

MINI - PROJET ENSEMBLE LEARNING

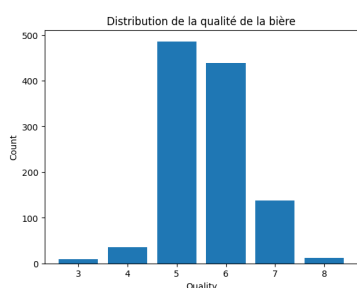
Introduction

Ce Mini-projet explore l'application d'algorithmes d'apprentissage ensembliste pour prédire la qualité de la bière, utilisant des techniques telles que les arbres de décision, AdaBoost, les réseaux de neurones, le bagging et la forêt aléatoire. Notre but est de déterminer la méthode la plus performante pour évaluer la qualité de la bière, en se focalisant sur l'efficacité, la rapidité et la fiabilité des prédictions.

A. Base des données

La base de données « beer quality » comporte 1600 exemples décrits par 11 caractéristiques quantitatives. L'objectif est d'évaluer, sachant ces variables, la qualité d'une bière, évaluée de 1 (sans commentaire !) à 10 (particulièrement excellente).

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol
count	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000	1119.000000
mean	8.309562	0.531132	0.270250	2.548302	0.087711	15.920465	46.966488	0.996778	3.314272	0.658820	10.417337
std	1.713899	0.182022	0.195492	1.427730	0.047143	10.273166	33.036693	0.001840	0.153980	0.172242	1.059751
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.740000	0.370000	8.400000
25%	7.100000	0.400000	0.090000	1.900000	0.071000	7.000000	22.000000	0.995685	3.220000	0.550000	9.500000
50%	7.900000	0.520000	0.260000	2.200000	0.080000	14.000000	38.000000	0.996800	3.310000	0.620000	10.200000
75%	9.200000	0.640000	0.430000	2.600000	0.090000	21.000000	64.000000	0.997845	3.400000	0.730000	11.000000
max	15.900000	1.580000	1.000000	15.500000	0.611000	68.000000	289.000000	1.003690	4.010000	2.000000	14.900000



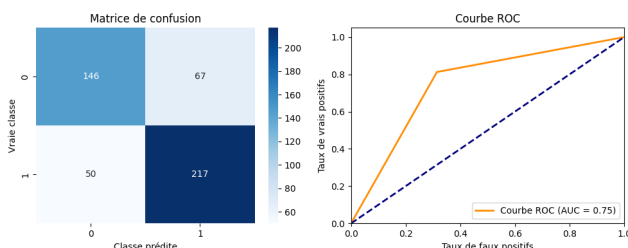
Commentaire: La variable « quality » et la majorité des autres variables présentent une distribution relativement centrée, comme on peut le voir par les moyennes proches des médianes (50e percentile).

B. Classification binaire

Dans cette section nous avons créé une nouvelle variable quantitative ybin à deux modalités (0 et 1) en fonction de la médiane de la variable y.

1. Optimisation d'un arbre de décision (random search)

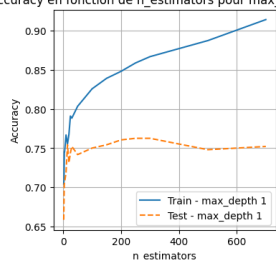
le meilleur modèle par faisant plusieurs recherches aléatoires est :



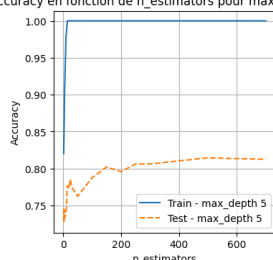
- best model: {'criterion': 'gini', 'max_depth': 20, 'max_features': 11, 'min_samples_split': 2}
- Accuracy : 0.75625
- Temps d'apprentissage : 0.006 s
- Temps d'inférence : 0.001 s

2. Utilisation d'AdaBoost()

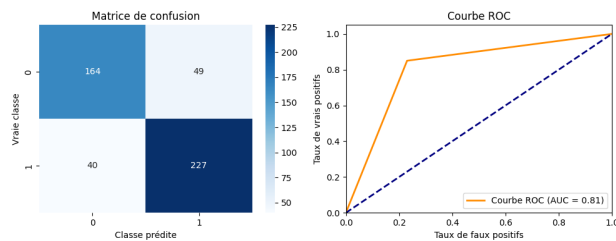
Accuracy en fonction de n_estimators pour max_depth = 1



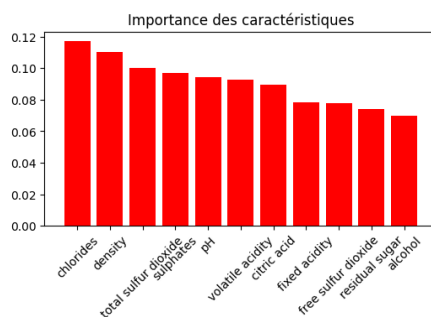
Accuracy en fonction de n_estimators pour max_depth = 5



Commentaire: l'algorithme adaptatif boosting bien qu'il soit plus lent en apprentissage et en inférence, il améliore les performances en agissant sur le biais. Le meilleur modèle est l'ensemble 500 d'arbres de décision de `max_depth = 5`.



