



Prédiction des maladies cardiovasculaires avec le machine learning

Tiouajni Sirine, Akriche Sahar et Arbi Aya
2024/2025



Table des matières



01

Introduction

02

Exploration et
Analyse des
données

03

Prétraitement des
données

04

Visualisation des
données

05

Algorithmes de
Machine Learning

06

comparaison et
sélection de modèle

07

Conclusion





01

Introduction



Les maladies cardio-vasculaires représentent une des principales causes de mortalité dans le monde.

17,900,000

de personnes sont mortes de maladies cardiovasculaires en 2019



Ce projet a pour objectif de répondre à la problématique suivante :

comment utiliser les données existantes pour construire un modèle capable de prédire avec précision la présence d'une maladie cardio-vasculaire ?

La solution proposée consiste à appliquer des techniques de machine learning pour analyser les données des patients et fournir un diagnostic basé sur leurs caractéristiques médicales.
Cela permettrait d'intervenir rapidement et de prévenir des complications graves.



02

Exploration et Analyse des données

Notre data Set a pour nom **Risk Factors for Cardiovascular Heart Disease** et localisée sur le site officiel de Kaggle
Une première observation montre que la data set est composée de 70000 lignes et 14 colonnes comme suit:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	index	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
2	0	0	18393	2	168	62	110	80	1	1	0	0	1	0
3	1	1	20228	1	156	85	140	90	3	1	0	0	1	1
4	2	2	18857	1	165	64	130	70	3	1	0	0	0	1
5	3	3	17623	2	169	82	150	100	1	1	0	0	1	1
6	4	4	17474	1	156	56	100	60	1	1	0	0	0	0
7	5	8	21914	1	151	67	120	80	2	2	0	0	0	0
8	6	9	22113	1	157	93	130	80	3	1	0	0	1	0
9	7	12	22584	2	178	95	130	90	3	3	0	0	1	1
10	8	13	17668	1	158	71	110	70	1	1	0	0	1	0
11	9	14	19834	1	164	68	110	60	1	1	0	0	0	0
12	10	15	22530	1	169	80	120	80	1	1	0	0	1	0
13	11	16	18815	2	173	60	120	80	1	1	0	0	1	0
14	12	18	14791	2	165	60	120	80	1	1	0	0	0	0
15	13	21	16909	1	158	70	110	70	1	1	0	0	1	0

L'analyse de dataset montre que l'apprentissage sera supervisé en utilisant des algorithmes de classification, **mais pourquoi?**

Apprentissage supervisé

Nous disposons d'une variable cible (y) déjà définie (la colonne cardio) indiquant si un patient est malade ou non. L'apprentissage supervisé repose sur l'utilisation de ces labels (ou outputs) pour entraîner les modèles, ce qui correspond parfaitement à notre cas.

Classification

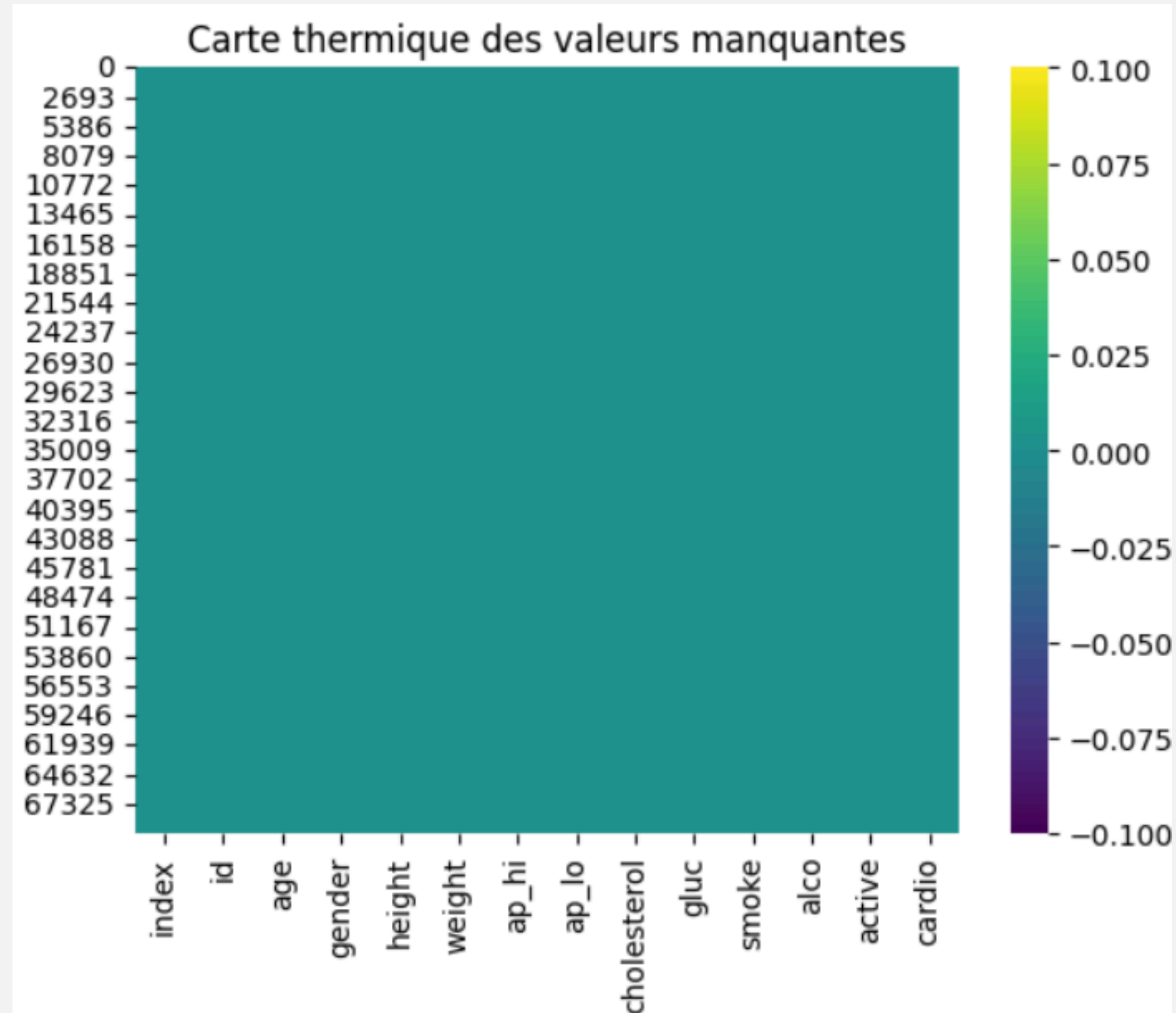
La nature discrète de la variable cible (valeurs binaires : 0 ou 1) indique clairement que le problème est de type classification, et non régression, qui aurait été utilisée pour prédire une variable continue. Les algorithmes de classification permettent ainsi de répondre à notre question principale : un patient est-il à risque (1) ou non (0) ?



