
# Big Data and Machine Learning

## Risque de défaillance des entreprises: Identifier les entreprises fragiles économiquement

CY CERGY PARIS UNIVERSITY M2 EDT
RUCHY

# Introduction

Objectif:

- Détecter les entreprises économiquement fragiles à partir de caractéristiques internes et externes.
- Comparer plusieurs modèles de Machine Learning (régression logistique, Random Forest, XGBoost).

Problématique:

Comment identifier précocement le risque de défaillance d'une entreprise à partir de ses signaux économiques et de marché ?

# Revue de littérature

- Altman (1968) introduit le concept du Z-score, un indicateur composite issu de ratios financiers permettant de discriminer les entreprises saines des entreprises en faillite.
- Ohlson (1980) et Zmijewski (1984) généralisent ce concept à travers la régression logistique, qui évalue la probabilité de défaut en fonction de quelques variables comptables clés.
  Ces modèles, bien qu'interprétables et simples, souffrent d'une faible adaptabilité à la dynamique des marchés modernes et à la complexité croissante des données d'entreprise.
- À partir des années 2000, l'essor des techniques de machine learning modifie profondément le champ de la prédiction du risque :
- Les arbres de décision et les forêts aléatoires (Breiman, 2001) permettent une meilleure détection des interactions non linéaires.
- Les méthodes de boosting comme XGBoost (Chen & Guestrin, 2016) améliorent considérablement les performances prédictives.
- Des études comme Gepp et al. (2017) et Zhang et al. (2021) montrent que les modèles d'ensemble surpassent la régression logistique sur la majorité des jeux de données financiers.

# Description du Data set

Le Data Set utilisé dans le cadre de notre projet est sovai/bankruptcy. C'est un ensemble de données publié sur Hugging Face.  Il regroupe des indicateurs économiques, statistiques et financiers d’entreprises afin de prédire leur risque de faillite (bankruptcy).

Chaque ligne du dataset représente une entreprise à une date donnée (ou une observation spécifique liée à son état économique).

Chaque colonne est un indicateur calculé selon différents modèles économiques ou statistiques.

# Description du Data set

| Colonne | Type | Description |
|---|---|---|
| probability | float | Agrégat global de probabilité de défaillance (output d'un modèle de scoring global) |
| Probability_light | float | Score prédit par un petit réseau de neurones léger |
| Probability_convolution | float | Probabilité calculée via un modèle de type CNN (extraction de patterns complexes) |
| Probability_Rocket | float | Mesure basée sur des méthodes Rocket/transformers temporelles |
| Probability_enconder | float | Probabilité issue d'un encodeur d'informations textuelles/structurées |
| Probability_fundamentale | float | ndicateur fondamental basé sur les ratios financiers réels (liquidité, endettement, etc.) |
| volatility | float | Mesure de la volatilité observée sur le titre/secteur lié à l'entreprise |
| multiplier | float | Mesure multiplicative associée à la variation de capitalisation (ex. PE ratio) |
| Sans_market | float | Variable simplifiée d'environnement de marché (0 ou 1 selon la présence de conditions de marché actives) |

# Méthodologie

Nettoyage et échantillonnage:

- Construction de la variable cible:L'objectif étant de prédire si une entreprise est défaillante ou saine, une variable binaire default a été définie à partir de la probabilité agrégée :"df["default"] = (df["probability"] > seuil_metier).astype(int) "  où seuil_metier est un seuil choisi (par exemple 1.5 dans notre script) en concertation avec la logique métier.  Default = 1 : entreprise considérée en situation de défaillance (ou à haut risque)  default = 0 : entreprise considérée comme saine
- Nettoyage et prétraitement: Échantillonnage et équilibrage des classes 1500 /1500
- Séparation des variables explicatives et de la cible:

X : sous-ensemble des colonnes explicatives (features)

y : colonne default

# Méthodologie

- Découpage en jeux d’apprentissage, validation et test
- 60 % entraînement, 20 % validation, 20 % test, avec `train_test_split` et stratification sur `y` pour préserver la proportion de classes
- Standardisation
- Pour la régression logistique (modèle linéaire sensible à l’échelle des variables), une standardisation par StandardScaler a été appliquée :
- apprentissage du scaler sur `X_train` ;
- application à `X_val` et `X_test`.

# Méthodologie

Modélisation

Trois modèles supervisés de classification binaire ont été retenus, représentant trois familles classiques en data science:

Régression logistique

- Modèle de base, linéaire et interprétable.
- Deux versions testées :
- sans prétraitement (variables brutes) ;
- avec standardisation (variables centrées-réduites).

Random Forest

- Modèle d’ensemble basé sur des arbres de décision (bagging).
- Paramétrage initial avec `class_weight="balanced"` pour tenir compte du déséquilibre de classes.
- Optimisation des hyperparamètres principaux (`n_estimators`, `max_depth`) via GridSearchCV (validation croisée 5-fold, scoring basé sur le F1-score).

