



Nom et Prénom : Rida Nasser

Licence : Data Science

Année universitaire : 2023 - 2024

Matière : Visualisation des données

Titre du Projet : "Visualisation de la Prédiction d'Infarctus"

PLAN :

- 1.** Introduction
- 2.** Description du Projet
- 3.** Bibliothèques Python Importées
- 4.** Analyse des Visualisations obtenues
- 5.** Conclusion
- 6.** Références
- 7.** Annexe

1. Introduction :

Ce projet de visualisation de données repose sur l'analyse approfondie de l'ensemble de données "Prédiction d'Infarctus".

L'objectif principal de ce projet de visualisation de données est de découvrir des insights significatifs à partir de l'ensemble de données "Prédiction d'Infarctus".

2. Description du Projet :

Ce jeu de données, riche en informations médicales et démographiques, constitue une ressource essentielle dans la recherche sur la santé cardiovasculaire. Comprendre et prédire les attaques cardiaques revêt une importance primordiale dans le domaine des soins de santé contemporains, faisant de ce jeu de données à la fois pertinent et substantiel pour une analyse approfondie.

Ce projet vise à combler le fossé entre les données brutes et les insights exploitables, offrant ainsi une compréhension nuancée des risques d'attaque cardiaque et contribuant au discours plus large sur la santé cardiovasculaire.

À travers des techniques de visualisation stratégiques, je cherche à communiquer ces résultats de manière efficace.

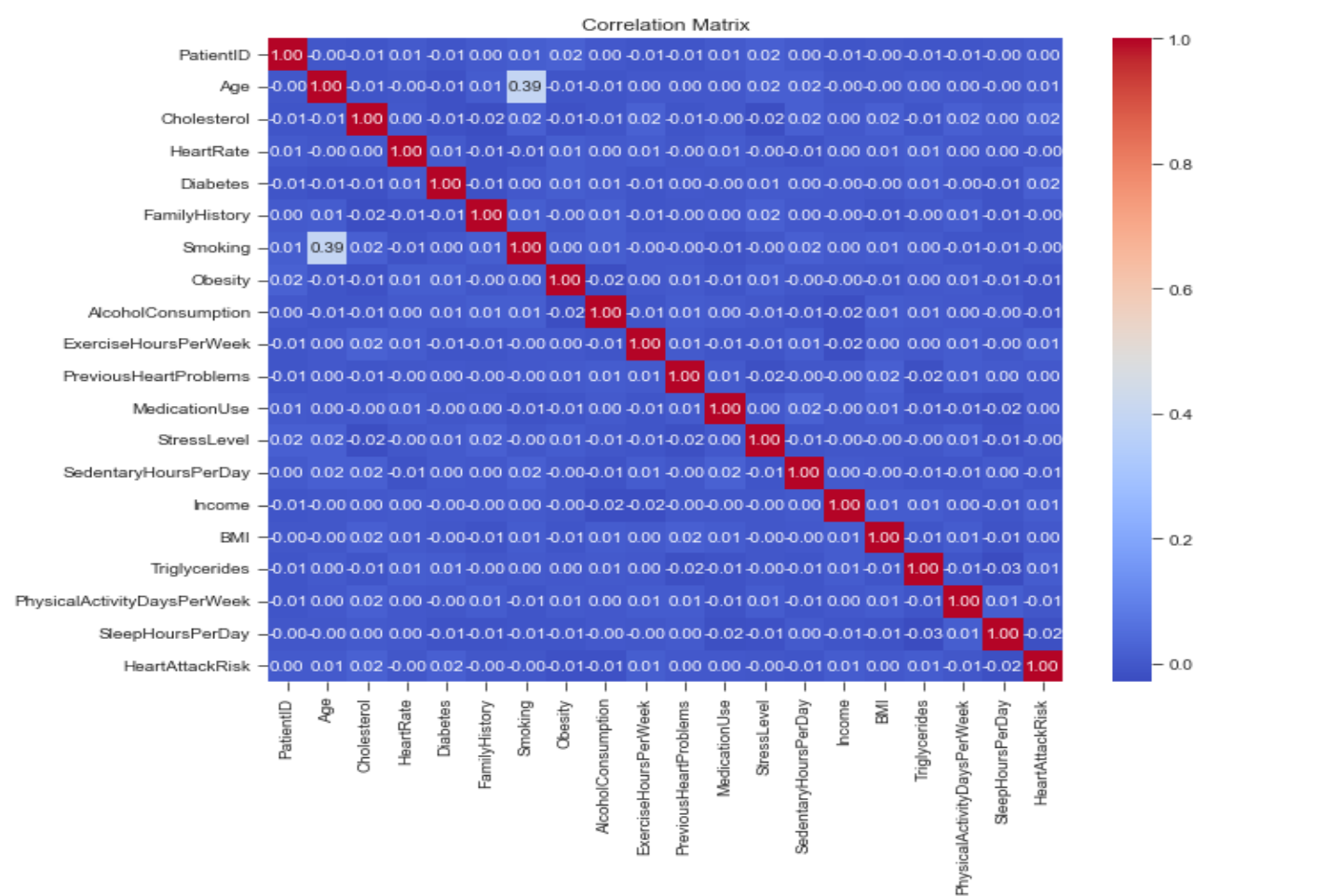
3. Bibliothèques Python Importées :

- Numpy
- Pandas
- Matplotlib.Pyplot
- Seaborn

4. Analyse des Visualisations Obtenues :

Dans cette section cruciale de l'étude, nous présentons une série de visualisations détaillées basées sur l'ensemble de données "Prédiction d'Infarctus". Chaque graphique offre un aperçu visuel des schémas et des tendances clés dans les données, accompagné d'une interprétation approfondie. Chaque visualisation est expliquée en détail, permettant aux lecteurs de comprendre les conclusions essentielles de notre analyse.

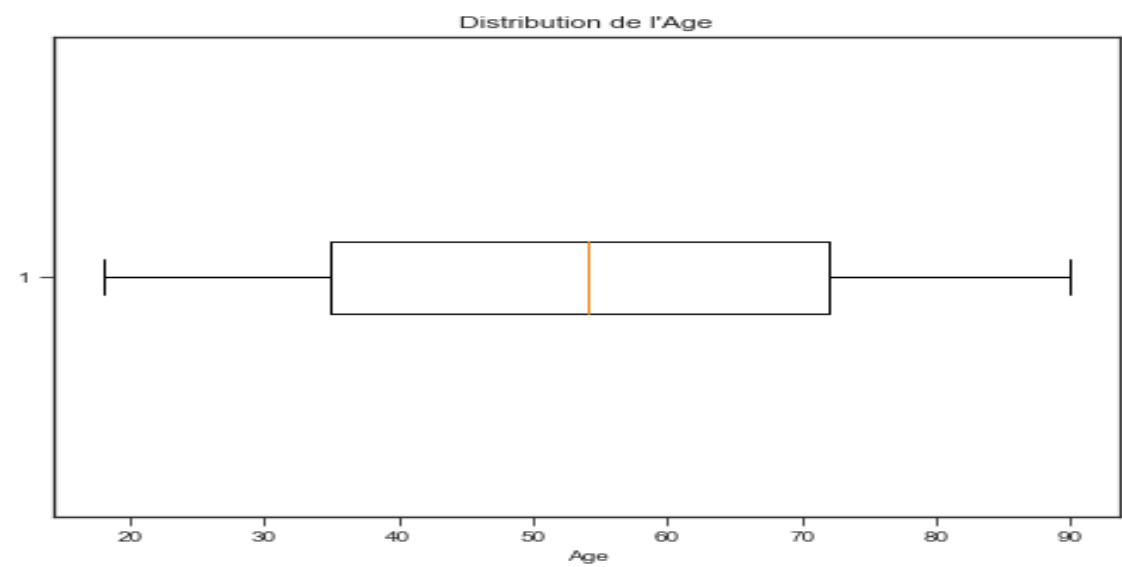
Carte Thermique de la Matrice de Corrélation



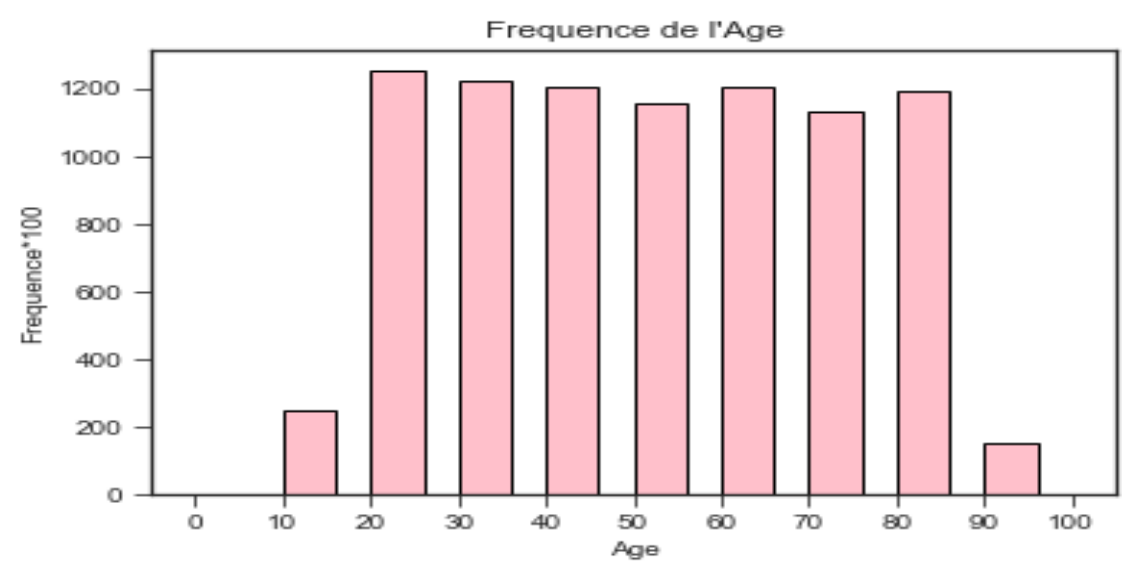
D’après la carte thermique donnée, on peut conclure que toutes les relations linéaires entre les attributs sont faibles, ou nulles. Il peut existe des relations non-linéaires ou quadratiques entre quelques attributs.

Visualisations des attributs numériques :

Boite à Moustache : Distribution de l’Age



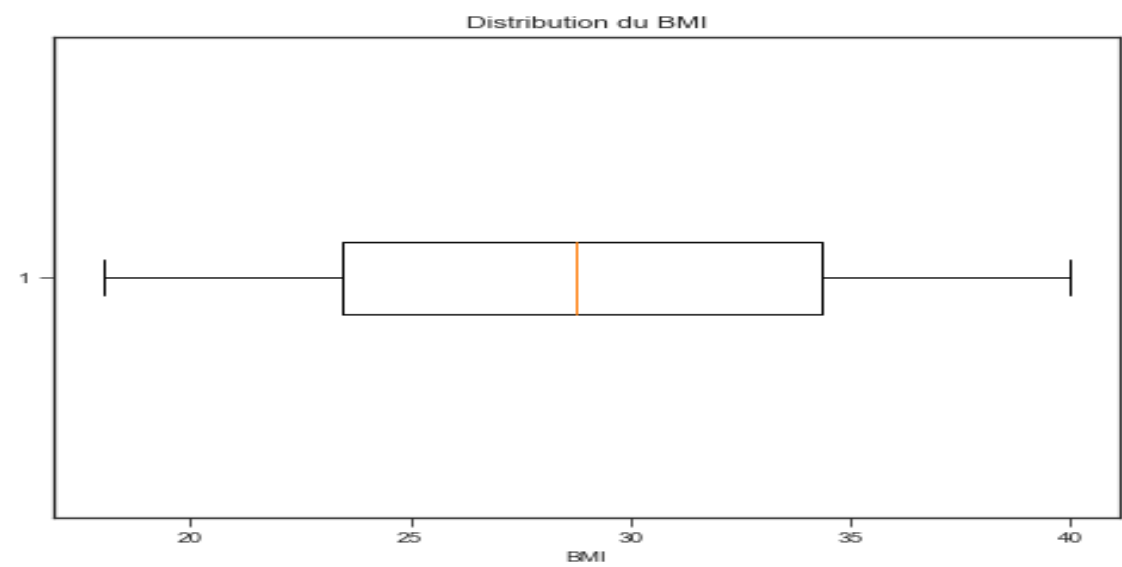
Histogramme : Distribution de l’Age



D’après l’histogramme et la boite à moustache, on peut conclure que les âges sont distribués entre 10 et 90. On conclut encore :

- $Q1 = 35$
- Médiane = 54
- $Q3 = 73$

Boite à Moustache : Distribution du BMI

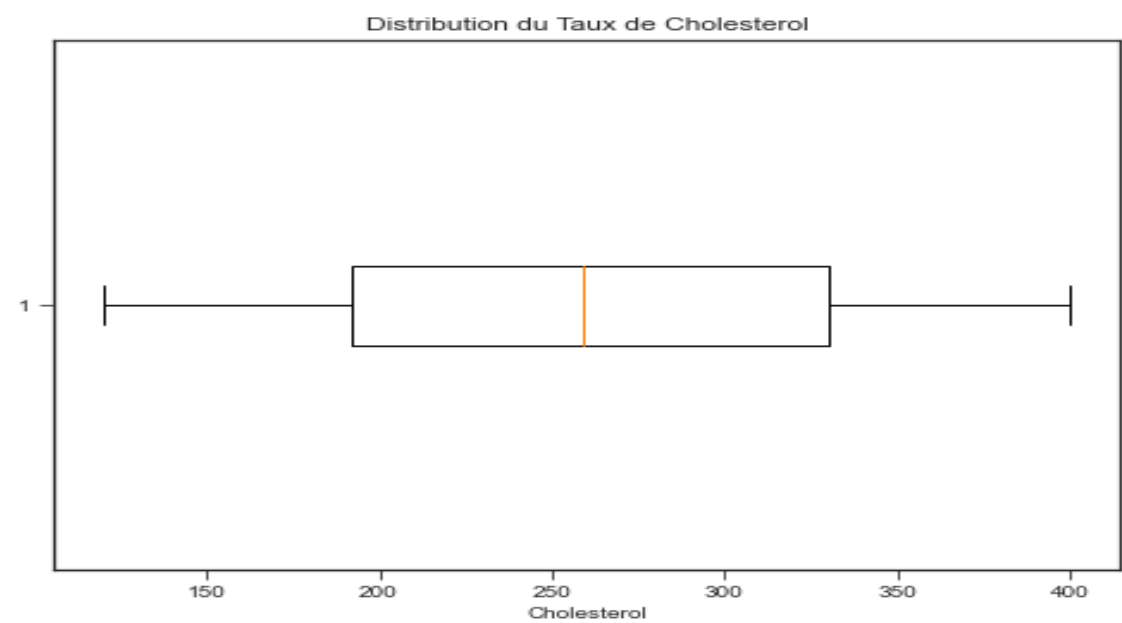


D’après la boîte à moustache, on déduit que le BMI est distribué entre ~18 et 40.

