République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université AMO de Bouira

Faculté des Sciences et des Sciences Appliquées

Département d'Informatique

Mémoire de Master

en Informatique

Spécialité : *Génie des Systèmes Informatiques*

Thème

Optimisation de la sélection d'attribut via l'algorithme génétique rival pour la prédiction des maladies cardiovasculaires

---

Encadré par :

— Mme Brahimi Farida

Réalisé par :

— Feradji Houria

— Bouchibane Bouchra

2023/2024

Remerciements

Tout d'abord, nous remercions Allah pour la force, la patience et le courage qu'il nous a donnés pour réaliser ce travail. Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué à la réalisation de ce mémoire.

Nous remercions notre encadrante, Mme BRAHIMI farida, enseignante à la Faculté des sciences et des sciences appliquées de Bouira, pour ses conseils précieux, sa disponibilité et son soutien indéfectible tout au long de ce projet. Ses encouragements et ses remarques constructives ont été déterminants pour mener à bien cette recherche.

Nous tenons également à exprimer notre profonde gratitude aux membres du jury d'avoir accepté d'évaluer ce travail. Leurs critiques constructives seront précieuses pour enrichir nos connaissances et améliorer notre recherche.

Nous adressons aussi nos sincères remerciements à tous nos enseignants du département d'informatique de l'Université de Bouira Akli Mohand Oulhadj pour leur enseignement de qualité et leur accompagnement tout au long de notre parcours académique.

Bouchibane Bouchra

Feradji Houria

Dédicaces

Je dédie ce travail à :

À mes très chers parents, source de vie, d'amour et d'affection, qui m'ont soutenu et encouragé durant ces années d'études. Vos encouragements et votre foi en moi ont été les piliers de ma réussite. Merci pour tout ce que vous avez fait pour moi et pour avoir toujours cru en mes capacités.

À mes chers frères et sœurs, et à leurs enfants, source de joie et de bonheur.

À toute ma famille, source d'espoir et de motivation.

À ma chère binôme Bouchra, pour son travail acharné, sa collaboration et son amitié. Ton dévouement et ta persévérance ont été essentiels à la réalisation de ce projet. Merci pour tous les moments partagés et pour avoir été un partenaire exceptionnel tout au long de cette aventure.

À tous mes amis, particulièrement Ahlem, Amina,Bouchra et Dhaouia, qui m'ont toujours encouragé, et à qui je souhaite plus de succès.

Feradji Houria.

Dédicaces

Ce travail est un hommage à tous ceux qui m'ont encouragé et soutenu dans ma carrière académique.

Je remercie mes parents pour leur amour indéfectible, leur soutien inconditionnel et leurs précieux conseils qui m'ont aidé à devenir la personne que je suis aujourd'hui.

Un merci particulier à mes frères.

À mon binôme Houria, dans cette aventure académique, ta collaboration et ton soutien ont été essentiels. Ensemble, nous avons surmonté les défis et grandi. Merci pour tout.

À tous ceux qui m'ont accompagné dès le premier pas jusqu'à l'avant-dernier, à ceux qui ont été comme des nuages bienfaisants pendant ces années, je vous suis profondément reconnaissant.

Bouchibane Bouchra.

Résumé

Les maladies cardiovasculaires (MCV) sont la principale cause de décès dans le monde. Les patients atteints de maladies cardiaques ne ressentent généralement pas de symptômes avant les stades avancés de la maladie, et beaucoup d'entre eux décèdent avant de recevoir un traitement. Par conséquent, la détection et la prévention précoces de ces maladies sont cruciales. Ces dernières années, diverses approches basées sur l'apprentissage automatique et l'apprentissage profond ont été développées dans le domaine médical pour prédire les maladies cardiaques avant leur manifestation.

La sélection d'attributs est une étape clé dans les tâches de classification. Elle consiste à sélectionner un sous-ensemble d'attributs pertinents qui peuvent contribuer de manière significative au processus de classification. Actuellement, de nombreux algorithmes d'optimisation métaheuristiques ont été appliqués avec succès à la sélection d'attributs.

Dans ce travail, la sélection d'attributs pertinents est réalisée à l'aide de l'algorithme génétique rival (AGR), qui est une amélioration de l'algorithme génétique (AG). L'AGR utilise une stratégie de compétition combinant de nouveaux schémas de sélection et de croisement pour améliorer la capacité de recherche globale. De plus, un taux de mutation dynamique est introduit pour optimiser le comportement de recherche de l'algorithme durant le processus de mutation.

Un modèle de prédiction des maladies cardiovasculaires basé sur le perceptron multicouche (MLPClassifier) a été développé en utilisant les attributs sélectionnés. Les performances de ce modèle de prédiction ont été évaluées et comparées à plusieurs techniques d'apprentissage automatique et d'apprentissage profond. Les résultats expérimentaux montrent que notre modèle proposé surpasse plusieurs méthodes en termes de métriques d'évaluation des performances. L'accuracy obtenue par le modèle proposé est de 94,92 %.

Mots clés : Apprentissage automatique, Apprentissage profond, Modèle de classification, Maladies cardiovasculaire, Sélection d'attributs, Algorithme génétique, Algorithme génétique rival. ...

Abstract

Cardiovascular diseases (CVD) are the leading cause of death worldwide. Patients with heart diseases generally do not experience symptoms until the advanced stages of the disease, and many of them die before receiving treatment. Therefore, early detection and prevention of these diseases are crucial. In recent years, various approaches based on machine learning and deep learning have been developed in the medical field to predict heart diseases before their manifestation.

Feature selection is a key step in classification tasks. It involves selecting a subset of relevant attributes that can significantly contribute to the classification process. Currently, many metaheuristic optimization algorithms have been successfully applied to feature selection.

In this work, relevant feature selection is carried out using the Rival Genetic Algorithm (RGA), which is an enhancement of the Genetic Algorithm (GA). The RGA employs a competition strategy that combines new selection and crossover schemes to improve global search capability. Additionally, a dynamic mutation rate is introduced to optimize the search behavior of the algorithm during the process.

A cardiovascular disease prediction model based on the Multi-Layer Perceptron (MLP) has been developed using the selected attributes. The performance of this prediction model was evaluated and compared to several machine learning and deep learning techniques. Experimental results show that our proposed model outperforms several methods in terms of performance evaluation metrics. The accuracy achieved by the proposed model is 94.92%.

Key words : Machine Learning, Deep Learning, Classification Model, Cardiovascular Diseases, Feature Selection, Genetic Algorithm, Rival Genetic Algorithm.. . .

Table des matières

Table des matières **i**

Table des figures **iv**

Liste des tableaux **vi**

Liste des abréviations **viii**

Introduction générale **1**

1 Généralités sur les maladies cardiovasculaires **4**

1.1 Introduction 4

1.2 Structure du coeur 4

1.3 Fonctionnement du coeur 5

1.4 Les maladies cardiovasculaires : 6

1.5 Les types et les symptômes des maladies cardiovasculaires 6

1.6 Les Facteurs de risque 7

1.6.1 Facteurs de risque non modifiables : 7

1.6.2 Facteurs de risque modifiables : 8

1.7 Prévention des maladies cardiovasculaires 8

1.7.1 La prévention primaire : 9

1.7.2 La prévention secondaire : 9

1.8 Conclusion 9

2 Apprentissage automatique et Sélection des attributs **10**

2.1 Introduction : 10

2.2 Apprentissage automatique 10

2.2.1 Types d'apprentissage automatique 10

2.2.2 Les algorithmes apprentissage automatique : 11

2.3 L'apprentissage profond 16

2.4 Resaux de neurones artificile : 17
2.4.1 Neurone biologique : 17
2.4.2 Neurone formel : 18
2.4.3 L'apprentissage des réseaux de neurones 19
2.4.4 Type de resaux de neurones artificile : 20
2.5 Sélection des attributs 22
2.5.1 Importance de la sélection des attributs 22
2.5.2 Processus de sélection des attributs 23
2.5.3 Les méthodes de sélection : 23
2.5.4 Algorithmes génétiques (AG) pour la sélection des attributs : 25
2.5.5 Algorithme génétique rival pour la sélection d'attributs 27
2.6 Évaluation des performances d'un modèle 30
2.6.1 Matrice de confusion : 30
2.6.2 Métriques d'évaluation : 31
2.6.3 Métriques d'erreur 31
2.7 Les explications additives de Shapley : 32
2.8 Conclusion 32

3 Méthodologie proposée **33**
3.1 Introduction : 33
3.2 Etat de l'art 33
3.3 L'approche proposée 34
3.4 Analyse exploratoire 35
3.4.1 Analyse exploratoire de la première base de données (Cleveland) 35
3.4.2 Analyse exploratoire de la deuxième base de données (Heart Failure) 45
3.5 Prétraitement : 49
3.5.1 Prétraitement de premier ensemble de données 49
3.5.2 Prétraitement de deuxième ensemble de données : 51
3.6 La sélection d'attributs 52
3.6.1 La sélection d'attributs par l'algorithme génétique 53
3.6.2 La sélection d'attributs par l'algorithme génétique rival 56
3.7 Classification des données 64
3.7.1 Hyperparamètres des modèles implémentés 65
3.8 Conclusion 66

4 Résultats expérimentaux **67**
4.1 Introduction 67
4.2 Évaluation des performances des modèles pour les données de Cleveland 67
4.3 Évaluation des performances des modèles pour les données de Heart Failure 69
4.4 Évaluation des métriques d'erreurs des modèles pour les données de Cleveland 70

4.5 Évaluation des métriques d'erreurs des modèles pour les données de Heart Failure 72
4.6 Comparaison des Travaux de Recherche 73
4.7 Détails des performances du modèle MLP_AGR pour le premier ensemble de données . . . 74
4.7.1 La matrice de confusion 74
4.7.2 Le rapport de classification 74
4.7.3 Courbe ROC et AUC : 75
4.7.4 Explication du modèle à l'aide de SHAP : 75
4.7.5 Test du modèle prédictif proposé 77
4.8 Conclusion 78

Conclusion générale et perspectives **79**

Table des figures

1.1 Vue générale du coeur [1] 5
1.2 Les maladies cardiovasculaires dans le corps [2] 6
1.3 Facteurs de risque modifiables [3] 8

2.1 Illustration du K plus proches voisins pour la classification [4] 12
2.2 Illustration de la classification binaire avec un SVM[5] 14
2.3 Exemple de réseau bayésien 15
2.4 Exemple de CPT 16
2.5 Neurone biologique [6] 18
2.6 Neurone formel [7] 18
2.7 Sigmoid 19
2.8 Tanh 19
2.9 ReLU 19
2.10 L'architecture de DNN[8] 20
2.11 L'architecture de RNN simple[9] 21
2.12 Processus de sélection des attributs 23
2.13 Méthodes de sélection 24
2.14 Concepts clés des algorithmes génétiques [10] 26
2.15 Fonctionnement de AG [11] 26
2.16 Exemple de croisement 27
2.17 Exemple de mutation 27
2.18 La stratégie de compétition 29
2.19 Matrice de confusion [12] 30

3.1 L'architecture du système proposé 35
3.2 Distribution statistiques des attributs 37
3.3 Visualisation de l'attribut cible 37
3.4 Distribution des variables continues 38
3.5 Distribution de fbs et exang 39

3.6 Distribution de restecg et sex . . . 40
3.7 Distribution de slope et target . . . 40
3.8 Distribution de thal . . . 40
3.9 Age,trestbps et chol Vs arget . . . 41
3.10 thalach et oldpeak Vs Target . . . 42
3.11 les caractéristiques catégorielles Vs target . . . 43
3.12 Matrice de corrélation . . . 44
3.13 les valeurs aberrantes pour les caractéristiques continues . . . 45
3.14 Distribution statistiques des attributs . . . 46
3.15 Visualisation de l'attribut cible . . . 47
3.16 Matrice de corrélation . . . 48
3.17 Les valeurs aberrantes . . . 49
3.18 Les valeurs manquantes . . . 50
3.19 Les valeurs de ca et thal . . . 50
3.20 Les valeurs manquantes (NaN). . . . 51
3.21 les valeurs dupliquées. . . . 52
3.22 Label Encoder . . . 52
3.23 La normalisation des attributs catégoriels et continues . . . 52
3.24 Représentation de chromosome en binaire . . . 56
3.25 Population initial de data1 . . . 57
3.26 Population initial de data2 . . . 57
3.27 Exemple de fonctionnement de mutation . . . 60

4.1 Visualisation des métriques d'evaluation . . . 68
4.2 Visualisation des métriques d'evaluation . . . 70
4.3 Visualisation des métriques d'erreur MAE et RMSE . . . 71
4.4 Visualisation des métriques d'erreur MAE et RMSE . . . 72
4.5 La matrice de confusion . . . 74
4.6 Classification report . . . 75
4.7 Courbe ROC . . . 75
4.8 Importance des caractéristiques selon les valeurs SHAP pour le modèle MLPClassifier_AGR 76
4.9 SHAP pour la ligne 225 . . . 76
4.10 SHAP pour la ligne 147 . . . 77

Liste des tableaux

1.1 Types et symptômes de maladies cardiaques 7

2.1 Structure et analogies du neurone formel 19

3.1 Description de la base de données 36
3.2 Description de la base de données 46
3.3 Population initiale avec leur fitness pour le premier ensemble de données 53
3.4 Population initiale avec leur fitness pour la deuxième ensemble de données 54
3.5 Paramètres d'exécution d'AG pour le dataset 1 55
3.6 Paramètres d'exécution d'AG pour le dataset 2 55
3.7 Résultats d'exécution de l'algorithme génétique pour data1 56
3.8 Résultats d'exécution de l'algorithme génétique pour data2 56
3.9 Exemples de la population initiale avec leur fitness pour le premier ensemble de données 57
3.10 Exemples de la population initiale avec leur fitness pour la deuxième ensemble de données 57
3.11 Paramètres d'exécution de l'AGR pour le dataset1 64
3.12 Paramètres d'exécution de l'AGR pour le dataset2 64
3.13 Rrésultats de l'exécution de AGR pour premier ensemble de données 64
3.14 Résultats de l'exécution de AGR pour deuxième ensemble de données 64
3.15 Hyperparamètres des modèles ML 65
3.16 Hyperparamètres du modèle MLPClassifier 65
3.17 Hyperparamètres du modèle DNN 66
3.18 Hyperparamètres des architectures RNN utilisées 66
3.19 Hyperparamètres de BN 66

4.1 Évaluation des performances des différents modèles 68
4.2 Évaluation des performances des différents modèles 69
4.3 Métriques d'erreur MAE et RMSE des différents modèles 71
4.4 Métriques d'erreur MAE et RMSE des différents modèles 72
4.5 Analyse comparative des travaux connexes 73

4.6 Tester le modèle avec certains patients dans un ensemble de données 77

Liste des abréviations

MCV	Maladies cardio-vasculaires
AVC	Accident vasculaiire cérébral
OMS	World Health Organization (Organisation mondiale de la santé)
MLPNN	Multi-Layer Perceptron Neural Network (Réseau de neurones à perceptron multicouche)
AOA	Angle of Arrival (Angle d'arrivée)
IoMT	Internet of Medical Things (Internet des objets médicaux)
HCNN	Hierarchical Convolutional Neural Network (Réseau de neurones convolutifs hiérarchique)
VGG16	Visual Geometry Group 16-layer network (Réseau de 16 couches du Groupe de Géométrie Visuelle)
ResNet	Residual Network (Réseau résiduel)
RAM	Random Access Memory (Mémoire vive)
LDL	lipoprotéines de basse densité
HDL	lipoprotéines de haute densité
RRS	Reynolds Risk Score
IA	Intelligence artificielle
AA	apprentissage automatique
KNN	K-Nearest Neighbors (Les plus proches voisins)
SVM	Support Vector Machine (Machine à vecteurs de support)
XGBoost	eXtreme Gradient Boosting
RF	Random Forest (Forêt aléatoire)
DT	Decision Tree (Arbre de décision)
DAG	Directed Acyclic Graph
BN	Réseaux Bayésiens
CPT	Tables de probabilité conditionnelle
ANN	Artificial Neural Network (Réseau de neurones artificiels)
DNN	Deep Neural Network (Réseau de neurones profond)
RNN	Recurrent Neural Network (Réseau de neurones récurrents)
CNN	Convolutional Neural Network (Réseau de neurones convolutifs)
MLP	Multi-Layer Perceptron (Perceptron multicouche)
LSTM	Long Short-Term Memory (Mémoire à long terme et court terme)
GRU	Gated Recurrent Unit (Unité récurrente à portes)

GPU	Graphics Processing Unit (Unité de traitement graphique)
CPU	Central Processing Unit (Unité de traitement centrale)
Anova	Analysis of Variance (Analyse de la variance)
VP	Vrais positifs
VN	Vrais négatifs
FP	Faux positifs
FN	Faux négatifs
MAE	erreur absolue moyenne
RMSE	la racine carrée de l'erreur quadratique moyenne
MSE	Mean Square Error (Erreur quadratique moyenne)
Lasso	Least Absolute Shrinkage and Selection Operator
ROC	Receiver Operating Characteristic (Caractéristique de fonctionnement du récepteur)
AG	Algorithme génétique (Genetic Algorithm)
AGR	Algorithme Génétique Rival
AGRF	Algorithme génétique rival rapide
SHAP	Explications Additives de Shapley
ML	Machine Learning (Apprentissage automatique)

Introduction générale

Selon l'Organisation mondiale de la santé, les maladies cardiovasculaires (MCV) sont la principale cause de décès dans le monde, causant environ 17,9 millions de morts chaque année [13]. Ces affections comprennent les maladies coronariennes, les maladies cérébrovasculaires et les cardiopathies rhumatismales. Plus de 80% des décès liés aux MCV sont dus à des crises cardiaques et des accidents vasculaires cérébraux, un tiers de ces décès survenant prématurément chez des individus de moins de 70 ans [14]. Cela souligne l'importance de la détection précoce et de la prévention des MCV. Malgré les progrès médicaux, le diagnostic et le traitement des MCV demeurent des défis majeurs, nécessitant une identification précise des facteurs de risque et une meilleure prévision des événements cardiovasculaires.

Récemment, avec les avancées technologiques et le développement de l'apprentissage automatique et de l'apprentissage profond, il est devenu possible d'améliorer considérablement la détection précoce et le diagnostic des maladies cardiovasculaires (MCV). L'apprentissage automatique, qui inclut une variété d'algorithmes capables d'apprendre à partir des données et de faire des prédictions ou des décisions sans être explicitement programmé pour chaque tâche, a transformé le domaine médical. Les techniques comme les forêts aléatoires, les machines à vecteurs de support (SVM), et le gradient boosting se sont révélées efficaces pour analyser de grandes quantités de données cliniques et identifier des motifs subtils indiquant un risque de MCV. L'apprentissage profond, sous-domaine de l'apprentissage automatique, utilise des réseaux de neurones artificiels pour modéliser des relations complexes dans les données. Les réseaux de neurones convolutifs (CNN) et les réseaux de neurones récurrents (RNN), par exemple, ont montré des performances exceptionnelles dans l'analyse des images médicales et des séries temporelles. Ces techniques permettent de détecter des anomalies dans les électrocardiogrammes (ECG) et les imageries médicales avec une précision et une rapidité accrues, améliorant ainsi la capacité à prévoir les événements cardiovasculaires et à personnaliser les traitements pour les patients à risque.

La prédiction des maladies cardiovasculaires repose sur une sélection efficace des caractéristiques, une étape cruciale dans les tâches de classification. Cette étape de prétraitement permet de choisir un petit sous-ensemble de caractéristiques significatives, maximisant ainsi la précision du processus de classification. De nos jours, de nombreux algorithmes d'optimisation métaheuristiques sont appliqués avec succès pour cette tâche. Parmi eux, l'algorithme génétique (AG) est couramment utilisé comme outil d'optimisa-

tion. L'algorithme génétique imite le processus de sélection naturelle en utilisant des opérations comme la sélection, le croisement et la mutation pour évoluer vers des solutions optimales. Toutefois, l'AG présente des limitations, notamment :

- **La sélection des parents :**Les parents sont choisis de manière aléatoire, ce qui peut entraıner la sélection de chromosomes de mauvaise qualité, ce qui entraıne la reproduction de solutions de mauvaise qualité.
- **Croisement :** En général, les AG utilisent un croisement entre deux parents avec un taux de croisement spécifique, ce qui peut ne pas élargir suffisamment l'espace de recherche, ce qui limite la capacité de l'algorithme à trouver des solutions optimales.
- **Mutation :**La mutation est une opération aléatoire qui modifie les gènes des chromosomes avec une probabilité donnée. Il est possible que l'exploration de nouvelles solutions et l'exploitation des solutions prometteuses ne soient pas équilibrées correctement avec un taux de mutation constant.

Pour résoudre ces problèmes lies aux algorithmes génétiques, nous implémentons une nouvelle approche de sélection d'attributs nommée Algorithme Génétique Rival (AGR) [15]. Cette méthode introduit de nouvelles stratégies, notamment :

- **Stratégie de compétition :**Il s'agit de diviser les chromosomes en gagnants et perdants. Cette méthode permet d'assurer une sélection de parents de haute qualité.
- **Croisement entre trois parents :** Le croisement entre trois parents, comprenant deux gagnants et un perdant, renforce l'exploration de l'espace de recherche et favorise la création de solutions de haute qualité.
- **Taux de mutation dynamique :** Le taux de mutation dynamique est ajusté en fonction de la fitness des individus et du numéro de génération pour améliorer la recherche de l'algorithme génétique et prévenir les pièges locaux.

L'AGR offre des avantages significatifs pour la sélection des caractéristiques dans la prédiction des maladies cardiovasculaires, notamment :

- **Amélioration de la précision de classification :** En sélectionnant les caractéristiques les plus significatives, l'AGR améliore la performance des modèles de classification utilisés dans la prédiction des MCV. Cela conduit à des outils de diagnostic plus précis et fiables, cruciaux pour la détection précoce et la planification du traitement.
- **Flexibilité et adaptabilité :** Les stratégies adaptatives de l'AGR le rendent flexible et capable de s'ajuster à différents types de données et de tâches de classification. Cette adaptabilité est bénéfique en recherche médicale, où la nature et la qualité des données peuvent varier considérablement.

Plan de Mémoire :

Dans le premier chapitre, nous offrons un aperçu général des maladies cardiovasculaires, en couvrant la structure et le fonctionnement du cœur, ainsi que les types, symptômes et facteurs de risque associés. Nous explorons également les stratégies de prévention primaire et secondaire pour contrer ces maladies.

Dans le deuxième chapitre, nous avons donné un aperçu de l'apprentissage automatique en expliquant certains algorithmes comme KNN, RF, DT, XGBoost, BN et SVM, puis nous avons exploré l'apprentissage profond. Ensuite, nous avons décrit le processus de sélection des attributs en mettant en évidence l'algorithme génétique et l'algorithme génétique rival. Nous avons également présenté les différentes métriques d'évaluation et les erreurs. Enfin, nous avons abordé les explications additives de Shapley.

Le troisième chapitre décrit en détail les ensembles de données utilisés, les étapes de la solution proposée, et présente quelques travaux de recherche récents.

Le quatrième chapitre présente une évaluation des performances des différents modèles implémentés tels que GRU, BN, KNN, RF, SVM, DT, LSTM, Simple RNN, DNN et le MLPClassifier, en utilisant trois configurations : sans sélection des attributs, sélection des attributs avec l'algorithme génétique et sélection des attributs avec l'algorithme génétique rival. Enfin, nous avons effectué une analyse détaillée du modèle MLPClassifier pour le jeu de données Cleveland, qui offre les meilleurs résultats.

Chapitre 1

Généralités sur les maladies cardiovasculaires

1.1 Introduction

Les maladies cardiovasculaires regroupent les pathologies qui touchent le cœur et l'ensemble des vaisseaux sanguins [16] qui peuvent provoquer de nombreuses complications aiguës ou chroniques telles que l'infarctus du myocarde, l'accident vasculaire cérébral, l'insuffisance cardiaque et l'insuffisance rénale chronique [17] .Leur prévalence croissante en fait une menace vitale à l'échelle mondiale.

L'objectif de ce chapitre est de présenter une vision globale des maladies cardiovasculaires en mettant l'accent sur les éléments clés, notamment en ce qui concerne la structure du cœur, le fonctionnement du cœur, les maladies cardiovasculaires, les types et les symptômes des maladies cardiovasculaires, les facteurs de risque non modifiables et modifiables, ainsi que de fournir quelques conseils pour la prévention.

1.2 Structure du coeur

Le cœur est formé par un muscle, le myocarde, qui agit comme la pompe du système sanguin. Il permet la circulation du sang dans le corps et l'apport. d'oxygène et de nutriments à l'ensemble des cellules des organes. Le cœur est situé dans la partie médiane de la cage thoracique délimitée par les 2 poumons, le sternum et la colonne vertébrale [18].

- **Dimensions et volume :**
  - **Longueur :**14 à 16 cm
  - **Diamètre :**12 à 14 cm chez l'adulte
  - **Volume :**50 à 60 cm^3
  - **Capacité de pompage :** en moyenne 8000 litres de sang par jour
- **Cavités du cœur :**Le cœur est composé de 4 cavités :
  - **Les oreillettes situées dans la partie supérieure**
  - **Les ventricules situés dans la partie inférieure**

Les oreillettes et les ventricules sont séparés de chaque côté par une épaisse paroi musculaire, le septum. Ainsi, aucun échange de sang entre la partie supérieure et la partie inférieure n'est pas possible. Le passage est unidirectionnel entre oreillette vers ventricule, et cela à travers les valves cardiaques[18].La figure 1.1 illustre une vue générale du coeur.

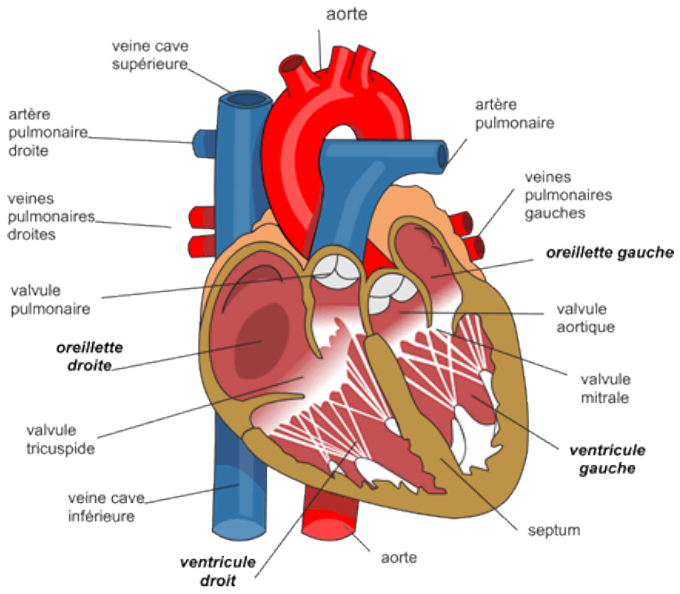


FIGURE 1.1 – Vue générale du coeur [1]

1.3 Fonctionnement du coeur

Le cœur est un organe essentiel pour la circulation du sang dans tout le corps, dont le fonctionnement repose sur un système complexe de chambres et de valves qui permettent au sang de circuler de façon efficace et organisée. Voici les étapes de fonctionnement du coeur[18].

1. Le côté droit du cœur renvoie le sang pauvre en oxygène aux poumons pour éliminer le dioxyde de carbone et ré oxygéner le sang.
2. L'oreillette droite reçoit le sang veineux apporté par la veine cave et propulsé dans le ventricule droit qui en se contractant envoie le sang dans les poumons à travers l'artère pulmonaire (qui est donc la seule artère transportant du sang pauvre en oxygène).
3. Le sang oxygéné dans les poumons revient alors au cœur gauche au niveau de l'oreillette à travers les 4 veines pulmonaires (ce sont les seules veines transportant du sang riche en oxygène).
4. Le sang est ensuite propulsé dans le ventricule gauche et doit traverser la valve mitrale, qui contrôle le débit.
5. En se contractant, le cœur propulse à travers la valve aortique puis l'aorte (plus gros vaisseau sanguin de l'organisme) le sang dans l'ensemble du réseau.

1.4 Les maladies cardiovasculaires :

Une maladie cardiovasculaire est une affection ou une lésion affectant le cœur, les vaisseaux sanguins approvisionnant le cœur ou le réseau de vaisseaux sanguins (c'est-à-dire les artères, les veines ou les petits vaisseaux sanguins) dans tout le corps et à l'intérieur du cerveau. Ces maladies peuvent entraîner de graves complications telles que l'infarctus du myocarde (crise cardiaque), l'accident vasculaire cérébral ou encore l'insuffisance cardiaque, pouvant évoluer vers le décès du patient[3][19].Selon l'organisation mondiale de la santé (OMS),les maladies cardiovasculaires sont la première cause de mortalité dans le monde, représentant environ 17,9 millions de morts chaque année, soit 31% de la mortalité mondiale totale[13]. Parmi ces décès, on estime que 7,4 millions sont dus à une cardiopathie coronarienne et 6,7 millions à un AVC (chiffres 2015).

