



RAPPORT TRAVAIL PRATIQUE EN  
DATA SCIENCE ET INTELLIGENCE ARTIFICIELLE

**PROJET DE MACHINE  
LEARNING :**

*Prévision de maladie cardio vasculaire*

## LISTE DES FIGURES

<b>Figure 1</b> : résultats de l'analyse descriptive .....	21
<b>Figure 2</b> : schéma d'un modèle de decision tree.....	24
<b>Figure 3</b> : arbre de decision technique .....	25
<b>Figure 4</b> : schéma d'un modèle bagging.....	26
<b>Figure 5</b> : schéma d'une regression logistique.....	26
<b>Figure 6</b> : PCA Visual.....	32
<b>Figure 7</b> : dispersion des PCA 1 et PCA 2 en fonction de la Target.....	35
<b>Figure 8</b> : precision Visual.....	35
<b>Figure 9</b> : calcul du f1-score.....	35
<b>Figure 10</b> : accuracy Visual.....	36
<b>Figure 11</b> : résultat score decision tree .....	39
<b>Figure 12</b> : résultat score bagging 80-20.....	39
<b>Figure 13</b> : résultat score bagging 60-40.....	40



## LISTE DES TABLEAUX

<b>Tableau 1 :</b> datasets.....	14
<b>Tableau 2 :</b> datasets avec les NA traités .....	16



## LISTE DES GRAPHIQUES

<b>Graphique 1</b> : fréquence des données en fonction du Target .....	19
<b>Graphique 2</b> : scatter plot du dataset .....	20
<b>Graphique 3</b> : PCA Visual.....	32
<b>Graphique 4</b> : matrice de confusion Visual .....	36
<b>Graphique 5</b> : matrice de confusion decision tree.....	37
<b>Graphique 6</b> : matrice de confusion bagging 80-20.....	37
<b>Graphique 7</b> : matrice de confusion 60-40 .....	38



## LISTE DES SYNTAXES

<b>Syntaxe 1</b> : decision tree du travail.....	27
<b>Syntaxe 2</b> : bagging du travail.....	28
<b>Syntaxe 3</b> : regression logistique.....	29
<b>Syntaxe 4</b> : PCA.....	31
<b>Syntaxe 5</b> : resultat PCA.....	33



## **SIGLES ET ABREVIATIONS**

**ML** : Machine Learning

**IA** : Intelligence Artificielle

**PCA** : Analyse en Composantes Principales

**NLP** : Traitement du langage naturel



## RESUME

Dans la vie de tous les jours, nous sommes soumis à des problèmes précis. Parlant ici des problèmes liés au développement et l'évolution de vie tels que la pauvreté, le manque d'infrastructure, technicité en arrière, ressources insuffisants etc... Notre but en tant que chercheur en intelligence artificielle et donc de créer des solutions à ces différents maux. Dans notre travail, il est question pour nous d'utiliser la sous branche de l'IA qui est le ML dans la médecine, plus précisément la prévision de maladie dans 10 ans. Pour cela, nous avons opté pour 03 modèles : le Bagging, les arbres de décision et le modèle logistique. Par la suite, nous passerons à l'évaluation de ceux-ci à partir de différentes métriques et nous en tirons des conclusions.

**Mots clés : Machine Learning, intelligence artificielle, decision tree, bagging, modèle.**



# ABSTRACT

In everyday life, we are subject to specific problems. Speaking here about the problems related to the development and evolution of life such as poverty, lack of infrastructure, backward technicality, insufficient resources etc... Our goal as a researcher in artificial intelligence and therefore to create solutions to these different evils. In our work, it is a question for us of using the sub-branch of AI which is ML in medicine, more precisely the prediction of disease in 10 years. For this, we opted for 3 models: Bagging, decision trees and the logistics model. Subsequently, we will move on to evaluating these using different metrics and drawing conclusions.

**Keywords** : Machine Learning, artificial intelligence, decision tree, bagging, model.



# SOMMAIRE

## **PARTIE 1 : PRESENTATION, TRAITEMENT ET ANALYSE DESCRIPTIVE DES DONNEES.....12**

CHAPITRE 1 : PRESENTATION ET TRAITEMENT DES DONNEES.....13

SECTION 1 : PRESENTATION DES DONNEES.....14

SECTION 2 : TRAITEMENT DES DONNEES.....16

CHAPITRE 2 : ANALYSE DESCRIPTIVE DES DONNEES.....18

SECTION 1 : VISUALISATION DES DONNEES .....19

SECTION 2 : RESULTATS DE L'ANALYSE DESCRIPTIVE.....20

## **PARTIE 2 : MODELES DE MACHINE LEARNING ET EVALUATION .....22**

CHAPITRE 3 : MODELE DE MACHINE LEARNING .....23

SECTION 1 : PRESENTATION THEORIQUE DES MODELES DE MACHINE LEARNING.....24

SECTION 2 : MISE EN ŒUVRE TECHNIQUE DES MODELES.....27

SECTION 3 : ANALYSE EN COMPOSANTES PRINCIPALES.....30

CHAPITRE 4 : EVALUATION.....34

SECTION 1 : PRESENTATION THEORIQUES DES METRIQUE.....35

SECTION 2 : TECHNICITE DES METRIQUES.....37



# INTRODUCTION GENERALE

En 1950, l'intelligence artificielle voit le jour avec JOHN VON NEUMANN et ALAN TURING et évolue au fil des années. Aujourd'hui les capacités et les applications de celle-ci sont bien plus améliorées comparées à ce qu'elles étaient avant et continueront à s'améliorer. L'IA comporte plusieurs sous branches que nous citons ici : Le machine Learning ou apprentissage machine, le Deep Learning, le NLP. Par la suite, nous nous attarderons sur le machine Learning appliquée à la prédiction des maladies cardiovasculaires. Ainsi donc, se dégage la question principale, comment le machine Learning permet d'assister les experts médicaux dans la prédiction efficace des maladies cardiovasculaires ?

Ce travail a pour objet principal d'utiliser les données à disposition pour visualiser la disparité et la spécificité des différentes variables, choisir les variables déterminantes afin de mettre sur pied des modèles de prédictions permettant de répondre au problème.

Une méthodologie quantitative s'impose à nous basée sur une base de données chiffrée à disposition constituée des caractéristiques des individus.

Les modèles d'apprentissages utilisés ici sont le **Décision tree**, le **Bagging** et le **modèle de regression logistique** et les métriques **la précision, le f1-score, l'accuracy et la matrice de confusion**.

Ainsi donc nous commencerons par faire une analyse descriptive et la visualisation des données, la programmation des modèles et nous en déduirons des observations en fonction des résultats obtenus.



## **PARTIE 1 : PRESENTATION, TRAITEMENT ET ANALYSE DESCRIPTIVE DES DONNEES**

Le traitement des données est un processus qui consiste à collecter, enregistrer, organiser, stocker, modifier, extraire, utiliser, communiquer, diffuser ou mettre à disposition des données personnelles. Ainsi cette phase constitue une étape très importante dans l'obtention de bons résultats finaux.



## **CHAPITRE 1 : PRESENTATION ET TRAITEMENT DES DONNEES**

Toute analyse prédictive demande la conception d'une base de données conséquente basée sur les caractéristiques qui peuvent avoir un impact sur l'élément que l'on veut prédire. Dans notre cas, nous disposons de 16 caractéristiques d'une population et dans ce travail, nous présenterons la phase de traitement des données, l'analyse descriptive et la visualisation des ceux-ci.