On conclut encore :

- $Q_1 = 23$
- Médiane = 28
- $Q_3 = 34$

Boîte à Moustache : Distribution du Taux de Cholestérol

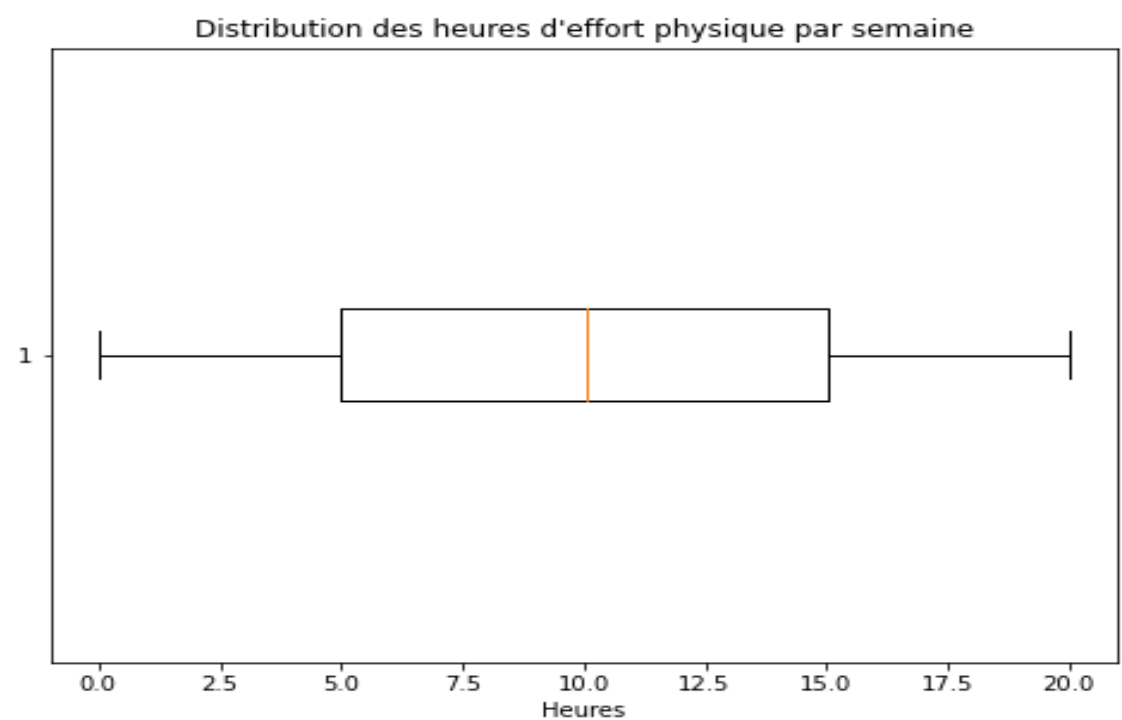


D’après la boîte à moustache, on déduit que le taux de cholestérol est distribué entre ~120 et 400.

On conclut encore :

- $Q_1 = 190$
- Médiane = 260
- $Q_3 = 330$

Boîte à Moustache : Distribution d’Heures d’Effort Physique par Semaine

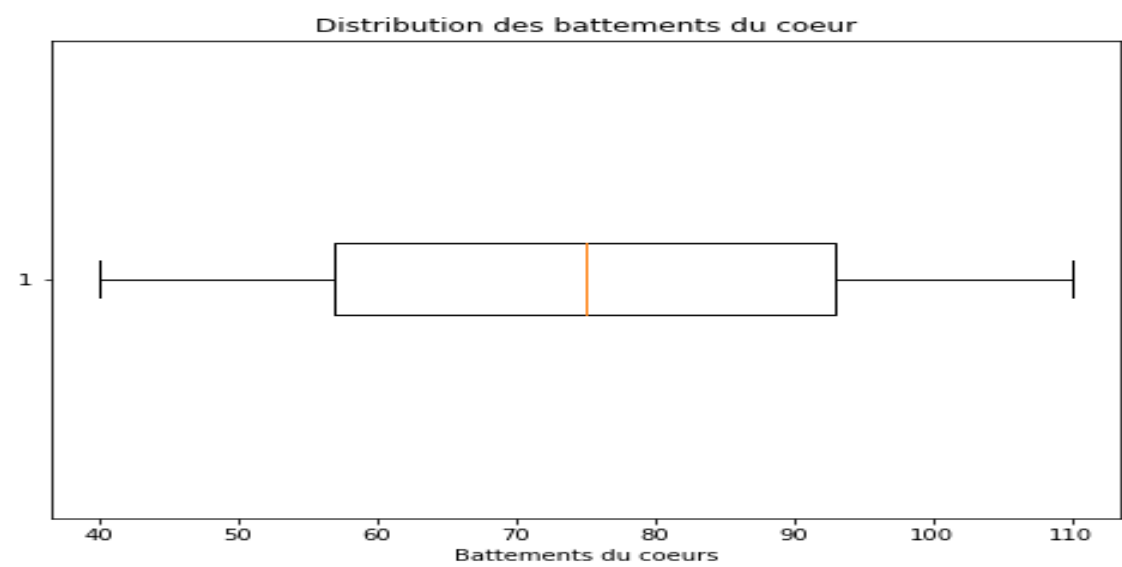


D’après la boîte à moustache, on déduit que les heures d’effort physique par semaine sont distribuées entre ~0 et 20.

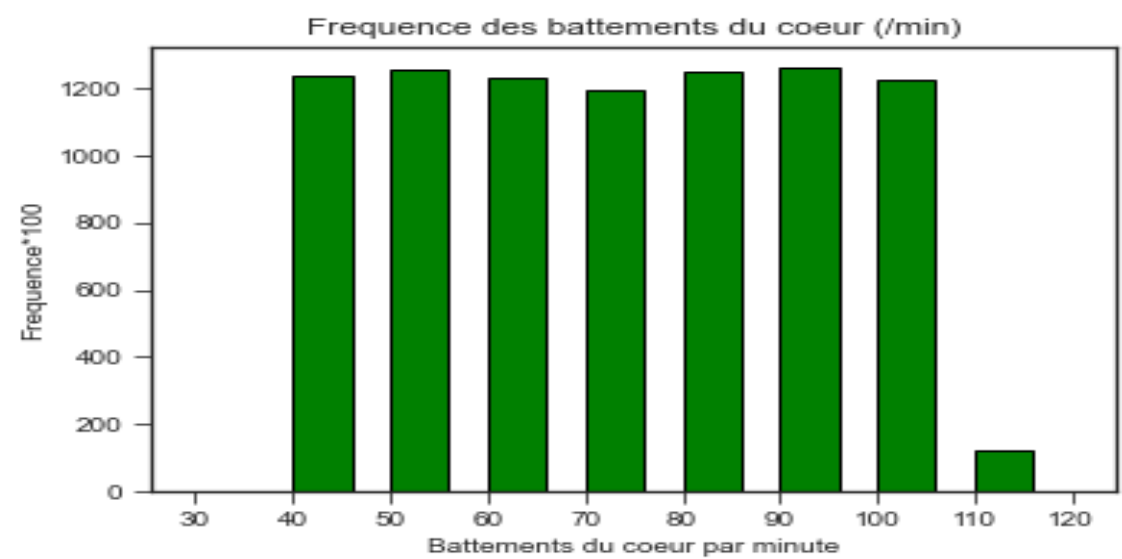
On conclut encore :

- $Q_1 = 5$
- Médiane = 10
- $Q_3 = 15$

Boite à Moustache : Distribution de la Fréquence Cardiaque



Histogramme : Distribution de la Fréquence Cardiaque



D’après la boite à moustache, on déduit que la fréquence cardiaque est distribuée entre 40 et 110.

On conclut encore :

- $Q1 = 57$
- Médiane = 75
- $Q3 = 93$

Boite à Moustache : Distribution des Revenus



D’après la boite à moustache, on déduit que les revenus sont distribués entre ~10,000 et 300,000.

On conclut encore :

- $Q1 = 80,000$
- Médiane = 155,000

- $Q3 = 225,000$

Boîte à Moustache : Distribution d’Heures de Sédentarité par Jour

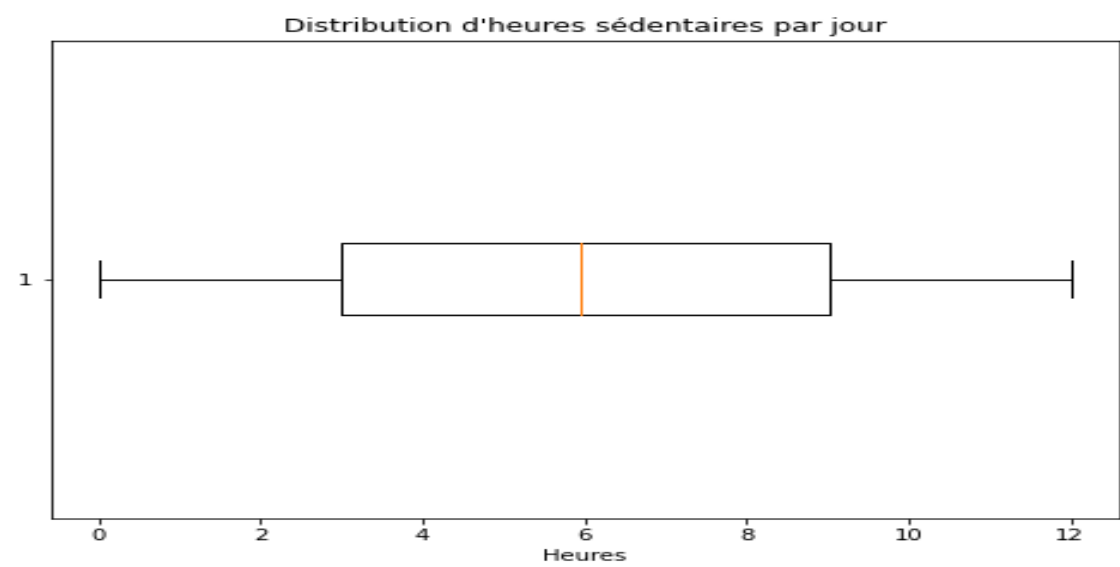
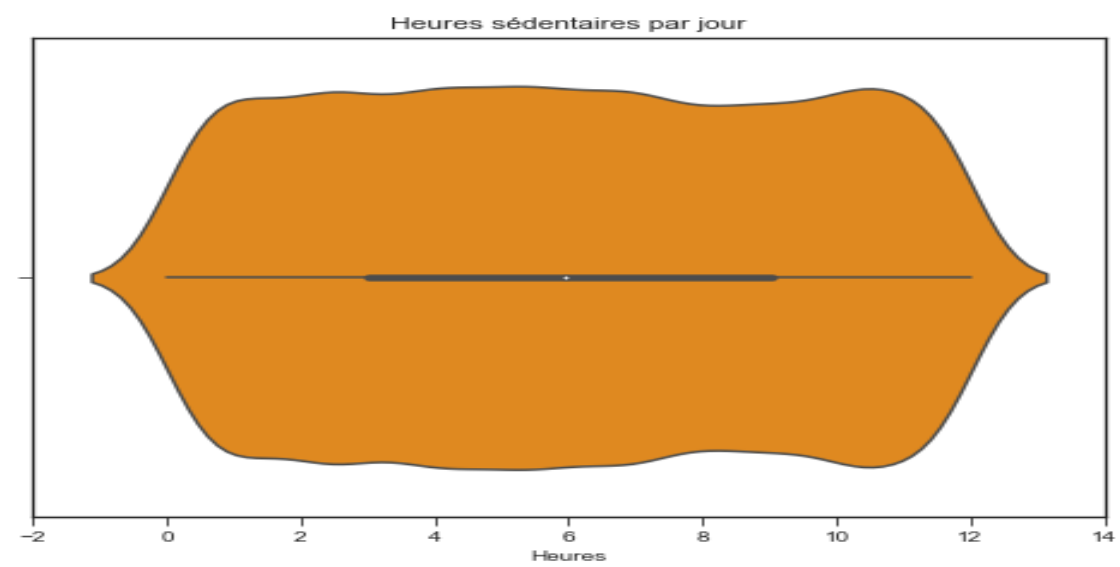


Diagramme en Violon : Distribution d’Heures de Sédentarité par Jour



D’après la boîte à moustache et le diagramme en violon, on déduit que les heures de sédentarité par jour sont distribuées entre 0 et 12.

