Rapport de projet Supervised Machine Learning

exercice 1:

exercice 2:

step 1

On considère les variables discrètes aléatoires (X,Y) suivantes:

- *x* = { 1 , … , 100}
- *y* = { 1, 2, 3, 4, 5}

Avec *x* représentant le nombre de personnes travaillant avec l'artiste (de 1 à 100, 100 étant un nombre non raisonnable donc pas besoin de mettre plus grand).
y représentant la catégorie de stream de la musique.
On peut estimer que plus il y aura de gens impliqués dans le projet, plus la musique aura de chance de plaire au plus grand nombre avec une limite. On peut donc estimer que *X* va suivre la fonction logarithmique +0.01.

On estime que *Y* dépend du talent du groupe qui va donc suivre la loi de Bernoulli tel que
Y = { B(⅘) si X > 0 et X < 0.2,
B(⅗) si X >= 0.2 et X < 0.4
B(⅖) si X >= 0.4 et X < 0.6
B(⅗) si X >= 0.6 et X < 0.8
B(⅘) si X >= 0.8 et X < 1

On sait que la condition attendu pour le prédicteur de Bayes pour la square loss est :

$$f^*(x) = E[Y|X = x]$$

On va essayer d'appliquer avec les variables prédites plus haut:
f*(0) = 0
f*(1) = 0.01
f*(2) = 0.31
f*(3) = 0.48
f*(4) = 0.61
f*(7) = 0.85

Nous ne savons pas comment calculer la condition attendu pour le prédicteur de Bayes pour l'absolute loss donc nous ne pouvons pas comparer.

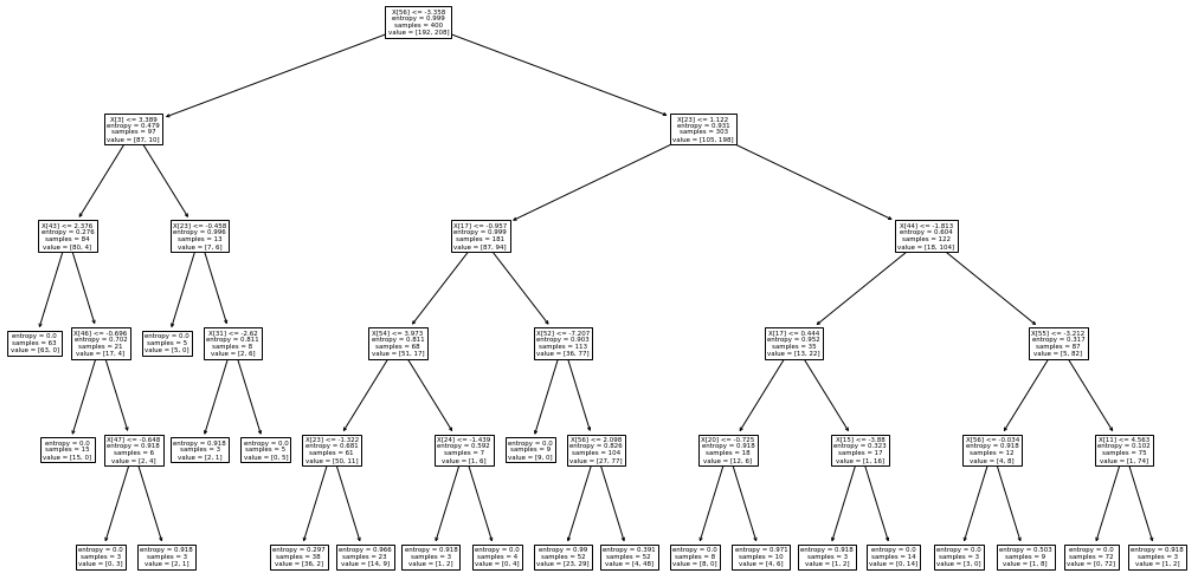
exercice 3:

Dans cet exercice, plusieurs méthodes de classification ont été utilisées pour prédire le gagnant d'un match de basketball.