## SECTION 1 : PRESENTATION DES DONNEES

En matière de conception de base de données pour l'étude des problèmes, les variables choisies doivent être déterminantes et étroitement liés aux problèmes que l'on souhaite prédire.

Le tableau fournit est constitué de :

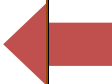
- 4240 lignes
- 16 colonnes

Ces caractéristiques se visualisent comme suit :

**Table 1 : Datasets**

male	age	education	currentSmo	cigsPerDay	BPMeds	prevalentSm	prevalentH	diabetes	totChol	sysBP	diaBP	BMI	heartRate	glucose	TenYearCHD
1	39	4	0	0	0	0	0	0	195	106	70	26.97	80	77	0
0	46	2	0	0	0	0	0	0	250	121	81	28.73	95	76	0
1	48	1	1	20	0	0	0	0	245	127.5	80	25.34	75	70	0
0	61	3	1	30	0	0	1	0	225	150	95	28.58	65	103	1
0	46	3	1	23	0	0	0	0	285	130	84	23.1	85	85	0
0	43	2	0	0	0	0	1	0	228	180	110	30.3	77	99	0
0	63	1	0	0	0	0	0	0	205	138	71	33.11	60	85	1
0	45	2	1	20	0	0	0	0	313	100	71	21.68	79	78	0
1	52	1	0	0	0	0	1	0	260	141.5	89	26.36	76	79	0
1	43	1	1	30	0	0	1	0	225	162	107	23.61	93	88	0
0	50	1	0	0	0	0	0	0	254	133	76	22.91	75	76	0
0	43	2	0	0	0	0	0	0	247	131	88	27.64	72	61	0
1	46	1	1	15	0	0	1	0	294	142	94	26.31	98	64	0
0	41	3	0	0	1	0	1	0	332	124	88	31.31	65	84	0
0	39	2	1	9	0	0	0	0	226	114	64	22.35	85	NA	0
0	38	2	1	20	0	0	1	0	221	140	90	21.35	95	70	1
1	48	3	1	10	0	0	1	0	232	138	90	22.37	64	72	0
0	46	2	1	20	0	0	0	0	291	112	78	23.38	80	89	1
0	38	2	1	5	0	0	0	0	195	122	84.5	23.24	75	78	0
1	41	2	0	0	0	0	0	0	195	139	88	26.88	85	65	0
0	42	2	1	30	0	0	0	0	190	108	70.5	21.59	72	85	0
0	43	1	0	0	0	0	0	0	185	123.5	77.5	29.89	70	NA	0
0	52	1	0	0	0	0	0	0	234	148	78	34.17	70	113	0
0	52	3	1	20	0	0	0	0	215	132	82	25.11	71	75	0
1	44	2	1	30	0	0	1	0	270	137.5	90	21.96	75	83	0
1	47	4	1	20	0	0	0	0	294	102	68	24.18	62	66	1
0	60	1	0	0	0	0	0	0	260	110	72.5	26.59	65	NA	0
1	35	2	1	20	0	0	1	0	225	132	91	26.09	73	83	0
0	61	3	0	0	0	0	1	0	272	182	121	32.8	85	65	1
0	60	1	0	0	0	0	0	0	247	130	88	30.36	72	74	0
1	36	4	1	35	0	0	0	0	295	102	68	28.15	60	63	0

**Source** = [kaggle.com](https://www.kaggle.com)



Les variables soumises à notre étude sont :

- male
- age
- education
- currentSmoker
- cigsPerDay
- BPMeds
- PrevalentStroke
- PrevalentHyp
- Diabetes
- totChol
- SysBP
- DiaBP
- BMI
- Heartrate
- Glucose
- TenYearCHD

## SECTION 2- TRAITEMENT DES DONNEES

Le traitement des données surtout en présence des valeurs manquantes est un aspect crucial de l'analyse des données et de la modélisation. Les ensembles de données incomplets peuvent poser des problèmes lors de l'analyse de données et donner lieu à des résultats biaisés ou inexacts. Pandas, une puissante bibliothèque Python pour la manipulation et l'analyse de données, fournit diverses fonctions pour traiter les données manquantes.

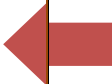
Dans notre travail, nous avons opté pour la méthode de l'imputation par la fonction interpolate des bibliothèques Pandas pour interpoler nos valeurs manquantes.

L'interpolation est une méthode permettant de combler les valeurs manquantes en les estimant sur la base des valeurs d'autres points de données.

**Table 2 : Datasets avec les NA traités**

	Sex_male	age	currentSmoker	cigsPerDay	BPMeds	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP	diaBP	BMI	heartRate
0	1	39	0	0.0	0.0	0	0	0	195.0	106.0	70.0	26.97	80.0
1	0	46	0	0.0	0.0	0	0	0	250.0	121.0	81.0	28.73	95.0
2	1	48	1	20.0	0.0	0	0	0	245.0	127.5	80.0	25.34	75.0
3	0	61	1	30.0	0.0	0	1	0	225.0	150.0	95.0	28.58	65.0
4	0	46	1	23.0	0.0	0	0	0	285.0	130.0	84.0	23.10	85.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
195	0	49	1	9.0	0.0	0	0	0	226.0	106.0	71.0	22.89	85.0
196	1	48	1	10.0	0.0	0	0	0	308.0	117.0	76.0	30.85	65.0
197	0	55	1	9.0	0.0	0	0	0	248.0	157.0	82.5	22.91	89.0
198	0	58	1	5.0	0.0	0	0	0	215.0	170.0	86.0	29.06	75.0
199	1	60	0	0.0	0.0	0	0	0	240.0	137.0	84.0	29.51	82.0

**Source** : groupe de travail



Le tableau ci-dessus fournit notre dataset sans données manquantes. Les espaces contenant ces éléments ont été remplacés par les valeurs interpolées par rapport à d'autres données autour par la méthode **interpolate()** de pandas.