On conclut encore :

- $Q1 = 3$
- Médiane = 6
- $Q3 = 9$

Boîte à Moustache : Distribution d’Heures de Sommeil par Jour

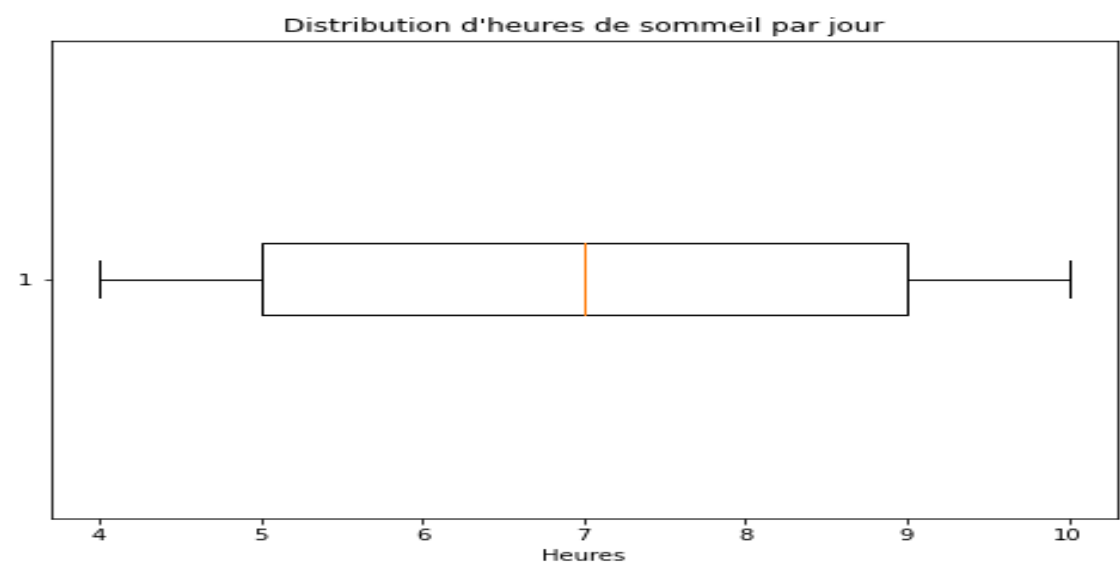
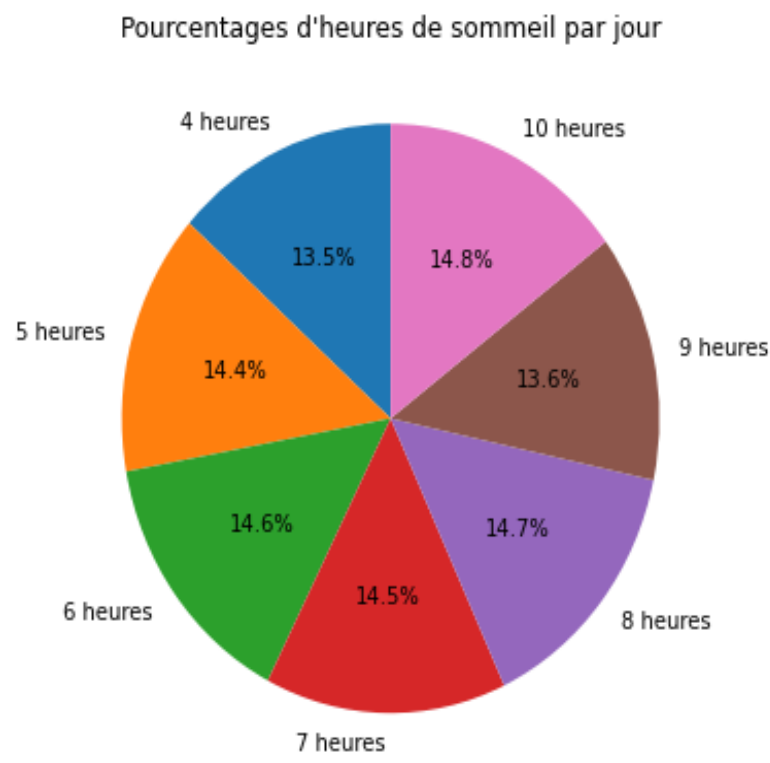


Diagramme Circulaire : Distribution d’Heures de Sommeil par Jour



D’après la boîte à moustache et le diagramme circulaire, on déduit que les heures de sommeil par jour sont distribuées entre 4 et 10.

On conclut encore :

- $Q1 = 5$
- Médiane = 7
- $Q3 = 9$

Boîte à Moustache : Distribution de Niveau du Stress

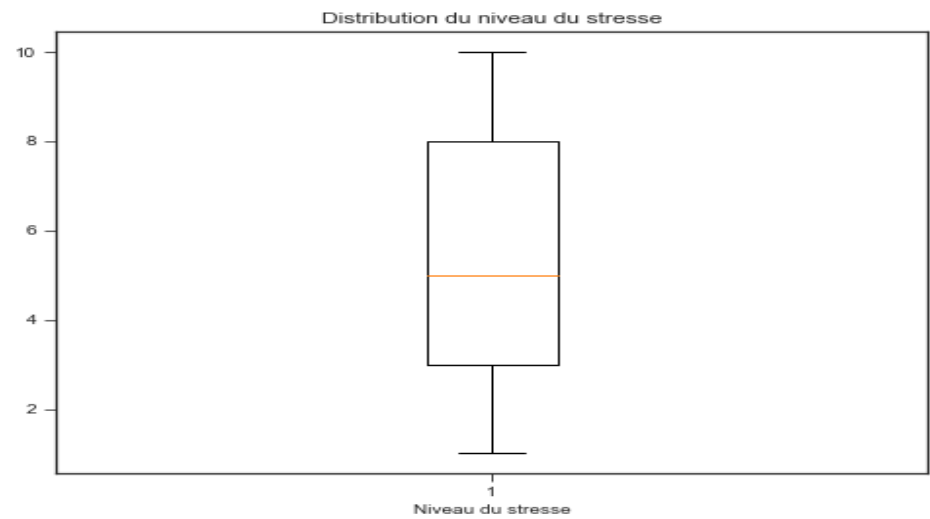
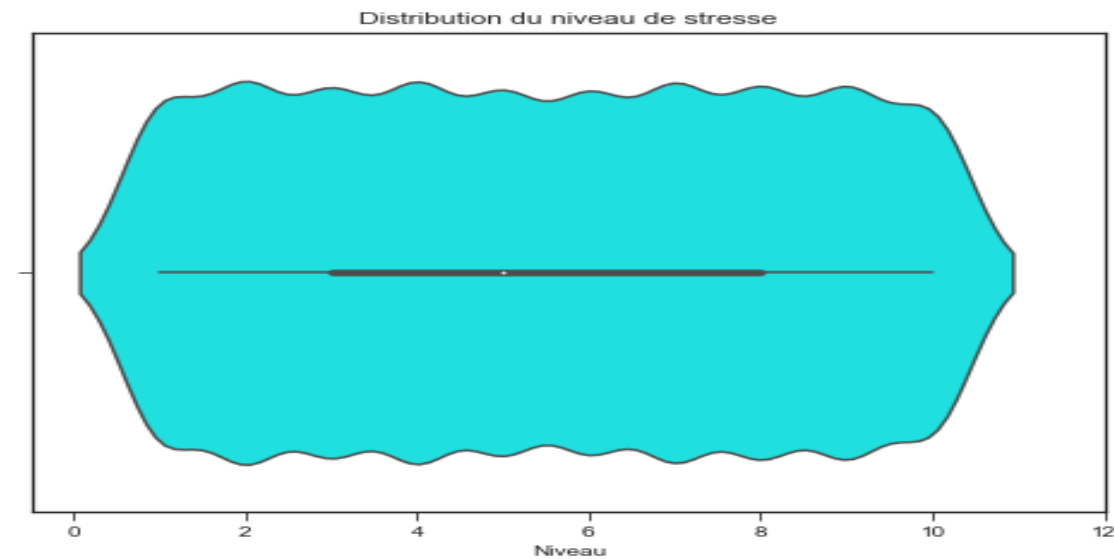


Diagramme en Violon : Distribution de Niveau du Stress

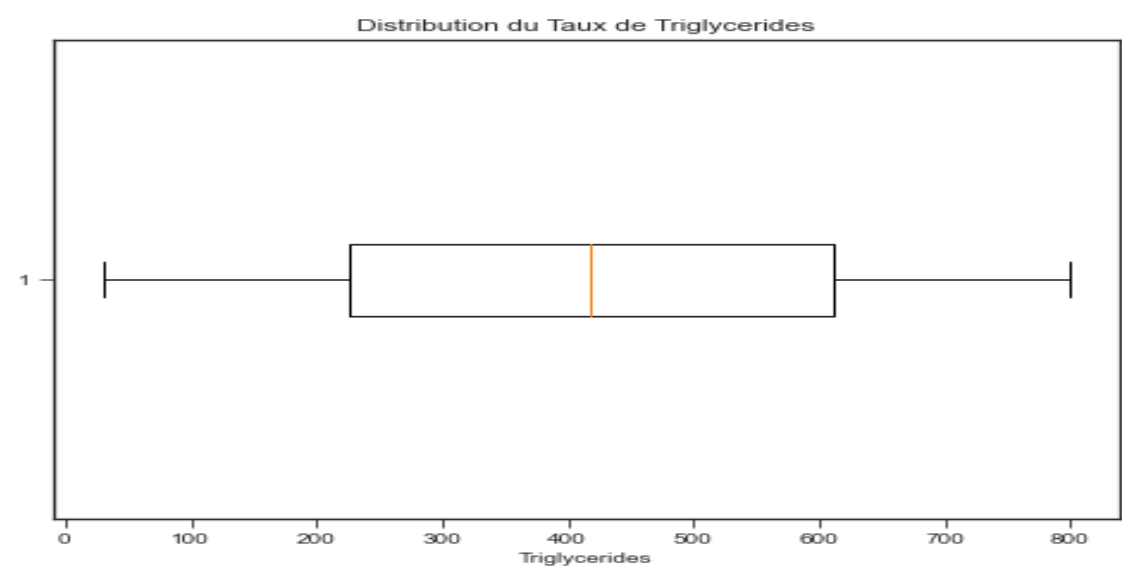


D’après la boîte à moustache et le diagramme en violon, on déduit que le niveau de stress est distribué entre 1 et 10.

On conclut encore :

- $Q1 = 3$
- Médiane = 5
- $Q3 = 8$

Boîte à Moustache : Distribution du Taux de Triglycérides

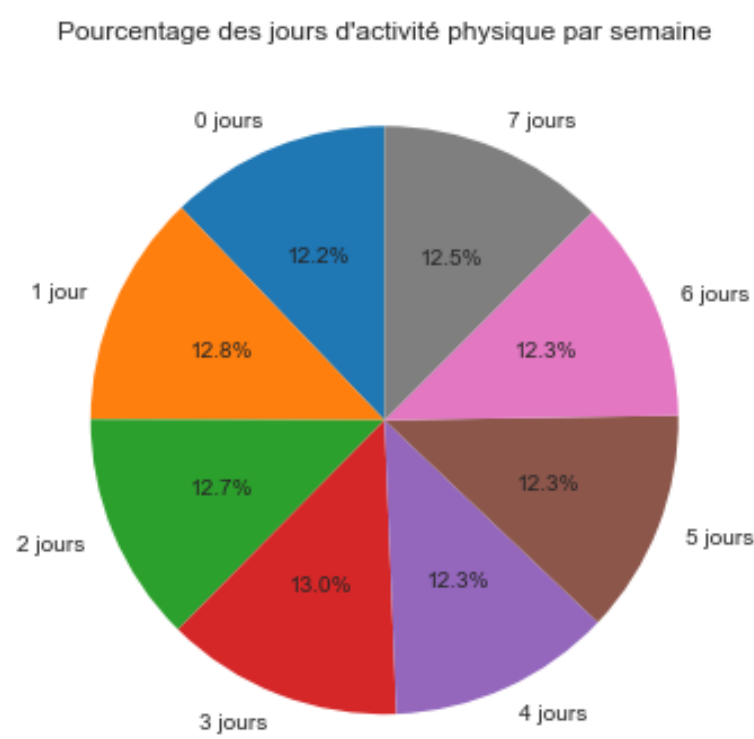


D’après la boîte à moustache, on déduit que le taux de triglycérides est distribué entre ~30 et 800.