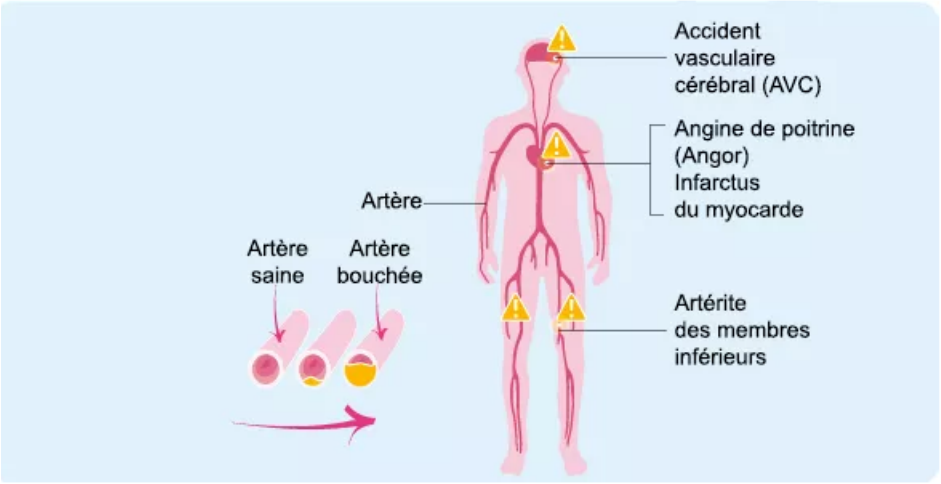


FIGURE 1.2 – Les maladies cardiovasculaires dans le corps [2]

1.5 Les types et les symptômes des maladies cardiovasculaires

Le tableau 1.1 illustre les principaux maladies cardiovasculaires et leur symptômes.

Type de maladie cardiovasculaires	**Description**	**Symptômes**
Maladie coronarienne	Destruction ou infection du sang principal navires. [20]	Douleur ou sensation de lourdeur dans la poitrine, douleur dans le bras, le cou ou la mâchoire, essoufflement, transpiration et nausées.[21]
Cardiomyopathie	Maladie génétique ou acquise affectant les muscles cardiaques. [20]	Respiration difficile pendant l'effort ou pendant les activités , étourdissements ou fatigue , douleur dans la poitrine (angine) anomalies cardiaques palpitations cardiaques (battements rapides ou irréguliers du coeur) gonflement aux mains, aux jambes.[21]
cérébro-vasculaires (AVC)	Elles affectent les vaisseaux sanguins qui alimentent le cerveau.[19]	Une faiblesse d'un seul côté de votre corps,un engourdissement ou un fourmillement au niveau du visage dans les bras ou dans les jambes,une difficulté à parler ou à comprendre ce que disent les autres des troubles de la vue.[22]
Insuffisance cardiaque	Un trouble à long terme lorsque le cœur est incapable de pomper e sang aussi vite qu'il le devrait.[20]	Manque d'énergie chronique , difficulté à vous endormir à cause de problèmes de congestion respiratoire , confusion et/ou troubles de la mémoire et besoin d'uriner la nuit plus fréquemment.[23]
Cardiopathie hypertensive	Hypertension artérielle et problèmes cardiaques[20]	Les céphalées, la fatigue, la nausée, les vomissements, l'essoufflement.[23]
Cardiopathie inflammatoire	Maladie cardiaque ou problèmes qui en découlent, infections bactériennes ou virales[20]	Une douleur au niveau de la poitrine , une arythmie c'est-à-dire une irrégularité du rythme cardiaque et fatigue.[24]

TABLE 1.1 – Types et symptômes de maladies cardiaques

1.6 Les Facteurs de risque

La majorité de ces problèmes cardiovasculaires découlent d'une maladie chronique des artères, connue sous le nom d'athérosclérose, qui se caractérise par le dépôt progressif de graisses et leur calcification [17]. Il existe deux types de facteurs qui contribuent à cette affection :

1.6.1 Facteurs de risque non modifiables :

Il s'agit d'éléments sur lesquels une personne n'a aucune influence ou qu'elle ne peut pas modifier [25].

— **Âge :**La probabilité d'avoir un accident cardiovasculaire ou cardiaque augmente nettement après 50 ans chez l'homme et après 60 ans chez la femme [3].

— **Sexe :**Les femmes, jusqu'à la ménopause, sont plus protégées que les hommes face aux maladies cardiovasculaires. En effet, les hormones (estrogènes et progestérone) les protègent. Mais après 60 ans, une femme a la même probabilité qu'un homme de développer une maladie cardiovasculaire[3].

— **Hérédité :**liée à la transmission génétique de facteurs de risque modifiables, tels que l'hypercholestérolémie familiale, l'hypertension artérielle et le diabète, et autres[26].

1.6.2 Facteurs de risque modifiables :

Il s'agit des facteurs de risque sur lesquels une personne peut agir pour réduire son risque de développer une maladie cardiovasculaire [25].La figure**1.3** illustre quelques facteurs de risque modifiables.

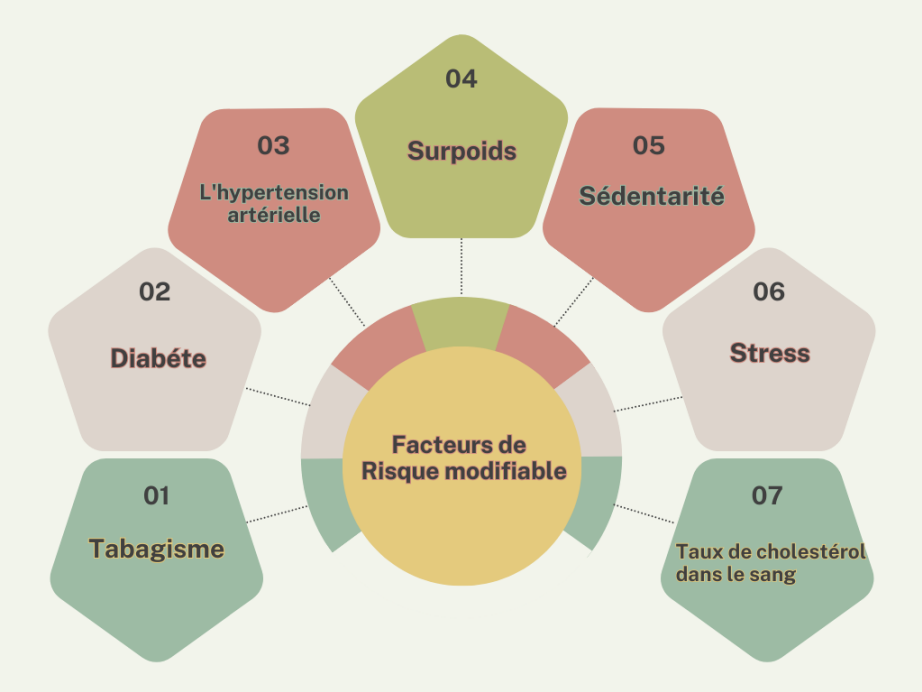


FIGURE 1.3 – Facteurs de risque modifiables [3]

1.7 Prévention des maladies cardiovasculaires

Il est essentiel de prévenir les maladies cardiovasculaires pour maintenir une bonne santé et limiter les risques de complications graves, comme les crises cardiaques et les accidents vasculaires cérébraux (AVC). Dans cette perspective, il existe deux types de prévention : la prévention primaire et la prévention secondaire.

1.7.1 La prévention primaire :

Vise à réduire les risques avant l'émergence de la maladie cardiovasculaire[27]. Les stratégies de préventions primaire comprennent :

- Faire de l'activité physique.
- Adopter une alimentation saine.
- Contrôler le poids.
- Limiter la consommation d'alcool.
- Surveiller la santé en contrôlant régulièrement la tension artérielle, le taux de cholestérol, etc.
- Réduire le stress.

1.7.2 La prévention secondaire :

La prévention secondaire est mise en œuvre suite à un accident vasculaire ou en présence de lésions vasculaires avérées, dans le but de minimiser le risque de récurrence et de ralentir l'évolution des lésions[27]. Les stratégies de préventions secondaires comprennent :

- Arrêter de fumer.
- Contrôle de la pression artérielle.
- Contrôle de la glycémie.
- Traitement médicamenteux pour contrôler les facteurs de risque.

1.8 Conclusion

Dans ce premier chapitre, nous avons d'abord décrit la structure et le fonctionnement du cœur. Ensuite, nous avons présenté un aperçu des maladies cardiovasculaires, en soulignant leurs types et leurs symptômes, puis en détaillant les principaux facteurs de risque. Enfin, nous avons abordé la prévention des maladies cardiovasculaires.

Dans le second chapitre, nous aborderons l'état des connaissances sur l'apprentissage automatique et l'apprentissage profond.

Chapitre 2

Apprentissage automatique et Sélection des attributs

2.1 Introduction :

L'apprentissage automatique (AA) transforme le domaine médical en améliorant le diagnostic, le traitement et la gestion des patients grâce à des algorithmes capables de détecter des schémas complexes dans de vastes ensembles de données. Les modèles d'AA surpassent souvent les performances humaines en interprétant des imageries médicales et en facilitant la médecine de précision avec des traitements personnalisés basés sur les données génomiques et cliniques. Ces avancées améliorent les soins aux patients tout en optimisant les ressources médicales et en réduisant les coûts. La sélection d'attributs est cruciale pour optimiser les modèles d'apprentissage automatique, car elle permet de concentrer les algorithmes sur les variables les plus pertinentes, améliorant ainsi leur performance et leur efficacité.

2.2 Apprentissage automatique

L'apprentissage automatique donne la capacité à un système informatique de perfectionner ses performances sur une tâche spécifique en s'appuyant sur l'expérience, ce qui implique l'analyse de vastes ensembles de données. Les domaines d'application courants comprennent la reconnaissance d'images, la recommandation de produits, la détection de fraudes et les diagnostics médicaux.

2.2.1 Types d'apprentissage automatique

Il existe fondamentalement 4 types d'apprentissage automatique : supervisé, non supervisé, semi-supervisé et renforcé. Les experts en apprentissage automatique estiment qu'environ 70 % des algorithmes d'apprentissage automatique utilisés actuellement sont supervisés[28].

1. **Apprentissage supervisé :**L'apprentissage supervisé est une méthode d'apprentissage automatique dans laquelle un algorithme est entraîné sur un ensemble de données étiquetées, c'est-à-dire

que chaque exemple d'entraînement est associé à une réponse ou une sortie correcte. L'objectif est que l'algorithme apprenne à mapper les entrées aux sorties correspondantes en général, afin de pouvoir prédire les réponses correctes pour de nouvelles données non vues[29][30].

2. **Apprentissage non supervisé :** L'apprentissage non supervisé est une méthode d'apprentissage automatique où l'algorithme est entraîné sur des données non étiquetées, c'est-à-dire sans réponses ou sorties correctes fournies. L'objectif est d'identifier des structures ou des motifs cachés dans les données. Les algorithmes d'apprentissage non supervisé regroupent généralement les données en clusters ou réduisent la dimensionnalité des données pour mieux les comprendre et les analyser[29][30].
3. **Apprentissage semi-supervisé :** L'apprentissage semi-supervisé est une méthode d'apprentissage automatique qui utilise à la fois des données étiquetées et non étiquetées pour entraîner un modèle. Cette approche combine les avantages de l'apprentissage supervisé, qui nécessite des données étiquetées, et de l'apprentissage non supervisé, qui utilise des données non étiquetées. L'objectif est d'améliorer la précision et la performance du modèle en exploitant une grande quantité de données non étiquetées, tout en bénéficiant des informations précises fournies par les données étiquetées. Cela est particulièrement utile lorsque l'étiquetage des données est coûteux ou difficile, permettant ainsi de maximiser l'utilisation des données disponibles pour créer des modèles plus robustes et performants[28].
4. **Apprentissage renforcé :**L'apprentissage par renforcement est une méthode d'apprentissage automatique dans laquelle un agent apprend à prendre des décisions en interagissant avec un environnement. L'agent reçoit des récompenses ou des punitions en fonction des actions qu'il effectue, et son objectif est de maximiser la somme des récompenses sur le long terme. Contrairement à l'apprentissage supervisé, il n'y a pas de réponses correctes fournies pour chaque action ; au lieu de cela, l'agent découvre progressivement la meilleure stratégie, ou politique, en explorant différentes actions et en évaluant leurs conséquences. Cette méthode est particulièrement efficace pour des problèmes complexes où les décisions doivent être prises en séquence, comme les jeux, la robotique ou la gestion de ressources[28].

2.2.2 Les algorithmes apprentissage automatique :

Il existe différents types d'algorithmes d'apprentissage automatique qui visent tous à résoudre différents problèmes. Dans cette section, nous expliquons certains de ces algorithmes.

1- k-Nearest Neighbors :

L'algorithme K-Nearest Neighbors (KNN) est une méthode couramment employée dans le domaine de l'apprentissage automatique pour les opérations de classification et de régression. Il s'appuie sur l'idée que des points de données similaires ont tendance à présenter des étiquettes ou des valeurs similaires[31]. Afin de prédire la classe d'une nouvelle donnée d'entrée, il recherchera les K voisins les plus proches (en utilisant la distance euclidienne ou autres) et sélectionnera la classe des voisins majoritaires.[32] :

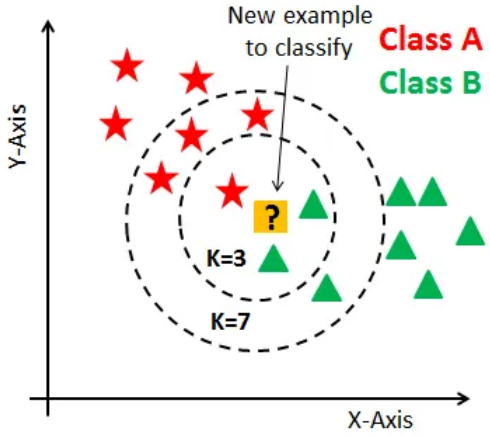


FIGURE 2.1 – Illustration du K plus proches voisins pour la classification [4]

Les étapes de l'algorithme KNN sont comme suit :

- **Choix du nombre k :**Sélectionner le nombre de voisins les plus proches (k) à utiliser dans la prédiction.
- **Calcul des distances :**Effectuez le calcul de la distance entre le point que vous voulez classer et tous les autres points du jeu de données. Il existe différentes mesures de distance qui peuvent être utilisées :
  - **Distance de Manhattan :**Est le total des valeurs absolues des écarts entre les coordonnées des deux points correspondants.

$$\text{distance}(P_1, P_2) = \sum_{i=1}^{n} |x_{2i} - x_{1i}| \tag{2.1}$$

  - **Distance euclidienne :**Les coordonnées correspondantes des deux points sont représentées par la racine carrée de la somme des carrés des différences entre eux.

$$\text{distance}(P_1, P_2) = \sqrt{\sum_{i=1}^{n} (x_{2i} - x_{1i})^2} \tag{2.2}$$

- **Détermination des voisins les plus proches en fonction de la distance calculée.**
- **Comptage des catégories :**Parmi ces k voisins, comptez le nombre de points appartenant à chaque catégorie.
- **Attribution de la catégorie :**Classez le nouveau point dans la catégorie la plus répandue parmi ces k voisins.

2- Support Vector Machine :

Machine à vecteurs de support (SVM) est un algorithme d'apprentissage supervisé utilisé principalement pour les tâches de classification et de régression. Il fonctionne en trouvant l'hyperplan optimal qui

sépare les données en différentes classes de manière à maximiser la marge entre les points de données les plus proches de l'hyperplan, appelés vecteurs de support, et l'hyperplan lui-même. Pour les problèmes non linéaires, SVM utilise des fonctions de noyau (kernels) pour transformer les données d'entrée dans un espace de dimension supérieure où un hyperplan linéaire peut être utilisé pour la séparation. L'objectif principal de SVM est de maximiser la marge de séparation, ce qui contribue à une meilleure généralisation du modèle sur des données non vues.[33]

La définition des concepts de base de l'algorithme SVM est comme suit :

- **Hyperplane :** Un hyperplan est une frontière de décision utilisée par le classificateur SVM pour séparer un ensemble de points de données ayant des étiquettes de classe différentes. L'objectif est de trouver un hyperplan qui maximise la marge entre les différentes classes. Cet hyperplan, appelé hyperplan de marge maximale, est déterminé par le SVM et définit un classificateur linéaire efficace pour cette séparation[5].En mathématiques, on peut définir un hyperplan dans un espace à n dimensions par une équation de la forme :

  $$w \cdot x + b = 0[34]$$

  Où :

  — Le vecteur **w** représente le vecteur normal à l'hyperplan. c'est-à-dire la direction perpendiculaire à l'hyperplan.

  — Le paramètre **b** st un paramètre constant qui influence la position de l'hyperplan le long de sa direction normale, déterminée par le vecteur **w**.

  — Le point **x** se trouve dans l'espace n-dimensionnel.

- **Vecteurs de support :**Les vecteurs de support sont les points de données les plus proches de l'hyperplan. Ces points de données sont essentiels pour calculer les marges et définir précisément la ligne de séparation ou l'hyperplan. [5].

- **Marge :**Une marge est l'espace de séparation entre l'hyperplan et les points de données les plus proches. Elle est calculée comme la distance perpendiculaire à l'hyperplan et englobe les vecteurs de support ou les points de données les plus proches. Dans le cadre du SVM, il est essentiel de maximiser cette marge de séparation afin d'atteindre une marge maximum.[5].

La figure 2.2 illustre visuellement ces concepts :

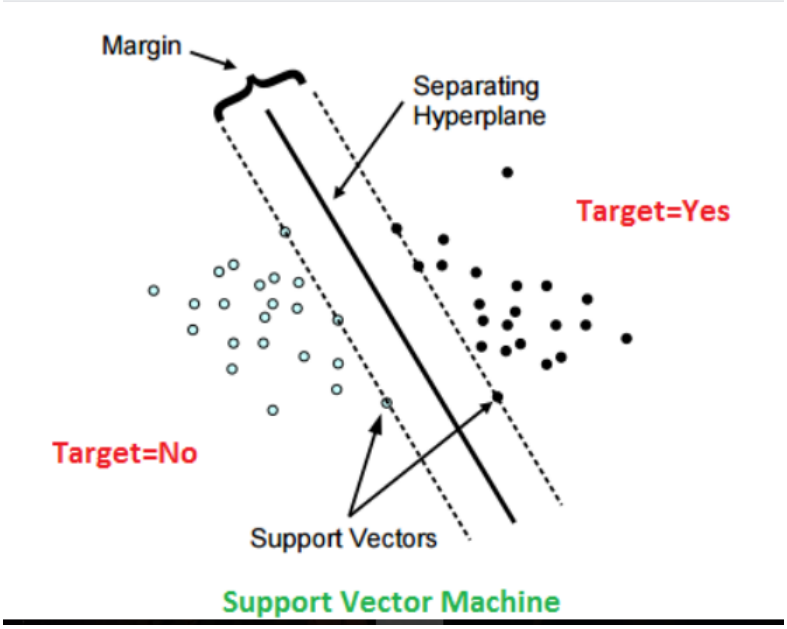


FIGURE 2.2 – Illustration de la classification binaire avec un SVM[5]

4- Random Forest :

Les approches de forêt aléatoire sont utilisées à la fois dans l'analyse de classification et dans l'analyse de régression. Les prévisions sont effectuées à l'aide d'une représentation arborescente des données. Lorsqu'elle est utilisée sur de grands ensembles de données, la technique de la forêt aléatoire peut encore fournir les mêmes résultats même lorsqu'une partie importante des entrées de données est manquante. L'analyse et la recherche effectuées par l'arbre de décision peuvent être enregistrées et appliquées à divers ensembles de données. Une forêt aléatoire se compose de deux étapes : la première étape est la génération de la forêt aléatoire, et la seconde est la projection qui utilise la classification créée lors de la première étape[35].

5- Decision Tree :

Un arbre de décision est une structure hiérarchique robuste et populaire utilisée pour la classification et la prédiction. Il est composé de différents types de nœuds et de branches. Voici la structure générale d'un arbre de décision [36] :

- **Nœud racine :**Le point de départ de l'arbre.
- **Nœuds internes :**Les points de décision intermédiaires entre le nœud racine et les nœuds feuilles.
- **Nœuds feuilles :**Les points terminaux qui représentent la classification finale ou la prédiction.
- **Branches :**Les connexions entre les nœuds, représentant les résultats des tests effectués à chaque nœud.

Pour chaque nœud dans un arbre de décision, les critères de division sont utilisés pour déterminer comment séparer les données en fonction des différents attributs. Les principaux critères sont[37] :

— Gain d'information

— Indice de gini (Gini Index)

— Ratio de gain (Gain Ratio)

6- XGBoost

XGBoost (eXtreme Gradient Boosting) est en effet un algorithme très performant pour les données structurées et tabulaires, et il est largement utilisé en apprentissage automatique supervisé pour des tâches de classification et de régression. Grâce à sa capacité à gérer efficacement les jeux de données volumineux et complexes, combinée à des techniques avancées de régularisation et de calcul parallèle, XGBoost s'est imposé comme une référence dans le domaine.

Les scientifiques de données apprécient XGBoost pour sa capacité à exécuter des calculs à grande vitesse, même avec des ensembles de données volumineux nécessitant des calculs hors mémoire. En combinant plusieurs arbres de décision, chaque nouvel arbre corrige les erreurs des précédents, ce qui améliore continuellement les prédictions[38]. XGBoost utilise les méthodes et les paramètres suivants pour optimiser l'algorithme et améliorer les résultats et les performances : régularisation ,élagage,construction de quantile pondéré,entraînement parallèle,recherche de fractionnement tenant compte de la rareté,blocs de calcul out-of-core [39]

7- Réseaux Bayésiens :

Un réseau bayésien est un modèle graphique probabiliste qui représente un ensemble de variables aléatoires et leurs dépendances conditionnelles via un graphe orienté acyclique (DAG). En général, un BN peut être utilisé pour calculer la probabilité conditionnelle d'un nœud, étant donné les valeurs attribuées aux autres nœud [40].La figure 2.3 illustre un exemple de réseau bayésien.

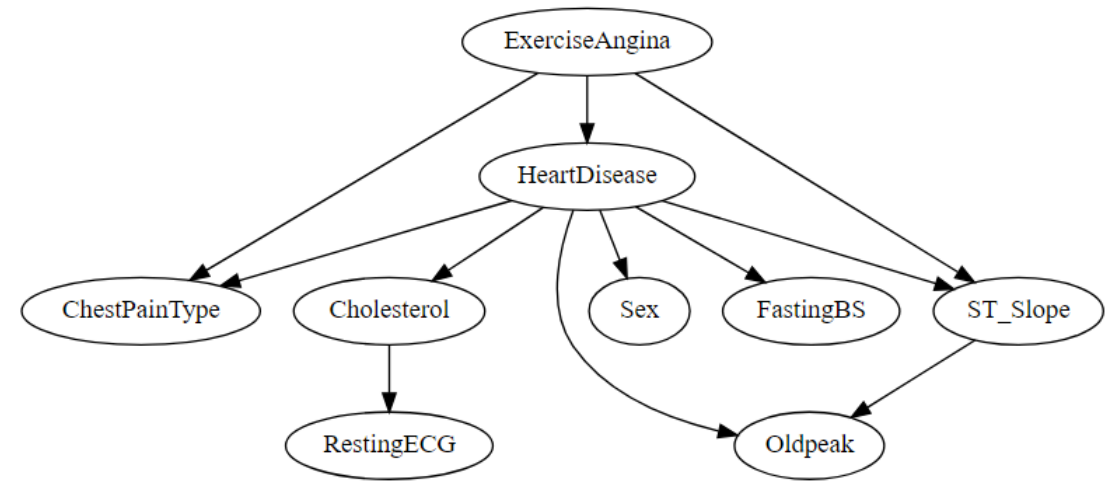


FIGURE 2.3 – Exemple de réseau bayésien

1. **Les principaux éléments d'un réseau bayésien sont les suivants :**
   - **Nœuds :**les nœuds représentent des variables aléatoires.
   - **Arcs :**Les arcs représentent des liens de dépendance conditionnelle entre les variables spécifiques.
   - **Tables de probabilité conditionnelle (CPT) :** Pour chaque variable X_i dans un réseau bayésien, une table de probabilité conditionnelle (CPT) détermine comment les états de ses parents $Pa(X_i)$ influencent la probabilité de X_i elle-même. Par exemple, si X_i a des parents $Pa(X_i)$, la CPT $P(X_i \mid Pa(X_i))$ spécifie les probabilités de X_i pour chaque combinaison possible des états de ses parents $Pa(X_i)$.La figure 2.4illustre exemple de CPT :

CPD of Cholesterol:

HeartDisease	HeartDisease(0)	HeartDisease(1)
Cholesterol(0)	0.12345679012345678	0.34413965087281795
Cholesterol(1)	0.3271604938271605	0.18453865336658354
Cholesterol(2)	0.2932098765432099	0.2169576059850374
Cholesterol(3)	0.25617283950617287	0.2543640897755611

FIGURE 2.4 – Exemple de CPT

2. **Construction d'un réseau bayésien :** Pour construire les réseaux bayésiens, il existe plusieurs méthodes et algorithmes. On peut définir directement les relations, par exemple à partir d'une matrice de corrélation ou utiliser des algorithmes dédiés. L'algorithme le plus utilisé est HillClimbSearch. Voici une explication des étapes de cet algorithme :
   - **Initialisation :**Démarre avec une configuration de réseau initialisée de manière aléatoire ou préétablie.
   - **Évaluation :**Évalue la qualité de la structure actuelle en utilisant des critères comme le BIC pour mesurer son adéquation aux données.
   - **Génération de voisins :** Crée de nouvelles structures en effectuant de légères modifications à la structure existante, telles que l'ajout ou le retrait de liens entre les variables.
   - **Sélection :**Choisi la structure voisine qui améliore le score de manière la plus significative.
   - **Répétition :** Répète les étapes d'évaluation, de génération de voisins et de sélection jusqu'à ce qu'aucune amélioration supplémentaire ne soit possible.

2.3 L'apprentissage profond

L'apprentissage profond (deep learning) est une sous-discipline de l'apprentissage automatique qui utilise des réseaux de neurones artificiels composés de multiples couches pour modéliser et comprendre des représentations complexes des données. Ces réseaux, appelés réseaux de neurones profonds, sont capables d'apprendre automatiquement des caractéristiques hiérarchiques des données en passant des couches d'entrée aux couches de sortie. Chaque couche apprend des représentations de plus en plus abstraites,

permettant au modèle de capturer des structures et des motifs complexes. L'apprentissage profond est particulièrement efficace pour les tâches impliquant des données volumineuses et non structurées, telles que la reconnaissance d'images, le traitement du langage naturel et l'analyse de la voix, grâce à sa capacité à apprendre directement des données brutes.[41]

2.4 Resaux de neurones artificile :

Les réseaux de neurones artificiels (ANN) sont des systèmes de calcul parallèles à grande échelle, composés de nombreuses unités de traitement interconnectées, qui s'inspirent du fonctionnement des neurones biologiques dans le cerveau humain. Danc avant d'aborder les réseaux de neurones artificiels, il est important d'abord comprendre la structure et le fonctionnement des neurones biologiques.

2.4.1 Neurone biologique :

Un neurone, tel qu'illustré dans la figure 2.5, est une cellule particulière qui possède des extensions qui lui permettent de transmettre des signaux (axones) ou de les recevoir (dendrites).

Les neurones sont connectés entre eux dans le cerveau par les axones et les dendrites. Ces filaments agissent comme des conducteurs électriques, transmettant des messages d'un neurone à un autre.

Un neurone émet un signal en réponse aux signaux reçus des autres neurones. Cela implique une somme des signaux reçus au fil du temps. Lorsque cette somme dépasse un certain seuil, le neurone génère un signal électrique.

La synapse est la jonction entre un axone et une dendrite où la transmission des signaux se fait par des substances chimiques appelées neurotransmetteurs. Lorsque le signal atteint la synapse, il déclenche la libération de neurotransmetteurs qui traversent l'espace synaptique pour se fixer sur des récepteurs de l'autre côté[42].