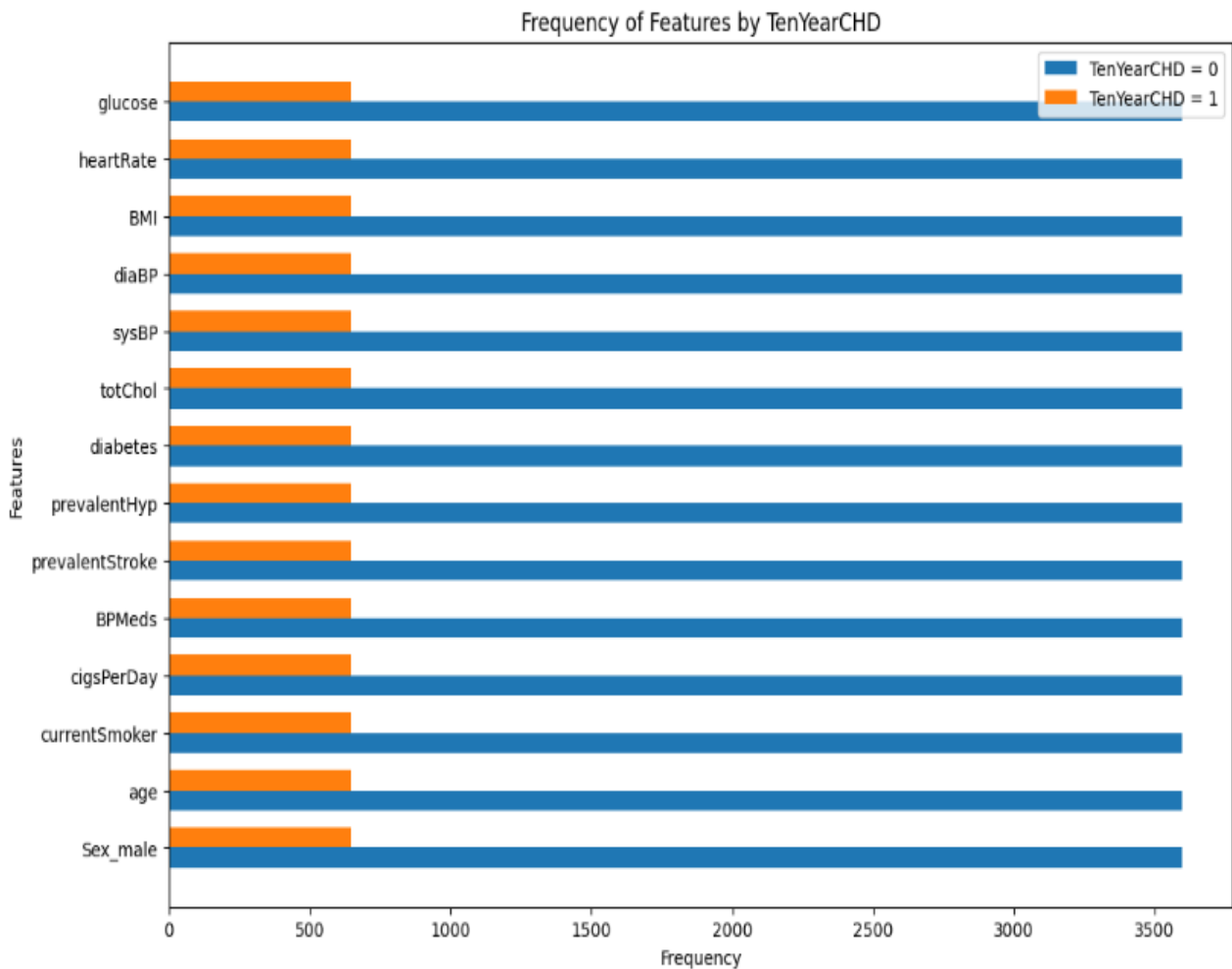


## **CHAPITRE 2 : ANALYSE DESCRIPTIVE DES DONNEES**

Plantant le décor de notre travail, nous allons présenter dans cette sous section l'aspect visuel de nos données, ainsi que certaines de ses caractéristiques descriptives.

## SECTION 1 : VISUALISATION DES DONNEES

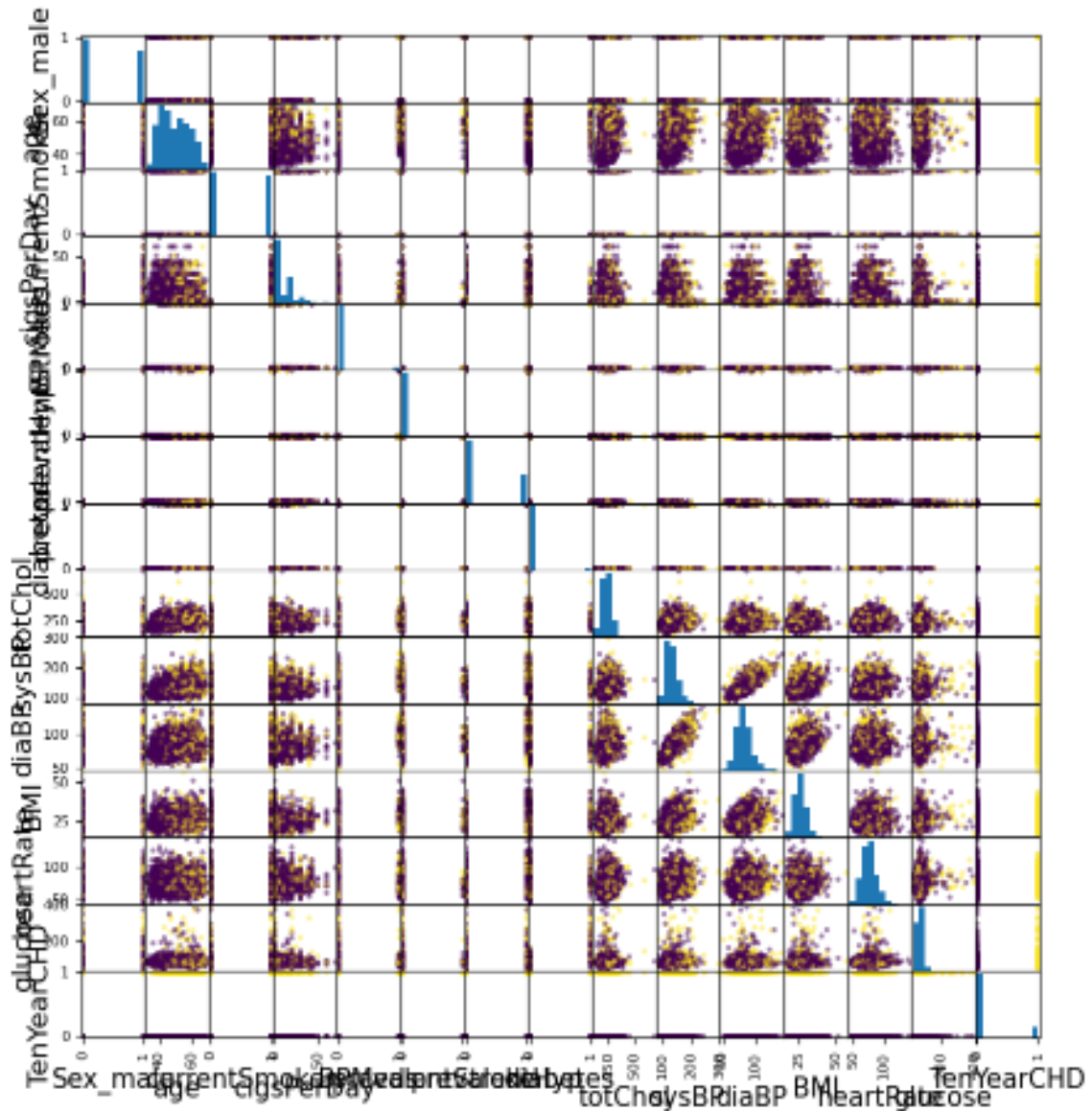
**Graphique 1 : fréquence des données en fonction du Target**



**Source : groupe de travail**

Nous observons que dans ce graphe, chaque variable du dataset est représentée en fonction de la variation de la Target.

Graphique 2 : scatter plot du dataset



Source : groupe de travail

Ici, les variables sont représentées en point et la Target est visible à travers la variation des couleurs.