On conclut encore :

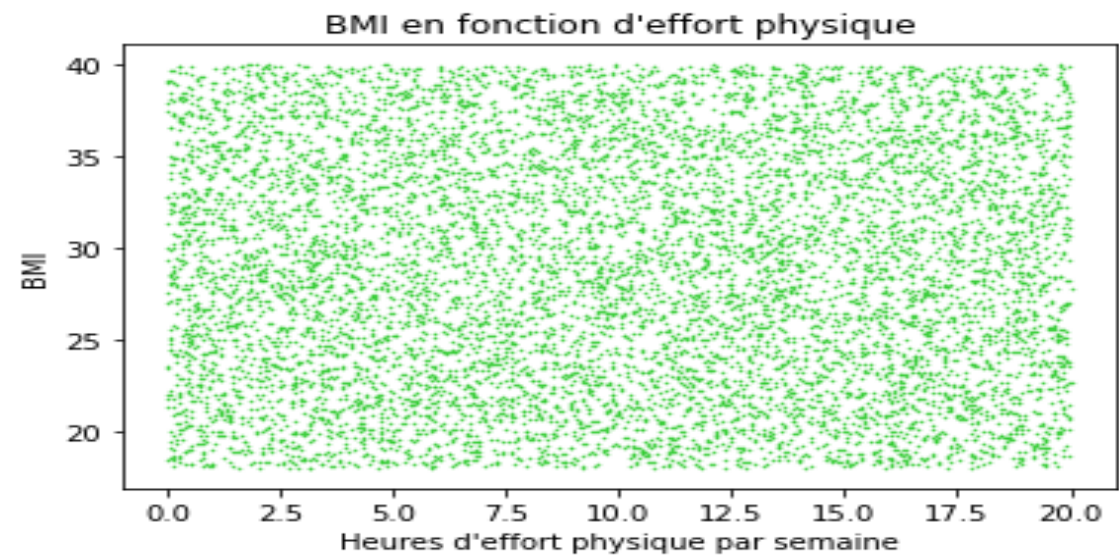
- $Q1 = 225$
- Médiane = 420
- $Q3 = 615$

Diagramme Circulaire : Distribution des Jours d’Activité Physique par Semaine



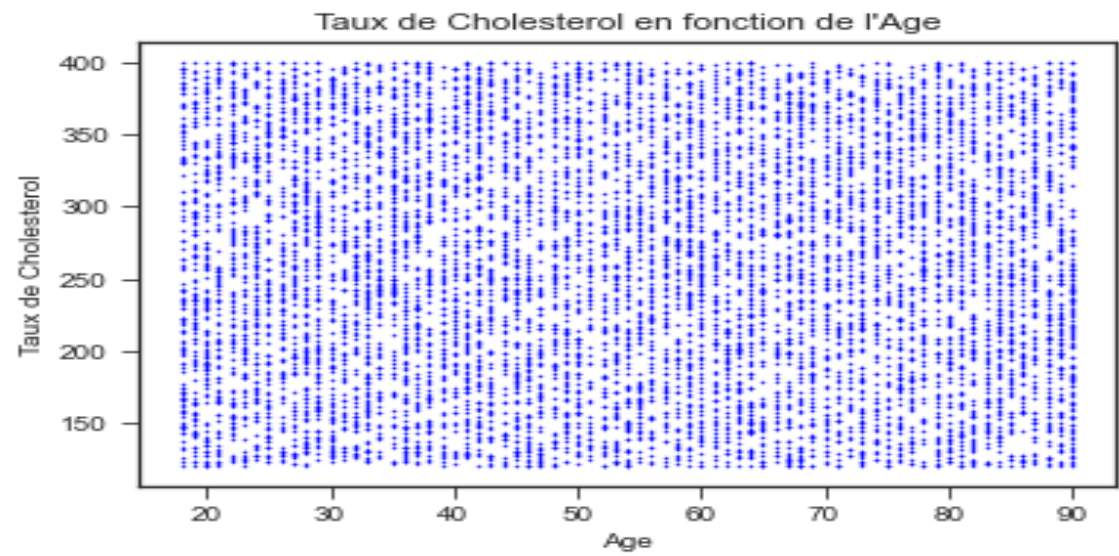
D’après le diagramme circulaire, on peut déduire que le nombre des jours d’activité physique par semaine est distribué également d’un pourcentage d’approximativement 12.5% entre les valeurs de 0 à 7.

Nuage de Points : BMI ~ Heures d’Activité Physique par Jour



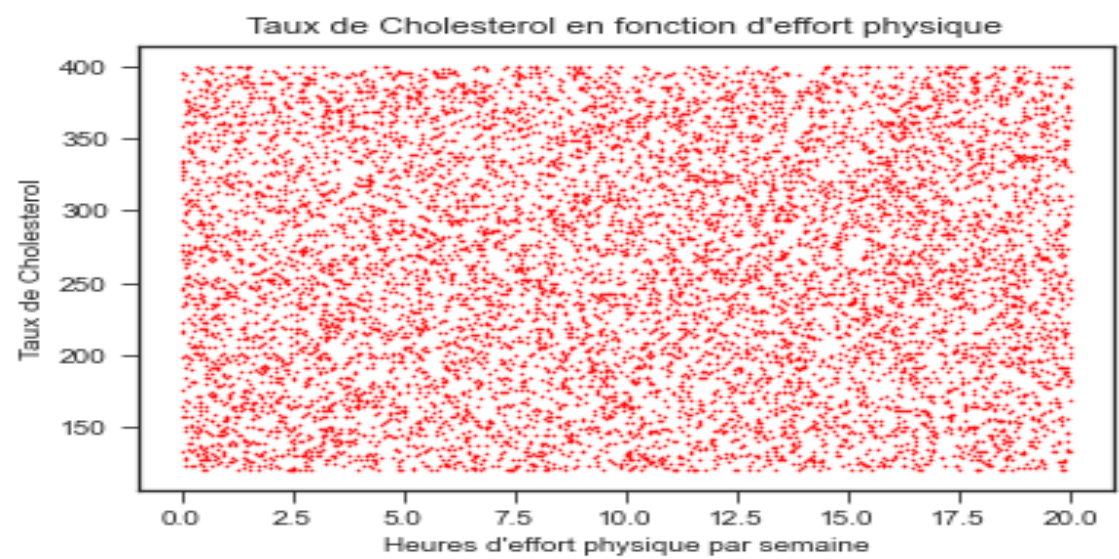
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre le BMI et les heures d’activité physique par semaine.

Nuage de Points : Cholestérol ~ Age



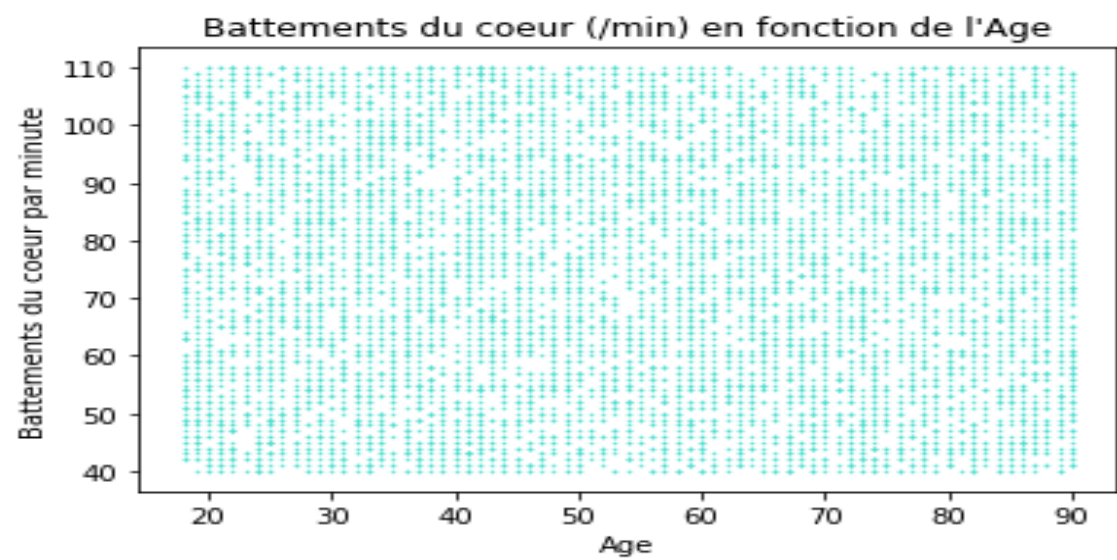
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre le taux de cholestérol et l’âge.

Nuage de Points : Cholestérol ~ Heures d’Activité Physique par Semaine



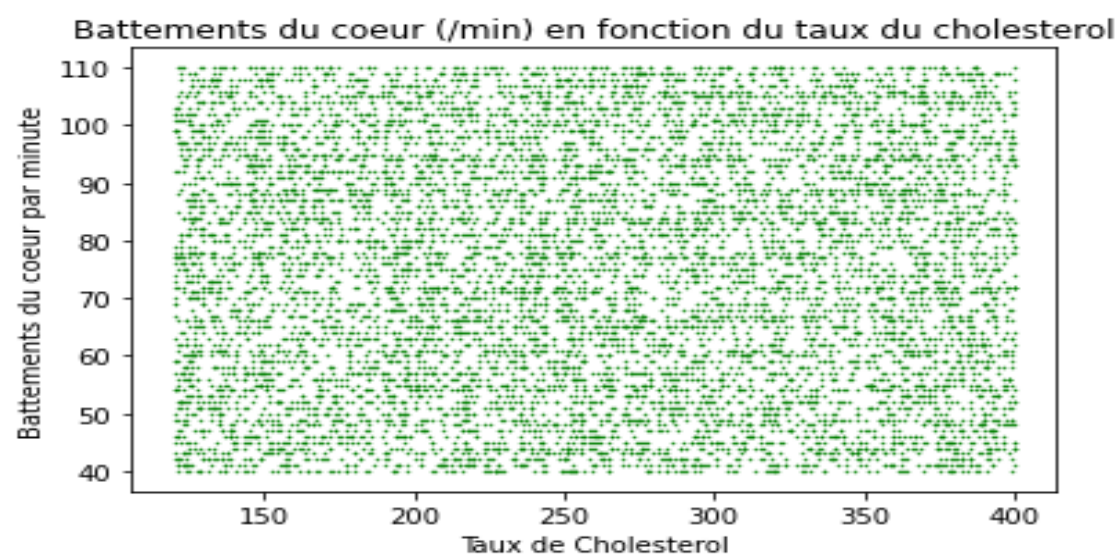
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre le taux de cholestérol et les heures d’activité physique par semaine.

Nuage de Points : Fréquence Cardiaque ~ Age



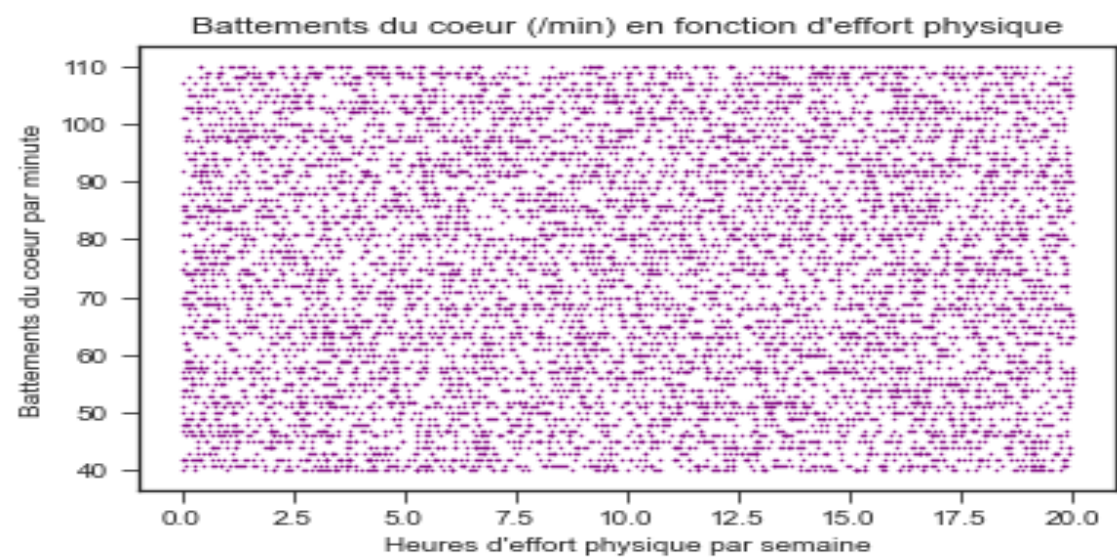
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre la fréquence cardiaque et l’âge.

Nuage de Points : Fréquence Cardiaque ~ Cholestérol



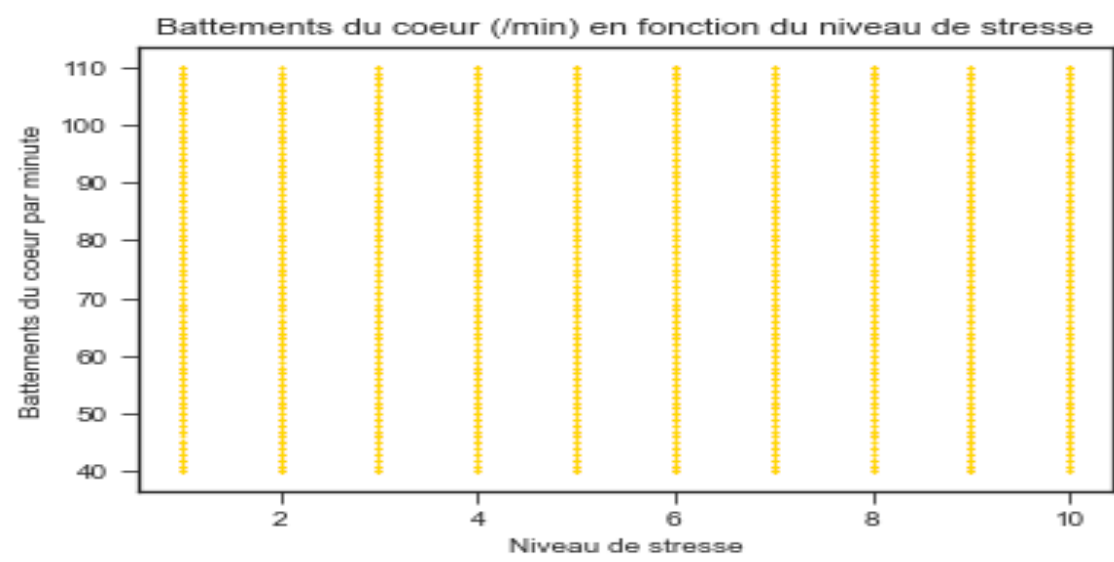
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre la fréquence cardiaque et le taux de cholestérol.