# Méthodologie

XGBoost (XGBClassifier)

- Méthode de gradient boosting, performante sur les données tabulaires.
- Paramètres choisis pour un bon compromis biais/variance : nombre d’arbres, profondeur maximale, taux d’apprentissage, sous-échantillonnage des lignes et des colonnes.
- Entraîné sur le même jeu d’entraînement que les autres modèles.

# Méthodologie

Interprétation et importance des variables

Pour Random Forest et XGBoost, les importances de variables (`feature_importances_`) ont été analysées afin de déterminer quels indicateurs contribuent le plus à la prédiction du risque :

- identification des probabilités “expertes” les plus déterminantes (light, convolution, rocket, encoder, fondamental) ;
- rôle des indicateurs de marché (`volatility`, `multiplier`, `sans_market`).

Cette étape permet de faire le lien entre les résultats des modèles et une interprétation économique du risque de défaillance.

# Évaluation des modèles

L’évaluation est effectuée sur le jeu de test (données jamais vues pendant l’entraînement ni la validation), en utilisant plusieurs métriques de classification :

- Accuracy : proportion de bonnes prédictions ;
- Précision : parmi les entreprises prédites “défaillantes”, combien le sont réellement ;
- Rappel (Recall) : parmi les entreprises réellement défaillantes, combien sont détectées ;
- F1-score : moyenne harmonique précision/rappel, utile en cas de déséquilibre ;
- ROC-AUC : capacité du modèle à bien classer les positives/negatives pour tous les seuils.

Les résultats sont résumés dans un tableau comparatif et complétés par des visualisations :

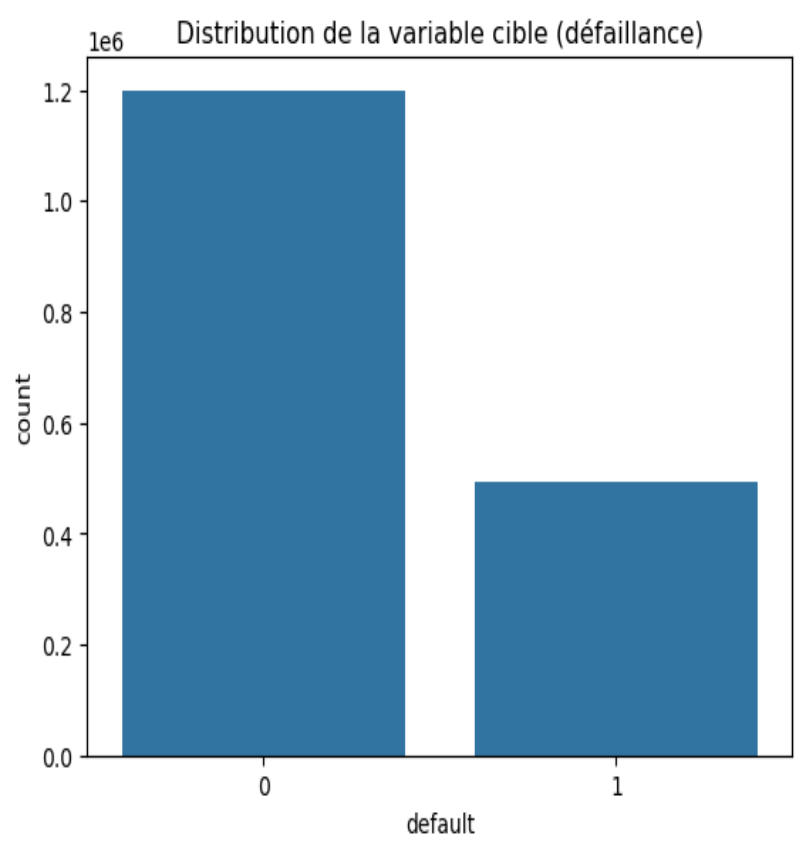
- Matrices de confusion pour chaque modèle (structure des erreurs) ;
- Courbes ROC superposées (comparaison globale) ;
- Analyse de l’impact de la standardisation sur la performance de la régression logistique.

# Résultats

Après l'analyse de notre data set il n'y avait pas de doublons, ni de valeur manquante. Après avoir séparé notre data set en deux classes nous avons étudié la distribution ce qui nous a donnée ce graphe. On observe un déséquilibre au niveau de la dispertion des classes. Par la suite nous avons étudié le model sous deux hypothèse:

I) 3000 observations aléatoires

II) 1500 Observations de chaques classes

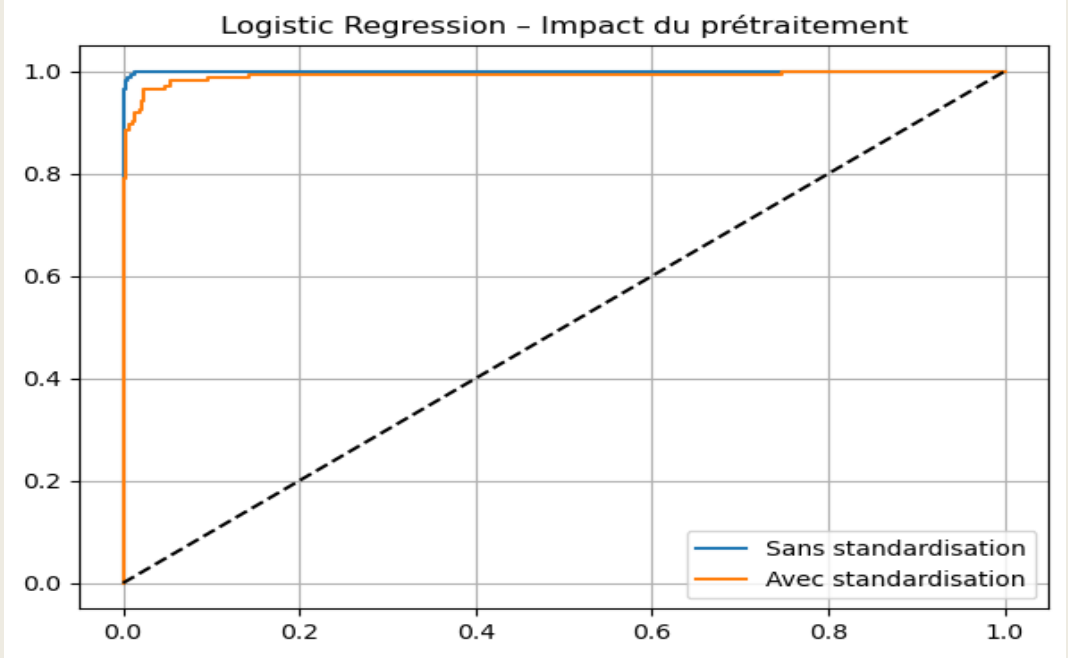


# Résultats

## RÉGRESSION LOGISTIQUE

D'après la courbe Roc de notre model avec normalisation et sans normalisation on remarque que le model est plus performant sans standardisation

## COURBE DE ROC

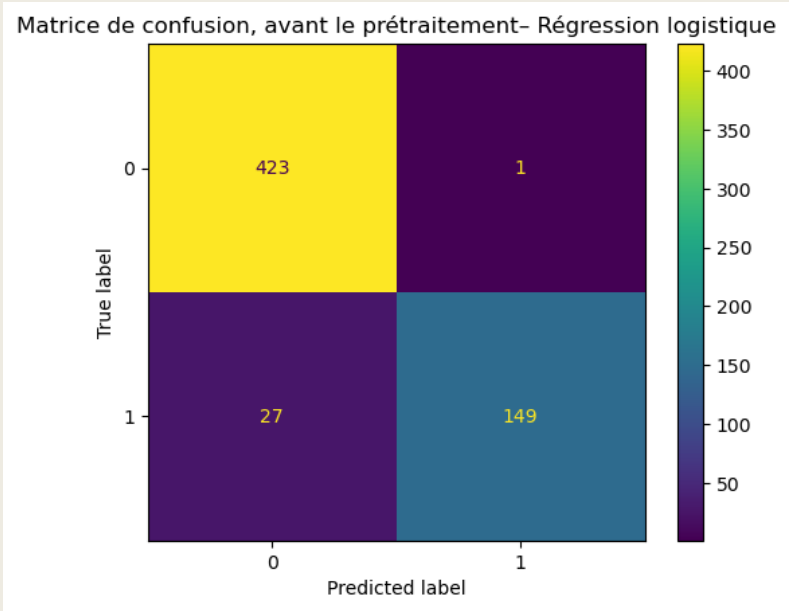


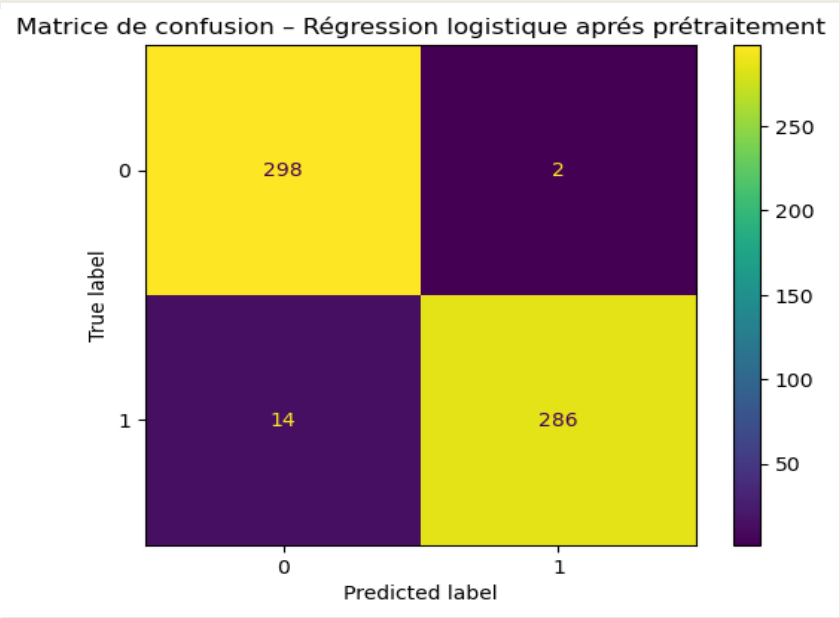
# Résultats

## RÉGRESSION LOGISTIQUE

Le model est excellent pour prédire les entreprises saines , là on a juste très peu d' entreprise défaillante classé saine. Au niveau de la prédiction des entreprise défaillante, le model fait beaucoup d'erreurs. Et on a moin erreur après l'équilibre des classes.