## SECTION 2 : STATISTIQUES DESCRIPTIVES

Figure 1 : résultats de l'analyse descriptive

	Sex_male	age	currentSmoker	cigsPerDay	BPMeds \
count	3751.000000	3751.000000	3751.000000	3751.000000	3751.000000
mean	0.445215	49.573447	0.488403	9.008531	0.030392
std	0.497056	8.570204	0.499932	11.925097	0.171686
min	0.000000	32.000000	0.000000	0.000000	0.000000
25%	0.000000	42.000000	0.000000	0.000000	0.000000
50%	0.000000	49.000000	0.000000	0.000000	0.000000
75%	1.000000	56.000000	1.000000	20.000000	0.000000
max	1.000000	70.000000	1.000000	70.000000	1.000000

	prevalentStroke	prevalentHyp	diabetes	totChol	sysBP \
count	3751.000000	3751.000000	3751.000000	3751.000000	3751.000000
mean	0.005599	0.311917	0.027193	236.928019	132.368435
std	0.074623	0.463338	0.162666	44.611594	22.046522
min	0.000000	0.000000	0.000000	113.000000	83.500000
25%	0.000000	0.000000	0.000000	206.000000	117.000000
50%	0.000000	0.000000	0.000000	234.000000	128.000000
75%	0.000000	1.000000	0.000000	264.000000	144.000000
max	1.000000	1.000000	1.000000	696.000000	295.000000

	diaBP	BMI	heartRate	glucose	TenYearCHD
count	3751.000000	3751.000000	3751.000000	3751.000000	3751.000000
mean	82.938550	25.808288	75.704079	81.880032	0.152493
std	11.932779	4.065599	11.956382	23.882233	0.359546
min	48.000000	15.540000	44.000000	40.000000	0.000000
25%	75.000000	23.085000	68.000000	71.000000	0.000000
50%	82.000000	25.410000	75.000000	78.000000	0.000000
75%	90.000000	28.060000	82.000000	87.000000	0.000000
max	142.500000	56.800000	143.000000	394.000000	1.000000

Source : groupe de travail

Cette analyse renvoie les différentes variations et les données des statistiques descriptives de chaque variables tels que : la moyenne, le minimum, le maximum, les quantiles.



## **PARTIE 2 : MODELES DE MACHINE LEARNING ET EVALUATION**

Après la mise sur pied d'un modèle, il va de soit qu'il faut l'évaluer afin de déterminer sa précision par rapport à d'autres modèles. L'objet de cette partie est de présenter de manière théorique et technique les différents modèles et évaluer leurs performances à travers différentes métriques.



## **CHAPITRE 3 : MODELES DE MACHINE LEARNING**

Par définition, un modèle est une représentation simplifiée de la réalité. En intelligence artificielle, ses applications sont multiples et au fil des années les méthodes de conception des modèles de plus en plus performantes voient le jour.

Dans notre projet, 03 principaux modèles seront portés à vos papilles :

- Le Decision Tree
- Le Bagging
- Le modèle de régression logistique.

# SECTION 1 : PRESENTATION THEORIQUE DES MODELES DE ML

Tout travail nécessitant une revue des normes théoriques, présentons dans ce qui suit les différents modèles théoriquement.

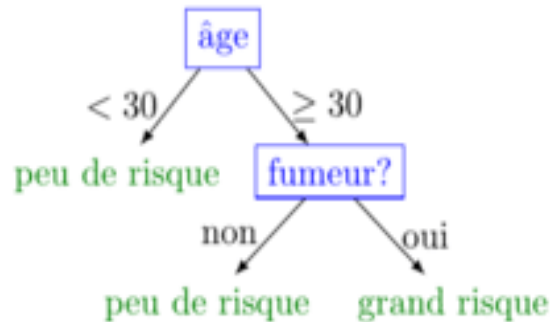
## I-1- Modèle de Decision Tree

Figure 2 : schéma d'un modèle de decision tree



Source : <https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.lemarson.com%2Farticle%2Fles-arbres-de-decision-de-lapprentissage>

**Figure 3 : arbre de decision technique**

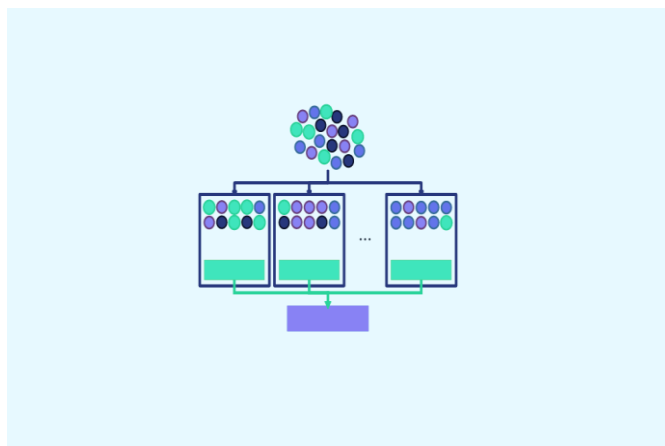


**Source :** [https://www.google.com/imgres?imgurl=https%3A%2F%2Fupload.wikimedia.org%2Fwikipedia%2Fcommons%2Fthumb%2F7%2F74%2FArbrededecision\\_risque\\_cardiaque.png%2F220px-Arbrededecision\\_risque\\_cardiaque.png&tbnid=qn9JY-](https://www.google.com/imgres?imgurl=https%3A%2F%2Fupload.wikimedia.org%2Fwikipedia%2Fcommons%2Fthumb%2F7%2F74%2FArbrededecision_risque_cardiaque.png%2F220px-Arbrededecision_risque_cardiaque.png&tbnid=qn9JY-)

Les **arbres de décision** sont un modèle populaire, utilisé dans la recherche opérationnelle, la planification stratégique et le **Machine Learning**. Chaque rectangle ci-dessus est appelé un **nœud**. Plus vous avez de nœuds, plus votre arbre décisionnel sera précis (en général). Les derniers nœuds de l'arbre décisionnel, où une décision est prise, sont appelés les « **feuilles** » de l'arbre. Les arbres décisionnels sont intuitifs et faciles à construire, mais ils font un peu défaut lorsqu'on parle de précision ou d'exactitude.

## I-2- Modèle de Bagging

**Figure 4 : Schéma d'un modèle de Bagging**



**Source :** [https://www.google.com/imgres?imgurl=https%3A%2F%2Fupload.wikimedia.org%2Fwikipedia%2Fcommons%2Fthumb%2F7%2F74%2FArbrededecision\\_risque\\_cardiaque.png%2F220px-Arbrededecision\\_risque\\_cardiaque.png&tbnid=qn9JY-](https://www.google.com/imgres?imgurl=https%3A%2F%2Fupload.wikimedia.org%2Fwikipedia%2Fcommons%2Fthumb%2F7%2F74%2FArbrededecision_risque_cardiaque.png%2F220px-Arbrededecision_risque_cardiaque.png&tbnid=qn9JY-)