Nuage de Points : Fréquence Cardiaque ~ Heures d’Activité Physique par Semaine



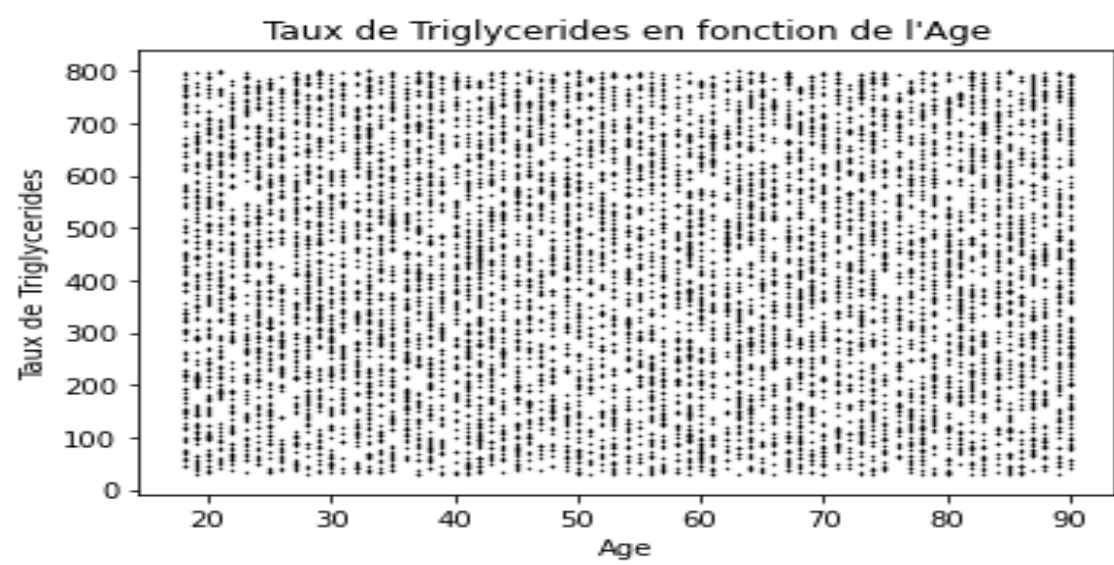
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre la fréquence cardiaque et les heures d’activité physique par semaine.

Nuage de Points : Fréquence Cardiaque ~ Niveau de Stress



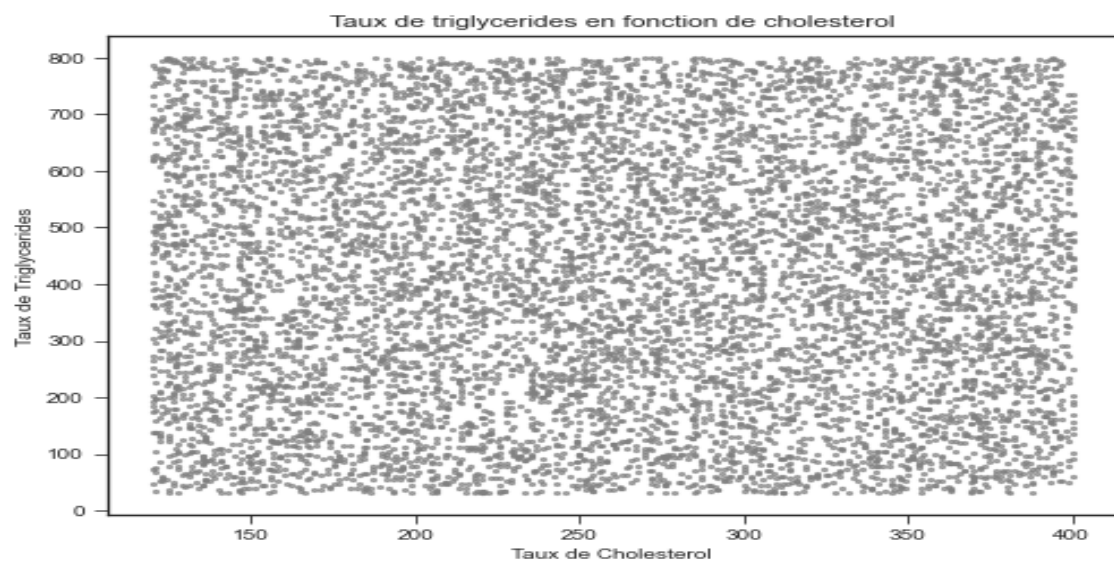
D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre la fréquence cardiaque et le niveau de stress.

Nuage de Points : Triglycérides ~ Age



D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre le taux de triglycérides et l’âge.

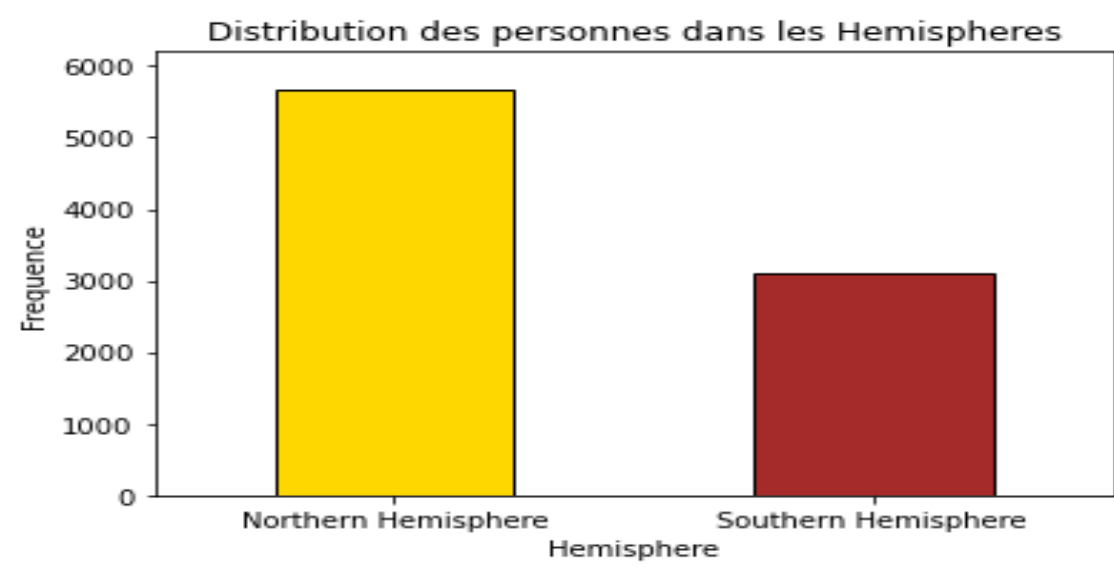
Nuage de Points : Triglycérides ~ Cholestérol



D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre le taux de triglycérides et le taux de cholestérol.

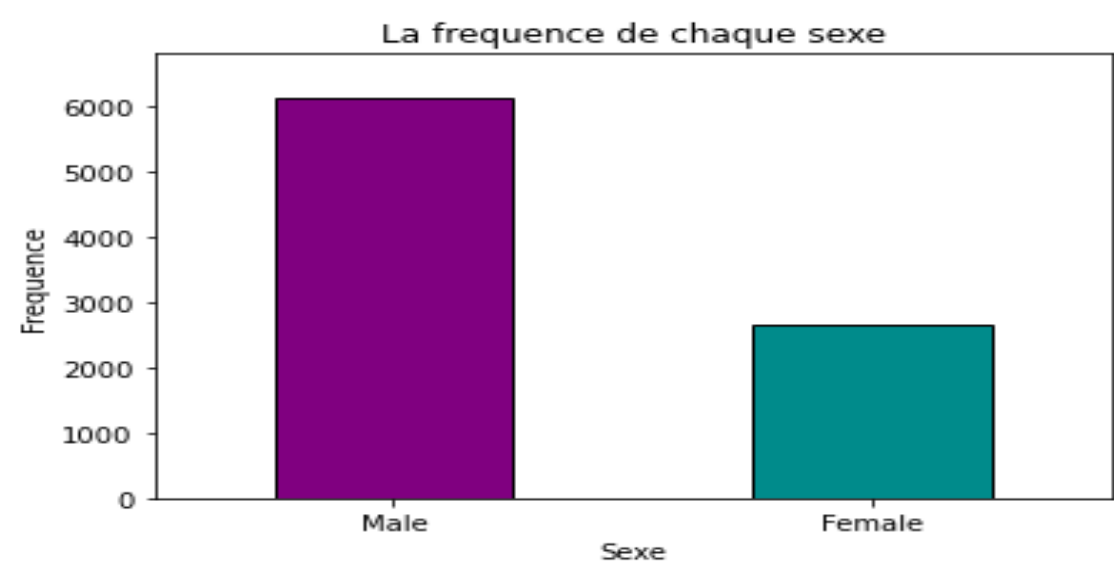
Visualisation des attributs binaires :

Diagramme en Bâtons : Distribution selon l’Hémisphère



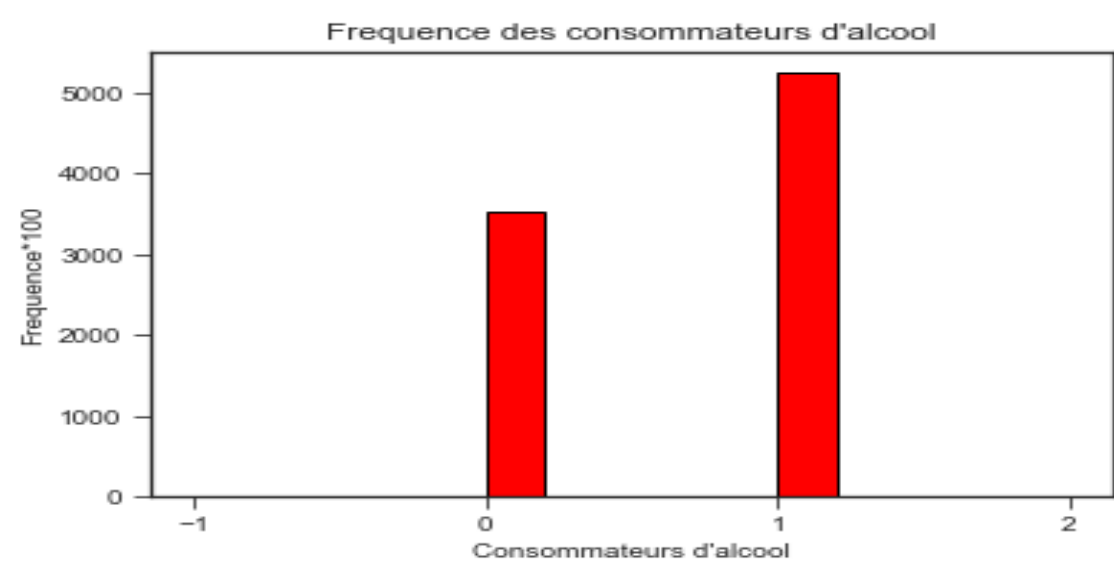
D’après ce diagramme en bâtons, on peut déduire que les patients qui habitent à l’hémisphère nord sont plus nombreux que les patients qui habitent à l’hémisphère sud. Alors, l’habitat peut avoir un impact direct ou indirect sur la probabilité d'une attaque cardiaque chez une personne.

Diagramme en Bâtons : Distribution selon le Sexe



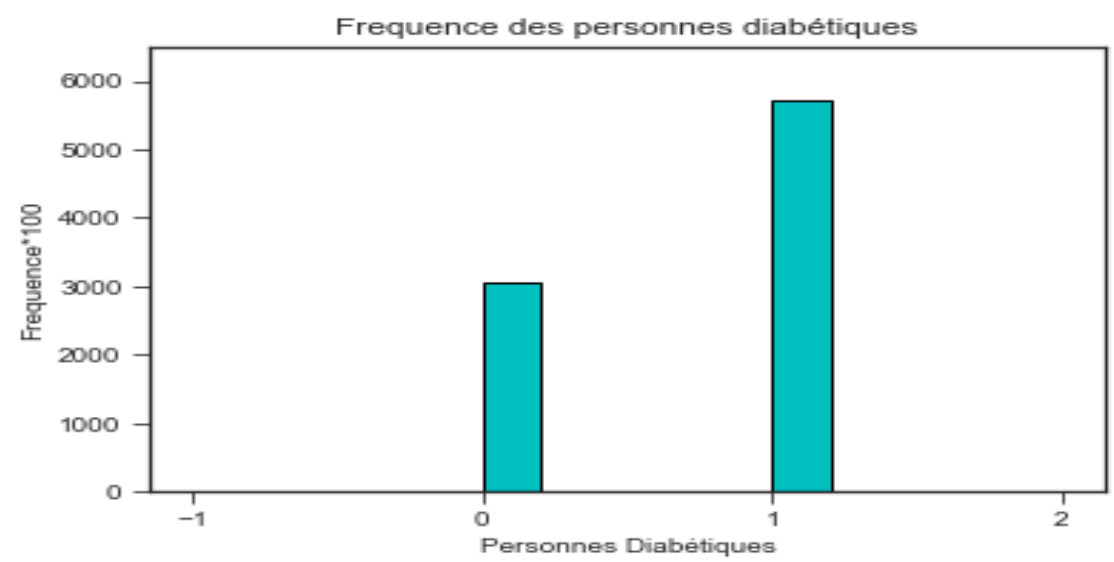
D’après ce diagramme en bâtons, on peut déduire que les patients males sont plus nombreux que les patients femelles. Alors, le sexe peut avoir un impact direct ou indirect sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon la Consommation d’Alcool