- Les principaux composants des neurones biologiques :

— **Corps cellulaire(cell body) :**La cellule neuronale est soutenue et structurée par le corps cellulaire ou soma, qui recueillit et traite les informations provenant d'autres neurones.

— **Axon (axone) :**L'axon, qui s'étend loin du corps cellulaire, constitue le trajet le long duquel les informations se déplacent.

— **Dendrite :**Les dendrites sont des extensions tubulaires ramifiées autour du soma, sont les voies principales par lesquelles les neurones reçoivent les informations.

— **Le point de jonction :**La synapse est le point de jonction entre l'axon d'un neurone et une dendrite d'un autre neurone, où les informations sont transmises via des neurotransmetteurs libérés lors de l'impulsion nerveuse.

— **Noyau :** qui abrite le matériel génétique (ADN), contrôle les activités cellulaires[7].

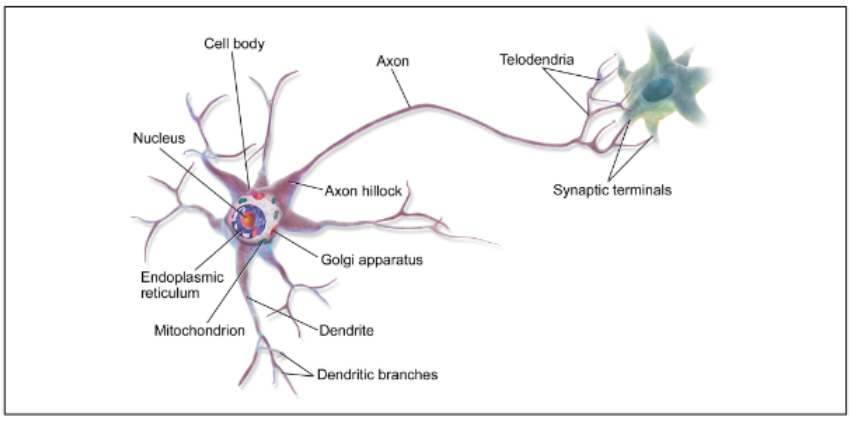


FIGURE 2.5 – Neurone biologique [6]

2.4.2 Neurone formel :

Un neurone formel est une représentation mathématique et informatique d'un neurone biologique,possède généralement plusieurs entrées et une sortie comme illustré dans la figure**2.6** [43].

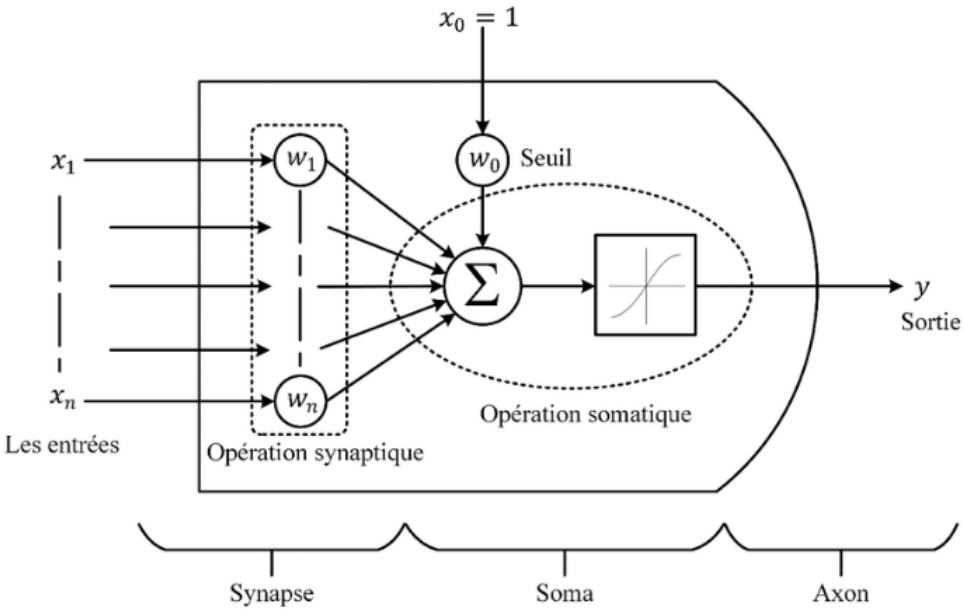


FIGURE 2.6 – Neurone formel [7]

- **Structure et analogies :**

Élément	**Description**
Entrées	Notées **x** sous forme de vecteur, représentant les dendrites [42]
Sortie	Notée y, représentant le cône d'émergence du neurone biologique (le point de départ de l'axone)[42]
Actions des synapses	**Coefficients numériques** : Les actions excitatrices et inhibitrices des synapses sont représentées par des coefficients numériques, appelés poids synaptiques, associés aux entrées[43]. **Phase d'apprentissage** : Les valeurs de ces coefficients sont ajustées lors d'une phase d'apprentissage[43].

TABLE 2.1 – Structure et analogies du neurone formel

- **Fonctionnement de base :**

  1. **Somme pondérée** : Dans sa version la plus simple, un neurone formel calcule la somme pondérée des entrées reçues, comme dans l'équation suivante :

  $$\text{Sortie} = \sum(x_i \cdot w_i) + b$$

  où b représente le biais. L'entrée b prend souvent les valeurs -1 ou +1, permettant d'ajouter de la flexibilité au réseau en faisant varier le seuil de déclenchement du neurone par l'ajustement des poids et du biais lors de l'apprentissage.

  2. **Fonction d'activation** : Représente le noyau ou soma dans le neurone biologique. Différentes fonctions de transfert peuvent être utilisées comme fonction d'activation du neurone.Les fonctions d'activation couramment utilisées est :

Sigmoid
$\sigma(x) = \frac{1}{1+e^{-x}}$

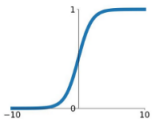


FIGURE 2.7 – Sigmoid

tanh
$\tanh(x)$

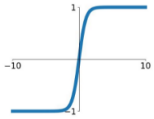


FIGURE 2.8 – Tanh

ReLU
$\max(0, x)$

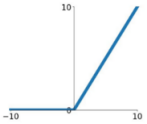


FIGURE 2.9 – ReLU

  3. **Sortie** : La valeur finale obtenue après l'application de la fonction d'activation constitue la sortie du neurone.

2.4.3 L'apprentissage des réseaux de neurones

1. **Propagation vers l'avant :** La propagation vers l'avant (forward propagation) dans un réseau de neurones correspond au processus par lequel les données d'entrée traversent les différentes couches du réseau, de l'entrée jusqu'à la sortie. À chaque étape, les données sont transformées par des opérations linéaires (multiplication par des poids) et non linéaires (application de fonctions d'activation), générant des représentations de plus en plus abstraites et discriminantes. Ce processus permet au réseau d'acquérir la capacité de coder les caractéristiques essentielles des données d'entrée pour

maximiser sa capacité de discrimination. Le calcul de la fonction de perte à la fin de la propagation vers l'avant permet d'évaluer les performances du modèle[44].

2. **Propagation vers l'arrière :** La propagation vers l'arrière (backpropagation) est la méthode par laquelle un réseau de neurones ajuste ses poids en fonction de l'erreur entre la sortie prévue et la sortie désirée. Après avoir calculé la fonction de perte lors de la propagation vers l'avant, la backpropagation est utilisée pour propager les gradients en sens inverse, de la sortie vers l'entrée du réseau. Ces gradients sont ensuite utilisés pour mettre à jour les poids en utilisant des techniques d'optimisation telles que la descente de gradient. Ce processus contrôle l'entraînement du réseau en fournissant des mises à jour correctives basées sur les erreurs observées[44].

2.4.4 Type de resaux de neurones artificile :

Il existe différents types de réseaux de neurones artificiels.Parmi eux, on peut citer :

1-Les réseaux neurones profonds(DNN) :

Les réseaux neurones profonds sont des réseaux de neurones avec un certain niveau de complexité. Ils se composent de plusieurs couches empilées, comprenant généralement une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Les DNN sont souvent utilisés pour traiter des données non étiquetées en effectuant deux étapes principales : la propagation vers l'avant pour traiter les données à travers le réseau et la rétropropagation pour ajuster les poids des connexions et améliorer les prédictions[45] ,comme illustré dans la figure 2.10 :

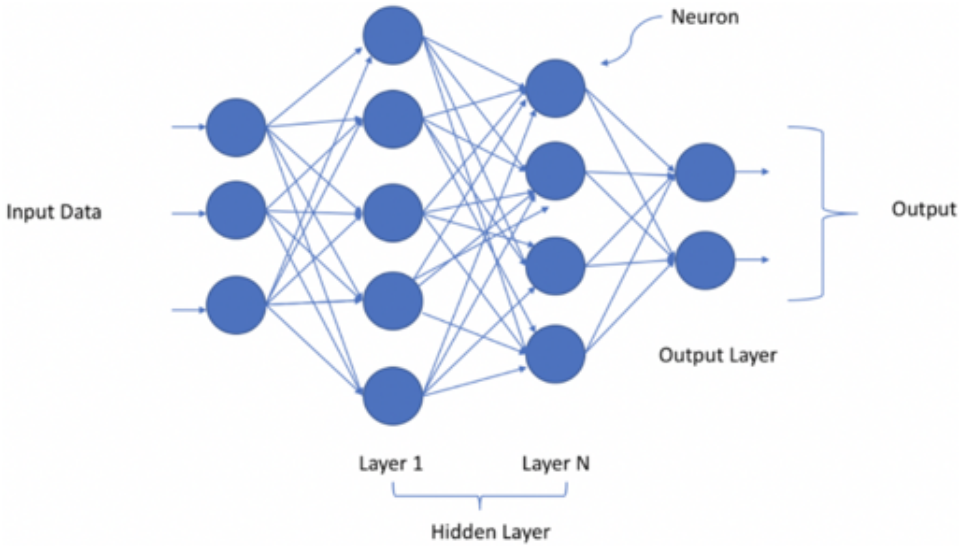


FIGURE 2.10 – L'architecture de DNN[8]

2- Perceptron multicouche(MLP) :

Le perceptron multicouche (MLP) est un type spécifique de réseau de neurones artificiels et peut être considéré comme une forme de DNN de base. Il se compose également d'une couche d'entrée, de couches cachées et d'une couche de sortie. La principale différence entre un MLP et d'autres formes de DNN plus complexes réside dans la profondeur et la complexité du réseau. Les MLP ont généralement moins de couches cachées, ce qui les rend moins complexes et plus adaptés aux tâches d'apprentissage automatique plus simples[46].

3-Les réseaux de neurones convolutifs(CNN) :

Les réseaux de neurones convolutifs sont spécifiquement destinés aux tâches de reconnaissance d'images et de vidéos. Les CNN sont capables d'apprendre automatiquement les caractéristiques des images, ce qui les rend bien adaptés à des tâches telles que la classification d'images, la détection d'objets et la segmentation d'images[9].

4-Les réseaux de neurones récurrents (RNN) :

Les réseaux de neurones récurrents (RNN) sont des modèles spécialisés dans le traitement de données séquentielles comme des séries chronologiques ou du langage naturel. Ils se distinguent par leur capacité à conserver un état interne qui capture des informations sur les entrées précédentes, ce qui les rend efficaces pour des tâches telles que la reconnaissance vocale, le traitement du langage naturel et la traduction linguistique[9]. Les principales architectures de réseaux de neurones récurrents incluent :

1. **RNN basique(simple) :**C'est la version la plus basique de RNN, où chaque neurone est relié récurremment à lui-même, mais ils rencontrent souvent des difficultés avec le "vanishing gradient" lorsqu'ils apprennent sur des séquences longues,la figure2.11 illustre l'architecture d'un RNN simple.

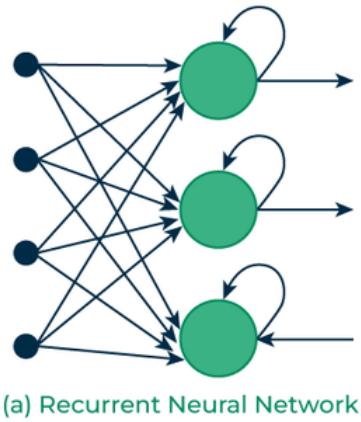


FIGURE 2.11 – L'architecture de RNN simple[9]

2. **GRU :** Le GRU (Gated Recurrent Unit) est une méthode évoluée dérivée du RNN de base. Il utilise deux portes principales pour réguler le flux d'informations : la porte de réinitialisation et

la porte de mise à jour. Ces portes sont représentées sous forme de rectangles avec leurs propres fonctions d'activation. La porte de réinitialisation du GRU détermine si les nouvelles informations doivent être oubliées ou non, tandis que la porte de mise à jour contrôle la mise à jour de la mémoire interne[47].

3. **LSTM :** Les LSTM est constitué d'une série de cellules de mémoire qui peuvent stocker et actualiser des informations sur de longues séquences. Chaque cellule possède trois portes : une porte d'entrée pour décider quelles informations ajouter à l'état actuel, une porte de sortie pour décider quelles informations utiliser pour la sortie, et une porte d'oubli pour décider quelles informations supprimer de l'état de la cellule. Ces portes sont apprises par le réseau en fonction des données d'entrée et de l'état précédent de la cellule[48].

2.5 Sélection des attributs

La sélection des attributs est un processus de recherche ou une technique utilisée pour trouver un sous-ensemble "pertinent" d'attributs parmi ceux de l'ensemble de départ [49], c'est-à-dire les mesures les plus intéressantes. L'objectif de cette méthode est d'améliorer les performances des modèles prédictifs en supprimant les attributs redondants ou non pertinents qui peuvent causer du bruit ou rendre le modèle plus complexe. Un attribut est classé comme suit [49] :

— **Très pertinente :**Une attribut est considérée comme très pertinente si son absence entraîne une détérioration significative de la performance du système de classification utilisé.

— **Peu pertinente :** Une attribut est dite peu pertinente si, bien qu'elle ne soit pas très pertinente, l'ajout de cette caractéristique à un sous-ensemble des attributs améliore significativement la performance du modèle.

— **Non pertinente :**Les attributs qui ne sont ni "peu pertinentes" ni "très pertinentes" sont considérées comme non pertinentes. En général, ces attributs seront supprimées de l'ensemble des attributs de départ.

2.5.1 Importance de la sélection des attributs

Il est essentiel de choisir les attributs lors de la modélisation des données, en particulier dans le domaine de l'apprentissage automatique et de l'analyse de données. La méthode implique de sélectionner un groupe pertinent d'attributs parmi l'ensemble initial, dans le but d'améliorer les performances et l'interprétabilité des modèles. Voici les principales raisons pour lesquelles il est crucial de choisir les attributs [15] :

1. Amélioration de la performance du modèle.
2. Faciliter l'interprétation des données.
3. Réduire le temps de calcul.
4. Réduction de la complexité.

2.5.2 Processus de sélection des attributs

Il existe quatre étapes de sélection des attributs : une procédure de génération, une fonction d'évaluation, un critère d'arrêt, et une procédure de validation. Ces étapes sont représentées par la figure**(2.12)**[50].

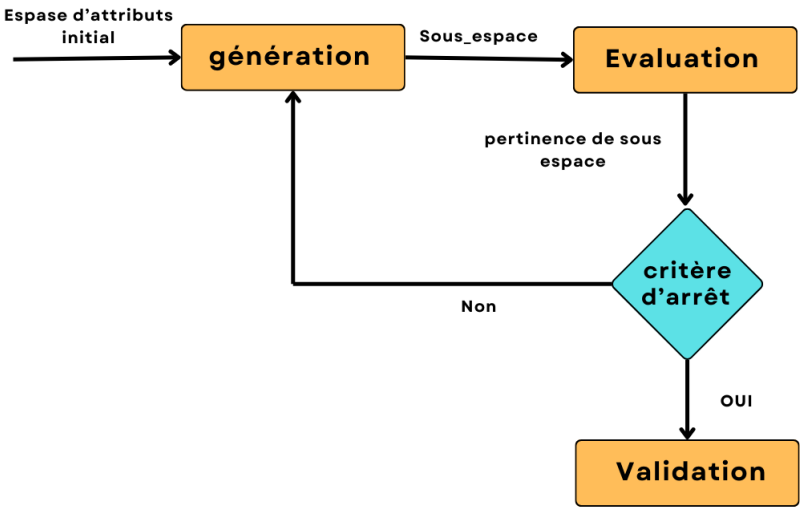


FIGURE 2.12 – Processus de sélection des attributs

- **Génération :** La génération est une méthode qui utilise l'espace de recherche pour élaborer diverses combinaisons de caractéristiques[51].Un sous-ensemble d'attributs est créé à chaque itération afin d'être évalué.
- **Évaluation :**La pertinence des sous-ensembles générés est mesurée par une fonction d'évaluation. Chaque sous-ensemble est comparé au meilleur précédemment trouvé. S'il est meilleur, il remplace le meilleur sous-ensemble retenu. Ce processus se répète jusqu'à satisfaction d'un critère d'arrêt.[52].
- **Le critère d'arrêt :**La sélection des attributs se poursuit jusqu'à l'atteinte d'un critère d'arrêt. La méthode de génération et la fonction d'évaluation ont un impact sur ce critère, dans le but d'améliorer considérablement les performances du modèle. Les performances du modèle, le nombre d'itérations ou d'autres conditions spécifiques au problème peuvent être incluses.
- **La validation :**Permet de tester la validité des sous-ensembles d'attribut sélectionnés en effectuant plusieurs tests sur des exemples de données générées réelles ou non [50].

2.5.3 Les méthodes de sélection :

Les techniques de sélection des attributs jouent un rôle crucial dans l'amélioration des performances des modèles d'apprentissages automatiques en réduisant la dimensionnalité des données, éliminant les caractéristiques non importantes, et accélérant le processus d'apprentissage[53]. Elles sont regroupées en trois catégories principales comme illustré dans la figure 2.13.

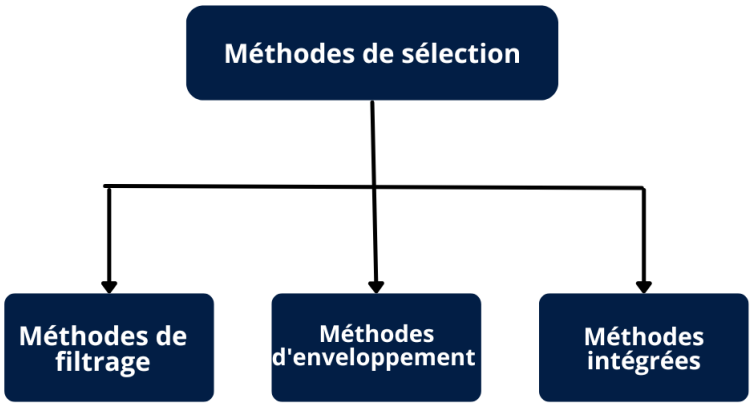


FIGURE 2.13 – Méthodes de sélection

- **Méthodes de filtrage (Filter Methods) :** Les méthodes de filtrage utilisent les propriétés statistiques des entités comme la corrélation, la variance ou l'information mutuelle. Elles classent les caractéristiques selon des critères prédéfinis et sélectionnent celles qui dépassent un seuil ou répondent à un nombre déterminé. Ces méthodes sont rapides, évolutives et indépendantes de l'algorithme d'apprentissage, mais elles ne considèrent pas les interactions entre les attributs ni leur pertinence par rapport à la variable cible[53].

  Les méthodes de filtrage se composent de diverses techniques, comme indiqué ci-dessous :

  1. Sélection de caractéristiques univariées [54]
  2. Gain d'information [54]
  3. Test du Khi-Deux (Chi-Square)[55]
  4. Anova [55]
  5. Matrice de corrélation avec carte thermique [54]

- **Les méthodes d'enveloppement :** Les méthodes d'enveloppement sont basées sur un modèle d'apprentissage sur différents sous-ensembles de l'espace des attributs. Les nouveaux attributs sont systématiquement inclus ou exclus en fonction des résultats comparés au modèle précédent, il est décidé si le nouvel attribut est inclus ou non. Différents modèles doivent être formés et évalués pour identifier l'ensemble optimal d'attributs, ce qui signifie que ces méthodes sont coûteuses en termes de calcul[56]. Les méthodes d'enveloppement les plus courantes sont :

  1. Forward selection [57]
  2. Backward elimination [57]
  3. Exhaustive feature selection [56]

4. Recursive feature elimination [57]

- **Méthodes intégrées :**Les méthodes intégrées combinent les qualités des méthodes de filtrage et des méthodes d'enveloppement. Elles sont implémentées par des algorithmes qui possèdent leurs propres méthodes de sélection des attributs intégrées[58].Parmi les exemples les plus populaires de ces méthodes sont :

  1. Lasso regulation [54]
  2. Random forest [58]
  3. Régression RIDGE [54]

2.5.4 Algorithmes génétiques (AG) pour la sélection des attributs :

Les algorithmes génétiques (AGs) sont des techniques de recherche inspirées par les mécanismes de sélection naturelle et de génétique . Ils utilisent une terminologie qui reflète celle de la génétique naturelle, comprenant des termes comme population, individu, chromosome et mutation. [10]. La figure 2.14 illustre les concepts clés des algorithmes génétiques .

— **Concepts clés des algorithmes génétiques :**

1. **Population :** Ensemble d'individus qui évoluent au cours des générations.
2. **Individu :**Solution candidate au problème, représentée par un ensemble de gènes
3. **Chromosome :**Structure qui contient l'information génétique d'un individu, souvent représentée par une séquence de bits(gène) dans le contexte de la sélection d'attributs.
4. **Gène :**Représente une caractéristique,dans le contexte de la sélection d'attributs, un bit avec un 0 signifie que l'attribut n'est pas sélectionné, tandis qu'un 1 signifie qu'il est sélectionné.
5. **Fonction objective :** Pour chaque génération, une fonction objectif est utilisée pour évaluer chaque solution candidate, également connue sous le nom de phénotype. Cette évaluation a pour but d'identifier les individus les plus performants, c'est-à-dire ceux qui se rapprochent le plus de la solution optimale recherchée.

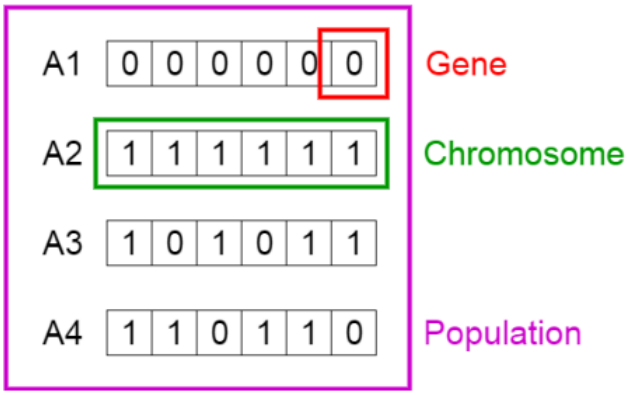


FIGURE 2.14 – Concepts clés des algorithmes génétiques [10]

— **Fonctionnement de AG :** La figure 2.15 illustre le fonctionnement du AG

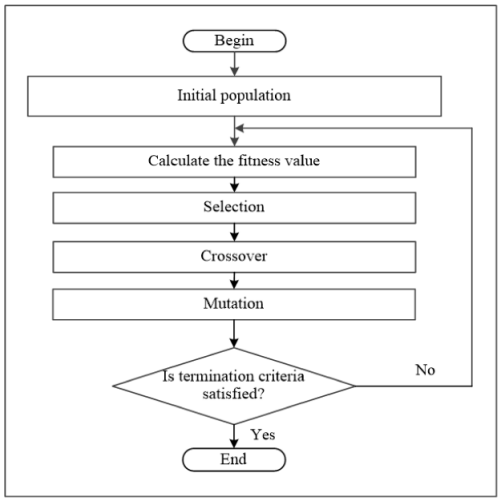


FIGURE 2.15 – Fonctionnement de AG [11]

1. **Génération de la population initiale :**La population initiale est générée aléatoirement en identifiant la taille de la population.
2. **Évaluation :**Pour chaque individu dans la population, on calcule la fonction objectif. Par exemple, dans le cadre de la sélection d'attributs, on entraîne un modèle sur les attributs sélectionnées, puis on évalue les performances de chaque chromosome.
3. **Sélection :** Il s'agit du processus de sélection de la solution la plus adaptée à partir de la population actuelle. Les solutions les plus adaptées servent alors de parents pour la génération suivante. La sélection peut être effectuée à l'aide de méthodes telles que la sélection par roulette, la sélection par tournoi, et d'autres techniques similaires[10].
4. **Croisement :**La recombinaison se produit lorsque les gènes des deux parents les plus aptes sont échangés de manière aléatoire pour former de nouveaux génotypes ou de nouvelles solu-

tions. Le croisement peut être un croisement à un point ou un croisement à plusieurs points, basé sur les segments d'échange de gènes des parents[10].

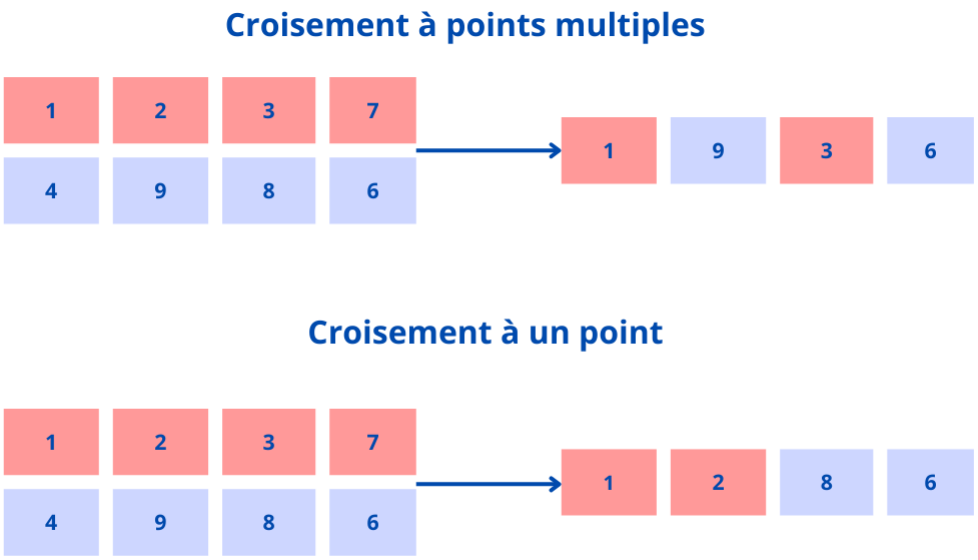


FIGURE 2.16 – Exemple de croisement

5. **Mutation :**Le processus de mutation consiste à modifier aléatoirement un génotype pour promouvoir la diversité de la population et trouver des solutions meilleures et optimisées[10].

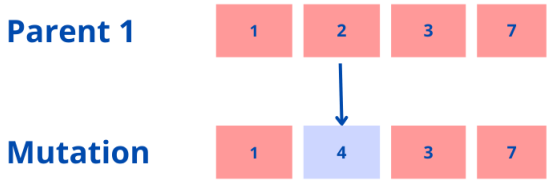


FIGURE 2.17 – Exemple de mutation

6. **Remplacement :**La population actuelle est remplacée par les nouveaux enfants afin de créer une nouvelle génération.
7. **Répétition du processus :**Ce processus est répété jusqu'à ce qu'un critère d'arrêt soit atteint

2.5.5 Algorithme génétique rival pour la sélection d'attributs

Les algorithmes génétiques (AG) sont largement utilisés pour la sélection d'attributs, mais ils souffrent de plusieurs limitations, notamment en termes d'efficacité de la sélection des parents, et de nature aléatoire des opérations de sélection et de mutation. L'Algorithme génétique rival (AGR) introduit des améliorations significatives pour pallier ces limitations, aboutissant à de meilleures performances et à une convergence plus rapide. Voici les principales améliorations apportées par l'AGR :

1. **Sélection de parents de haute qualité :**
   - **Problème :**Dans les algorithmes génétiques (AG), la sélection des parents se fait souvent de manière aléatoire, ce qui peut aboutir à la sélection de chromosomes de faible qualité. Cela risque de produire des solutions moins optimales lors de la reproduction.
   - **Solution AGR :**L'AGR introduit une nouvelle stratégie de compétition qui classe les chromosomes en gagnants et perdants selon leur valeur de fitness. Les parents sont ensuite sélectionnés parmi les gagnants, assurant ainsi une sélection de haute qualité.
2. **Amélioration de la recherche par mutation dynamique :**
   - **Problème :**Dans les algorithmes génétiques , les processus de sélection et de mutation sont généralement aléatoires, ce qui peut entraîner des résultats variés et parfois imprévisibles.
   - **Solution AGR :**L'AGR innove en introduisant un taux de mutation dynamique qui s'ajuste selon la fitness et la génération. Cette méthode cherche à optimiser la stratégie de recherche de l'algorithme, en minimisant les risques de se retrouver dans des solutions locales et en facilitant une exploration plus précise du paysage des solutions.
3. **Croisement :**
   - **Problème :**Le croisement entre deux parents peut parfois ne pas explorer de manière exhaustive l'ensemble de l'espace de recherche, ce qui pourrait restreindre la capacité de l'algorithme à découvrir les solutions optimales disponibles.
   - **Solution AGR :** L' AGR effectue le croisement entre trois parents (deux gagnants et un perdant), ce qui améliore la capacité d'exploration de l'espace de recherche et permet de générer des solutions de haute qualité.