Arbre de décision

L'arbre de décision est un algorithme de machine learning utilisé pour faire de la classification.
Sur ce dataset, la meilleure précision est obtenue par l'arbre de décision est de 0.77 et le meilleure score de validation croisé est de 0.81, avec une profondeur de 5 couche.

schéma de l'arbre de décision obtenue



Support vector machine

Le support vector machine est un algorithme de machine learning utilisé pour les tâches de classification et de régression.
Sur ce dataset, la meilleure précision obtenue par le support vector machine est de 0.82 et le meilleur score de validation croisée est de 0.88.

Régression logistique

La régression logistique est un algorithme de machine learning utilisé pour les tâches de classification.
Sur ce dataset, la meilleure précision obtenue par la régression logistique est de 0.80 et le meilleur score de validation croisée est de 0.93.

exercice 4:

Dans cet exercice, plusieurs algorithmes de machine learning ont été utilisés: Régression linéaire, random forest, régression lasso, support vector regression.

Parmi toutes ces méthodes, la seule à obtenir un résultat convenable est la Régression linéaire avec un score R2 de 0.72. Les autres méthodes obtiennent des scores R2 inférieur à 0.50.

exercice 5:

dataset: Heart Failure Detection

Présentation

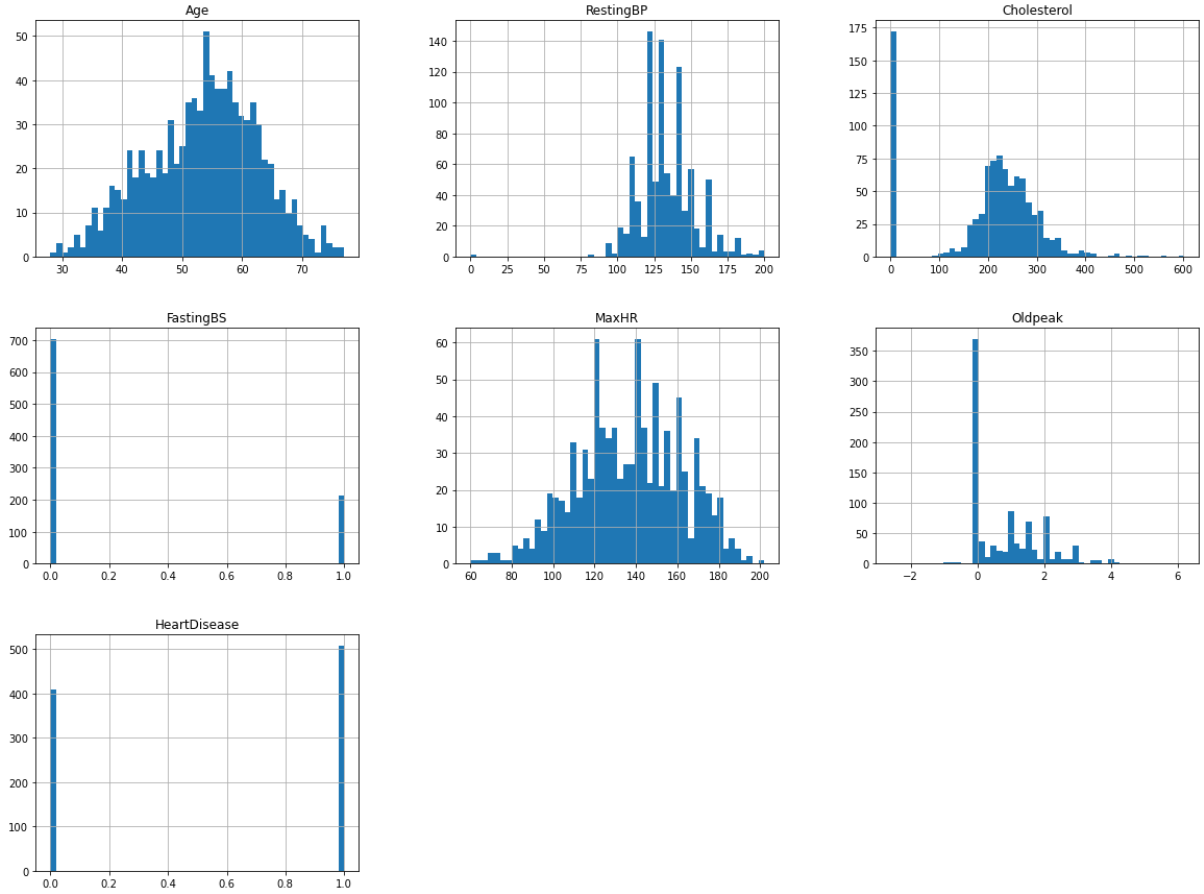
Nous avons trouvé sur Kaggle.com un dataset très complet et super intéressant sur les problèmes cardiaques. Le dataset est bien noté, à l'air intéressant, complet et facile à utiliser. Il comporte 12 colonnes dont 3 catégoriques, 4 binaires et 5 quantitatives avec des informations sur l'age, le sexe, le type de douleur du patient, si il est atteint d'une maladie cardiaque et biens d'autres données médicales comme le taux de cholesterol.
Le dataset comporte 918 observations complètes, ce qui nous permet de travailler sur un échantillon intéressant de données.
La question qu'on s'est posée est qu'on voulait savoir s' il était possible de prédire si un patient est atteint d'une maladie cardiaque ou pas en fonction de ses données médicales qui sont à notre disposition dans le dataset.