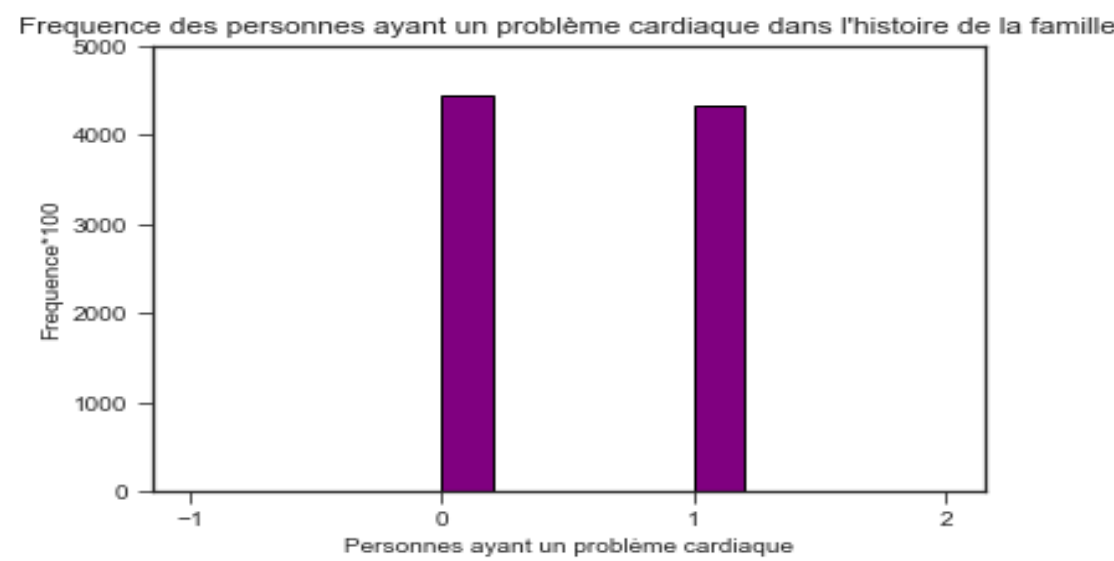
D’après cet histogramme, on peut déduire que les patients consommateurs d’alcool sont plus nombreux que les patients non-consommateurs d’alcool. Alors, la consommation d’alcool peut avoir un impact direct ou indirect sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon la Diabète



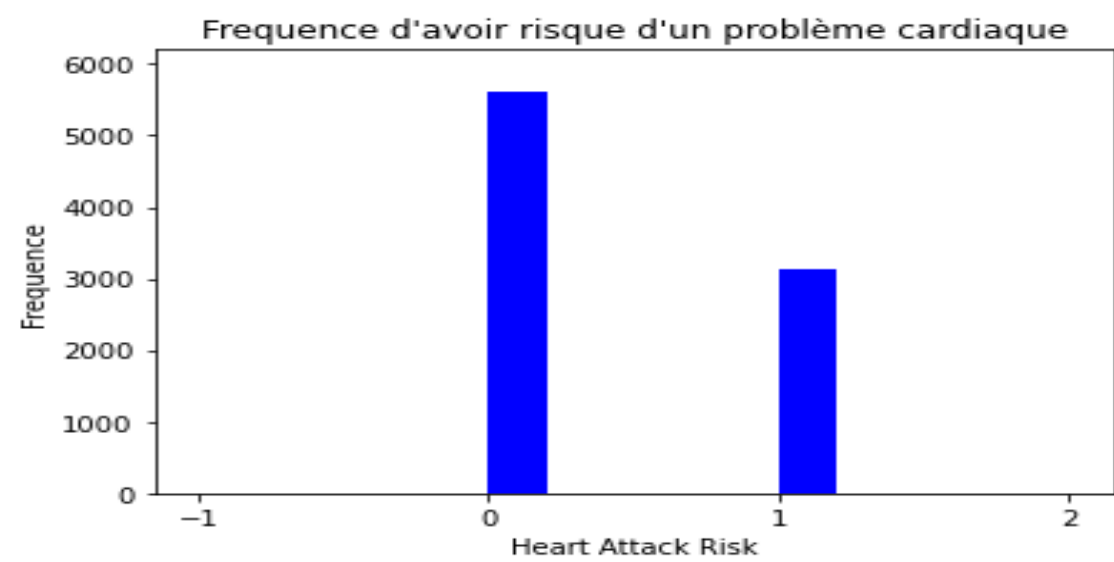
D’après cet histogramme, on peut déduire que les patients diabétiques sont plus nombreux que les patients non-diabétiques. Alors, le diabète peut avoir un impact direct ou indirect sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon l’Avoir d’Infarctus dans l’Histoire de la Famille



D’après cet histogramme, on peut déduire que le nombre des patients ayant un problème cardiaque dans l’histoire de la famille et le nombre des patients n’ayant pas sont approximativement égaux. Alors, l’avoir d’Infarctus dans l’histoire de la famille n'a pas d'impact sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon le Risque d'Attaque Cardiaque



D’après cet histogramme, on peut déduire que les patients n’ayant pas une attaque cardiaque sont plus nombreux que les patients l’en ayant.

Histogramme : Distribution selon l’Utilisation d’un Médicament

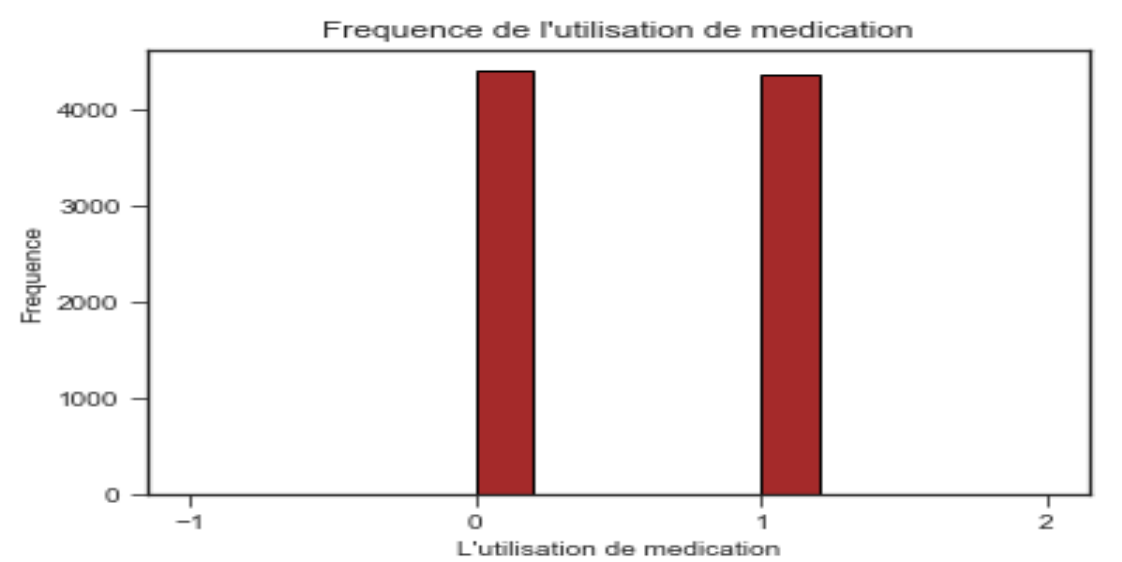
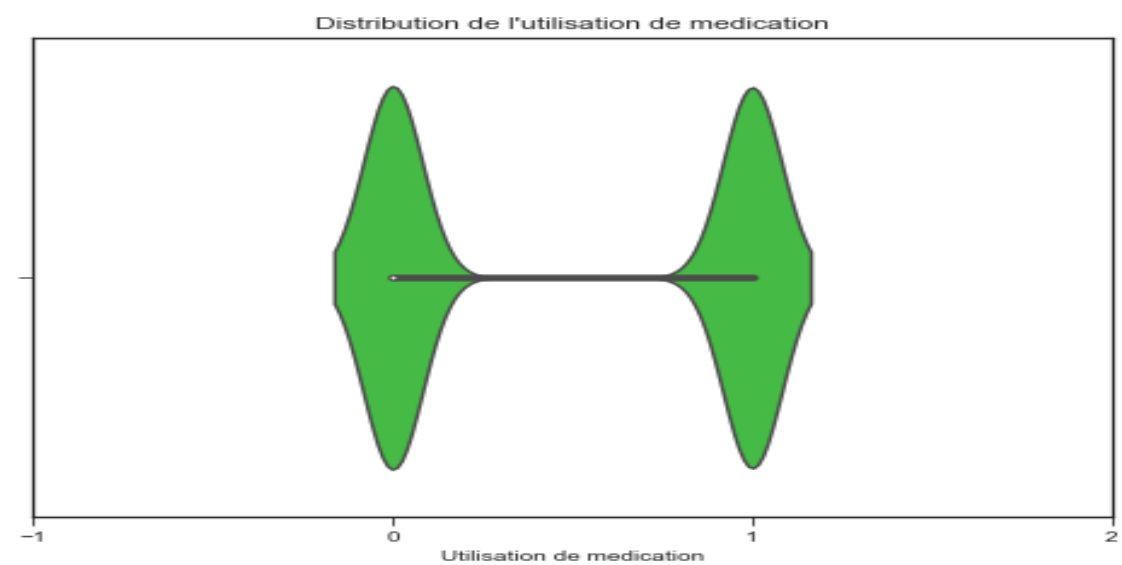
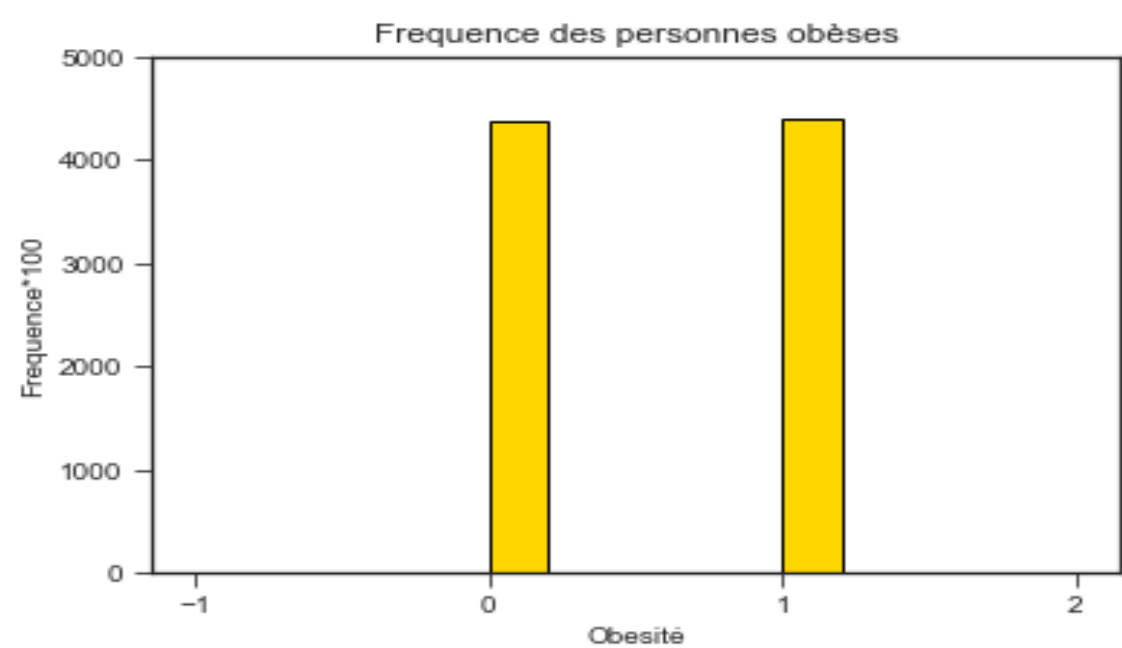


Diagramme en Violon : Distribution selon l’Utilisation d’un Médicament



D’après cet histogramme et le diagramme en violon, on peut déduire que le nombre des patients qui utilisent un médicament et le nombre des patients qui n’utilisent pas un médicament sont approximativement égaux. Alors, l’utilisation d’un médicament n'a pas d'impact sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon l’Obésité



D’après cet histogramme, on peut déduire que le nombre des patients obèses et le nombre des patients non-obèses sont approximativement égaux. Alors, l’obésité n'a pas d'impact sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon l’avoir des Problèmes Cardiaques Antérieurs