2.5.5.1 Fonctionnement de AGR

1. **La stratégie de compétition :**Une fois la population initiale initialisée, les chromosomes sont choisis de manière aléatoire par paires dans la population afin de participer à la compétition, où deux chromosomes de chaque couple sont en compétition.L' AGR divise les chromosomes en deux groupes (groupe gagnant et groupe perdant). Selon la règle de survie, un chromosome ayant une meilleure fitness est ajouté au groupe gagnant, tandis que le perdant de la compétition est transféré dans le groupe perdant[15].La figure **(2.18)**montre le fonctionnement de la stratégie de compétition.

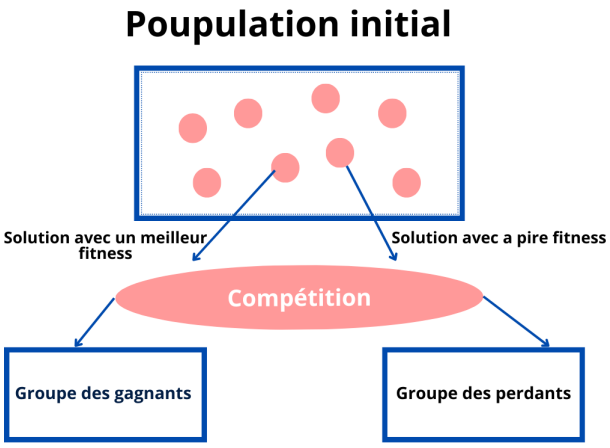


FIGURE 2.18 – La stratégie de compétition

2. **Séléction :** :Dans l'algorithme génétique classique, les parents sont sélectionnés de manière aléatoire dans la population qui utilise la roulette ou le tournoi. Cette méthode a des inconvénients., Par exemple, avec la sélection par roulette, la probabilité qu'un chromosome devienne parent est basée sur sa valeur de fitness, mais cette probabilité est calculée pour tous les chromosomes de la population, ce qui signifie qu'un chromosome faible a également une chance d'être sélectionné.De plus, il y a trop de chromosomes participant au processus de sélection. En conséquence, deux chromosomes faibles peuvent devenir parents, reproduisant ainsi une solution de très mauvaise qualité.L'AGR résout ce problème en sélectionnant les premiers et deuxièmes parents (parent 1 et parent 2) à partire du groupe gagnant, en se basant sur la stratégie de compétition dont nous avons discuté précédemment. Ainsi, les premiers et deuxièmes parents sont choisis à partire du groupe gagnant, tandis que le troisième parent est choisi dans le groupe perdant(La sélection par roulettes est employée)[15].

3. **Croissement :**Après la sélection des parents, le croisement est effectué entre les trois parents pour obtenir de nouvelles solutions. Le croisement est réalisé selon l'équation suivante :

$$X_{\text{new}} = \text{Crossover}(X_{\text{w1}}, X_{\text{w2}}, X_{\text{l}}), \tag{2.3}$$

Une stratégie de croisement stochastique simple est utilisée par dimension pour croiser les parents comme suite[15] :

$$\text{Si } r_1 \leq \frac{1}{3}, \text{ alors } X_{\text{new}}(d) = X_{\text{w1}}(d) \tag{2.4}$$

$$\text{Si } \frac{1}{3} < r_1 \leq \frac{2}{3}, \text{ alors } X_{\text{new}}(d) = X_{\text{w2}}(d) \tag{2.5}$$

$$\text{Sinon, } X_{\text{new}}(d) = X_{\text{l}}(d) \tag{2.6}$$

Où X_{w1} est la première composante du premier parent, X_{w2} est la première composante du deuxième parent, et X_{l} est la première composante du troisième parent. r_1 est un nombre aléatoire compris entre 0 et 1 utilisé pour déterminer lequel des trois parents contribue à la nouvelle solution pour chaque dimension.

4. **Mutation :**Dans l'algorithme génétique traditionnel, les nouvelles solutions (individus) sont mutées en utilisant un taux de mutation constant prédéterminé. En revanche, dans l'AGR,le taux de mutation est calculé dynamiquement en fonction de la condition de la fonction objective et de la génération.La formule pour le taux de mutation dynamique $MR(g)$ peut être écrite comme suit[15] :

$$MR(g) = \text{Favg} - g\left(\frac{\text{Favg}}{\text{MaxGene}}\right) \tag{2.7}$$

La formule pour Favg peut être écrite comme suit :

$$\text{Favg} = \frac{1}{N}\sum_{i=1}^{N}\text{Fiti}(g) \tag{2.8}$$

où **g** est la génération actuelle,**MaxGene** est le nombre maximum de générations, **Favg** est la valeur de la fonction objective moyenne obtenue à la génération g, **N** est le nombre des chromosomes, et **Fit** est la valeur de foction objective.

2.6 Évaluation des performances d'un modèle

2.6.1 Matrice de confusion :

Une matrice de confusion fournit une représentation tabulaire des prédictions faites par un modèle par rapport aux étiquettes de vérité terrain. Elle est généralement présentée sous forme de matrice 2x2, en considérant des scénarios de classification binaire. Il y a quatre éléments clés dans cette matrice : les vrais positifs, les vrais négatifs, les faux positifs et les faux négatifs[59].La figure 2.19 illustre la structure de la matrice de confusion.

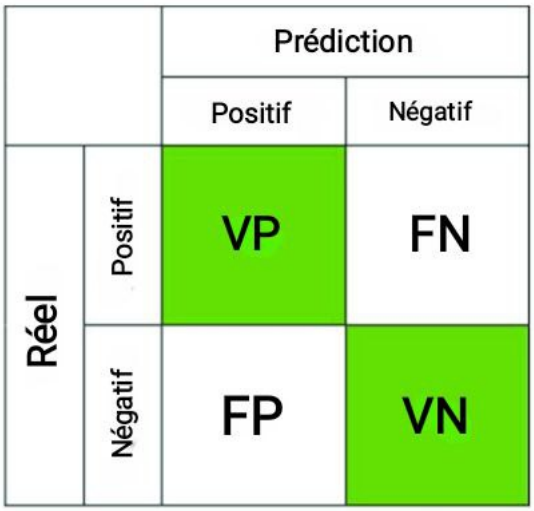


FIGURE 2.19 – Matrice de confusion [12]

— **VP (Vrais positifs)** : Nombre d'exemples positifs correctement classés.

— **VN (Vrais négatifs)** : Nombre d'exemples négatifs correctement classés.

— **FP (Faux positifs)** : Nombre d'exemples négatifs incorrectements classés comme positifs.

— **FN (Faux négatifs)** : Nombre d'exemples positifs incorrectement classés comme négatifs.

2.6.2 Métriques d'évaluation :

Il existe différentes mesures de performance utilisées pour évaluer l'efficacité des modèles prédictifs.

1. **Accuracy :** L'accuracy est une métrique pour évaluer la performance des modèles de classification à 2 classes ou plus.Elle représente le pourcentage de prédictions correctes parmi l'ensemble des prédictions faites par le modèle. Plus précisément, elle se calcule comme suit [60] :

$$\text{Accuracy} = \frac{\text{Nombre de prédictions correctes}}{\text{Nombre total de prédictions}} = \frac{VP + VN}{VP + FN + FP + VN} \tag{2.9}$$

2. **Taux de précision (Precision) :** Est un metrique pour évaluer la performance des modèles. La précision est également appelée Positive Predictive Value. Elle correspond au taux de prédictions correctes parmi les prédictions positives [60] :

$$\text{Taux de précision} = \frac{\text{Vrais positifs}}{\text{Vrais positifs + Faux positifs}} = \frac{VP}{VP + FP} \tag{2.10}$$

3. **Rappel (Recall) :** Est un metrique pour evalue la performance des modèles.Le recall est également appelé sensitivity (sensibilité), taux de vrais positifs ou encore taux de détection. Il correspond au taux d'individus positifs détectés par le modèle[60] :

$$\text{Rappel} = \frac{\text{Vrais positifs}}{\text{Vrais positifs + Faux négatifs}} = \frac{VP}{VP + FN} \tag{2.11}$$

4. **Score F1 (F1 Score) :** Le score F1 peut être interprété comme une moyenne harmonique de la précision et du rappel, où un score F1 atteint sa meilleure valeur à 1 et son pire score à 0. La contribution relative de la précision et du rappel au score F1 est égale. La formule du score F1 est la suivante [60] :

$$\text{Score F1} = 2 \times \frac{\text{Taux de précision} \times \text{Rappel}}{\text{Taux de précision} + \text{Rappel}} \tag{2.12}$$

2.6.3 Métriques d'erreur

Les métriques d'erreur sont les indicateurs les plus couramment utilisés dans la littérature pour évaluer la qualité des prédictions générées. Il existe plusieurs types de métriques d'erreur qui mesurent la « distance » moyenne entre les prévisions et les observations. Une valeur proche de 0 indique des prédictions parfaites, tandis qu'une valeur proche de 1 signale de mauvaises prédictions[61]. Pour mieux illustrer les mesures à venir, supposons que w_i représente la valeur observée et r_i la valeur prédite pour un ensemble de prévisions B_i.

1. **L'erreur absolue moyenne (MAE : Mean Absolute Error) :** Cette mesure est la métrique d'erreur la plus couramment utilisée. Elle permet d'évaluer la qualité des prédictions fournies. En ce sens, le MAE mesure la déviation absolue moyenne entre une estimation prévue et l'estimation réelle de l'utilisateur[62].

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{2.13}$$

2. **La racine de l'erreur quadratique moyenne (RMSE : Root Mean Squared Error) :** Le RMSE est une mesure couramment utilisée pour évaluer la précision d'un modèle de prévision ou de régression. Dans la littérature, le RMSE est largement préféré au MSE (Mean Square Error) pour l'évaluation. Il a notamment été utilisé dans le célèbre concours Netflix Prize pour identifier les meilleurs algorithmes de filtrage[62][63].

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.14}$$

2.7 Les explications additives de Shapley :

SHAP est une méthode d'explicabilité des modèles d'apprentissage automatique basée sur la théorie des valeurs de Shapley. Elle attribue une contribution à chaque caractéristique d'entrée pour expliquer la prédiction d'un modèle. SHAP fournit des explications à deux niveaux : l'importance globale des caractéristiques sur l'ensemble du dataset et l'importance locale pour des prédictions spécifiques. Cela permet de comprendre comment chaque caractéristique influence les décisions du modèle de manière cohérente et transparente, quelle que soit la complexité du modèle utilisé.
Pour calculer les valeurs de Shapley (ϕ_i) pour une caractéristique i, l'équation mathématique est [64] :

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[v(S \cup \{i\}) - v(S)\right] \tag{2.15}$$

où :

— N est l'ensemble des caractéristiques.

— S est un sous-ensemble de N ne contenant pas i.

— $v(S)$ est la valeur de la fonction de prédiction pour le sous-ensemble S.

2.8 Conclusion

En conclusion, ce chapitre a fourni une vue générale sur les différents concepts de l'apprentissage automatique et de l'apprentissage profond. Nous avons commencé par définir les types d'apprentissage automatique avec une explication de certains algorithmes comme Random Forest, SVM, KNN, etc. Ensuite, nous avons donné un aperçu de l'apprentissage profond, incluant les informations fondamentales sur les réseaux de neurones artificiels et leurs différents types. Nous avons également parlé de la sélection des attributs en présentant les différentes méthodes de sélection, comme les algorithmes génétiques et AGR. Enfin, nous avons examiné les techniques d'évaluation des performances des modèles , les mesures d'erreur couramment utilisées,ainsi que SHAP.

Dans le chapitre suivant, nous concentrerons sur l'implémentation des modèles utilisés pour la prédiction des maladies cardiovasculaires, en détaillant les architectures employées et les différentes méthodes appliquées.

Chapitre 3

Méthodologie proposée

3.1 Introduction :

La détection précoce des maladies cardiaques demeure un défi majeur, ces maladies étant parmi les principales causes de mortalité à l'échelle mondiale. Les récentes avancées dans les techniques d'apprentissage automatique offrent des perspectives prometteuses. Il est donc essentiel de développer un modèle de classification efficace pour anticiper l'état cardiaque des patients.

Dans ce chapitre, nous présentons une approche basée sur un modèle de réseau de neurones multicouches (MLPClassifier) pour prédire les maladies cardiaques. Afin d'améliorer les performances de ce modèle, nous avons implémenté un algorithme génétique rival pour choisir les caractéristiques les plus pertinentes. Les résultats de notre modèle seront comparés à ceux obtenus avec d'autres modèles tels que SVM, XGBoost, DT, RF, KNN, BN, DNN, SimpleRNN, GRU, et LSTM. Cette comparaison permettra de mettre en évidence l'efficacité de notre méthode en termes de l'accuracy, précision, rappel, et F1 score

3.2 Etat de l'art

Dans cette section, nous avons exposé quelques recherches effectuées récemment dans le domaine de la prédiction des maladies cardio-vasculaires.

A.Gupta et al. [65] L'étude utilise une approche hybride combinant l'élimination récursive des caractéristiques basée sur un algorithme génétique (AG-RFE) avec l'algorithme AdaBoost pour améliorer la prédiction des maladies cardiaques et du diabète. Les jeux de données Cleveland et Pima, provenant de l'UCI, ont été utilisés pour valider le modèle. Le prétraitement des données a impliqué la gestion des valeurs manquantes à l'aide de l'imputation multiple par équations enchaînées (MICE). L'algorithme GA-RFE a sélectionné 8 caractéristiques pertinentes pour le jeu de données Pima et 7 pour le jeu de données Cleveland. L'AdaBoost a ensuite été entraîné avec ces caractéristiques sélectionnées, améliorant ainsi la performance du modèle. Les performances du modèle ont été mesurées en termes de précision,

sensibilité, spécificité et F-mesure. Le modèle AdaBoost a atteint une précision de classification de 91,9 % sur le jeu de données Cleveland et de 96,6 % sur le jeu de données Pima.

Fahad A. Alghamdi et al.[66]L'étude utilise une approche combinant un réseau de neurones perceptron multicouche (MLPNN) avec un algorithme d'optimisation arithmétique (AOA) pour sélectionner les caractéristiques les plus pertinentes. Le modèle atteint une précision de 88,89% en utilisant le jeu de données de cleveland.

J. Nageswara Rao et al.[67] Les auteurs proposent un système de prédiction des maladies cardiaques en utilisant la base de données cleveland pour leur étude, qui comprend 13 attributs différents. Ils ont appliqué des techniques d'apprentissage automatique, avec RF comme algorithme le plus performant. Ce dernier a produit les résultats les plus fiables, avec une précision de 90,16%.

Aarti Karandikar et al.[68] L'étude explore une nouvelle approche pour prédire les maladies cardiaques en utilisant HCNN dans le cadre de l'internet des objets médicaux (IoMT). À partir d'un ensemble de données, l'HCNN, qui intègre des couches convolutives, des blocs résiduels et des mécanismes d'attention, peut extraire des caractéristiques complexes. Cette simulation a obtenu une précision de 92%.

3.3 L'approche proposée

L'architecture globale de l'approche proposée pour la prédiction des maladies cardiovasculaires est illustrée dans la figure 3.1. Cette architecture se compose de cinq étapes principales : l'analyse exploratoire des données, le prétraitement des données, la sélection des attributs, la classification des données et enfin l'évaluation des performances. La première étape a pour objectif d'identifier les relations potentielles entre les attributs. La deuxième étape sert à la gestion des données manquantes et dupliquées, la normalisation des données, La transformation des valeurs catégoriques en numériques ainsi que la discrétisation. La troisième étape consiste à sélectionner les meilleurs attributs du système. La quatrième étape se concentre sur la classification des données en construisant un modèle de prédiction basé sur différents algorithmes tels que SVM, KNN, XGBoost, RF, DT, BN, SimpleRNN, GRU, LSTM, DNN et MLPClassifier. Enfin, la cinquième étape vise à évaluer les performances des modèles construits lors de l'étape précédente afin de choisir le meilleur modèle pour prédire les maladies cardiovasculaires.

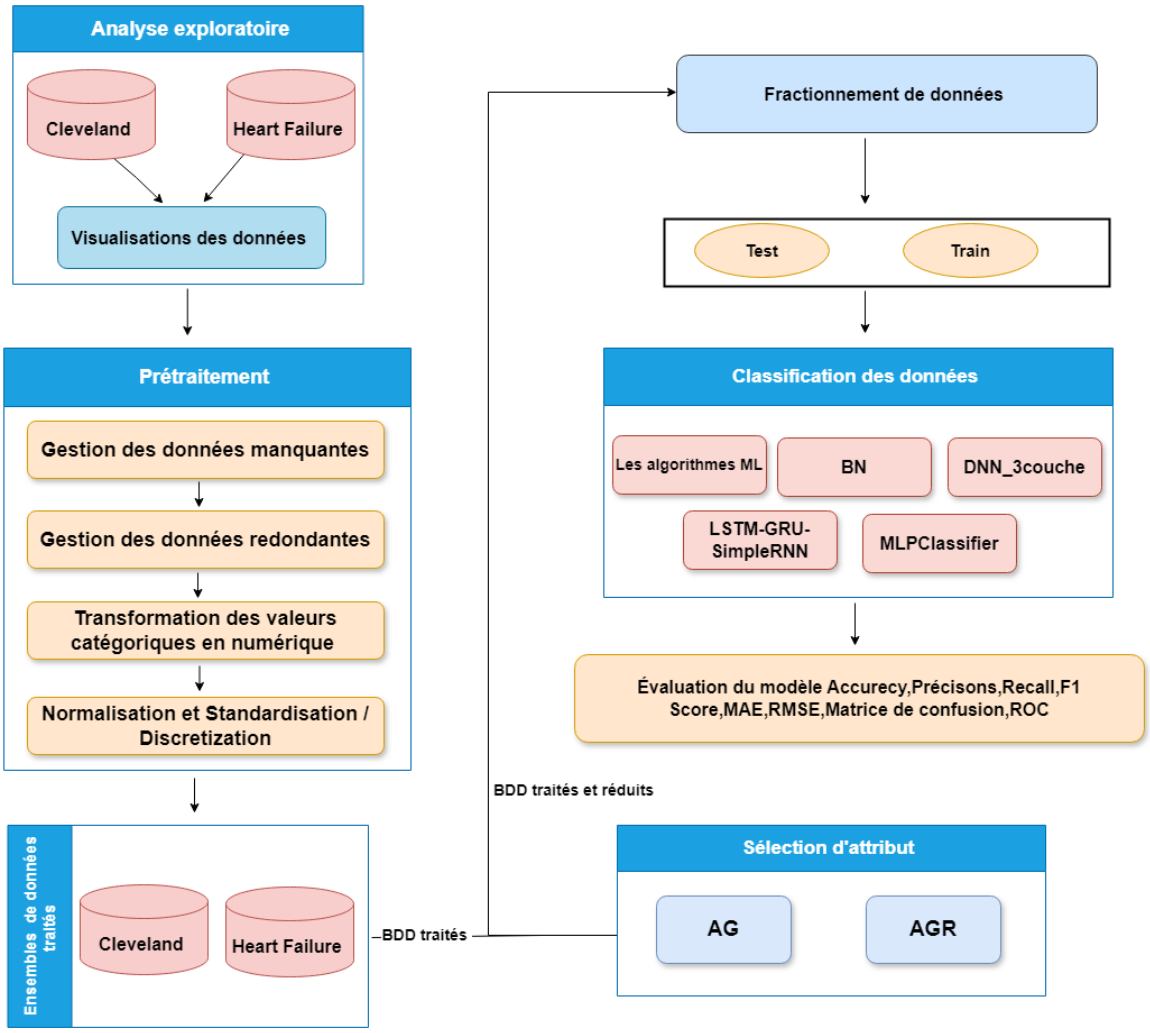


FIGURE 3.1 – L'architecture du système proposé

3.4 Analyse exploratoire

L'analyse exploratoire est une étape cruciale dans l'analyse de données. Elle fait usage de graphiques et de visualisations pour examiner et interpréter un ensemble de données. Cette approche permet d'identifier les relations potentielles entre les variables tout en fournissant une vue d'ensemble de la nature, de la structure et des interactions des données. En outre, elle joue un rôle essentiel dans la détection et la correction des erreurs et des incohérences présentes dans l'ensemble des données[69].

3.4.1 Analyse exploratoire de la première base de données (Cleveland)

Pour notre étude, nous utilisons le jeu de données Cleveland Heart Disease [70], un ensemble de données bien connu et accessible au public. Ce jeu de données regroupe des informations médicales de patients adressés à la Cleveland Clinic Foundation entre 1988 et 1990 pour suspicion de maladie

cardiaque. Il contient 303 instances avec 13 variables indépendantes représentant des mesures cliniques et des informations démographiques, ainsi qu'une variable dépendante servant de cible. La variable cible se divise en deux classes : la classe 1 indique la présence d'une maladie cardiaque, tandis que la classe 0 indique son absence. Le tableau 3.1 présente une description détaillée du jeu de données Cleveland.

1. **Analyse de la forme :**

   - **Description De base de données :**

Variable	**Description**
age	Âge du patient en années
sex	Sexe du patient (1 = homme, 0 = femme)
cp	Type de douleur thoracique : 0 : angine typique 1 : angine atypique 2 : douleur non angineuse 3 : asymptomatique
trestbps	Tension artérielle au repos en mm Hg
chol	Cholestérol sérique en mg/dl
fbs	Taux de sucre dans le sang à jeun, classé comme supérieur à 120 mg/dl (1 = vrai, 0 = faux)
restecg	Résultats électrocardiographiques au repos : 0 : Normal 1 : Anomalie de l'onde ST-T 2 : Hypertrophie ventriculaire gauche probable ou certaine selon les critères d'Estes
thalach	Fréquence cardiaque maximale atteinte lors d'un test d'effort
exang	Angine induite par l'exercice (1 = oui, 0 = non)
oldpeak	Dépression ST induite par l'exercice par rapport au repos
slope	Segment ST mesuré en termes de pente pendant l'exercice de pointe : 0 : vers le haut 1 : plat 2 : en pente descendante
ca	Nombre de vaisseaux sanguins principaux (0–3)
thal	Un trouble sanguin appelé thalassémie : 0 : NULL 1 : flux sanguin normal 2 : défaut corrigé 3 : défaut réversible
target	Statut de maladie cardiaque (0 = aucune maladie, 1 = présence d'une maladie)

TABLE 3.1 – Description de la base de données

La figure 3.2 illustre la distribution statistique des différents attributs de notre ensemble de données, comprenant les valeurs minimales, maximales, moyennes et l'écart type (STD).

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
count	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00	303.00
mean	54.37	0.68	0.97	131.62	246.26	0.15	0.53	149.65	0.33	1.04	1.40	0.73	2.31	0.54
std	9.08	0.47	1.03	17.54	51.83	0.36	0.53	22.91	0.47	1.16	0.62	1.02	0.61	0.50
min	29.00	0.00	0.00	94.00	126.00	0.00	0.00	71.00	0.00	0.00	0.00	0.00	0.00	0.00
25%	47.50	0.00	0.00	120.00	211.00	0.00	0.00	133.50	0.00	0.00	1.00	0.00	2.00	0.00
50%	55.00	1.00	1.00	130.00	240.00	0.00	1.00	153.00	0.00	0.80	1.00	0.00	2.00	1.00
75%	61.00	1.00	2.00	140.00	274.50	0.00	1.00	166.00	1.00	1.60	2.00	1.00	3.00	1.00
max	77.00	1.00	3.00	200.00	564.00	1.00	2.00	202.00	1.00	6.20	2.00	4.00	3.00	1.00

FIGURE 3.2 – Distribution statistiques des attributs

La figure 3.3 présente les graphiques de visualisation de la classe cible. Il est observé que 138 individus n'ont pas de maladie cardiaque, représentant 45,5 % de l'échantillon, tandis que 165 individus en sont atteints, soit 54,5 %. Cela indique que le pourcentage d'individus avec et sans maladie cardiaque est proche, mais pas tout à fait équivalent, c'est-à-dire que l'ensemble de données est presque équilibré. Un ensemble de données équilibré est essentiel pour obtenir des résultats de classification optimaux, car il permet au modèle de s'entraîner sur un nombre égal,exemples positifs et négatifs.

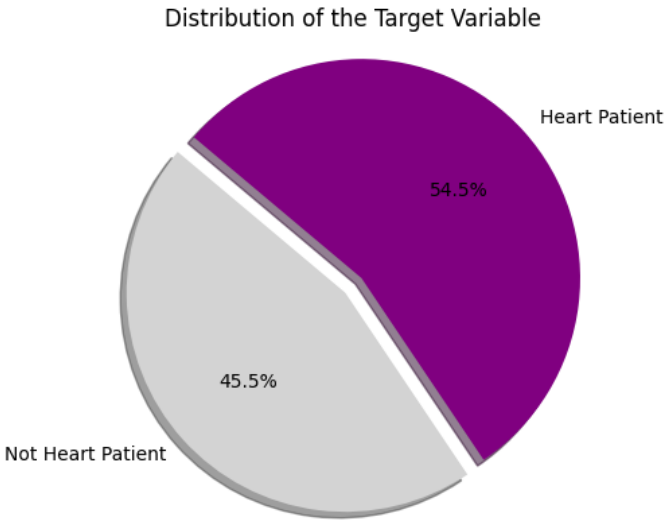


FIGURE 3.3 – Visualisation de l'attribut cible

2. **Analyse univariée :** Pour chaque attribut du jeu de données, explorez sa distribution et sa va-

riabilité en utilisant des outils visuels tels que des histogrammes, des graphiques à barres et des diagrammes en boîte. Ces visualisations permettent d'analyser la répartition et l'étendue des valeurs présentes dans chaque attribut.

- **Analyse univariée des attributs continues :**
  - **age** : La distribution est relativement uniforme, mais il y a un pic autour de la fin de la cinquantaine. L'âge moyen est d'environ 54,37 ans, avec un écart-type de 9,08.
  - **trestbps** : La pression artérielle au repos pour la plupart des individus est concentrée autour de 120-140 mm Hg, avec une moyenne d'environ 131,62 mm Hg et un écart-type de 17,54 mm Hg.
  - **chol** : La plupart des individus ont des niveaux de cholestérol entre 200 et 300 mg/dl. Le niveau moyen de cholestérol est d'environ 246,26 mg/dl, avec un écart-type de 51,83 mg/dl.
  - **thalach** : La majorité des individus atteignent une fréquence cardiaque entre 140 et 170 bpm lors d'un test d'effort. La fréquence cardiaque moyenne atteinte est d'environ 149,65 bpm, avec un écart-type de 22,91 bpm.
  - **oldpeak** : La plupart des valeurs sont concentrées autour de 0, ce qui indique que de nombreux individus n'ont pas connu de dépression ST significative pendant l'exercice. La valeur moyenne de la dépression ST est de 1,04, avec un écart-type de 1,16.