## MATRICE DE CONFUSION



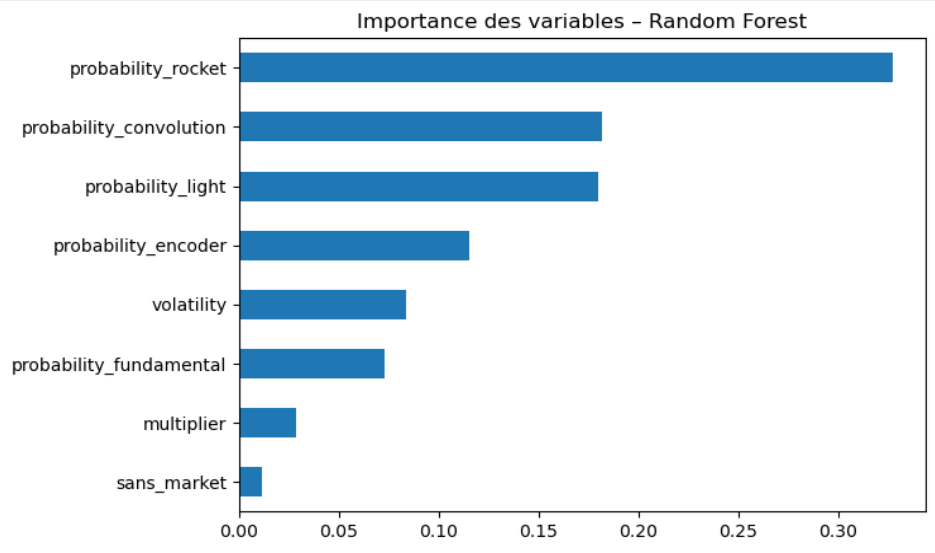



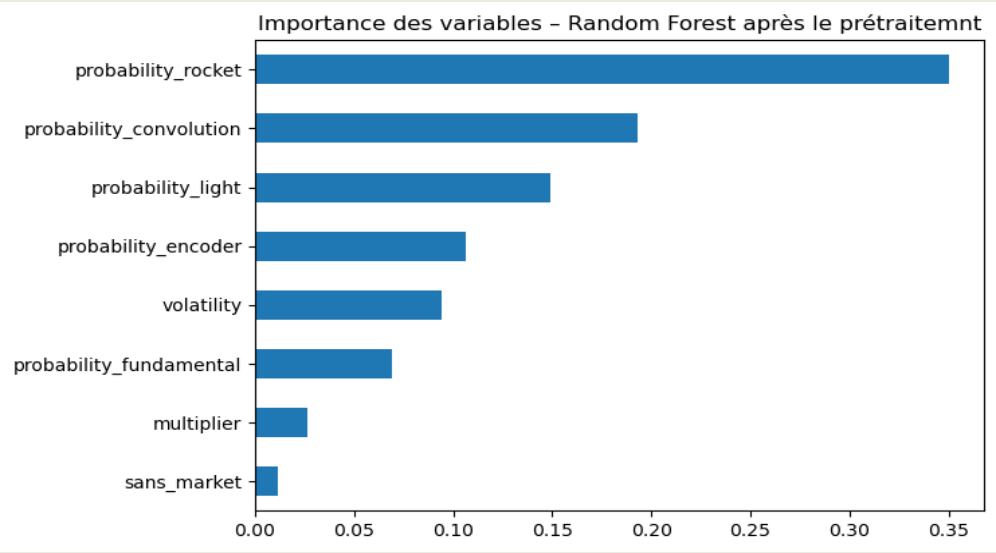
# Résultats

## RANDOM FOREST

Ici on observe une légère différence entre les variables avant le prétraitement et après.

