner des points situés le long de la diagonale (FPR = TPR). Plus la courbe se rapproche de la diagonale à 45 degrés de l'espace ROC, moins la classification est précise.

Cette courbe sert également à comparer différents classifieurs. Ainsi, une courbe ayant davantage de valeurs élevées a donc une aire sous la courbe plus grande et le classifieur fait donc moins d'erreurs. Bien que généralement efficaces, la courbe ROC peut être optimiste en cas de déséquilibre grave de la classe, en particulier lorsque le nombre d'exemples dans la classe minoritaire est faible.

C'est pour cela qu'on opte bien souvent pour son homologue qui est la PR-curve et qui s'appuie cette fois-ci sur les composantes precision-recall qui sont très intéressantes pour le traitement de classes minoritaires en prenant à la fois en compte les taux de faux positifs et de faux négatifs (voir [5]).

En effet, l'utilisation de la ROC Curve peut amener à des situations où l'on cherche à maximiser le recall en souhaitant optimiser la classification de la classe minoritaire. Cela aura pour effet de pénaliser les décisions liées à la classe majoritaire et fera exploser la proportion de faux négatifs.

Bien souvent, cela est moins grave en ce qui concerne la gestion des spams ou dans le domaine bancaire ainsi que dans le traitement du churn. En ce sens, on préférera classifier des clients comme étant plus risqués avec une règle de décision plus conservatrice, ce qui est beaucoup moins souhaitable dans le domaine médical.

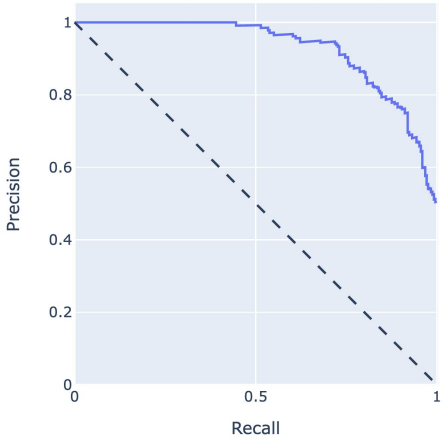
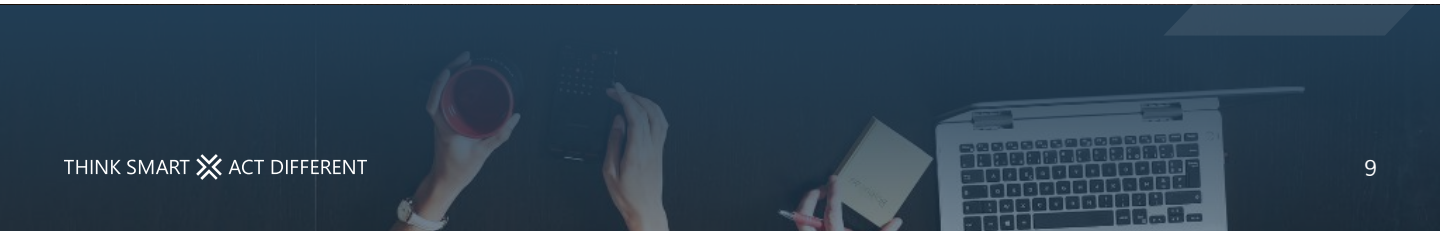


Figure 5. Une représentation de la courbe PR

Sur cette figure, une aire élevée sous la courbe représente à la fois un recall et une précision élevés, où une précision élevée correspond à un faible taux de faux positifs, et un recall élevé correspond à un faible taux de faux négatifs. Des scores élevés pour les deux indiquent que le classificateur renvoie des résultats très pertinents (précision élevée), ainsi qu'une majorité de résultats positifs (recall élevé). Un classifieur random aura tendance à obtenir des résultats proches de la première diagonale.

La PR-Curve mesure donc très bien le trade-off Précision-Recall. Pour chaque seuil de décision du modèle de régression logistique, cette courbe donne le couple précision-recall correspondant. On peut ainsi sélectionner le seuil le mieux adapté au cadre d'étude considéré.