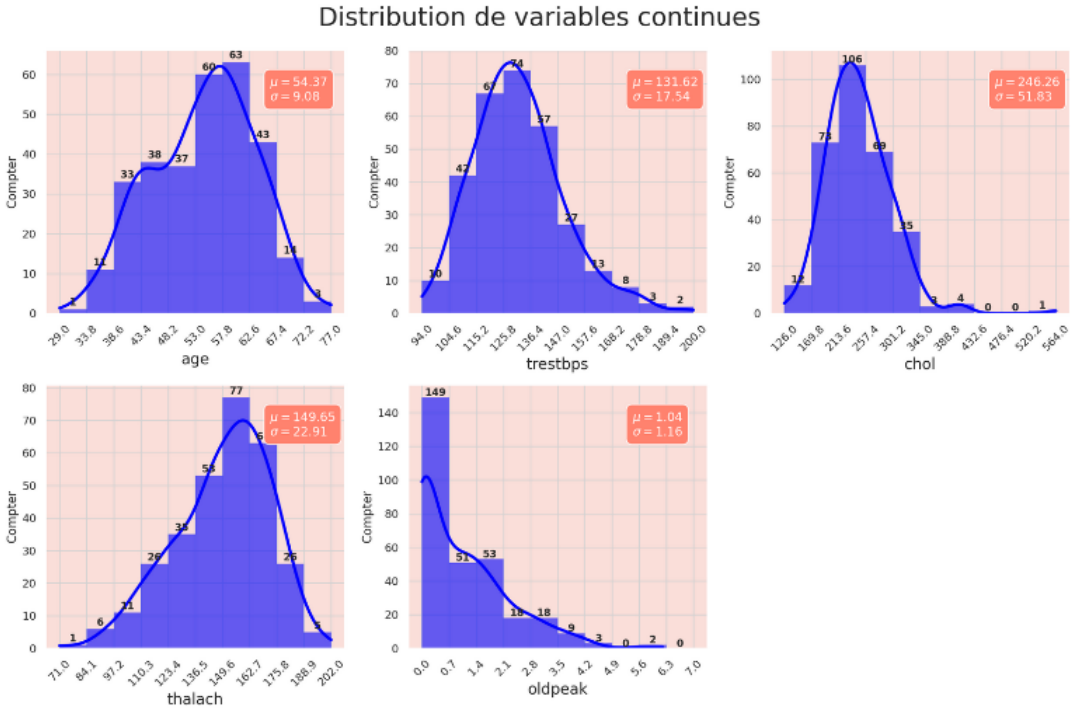
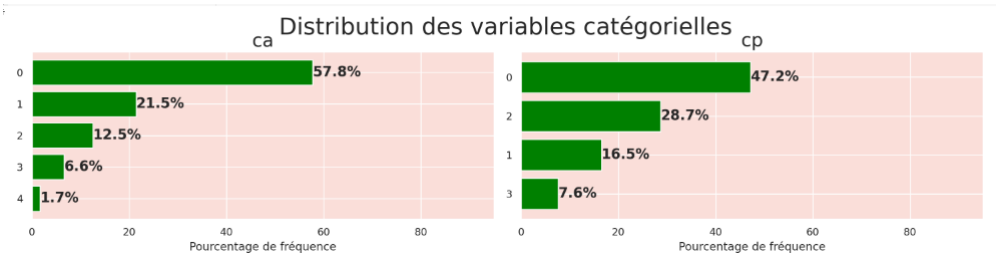


FIGURE 3.4 – Distribution des variables continues

- **Analyse univariée des attributs catégoriels :**

- **Type de douleur thoracique (cp)** :Les données montrent une variété de types de douleur thoracique chez les patients. Bien que le type 0 (Angine typique) semble être le plus répandu.
- **ca** : La plupart des patients ont moins de gros vaisseaux colorés par fluoroscopie, '0' étant le plus fréquent.



- **fbs :**La plupart des individus (85.1 %) ont une glycémie à jeun inférieure à 120 mg/dl (valeur 0).En conséquence, le fbs n'est pas une affection répandue dans ce jeu de données ; seuls 14,9 % des personnes ont une glycémie à jeun supérieure à 120 mg/dl (valeur 1).
- **exang :**La majorité des patients ne ressentent pas d'angine induite par l'exercice, ce qui suggère que ce n'est peut-être pas un symptôme courant parmi les patients de cette collection de données.

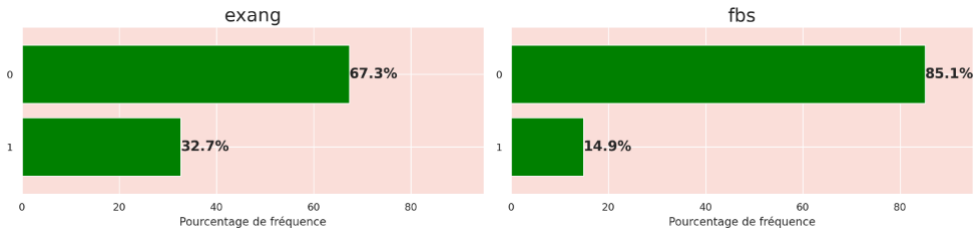


FIGURE 3.5 – Distribution de fbs et exang

- **restecg :**Les résultats anormaux (anomalies ST-T (1)) de l'électrocardiogramme sont légèrement plus fréquents dans cet ensemble par rapport aux autres types.
- **sex :**Les hommes représentent la majorité du personnel du jeu de données (68.3%).

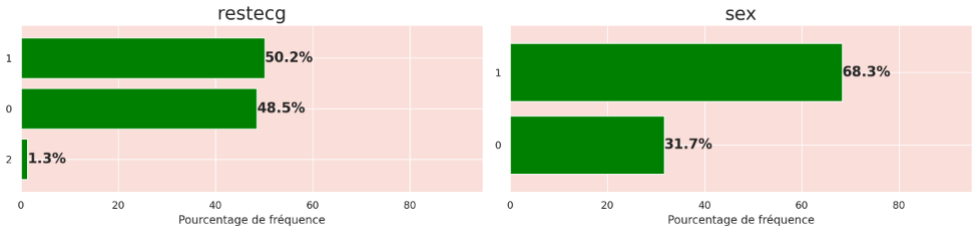


FIGURE 3.6 – Distribution de restecg et sex

- **slope :**Selon la distribution de la variable "slope", les catégories 2 (en pente descendante) et 1 (plat) sont presque également réparties, avec 46,9 % pour les catégories 2 et 46,2 % pour les catégories 1. Seuls 6,9 % des observations appartiennent à la catégorie 0.
- **target :**Le jeu de données est presque équilibré en termes de présence de maladie cardiaque, avec environ 54,5 % ayant une maladie cardiaque et 45,5 % sans.

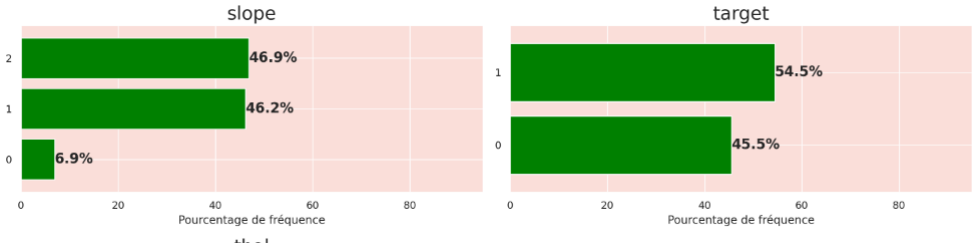


FIGURE 3.7 – Distribution de slope et target

- **Thal :**La plupart des données concernant la variable 'thal' ont été attribuées principalement aux catégories 2 (défaut corrigé) (54,8%) et 3 (défaut réversible) (38,6 %), ce qui suggère que ces deux catégories ont une forte influence dans l'échantillon. Les catégories 1 (5,9 %) et 0 (0,7 %) ont une représentation significativement inférieure.

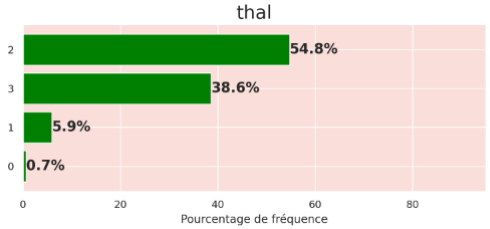


FIGURE 3.8 – Distribution de thal

3. **Analyse bivariée :** L'analyse bivariable consiste à explorer la relation entre chaque caractéristique et la variable cible (target).
   - **Analyse bivariée des attributs continues :** Pour visualiser l'impact de chaque variable continue sur la variable cible.
     — **Âge :** Les distributions montrent une légère différence entre les patients atteints de maladie cardiaque et ceux qui n'en ont pas. Les patients sans maladie cardiaque ont généralement un âge plus élevé. La densité de concentration est d'environ 60 à 65 ans pour ceux sans maladie cardiaque et d'environ 50 à 60 ans pour ceux avec maladie cardiaque, ce qui signifie que la majorité des patients atteints de maladie cardiaque ont entre 50 et 60 ans. Les personnes sans maladie cardiaque ont généralement entre 60 et 65 ans.
     — **Trestbps :** Les distributions des deux catégories se chevauchent dans le tracé KDE, avec des valeurs moyennes presque identiques, indiquant un pouvoir discriminant limité pour cette caractéristique. Cependant, il existe une légère différence où la courbe pour les personnes souffrant de maladie cardiaque (courbe verte) est un peu plus étalée autour de 120-140.
     — **Chol :** Les niveaux de cholestérol des deux groupes sont assez comparables, mais le niveau moyen des patients atteints de maladie cardiaque est légèrement plus bas.

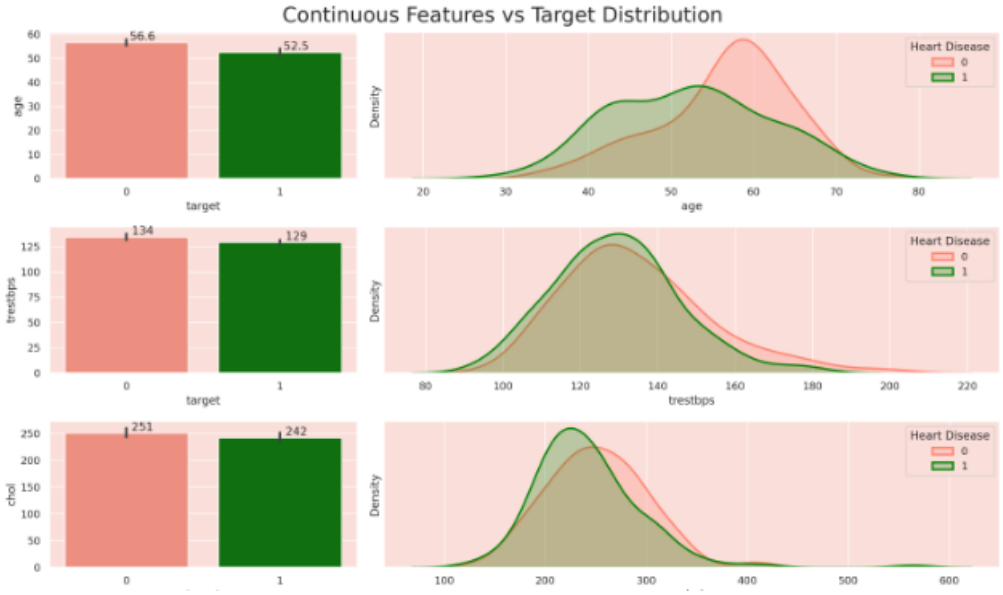


FIGURE 3.9 – Age,trestbps et chol Vs arget

— **Thalach :** Les distributions diffèrent considérablement. Par rapport à ceux sans maladie, les patients atteints de maladie cardiaque ont tendance à avoir une fréquence cardiaque maximale plus élevée lors des tests de stress.
— **Oldpeak :** La dépression ST induite par l'exercice est significativement plus faible pour les patients atteints de maladie cardiaque par rapport au repos. Leur répartition est presque

nulle, tandis que la catégorie sans maladie a une répartition plus large.

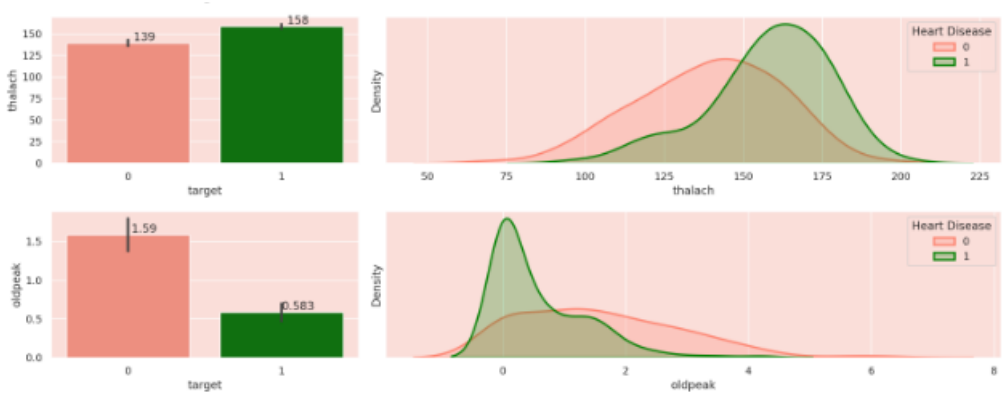


FIGURE 3.10 – thalach et oldpeak Vs Target

Résumé

- **Résumé :** En se basant sur les différences visuelles dans les distributions et les valeurs moyennes, la fréquence cardiaque maximale (**thalach**) semble avoir le plus d'impact sur le statut de la maladie cardiaque, suivie de la dépression ST (**oldpeak**) et de l'âge (**âge**).

- **Analyse bivariée des attributs catégoriels :**

  Nous utilisons 'sns.countplot' pour créer des diagrammes de comptage pour chaque variable catégorielle par rapport à la variable cible. Les nombres sont affichés au-dessus de chaque barre et une légende est incluse pour la variable cible.

FIGURE 3.11 – les caractéristiques catégorielles Vs target

— **cp** : Les proportions de présence de maladie cardiaque varient selon les différents types de douleur thoracique. En particulier, les types 1, 2 et 3 présentent une probabilité plus élevée de maladie cardiaque par rapport au type 0. Cela suggère que le type de douleur thoracique affecte la prédiction de la maladie.

— **exang** : Les patients qui n'ont pas eu d'angine induite par l'exercice (0) présentent une proportion plus élevée de présence de maladie cardiaque par rapport à ceux qui l'ont eu (1). Cette caractéristique semble avoir un impact significatif sur la variable cible.

— **fbs** : La glycémie à jeun supérieure à 120 mg/dl (1) et les personnes sans glycémie élevée (0) sont relativement similaires, suggérant que la glycémie à jeun peut avoir un impact limité sur la prédiction de la maladie cardiaque.

— **restecg** : Le type 1 affiche une proportion plus élevée de présence de maladie cardiaque, ce qui indique que cette caractéristique pourrait avoir une certaine influence sur le résultat.

— **sex** : Les taux de maladie cardiaque sont plus élevés chez les femmes (1) que chez les hommes (0). Cela suggère que le sexe est un facteur influent dans la prédiction de la maladie cardiaque.

— **slope** : La pente de type 2 présente une proportion notablement plus élevée de présence de maladie cardiaque, ce qui indique son potentiel en tant que prédicteur significatif.

— **thal** : Par rapport aux autres catégories, la catégorie de défaut corrigé (2) présente une proportion plus élevée de maladie cardiaque, ce qui indique son potentiel en tant que prédicteur significatif.

En résumé, d'après la représentation visuelle :

- **Impact Élevé sur la Cible :**ca, cp, exang, sex, slope et thal
- **Impact Modéré sur la Cible :**restecg
- **Faible Impact sur la Cible :**fbs

4. **Analyse de la corrélation entre les attributs**

Le but de cette matrice est de montrer les relations entre les caractéristiques dans un ensemble de données. Plus la valeur de corrélation est proche de zéro, plus la corrélation entre la caractéristique et la variable cible est faible. En revanche, plus la valeur de corrélation est proche de -1 ou 1, cela indique une forte corrélation.

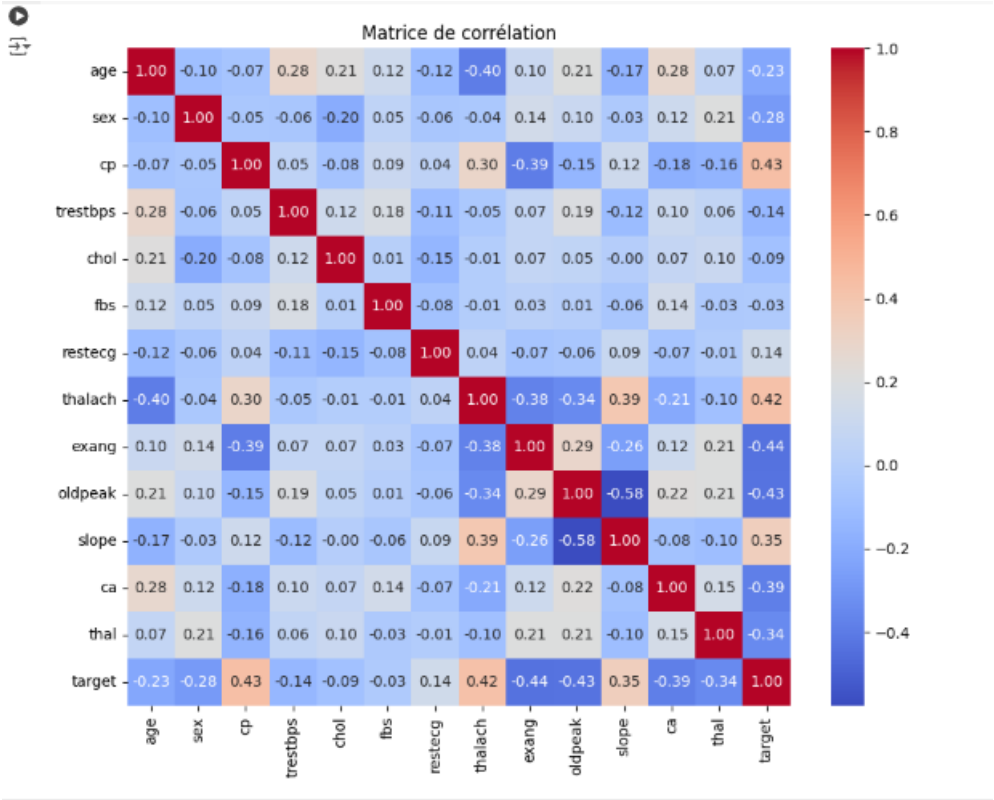


FIGURE 3.12 – Matrice de corrélation

- Selon cette matrice, nous observons que les caractéristiques 'fbs' et chol sont les moins associées à la variable target.
- cp ,exang,slope, thalach et oldpeak présentent une corrélation significative avec le target.

5. **Gestion des valeurs aberrantes**

Nous avons utilisé l'approche de l'écart interquartile pour détecter les valeurs aberrantes, car elle est la plus couramment utilisée et la plus fiable comme illustre dans la figure **3.13**.

```
Q1 = df[continuous_features].quantile(0.25)
Q3 = df[continuous_features].quantile(0.75)
IQR = Q3 - Q1
outliers_count_specified = ((df[continuous_features] < (Q1 - 1.5 * IQR)) | (df[continuous_features] > (Q3 + 1.5 * IQR))).sum()

outliers_count_specified
```

```
age         0
trestbps    9
chol        5
thalach     1
oldpeak     5
dtype: int64
```

FIGURE 3.13 – les valeurs aberrantes pour les caractéristiques continues

3.4.2 Analyse exploratoire de la deuxième base de données (Heart Failure)

Nous utilisons le jeu de données Heart Failure [71], un ensemble de données bien connu et accessible au public. Il contient 918 instances avec 12 variables.

1. **Analyse de la forme :**

   Le tableau 3.2 présente une description détaillée du jeu de données Heart Failure.

Variable	**Description**
Âge	L'âge de l'individu en années.
Sex	Le sexe de l'individu, représenté par "M" pour masculin et "F" pour féminin.
ChestPainType	Le type de douleur thoracique rapportée par l'individu : — ATA : Typical Angina — NAP : Non-Anginal Pain — ASY : Atypical Angina — TA : Atypical Chest Pain
RestingBP	La pression artérielle au repos de l'individu, mesurée en millimètres de mercure (mmHg).
Cholesterol	Le niveau de cholestérol de l'individu, généralement mesuré en milligrammes par décilitre (mg/dL).
FastingBS	Le taux de glycémie à jeun de l'individu, indiqué généralement par "0" pour normal et "1" pour élevé.
RestingECG	Les résultats de l'électrocardiogramme (ECG) au repos de l'individu. Cela peut inclure des catégories telles que "Normal", "ST" (Suspect de l'hypertrophie ventriculaire gauche).
MaxHR	La fréquence cardiaque maximale atteinte par l'individu pendant un test d'effort ou un exercice, mesurée en battements par minute (bpm).

ExerciseAngina	Indique si l'individu ressent de l'angine (douleur ou inconfort thoracique) pendant l'activité physique : — "Y" : indique généralement que l'individu ressent de l'angine pendant l'exercice. — "N" : indique généralement que l'individu ne ressent pas d'angine pendant l'exercice.
Oldpeak	Représente la dépression du segment ST par rapport au repos.
ST-slope	Indique la pente du segment ST sur l'électrocardiogramme (ECG) pendant l'exercice : — "Up" : indique une pente ascendante du segment ST. — "Flat" : indique un segment ST plat. — "Down" : indique une pente descendante du segment ST.
HeartDisease	Indique si l'individu a une maladie cardiaque : — "1" : indique la présence de maladie cardiaque. — "0" : indique l'absence de maladie cardiaque.

TABLE 3.2 – Description de la base de données

La figure 3.14 illustre la distribution statistique des différents attributs de notre ensemble de données, comprenant les valeurs minimales, maximales, moyennes et l'écart type (STD).

	Age	Sex	ChestPainType	RestingBP	Cholesterol	FastingBS	RestingECG	MaxHR	ExerciseAngina	Oldpeak	ST_Slope	HeartDisease
count	918.00	918.00	918.00	918.00	918.00	918.00	918.00	918.00	918.00	918.00	918.00	918.00
mean	53.51	0.79	0.78	132.40	198.80	0.23	0.99	136.81	0.40	0.89	1.36	0.55
std	9.43	0.41	0.96	18.51	109.38	0.42	0.63	25.46	0.49	1.07	0.61	0.50
min	28.00	0.00	0.00	0.00	0.00	0.00	0.00	60.00	0.00	-2.60	0.00	0.00
25%	47.00	1.00	0.00	120.00	173.25	0.00	1.00	120.00	0.00	0.00	1.00	0.00
50%	54.00	1.00	0.00	130.00	223.00	0.00	1.00	138.00	0.00	0.60	1.00	1.00
75%	60.00	1.00	2.00	140.00	267.00	0.00	1.00	156.00	1.00	1.50	2.00	1.00
max	77.00	1.00	3.00	200.00	603.00	1.00	2.00	202.00	1.00	6.20	2.00	1.00

FIGURE 3.14 – Distribution statistiques des attributs

La figure 3.15 présente les graphiques de visualisation de la classe cible. Il est observé que 410 individus n'ont pas de maladie cardiaque, représentant 44,7 % de l'échantillon, tandis que 508 individus en sont atteints, soit 55,3 %. Cela indique que le pourcentage d'individus avec et sans

maladie cardiaque n'est pas équivalent, l'ensemble de données étant légèrement déséquilibré en faveur des patients atteints de maladie cardiaque.

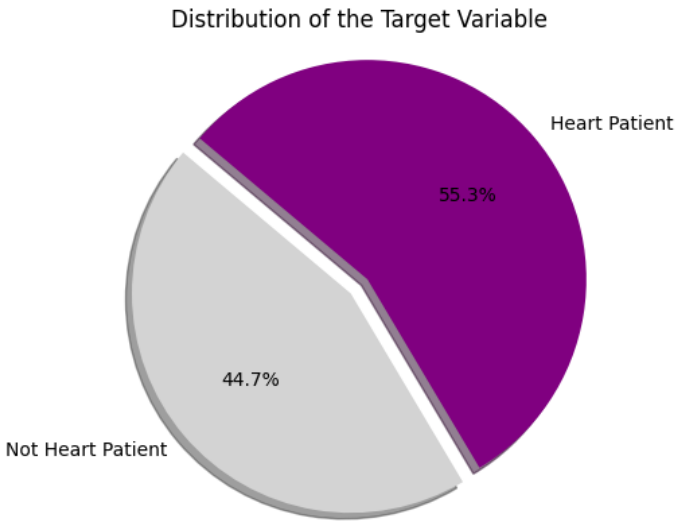


FIGURE 3.15 – Visualisation de l'attribut cible

2. **Analyse univariée et bivarié :** Les mêmes analyses univariées et bivariées ont été effectuées sur le deuxième ensemble de données. Les points essentiels de cette analyse sont résumés comme suit :
   - **Analyse univariée des caractéristiques catégorielles :**
     - Dans le jeu de données, la majorité des individus sont des hommes.
     - Le type de douleur thoracique le plus fréquent est ASY (54,0%), suivi par NAP, ATA, et TA (chacun à 5%).
     - La majorité des patients (59,6%) ne ressentent pas d'angine induite par l'exercice.
     - Le type 'Normal' de l'ECG au repos est le plus courant.
     - Concernant la pente du segment ST, "flat" est le plus spécifique, avec "up" en maximum et "down" en moyenne.
     - La majorité des individus (76,7%) ont une glycémie à jeun inférieure à 120 mg/dl (indiqué par la valeur 0).
   - **Analyse univariée des caractéristiques numériques :**
     - Les individus ont principalement un âge autour de 50 ans, avec une variation notable selon l'écart-type.
     - La majorité des individus présente une pression artérielle au repos dans une fourchette spécifique, avec une certaine variation autour de la moyenne.
     - Les niveaux de cholestérol sont généralement élevés et montrent une grande variation parmi les individus
     - Le rythme cardiaque maximal présente une grande variabilité autour de la moyenne.

— La plupart des individus n'ont pas connu de dépression ST significative pendant l'exercice, avec une moyenne proche de 0 mais une variation notable.

- **Analyse bivarié des caractéristiques catégorielles et numériques :**
  - **Âge :**Âge moyen plus élevé pour les personnes avec maladie cardiaque.
  - **RestingBP :**Moyenne similaire pour les cas avec et sans maladie cardiaque.
  - **Cholesterol :** Niveau moyen plus élevé pour les cas sans maladie cardiaque.
  - **MaxHR :**Niveau moyen plus élevé pour les cas sans maladie cardiaque.
  - **Oldpeak :**Niveau moyen plus élevé pour les cas avec maladie cardiaque.
  - **ChestPainType :**ASY plus de cas avec maladie cardiaque.
  - **ExerciseAngina :** Y plus de cas avec maladie cardiaque.
  - **FastingBS :** Niveau de glycémie élevé, plus de cas avec maladie cardiaque.
  - **RestingECG :** Normal plus de cas avec maladie cardiaque.
  - **ST_Slope :**Flat plus de cas avec maladie cardiaque.
  - **Sex :**M plus de cas avec maladie cardiaque.

3. **Analyse de la corrélation entre les attributs :** Cette matrice vise à mettre en évidence les liens entre les attributs dans un ensemble de données.

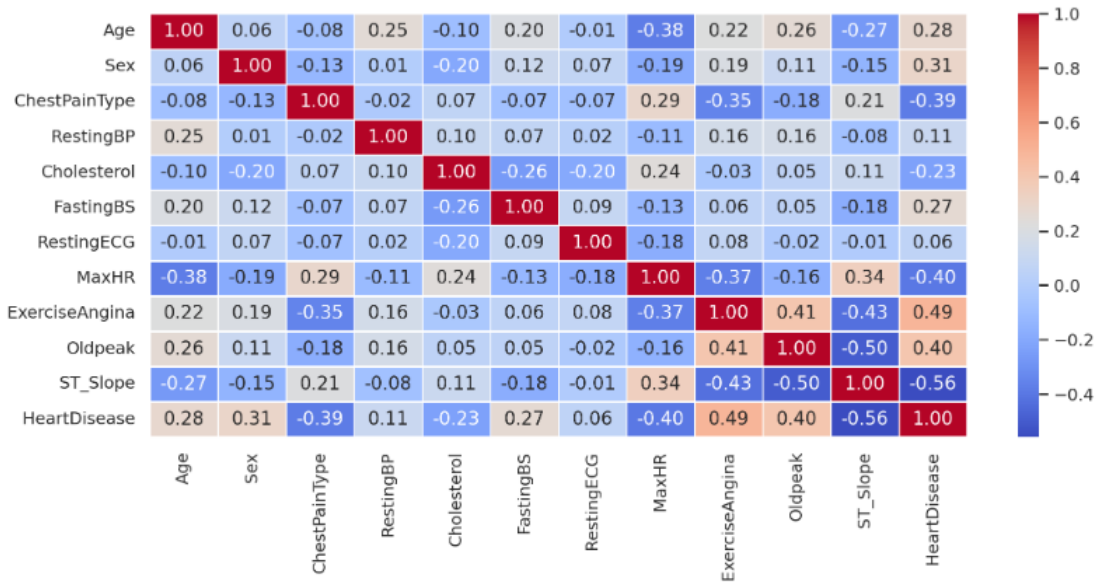


FIGURE 3.16 – Matrice de corrélation

- Selon cette matrice, nous observons que les attributs 'Oldpeak','MaxHR', 'St_Slope', 'ExerciseAngina' et 'ChestPainType' présentent une corrélation significative avec le HeartDisease.
- 'RestingECG' et 'RestingBP' sont les moins associées à la variable HeartDisease.

