

# Projet Python

## Les\_Maladies\_Cardiaques

### I. Analyse de Clustering et de Classification sur le Jeu de Données des Maladies Cardiaques

#### 1. Jeu de Données :

- Jeu de Données : [Heart Attack UCI](#)
- Description : Ce jeu de données comprend divers attributs liés aux maladies cardiaques. L'objectif est de prédire la présence ou l'absence de maladies cardiaques.

#### 2. Tâches de Classification :

##### a. Préparation des Données :

- Utiliser le jeu de données des maladies cardiaques avec la variable cible indiquant la présence ou l'absence de maladies cardiaques.
- Effectuer le prétraitement nécessaire des données, traiter les valeurs manquantes, encoder les variables catégorielles, etc.

##### b. Algorithmes de Classification :

###### b.1 . Arbre de Décision :

- Implémenter un classifieur d'arbre de décision.
- Visualiser l'arbre de décision généré.
- Évaluer le classifieur en utilisant des métriques (précision, rappel, etc.).

###### b.2 .Forêt Aléatoire :

- Implémenter un classifieur de forêt aléatoire.
- Visualiser l'importance des caractéristiques.
- Ajuster les hyperparamètres si nécessaire.
- Évaluer le classifieur.

###### b.3 .AdaBoost :

- Implémenter un classifieur AdaBoost.
- Visualiser l'importance des classificateurs faibles.
- Évaluer le classifieur.

###### b.4 .Machine à Vecteurs de Support (SVM) :

- Implémenter un classifieur SVM.

- Visualiser la séparation des classes dans l'espace des caractéristiques.
- Ajuster les hyperparamètres si nécessaire.
- Évaluer le classifieur.

#### **b.5 .K-Nearest Neighbors (KNN) :**

- Implémenter un classifieur KNN.
- Visualiser les régions de décision.
- Ajuster les hyperparamètres si nécessaire.
- Évaluer le classifieur.

#### **b.6 .Naive Bayes :**

- Implémenter un classifieur Naive Bayes.
- Visualiser les probabilités a priori et postérieures.
- Évaluer le classifieur.

### **3.Tâches de Clustering :**

#### **a. Chargement des Données et Visualisation :**

- Charger le jeu de données sur les maladies cardiaques.
- Visualiser le jeu de données dans une grille 2D avec des points en utilisant des caractéristiques pertinentes.

#### **b. Algorithmes de Clustering :**

##### **b.1 .k-Means Clustering :**

- Appliquer k-Means avec un nombre prédéfini de clusters
- Visualiser les résultats du clustering.
- Explorer les métriques (silhouette score, etc.) pour évaluer la qualité du clustering.
- Itérer pour trouver le nombre optimal de clusters en utilisant les métriques d'évaluation. Mesurer le temps de calcul.
- Expliquer les limites de K-means avec un exemple de jeu de données

##### **b.2 .Clustering Hiérarchique Agglomératif :**

- Appliquer le clustering hiérarchique agglomératif avec un nombre prédéfini de clusters. - Visualiser les dendrogrammes et les résultats du clustering.
- Utiliser les métriques d'évaluation pour analyser la qualité du clustering. - Itérer pour trouver le nombre optimal de clusters. Mesurer le temps de calcul. - Expliquer les limites du clustering agglomératif avec un exemple de jeu de données