F1 - Score

De même, le F1-score est très intéressant quand on veut avoir à la fois un bon recall et une bonne précision. Cependant, ce score fait la moyenne harmonique de la précision et du recall de manière égale et parfois le métier est plus intéressé par un aspect du problème que par l'autre.

Dans le cas de la classification multi-classe, le F1-score existe aussi sous d'autres formes. Il est facile de voir la similitude entre la précision et le recall. Alors que le recall mesure le degré auquel un modèle classifie correctement tous les vrais positifs, la précision mesure le degré auquel un modèle classifie correctement les positifs par rapport à toutes les observations qu'il classe comme positives. Ils tiennent donc compte de différentes erreurs de classification d'un modèle.

La justification du choix de cette métrique réside dans son utilisation répandue lors de la comparaison de modèles et son intuitivité lorsqu'elle est comparée au recall : si le F1-score d'un modèle est supérieur à son recall, alors la précision est supérieure au recall, et vice versa.

Le F1-score est donc une mesure intéressante pour comparer les résultats de différents modèles.

$$F - score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (6)$$

III. Tests et résultats

Mode de traitement

L'objectif est d'implémenter et mesurer les performances de ces algorithmes de rééquilibrage de classes, sur différentes bases de données et pour différents modèles de classification. Les méthodes de rééquilibrage mises en œuvre sont :

1. **Random Undersampling**
2. **Random Oversampling**
3. **SMOTE**
4. **ADASYN**

Dans cette optique, les algorithmes de rééquilibrage ont été appliqués après les différentes étapes de feature engineering requises (nettoyage des données notamment). De plus, l'algorithme de rééquilibrage utilisé a été appliqué uniquement au niveau de l'échantillon d'entraînement afin de ne pas altérer les données de validation et que ces dernières conservent la même distribution que celle des données initiales.

Afin de mesurer la performance des algorithmes et des modèles, la **précision,** le **recall** et le **F1-score** ont été sélectionnés comme principales mesures de performance.

Le recall étant particulièrement important si l'on souhaite limiter la quantité de faux négatifs ce qui est le cas dans le domaine bancaire.

Une K-Fold cross-validation a été réalisée sur les données. Cependant, dans le cas de datasets ayant des classes déséquilibrées et lorsque la distribution est fortement asymétrique, il est probable qu'un ou plusieurs Fold auront peu ou pas d'exemples de la classe minoritaire. Cela signifie que certaines, voire la plupart des évaluations du modèle seront trompeuses, car le modèle ne doit prédire correctement que la classe majoritaire. Il est donc important de réaliser une segmentation qui préserve la distribution déséquilibrée des classes dans chaque Fold. Il s'agit d'une Stratified K-fold cross-validation, qui force la distribution des classes dans chaque fold des données à correspondre à la distribution de l'ensemble complet des données d'apprentissage.

L'objectif étant de calculer les scores pour chacun des modèles sur différentes graines afin de réduire le biais et ensuite d'agréger les scores par une moyenne. En effet, chacun des modèles a été entraîné puis testé en utilisant une Stratified k-fold cross-validation 10 fois, ensuite, une moyenne arithmétique des scores a été réalisée et c'est celle-ci qui figure dans la table de résultats.

Concernant ces différents tests, les principaux modèles retenus sont les suivants :

1. **Algorithme de boosting type LGBM**
2. **Random Forest**
3. **Ridge appliquée à la régression logistique**

Ajustement des modèles et méthodes

Concernant le choix des paramètres de l'oversampling et de l'undersampling, celui-ci a été réalisé par tâtonnement pour chacune des bases de données, il est à noter que ce choix varie en fonction de la taille du dataset et du déséquilibre.

Au niveau des modèles, **les hyperparamètres** ont été choisis arbitrairement et ont été conservés pour l'ensemble des datasets et pour l'application de chacune des approches. Une pénalité **L2** ou « **Ridge** » a été appliquée au modèle de régression logistique.

En outre, en ce qu'il en est du modèle de régression logistique, dans certains cas, l'utilisation des **ROC-Curve** et des **PR-Curve** peut permettre de calculer directement le seuil optimal. Dans d'autres cas, il est possible d'utiliser un **GridSearch** pour ajuster le seuil et localiser la valeur optimale.