4. **Gestion des valeurs aberrantes :** Nous vérifions les valeurs aberrantes en utilisant la méthode IQR(Intervalle Interquartile)pour les fonctionnalités continues.Dans la figure 3.17 nous observons l'existence des valeurs aberrantes dans notre ensemble de données dans les colonnes suivantes : [RestingBP, Cholesterol,MaxHR,Oldpeak].

FIGURE 3.17 – Les valeurs aberrantes

3.5 Prétraitement :

L'objectif de cette étape (data preprocessing) est de préparer les données brutes pour leur utilisation dans des modèles d'apprentissage automatique, en assurant que les données sont propres, cohérentes et dans un format adéquat pour l'analyse ou pour l'entraînement de ces modèles. Cela permet d'obtenir des résultats plus précis et fiables et peut se faire comme suit :

3.5.1 Prétraitement de premier ensemble de données

1- Gestion des données manquantes :

La gestion des données manquantes implique de traiter les valeurs manquantes (”NaN” ou ”Na”) causées par des erreurs de collecte et des enregistrements incomplets. Pour gérer ces valeurs, on peut supprimer les lignes concernées ou les remplacer par la moyenne ou la médiane des valeurs présentes dans chaque colonne. Nous avons utilisé instruction illustré dans la figure 3.18 pour vérifier s'il y avait des valeurs manquantes et nous avons constaté que cet ensemble de données ne contient aucune valeur manquante

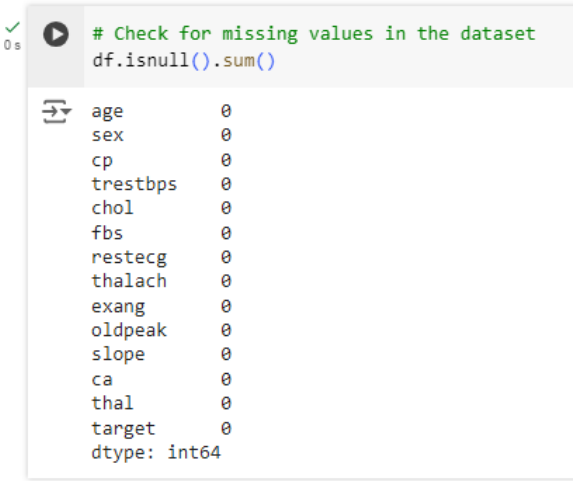


FIGURE 3.18 – Les valeurs manquantes

2- Gestion des valeurs aberrantes :

Le nombre de vaisseaux sanguins principaux **(ca)** varie entre 0 et 3, tandis que la thalassémie (thal) présente des valeurs allant de 1 à 3. Cependant, lors de l'analyse exploratoire, il a été remarqué que certaines cellules contiennent des valeurs de 4 pour le ca et 0 pour le thal. Nous avons également utilisé la fonction unique pour vérifier le nombre unique de valeurs pour ca et thal, comme illustré dans la figure 3.19.Ces valeurs aberrantes seront supprimés du jeu de données du Cleveland.

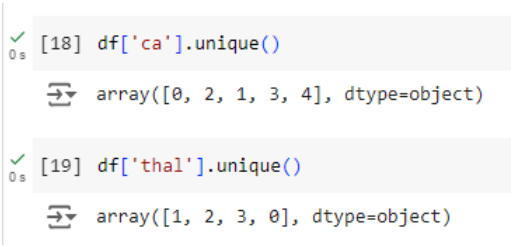


FIGURE 3.19 – Les valeurs de ca et thal

3- Gestion des donnes redondante :

Les données peuvent inclure des lignes répétées plusieurs fois.Ces lignes doivent être supprimées pour garantir la qualité des données. L'ensemble de données du Cleveland contenait une ligne redondante qui a été supprimée lors de la gestion des valeurs aberrantes.

4- Normalisation :

La normalisation des données est un aspect crucial dans les domaines de l'apprentissage automatique . Son rôle est d'ajuster les valeurs des caractéristiques numériques pour les mettre à la même échelle, ce

qui simplifie les opérations et contribue à améliorer l'efficacité des modèles en minimisant les disparités de magnitude entre les différentes variables.Parmi les techniques couramment utilisées pour la normalisation, on trouve le MinMaxScaler, le RobustScaler et la standardisation.Selon nos exigences.Dans notre ensemble de données, nous avons appliqué deux techniques de normalisation adaptées à chaque type de caractéristique présente :

— **Robust Scaling :**Nous avons utilisé RobustScaler pour les attributs **'trestbps'** , **'chol'**, **'thalach'**, et**'oldpeak'**.L'avantage de cette méthode réside dans sa capacité à redimensionner les caractéristiques en utilisant la médiane et l'intervalle interquartile (IQR), ce qui la rend moins sensible aux valeurs aberrantes [72].

$$X_{\text{scaled}} = \frac{X - X_{\text{median}}}{\text{IQR}} \tag{3.1}$$

— **Standardization (Z-score Scaling) :**Nous avons utilisé la standardisation pour les autres attributs restants tels que l'âge, fbs et exang ....ect. Cette technique transforme les caractéristiques pour avoir une moyenne nulle et une variance unitaire, assurant ainsi une mise à l'échelle uniforme et facilitant les comparaisons entre les différentes attributs[72] .

$$X_{\text{scaled}} = \frac{X - X_{\text{mean}}}{X_{\text{std}}} \tag{3.2}$$

3.5.2 Prétraitement de deuxième ensemble de données :

1- Gestion des données manquantes :

La figure 3.20 montre que ensemble de données ne contient aucune valeur manquante.

```
[ ] df.isnull().sum()

Age              0
Sex              0
ChestPainType    0
RestingBP        0
Cholesterol      0
FastingBS        0
RestingECG       0
MaxHR            0
ExerciseAngina   0
Oldpeak          0
ST_Slope         0
HeartDisease     0
dtype: int64
```

FIGURE 3.20 – Les valeurs manquantes (NaN).

2- Gestion des données redondantes :

La figure 3.21 montre qu'il n'y a pas de lignes dupliquées.

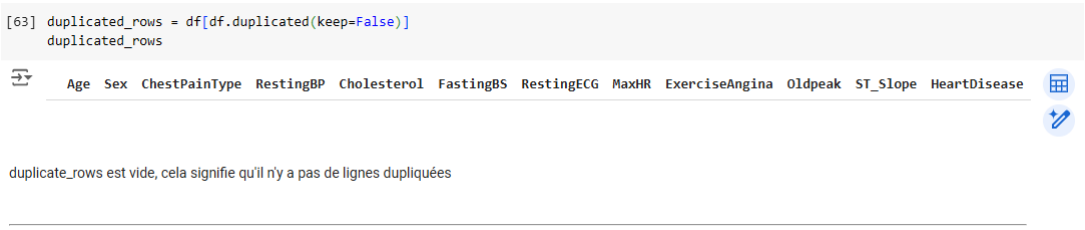


FIGURE 3.21 – les valeurs dupliquées.

3- Normalisation des données :

- **Transformation des valeurs catégoriques à des valeurs numériques :** Nous utilisons un encodeur d'étiquettes (LabelEncoder) de la bibliothèque scikit-learn pour convertir des colonnes catégorielles en valeurs numériques. La figure 3.22 illustre l'implémentation de cette technique.

```
[33] label_encoder = LabelEncoder()
     for col in categorical_cols:
         X_train[col] = label_encoder.fit_transform(X_train[col])
         X_test[col] = label_encoder.transform(X_test[col])
```

FIGURE 3.22 – Label Encoder

- **Mise à l'échelle des données :** De la même manière pour le premier ensemble de données, nous avons utilisé RobustScaler pour les attributs **'Age'** , **'RestingBP'**, **'Cholesterol'**, **'Oldpeaket''MaxHR'**.Et pour les autres attributs restants, nous avons appliqué la standardisation (StandardScaler).La figure 3.23 illustre l'implémentation de ces techniques de normalisation.

```
# Normaliser les fonctionnalités continues avec RobustScaler
df[continuous_features] = robust_scaler.fit_transform(df[continuous_features])
# Standardiser les fonctionnalités catégorielles avec StandardScaler
categorical_features_to_normalize = [col for col in categorical_features if col != 'Sex']
#df[categorical_features_to_normalize] = std_scaler.fit_transform(df[categorical_features_to_normalize])
df[categorical_features] = std_scaler.fit_transform(df[categorical_features])
print(df)
return df
```

FIGURE 3.23 – La normalisation des attributs catégoriels et continues

3.6 La sélection d'attributs

Notre objectif dans cette recherche est de repérer les attributs les plus performantes et d'éliminer celles qui ne contribuent pas de manière significative aux performances des modèles prédictifs. Les algorithmes

génétiques sont couramment utilisés pour cette tâche, mais ils souffrent de problèmes que nous avons abordés dans le chapitre 2. Pour remédier à ces limitations, nous avons utilisé une nouvelle approche de sélection appelée algorithme génétique rival , qui résout les problèmes liés aux AG en introduisant des stratégies de compétition et de croisement améliorées ainsi qu'un taux de mutation dynamique.

3.6.1 La sélection d'attributs par l'algorithme génétique

Le principe de fonctionnement de l'algorithme génétique est déjà expliqué dans le chapitre 2.Les étapes de l'AG est la suivante :

— **Étape 1 : Codage**

Nous avons choisi d'utiliser le codage binaire, où 0 représente un attribut non sélectionné et 1 représente un attribut sélectionné.

— **Étape 2 : Initialisation de la population :**

Nous travaillons sur deux ensembles de données : le premier comportant 13 attributs et le deuxième 11 attributs.Nous avons fixé la taille de la population à N=15.

— **Étape 3 : Calcule de la fonction d'évaluation (fitness) :**

La fonction fitness est cruciale dans les algorithmes génétiques. Dans cette expérimentation, la fitness est évaluée en utilisant MLPClassifier et la validation croisée.

La métrique utilisée est le f1_macro , qui est adapté pour les ensembles de données déséquilibrés.Les tableaux 3.3 et 3.4 illustrent des exemples d' evaluation de la fitness de la population initiale pour le premier et le deuxième ensemble de données.

Individu	**Chromosome**	**Codage Binaire**	**Fitness**
1	[age ,sex , restecg ,thalach, exang, oldpeak, slope, ca]	1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0	0.8211579243950293
2	[sex ,cp ,chol , restecg ,thalach ,oldpeak, slope ,ca ,thal]	0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1	0.8509992291863734
3	[sex, cp , fbs ,restecg, thalach ,exang, slope, ca ,thal]	0, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 1, 1	0.8526164752626946
4	[sex, cp , fbs, restecg, thalach ,exang, oldpeak, ca, thal]	0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1	0.8414163297575483

TABLE 3.3 – Population initiale avec leur fitness pour le premier ensemble de données

Individu	Chromosome	Codage Binaire	Fitness
1	[Sex ,ChestPainType ,RestingBP ,Cholesterol ,FastingBS, RestingECG, MaxHR ,ExerciseAngina , ST_Slope]	0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1	0.8489738626595889
2	[Age ,Sex , RestingBP , FastingBS ,RestingECG, MaxHR ,ExerciseAngina, Oldpeak, ST_Slope]	1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1	0.8188250247748797
3	[Sex, ChestPainType ,RestingBP ,Cholesterol ,FastingBS , MaxHR ,ExerciseAngina, Oldpeak ST_Slope]	0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1	0.8524775683748679
4	[Sex, ChestPainType ,RestingBP ,Cholesterol, FastingBS, RestingECG ,MaxHR ,ExerciseAngina , ST_Slope]	0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1	0.8511796723861748

TABLE 3.4 – Population initiale avec leur fitness pour la deuxième ensemble de données

— **Étape 4 : Sélection**

Nous avons choisi d'utiliser la méthode de sélection par roulette. Voici comment cette sélection est mise en œuvre :

1. Calculer la somme de toutes les valeurs de fitness dans la population.
2. Calculer les valeurs de fitness normalisées en divisant chaque valeur de fitness par la somme totale des fitness.
3. Calculer les valeurs de fitness cumulées.
4. Générer un nombre aléatoire p dans l'intervalle [0, 1].
5. Sélectionner l'individu pour lequel p est juste inférieur à la somme cumulée

— **Étape 5 : Croisement**

Dans les algorithmes génétiques, différentes techniques de croisement telles que le croisement à un point, à deux points et uniforme sont utilisées pour échanger du matériel génétique entre les parents. Dans ce travail , nous avons utilisé le croisement à deux points

Le croisement à deux points implique de choisir deux points aléatoires sur les chromosomes des parents et d'échanger les segments situés entre ces points. Ce processus est généralement associé à une probabilité de croisement, qui détermine la chance qu'une paire de parents soit sélectionnée pour ce processus. Cette probabilité de croisement est souvent fixée entre 0,65 et 0,80. Dans notre cas, nous utilisons une probabilité de croisement de 0,78.

— **Étape 6 : Mutation**

La mutation joue un rôle crucial pour maintenir la diversité génétique au sein de la population. Elle consiste généralement à modifier aléatoirement un bit d'un individu choisi au hasard

dans la population. Cette opération est essentielle pour introduire de la variabilité après le croisement génétique. La fonction 'mutation' est spécifiquement conçue pour ce faire en sélectionnant aléatoirement deux positions dans chaque individu de 'crossover_population' et en échangeant les valeurs correspondantes entre ces positions.

— **Étape 7 : Critère d'arrêt**

Le critère d'arrêt est une condition établie dans le domaine des algorithmes génétiques afin de déterminer quand l'algorithme doit arrêter de chercher de nouvelles solutions. Il est crucial d'assurer le contrôle de la convergence de l'algorithme et d'éviter qu'il ne continue à s'exécuter indéfiniment.Certains des critères d'arrêt sont listés ci-dessous :

- Le nombre limite de générations autorisées a été atteint.
- Absence de progrès dans la fitness du meilleur individu de la population.
- Maintien d'une limite supérieure à la variance des valeurs de fitness dans une population, il implique que cette variance ne doit pas dépasser un certain seuil.

Dans notre travail, nous avons utilisé le troisième critère comme un critère d'arrêt. Le tableau 3.5 et le tableau 3.6 Illustrent respectivement les paramètres d'exécution de l'algorithme génétique pour le premier et le deuxième dataset.

Nombre d'attributs	13 + cible
Taille de la population	15
Critère d'arrêt	0.00005
Nombre d'attributs à sélectionner	9

TABLE 3.5 – Paramètres d'exécution d'AG pour le dataset 1

Nombre d'attributs	11 + cible
Taille de la population	15
Critère d'arrêt	0.00005
Nombre d'attributs à sélectionner	9

TABLE 3.6 – Paramètres d'exécution d'AG pour le dataset 2

Les résultats d'exécution de l'AG pour le premier et le deuxième dataset sont illustrés respectivement dans les tableaux3.7 et 3.8.

Mesure	**Valeur**
Variance	$4.776756540557082 \times 10^{-5}$
Nombre de générations	2
AVG Fitness	0.8491693183929979
Attributs sélectionnés	sex, cp, fbs, restecg, thalach, exang, slope, ca, thal

TABLE 3.7 – Résultats d'exécution de l'algorithme génétique pour data1

Mesure	**Valeur**
Variance	$4.1891896721732326e \times 10^{-5}$
Nombre de générations	16
AVG Fitness	0.8423174131319234
Attributs sélectionnés	Sex, ChestPainType, RestingBP, FastingBS, RestingECG, MaxHR, ExerciseAngina, Oldpeak, ST_Slope

TABLE 3.8 – Résultats d'exécution de l'algorithme génétique pour data2

3.6.2 La sélection d'attributs par l'algorithme génétique rival

Le principe de fonctionnement de l'algorithme génétique rival est déjà expliqué dans le chapitre 2. Les étapes d'implémentation de l'AGR sont comme suit :

1. **codage :** Nous avons utilisé le codage binaire pour les chromosomes, où 1 représente un attribut sélectionné et 0 un attribut non sélectionné. La figure 3.24 illustre un exemple de chromosome.Dans cet exemple , les attributs sélectionnés sont : sex, cp, restecg, thalach, exang, oldpeak,slope, ca, et thal.

FIGURE 3.24 – Représentation de chromosome en binaire

2. **Initialisation de la population :** La présence d'une population importante ralentirait l'algorithme, il est donc crucial de sélectionner la taille de population la plus appropriée. Nous avons fixé N=10. La figure 3.25 et 3.26 illustrent un exemple de population initiale pour le premier et le deuxième ensemble de données.

```
Population:
Individual 1: [1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1]
Individual 2: [0, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1]
Individual 3: [1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0]
Individual 4: [1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1]
Individual 5: [1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1]
Individual 6: [0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0]
Individual 7: [1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1]
Individual 8: [1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1]
Individual 9: [1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0]
Individual 10: [0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1]
```

FIGURE 3.25 – Population initial de data1

```
Population:
Individual 1: [1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0]
Individual 2: [1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1]
Individual 3: [1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0]
Individual 4: [1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1]
Individual 5: [1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1]
Individual 6: [1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0]
Individual 7: [0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1]
Individual 8: [1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1]
Individual 9: [0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0]
Individual 10: [1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1]
```

FIGURE 3.26 – Population initial de data2

3. **Évaluation de la fonction objectif :**

Chaque individu est évalué en fonction de sa performance dans un modèle prédictif. Nous avons utilisé comme fonction fitness MLPClassifier car le MLP est capable d'apprendre des relations complexes dans les données.

Le F1-score est utilisé comme métrique pour évaluer les ensembles de données déséquilibrés. Comme nous travaillons sur deux ensembles de données, le premier dataset est moins déséquilibré tandis que le deuxième est déséquilibré. La sélection du F1-score, qui combine précision et rappel, permet d'avoir une vision équilibrée de la capacité du modèle à identifier de manière adéquate les classes positives et négatives.Les tableaux 3.9 et 3.10 illustrent des exemples d'évaluation de la fitness de la population initiale pour le premier et le deuxième ensemble de données.

Individu	**Chromosome**	**Codage Binaire**	**Fitness**
1	[age, sex, cp, trestbps, exang, oldpeak, slope, ca, thal]	1111000011111	0.8366
2	[sex, cp, restecg, thalach, exang, oldpeak, slope, ca]	0110001111111	0.8574
3	[age, sex, cp, chol, fbs, restecg, thalach, slope, ca]	1110111100110	0.8182
4	[age, sex, cp, chol, restecg, thalach, oldpeak, ca, thal]	1110101101011	0.8414

TABLE 3.9 – Exemples de la population initiale avec leur fitness pour le premier ensemble de données

Individu	**Chromosome**	**Codage Binaire**	**Fitness**
1	[Age, ChestPainType, RestingBP, Cholesterol, ExerciseAngina, Oldpeak, ST_Slope]	10111111010	0.8007
2	[Age, Sex, RestingBP, Cholesterol, ExerciseAngina, Oldpeak, ST_Slope, CA, Thal]	11010111111	0.8259
3	[Age, Sex, ChestPainType, RestingBP, Cholesterol, FastingBS, MaxHR, ST_Slope, CA]	11111110110	0.8060
4	[Age, Sex, ChestPainType, RestingBP, Cholesterol, ExerciseAngina, Oldpeak, ST_Slope, Thal]	11011111101	0.8115

TABLE 3.10 – Exemples de la population initiale avec leur fitness pour la deuxième ensemble de données

4. **Calcule dynamiquement du taux de mutation :** Nous avons expliqué en détail dans le chapitre 2 comment le taux de mutation est calculé. La fonction dynamic_mutation_rate calcule un taux de mutation dynamique basé sur la génération actuelle et le nombre maximum de générations. Cette fonction prend trois arguments : generation, max_generations et fitness_values, et elle retourne le mutation_rate.

   Exemple :

   — **Le nombre maximale de génération :**35.

   — **Nombre de chromosome :**10

   — **Fitness de population initial :** [0.8366130573122257, 0.8574237250102099, 0.8182483086225103, 0.841445502217471, 0.805133735487102, 0.7873179275182678, 0.8153133265671298, 0.8150521652843269, 0.7640092059583733, 0.8309337870252836]

   — **Formule de calcul de Favg :** La valeure de $(Favg)$ est calculée comme :

$$Favg = \frac{0.8366130573122257 + \ldots + 0.8309337870252836}{10} = 0.8209340430526986$$

   — **Calcul du taux de mutation (mutation_rate) pour la première génération :**Pour la première génération $(generation = 1)$, le taux de mutation est calculé comme :

$$mutation_rate = Favg - 1 \times \frac{Favg}{max_generations} = 0.8171$$

5. **La division de la population :** Nous avons créé une fonction **divide_population(population)** pour diviser la population en paires pour la compétition. Voici l'explication de l'implémentation de cette fonction :

   - **Entrée :**La fonction prend en entrée une liste nommée population, qui contient tous les individus de la population.
   - **Sortie :**Elle retourne une liste de couples (paires), où chaque couple est constitué de deux individus sélectionnés de la population. Si le nombre d'individus est impair, le dernier couple ne contiendra qu'un seul individu.

6. **Competition :** Une fois que la population est divisée en paires, nous procédons à la compétition qui consiste à comparer les individus par paires en fonction de leur fitness. Les individus ayant une meilleure fitness sont classés dans le groupe gagnant, tandis que les individus ayant une moins bonne fitness sont classés dans le groupe perdant. Voici comment cette fonction est mise en œuvre :

   - **Entrée :**La fonction **competition(couples, fitness_values)** reçoit deux listes en entrée. Un ensemble de couples (paires) d'individus.ainsi qu'une liste des valeurs de fitness qui correspondent à chaque individu de la population.
   - **Sortie :**Les listes des gagnants, des perdants et leurs valeurs de fitness correspondantes sont retournées par la fonction.
     — **winner_group**
     — **loser_group**

— **fitness_winner_group**

— **fitness_loser_group**

7. **Selection :** Après avoir divisé la population en paires et identifié les gagnants et les perdants, nous procédons à la sélection des parents pour la prochaine génération. La fonction **select_parents** reçoit en entrée **winner_group** ,**loser_group** ,**fitness_winner_group** et **fitness_loser_group** . Pour sélectionner les parents, elle utilise une méthode de sélection par roulette *(roulette_selection)*. Les deux premiers parents sont choisis parmi les gagnants en fonction de leur fitness respective, tandis que le troisième parent est sélectionné parmi les perdants. La fonction garantit que les parents sélectionnés sont distincts les uns des autres et de parent1. Si ce n'est pas le cas, de nouveaux parents sont choisis jusqu'à ce que cette condition soit remplie.

   — **Fonctionnement de la fonction roulette_selection(population, fitness_values) :**

   - La fonction commence par calculer la somme totale de toutes les valeurs de fitness de la population.
   - Normalisation des valeurs de fitness : Chaque valeur de fitness est ensuite divisée par la somme totale des valeurs de fitness pour obtenir des valeurs normalisées.Cela donne lieu à une distribution de probabilité cumulée où la somme de toutes les valeurs normalisées est égale à 1.
   - Calcul des valeurs cumulées de fitness
   - Créer un nombre aléatoire (random_number) compris entre 0 et 1, puis comparer ce nombre aléatoire aux valeurs cumulées du fitness. On choisit l'individu correspondant à l'intervalle dans lequel se situe le nombre aléatoire.

8. **Croisement :**Nous avons déjà expliqué le principe de fonctionnement du croisement dans le chapitre 2. Le croisement est effectué entre trois parents à l'aide de la fonction **crossover_parents(parent1, parent2, parent3)**. Cette fonction reçoit en entrée trois parents sélectionnés par la méthode de la roulette. Avant de procéder au croisement, elle vérifie si parent1, parent2 et parent3 sont des listes. Si ce n'est pas le cas, elle les convertit en listes contenant chacun leur parent respectif.

   — Chaque triplet de gènes (gene1, gene2, gene3) issus de parent1, parent2 et parent3 génère un nombre aléatoire r1 distribué de manière aléatoire entre 0 et 1. Selon la valeur de r1, on détermine le nouveau gène pour la nouvelle solution de la manière suivante :

   - Si r1 est inférieur ou égal à 1/3, le nouveau gène sera gene1.
   - Si r1 est strictement supérieur à 1/3 mais inférieur ou égal à 2/3, le nouveau gène sera gene2.
   - Sinon, si r1 est supérieur à 2/3, le nouveau gène sera gene3.

   — la fonction retourne la liste new_solution, qui contient les gènes de la nouvelle solution générée par croisement.

9. **Mutation :**L'opération de mutation est effectuée par une fonction **mutate(individual, mutation_rate)** qui reçoit en entrée le taux de mutation, calculé dynamiquement, et l'individu. Son fonctionnement est le suivant :

- Vérifier si individual est une liste, et sinon, la convertir en liste.
- Afin de pouvoir utiliser des opérations vectorielles plus efficaces, on convertit la liste individual en un tableau numpy.
- Nous générons un masque de mutation en créant un tableau de nombres aléatoires entre 0 et 1 de même longueur que 'individual.shape[0]'. Si ce masque est inférieur au taux de mutation, nous inversons les gènes sélectionnés par le masque de mutation (si le gène est 1, il devient 0, et s'il est 0, il devient 1).

Nous avons choisi cette méthode au lieu de déclarer une boucle et de générer à chaque fois un nombre aléatoire entre 0 et 1 pour vérifier et inverser, car la méthode utilisant numpy est plus efficace et simple.la figure 3.27 illustre un exemple de fonctionnement de la mutation.

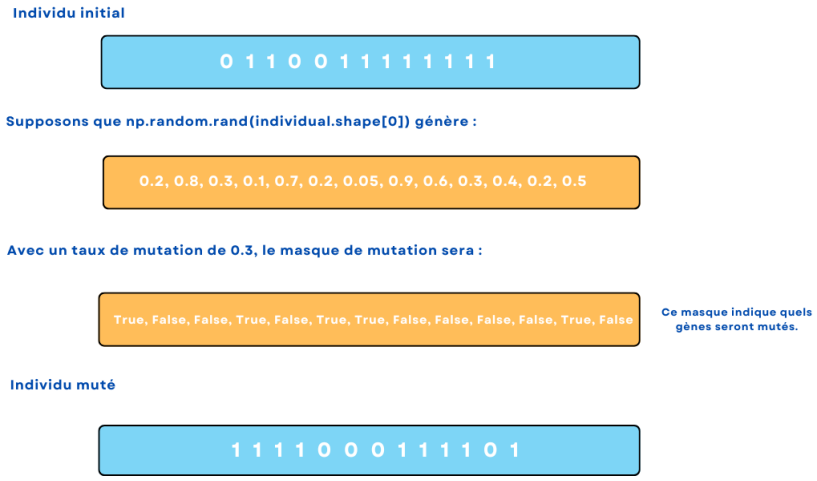


FIGURE 3.27 – Exemple de fonctionnement de mutation

10. **Critère d'arrêt :** Nous avons choisi un critère d'arrêt fixe avec max_generations = 35 pour cet algorithme, car nous avons besoin d'un nombre maximal de générations pour calculer dynamiquement le taux de mutation. Nous avons opté pour 35 générations spécifiquement car cela a entraîné des résultats améliorés pour notre problème.
11. **Évaluation de la fitness pour la nouvelle population mutée :** Une fois la mutation effectuée sur les individus sélectionnés, on évalue les fitness des nouveaux individus en utilisant la fonction get_fitness.
12. **Fusion des populations :**Cela consiste à combiner la population originelle et population mutée.On regroupe également les valeurs de fitness de ces deux populations dans une seule liste.
13. **Tri de la population fusionnée :**On classe la population fusionnée en fonction de son niveau de fitness décroissant. On effectue cela en utilisant la fonction zip afin d'associer chaque individu à sa valeur de fitness, ce qui permet de trier les deux listes en même temps.
14. **Sélection des meilleurs individus pour la prochaine génération :**Les meilleurs individus

sont sélectionnés pour former la nouvelle génération. Seuls les premiers individus jusqu'à la limite de population_size sont choisis pour maintenir la taille de la population.

15. **Mise à jour de la solution globale la plus performante :**Si la valeur de fitness actuelle est supérieure à la valeur de fitness globale enregistrée, cette valeur et l'individu correspondant sont mis à jour.