Attribute Information

1. Age: age of the patient [years]
2. Sex: sex of the patient [M: Male, F: Female]
3. ChestPainType: chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic]
4. RestingBP: resting blood pressure [mm Hg]
5. Cholesterol: serum cholesterol [mm/dl]
6. FastingBS: fasting blood sugar [1: if FastingBS > 120 mg/dl, 0: otherwise]
7. RestingECG: resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes' criteria]
8. MaxHR: maximum heart rate achieved [Numeric value between 60 and 202]
9. ExerciseAngina: exercise-induced angina [Y: Yes, N: No]
10. Oldpeak: oldpeak = ST [Numeric value measured in depression]
11. ST_Slope: the slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping]
12. HeartDisease: output class [1: heart disease, 0: Normal]

Sur l'image précédente, vous avez la liste des informations disponible dans le dataset, ainsi que leurs définition et le type de variables inclus.

Data exploration



En créant un histogramme pour certaines colonnes du dataset, on peut observer que le dataset contient plusieurs types de données: catégoriel et numérique. Ces histogrammes nous permettent aussi de voir la distribution des valeurs dans ces colonnes.

Classification bayésienne

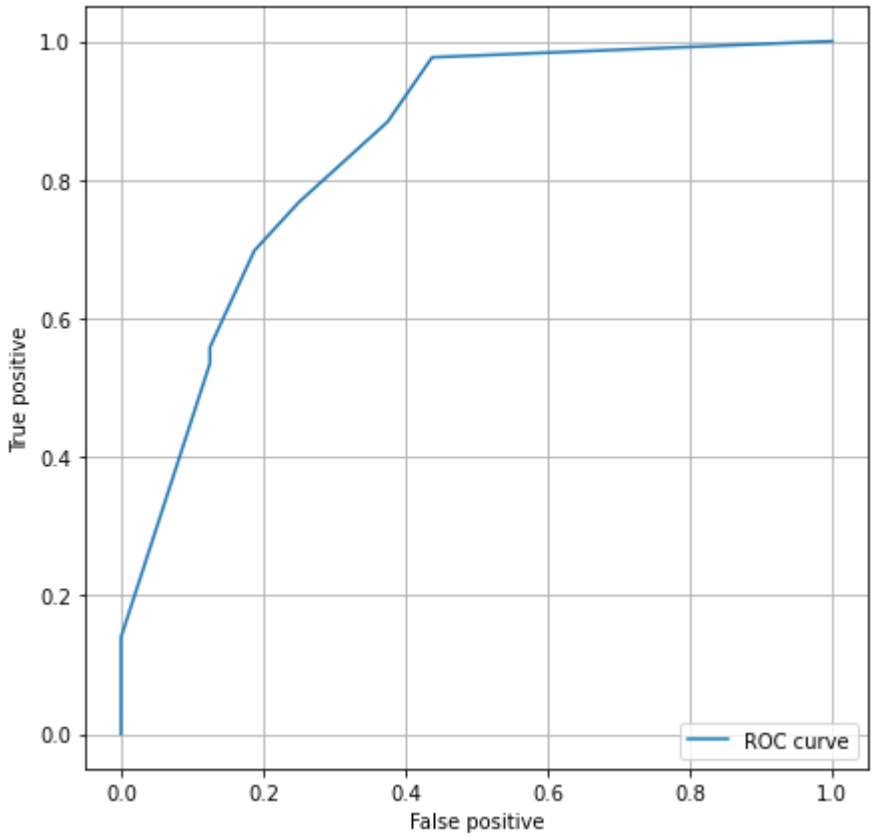
Dans une première partie nous avons fait une classification bayésienne afin de déterminer certaines métriques qui nous permettront de savoir quelles données sont liées aux maladies cardiaques ou non.
On voit clairement que c'est la combinaison âge / fréquence cardiaque (oldpeak = ST) est celle qui nous donne la plus grande précision (70.8%). C'est donc cette combinaison que nous allons utiliser ensuite.

Arbres de décision

Nous avons ensuite essayé les arbres de décisions, nous avons eu des résultats de 76% de précision ce qui est le meilleur résultat qu'on à depuis le début. On se retrouve avec cette matrice de confusion:

	observé négatif	observé positif
prédit négatif	39.56 %	10.98 %
prédit positif	13.18 %	36.26 %

Nous avons ensuite calculé la courbe ROC pour voir s' il y avait une différence entre les jeux de données.



