UQAC

TP3 - 8INF867 FONDEMENTAUX DE L'APPRENTISSAGE AUTOMATIQUE
Algorithmes de classification

November 16, 2023

Ali Ghammaz GHAA27070100
Atoumane Gaye GAYA19039305
Eric Moussinga

Contents

List of Figures

1 Introduction

L'objectif fondamental de ce projet réside dans le traitement de données provenant d'un problème de classification spécifique, à savoir la classification des scores de crédit. Dans le contexte financier, où l'évaluation précise du risque de crédit revêt une importance capitale pour les institutions financières et les prêteurs, notre choix s'inscrit dans la pertinence et l'omniprésence de cette tâche. La classification des scores de crédit permet d'assigner à chaque individu ou entreprise une catégorie de solvabilité, facilitant ainsi la prise de décisions éclairées en matière de prêt.

Notre approche méthodologique se concentre sur la mise en place d'un pipeline complet englobant les étapes clés, allant du prétraitement des données à l'entraînement et au test de modèles de classification, en passant par l'évaluation comparative de divers algorithmes tels que les arbres de décision, les k-plus proches voisins et random forest, parmi d'autres. Au cours de ce processus, notre ambition est de développer des compétences pratiques en apprentissage automatique, appliquées à un contexte financier concret. Le résultat attendu de ce projet se matérialise par la production d'un rapport analytique mettant en lumière nos choix, observations et conclusions, témoignant ainsi de notre engagement dans une démarche scientifique rigoureuse.

2 Description du jeu de données

Le jeu de données sélectionné provient de kaggle . Elle est structurée sous la forme d'un tableau csv contenant des informations sur un échantillon de plus de 100 personnes à travers le monde. Les données incluent les informations suivantes :Age, Genre, Revenu, Niveau d'éducation, Statut matrimonial, Nombre d'enfants, Propriété du logement, Score de crédit.

# Age	Gender	# Income	Education	Marital Sta...	# Number of...
25	Female	50000	Bachelor's Degree	Single	0
30	Male	100000	Master's Degree	Married	2
35	Female	75000	Doctorate	Married	1
40	Male	125000	High School Diploma	Single	0
45	Female	100000	Bachelor's Degree	Married	3
50	Male	150000	Master's Degree	Married	0

Figure 1: Aperçu de la dataframe

L'ensemble de donnée ne contient aucune valeur manquantes ,3 variables numériques et 5 variables catégoriques.

Variable	Nb Valeurs manquantes	Dtypes
Age	0	int64
Gender	0	object
Income	0	int64
Education	0	object
Martial Status	0	object
Number of Children	0	int64
Home Ownership	0	object
Credit Score	0	object

Figure 2: Type et nombre de valeurs manquantes pour chaque variable

Afin de bien décrire notre dataset nous allons procéder par une description univariable d'une part en cherchant à obtenir des informations sur la distribution, la tendance centrale et la dispersion des valeurs d'une variable spécifique. Et une description bivariable d'une autre part en déterminant l'existence d'une corrélation, une dépendance ou une tendance commune entre deux variables.

2.1 Analyse univariable

Age : La plage d'âge varie de 25 à 53 ans, avec une moyenne d'environ 38 ans. La majorité des individus dans l'échantillon ont entre 30 et 45 ans, comme indiqué par le 25e et le 75e percentile. L'écart type de 8.48 suggère une dispersion modérée autour de la moyenne.

Revenu (Income) : Les revenus varient de 25 000 à 162 500, avec une moyenne d'environ 83 765. La distribution des revenus semble plutôt équilibrée, avec un écart type de 32 457. Les 25e, 50e, et 75e percentiles fournissent une vue d'ensemble des valeurs centrales.