03

Prétraitement des données

1. Analyse des valeurs manquants



=> pas de valeurs manquants pour chaque variable

2. Analyse des éléments dupliqués

=> pas de valeurs dupliqués (voir code)

3. Analyse des colonnes inutiles

Après analyse de notre dataset , nous avons constaté que les colonnes index et id sont inutiles => suppression de la colonnes id et index

	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
0	18393	2	168	62.0	110	80	1	1	0	0	1	0
1	20228	1	156	85.0	140	90	3	1	0	0	1	1
2	18857	1	165	64.0	130	70	3	1	0	0	0	1
3	17623	2	169	82.0	150	100	1	1	0	0	1	1
4	17474	1	156	56.0	100	60	1	1	0	0	0	0

4. Ajustement des valeurs

Puisque la variable age est exprimé par jours , on va faire la conversion en ans

age		age
18393	=>	50
20228		55
18857		51
17623		48
17474		47

=> Données plus compréhensible

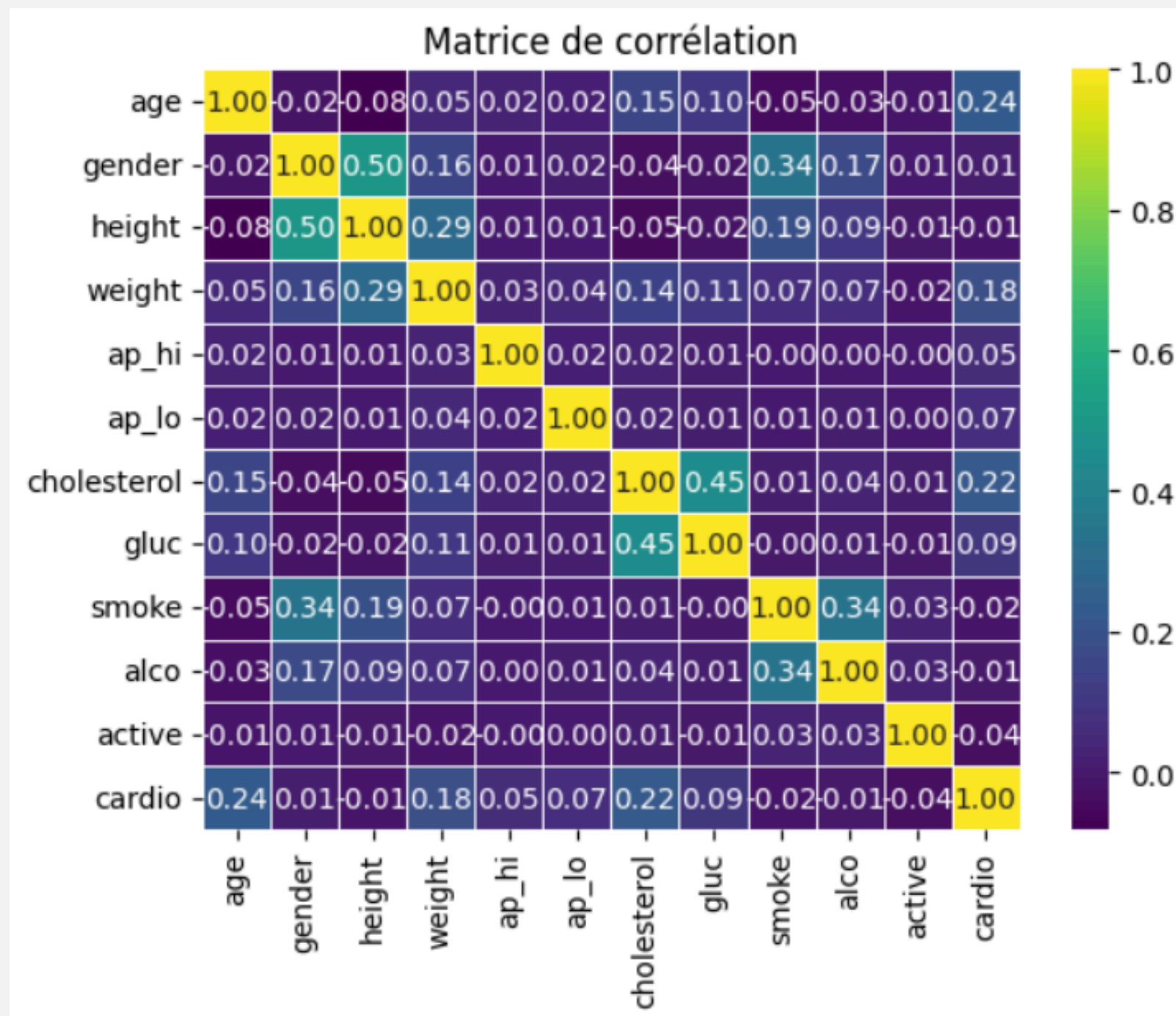


04

Visualisation des données

1. Visualisation de la matrice de corrélation avec un heatmap

La matrice de corrélation affiche les coefficients de corrélation entre chaque paire de variables



r proches de -1 => Corrélation négative
si l'un augmente l'autre diminue

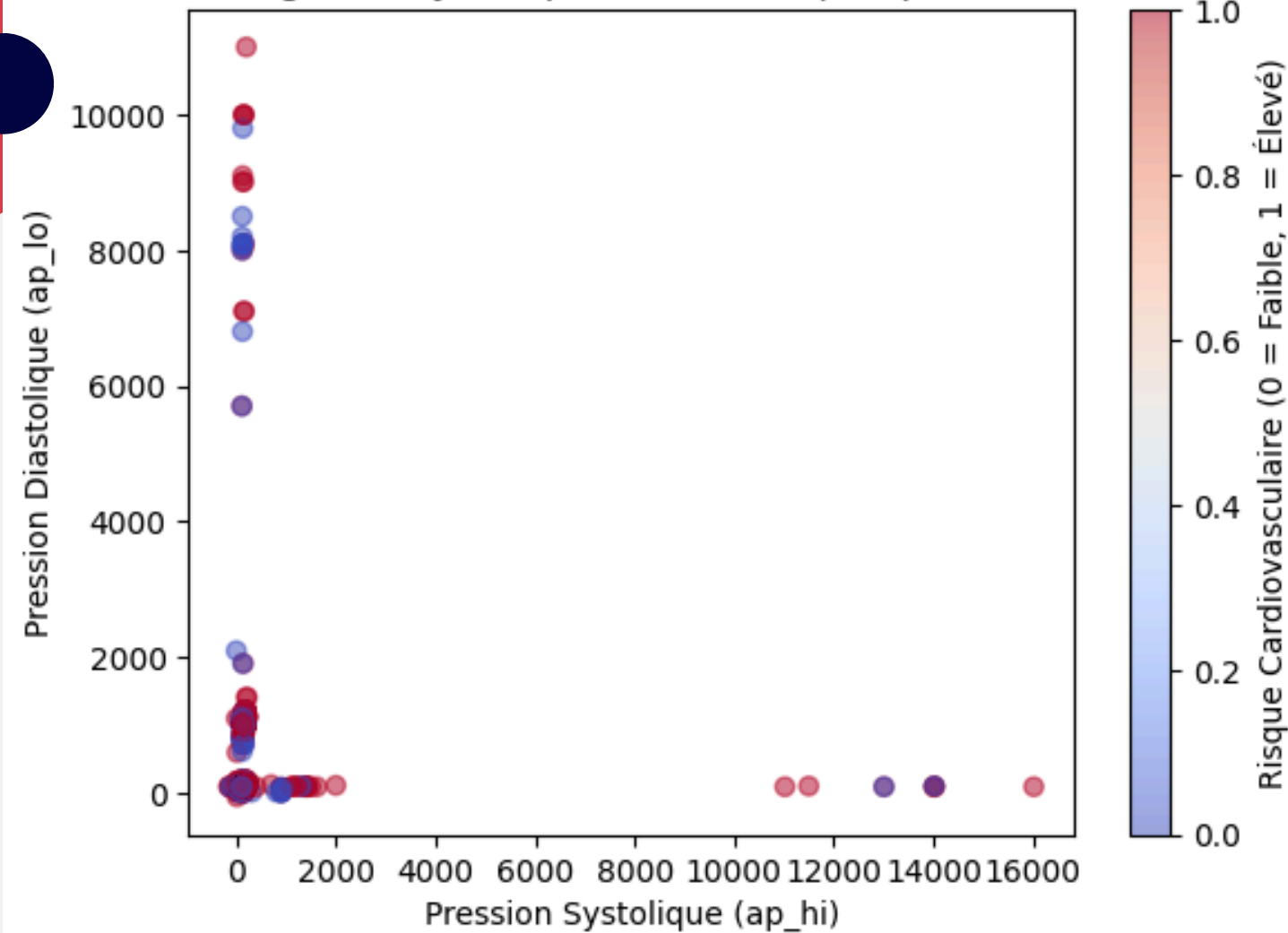
r proches de 0 => Faible corrélation
Les deux variables sont indépendantes