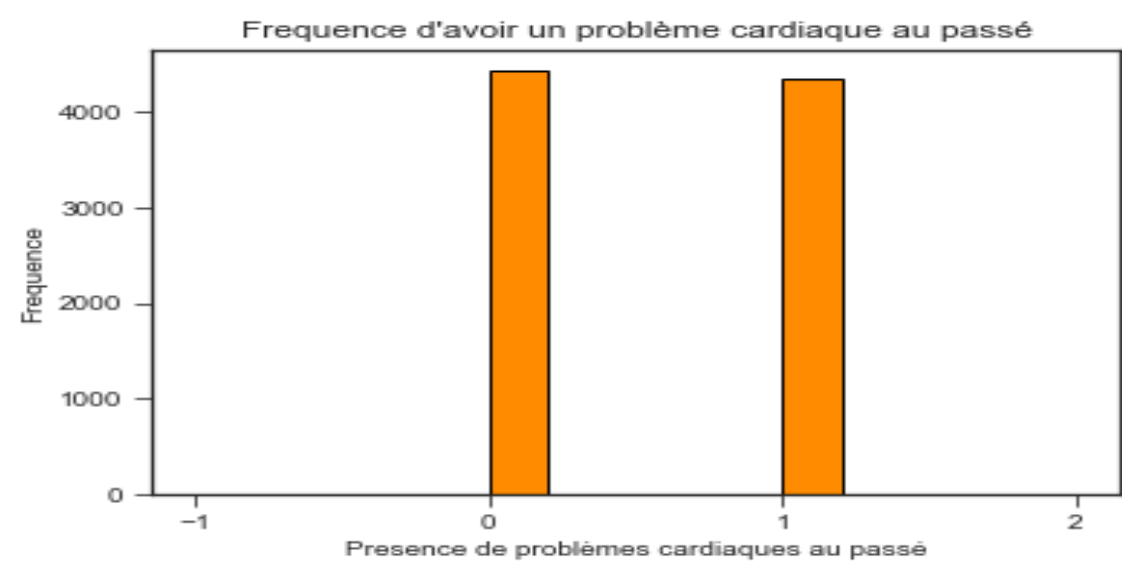
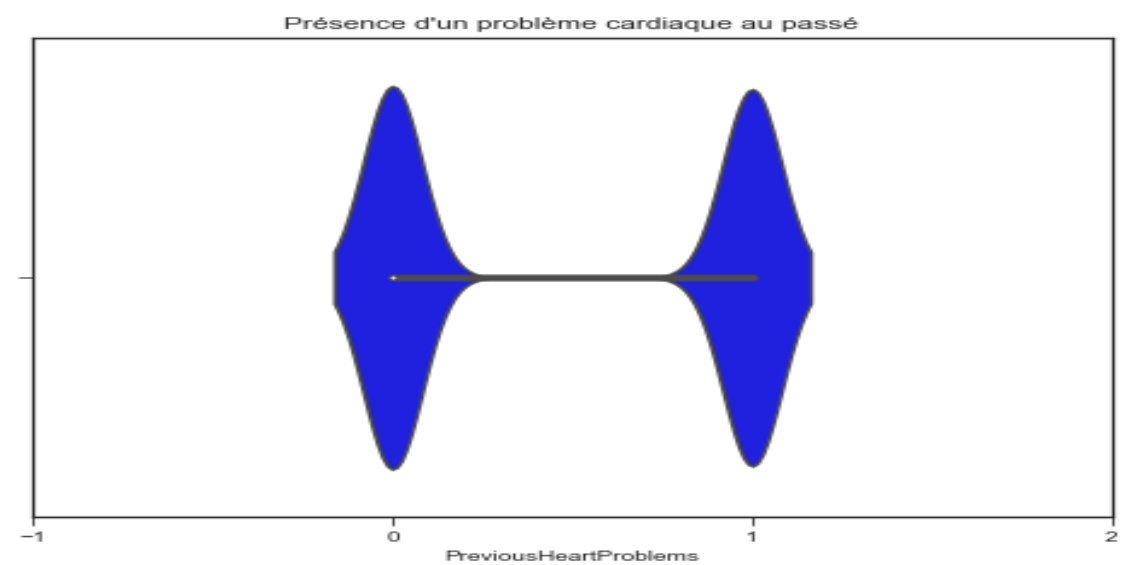
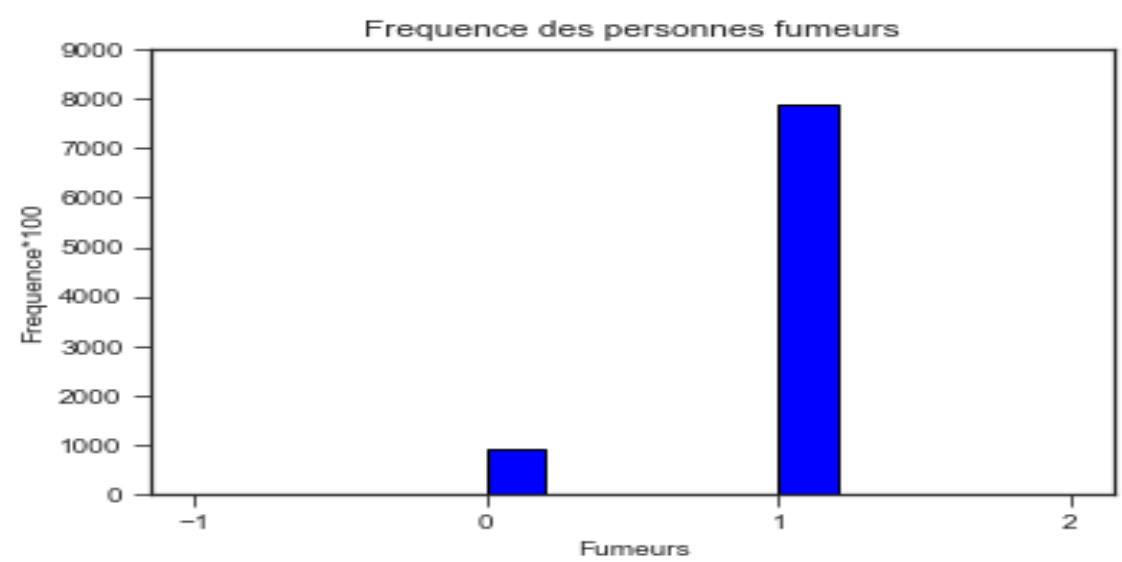


Diagramme en Violon : Distribution selon l’avoir des Problèmes Cardiaques Antérieurs



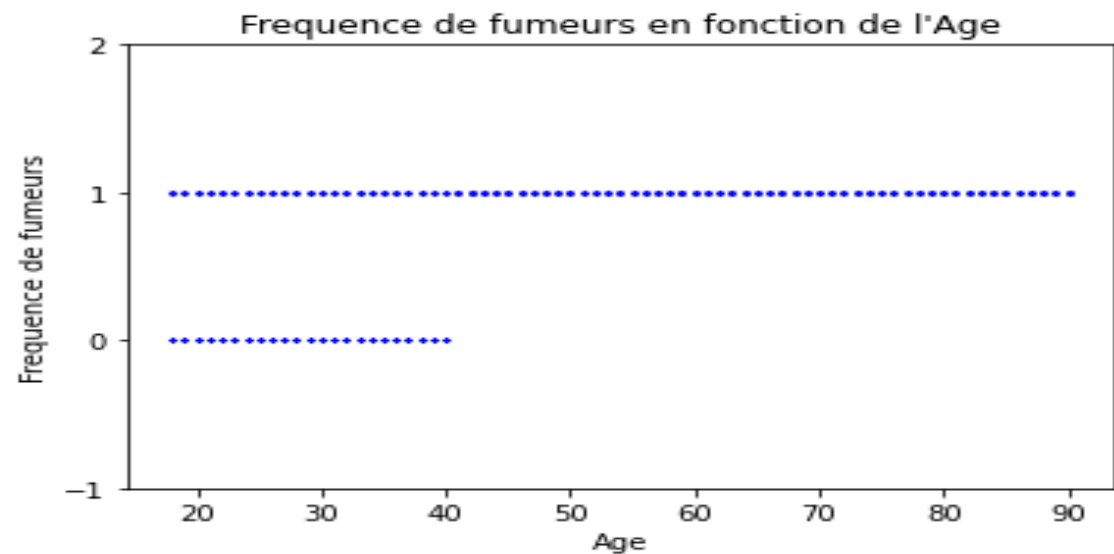
D’après cet histogramme et le diagramme en violon, on peut déduire que le nombre des patients qui ont eu un problème cardiaque antérieur et le nombre des patients qui n’ont eu pas un problème cardiaque antérieur sont approximativement égaux. Alors, avoir un problème cardiaque antérieur n'a pas d'impact sur la probabilité d'une attaque cardiaque chez une personne.

Histogramme : Distribution selon être Fumeur ou Non-Fumeur



D’après cet histogramme, on peut déduire que les patients fumeurs sont plus nombreux d’une manière significative que les patients non-fumeurs. Alors, la fumée peut avoir un impact direct ou indirect sur la probabilité d'une attaque cardiaque chez une personne.

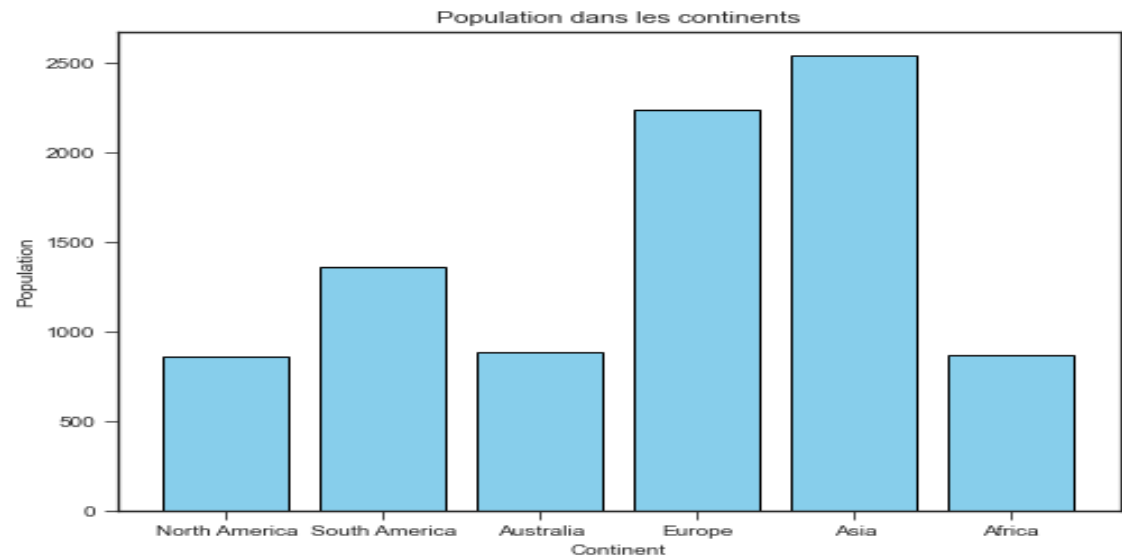
Nuage des Points : Fumer ~ Age



D’après ce nuage des points, on conclut qu’il n’y existe pas une relation linéaire ni une relation non-linéaire entre la fréquence des fumeurs et l’âge.

Visualisation des attributs nominaux :

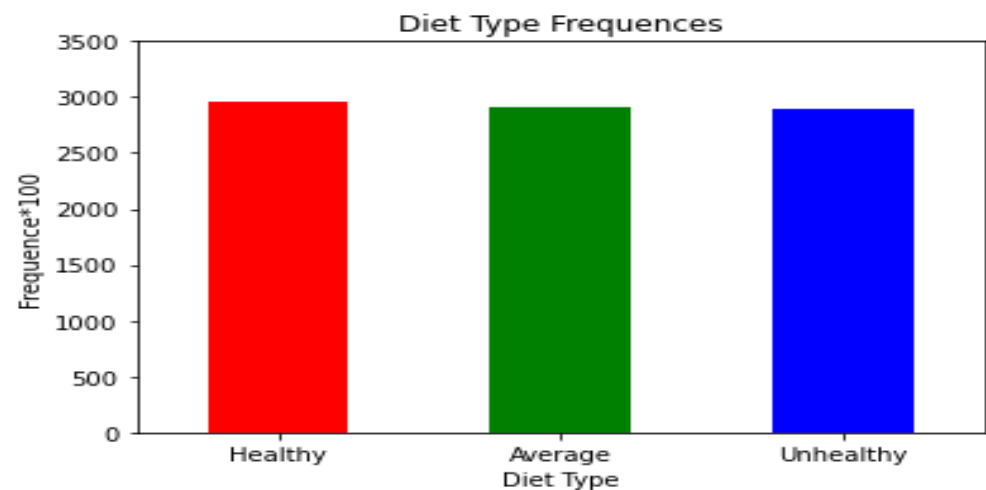
Diagramme en Bâtons : Distribution selon le Continent



D’après le diagramme en bâtons, on conclut que les patients sont distribués le plus dans Asie, Europe, et l’Amérique du Sud. Alors, l’habitat peut avoir un impact direct ou indirect sur la probabilité d'une attaque cardiaque chez une personne.

Visualisation des attributs ordinaux :

Diagramme en Bâtons : Distribution selon l’Alimentation



D’après le diagramme en bâtons, on conclut que le nombre des patients ayant un régime alimentaire sain, moyen, et peu sain est approximativement égal. Alors, l'alimentation n'a pas d'impact sur la probabilité d'une attaque cardiaque chez une personne.

5. Conclusion

Dans cette étude approfondie sur la prédiction des attaques cardiaques, nous avons exploré un éventail de facteurs de risque potentiels, allant de l'alimentation et du mode de vie aux antécédents médicaux. Nos interprétations ont révélé un constat intéressant : contrairement à nos attentes, il n'existe pas de relation linéaire ou non linéaire évidente entre un seul facteur et la probabilité d'une attaque cardiaque.

Cependant, les visualisations, notamment les histogrammes et les diagrammes à barres, ont mis en évidence des tendances significatives. En particulier, des facteurs tels que le tabagisme, le diabète, la consommation d'alcool, et d'autres variables liées au mode de vie semblent avoir un impact direct ou indirect sur la prédisposition à une attaque cardiaque. Ces résultats soulignent l'importance complexe des interactions entre divers éléments de notre mode de vie et de notre santé.

Il est essentiel de comprendre ces nuances pour élaborer des stratégies de prévention et de sensibilisation plus ciblées. Cette étude met en lumière la nécessité d'une approche holistique et personnalisée pour évaluer et réduire les risques d'attaques cardiaques, offrant ainsi des pistes précieuses pour les professionnels de la santé et les décideurs dans leur mission continue de promouvoir la santé cardiaque.