Nombre d'enfants (Number of Children) :

La plupart des individus n'ont pas d'enfants, comme indiqué par le 75e percentile à 0. La moyenne est d'environ 0.65, ce qui suggère qu'il y a une petite proportion d'individus ayant des enfants. La valeur maximale de 3 indique la présence de quelques individus avec un nombre plus élevé

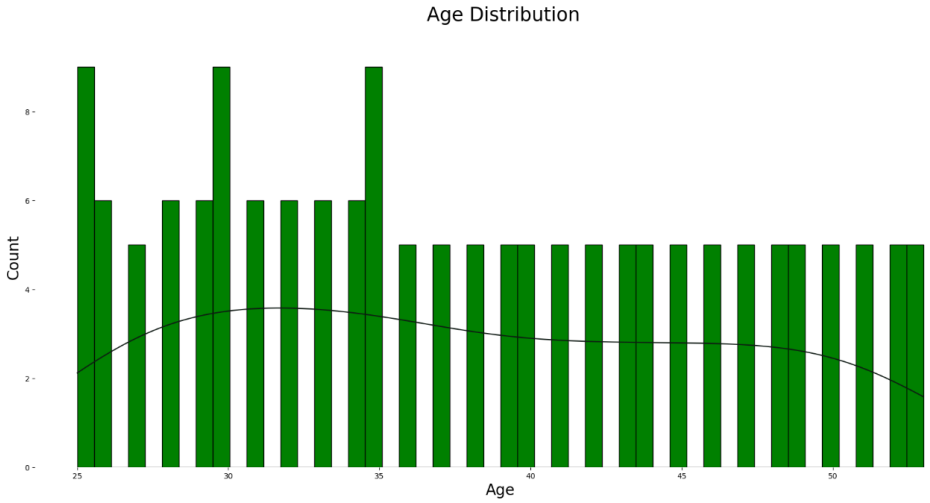


Figure 3: Disribution de la variable Age

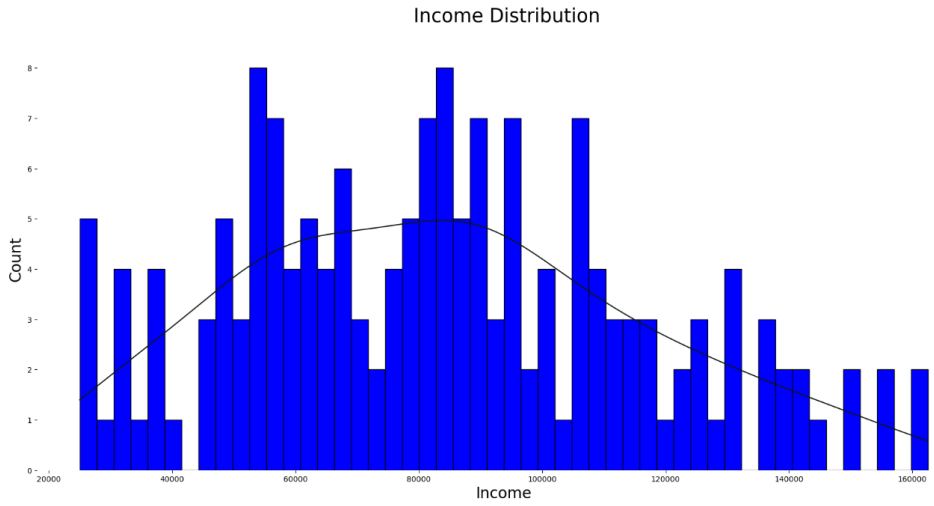


Figure 4: Disribution de la variable Income

d'enfants.