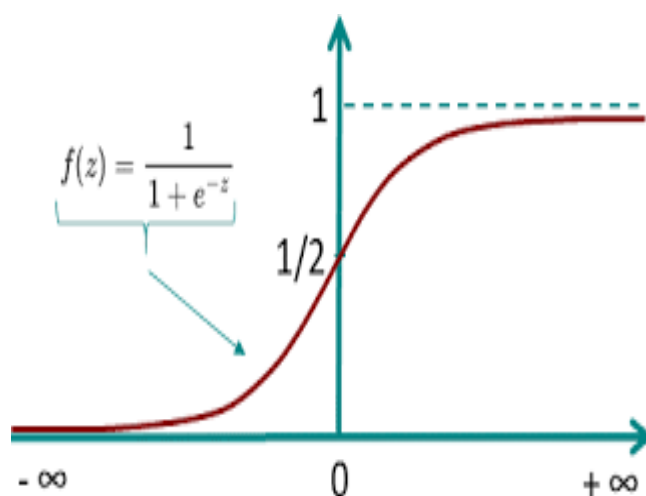
**“Ensemble on est plus fort”** : On pourrait symboliser le bagging par cette citation. En effet, cette technique fait partie des méthodes d'ensemble, qui consiste à considérer un ensemble de modèles pour prendre la décision finale. Nous allons voir en détail le cas du bagging. Le bagging, ou Bootstrap aggregation, est la méthode d'entraînement ensembliste couramment utilisée pour réduire la variance dans un fichier bruyant. Dans le bagging, un échantillon aléatoire de données dans un ensemble d'entraînement est sélectionné avec remplacement, ce qui signifie que les points de données individuels peuvent être choisis plusieurs fois. Après avoir généré plusieurs échantillons de données, ces modèles faibles sont ensuite entraînés indépendamment et, selon le type de tâche (régression ou classification, par exemple), la moyenne ou la majorité de ces prédictions produisent une estimation plus précise. Pour rappel, l'algorithme de forêt aléatoire est considéré comme une extension de la méthode bagging, car il utilise à la fois le bagging et le caractère aléatoire de la fonction pour créer une forêt non corrélée d'arbres de décision.

### I-3- Modèle de régression logistique

La régression logistique est semblable à la régression linéaire, mais elle est utilisée pour modéliser la probabilité d'un nombre fini de résultats, généralement deux. Il y a plusieurs raisons pour lesquelles la régression logistique est utilisée par rapport à la régression linéaire lors de la modélisation des probabilités de résultats.

Une équation logistique est créée de telle sorte que les valeurs des résultats ne peuvent être qu'entre 0 et 1 (voir ci-dessous).

**Figure 5 : Schéma d'une régression logistique**



**Source** : <https://www.google.com/imgres?imgurl=https%3A%2F%2Fdatatab.fr%2Fassets%2Ftutorial%2FLogisticfunction.png>

## SECTION 2 : MISE EN ŒUVRE TECHNIQUE DES MODELES

Passons à présent à l'aspect technique du travail consistant à la mise en œuvre et au codage.

### 2-1- Modèle de Decision Tree

#### Syntaxe 1 : Decision Tree du travail

```
[34] from matplotlib import pyplot as plt
      from sklearn.tree import DecisionTreeClassifier
      from sklearn.metrics import accuracy_score, f1_score, precision_score, confusion_matrix, recall_score, roc_auc_score, classification_report
      import numpy as np
      from collections import Counter
      from imblearn.over_sampling import SMOTE
```

```
▶ X = np.asarray(dataset[['age', 'Sex_male', 'BPMeds', 'prevalentStroke',
                        , 'prevalentHyp', 'diabetes', 'BMI', 'totChol', 'sysBP',
                        'diaBP', 'glucose']])
y = np.asarray(dataset['TenYearCHD'])

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.4, random_state = 100)

sm = SMOTE()
X_res, y_res = sm.fit_resample(X_train, y_train)
#new_train = pd.concat([X_res, y_res], axis=1)

model1 = DecisionTreeClassifier(criterion = "entropy", max_depth=3)

model1.fit(X_train, y_train)

y_pred = model1.predict(X_test)

classRep = classification_report(y_test, y_pred)
#confMat = confusion_matrix(y_test, y_pred)
```

```

#Precision metrics
precision = precision_score(y_test, y_pred)
print("precision:", round(precision*100, 2), "%")

#recall metrics
# recall = recall_score(y_test, y_pred)
# print("recall:", round(recall, 2))

#F1-score Metrics
f1 = f1_score(y_test, y_pred)
print("f1_score:", round(f1, 2))
#print(y_pred)

score1 = model1.score(X_test, y_test)
score2 = model1.score(X_train, y_train)

#Precision in test:
print(f'Score_Test: {round(score1, 3)*100}', '%')
#print("Accuracy", metrics.accuracy_score(y_test, y_pred))
#Precision in training:
print(f'Score_Training: {round(score2, 3)*100}', '%')

precision: 36.36 %
f1_score: 0.03
Score_Test: 85.7 %

```

Source : groupe de travail

## 2-2- Modèle de Bagging

### Syntaxe 2 : Bagging du travail

```

# Division des données en 80% - 20%
[68] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

```

```

▶ model2 = BaggingClassifier(base_estimator=KNeighborsClassifier(),
                             n_estimators=100)
model2.fit(X_train, y_train)
# print(model.score(X_test, y_test))
# print()

y_pred = model2.predict(X_test)

#Precision metrics
precision = precision_score(y_test, y_pred)
print("precision:", round(precision*100, 2), "%")

#recall metrics
# recall = recall_score(y_test, y_pred)
# print("recall:", round(recall, 2))

#F1-score Metrics
f1 = f1_score(y_test, y_pred)
print("f1_score:", round(f1, 2))
#print(y_pred)

score1 = model1.score(X_test, y_test)
score2 = model1.score(X_train, y_train)

#Precision in test:
print(f'Score_Test: {round(score1, 3)*100}', '%')
#print("Accuracy", metrics.accuracy_score(y_test, y_pred))
#Precision in training:
print(f'Score_Training: {round(score2, 3)*100}', '%')

```

Source : groupe de travail

## 2-3- Modèle de régression logistique

### Syntaxe 3 : régression logistique

#### LOGISTIC REGRESSION MODEL

```
X = np.asarray(dataset[['age', 'Sex_male', 'BPMeds', 'prevalentStroke',  
                        'prevalentHyp', 'diabetes', 'totChol', 'sysBP',  
                        'diaBP', 'BMI', 'glucose']])  
y = np.asarray(dataset['TenYearCHD'])
```

```
# Normalisation du dataset  
X = StandardScaler().fit(X).transform(X)
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.4, random_state = 3)
```

```
modell = LogisticRegression()  
modell.fit(X_train, y_train)  
  
# Le recall et la precision et Le recall  
from sklearn.metrics import recall_score, precision_score  
y_pred = modell.predict(X_test)  
print('Le recall vaut', recall_score(y_pred, y_test))  
print('La precision vaut', precision_score(y_pred, y_test))  
  
print("La performance à l'entrainement vaut", modell.score(X_train, y_train))  
print("La performance au test vaut", modell.score(X_test, y_test))
```