- best model: {'estimator': DecisionTreeClassifier(max_depth=5), 'n_estimators': 500, 'random_state': 42}
- Accuracy : 0.814583
- Temps d'apprentissage : 3.898 s
- Temps d'inférence : 0.116 s



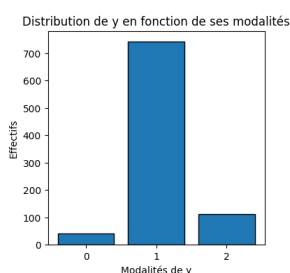
oui, il est possible de mesurer l'importance d'une caractéristique dans la décision. AdaBoost ajuste les poids des observations à chaque itération, donnant plus d'importance aux cas difficiles. L'importance des caractéristiques est calculée en fonction de la contribution de chaque caractéristique à améliorer la précision des arbres de décision faibles au sein de l'ensemble.

Conclure sur le biais et la variance de l'algorithme

- **Biais :** AdaBoost réduit le biais en se concentrant de manière itérative sur les exemples les plus difficiles à classer. Cependant, dans le cas des modèles faibles eux-mêmes très biaisés (arbre avec `max_depth` très petit), AdaBoost n'a pas réussi à produire un modèle final avec un biais plus faible que celui obtenu avec un arbre décision optimal obtenu par recherche aléatoire.
- **Variance :** l'impact de AdaBoost sur la variance du modèle final est peu présent, car si les estimateurs sont trop complexes (profonds), le modèle AdaBoost surajuste les données d'entraînement, conduisant à une variance élevée.

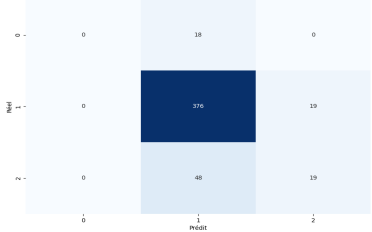
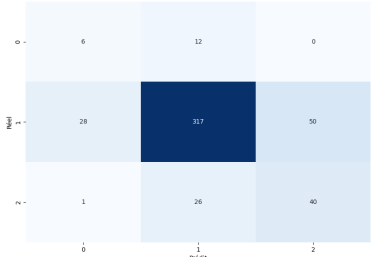
C. Classification multiclasse

Dans cette section nous avons créé une nouvelle variable quantitative `ymulti` à 03 modalités qualité basse (0), moyenne (1) ou élevée (2).



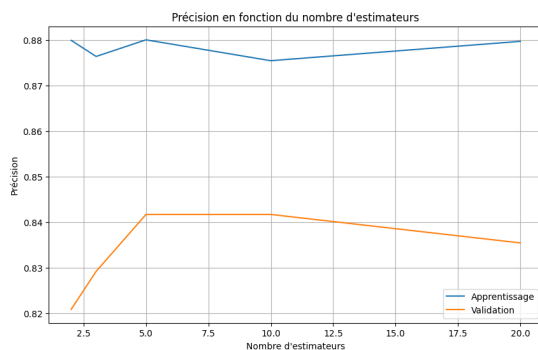
Commentaire: Avec une catégorisation de note: basse([0;5]), moyenne([5;7]) et élevée([7;10]), il est clair que les classes de qualité basse et élevée sont sous-représentées par rapport à la classe moyenne. Cette disproportion pourrait conduire à un biais dans le modèle de classification en faveur de la classe Moyenne plus représentée. Pour remédier à cela, nous pouvons envisager d'équilibrer les classes en utilisant l'augmentation des données pour les classes minoritaires (par exemple, en utilisant SMOTE)

Partie 1:

Modèle sans équilibrage des données:	<p>Matrice de Confusion pour modèle avec 16 neurones</p> 	<ul style="list-style-type: none"> → nombre de neurones: 16 → Test Loss : 0.4099 → Test Accuracy : 0.8229 → Temps d'apprentissage : 1.124 s → Temps d'inférence : 0.048 s
Modèle avec équilibrage des données:	<p>Matrice de Confusion pour modèle avec 64 neurones</p> 	<ul style="list-style-type: none"> → Nombre de neurones: 64 → Test Loss : 0.6055 → Test Accuracy : 0.7563 → Temps d'apprentissage : 5.44 s → Temps d'inférence : 0.055 s