Le pseudocode de l'algorithme est comme suit :

Algorithm 1 Algorithme génétique rival pour la sélection d'attributs

Require: *heart_data*, caractéristiques (*features*), cible (*target*), taille de la population (*population_size*), *selected_features*

Ensure: Meilleur chromosome et meilleure fitness globale

```
1: 1. Initialiser la population :
2: function INIT_POPULATION(population_size, c, selected_features)
3:     Initialiser une population vide
4:     while taille de la population < population_size do
5:         Créer un individu avec des gènes aléatoires
6:         if l'individu est unique then
7:             Ajouter l'individu à la population
8:         end if
9:     end while
10: end function
11: 2. Calculer la fitness :
12: function CALCULATE_FITNESS(features, target)
13:     Utiliser le modèle MLP
14:     return Le score moyen de la validation croisée
15: end function
16: 3. Obtenir la fitness de la population :
17: function GET_FITNESS(population, heart_data)
18:     for chaque individu dans la population do
19:         Créer un sous-ensemble de données basé sur les gènes de l'individu
20:         Calculer la fitness de cet individu
21:         Ajouter la fitness à la liste des valeurs de fitness
22:     end for
23:     return La liste des valeurs de fitness
24: end function
25: 4. Diviser la population en couples :
26: function DIVIDE_POPULATION(population)
27:     Diviser la population en couples pour la compétition
28:     return La liste des couples
29: end function
```

5. Compétition :

function COMPETITION($couples, fitness_values$)

    Initialiser des listes vides pour les groupes de gagnants et de perdants

    for chaque couple dans les couples **do**

        Comparer les valeurs de fitness et sélectionner les gagnants et les perdants

    end for

    return Les groupes de gagnants et de perdants avec leur fitness

end function

6. Sélection des parents :

function SELECT_PARENTS($winner_group, loser_group, fitness_winner_group, fitness_loser_group$)

    6. Sélection des parents :

    function SELECT_PARENTS($winner_group, loser_group, fitness_winner_group, fitness_loser_group$)

        Sélectionner deux parents du groupe des gagnants et un du groupe des perdants en utilisant la méthode de la roulette.

        Assurer que les parents sont distincts

        return Les trois parents sélectionnés

    end function

    7. Croisement des parents :

    function CROSSOVER_PARENTS($parent1, parent2, parent3$)

        Initialiser une nouvelle solution vide

        for chaque gène $gene1, gene2, gene3$ dans $parent1, parent2, parent3$ respectivement **do**

            Générer un nombre aléatoire $r1$ uniformément distribué entre 0 et 1

            if $r1 \leq \frac{1}{3}$ **then**

                $new_gene \leftarrow gene1$

            else if $\frac{1}{3} < r1 \leq \frac{2}{3}$ **then**

                $new_gene \leftarrow gene2$

            else

                $new_gene \leftarrow gene3$

            end if

            Ajouter le gène sélectionné à la nouvelle solution

        end for

        return La nouvelle solution

    end function

    8. Calculer le taux de mutation dynamique :

    function DYNAMIC_MUTATION_RATE($generation, max_generations, fitness_values$)

        Calculer le taux de mutation basé sur la génération actuelle et les valeurs de fitness

        return Le taux de mutation

    end function

9. Appliquer la mutation :
function MUTATE(*individual*, *mutation_rate*)
    Convertir la liste en un tableau numpy pour utiliser les opérations vectorielles
    Générer un masque de mutation
    Inverser les gènes sélectionnés par le masque de mutation
    Reconvertir le tableau numpy en liste avant de le retourner
    return L'individu muté
end function
10. Algorithme principal :
Initialiser la population
Définir *max_generations*
Initialiser les meilleures valeurs globales
for chaque génération **do**
    Calculer la fitness de la population
    Calculer le taux de mutation dynamique
    Diviser la population en couples
    Réaliser la compétition
    Sélectionner les parents
    Réaliser le croisement
    Appliquer la mutation
    Évaluer la nouvelle population
    Fusionner et trier la population pour la prochaine génération
    Mettre à jour les meilleures valeurs globales
end for
return Le meilleur chromosome et la meilleure fitness globale

Les paramètres clés d'exécution de l'AGR pour le premier et le deuxième ensembles de données sont illustrés respectivement dans les tableau 3.11 et le tableau 3.12.

Nombre d'attributs	13 + cible
Taille de la population	10
Critère d'arrêt	35 générations
Nombre d'attributs à sélectionner	9

TABLE 3.11 – Paramètres d'exécution de l'AGR pour le dataset1

Nombre d'attributs	11 + cible
Taille de la population	10
Critère d'arrêt	35 générations
Nombre d'attributs à sélectionner	9

TABLE 3.12 – Paramètres d'exécution de l'AGR pour le dataset2

Les attributs sélectionnées par l'AGR, pour le premier et le deuxième ensemble de données, sont illustrés dans les tableaux 3.13et 3.14.

Mesure	**Valeur**
AVG Fitness	0.8682588709340111
Attributs sélectionnés	'sex', 'cp', 'restecg', 'thalach', 'exang', 'oldpeak', 'slope', 'ca', 'thal'

TABLE 3.13 – Rrésultats de l'exécution de AGR pour premier ensemble de données

Mesure	**Valeur**
AVG Fitness	0.8672956916295312
Attributs sélectionnés	'Sex', 'ChestPainType', 'RestingBP', 'Cholesterol', 'FastingBS', 'RestingECG', 'ExerciseAngina', 'Oldpeak', 'ST_Slope'

TABLE 3.14 – Résultats de l'exécution de AGR pour deuxième ensemble de données

3.7 Classification des données

Cette phase consiste à construire un modèle de prédiction des maladies cardiaques. Un score d'évaluation sera calculé pour chaque patient. Nos ensembles de données sont divisés en deux sous-ensembles : un ensemble d'entraînement et un ensemble de test. Dans cette approche, les données nettoyées et normalisées sont réparties aléatoirement en 80% pour l'entraînement et 20 % pour le test afin de former et tester plusieurs classificateurs supervisés tels que DNN, SimpleRNN, LSTM, GRU, SVM, KNN, XGBoost, DT, BN, RF et MLPClassifier. Le processus de classification se fait de trois manières : soit en classifiant les données sans sélectionner les attributs pertinents, soit en utilisant un algorithme génétique pour

sélectionner les meilleurs attributs, soit en utilisant un algorithme génétique rival pour la sélection des attributs pertinents. La technique BayesSearchCV est utilisée pour sélectionner les meilleures valeurs des hyperparamètres des modèles Simple RNN et GRU, tandis que les autres modèles utilisent GridSearchCV.

3.7.1 Hyperparamètres des modèles implémentés

Dans cette section, nous présentons les différents hyperparamètres utilisés pour optimiser les modèles de ML, les architectures RNN (SimpleRNN, LSTM, GRU),BN et les architectures DNN et MLPClassifier.

1. **Hyperparemtre de ML**

Modèle	**Hyperparamètres**
SVM	C=0.1, gamma='scale', kernel='sigmoid'
KNN	metric='manhattan', n_neighbors=16,weights='distance', algorithm='auto', p=1
XGBoost	use_label_encoder=False, eval_metric='logloss',gamma= 1, learning_rate=0.2,max_depth= 3, n_estimators= 100, reg_alpha= 0,reg_lambda= 100
DT	criterion='gini',max_depth=10
RF	criterion='gini', max_depth=None, n_estimators= 200

TABLE 3.15 – Hyperparamètres des modèles ML

2. **Hyperparamètres de MLPClassifier :**

Hyperparamètre	**Valeur**
`alpha`	0.0001
`hidden_layer_sizes`	(110, 41, 136, 118)
`learning_rate_init`	0.0001
`max_iter`	150
`Optimiseur`	adam

TABLE 3.16 – Hyperparamètres du modèle MLPClassifier

3. **Hyperparamètres du modèle DNN :**

Hyperparamètre	**Valeur**
`Taux de dropout`	0.3
`hidden_layer_sizes`	(147, 116, 51)
`weight _decay`	0.01
`Taux d'apprentissage`	0.01
`Optimiseur`	adam

TABLE 3.17 – Hyperparamètres du modèle DNN

4. **Hyperparamètres des architectures RNN :**

Architecture	**Couches**	**Optimiseur**	**Taux de dropout**	**Taux d'apprentissage**	**Pénalité L2**	**Unités par couche**
SimpleRNN	3	adam	0.5	0.001	0.001	60, 123, 85
GRU	4	sgd	0.5	0.1	0.01	188, 155, 106, 106
LSTM	2	adam	0.3	0.01	0.001	31, 147

TABLE 3.18 – Hyperparamètres des architectures RNN utilisées

5. **Réseau Bayésien :**

Hyperparamètre	**Description**
`scoring_method`	Méthode utilisée pour évaluer la qualité de la structure du réseau est BIC Score.
`search_algorithm`	Algorithme utilisé pour rechercher la structure du réseau est HillClimbSearch.
`estimator`	Estimateur utilisé pour apprendre les CPDs est Maximum Likelihood Estimator.

TABLE 3.19 – Hyperparamètres de BN

3.8 Conclusion

Dans ce chapitre, nous avons présenté certaines recherches effectuées précédemment dans le domaine de la prédiction des maladies cardiovasculaires. Nous avons défini l'architecture et le processus global de construction d'un modèle de prédiction. Nous avons visualisé nos données et effectué leur prétraitement. Ensuite, nous avons détaillé notre approche proposée pour résoudre la problématique de notre étude, en explicitant ses différentes étapes. Dans le prochain chapitre, nous aborderons les résultats obtenus.

Chapitre 4

Résultats expérimentaux

4.1 Introduction

Dans ce chapitre, nous présentons et comparons les résultats expérimentaux du modèle proposé, MLPClassifier, avec ceux d'autres classificateurs, notamment DNN, SimpleRNN, LSTM, GRU, SVM, KNN, XGBoost, DT, BN, et RF, sur deux ensembles de données. Les comparaisons sont effectuées dans trois catégories : sans sélection, avec AG et avec AGR.

Les performances de ces modèles sont évaluées à l'aide de plusieurs métriques d'évaluation, telles que la courbe ROC,accurecy, la précision, le rappel, le F1 score, ainsi que les métriques d'erreur MAE et RMSE . En outre, nous utilisons l'analyse SHAP pour interpréter les contributions des caractéristiques dans la prédiction de modèle MLPClassifier.

4.2 Évaluation des performances des modèles pour les données de Cleveland

Les résultats de comparaison des métriques d'évaluation du modèle MLPClassifier par rapport aux autres modèles sont illustrés dans le tableau4.1 et la figure 4.1, à la fois sans et avec sélection par AGR et AG.

Modèle	**Acc**			**Pre**			**Rap**			**FM**		
	13	**AG**	**AGR**	**13**	**AG**	**AGR**	**13**	**AG**	**AGR**	**13**	**AG**	**AGR**
SVM	0.833	0.915	0.915	0.825	0.909	0.909	0.916	0.937	0.937	0.868	0.923	0.923
KNN	0.850	0.864	0.881	0.829	0.875	0.857	0.944	0.875	0.937	0.883	0.875	0.895
XGBoost	0.833	0.915	0.932	0.842	0.909	0.937	0.888	0.937	0.937	0.864	0.923	0.937
DT	0.733	0.745	0.762	0.812	0.774	0.821	0.722	0.750	0.718	0.764	0.761	0.766
RF	0.866	0.864	0.881	0.868	0.852	0.857	0.916	0.906	0.937	0.891	0.878	0.895
BN	0.783	0.796	0.796	0.714	0.697	0.697	0.892	0.920	0.920	0.793	0.793	0.793
GRU	0.833	0.881	0.915	0.842	0.903	0.909	0.889	0.875	0.938	0.865	0.889	0.923
LSTM	0.817	0.864	0.915	0.879	0.853	0.935	0.806	0.906	0.906	0.841	0.879	0.921
SimpleRNN	0.867	0.881	0.915	0.868	0.857	0.909	0.917	0.938	0.938	0.892	0.896	0.923
DNN	0.866	0.915	0.932	0.888	0.935	0.937	0.888	0.906	0.937	0.888	0.920	0.937
MLPClassifier	0.85	0.915	0.949	0.856	0.915	0.95	0.85	0.915	0.949	0.85	0.915	0.949

TABLE 4.1 – Évaluation des performances des différents modèles

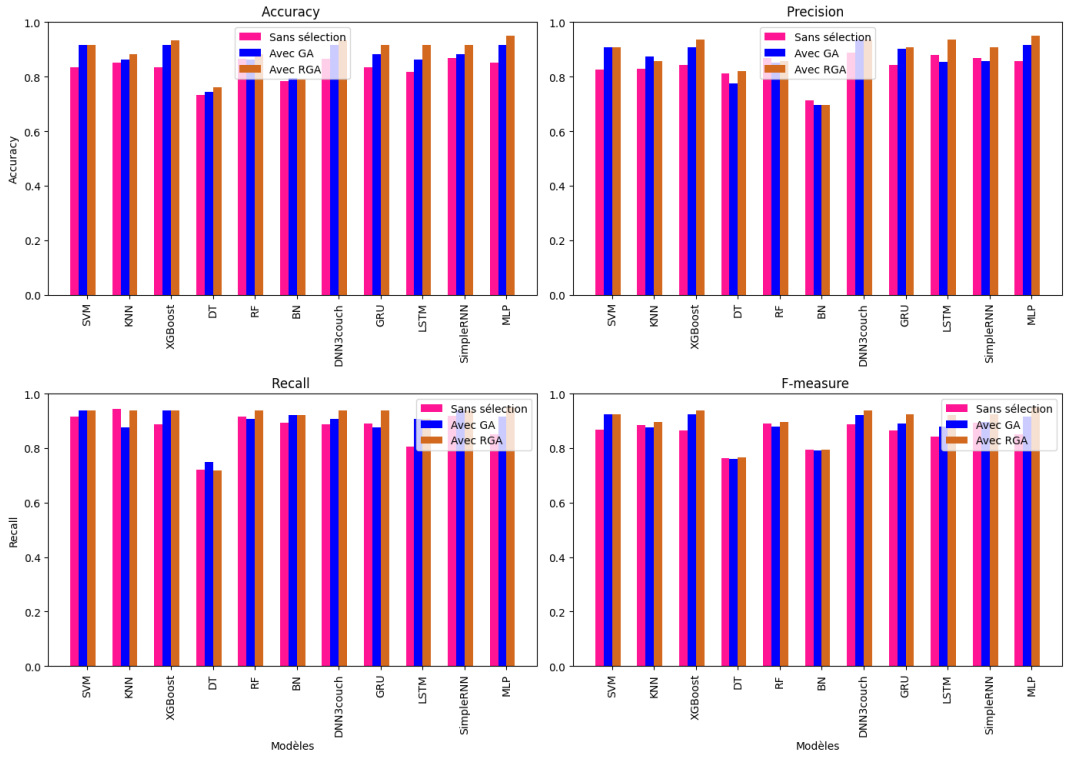


FIGURE 4.1 – Visualisation des métriques d'evaluation

D'après l'illustration graphique et le tableau de comparaison précédents, plusieurs éléments significatifs

peuvent être observés concernant les performances des différents modèles :

— Le modèle MLPClassifier avec sélection par AGR présente des performances supérieures aux autres modèles pour les différentes métriques d'évaluation, mettant ainsi en évidence son efficacité et sa robustesse pour le premier ensemble de données.

— Pour SVM et BN, les performances restent similaires, que ce soit avec AG ou AGR.

— Les performances du modèle XGBoost avec l'algorithme AGR sont les plus élevées par rapport aux autres modèles d'apprentissage automatique.

— Les performances de presque tous les modèles XGBoost, KNN, DT, RF, GRU, LSTM, DNN , SimpleRNN, sont généralement améliorées par la sélection de caractéristiques par AGR par rapport à la sélection par AG et à l'absence de sélection. Cette observation justifie l'utilisation de l'algorithme AGR pour optimiser les modèles.

4.3 Évaluation des performances des modèles pour les données de Heart Failure

Les résultats de la comparaison des métriques d'évaluation du modèle MLP avec d'autres modèles sont illustrés dans le tableau 4.2 et la figure 4.2, aussi bien sans sélection qu'avec sélection par AGR et AG.

Modèle	**Acc**			**Pre**			**Rap**			**FM**		
	11	**AG**	**AGR**	**11**	**AG**	**AGR**	**11**	**AG**	**AGR**	**11**	**AG**	**AGR**
SVM	0.847	0.873	0.890	0.831	0.887	0.890	0.884	0.878	0.890	0.857	0.883	0.890
KNN	0.860	0.895	0.912	0.875	0.908	0.903	0.875	0.899	0.923	0.875	0.903	0.913
XGBoost	0.847	0.857	0.884	0.819	0.854	0.857	0.905	0.888	0.923	0.860	0.871	0.888
DT	0.766	0.807	0.835	0.760	0.807	0.858	0.800	0.848	0.802	0.779	0.827	0.829
RF	0.853	0.895	0.890	0.840	0.884	0.881	0.884	0.929	0.901	0.861	0.906	0.891
BN	0.820	0.879	0.890	0.818	0.896	0.900	0.887	0.896	0.900	0.852	0.896	0.900
GRU	0.859	0.885	0.885	0.848	0.890	0.865	0.884	0.899	0.912	0.883	0.894	0.888
LSTM	0.815	0.874	0.912	0.835	0.880	0.921	0.800	0.889	0.901	0.817	0.884	0.911
SimpleRNN	0.870	0.863	0.879	0.859	0.894	0.871	0.895	0.848	0.890	0.876	0.870	0.880
DNN	0.842	0.862	0.906	0.817	0.870	0.870	0.894	0.878	0.956	0.854	0.874	0.911
MLP	0.826	0.890	0.901	0.827	0.890	0.901	0.826	0.890	0.901	0.825	0.889	0.901

TABLE 4.2 – Évaluation des performances des différents modèles

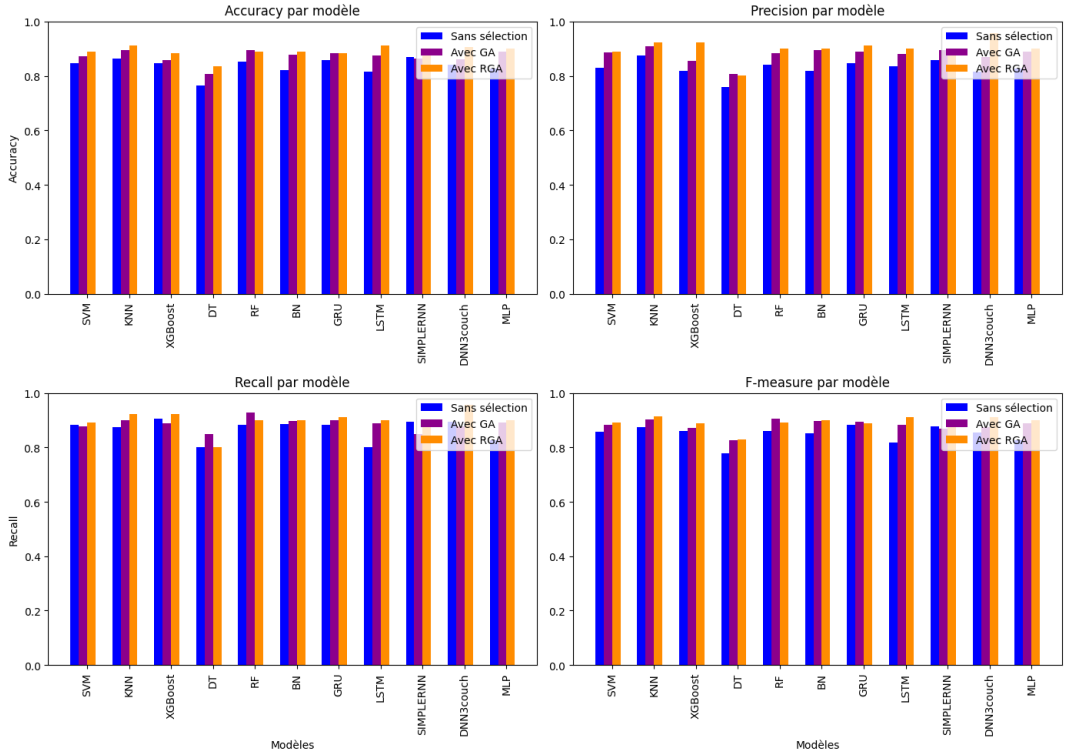


FIGURE 4.2 – Visualisation des métriques d'evaluation

D'après l'illustration graphique et le tableau de comparaison précédents, plusieurs éléments significatifs peuvent être observés concernant les performances des différents modèles :

— Les modèles de réseaux de neurones montrent des améliorations diverses après sélection par AGR. Notamment, MLP présente des performances stables avec des valeurs de précision, rappel et F-mesure identiques après AGR.

— LSTM présente les meilleures performances globales parmi les modeles SimpleRNN,GRU,DNN.

— KNN présente les meilleures performances globales parmi les autres modèles d'apprentissage automatique après sélection par AGR.

4.4 Évaluation des métriques d'erreurs des modèles pour les données de Cleveland

Les résultats de comparaison des métriques d'erreur du modèle MLP par rapport aux autres modèles sont illustrés dans le tableau4.3 et la figure 4.3, à la fois sans et avec sélection par AGR et AG.

Modèle	**MAE (13/AG/AGR)**	**RMSE (13/AG/AGR)**
SVM	0.1667/0.0847/0.0847	0.4082/0.291/0.291
KNN	0.15/0.135/0.1186	0.3873/0.3682/0.3444
XGBoost	0.1667/0.0847/0.0678	0.4082/0.2911/0.2604
DT	0.2667/0.2542/0.2373	0.5164/0.5042/0.4871
RF	0.133/0.1356/0.1186	0.3651/0.3682/0.3444
GRU	0.167/0.119/0.102	0.340/0.335/0.320
LSTM	0.183/0.136/0.085	0.364/0.330/0.313
SimpleRNN	0.133/0.119/0.085	0.350/0.350/0.323
DNN	0.133/0.085/0.0678	0.3651/0.2911/0.2604
MLP	**0.15/0.0847/0.0508**	**0.387/0.291/0.225**

TABLE 4.3 – Métriques d'erreur MAE et RMSE des différents modèles

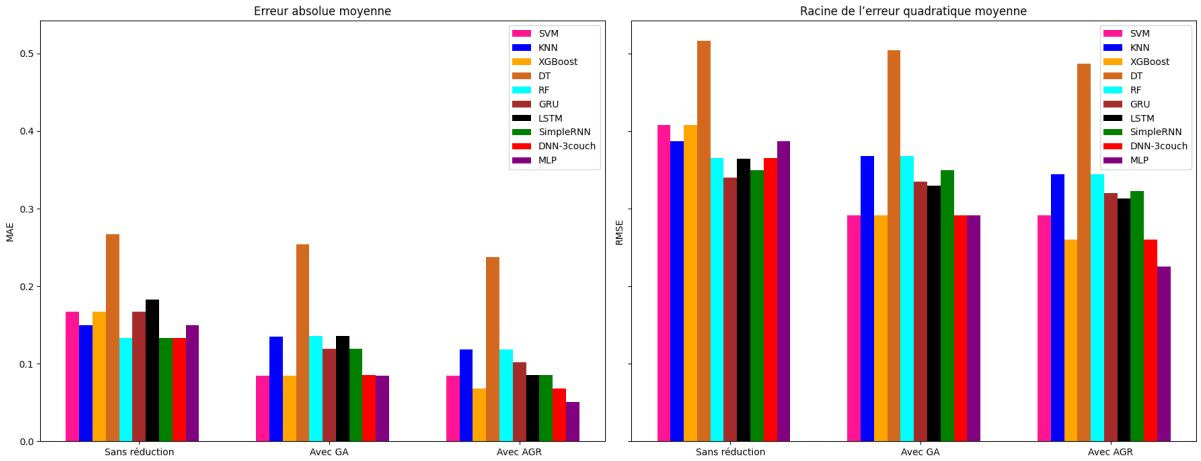


FIGURE 4.3 – Visualisation des métriques d'erreur MAE et RMSE

Selon l'illustration graphique et le tableau de comparaison ci-dessus, plusieurs observations sont à noter :

— Le modèle proposé MLPClassifier montre une amélioration notable avec l'utilisation de l'AGR.

— Une réduction significative du RMSE et du MAE est observée, en particulier pour MLP, DNN.

— Des erreurs plus élevées sont notées pour le DT , suggérant qu'il pourrait être moins efficace que les autres méthodes pour ce jeu de données.

— L' AGR se révèle particulièrement efficace pour améliorer la performance des modèles, en réduisant les erreurs de manière plus significative que AG pour tous les modèles, que ce soit en termes de RMSE ou de MAE.

4.5 Évaluation des métriques d'erreurs des modèles pour les données de Heart Failure

Les résultats de comparaison des métriques d'erreur du modèle MLP par rapport aux autres modèles sont présentés dans le tableau 4.4 et la figure 4.4, à la fois sans et avec sélection par AGR et AG.

Modèle	**MAE (11/AG/AGR)**	**RMSE (11/AG/AGR)**
SVM	0.152/0.126/0.331	0.390/0.355/ 0.109
KNN	0.135/ 0.104/0.087	0.368/ 0.323/0.296
XGBoost	0.152/ 0.142/0.115	0.390/0.378/0.339
DT	0.233/0.192/0.164	0.483/0.438 /0.406
RF	0.146 /0.104/0.109	0.383/0.323/0.331
GRU	0.141/0.115/0.115	0.341/0.308/0.315
LSTM	0.185/0.137/0.088	0.360/0.298/0.281
SimpleRNN	0.130/0.137/0.121	0.340/0.319/0.318
DNN	0.157/0.137/0.093	0.397/0.370/0.305
MLPClassifier	**0.173/0.109/0.098**	**0.417/ 0.331 /0.314**

TABLE 4.4 – Métriques d'erreur MAE et RMSE des différents modèles

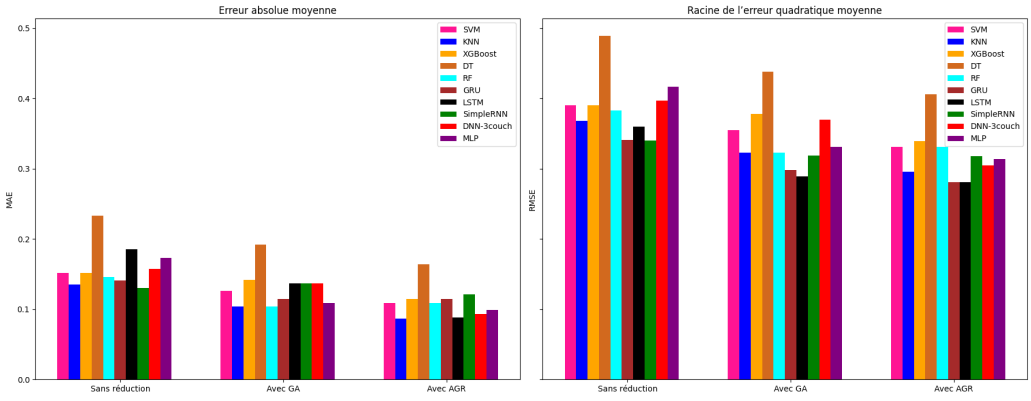


FIGURE 4.4 – Visualisation des métriques d'erreur MAE et RMSE

D'après le tableau 4.4 et le graphique 4.4 qui présentent les métriques d'erreur MAE et RMSE pour différents modèles avec les sélections par AG et AGR, voici les observations pour chaque modèle :

— MLP présente les performances les plus améliorées en termes d'erreur MAE et RMSE avec AGR, montrant une efficacité et une précision exceptionnelles.

— DNN montrent des réductions significatives de l'erreur MAE et RMSE après sélection par AGR, montrant une amélioration globale de la précision et de la robustesse des prédictions.

— La sélection par AGR améliore les modèles SVM, KNN, DT et RF en augmentant leurs capacités prédictives, avec des réductions significatives des erreurs et des performances consolidées pour XGBoost.

4.6 Comparaison des Travaux de Recherche

Le tableau 4.5 qui illustre la comparaison entre les travaux connexes et l'approche proposée

Travail de recherche	**Année**	**Sélection d'attributs**	**Modèle de prédiction**	**Accuracy**
J. Nageswara Rao, Sahithi Masimukku,et al[67]	2021	Non	- Decision Tree - Naïve Bayes - **Random Forest** - Logistic Regression - SVM	90,16%
A.Gupta et al [65]	2022	AG-RFE	AdaBoost	91%(Cleveland) 96% (Pima)
Fahad A. Alghamdi, Haitham Almanaseer, et al.[66]	2024	AOA	- MLPNN_APO -**MLPNN** - SVM - DT - KNN - Naive Bayes	88.89%
Aarti Karandikar, Komal Jaisinghani,et al.[68]	2024	Non	- **HCNN** - VGG16 - ResNet - LeNet	92%
Notre Approche	2024	AGR	- **MLPClassifier** - DNN - GRU - LSTM - simpleRnn - SVM - KNN - XGBoost - BN	0.949 (Cleveland) 0.901 (Heart Failure)

TABLE 4.5 – Analyse comparative des travaux connexes

4.7 Détails des performances du modèle MLP_AGR pour le premier ensemble de données

4.7.1 La matrice de confusion

La figure 4.5 illustre la matrice de confusion du modèle proposé MLP avec l'algorithme de sélection d'attributs AGR.