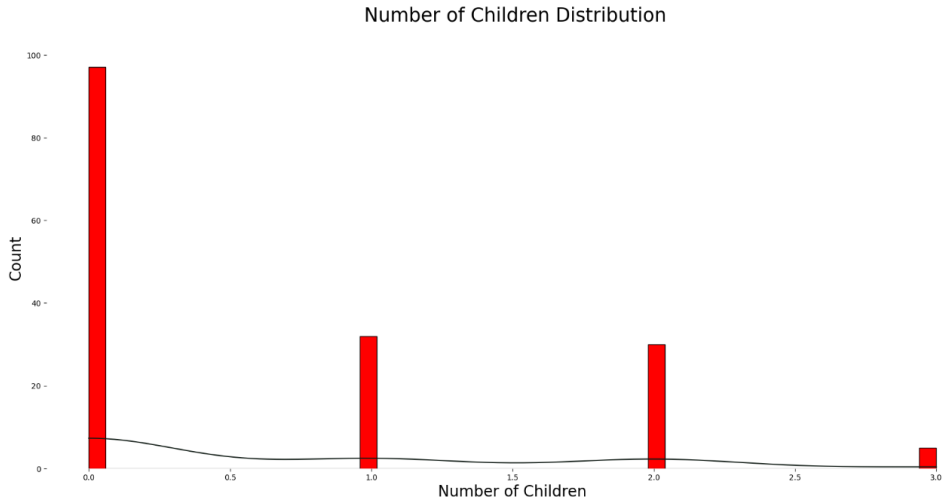
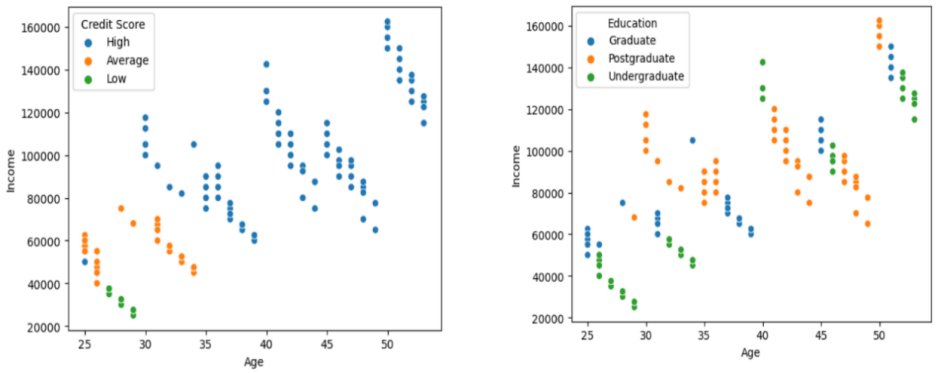


Figure 5: Disribution de la variable Income

2.2 Analyse bivariable



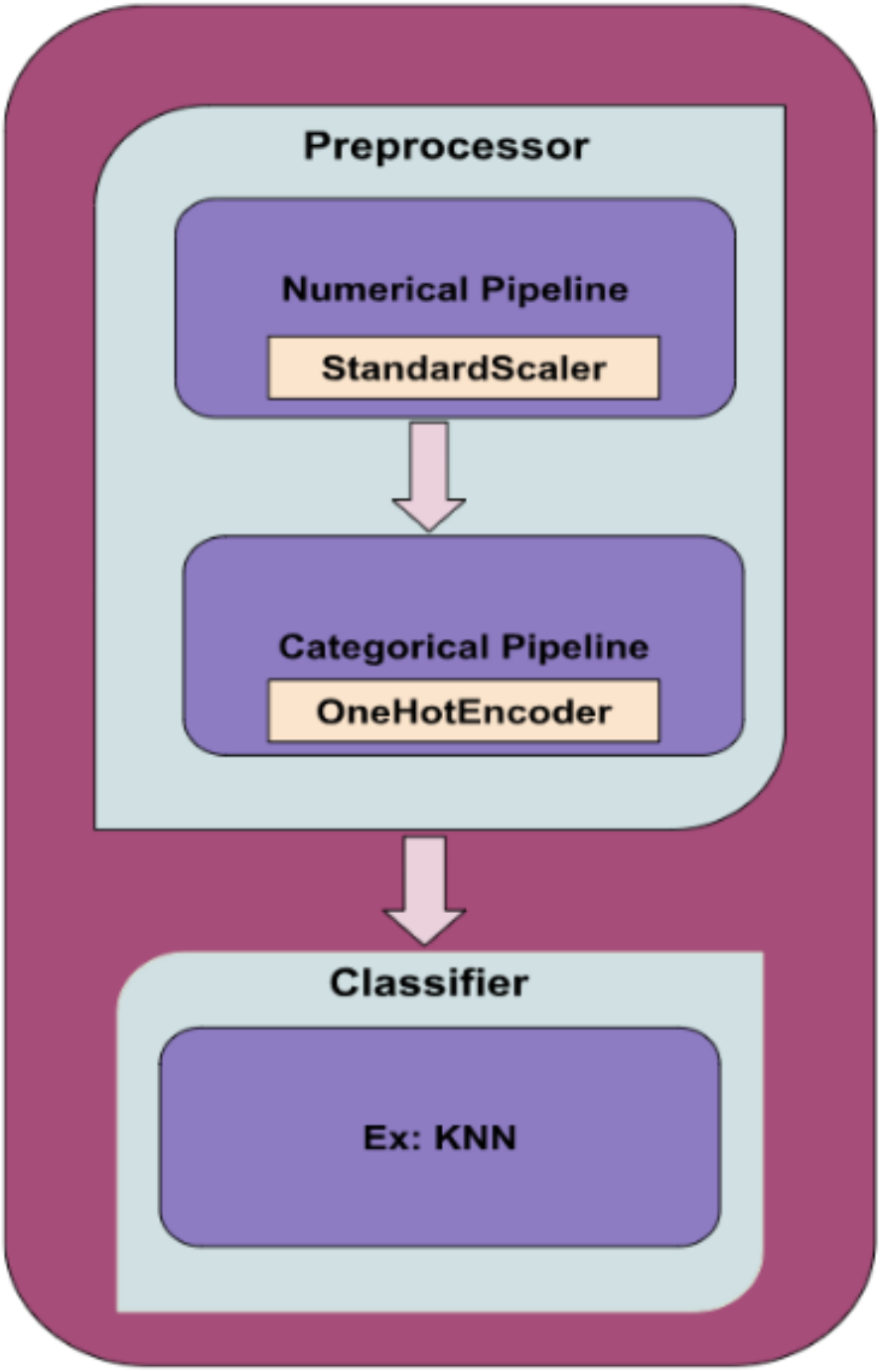
Il existe une corrélation positive entre l'âge et le revenu, les données illustrant que le revenu a tendance à augmenter avec l'âge, comme le montre le graphique