r proches de 1 => Corrélation positive
lorsque la valeur d'une variable
augmente, l'autre augmente de
manière proportionnelle.

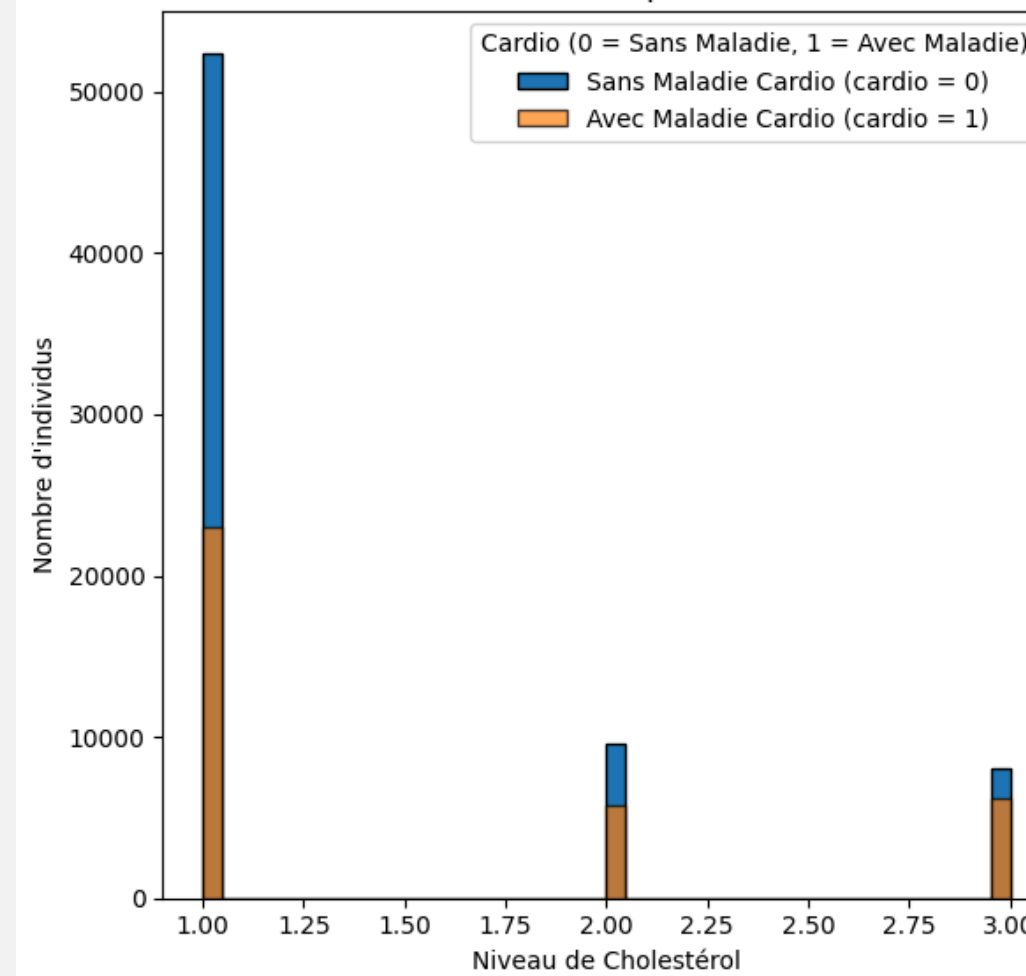


Cette ensemble des courbes represente la l'influence de certains critères (age, genre, taille,alcool...) sur la santé cardio-vasculaire de patient.

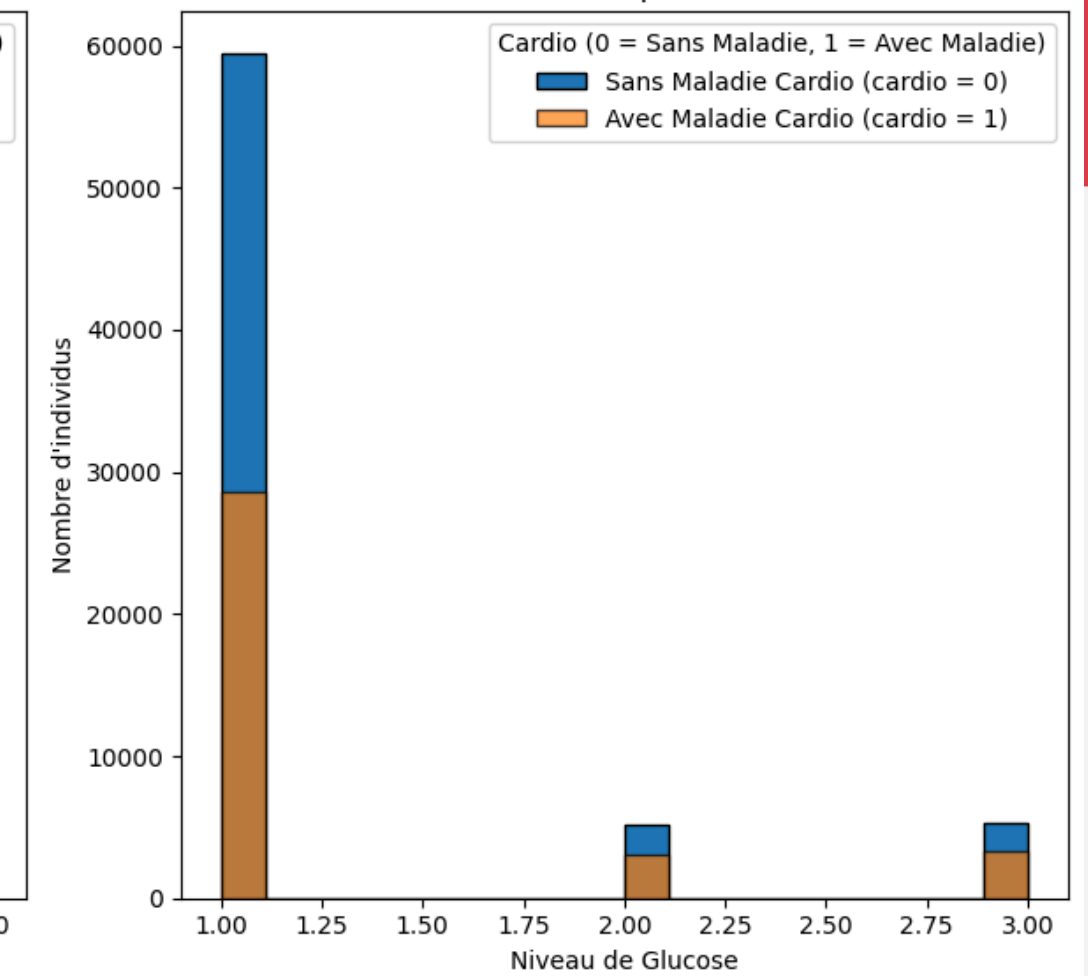
Pression Sanguine (Systolique vs Diastolique) par Statut Cardio