## IMPORTANCE DES VARIABLES



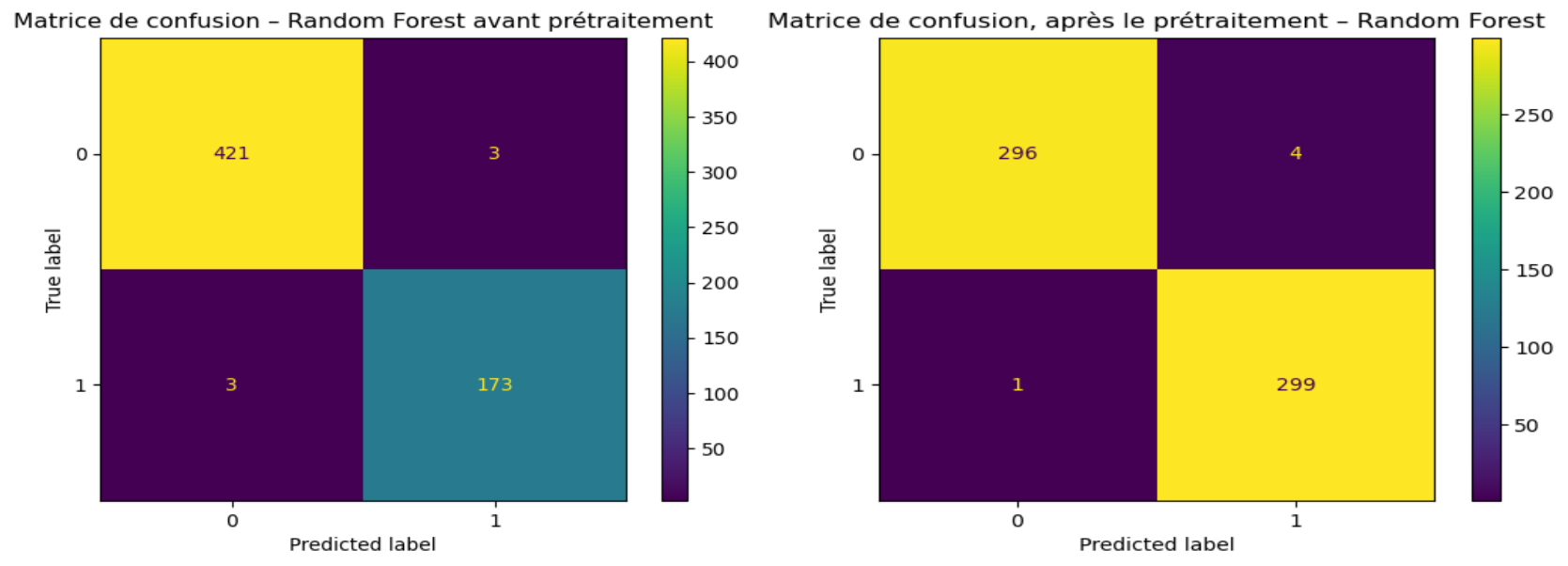



# Résultats

## RANDOM FOREST

Sur cette matrice, on voit que le Random Forest fait très peu d’erreurs. Le modèle reconnaît très bien les entreprises saines et défaillante avec peu d'erreur et après l'équilibre des classe on a encore moin d'erreurs