La première figure que, pour les utilisateurs plus jeunes, leur revenu a tendance à être inférieur à celui des autres et le score de crédit est bas. En revanche, pour les utilisateurs plus âgés, le revenu a tendance à être plus élevé, et le score de crédit est élevé, comme le suggère le graphique.

La deuxième figure montre une corrélation entre différents groupes d'âge, où le niveau d'éducation a une forte influence sur le revenu, indépendamment de l'âge.

3 Implémentation du Pipeline

Ce pipeline de traitement de données a pour objectif de préparer les caractéristiques de notre ensemble de données avant de les utiliser pour l'entraînement des modèles de classification.



Voici une description détaillée du pipeline :

3.1 Identification des Caractéristiques

Les caractéristiques sont séparées en deux catégories : numériques et catégoriques. Les caractéristiques numériques sont celles de type entier (int64), tandis que les caractéristiques catégoriques sont celles de type objet. Transformateurs pour les Caractéristiques Numériques et Catégoriques :

Pour les caractéristiques numériques, un transformateur est créé avec une étape de mise à l'échelle (scaler) en utilisant le StandardScaler. Cela permet de standardiser les valeurs numériques. Pour les caractéristiques catégoriques, un transformateur est créé avec une étape de codage à chaud (onehot) en utilisant OneHotEncoder. Cela permet de convertir les caractéristiques catégoriques en une représentation numérique. Combinaison des Transformateurs avec ColumnTransformer :

Un transformateur de colonne (ColumnTransformer) est créé pour combiner les transformateurs numériques et catégoriques. Il applique les transformations spécifiques à chaque type de caractéristique. Vu que notre ensemble de données ne contient pas des valeurs manquantes , l'ajout d'un imputer dans le pipeline n'est pas nécessaire.

3.2 Création du Pipeline Global

Un pipeline global est créé en associant le transformateur de colonne (preprocessor) et un classificateur donnée Le pipeline garantit que les données sont prétraitées de manière cohérente avant d'être alimentées aux modèles de classification.

4 Entrainement et Evaluation des modèles

Nous avons évalué la performance de notre modèle de classification en utilisant une technique de validation croisée à 5 plis avec la fonction cross validate de scikit-learn. Cette approche nous permet d'estimer la capacité de notre modèle à généraliser sur des données non vues. Nous avons choisi la métrique de précision (accuracy) pour mesurer la justesse de nos prédictions. Les données d'entraînement ont été divisées en 5 ensembles, et le modèle a été entraîné et évalué cinq fois, chaque fois sur un ensemble de test différent. Cette méthodologie offre une évaluation robuste de la performance de notre modèle sur divers sous-ensembles de données.

Ensuite, nous avons comparé les performances de plusieurs algorithmes de classification sur notre jeu de données de scores de crédit.

4.1 Régression Logistique

La régression logistique a démontré une précision moyenne de 96.97 avec une classification très précise des catégories de crédit.

4.2 SVM (Support Vector Machine)

Le SVM a montré une précision moyenne de 90.91, indiquant une performance solide bien que légèrement inférieure à la régression logistique.