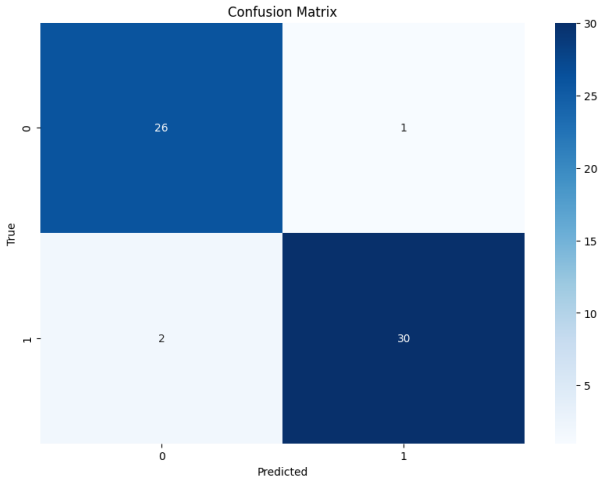


FIGURE 4.5 – La matrice de confusion

Interprétation de la matrice :

- **Vrais Négatifs (TN) :**Le modèle a correctement prédit que 26 patients n'ont pas de maladie cardiovasculaire.
- **Faux Positifs (FP) :**Le modèle a prédit à tort que 1 patient a une maladie cardiovasculaire alors qu'il n'en a pas.
- **Faux Négatifs (FN) :**Le modèle a erroné en prédisant que 2 patients ne présentent pas de maladie cardiovasculaire alors qu'ils en présentent.
- **Vrais Positifs(TP) :**Le modèle a correctement prédit que 30 patients ont une maladie cardiovasculaire.

Selon cette matrice de confusion, il est évident que le modèle MLP optimisé par AGR est relativement précis, avec un nombre limité de faux positifs et de faux négatifs. Cela signifie que le modèle est efficace pour prédire correctement la présence ou l'absence de maladie cardiovasculaire.

4.7.2 Le rapport de classification

Le rapport de classification présenté dans figure 4.6 est une évaluation détaillée des performances de MLP.

Classification Report:

	precision	recall	f1-score	support
0	0.93	0.96	0.95	27
1	0.97	0.94	0.95	32
accuracy			0.95	59
macro avg	0.95	0.95	0.95	59
weighted avg	0.95	0.95	0.95	59

FIGURE 4.6 – Classification report

Le modèle affiche une précision de 0.93 et un rappel de 0.96 pour la classe 0, avec une précision de 0.97 et un rappel de 0.94 pour la classe 1, aboutissant à une accuracy globale de 0.95. Les F1-scores pour les deux classes sont de 0.95, montrant une très bonne performance générale. Le support est de 27 pour la classe 0 et de 32 pour la classe 1.

4.7.3 Courbe ROC et AUC :

La courbe ROC du modèle MLPClassifier_AGR, utilisant la méthode de validation « split-validation » sur le premier ensemble de données, illustre le compromis entre le taux de vrais positifs et le taux de faux positifs à différents seuils de probabilité. Cette représentation est montrée dans la figure 4.7.

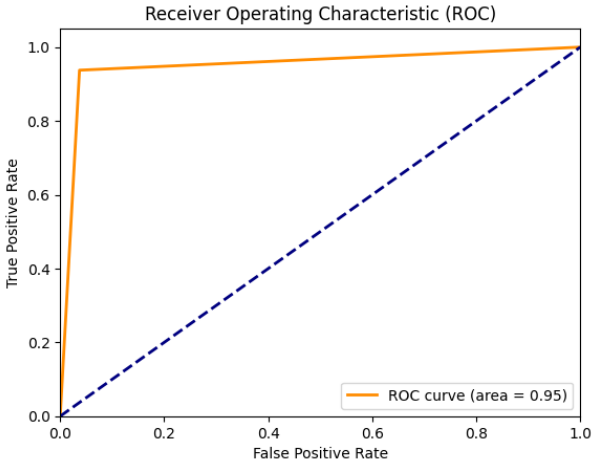


FIGURE 4.7 – Courbe ROC

L'aire sous la courbe ROC du modèle proposé avec la méthode de validation « split-validation » est de 0.95 (95%).

4.7.4 Explication du modèle à l'aide de SHAP :

La technique SHAP a été utilisée pour mieux comprendre le fonctionnement du modèle. Son objectif est d'attribuer une importance aux différentes caractéristiques d'entrée du modèle en évaluant comment chacune contribue aux prédictions finales. Dans cette section, nous discutons des résultats obtenus grâce à SHAP.

— **Importance des caractéristiques selon les valeurs SHAP :**

La figure 4.8 présente l'importance moyenne des caractéristiques du modèle, exprimée en valeurs SHAP. Ces valeurs représentent l'influence absolue moyenne d'une caractéristique sur les prédictions du modèle. La longueur de chaque barre indique l'importance relative de la caractéristique : plus elle est longue, plus la caractéristique est importante pour les prédictions du modèle.

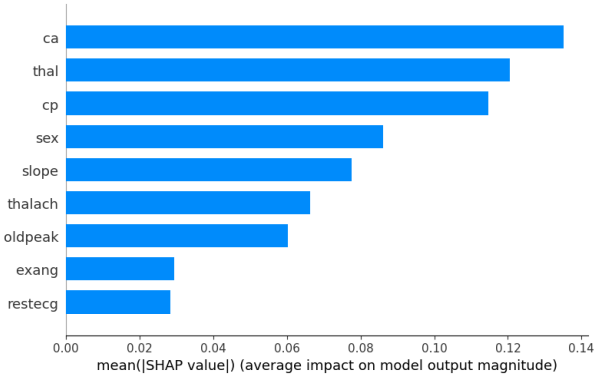


FIGURE 4.8 – Importance des caractéristiques selon les valeurs SHAP pour le modèle MLPClassifier_AGR

Voici une explication des résultats présentés dans la figure :

- **Ca** a la plus grande influence sur les prédictions de notre modèle.
- **thal** et **cp** suivent en termes d'importance.
- Les caractéristiques sex, slope, thalach, oldpeak, exang, et restecg ont un impact moindre mais toujours significatif.

— **Explication des valeurs SHAP pour les prédictions du modèle :**

Nous avons appliqué SHAP aux deux premières lignes de l'ensemble de test.

1. **Le résultat de la première ligne de l'ensemble de test (225) :**

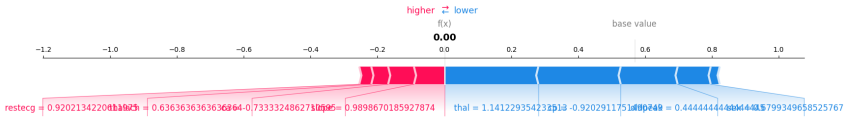


FIGURE 4.9 – SHAP pour la ligne 225

- **Valeur de base** : La valeur de base représente la référence de prédiction attendue. Dans la figure 4.9, la valeur de base est de 0,00, ce qui correspond à la sortie modèle attendue.
- **Impact des caractéristiques** :
  - **Valeurs SHAP positives (Rouge)** :Selon la figure 4.9, les attributs qui contribuent positivement à l'augmentation de la prédiction sont restecg, thalach, ca, et slope.
  - **Valeurs SHAP négatives (Bleu)** : D'après la figure 4.9, les attributs thal, cp,exang, oldpeak et sex ont un impact négatif en diminuant la prédiction.

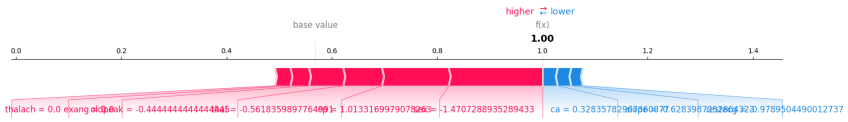


FIGURE 4.10 – SHAP pour la ligne 147

2. **Le résultat de la deuxième ligne de l'ensemble de test (147) :**

- **Valeur prédite :** La valeur prédite est 1.0.
- **Impact des caractéristiques :**
  - **Valeurs SHAP positives (Rouge)** :Selon la figure 4.9, les attributs qui contribuent positivement à l'augmentation de la prédiction sont thalach, exang, oldpeak, thal, sex et cp.
  - **Valeurs SHAP négatives (Bleu)** : D'après la figure 4.9, les attributs ca, slope et restecg ont un impact négatif en diminuant la prédiction.

4.7.5 Test du modèle prédictif proposé

Le tableau 4.6 illustre les résultats de prédiction avec le modèle prédictif proposé.Nous avons sélectionné de manière aléatoire 6 patients malades(3,7,13,20,30) et 5 patients non malades à partir de notre ensemble de données

Données des patients	Valeur réelle	Valeur prédite
P1 : [1, 2, 1, 187, 0, 3.5, 0, 0, 2]	1	1
P2 : [1, 0, 1, 148, 0, 0.4, 1, 0, 1]	1	1
P3 : [0, 2, 1, 139, 0, 0.2, 2, 0, 2]	1	1
P4 : [1, 0, 1, 171, 0, 1.5, 2, 0, 2]	1	1
P5 : [0, 2, 0, 157, 0, 0.8, 2, 1, 2]	1	1
P6 : [1, 1, 0, 143, 0, 0.0, 2, 0, 2]	1	1
P7 : [0, 0, 0, 133, 1, 4.0, 0, 2, 3]	0	0
P8 : [0, 0, 1, 123, 1, 0.2, 1, 0, 3]	0	0
P9 : [1, 0, 0, 131, 1, 2.2, 1, 3, 3]	0	0
P10 :[1, 0, 1, 95, 1, 2.0, 1, 2, 3]	0	0
P11 :[0, 0, 0, 136, 1, 3.0, 1, 0, 3]	0	0

TABLE 4.6 – Tester le modèle avec certains patients dans un ensemble de données

Selon le tableau 4.6, le modèle a correctement identifié les 6 patients présentant une maladie cardiaque et les 5 patients qui n'en avaient pas.

4.8 Conclusion

Dans ce quatrième et dernier chapitre, nous avons comparé les performances de plusieurs modèles SVM, KNN, XGBoost, DT, BN, RF, DNN , RNNSimple, GRU et LSTM, ainsi que MLPClassifier, dans trois cas : sans sélection, avec AG et avec AGR, pour les ensembles de données Cleveland et Heart Failure. L'objectif était de démontrer l'efficacité d'AGR par rapport à AG. En plus des métriques d'évaluation classiques, nous avons examiné en détail la matrice de confusion,le rapport de classification ,la courbe ROC et SHAP spécifiquement pour le modèle MLPClassifier de premier ensemble de données.

Enfin, nous avons testé notre modèle prédictif sur un échantillon de 11 patients choisis aléatoirement dans notre ensemble de données. Les résultats, évalués à l'aide de métriques d'erreur et d'évaluation, ont démontré que le MLPClassifier, lorsqu'il est associé à AGR, est à la fois performant et flexible. Bien que des modèles comme le DNN ainsi que XGBoost aient montré des performances exceptionnelles dans certains cas particuliers, nous avons retenu le MLPClassifier pour sa robustesse générale.

Conclusion générale et perspectives

Dans ce travail, nous avons introduit une nouvelle méthode de sélection appelée "algorithme génétique rival". Cette approche vise à améliorer les performances des algorithmes génétiques en accélérant leur convergence et en optimisant la qualité des solutions obtenues.

Nous avons travaillé avec deux ensembles de données : "Cleveland " et "Heart Failure ", qui contiennent des informations médicales sur les risques de maladies cardiaques.Après une analyse approfondie et un prétraitement des données, nous les avons préparées pour la phase d'entraînement en utilisant la méthode de validation split pour diviser nos données en ensembles d'entraînement et de test. Ensuite, nous avons appliqué divers algorithmes d'apprentissage automatique et profond, tels que KNN, SVM, XGBoost,DT, RF,BN,DNN , SimpleRNN,GRU, LSTM et MLPClassifier. Notre objectif était de sélectionner le modèle prédictif offrant la meilleure accuracy.

De plus, nous avons comparé l'algorithme génétique standard à une version améliorée, l'algorithme génétique rival . Nos résultats ont montré que RGA produisait de meilleurs résultats pour la sélection des attributs, ce qui conduisait à une amélioration significative de la précision des modèles. En particulier, MLPClassifier a été identifié comme offrant la meilleure précision, atteignant près de 95%, sur le premier ensemble de données.

Cette étude démontre l'efficacité et la flexibilité du MLPClassifier dans des contextes variés et souligne l'importance d'une sélection optimisée des attributs pour améliorer les performances des modèles prédictifs. Ainsi, le MLPClassifier, avec l'optimisation AGR, s'impose comme une solution robuste et fiable .

Pour les travaux futurs, nous envisageons d'explorer et d'implémenter des méthodes encore plus avancées pour améliorer les performances de nos modèles prédictifs. Une direction prometteuse serait de développer une version améliorée de l'algorithme génétique rival, nommée FRGA.

La version rapide de l'algorithme génétique rival (FRGA) améliore l'efficacité computationnelle de l'AGR en utilisant un schéma de croisement variable dans le temps. Ce schéma ajuste dynamiquement

le nombre de croisements au fil des générations, permettant ainsi de réduire la complexité computationnelle tout en maintenant un bon équilibre entre l'exploration de nouvelles solutions et l'exploitation des solutions prometteuses.

Ainsi, en poursuivant ces investigations, nous espérons contribuer à l'avancement des techniques de sélection d'attributs et à l'amélioration globale des performances des modèles .

Bibliographie

[1] Dr Abdel Latif. Fonctionnement du cœur, 2016. https://www.elhallak.fr/fonctionnement-du-coeur/,Consultéle 2024-06-11.

[2] Améli. Maladies cardiovasculaires, 2022. https://www.ameli.fr/assure/sante/themes/risque-cardiovasculaire/maladies-cardiovasculaires. Consulté le 2024-06-11.

[3] Assurance Maladie. Le risque cardiovasculaire et ses facteurs, 2022. https://www.ameli.fr/assure/sante/themes/risque-cardiovasculaire/definition-facteurs-favorisants#:~:text=La%20probabilit%C3%A9%20d'avoir%20un,estrog%C3%A8nes%20et%20progest%C3%A9rone%20)%20les%20prot%C3%A8gent., Consulté le 2024-02-01.

[4] Kenza Harifi. Comprendre l'algorithme des k-plus-proches voisins, 2019. https://medium.com/@kenzaharifi/bien-comprendre-lalgorithme-des-k-plus-proches-voisins-fonctionnement-et-impl%C3%A9mentation-sur-r-et-a66d2d372679, Consulté le 2024-03-10.

[5] Prashant Banerjee. Svm classifier tutorial, 2020. https://www.kaggle.com/code/prashant111/svm-classifier-tutorial, Consulté le 2024-03-10.

[6] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow : Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, 2017.

[7] Salah Ghamri. *Commande Neuronale d'un Robot Mobile Non-Holonome avec Glissement des Roues*. PhD thesis, 03 2016.

[8] Priya Pedamkar. Dnn neural network, 2023. https://www.educba.com/dnn-neural-network/, Consulté le 2024-03-11.

[9] GeeksforGeeks. Introduction to deep learning, 2024. https://www.geeksforgeeks.org/introduction-deep-learning/,Consulté le 2024-06-05.

[10] TANMAY UNHALE. Genetic algorithm for feature selection, 2021. https://www.kaggle.com/code/tanmayunhale/genetic-algorithm-for-feature-selection, Consulté le 2024-06-04.

[11] Musatafa Abbas Albadr, Sabrina Tiun, Masri Ayob, and Fahad Al-Dhief. Genetic algorithm based on natural selection theory for optimization problems. *Symmetry*, 12(11) :1758, 2020.

[12] Tuong Le, Mi Lee, Jun Park, and Sung Baik. Oversampling techniques for bankruptcy prediction : Novel features from a transaction dataset. *Symmetry*, 10 :79, 03 2018.

[13] Organisation mondiale de la santé. Maladies cardiovasculaires (mcv), 2021. https://www.who.int/fr/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds), Consulté le 2024-02-09.

[14] Organisation mondiale de la santé. Maladies cardiovasculaires (mcv). https://www.who.int/fr/health-topics/cardiovascular-diseases#tab=tab_1, Consulté le 2024-02-09.

[15] Jingwei Too and Abdul Rahim Abdullah. A new and fast rival genetic algorithm for feature selection. *The Journal of Supercomputing*, 77(3) :2844–2874, 2021.

[16] Fondation pour la Recherche Médicale. Recherches sur les maladies cardiovasculaires. https://www.frm.org/recherches-maladies-cardiovasculaires
Consulté le 2024-02-01.

[17] Les maladies cardiovasculaires, c'est quoi ? https://www.amgen.fr/aires-therapeutiques/maladies-cardiovasculaires/les_maladies_cardiovasculaires_cest_quoi,Consulté le 2024-02-01.

[18] Erich Schaffner. Structure de cœur. https://www.academia.edu/search?q=structure%20de%20coeur,Consultéle 2024-06-11.

[19] Fondation Hospices Civils de Lyon. Maladies cardiovasculaires : liste, symptômes et traitements du cœur. https://fondationhcl.fr/notre-action/recherche-et-innovation/informations-sur-les-maladies-et-la-recherche-medicale/maladies-cardiovasculaires-liste-symptomes-et-traitements-du-coeur/, Consulté le 2024-02-08.

[20] Khandaker Mohammad Mohi Uddin, Rokaiya Ripa, Nilufar Yeasmin, Nitish Biswas, and Samrat Kumar Dey. Machine learning-based approach to the diagnosis of cardiovascular vascular disease using a combined dataset. *Intelligence-Based Medicine*, 7 :100100, 2023.

[21] Institut de cardiologie de l'Université d'Ottawa. Maladie coronarienne. https://www.ottawaheart.ca/fr/maladie-du-c%C5%93ur/maladie-coronarienne-ath%C3%A9roscl%C3%A9rose, Consulté le 2024-02-05.

[22] Gouvernement du Canada. Signes et symptômes de l'accident vasculaire cérébral, 2018. https://www.canada.ca/fr/sante-publique/services/maladies/sante-coeur/accident-vasculaire-cerebral/signes-symptomes-accident-vasculaire-cerebral.html, Consulté le 2024-02-08.

[23] Medtronic. Insuffisance cardiaque. https://www.medtronic.com/fr-fr/patients/pathologies/insuffisance-cardiaque.html, Consulté le 2024-02-08.

[24] PasseportSanté. Myocardite. https://www.passeportsante.net/fr/Maux/Problemes/Fiche.aspx?doc=myocardite, Consulté le 2024-02-08.

[25] Les maladies cardiovasculaires. https://www.inspq.qc.ca/sites/default/files/responsabilite-populationnelle/f005_maladies_cardiovasculaires.pdf, Consulté le 2024-02-01.

[26] Chapitre 2 - item 222 : Facteurs de risque cardiovasculaire et prÉvention. https://www.sfcardio.fr/page/chapitre-2-item-222-facteurs-de-risque-cardiovasculaire-et-prevention, Consulté le 2024-02-01.

[27] Facteurs de risque cardiovasculaire et prévention. https://www.sfcardio.fr/sites/default/files/2019-11/2015-2e_Ref_Cardio_ch02_facteurs_cv.pdf, Consulté le 2024-02-05.

[28] Trend Micro. Qu'est-ce que l'apprentissage automatique. https://www.trendmicro.com/fr_fr/what-is/machine-learning.html,Consulté le 2024-06-05.

[29] Les types d'apprentissage automatique, 2024. https://course.elementsofai.com/fr/4/1,Consulté le 2024-06-05.

[30] IBM. Qu'est-ce que l'apprentissage supervisé, 2024. https://www.ibm.com/fr-fr/topics/supervised-learning,Consulté le 2024-06-05.

[31] Tavish Srivastava. Guide to k-nearest neighbors algorithm in machine learning, 2024. https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/#:~:text=The%20K%2DNearest%20Neighbors%20(KNN)%20algorithm%20is%20a%20popular,have%20similar%20labels%20or%20values, Consulté le 2024-03-10.

[32] DataScientest. Qu'est ce que l'algorithme knn, 2020. https://datascientest.com/knn, Consulté le 2024-03-10.

[33] Équipe Blent. Support vector machine (svm), 2022. https://blent.ai/blog/a/svm-support-vector-machine, Consulté le 2024-03-10.

[34] GeeksforGeeks. Support vector machine algorithm, 2024. https://www.geeksforgeeks.org/support-vector-machine-algorithm/, Consulté le 2024-03-10.

[35] Khandaker Mohammad Mohi Uddin, Rokaiya Ripa, Nilufar Yeasmin, Nitish Biswas, and Samrat Kumar Dey. Machine learning-based approach to the diagnosis of cardiovascular vascular disease using a combined dataset. *Intelligence-Based Medicine*, 7 :100100, 2023.

[36] JunYi Xu. Systematic analysis and application prospect of decision tree. *Highlights in Science, Engineering and Technology*, 71 :163–170, 11 2023.

[37] Jeongeon Lee, Min Sim, and Jung-Sik Hong. Assessing decision tree stability : A comprehensive method for generating a stable decision tree. *IEEE Access*, PP :1–1, 01 2024.

[38] Shahad Ibrahim Mohammed, Nazar K Hussein, Outman Haddani, Mansourah Aljohani, Mohammed Abdulrazaq Alkahya, and Mohammed Qaraad. Fine-tuned cardiovascular risk assessment : Locally weighted salp swarm algorithm in global optimization. *Mathematics*, 12(2) :243, 2024.

[39] Esri. Fonctionnement de l'algorithme xgboost. https://pro.arcgis.com/fr/pro-app/latest/tool-reference/geoai/how-xgboost-works.htm,Consulté le 2024-06-25.

[40] Usman Musa, Muhammad Aliyu, and Sani Dauda Abdullahi. Improving performance of heart disease prediction with bayesian network model with feature subset selection. 08 2022.

[41] LeBigData. Deep learning ou apprentissage profond : définition, concept, 24 juillet 2022. https://www.lebigdata.fr/deep-learning-definition,Consulté le 2024-06-05.

[42] Youcef Djeriri. *Les Réseaux de Neurones Artificiels*. 09 2017.

[43] Mustapha Bourahla. Chapitre 2 : Le neurone formel, 2021. https://elearning.univ-msila.dz/moodle/pluginfile.php/398600/mod_resource/content/1/_RN2A_papier_chapitre_2.pdf Consulté le 2024-06-01.

[44] Guotian Xie and Jianhuang Lai. An interpretation of forward-propagation and back-propagation of dnn. In *Pattern Recognition and Computer Vision : First Chinese Conference, PRCV 2018, Guangzhou, China, November 23-26, 2018, Proceedings, Part II 1*, pages 3–15. Springer, 2018.

[45] DECI AI. Deep neural network (dnn). `https://deci.ai/deep-learning-glossary/deep-neural-network-dnn/`,Consulté le 2024-06-05.

[46] Baeldung. Multi-layer perceptron vs deep neural network, 2023. `https://www.baeldung.com/cs/mlp-vs-dnn`,Consulté le 2024-06-05.

[47] I Nyoman Gede Arya Astawa, I Putu Bagus Arya Pradnyana, and I Ketut Suwintana. Comparison of rnn, lstm, and gru methods on forecasting website visitors. *Journal of Computer Science and Technology Studies*, 4(2) :11–18, 2022.

[48] Ph.D. Raghurami Etukuru. Quelles sont les différences et les similitudes entre lstm et gru ?, 2024. `https://fr.linkedin.com/advice/0/what-differences-similarities-between-lstm-gru?lang=fr#:~:text=LSTM%20a%20un%20%C3%A9tat%20de,qui%20peut%20limiter%20sa%20capacit%C3%A9.`.

[49] Hassan Chouaib. Sélection de caractéristiques : méthodes et applications. *Paris Descartes University : Paris, France*, 2011.

[50] Mariam Kalakech. *Sélection semi-supervisée d'attributs : Application à la classification de textures couleur*. PhD thesis, Thèse de doctorat, Université Lille 1, Sciences et Technologies, 2011.

[51] Melle MENGHOUR Kamilia. *Approches Bio-inspirées pour la Sélection d'Attributs*. PhD thesis, Université Badji Mokhtar-Annaba, 1945.

[52] Edmundo Huerta. Logique floue et algorithmes génétiques pour le pré-traitement de données de biopuces et la sélection de gènes.

[53] Les principales techniques de sélection de fonctionnalités, 2024. `https://fr.linkedin.com/advice/3/what-top-techniques-selecting-features-machine-mng9f?lang=fr`, Consulté le 2024-06-04.

[54] Prashant Banerjee. Comprehensive guide on feature selection. `https://www.kaggle.com/code/prashant111/comprehensive-guide-on-feature-selection`, Consulté le 2024-06-04.

[55] ARUNMOHAN. Feature selection tutorial, 2023. `https://www.kaggle.com/code/arunmohan003/feature-selection-tutorial`, Consulté le 2024-06-04.

[56] DataMapu. Feature selection methods, 2023. `https://datamapu.com/posts/ml_concepts/feature_selection/`, Consulté le 2024-06-04.

[57] Piyush Agnihotri. Feature selection techniques in machine learning. `https://www.kaggle.com/code/piyushagni5/feature-selection-techniques-in-machine-learning`, Consulté le 2024-06-04.

[58] Elli Tzini. Feature selection : Embedded methods, 2020. `https://medium.com/analytics-vidhya/feature-selection-embedded-methods-a7940036973f`, Consulté le 2024-06-04.

[59] Picsellia. La matrice de confusion en computer vision. `https://www.picsellia.fr/post/matrice-confusion-computer-vision`,Consulté le 2024-06-05.

[60] Kobia. Accuracy : définition, calcul et limites. https://kobia.fr/classification-metrics-accuracy/#:~:text=L'accuracy%0est%0une%0m%3%9trique,'article%0Precision%20%26%20Recall).,Consulté le 2024-06-05.

[61] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1) :5–53, 2004.

[62] Arnaud De Myttenaere, Bénédicte Le Grand, and Fabrice Rossi. Consistance de la minimisation du risque empirique pour l'optimisation de l'erreur relative moyenne. In *47èmes Journées de Statistique de la SFdS*, 2015.

[63] Abderrazek SEBBAR, Salim HEDDAM, and Lakhdar DJEMILI. Modélisation de l'évaporation journalière (ep) des réservoirs de barrages à l'aide d'une approche basée sur les réseaux de neurones à fonction de régression générale (grnn) : Cas du réservoir du barrage d'el agram, jijel, a l'est de l'algérie modelling daily evaporation (ep) from dams reservoirs using.

[64] Yasunobu Nohara, Koutarou Matsumoto, Hidehisa Soejima, and Naoki Nakashima. Explanation of machine learning models using shapley additive explanation and application for real data in hospital. *Computer Methods and Programs in Biomedicine*, 214 :106584, 2022.

[65] Aditya Gupta and Amritpal Singh. An optimal multi-disease prediction framework using hybrid machine learning techniques. *Kuwait Journal of Science*, 49, 2022.

[66] Fahad A Alghamdi, Haitham Almanaseer, Ghaith Jaradat, Ashraf Jaradat, Mutasem K Alsmadi, Sana Jawarneh, Abdullah S Almurayh, Jehad Alqurni, and Hayat Alfagham. Multilayer perceptron neural network with arithmetic optimization algorithm-based feature selection for cardiovascular disease prediction. *Machine Learning and Knowledge Extraction*, 6(2) :987–1008, 2024.

[67] J Nageswara Rao. Cardiovascular disease prediction using machine learning techniques. *Turkish Journal of Physiotherapy and Rehabilitation*, 32 :6875–6880.

[68] Aarti Karandikar. Advanced heart disease prediction : Deep learning-enhanced convolutional neural network in the internet of medical things environment. *Journal of Electrical Systems*, 20 :1–10, 03 2024.

[69] JMP. Analyse exploratoire des données. https://www.jmp.com/fr_ca/statistics-knowledge-portal/exploratory-data-analysis.html, Consulté le 2024-06-05.

[70] Heart disease cleveland dataset. https://archive.ics.uci.edu/dataset/45/heart+disease,Consultle2024-02-01.

[71] Heart failure prediction dataset. https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction,Consultle2024-02-01.

[72] Yash Baravaliya. Feature scaling and normalization, 2023. https://medium.com/@yashbaravaliya206/feature-scaling-and-normalization-ca484a16882a#:~:text=Min%DMax%0Scaling%20(Normalization),the%0maximum%0value%0of%0X.,Consulté le 2024-06-05.

[73] IZZET TURKALP AKBASLI. Tutorial of genetic algorithm, 2022. https://www.kaggle.com/code/zzettrkalpakbal/tutorial-of-genetic-algorithm, Consulté le 2024-06-04.