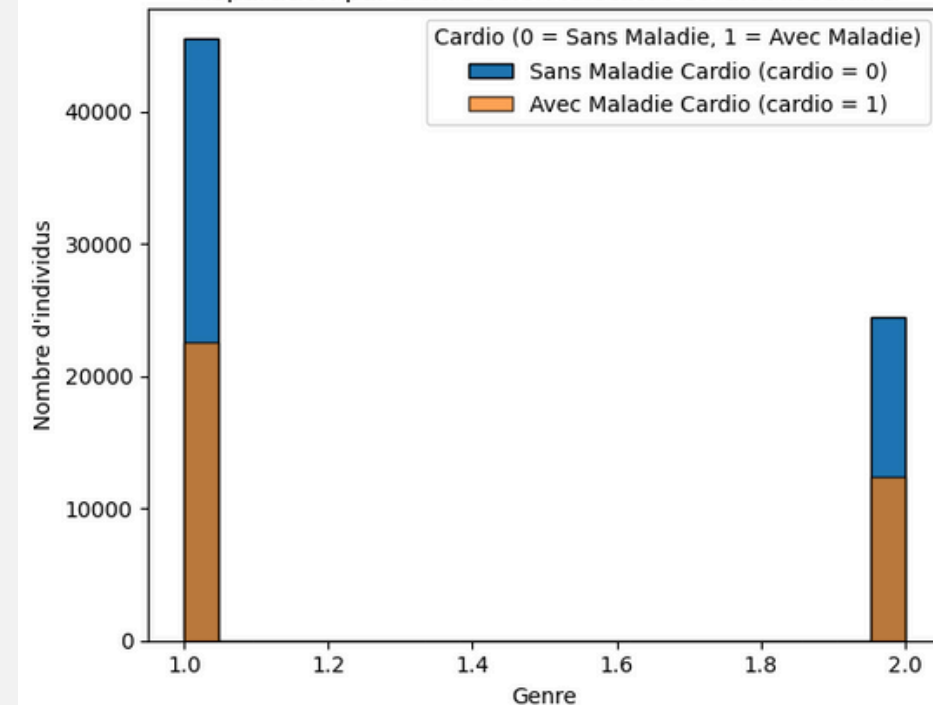
Niveau de Cholestérol par Statut Cardio



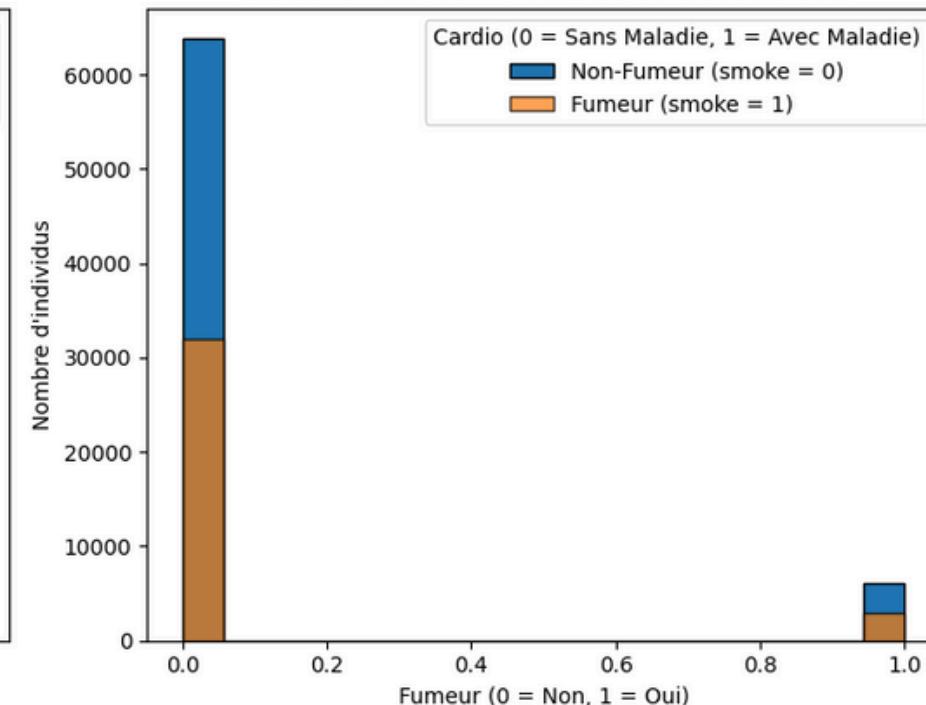
Niveau de Glucose par Statut Cardio



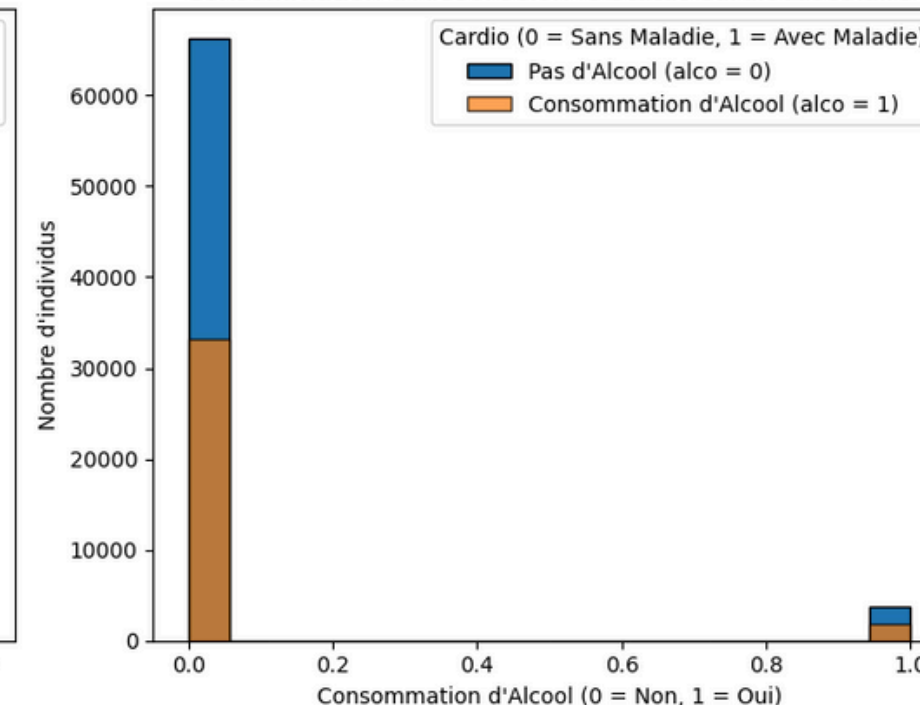
Répartition par Genre selon le Statut Cardiovasculaire



Statut Fumeur selon le Statut Cardiovasculaire



Consommation d'Alcool selon le Statut Cardiovasculaire



Ces histogrammes
representent la
distribution des patients
selon la pression
sanguine, le taux du
cholestérol, la
consommation d'alcool
et tabac par statut
cardio-vasculaire.



05

Algorithmes de Machine Learning



Avant l'application des algorithmes :

Normaliser les données :

transformer les valeurs de vos données **pour qu'elles soient sur une échelle commune**, sans les écarter de leur distribution d'origine. Cela améliore l'efficacité et la précision des algorithmes de machine learning