4.3 K-Nearest Neighbors (KNN)

Le modèle KNN a fourni une précision moyenne de 90.91, montrant des résultats similaires au SVM.

4.4 Random Forest

Le Random Forest a présenté une précision moyenne de 90.91, aligné avec les résultats du SVM et du KNN.

4.5 Decision Tree

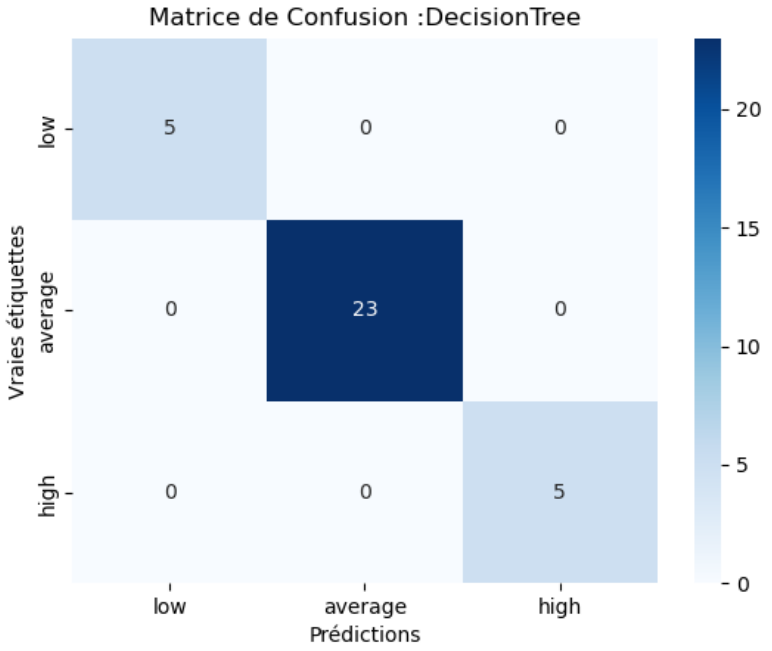
Decision Tree

L'arbre de décision a démontré une performance exceptionnelle avec une précision de 100, suggérant une adaptation parfaite aux données d'entraînement.

4.6 Gradient Boosting

Le modèle Gradient Boosting a également atteint une précision de 100, montrant une adaptation et une capacité de généralisation exceptionnelles.

Dans l'ensemble, ces résultats mettent en évidence l'efficacité de divers algorithmes de classification pour résoudre le problème de classification des scores de crédit. La régression logistique, le SVM, le KNN, le Random Forest, l'arbre de décision et le Gradient Boosting ont tous montré des performances robustes, bien que l'arbre de décision et le Gradient Boosting se soient démarqués avec une précision parfaite. Ces résultats soulignent l'importance de choisir le modèle approprié en fonction des caractéristiques spécifiques du problème et des données disponibles.



Decision Tree :

La classe 0 a 5 vrais positifs, 0 faux négatif, et 0 faux positifs. La classe 1 a 23 vrais positifs, 0 faux négatif, et 0 faux positifs. La classe 2 a 5 vrais positifs, 0 faux négatif, et 0 faux positifs.

SVM :

La classe 0 a 4 vrais positifs, 1 faux négatif, et 0 faux positifs. La classe 1 a 22 vrais positifs, 1 faux négatif, et 0 faux positifs. La classe 2 a 4 vrais positifs, 1 faux négatif, et 0 faux positifs. (Voir Annexe)

	fit time mean	test score mean	F1 score	Precision	Recall	Accuracy
Logistic Regression	0.0126	0.969	0.9696	0.9696	0.9696	0.9696
SVM	0.00846	0.954	0.9090	0.9090	0.9090	0.9090
KNN 3	0.0171	0.977	0.9090	0.9090	0.9090	0.9090
Random F	0.1144	0.977	0.9090	0.9090	0.9090	0.9090
Gradient B	0.2194	0.969	1.0	1.0	1.0	1.0
Decision T	0.0115	0.969	1.0	1.0	1.0	1.0