Notre approche a été celle de choisir un seuil qui donne le meilleur équilibre entre la précision et le recall, cela revient à optimiser le **F1-score** qui représente comme précisé précédemment la moyenne harmonique des deux métriques.

Cette approche permet de déterminer le seuil optimal en calculant le F1-score pour chaque seuil. On choisit ensuite le seuil pour lequel le F1-score est le plus élevé. C'est ce que l'on peut constater dans la ***Figure 6.***

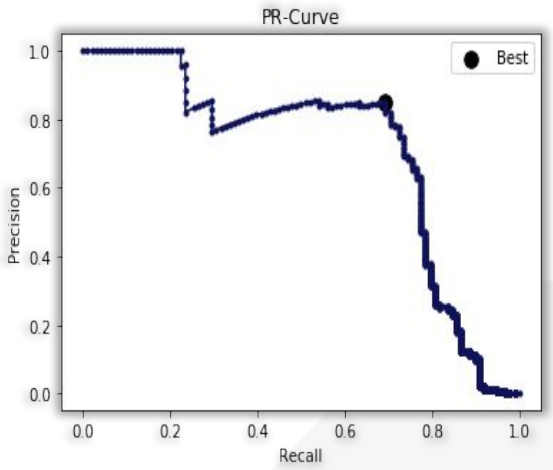


Figure 6. PR-Curve et seuil optimal pour la base crédit card fraud

Une autre approche intéressante est celle basée sur la Statistique J de Youden, qui se calcule comme suit :

$$J - stat = Recall + \frac{true\ negatives}{true\ negatives + false\ positives} - 1 \qquad (7)$$

De la même manière que la méthode précédente, cette approche consiste à calculer cette statistique pour chacun des seuils et à sélectionner le seuil pour laquelle cette statistique est maximale.

L'ensemble des métriques ont été calculées sur les échantillons de **test** et non d'entraînement.

Etude de cas sur différentes bases de données

Les algorithmes de rééquilibrage retenus ont été implémentés sur différentes bases de données présentant des données déséquilibrées.

Nom	Description
Crédit Card Fraud	Transactions étiquetées comme frauduleuses ou authentiques
Adult census data	Prédire si le revenu dépasse 50 000 dollars par an sur la base des données du recensement
Churn Modelling	Prédire si le client va quitter la banque

Figure 7. Description des bases de données

Le dataset credit card fraud contient les transactions effectuées par cartes de crédit en septembre 2013 par des titulaires de cartes bancaires européennes. Pour des raisons de confidentialité, il ne contient que des variables d'entrée numériques qui sont le résultat d'une ACP.

Les seules caractéristiques qui n'ont pas été transformées par l'ACP sont 'Time' et 'Amount'. La variable "Time" représente pour une observation les secondes écoulées entre la transaction et la première transaction du dataset. La caractéristique 'Amount' est le montant de la transaction. La caractéristique "Class" est la variable cible et prend la valeur 1 en cas de fraude et 0 sinon.

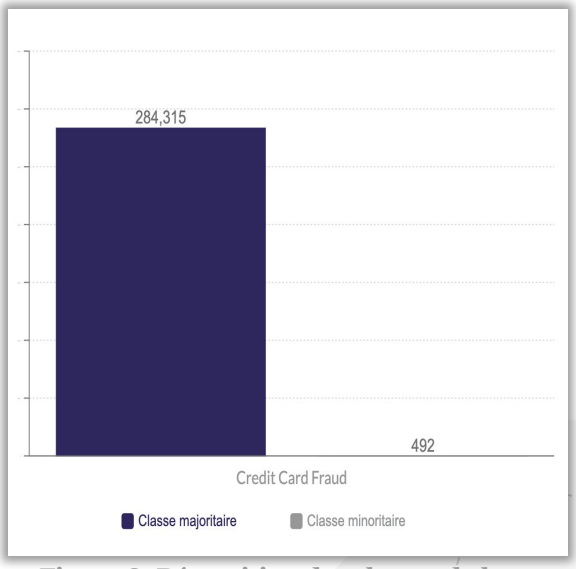


Figure 8. Répartition des classes de la base crédit card fraud

D'après la ***Figure 8***, nous constatons que notre classe de fraudeurs est très déséquilibrée - seulement 0,17% de notre ensemble de données appartient à cette classe minoritaire !