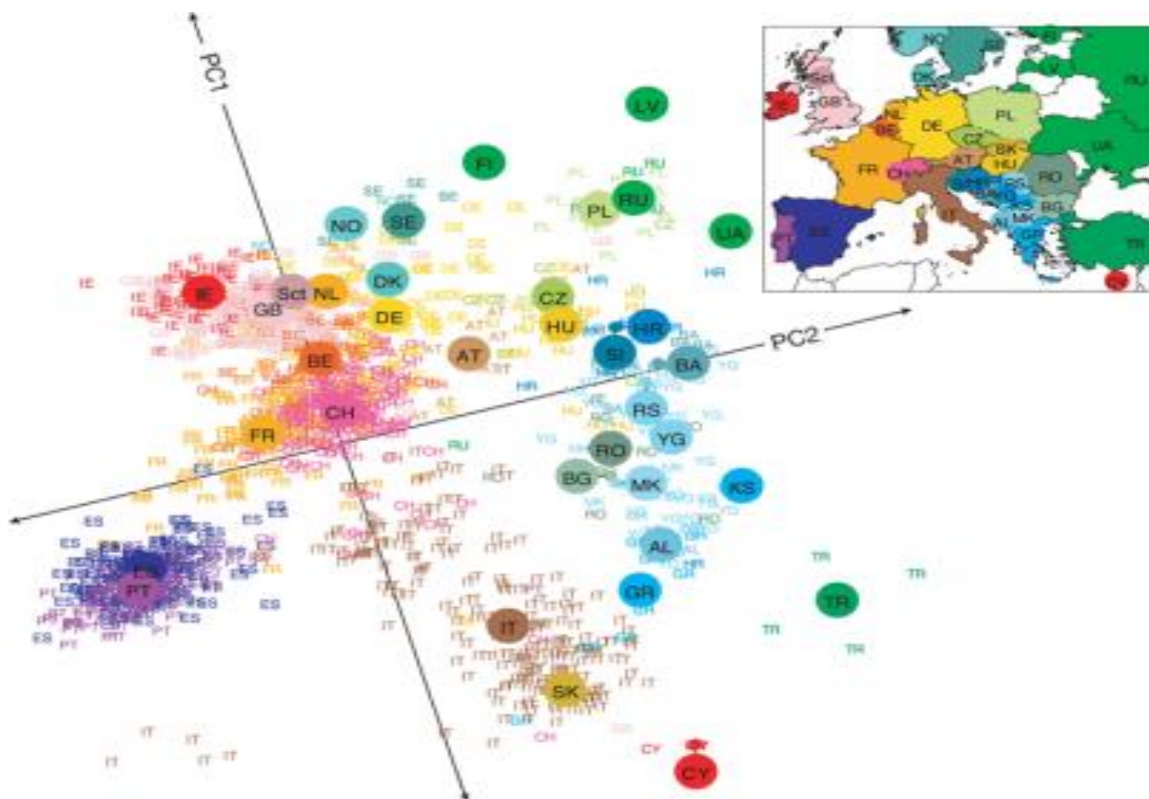
```
Le recall vaut 0.47619047619047616  
La precision vaut 0.045662100456621  
La performance à l'entrainement vaut 0.8475555555555555  
La performance au test vaut 0.8534310459693538
```

**Source : groupe de travail**

## SECTION 3 : ANALYSE EN COMPOSANTES PRINCIPALES (PCA)

La PCA est une méthode de réduction de la dimensionnalité permettant d'obtenir une base de données inférieure à celle que l'on avait au départ tout en conservant une certaine proportion de la fiabilité des résultats ; généralement 95 à 99 %.

### Figure 6 : PCA Visual



**Source :**<https://www.google.com/url?sa=i&url=https%3A%2F%2Fopenclassrooms.com%2Ffr%2Fcourses%2F4379436-explorez-vos-donnees-avec-des-algorithmes-non-supervises%2F4379481-calculez-les-composantes-principales-de-vos-donnees>

## Syntaxe 4 : PCA

```
import pandas as pd
import numpy as np
import random as rd

mydata = pd.read_csv("framingham.csv")
```

```
mydata = pd.read_csv("framingham.csv")
mydata.fillna(mydata.mean(), inplace = True)
df = pd.DataFrame(mydata)
```

```
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
```

```
scaler = MinMaxScaler()
scaler.fit(df)
```

```
MinMaxScaler()
```

```
scaled_data = scaler.transform(df)
```

```
from sklearn.decomposition import PCA
```

```
pca = PCA(n_components = 16)
```

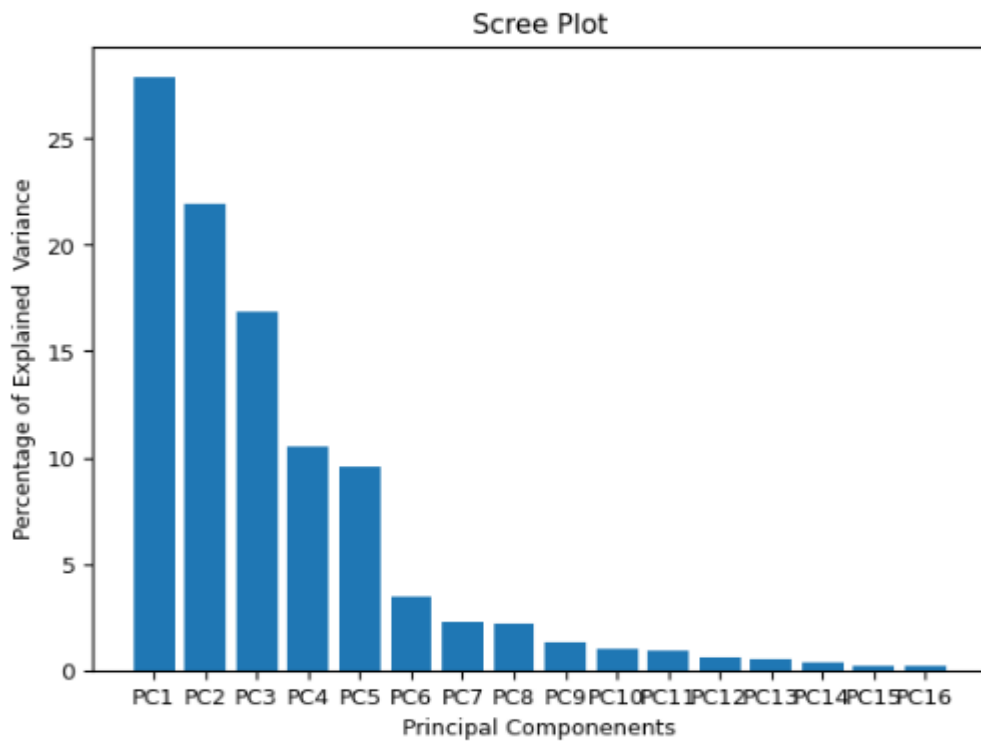
```
pca.fit(scaled_data)
```

```
PCA(n_components=16)
```

```
per_var = np.round(pca.explained_variance_ratio_*100, decimals = 1)
labels = ['PC' + str(x) for x in range(1, len(per_var)+1)]
plt.bar(x=range(1, len(per_var)+1), height = per_var, tick_label = labels)
plt.ylabel('Percentage of Explained Variance')
plt.xlabel('Principal Componentents')
plt.title('Scree Plot')
plt.show()
```