5 Conclusion

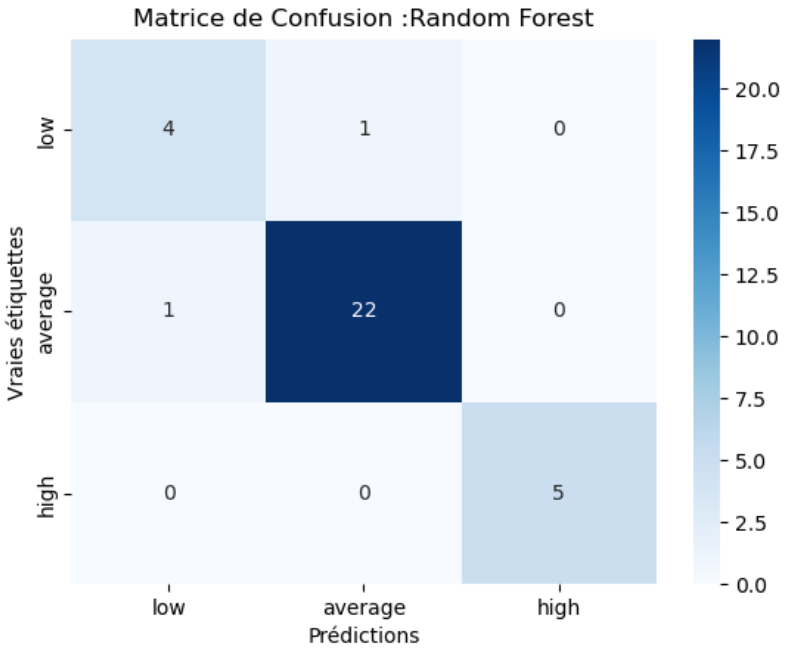
En conclusion, ce projet de classification des scores de crédit a été une expérience enrichissante qui a permis d'explorer en profondeur le processus complet d'analyse de données, du choix de l'ensemble de données à la comparaison des performances des algorithmes de classification.

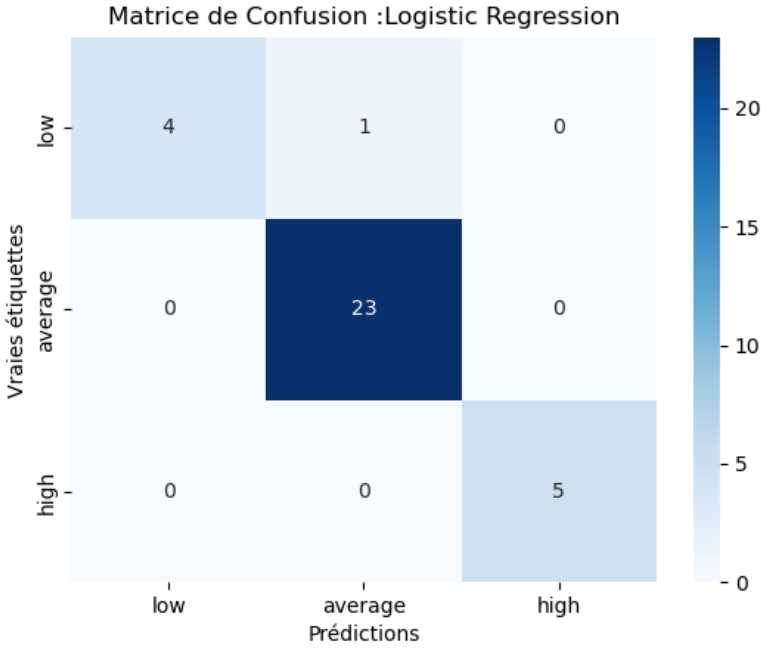
Le prétraitement des données a constitué une phase cruciale du projet, impliquant le traitement des variables catégoriques. Les différentes étapes de prétraitement ont été soigneusement décrites, justifiées, et illustrées par des graphiques de distribution, offrant ainsi une compréhension approfondie de la nature des données manipulées.