C'est un problème car de nombreux modèles d'apprentissage automatique sont conçus pour maximiser l'accuracy, ce qui, en particulier avec des classes déséquilibrées, peut ne pas être la meilleure métrique à utiliser. Par exemple, si l'objectif était simplement de prédire toutes les transactions non frauduleuses, nous aurions obtenu un score d'exactitude de classification de plus de 99%.

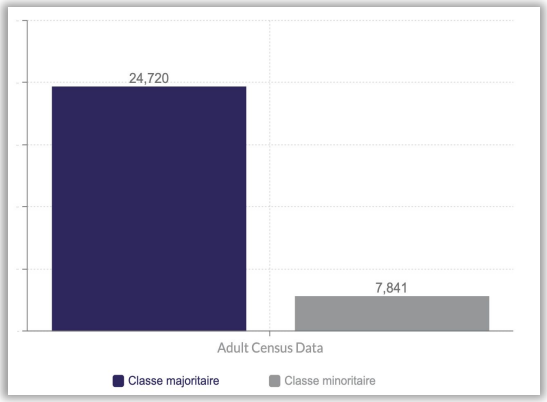


Figure 9. Répartition des classes de la base Adult Census Data

Concernant la base de données Adult Census Data, l'ensemble de données est attribué à Ronny Kohavi et Barry Becker et a été tiré des données du Bureau du recensement des États-Unis de 1994. Il contient des caractéristiques personnels tels que l'âge, le statut marital ou le niveau d'éducation pour prédire si un individu gagnera plus ou moins de 50 000 dollars par an.

Sur la ***Figure 9***, le déséquilibre est beaucoup moins important, les échantillons de classe minoritaire représentant environ 24% des données globales du dataset.

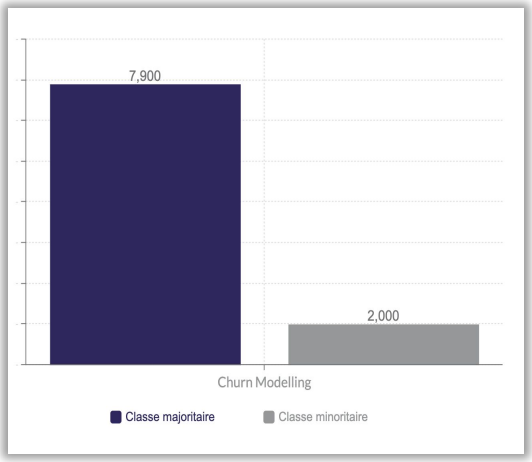


Figure 10. Répartition des classes de la base Churn Modelling

Enfin, concernant le dataset Churn, là encore seulement 20% de notre ensemble de données appartient à la classe cible.

Résultats

	Precision	Recall	F1-Score
Logistic Ridge	0.8253	0.5306	0.6459
Random Forest	0.9625	0.7857	0.8651
LGBM	0.9629	0.7959	0.8715

Figure 11. Métriques de performances sans rééquilibrage pour la base crédit card fraud

	Precision	Recall	F1-Score
Logistic Ridge	0.1028	0.8877	0.1843
Random Forest	0.8829	0.8469	0.8645
LGBM	0.3548	0.8979	0.5086

Figure 12. Métriques de performances avec SMOTE pour la base crédit card fraud

	Precision	Recall	F1-Score
Logistic Ridge	0.1026	0.8865	0.1836
Random Forest	0.8791	0.8421	0.8611
LGBM	0.3519	0.8951	0.5023

Figure 13. Métriques de performances avec ADASYN pour la base crédit card fraud

On constate sur les ***Figure 12*** et ***Figure 13*** que la précision a fortement diminué pour le modèle logistic et le LGBM par rapport à nos résultats sans rééquilibrage (***Figure 11***).

C'est aussi le cas pour le F1-score, c'est une chute mécanique liée à la forte hausse des effectifs de la classe minoritaire, en effet, le F1-score est à la fois lié au recall et à la précision. Ainsi, ici le recall a augmenté tandis que la précision a fortement chuté, le F1-score a donc mécaniquement suivi ce comportement.