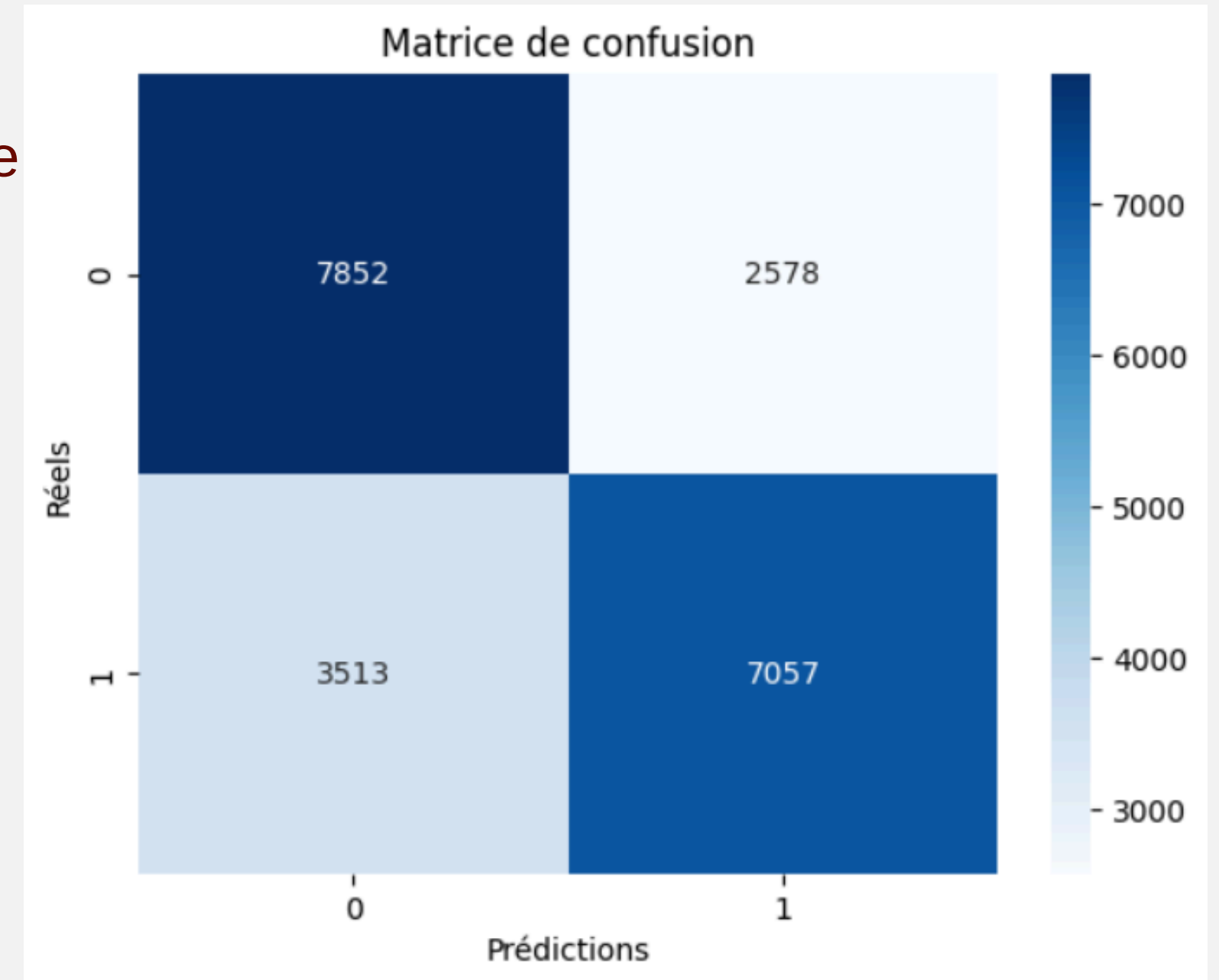
Diviser les données en ensemble d'entraînement et de test :

garantit que notre modèle n'est pas seulement performant sur les données sur lesquelles il a été entraîné, mais qu'il peut également bien fonctionner sur de nouvelles données.

- **30% de données de test**
 - **70% d'entraînements .**
- 

La régression logistique

- La régression logistique prédit une probabilité qu'une observation appartienne à la classe indiquant la présence d'une maladie cardiovasculaire (classe 1).
- Elle utilise la fonction sigmoïde pour transformer les prédictions linéaires en probabilités qui se situent entre 0 et 1.
- Si la probabilité est $\geq 0,5$, on prédit la classe 1.
- Si elle est $< 0,5$, on prédit la classe 0.