## MATRICE DE CONFUSION

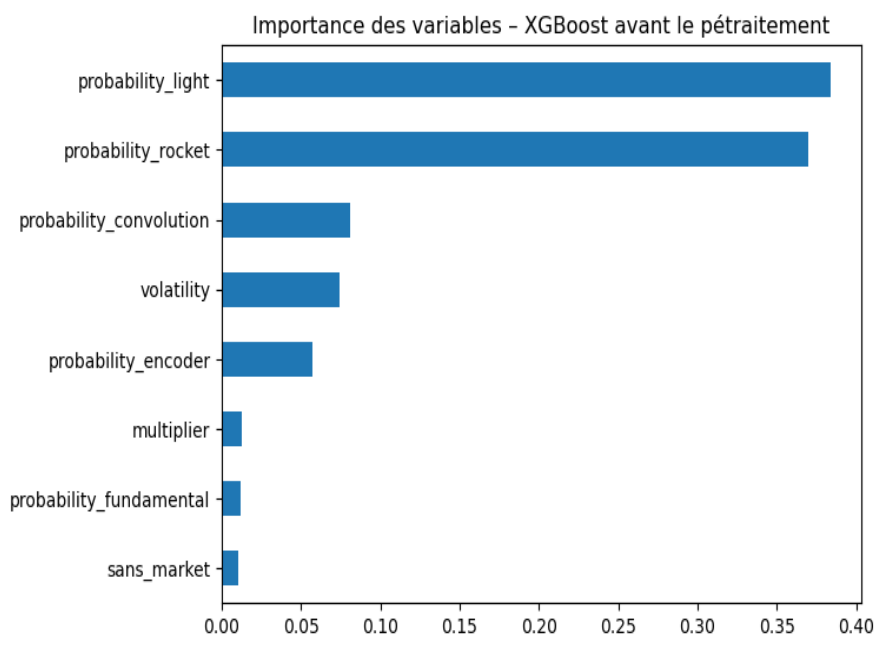


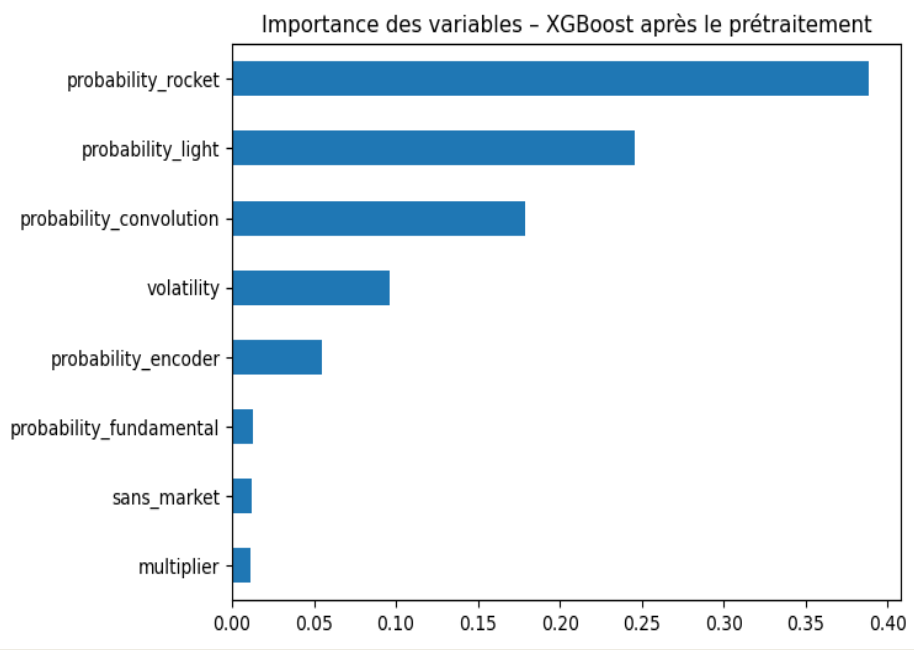
# Résultats

## XGBOOST

## IMPORTANCE DES VARIABLES

Ici on constate qu'avant le prétraitement et après le pétraitement le classement des variables par leur importance est pareille jusqu'à Probaility_encoder après le classement est différent.



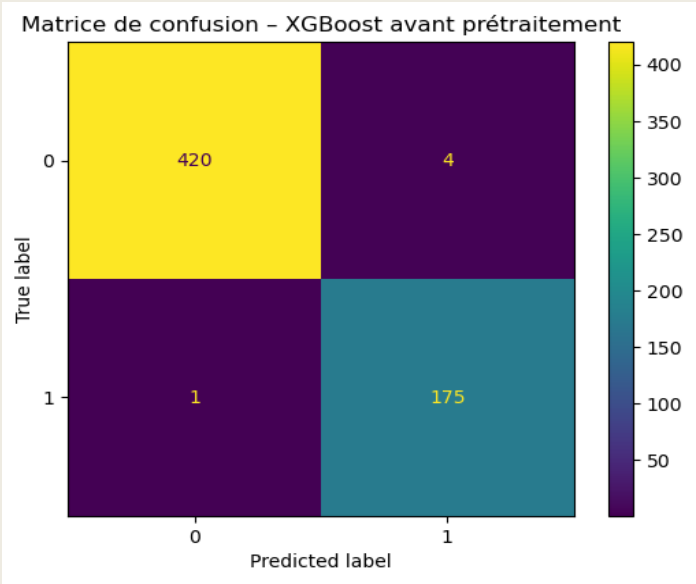



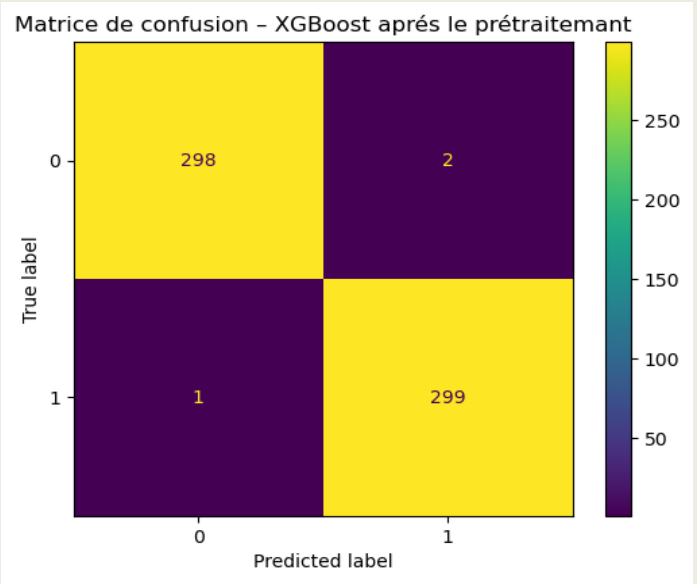
# Résultats

## XGBOOST

Le model XGBOOST prédire et faire très peu d'erreur. On observe un seule erreur au niveau des entreprise sainte et 4 erreurs au niveau des entreprises défaillantes et seulement 2 près le prétraitement.