Dans le domaine bancaire, on peut être davantage intéressé par les faux négatifs que les faux positifs, cela signifie que le recall nous importe davantage que la précision.

Or ici, et notamment pour le modèle logistique, l'importante diminution de la précision semble remettre en cause l'usage des algorithmes de rééquilibrage tant l'arbitrage précision-recall est largement défavorable à la première métrique.

On peut voir que c'est nettement moins le cas pour le modèle Random Forest où le F1-score se maintient à un niveau relativement proche du niveau du F1-score sans traitement avec un recall plus élevé. On note également que les différences entre les performances de SMOTE et ADASYN sont relativement similaires.

	Precision	Recall	F1-Score
Logistic Ridge	0.0848	0.8979	0.1550
Random Forest	0.9868	0.7653	0.8620
LGBM	0.3901	0.8877	0.5420

Figure 14. Métriques de performances avec Over-sampling pour la base crédit card fraud

Comme le montre la ***Figure 14*** avec l'oversampling classique, on voit qu'en privilégiant le Recall, on obtient de meilleures performances si on considère la régression logistique ou le LGBM.

	Precision	Recall	F1-Score
Logistic Ridge	0.0757	0.9081	0.1398
Random Forest	0.0809	0.9081	0.1485
LGBM	0.0503	0.9285	0.0955

Figure 15. Métriques de performances avec Under-sampling pour la base crédit card fraud

Comme le montre la ***Figure 15***, avec le random undersampling, on obtient de bien meilleurs recall au prix d'une précision encore plus faible, on a un taux de faux négatifs qui s'approche de zéro.

Ainsi, le modèle semble donc très bien prédire les éléments de la classe minoritaire. Cela dit, nous avons un taux de faux positifs qui est très élevé et énormément d'éléments de la classe majoritaire sont mal classifiés.

La réduction de la quantité d'échantillons appartenant à la classe majoritaire rend la tâche d'apprentissage du modèle relativement plus difficile.

Conclusion

Parmi l'ensemble des modèles, les résultats suggèrent qu'il n'existe pas de méthode de prétraitement qui améliore systématiquement les performances de classification en ce qui concerne les métriques dans leur globalité. En revanche, selon la métrique que l'on souhaite privilégier, par exemple le recall ou la précision, il peut être intéressant de retenir une approche plutôt qu'une autre.

Une expertise métier est ainsi à coupler au choix de l'approche avant de considérer son choix.

De même, la performance de ces différentes approches varie grandement selon le modèle d'apprentissage choisi.

De la même manière, il apparaît comme nécessaire de réaliser certains traitements et ajustements de modèle comme une **Stratified K fold Validation** dans la même visée de préserver l'équilibre des distributions et d'avoir un effectif d'échantillons représentatif au sein du dataset de validation ou alors l'optimisation du seuil si un modèle basé sur une régression logistique est de mise.

Annexes & références

Références

1. Bee Wah Yap, Khatijahhusna Abd Rani, Hezlin Aryani Abd Rahman, Simon Fong, Zuraida Khairudin, and Nik Nik Abdullah. An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets. In Proceedings of the first international conference on advanced data and information engineering (DaEng-2013), pages 13–22. Springer, 2014.
2. Alberto Fernández, Salvador García, Mikel Galar, Ronaldo C Prati, Bartosz Krawczyk, and Francisco Herrera. Learning from imbalanced data sets, volume 10. Springer, 2018.
3. Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote : a new over-sampling method in imbalanced data sets learning. In International conference on intelligent computing, pages 878–887. Springer, 2005.
4. Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. Adasyn : Adaptive synthetic sampling approach for imbalanced learning. In 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), pages 1322–1328, 2008.
5. Jason Brownlee. Imbalanced classification with python : Better metrics, balance skewed classes, cost-sensitive learning. Machine Learning Mastery, 2020.