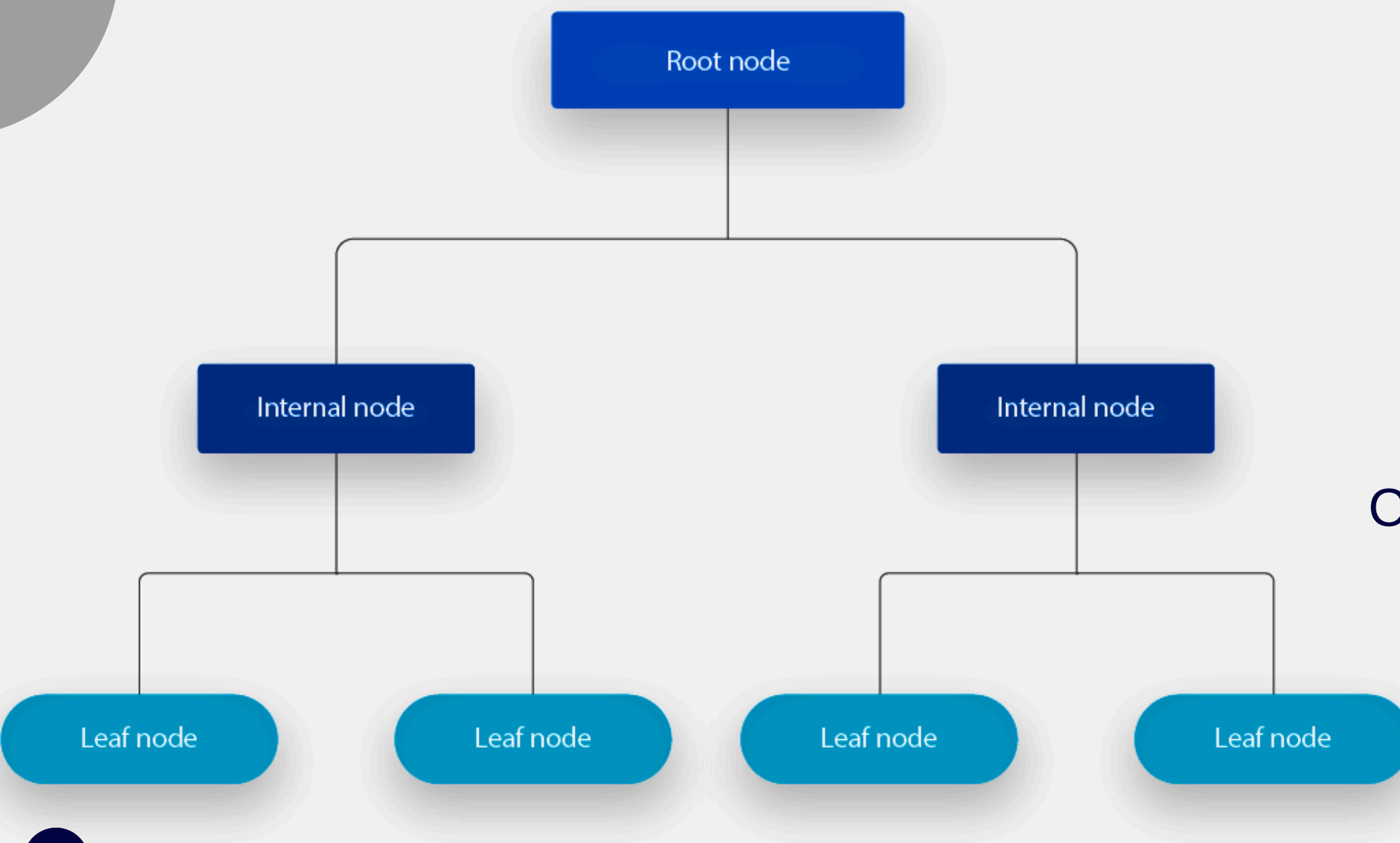
k-plus proches voisins (KNN)

Il classe un nouvel exemple en fonction de la majorité des voisins les plus proches.

- Calcul de la distance : entre le nouvel exemple et tous les exemples d'entraînement.
- Trouver les K voisins les plus proches : La valeur de K est un hyperparamètre que vous choisissez.
- Vote majoritaire : Si la majorité des K voisins appartiennent à une certaine classe, alors le nouvel exemple sera classé dans cette classe.

```
Score de précision pour k=2 : 63.77%  
Score de précision pour k=12 : 70.95%  
Score de précision pour k=22 : 71.25%
```

Arbre de décisions



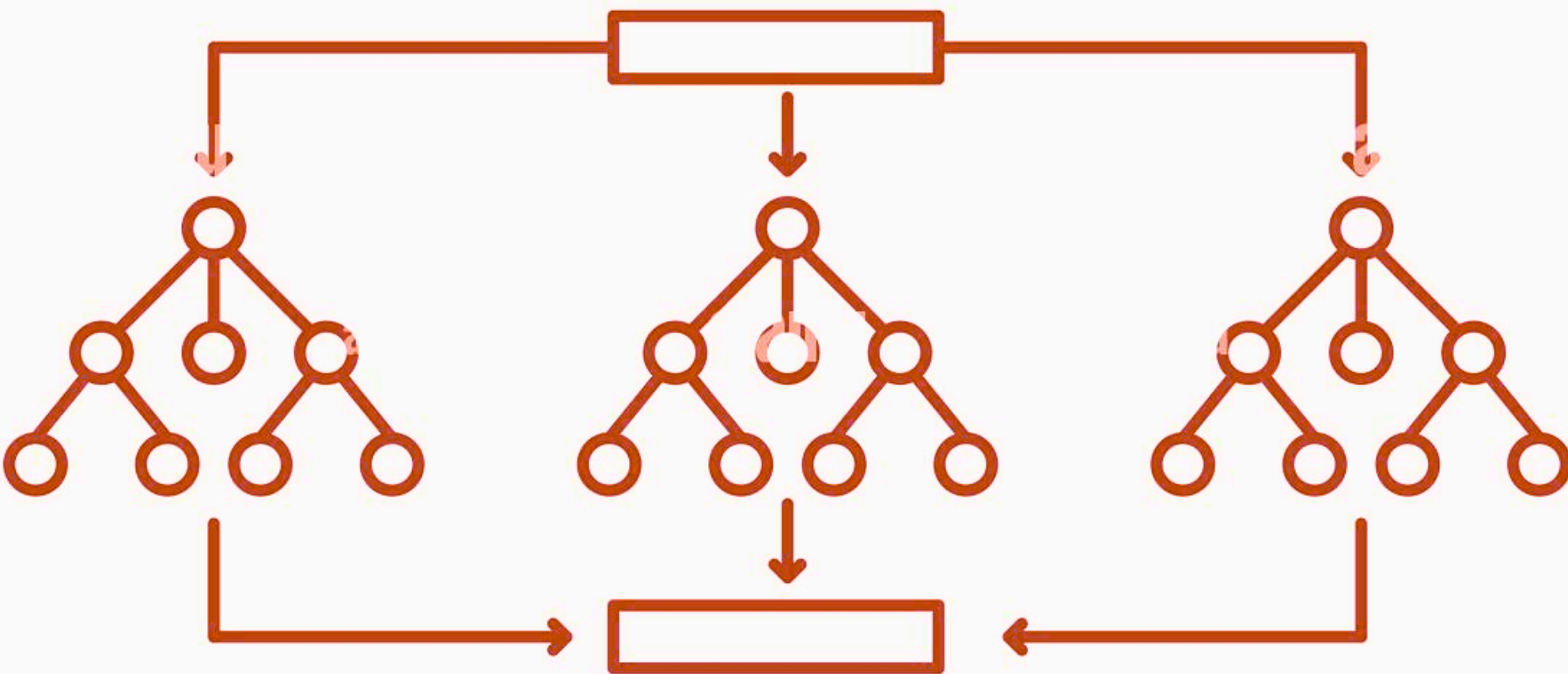
- Divise l'espace des données en segments à l'aide de questions successives sur les caractéristiques.
- Les nœuds internes représentent des conditions (ex. : "âge > 50 ans").
- Les feuilles correspondent aux décisions finales (ex. : classe 0 ou 1 dans notre cas).