## MATRICE DE CONFUSION

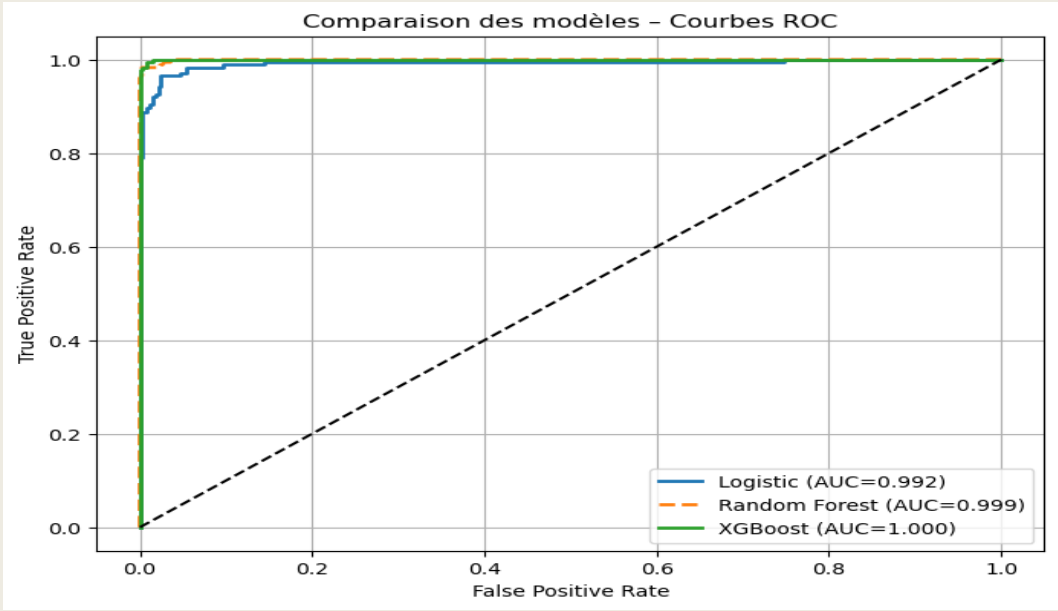


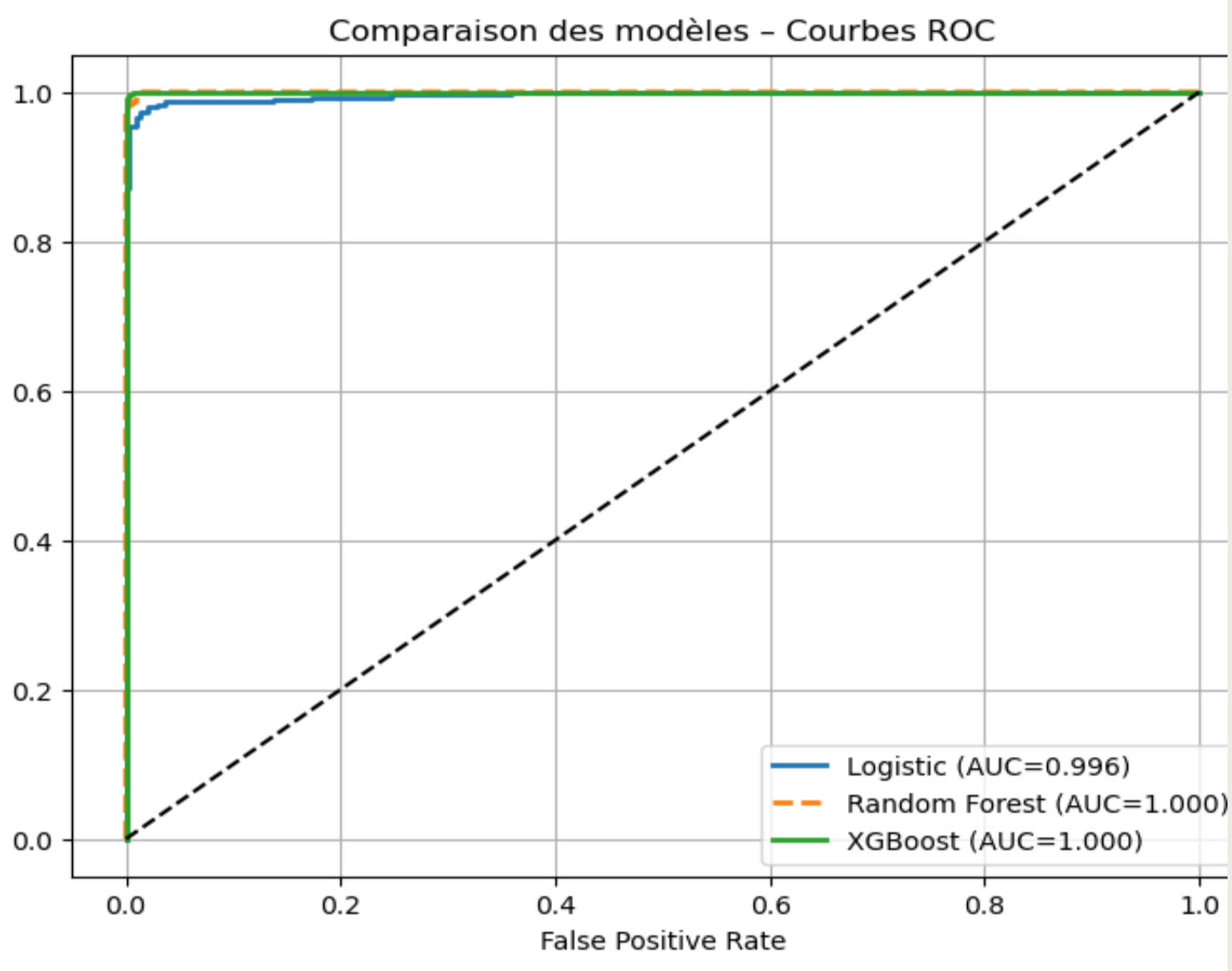



# Courbes Roc de comparaison des modèles avant le prétraitement

## LA COURBE ROC

- **La régression logistique** a une AUC d'environ **0,99**. Elle fonctionne déjà très bien, ce qui montre que les variables contiennent une information forte sur le risque de défaillance.
- **Le Random Forest** fait encore mieux, avec une AUC proche de **0,999**. Il capte mieux les relations complexes entre les variables.
- **XGBoost** atteint quasiment **1**, ce qui indique une capacité de discrimination presque parfaite entre les entreprises défaillantes et non défaillantes






# Courbe ROC de comparaison des modèles après le prétraitement

On constate ici que l' hypothèse 2 a amélioré l'AUC des 3 modèles. Et les meilleurs modèles sont le random Forest et le XGBoost.

# Comparaison des différents modèles avant le prétraitemnt

| Model | accuracy | precision | recall | F1-score | ROC-AUC |
|---|---|---|---|---|---|
| Logistique regression | 0.953333 | 0.993333 | 0.846591 | 0.914110 | 0.991879 |
| Random forest | 0.990000 | 0.982955 | 0.982955 | 0.982955 | 0.999464 |
| XGBoost | 0.991667 | 0.977654 | 0.994318 | 0.985915 | 0.999826 |

D'après ce tabeau de comparaison on constate que les meilleurs models ici sont: le Random forest et le Xboost parce qu'ont des valeurs supérieures à celui de la regression logistique au niveau de la plus part des Métrique. Et le Xboost est meilleur que le Random forest en observant l'accuracy, le recall , le F1-score et l' AUC.

# Tableau de comparaison des modèles après le prétraitement.

| Model | accuracy | precision | recall | F1-score | ROC-AUC |
|---|---|---|---|---|---|
| Logistique regression | 0.973333 | 0.993056 | 0.953333 | 0.972789 | 0.996089 |
| Random forest | 0.991667 | 0.982955 | 0.986799 | 0.996667 | 0.999817 |
| XGBoost | 0.995000 | 0.993355 | 0.996667 | 0.995008 | 0.999967 |

On remarque ici que la performance de chacun des modèles a augmenté et les meilleures modèles reste le Random forest et le XGBoost. Le model le plus performant reste le Random forest.