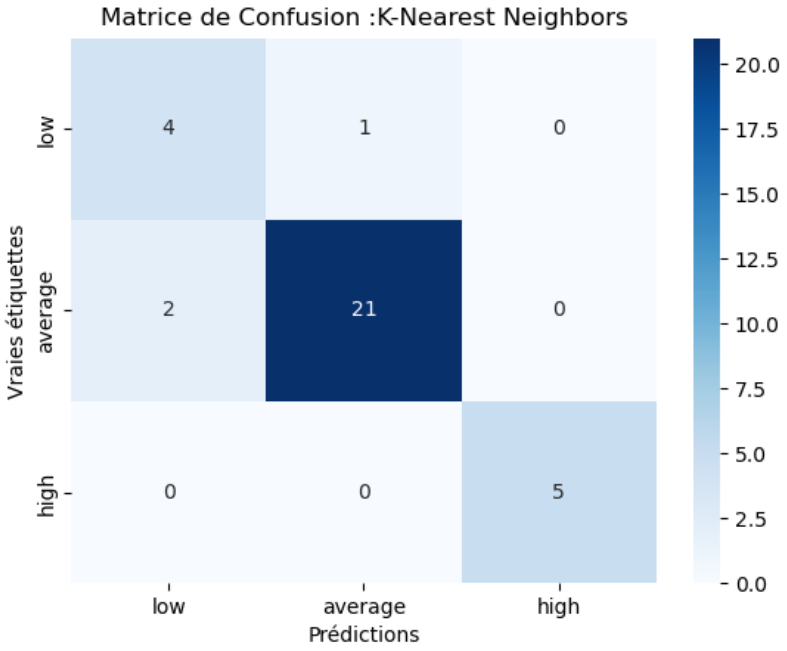
La mise en place du pipeline de classification a été réalisée de manière méthodique, intégrant des transformateurs spécifiques pour les caractéristiques numériques et catégoriques. Ce pipeline a permis d'assurer une préparation cohérente des données avant l'entraînement des modèles de classification.

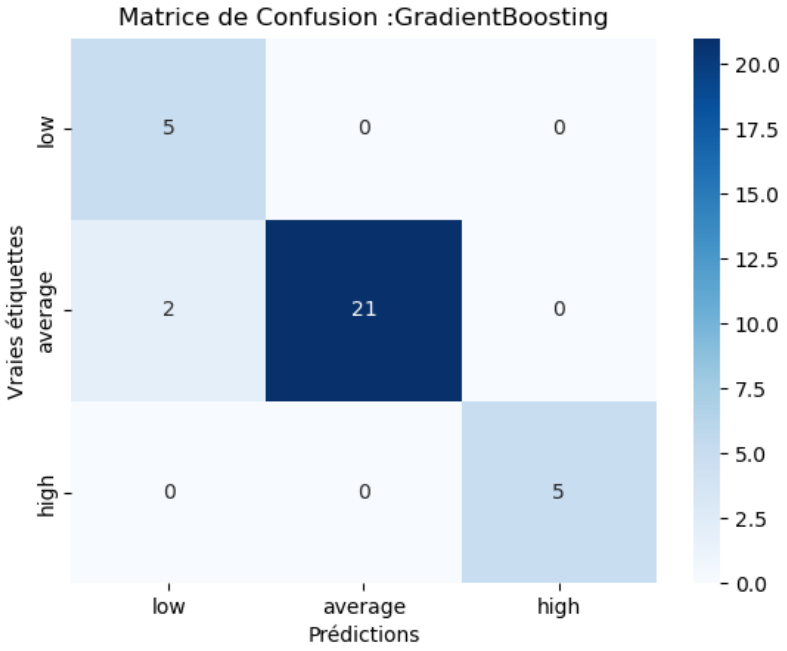
La comparaison des différents algorithmes, tels que l'arbre de décision, les k-plus proches voisins, et random forest, a démontré des performances robustes de chaque modèle. bien que l'arbre de décision et le Gradient Boosting se soient démarqués avec une précision parfaite. Cette phase a souligné l'importance de choisir un algorithme adapté aux spécificités du problème, renforçant ainsi la pertinence des choix méthodologiques effectués.

6 Annexe


Matrice de Confusion :Random Forest
Vraies étiquettes
low
average
high
4
1
0
1
22
0
0
0
5
low
average
high
Prédictions
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0


Matrice de Confusion :Logistic Regression
Vraies étiquettes
low
average
high
4
1
0
0
23
0
0
0
5
low
average
high
Prédictions
20
15
10
5
0


Matrice de Confusion :K-Nearest Neighbors
Vraies étiquettes
low
average
high
4
1
0
2
21
0
0
0
5
low
average
high
Prédictions
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0


Matrice de Confusion :GradientBoosting
Vraies étiquettes
low
average
high
5
0
0
2
21
0
0
0
5
low
average
high
Prédictions
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0