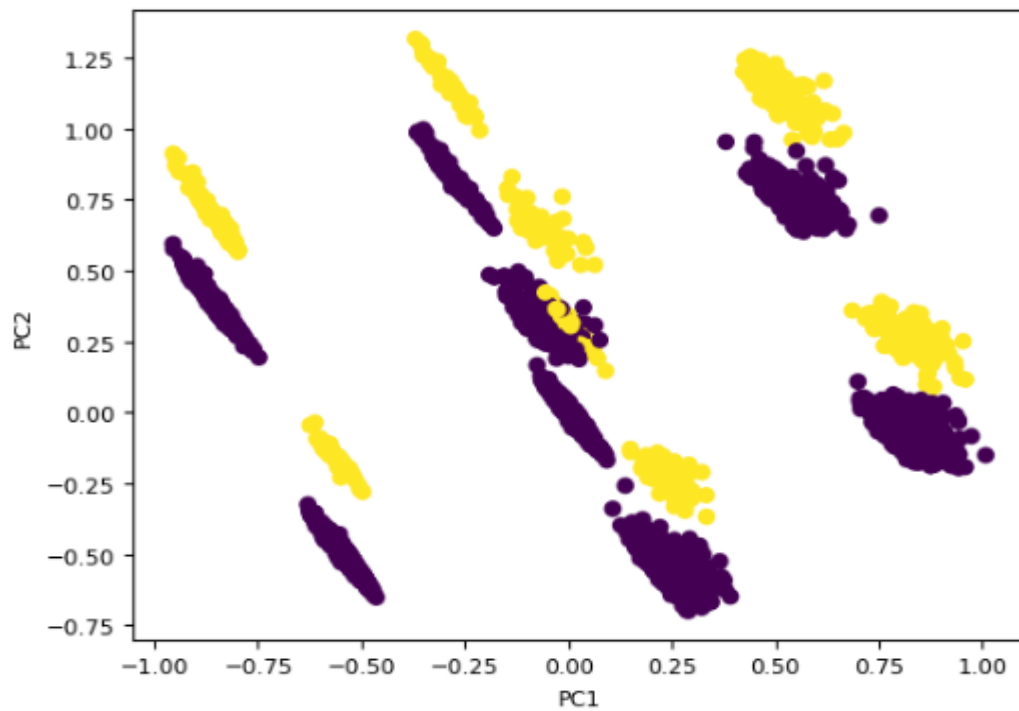
**Source : groupe de travail**

### Graphique 3 : PCA visual



Source : groupe de travail

Figure 7 : dispersion des PCA 1 et PCA 2 en fonction de la Target



Source : groupe de travail

### Syntaxe : résultat de la PCA

```
: loading_scores = pd.Series(pca.components_[0])
sorted_loading_scores = loading_scores.abs().sort_values(ascending = False)
top_10_features = sorted_loading_scores[0:10]
top_10_features

: 3    0.739096
0    0.560280
7    0.256468
4    0.224375
1    0.115492
10   0.055156
11   0.046518
2    0.044188
5    0.041104
12   0.028277
dtype: float64
```

**Source :** groupe de travail

D'après ce résultat on conclut qu'après l'analyse, seuls 10 features ont été retenus pour la prevision du target.





## **CHAPITRE 4 : EVALUATION**

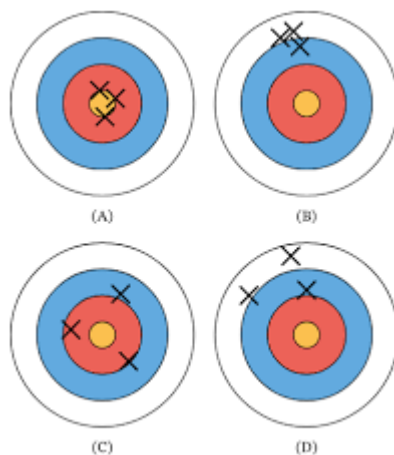
L'évaluation d'un modèle est une étape cruciale dans le processus de modélisation. Elle permet de mesurer la qualité du modèle et de déterminer d'il est approprié pour résoudre le problème pour lequel il a été conçu.

## SECTION 1 : PRESENTATION THEORIQUES DES METRIQUES

### I-1- la précision

La précision représente le nombre de prévisions correctes par rapport à toutes celles qui sont établies.

**Figure 8 : Précision Visual**



**Source :** <https://images.nagwa.com/figures/explainers/946105690524/3.svg>

### I-2- le f1-score

Le f1-score est une métrique qui prend en compte la notion de faux positifs et faux négatifs. Elle se base sur le calcul de deux mesures qui font appel à la matrice de confusion.

**Figure 9 : calcul du f1-score**

$$\begin{aligned} \text{F1 Score} &= \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}} \\ &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \end{aligned}$$

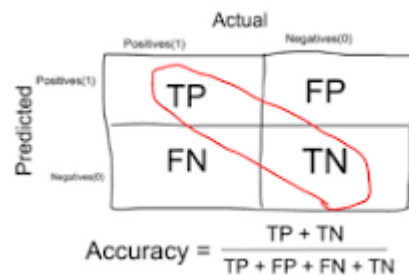
**Source :**

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.v7labs.com%2Fblog%2Ff1-score->

### I-3- le score (accuracy)

L'accuracy mesure l'efficacité d'un modèle à prédire correctement à la fois les individus positifs et négatifs.

**Figure 10 : accuracy visual**



**Source :**

<https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcTQ0LTeu1QrGR4iOW0Edl7uOG0sPZdzeX5dz1bFuP0enw&s>

### I-4- la matrice de confusion

La matrice de confusion permet de connaître d'une part les différentes erreurs commises par un algorithme de prédiction, mais plus important encore, de connaître les différents types d'erreurs commis.

**Graphique 4 : matrice de confusion visual**

		True Class	
		P	N
Predicted Class	P	TP	FP
	N	FN	TN

**Source :** <https://www.google.com/url?sa=i&url=https%3A%2F%2Fintelligence-artificielle.com%2Fconfusion-matrix-dossier-complet>

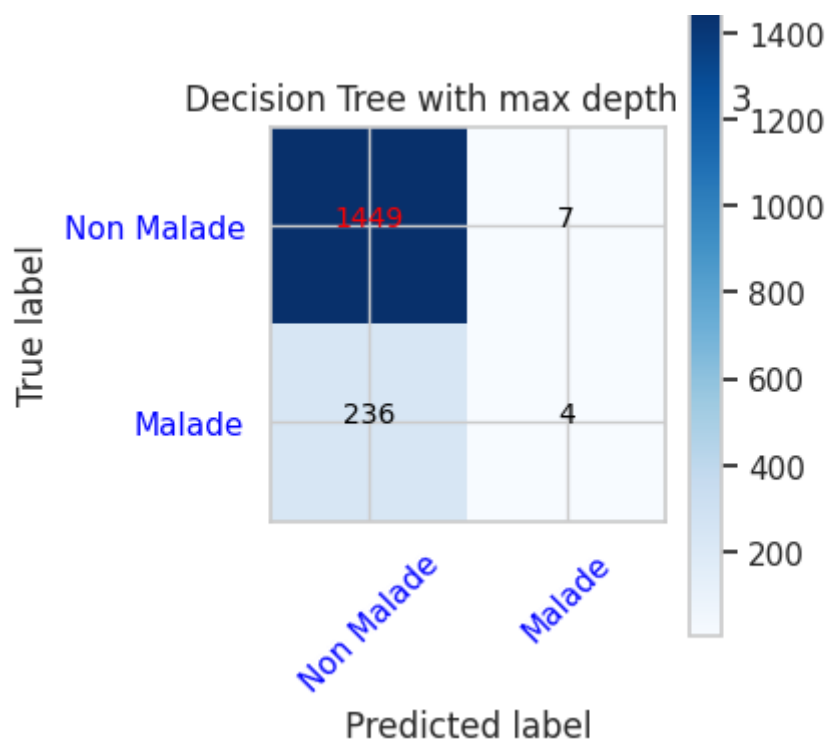
## SECTION 2 : TECHNICITE DES METRIQUES

### 2-1- La matrice de confusion

Dans notre cas d'étude, nous avons établi les matrices de confusion pour les différents modèles que nous avons choisi d'entraîner et de tester. Les résultats qui s'en sont donc suivi ont été obtenus comme suit :