Construction de l'arbre :

- Basée sur des critères tels que le gain d'information, l'entropie, ou la réduction de l'impureté (Gini).

Forêt Aléatoire

- Entraîne plusieurs arbres de décision sur des sous-échantillons aléatoires des données.
- Combine leurs prédictions par Vote majoritaire



- Avantages :
 - Performant même sur des données bruyantes.
 - Moins sujet au surapprentissage par rapport aux arbres de décision individuels.
- Limites :
 - Moins interprétable que les arbres de décision.
 - Plus coûteux en calcul, notamment pour les ensembles de données volumineux.

Gradient Boosting Machine (GBM)

combinaison de plusieurs modèles faibles (souvent des arbres de décision) pour créer un modèle global plus robuste

- Initialiser un modèle GBM avec des hyperparamètres de base.
- définition des: Nombre d'arbres / Taux d'apprentissage / Profondeur maximale des arbres / Réplicabilité
- Le GBM apprend itérativement en ajustant les erreurs (résidus) des prédictions précédentes.
- Chaque arbre est entraîné pour minimiser une fonction de perte, telle que la log-loss dans un problème de classification.
- Le modèle utilise les arbres entraînés pour prédire les probabilités des classes ou les classes elles-mêmes.

Pour chaque modèle développé on a utilisé le GridSearch **Une méthode de recherche exhaustive de la meilleure combinaison d'hyperparamètres.** qui nous a donné ces résultats

Scores de validation croisée et meilleurs hyperparamètres pour chaque modèle :

KNN :

Score : 0.7069

Meilleurs hyperparamètres : {'n_neighbors': 9, 'weights': 'uniform'}

Random Forest :

Score : 0.7351

Meilleurs hyperparamètres : {'max_depth': 10, 'min_samples_split': 2, 'n_estimators': 200}

Decision Tree :

Score : 0.7283

Meilleurs hyperparamètres : {'max_depth': 10, 'min_samples_split': 5}

Logistic Regression :

Score : 0.7226

Meilleurs hyperparamètres : {'C': 10, 'solver': 'liblinear'}

GBM :

Score : 0.7364

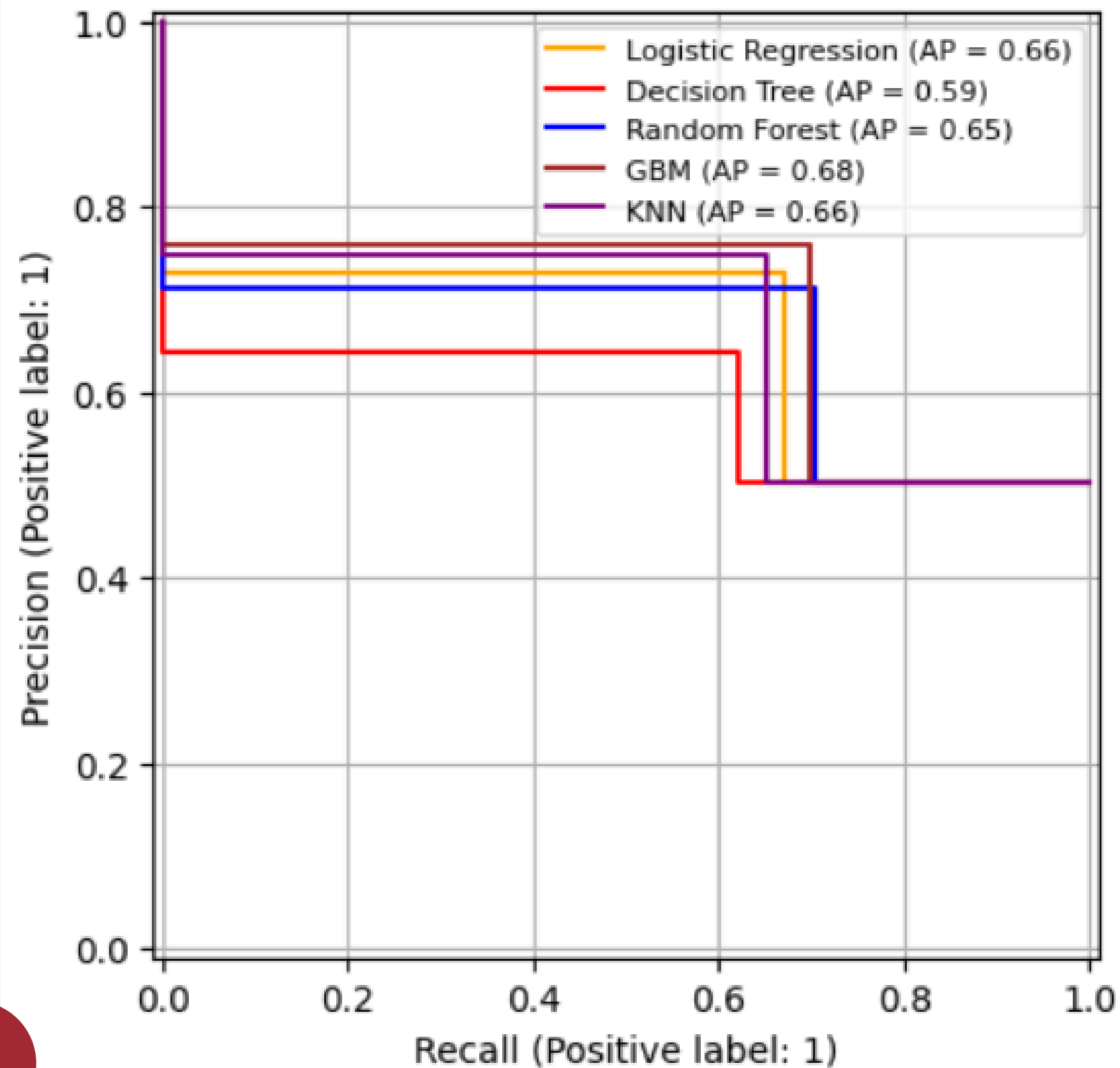
Meilleurs hyperparamètres : {'learning_rate': 0.1, 'max_depth': 5, 'n_estimators': 100}