Conclusion Pour le meilleur modèle est sans équilibrage, comme on l'avait prédit le modèle à un fort biais pour les classes minoritaires mais grâce à une surreprésentation de la classe 1 (qualité moyenne), l'accuracy du modèle est élevée. l'augmentation de données à permis de réduire le biais mais à côté a augmenté le temps d'apprentissage et réduit la performance de globale du modèle. Cette méthode d'équilibrage ne semble pas adaptée à la situation.

Application de la technique de bagging avec le réseau de neurones

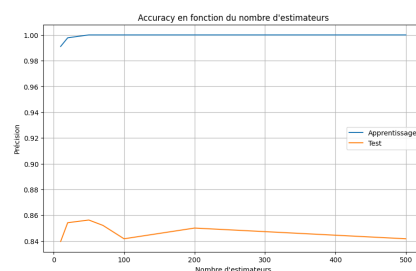


Commentaire: Le modèle avec 5 estimateurs présente un bon compromis avec une accuracy en apprentissage et en validation et un temps d'apprentissage raisonnable.

- Accuracy apprentissage: 0.880
- Accuracy validation: 0.8416
- Temps d'apprentissage : 14.096s
- Temps d'inférence : 0.264 s

Conclusion: L'analyse des résultats du bagging indique un équilibre entre le biais et la variance. En augmentant le nombre d'estimateurs, l'accuracy en apprentissage reste relativement stable, tandis que l'accuracy en validation atteint un plateau. Cela suggère que l'ajout d'estimateurs supplémentaires ne conduit pas à un sur apprentissage significatif, mais n'améliore pas non plus de manière significative la capacité du modèle à généraliser.

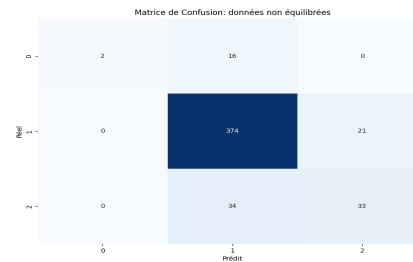
Partie 2:



Optimisation par random search:

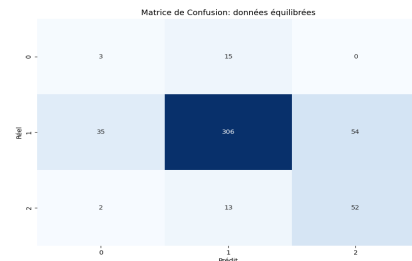
Données non équilibrées

- Best model: {'n_estimators': 70, 'max_depth': 45}
- Train accuracy: 1.0
- Test Accuracy: 0.8520
- Temps d'apprentissage: 0.0975 s
- Temps d'inférence: 0.0037s

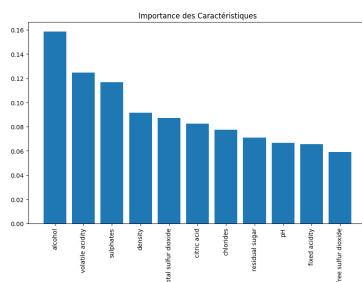


Données équilibrées

- Best model: {'n_estimators': 200, 'max_depth': None}
- Train Accuracy: 1.0
- Test Accuracy: 0.7520
- Temps d'apprentissage: 0.7221s
- Temps d'inférence: 0.0119s



Commentaire: Le meilleur modèle est celui obtenu par recherche aléatoire sur les données non équilibrées, avec une précision d'apprentissage aussi de 1.0 mais une précision de test nettement meilleure (0.8521), et des temps d'apprentissage et d'inférence significativement réduits.



- Non, l'importance données aux caractéristiques n'est pas la même qu'en B.3

biais et variance

Les forêts aléatoires, en combinant les prédictions de nombreux arbres de décision, visent à réduire à la fois le biais et la variance, améliorant ainsi la généralisation sur de nouvelles données

Conclusion générale

Technique	Accuracy	Temps d'apprentissage	Temps d'inférence (secondes)
AdaBoost avec arbre de décision	0.814583	3.898	0.116
bagging avec le réseau de neurones	0.8416	14.096	0.264
random search avec forêt aléatoire	0.8520	0.0975	0.0037

Le meilleur modèle est celui obtenu par recherche aléatoire, avec une précision de test nettement meilleure, et des temps d'apprentissage et d'inférence significativement meilleurs que ces concurrents.