Annexes

	Precision	Recall	F1-Score
Logistic Ridge	0.4029	0.06870	0.1173
Random Forest	0.6910	0.4325	0.5320
LGBM	0.7164	0.4885	0.5809

Figure 16. Métriques de performances sans traitement pour la base Churn

	Precision	Recall	F1-Score
Logistic Ridge	0.3144	0.6921	0.4324
Random Forest	0.6632	0.4910	0.5643
LGBM	0.5091	0.7760	0.6149

Figure 17. Métriques de performances avec Oversampling pour la base Churn

	Precision	Recall	F1-Score
Logistic Ridge	0.3135	0.6870	0.4306
Random Forest	0.4655	0.7048	0.5607
LGBM	0.4781	0.7811	0.5932

Figure 18. Métriques de performances avec undersampling pour la base Churn

Annexes & références

	Precision	Recall	F1-Score
Logistic Ridge	0.4285	0.5648	0.4873
Random Forest	0.6397	0.4834	0.5507
LGBM	0.6313	0.6361	0.6337

Figure 19. Métriques de performances avec SMOTE pour la base Churn

	Precision	Recall	F1-Score
Logistic Ridge	0.4411	0.5343	0.4833
Random Forest	0.6349	0.5089	0.5649
LGBM	0.6315	0.6412	0.6363

Figure 20. Métriques de performances avec ADASYN pour la base Churn

	Precision	Recall	F1-Score
Logistic Ridge	0.6996	0.2530	0.3717
Random Forest	0.7086	0.5744	0.6345
LGBM	0.7890	0.6180	0.6931

Figure 21. Métriques de performances sans traitement pour la base Adult Census Data

	Precision	Recall	F1-Score
Logistic Ridge	0.5603	0.8301	0.6691
Random Forest	0.6655	0.6343	0.6495
LGBM	0.6054	0.8555	0.7090

Figure 22. Métriques de performances avec Oversampling pour la base Adult Census Data

	Precision	Recall	F1-Score
Logistic Ridge	0.5223	0.7996	0.6318
Random Forest	0.5731	0.7722	0.6579
LGBM	0.5942	0.8594	0.7026

Figure 23. Métriques de performances avec Undersampling pour la base Adult Census Data

	Precision	Recall	F1-Score
Logistic Ridge	0.6831	0.6213	0.6507
Random Forest	0.6921	0.5953	0.6400
LGBM	0.7449	0.6499	0.6942

Figure 24. Métriques de performances avec SMOTE pour la base Adult Census Data

	Precision	Recall	F1-Score
Logistic Ridge	0.5928	0.7104	0.6463
Random Forest	0.6788	0.5901	0.6313
LGBM	0.7382	0.6551	0.6942

Figure 25 – Métriques de performances avec ADASYN pour la base Adult Census Data

Nexialog Consulting

STRATÉGIE | ACTUARIAT | GESTION DES RISQUES

Nexialog Consulting est un cabinet de conseil spécialisé en Stratégie, Actuariat et Gestion des risques qui dessert aujourd'hui les plus grands acteurs de la banque et de l'assurance. Nous aidons nos clients à améliorer de manière significative et durable leurs performances et à atteindre leurs objectifs les plus importants.

Les besoins de nos clients et les réglementations européennes et mondiales étant en perpétuelle évolution, nous recherchons continuellement de nouvelles et meilleures façons de les servir. Pour ce faire, nous recrutons nos consultants dans les meilleures écoles d'ingénieur et de commerce et nous investissons des ressources de notre entreprise chaque année dans la recherche, l'apprentissage et le renforcement des compétences.

Quel que soit le défi à relever, nous nous attachons à fournir des résultats pratiques et durables et à donner à nos clients les moyens de se développer.

CONTACTS

Ali Behbahani
Associé, Fondateur
+ 33 (0) 1 44 73 86 78
abehbahani@nexialog.com

Christelle BONDOUX
Associée, Directrice commerciale
+ 33 (0) 1 44 73 75 67
cbondoux@nexialog.com

Adrien Misko
Manager R&D
+ 33 (0) 6 69 27 62 26
amisko@nexialog.com

Paul-Antoine DELETOILLE
Account Manager Senior – Global Markets
+33 (0)1 44 73 75 70
+33 (0)7 64 57 86 69
padeletoille@nexialog.com

Areski COUSIN
Directeur Scientifique
acousin@nexialog.com

www.nexialog.com

Retrouvez toutes nos publications sur Nexialog R&D