#### 2-1-1- DECISION TREE :

Graphique 5 : matrice de confusion decision Tree

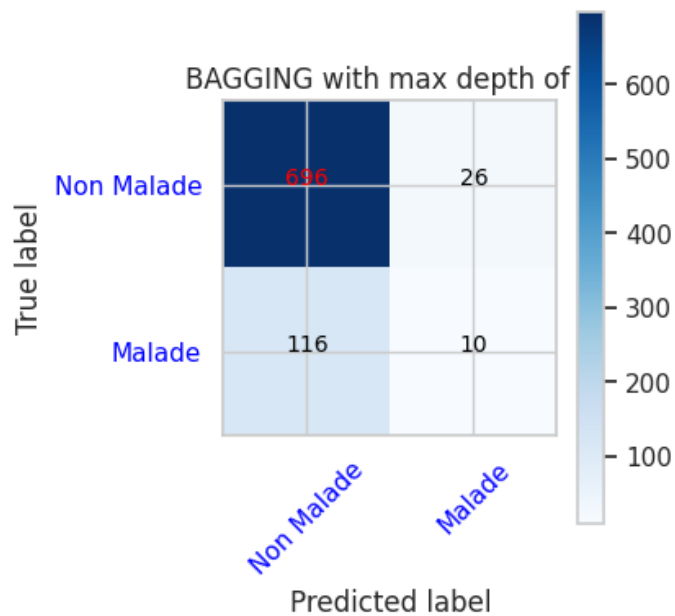


Source : groupe de travail

## 2-1-2- BAGGING

Avec un découpage de 80% - 20%

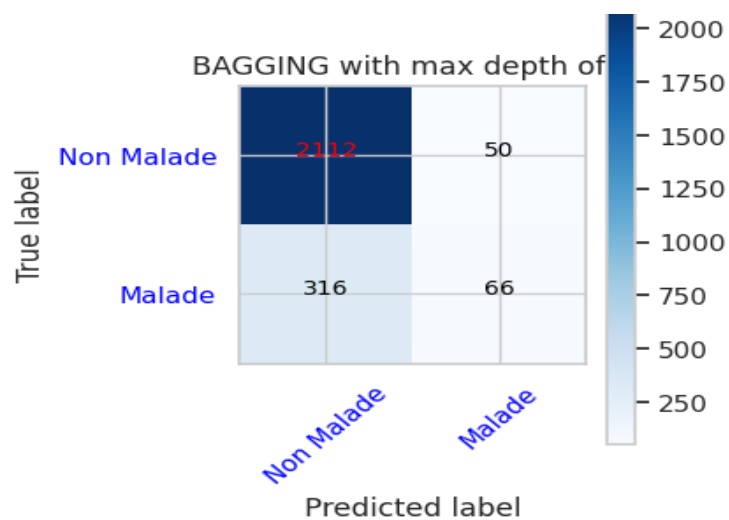
**Graphique 6 : matrice de confusion bagging 80-20**



**Source :** groupe de travail

Avec un découpage de 60% - 40%

**Graphique 7 : matrice de confusion bagging 60-40**



**Source :** groupe de travail

## 2-2- L'accuracy, precision et f1\_score

Au cours de notre travail, nous avons noté le accuracy pour chacun des modèles dans les proportions que nous avons également choisies avec trois autres métriques, **la précision, le f1-score, l'accuracy et la matrice de confusion** présentée plus haut.

Les chiffres obtenus sont alors :

### 2-2-1-Decision Tree

**Figure 11 : résultat score Decision tree**

```
precision: 36.36 %  
f1_score: 0.03  
Score_Test: 85.7 %  
Score_Training: 84.3 %
```

**Source :** groupe de travail

### 2-2-2- Bagging

En découpage 80% - 20%

**Figure 12 : Résultat score Bagging 80-20**

```
precision: 27.27 %  
f1_score: 0.11  
Score_Test: 85.0 %  
Score_Training: 84.8 %
```

**Source :** groupe de travail

En découpage 60% - 40%

**Figure 12 : Résultat score bagging 60-40**

```
precision: 56.9 %  
f1_score: 0.27  
Score_Test: 85.0 %  
Score_Training: 84.7 %
```

**Source :** groupe de travail



## CONCLUSION

Un modèle de machine learning lorsqu'il est entraîné, a pour but de pouvoir faire des prédictions des targets relatives aux features d'un ensemble de données précises. Le modèle doit donc être capable de pouvoir donné un résultat adéquat sur la base d'une référence que sont les données sur lesquelles elle a été entraîné au préalable. Pour évaluer le modèle, les métriques sont faites pour cette phase et dans notre cas, les évaluations sont plutôt déséquilibrées pour chacun des modèles que nous avons utilisés. Nous avons d'un côté en test le decision tree qui est plutôt intéressant en test et par contre en entraînement c'est le bagging dans les divisions 80-20 qui est meilleur. Ceci soulève donc cette exclamation : Retrouvons nous dans 10 ans pour en avoir le cœur net !





## GLOSSAIRE

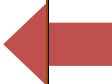
**Intelligence artificielle :** ensemble des théories et des techniques développant des programmes informatiques complexes capables de simuler certains traits de l'intelligence humaine (raisonnement, apprentissage...)

**Machine Learning :** sous-ensemble de l'intelligence artificielle (IA) qui vise à apprendre aux machines à tirer des enseignements des données et à s'améliorer avec l'expérience, au lieu d'être explicitement programmées pour le faire.

**Deep Learning :** Technologie basée sur des réseaux de neurones artificiels (en couches) permettant à une machine d'apprendre par elle-même, utilisée dans de nombreux domaines de l'intelligence artificielle (reconnaissance d'images, voiture autonome, diagnostic médical, etc.)

# TABLE DES MATIERES

<b>PARTIE 1 : PRESENTATION, TRAITEMENT ET ANALYSE DESCRIPTIVE DES DONNEES.....</b>	<b>12</b>
CHAPITRE 1 : PRESENTATION ET TRAITEMENT DES DONNEES.....	13
SECTION 1 : PRESENTATION DES DONNEES.....	14
SECTION 2 : TRAITEMENT DES DONNEES.....	16
CHAPITRE 2 : ANALYSE DESCRIPTIVE DES DONNEES.....	18
SECTION 1 : VISUALISATION DES DONNEES .....	19
SECTION 2 : RESULTATS DE L'ANALYSE DESCRIPTIVE.....	20
<b>PARTIE 2 : MODELES DE MACHINE LEARNING ET EVALUATION .....</b>	<b>22</b>
CHAPITRE 3 : MODELE DE MACHINE LEARNING .....	23
SECTION 1 : PRESENTATION THEORIQUE DES MODELES DE MACHINE LEARNING.....	24
1-1- Modèle decision tree.....	24
1-2- Modèle de Bagging.....	25
1-3- modèle de régression logistique.....	26
SECTION 2 : MISE EN ŒUVRE TECHNIQUE DES MODELES.....	27
2-1- Modèle decision tree.....	27
2-2- Modèle de Bagging.....	28
2-3- modèle de régression logistique.....	29
SECTION 3 : ANALYSE EN COMPOSANTES PRINCIPALES.....	30
CHAPITRE 4 : EVALUATION.....	34
SECTION 1 : PRESENTATION THEORIQUES DES METRIQUE.....	35
1-1- la précision.....	35
1-2- le f1-score.....	35
1-3- l'accuracy score.....	36
1-4- la matrice de confusion.....	36



SECTION 2 : TECHNICITE DES METRIQUES.....	37
2-1- la matrice de confusion .....	37
2-1-1- Decision Tree.....	37
2-1-2- Bagging.....	38
2-2- L'accuracy, la précision et le f1-score.....	39
2-2-1- Decision Tree.....	39
2-2-2- Bagging .....	39