# Analyse et critique

Malgré des performances très élevées, notamment pour Random Forest et XGBoost (ROC-AUC proche de 1), plusieurs limites doivent être soulignées. D'abord, la variable cible `default` est construite à partir d'un seuil sur une probabilité agrégée, ce qui introduit une part d'arbitraire et peut influencer la répartition des classes. Ensuite, les scores quasi parfaits laissent envisager un risque de sur-apprentissage sur ce dataset particulier ; une validation sur d'autres périodes ou d'autres sources de données serait nécessaire pour confirmer la robustesse des modèles. Par ailleurs, les méthodes d'ensemble utilisées offrent une excellente performance mais restent moins explicables que la régression logistique, ce qui peut poser problème dans un contexte réglementé où la justification des décisions est importante. Enfin, le choix du seuil de décision a un impact direct sur l'équilibre entre détection des défauts et fausses alertes, et devrait être calibré en fonction des coûts métier associés aux erreurs.

# conclusion

Ce projet avait pour objectif de prédire le risque de défaillance d'entreprises à partir du dataset *sovai/bankruptcy* et de comparer plusieurs modèles de machine learning. Après un prétraitement complet (nettoyage, création de la variable `default`, équilibrage des classes, découpage train/validation/test, standardisation), trois modèles ont été évalués : régression logistique, Random Forest et XGBoost.

Les résultats montrent que la régression logistique fournit déjà des performances très élevées tout en restant facilement interprétable. Les modèles d'ensemble, en particulier Random Forest et XGBoost, atteignent des scores quasi parfaits (ROC-AUC proche de 1) et démontrent une excellente capacité de discrimination entre entreprises saines et défaillantes. Ces performances doivent toutefois être relativisées au regard d'un possible sur-apprentissage et de l'absence de validation sur d'autres jeux de données.

Sur le plan métier, le projet souligne l'importance du choix du seuil de décision, qui détermine le compromis entre détection des défaillances et fausses alertes. Avant un usage opérationnel, il serait nécessaire de calibrer ce seuil en fonction des coûts associés aux erreurs, et de renforcer la validation externe des modèles.

En résumé, le travail confirme l'apport des méthodes de machine learning pour la détection précoce du risque de défaut, tout en rappelant la nécessité d'un cadre de validation rigoureux et d'une attention particulière à l'explicabilité des modèles utilisés.