06

Comparaison et sélection de modèle

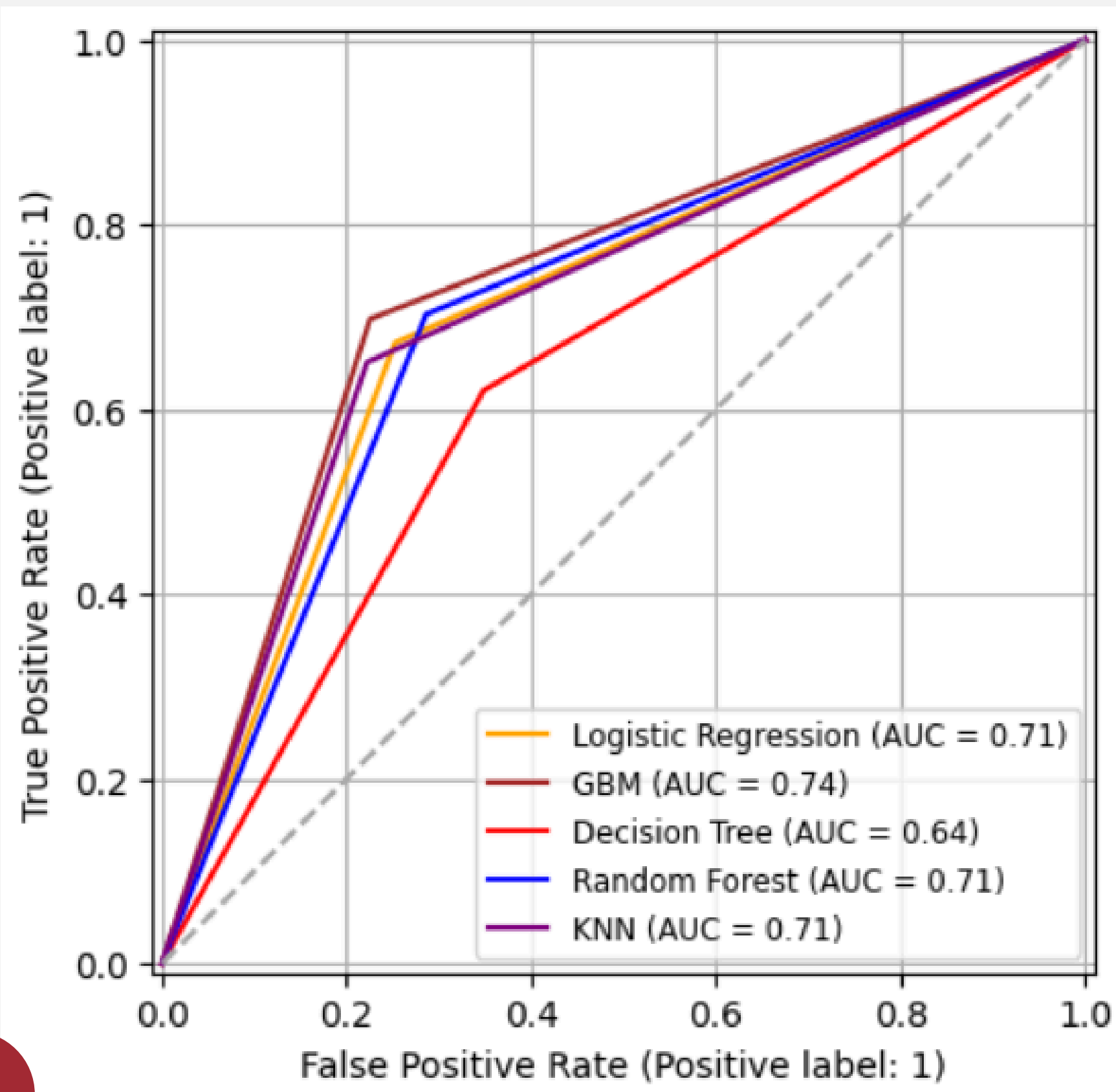
Pour évaluer la performance de nos modèles de prédiction, nous avons utilisé deux méthodes d'évaluation couramment utilisées en Machine Learning : **la courbe Précision-Rappel**



Cette courbe nous a permis d'évaluer l'équilibre entre la précision et le rappel pour prédire correctement les patients malades, en particulier pour les situations où nous souhaitons éviter les faux négatifs.

on peut conclure que les modèles **SVC** et **GBM** offrent les meilleurs performance

La courbe ROC



La courbe ROC nous a permis d'analyser la capacité de nos modèles à séparer correctement les classes, en mesurant le compromis entre le taux de vrais positifs (sensibilité) et le taux de faux positifs.

En observant l'AUC, nous avons pu identifier le modèle offrant la meilleure performance qui est **GBM**



Accuracy

Scores de validation croisée pour chaque modèle :

KNN : 0.7069

Random Forest : 0.7351

Decision Tree : 0.7283

Logistic Regression : 0.7226

GBM : 0.7364

Nous avons sélectionné le modèle
GBM comme étant le plus
performant



GBM sur les données Test

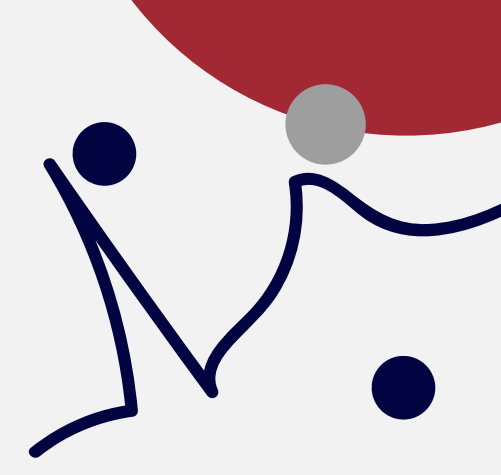
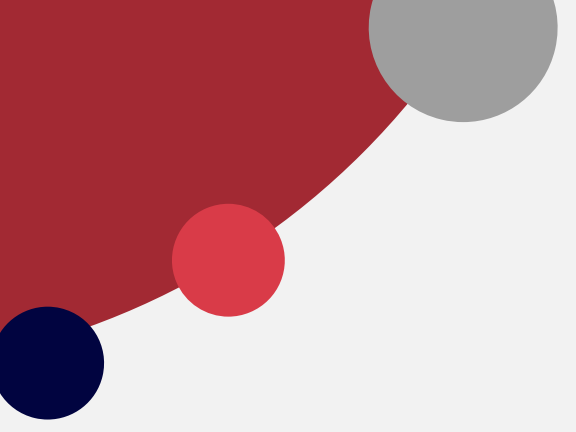
Précision du meilleur modèle (GBM) sur les données de test : 0.7348

Rapport de classification :

	precision	recall	f1-score	support
0	0.71	0.78	0.74	10430
1	0.76	0.69	0.73	10570
accuracy			0.73	21000
macro avg	0.74	0.74	0.73	21000
weighted avg	0.74	0.73	0.73	21000



07 Conclusion



Dans ce projet de prédiction des maladies cardiovasculaires, nous avons testé plusieurs modèles de machine learning:

l'Arbre de Décision, Random Forest, K-Nearest Neighbors (KNN), Gradient Boosting Machine (GBM), et la Régression Logistique.

Parmi ces modèles, le Gradient Boosting Machine (GBM) s'est avéré être le plus performant en se basant sur l'évaluation de la précision, le rappel, ainsi que l'AUC des courbes ROC et Précision-Rappel,

Ce modèle a donc été retenu comme le meilleur choix pour notre cas de prédiction.





Merci pour votre
attention !