L'impression des métriques et le tracé de la courbe ROC montrent une faible différence entre le jeu de données classique et le jeu de données obtenu avec 10 plis. En effet, les différentes métriques présentées ci-dessus tendent à montrer que les deux jeux de données offrent de bonnes performances dans la détection des maladies cardiaques. Ces bonnes performances permettent de conclure que les arbres de décision se généralisent bien aux nouvelles données.

Nous avons ensuite essayé plusieurs modifications sur l'arbre comme sa profondeur ou changer la taille du dataset mais les résultats sont moins bons.

Régression Linéaire

Nous avons essayé de faire une régression linéaire pour voir si l'on pouvait séparer les données mais les résultats n'étaient pas très bons, en effet on obtient un R2 de 0.19 et une variance de 0.21. On aurait pu s'y attendre avec la visualisation des observations donc les résultats sont logiques.

Réseaux de neurones

Pour aller plus loin, nous avons essayé les réseaux de neurones.
Un nombre plus élevé de couches permet la reconnaissance de motifs de plus haut niveau et une meilleure généralisation. Pour ce jeu de données, la meilleure performance est obtenue avec un réseau neuronal à 3 couches cachées de 20 neurones chacune.
Le modèle obtient rapidement son poids optimal.
En ce qui concerne le taux d'apprentissage, le modèle obtient ses meilleures performances plus rapidement avec une valeur comprise entre 0.01 et 0.001, une valeur supérieure ou inférieure diminue les performances.

Organisation
Pour ce projet, nous avons utilisé google colab pour la plupart des exercices, cet outil nous permet de travailler sur des notebooks python simultanément comme google doc.

Exercice 1: Réalisé par Philippe-Jacques, il a eu des soucis donc nous avons essayé de l'aider mais en vain.

Exercice 2: Réalisé par Baptiste, il a eu aussi des soucis, nous avons aussi travaillé dessus ensemble dessus mais nous n'avons pas réussi à compléter l'exercice.

Exercice 3: Réalisé par Tristan.

Exercice 4: Réalisé par Tristan.

Exercice 5: Réalisé par Tristan, Philippe-jacques et Baptiste.