6. Références

- <https://www.geeksforgeeks.org/>
- <https://www.w3schools.com/>
- <https://stackoverflow.com/>

7. Annexe

```
import pandas as pd
```

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
import numpy as np
```

```
# Carte Thermique de la Matrice de Corrélation
```

```
data = pd.read_csv('C:/Users/User/Desktop/University/Semestre 5/Visualisation/Final Project/Heart Attack Prediction Dataset/heart_attack_prediction_dataset.csv')
```

```
numeric_columns = data.select_dtypes(include=['number'])
```

```
correlation_matrix = numeric_columns.corr()
```

```
non_numeric_columns = data.select_dtypes(exclude=['number'])
```

```
print(non_numeric_columns.columns)
```

```
df = data.select_dtypes(include=['number'])
```

```
correlation_matrix = df.corr()
```

```
pd.set_option('display.max_rows', None)
```

```
pd.set_option('display.max_columns', None)
```

```
print(correlation_matrix)
```

```
plt.figure(figsize=(10, 8))
```

```
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
```

```
plt.title('Correlation Matrix')
```

```
plt.show()
```

```
# Boite à Moustache : Distribution de l'Age
```

```
data = pd.read_csv("C:/Users/User/Desktop/University/Semestre 5/Visualisation/Final Project/Heart Attack Prediction Dataset/heart_attack_prediction_dataset.csv")
```

```
age = pd.Series(df.Age)
```

```
plt.figure(figsize=(8, 6))
```

```
plt.boxplot(age, vert=False)
```

```
plt.xlabel('Age')
```

```
plt.title('Distribution de l'Age')
```

```
plt.show()
```

```
# Histogramme : Distribution de l'Age
```

```
age = pd.Series(data.Age)
```

```
plt.style.use('seaborn-v0_8-ticks')
```

```
plt.hist(age,  
         width = 6,  
         edgecolor = "k",  
         color = "pink",  
         bins = [0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100])
```

```
plt.xlabel('Age')  
plt.ylabel('Frequence*100')  
plt.title('Frequence de l\'Age')  
plt.xticks([0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100])
```

```
plt.show()
```

```
# Boite à Moustache : Distribution du BMI
```

```
BMI = pd.Series(data.BMI)
```

```
plt.figure(figsize=(8, 6))  
plt.boxplot(BMI, vert=False)  
plt.xlabel('BMI')  
plt.title('Distribution du BMI')  
plt.show()
```

```
# Boite à Moustache : Distribution du Taux de Cholestérol
```

```
cholesterol = pd.Series(data.Cholesterol)
```

```
plt.figure(figsize=(8, 6))  
plt.boxplot(cholesterol, vert=False)  
plt.xlabel('Cholesterol')  
plt.title('Distribution du Taux de Cholesterol')  
plt.show()
```

```
# Boite à Moustache : Distribution d’Heures d’Effort Physique par Semaine
```

```
exerciseHours = pd.Series(data.ExerciseHoursPerWeek)
```

```
plt.figure(figsize=(8, 6))  
plt.boxplot(exerciseHours, vert=False)  
plt.xlabel('Heures')  
plt.title('Distribution des heures d\'effort physique par semaine')  
plt.show()
```

```
# Boite à Moustache : Distribution de la Fréquence Cardiaque
```

```
heartRate = pd.Series(data.HeartRate)
```

```
plt.figure(figsize=(8, 6))
plt.boxplot(heartRate, vert=False)
plt.xlabel('Battements du coeurs')
plt.title('Distribution des battements du coeur')
plt.show()
```

```
# Histogramme : Distribution de la Fréquence Cardiaque
```

```
heartRate = pd.Series(data.HeartRate)
plt.style.use('seaborn-v0_8-ticks')
```

```
plt.hist(heartRate,
         width = 6,
         edgecolor = "k",
         color = "g",
         bins = [30, 40, 50, 60, 70, 80, 90, 100, 110, 120])
```

```
plt.xlabel('Battements du coeur par minute')
plt.ylabel('Frequence*100')
plt.title('Frequence des battements du coeur (/min)')
plt.xticks([30, 40, 50, 60, 70, 80, 90, 100, 110, 120])
```

```
plt.show()
```

```
# Boite à Moustache : Distribution des Revenus
```

```
plt.boxplot(data.Income)
```

```
plt.title('Income')
plt.ylabel('Income Values')
```

```
plt.show()
```

```
# Boite à Moustache : Distribution d’Heures de Sédentarité par Jour
```

```
sedHrs = pd.Series(data.SedentaryHoursPerDay)
```

```
plt.figure(figsize=(8, 6))
plt.boxplot(sedHrs, vert=False)
plt.xlabel('Heures')
plt.title('Distribution d\'heures sédentaires par jour')
plt.show()
```

```
# Diagramme en Violon : Distribution d’Heures de Sédentarité par Jour
```

```
plt.figure(figsize=(8, 6))

sns.violinplot(x = 'SedentaryHoursPerDay',

              data = data,

              color = 'darkorange',

              bins = [-2, 0, 2, 4, 6, 8, 10, 12, 14])

plt.title('Heures sédentaires par jour')

plt.xlabel('Heures')

plt.xticks([-2, 0, 2, 4, 6, 8, 10, 12, 14])

plt.show()
```

```
# Boite à Moustache : Distribution d’Heures de Sommeil par Jour

sleepHours = pd.Series(data.SleepHoursPerDay)
```

```
plt.figure(figsize=(8, 6))

plt.boxplot(sleepHours, vert=False)

plt.xlabel('Heures')

plt.title('Distribution d\'heures de sommeil par jour')

plt.show()
```

```
# Diagramme Circulaire : Distribution d’Heures de Sommeil par Jour

labels = ['4 heures', '5 heures', '6 heures', '7 heures', '8 heures', '9 heures', '10 heures']
```

```
size4 = len(data[data['SleepHoursPerDay'] == 4])

size5 = len(data[data['SleepHoursPerDay'] == 5])

size6 = len(data[data['SleepHoursPerDay'] == 6])

size7 = len(data[data['SleepHoursPerDay'] == 7])

size8 = len(data[data['SleepHoursPerDay'] == 8])

size9 = len(data[data['SleepHoursPerDay'] == 9])

size10 = len(data[data['SleepHoursPerDay'] == 10])
```

```
sizes = [size4, size5, size6, size7, size8, size9, size10]
```

```
plt.figure(figsize=(8, 6))

plt.pie(sizes,

       labels=labels,

       autopct='%1.1f%%',

       startangle=90)

plt.title("Pourcentages d'heures de sommeil par jour")

plt.show()
```

```
# Boite à Moustache : Distribution de Niveau du Stress
```

```
stressLevel = pd.Series(data.StressLevel)
```

```
plt.figure(figsize=(8, 6))
```

```
plt.boxplot(stressLevel, vert=True)
```

```
plt.xlabel('Niveau du stress')
```

```
plt.title('Distribution du niveau du stress')
```

```
plt.show()
```

```
# Diagramme en Violon : Distribution de Niveau du Stress
```

```
plt.figure(figsize=(8, 6))
```

```
sns.violinplot(x = 'StressLevel',
```

```
              data = data,
```

```
              color = 'cyan',
```

```
              bins = [0, 2, 4, 6, 8, 10, 12])
```

```
plt.title('Distribution du niveau de stress')
```

```
plt.xlabel('Niveau')
```

```
plt.xticks([0, 2, 4, 6, 8, 10, 12])
```

```
plt.show()
```

```
# Boite à Moustache : Distribution du Taux de Triglycérides
```

```
triglycerides = pd.Series(data.Triglycerides)
```

```
plt.figure(figsize=(8, 6))
```

```
plt.boxplot(triglycerides, vert=False)
```

```
plt.xlabel('Triglycerides')
```

```
plt.title('Distribution du Taux de Triglycerides')
```

```
plt.show()
```

```
# Diagramme Circulaire : Distribution des Jours d'Activité Physique par Semaine
```

```
labels = ['0 jours', '1 jour', '2 jours', '3 jours', '4 jours', '5 jours', '6 jours', '7 jours']
```

```
size0 = len(data[data['PhysicalActivityDaysPerWeek'] == 0])
```

```
size1 = len(data[data['PhysicalActivityDaysPerWeek'] == 1])
```

```
size2 = len(data[data['PhysicalActivityDaysPerWeek'] == 2])
```

```
size3 = len(data[data['PhysicalActivityDaysPerWeek'] == 3])
```

```
size4 = len(data[data['PhysicalActivityDaysPerWeek'] == 4])
```

```
size5 = len(data[data['PhysicalActivityDaysPerWeek'] == 5])
```

```
size6 = len(data[data['PhysicalActivityDaysPerWeek'] == 6])
```

```
size7 = len(data[data['PhysicalActivityDaysPerWeek'] == 7])
```

```
sizes = [size0, size1, size2, size3, size4, size5, size6, size7]
```

```
plt.figure(figsize=(8, 6))
plt.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=90)
plt.title("Pourcentage des jours d'activité physique par semaine")
plt.show()
```

```
# Nuage de Points : BMI ~ Heures d'Activité Physique par Jour
exerciseHours = pd.Series(data.ExerciseHoursPerWeek)
BMI = pd.Series(data.BMI)
```

```
plt.scatter(exerciseHours,
            BMI,
            s = 1,
            marker = '.',
            c = 'limegreen')
```

```
plt.title("BMI en fonction d'\effort physique")
plt.xlabel("Heures d'effort physique par semaine")
plt.ylabel("BMI")
```

```
plt.show()
```

```
# Nuage de Points : Cholestérol ~ Age
age = pd.Series(data.Age)
cholesterol = pd.Series(data.Cholesterol)
```

```
plt.scatter(age,
            cholesterol,
            s = 1,
            marker = '.',
            c = 'b')
```

```
plt.title("Taux de Cholesterol en fonction de l'Age")
plt.xlabel("Age")
plt.ylabel("Taux de Cholesterol")
```

```
plt.show()
```

```
# Nuage de Points : Cholestérol ~ Heures d'Activité Physique par Semaine
exerciseHours = pd.Series(data.ExerciseHoursPerWeek)
cholesterol = pd.Series(data.Cholesterol)
```

```
plt.scatter(exerciseHours,
            cholesterol,
```



```
s = 1,  
marker = '.',  
c = 'r')
```

```
plt.title("Taux de Cholesterol en fonction d\'effort physique")  
plt.xlabel("Heures d'effort physique par semaine")  
plt.ylabel("Taux de Cholesterol")  
  
plt.show()
```

```
# Nuage de Points : Fréquence Cardiaque ~ Age  
age = pd.Series(data.Age)  
heartRate = pd.Series(data.HeartRate)
```

```
plt.scatter(age,  
            heartRate,  
            s = 1,  
            marker = '.',  
            c = 'turquoise')
```

```
plt.title("Battements du coeur (/min) en fonction de l\'Age")  
plt.xlabel("Age")  
plt.ylabel("Battements du coeur par minute")  
  
plt.show()
```

```
# Nuage de Points : Fréquence Cardiaque ~ Cholestérol  
heartRate = pd.Series(data.HeartRate)  
cholesterol = pd.Series(data.Cholesterol)
```

```
plt.scatter(cholesterol,  
            heartRate,  
            s = 1,  
            marker = '.',  
            c = 'g')
```

```
plt.title("Battements du coeur (/min) en fonction du taux du cholesterol")  
plt.xlabel("Taux de Cholesterol")  
plt.ylabel("Battements du coeur par minute")  
  
plt.show()
```

```
# Nuage de Points : Fréquence Cardiaque ~ Heures d’Activité Physique par Semaine
```

```
exerciseHours = pd.Series(data.ExerciseHoursPerWeek)
heartRate = pd.Series(data.HeartRate)
```

```
plt.scatter(exerciseHours,
            heartRate,
            s = 1,
            marker = '.',
            c = 'purple')

plt.title("Battements du coeur (/min) en fonction d\'effort physique")
plt.xlabel("Heures d\'effort physique par semaine")
plt.ylabel("Battements du coeur par minute")

plt.show()
```

Nuage de Points : Fréquence Cardiaque ~ Niveau de Stress

```
heartRate = pd.Series(data.HeartRate)
stressLevel = pd.Series(data.StressLevel)

plt.scatter(stressLevel,
            heartRate,
            s = 1,
            marker = '.',
            c = 'gold')

plt.title("Battements du coeur (/min) en fonction du niveau de stresse")
plt.xlabel("Niveau de stresse")
plt.ylabel("Battements du coeur par minute")

plt.show()
```

Nuage de Points : Triglycérides ~ Age

```
age = pd.Series(data.Age)
triglycerides = pd.Series(data.Triglycerides)

plt.scatter(age,
            triglycerides,
            s = 1,
            marker = '.',
            c = 'k')

plt.title("Taux de Triglycerides en fonction de l\'Age")
plt.xlabel("Age")
```

```
plt.ylabel("Taux de Triglycerides")
```

```
plt.show()
```

```
# Nuage de Points : Triglycérides ~ Cholestérol
```

```
cholesterol = pd.Series(data.Cholesterol)
```

```
triglycerides = pd.Series(data.Triglycerides)
```

```
plt.figure(figsize=(8, 6))
```

```
plt.scatter(cholesterol,
```

```
            triglycerides,
```

```
            alpha=0.7,
```

```
            c='grey',
```

```
            s = 5)
```

```
plt.xlabel('Taux de Cholesterol')
```

```
plt.ylabel('Taux de Triglycerides')
```

```
plt.title('Taux de triglycerides en fonction de cholesterol')
```

```
plt.show()
```

```
# Diagramme en Bâtons : Distribution selon l’Hémisphère
```

```
colors = ['gold', 'brown']
```

```
hemisphereCounts = data['Hemisphere'].value_counts()
```

```
ax = hemisphereCounts.plot(kind = 'bar',
```

```
                             color = colors,
```

```
                             edgecolor = 'k')
```

```
ax.set_xticklabels(hemisphereCounts.index,
```

```
                   rotation = 0)
```

```
plt.ylim(0, 6200)
```

```
plt.title('Distribution des personnes dans les Hemispheres')
```

```
plt.xlabel('Hemisphere')
```

```
plt.ylabel('Frequence')
```

```
plt.show()
```

```
# Diagramme en Bâtons : Distribution selon le Sexe
```

```
colors = ['purple', 'darkcyan']
```

```
sexCounts = data['Sex'].value_counts()

ax = sexCounts.plot(kind = 'bar',

                    color = colors,

                    edgecolor = 'k')

ax.set_xticklabels(sexCounts.index,

                  rotation = 0)

plt.ylim(0, 6800)

plt.title('La frequence de chaque sexe')
plt.xlabel('Sexe')
plt.ylabel('Frequence')

plt.show()

# Histogramme : Distribution selon la Consommation d’Alcool
alcoholConsumption = pd.Series(data.AlcoholConsumption)

plt.hist(alcoholConsumption,

        width = 0.2,

        edgecolor = "k",

        color = "red",

        bins = [-1, 0, 1, 2])

plt.xlabel('Consommateurs d\'alcool')
plt.ylabel('Frequence*100')
plt.title('Frequence des consommateurs d\'alcool')
plt.xticks([-1, 0, 1, 2])

plt.show()

# Histogramme : Distribution selon la Diabète
diabetes = pd.Series(data.Diabetes)
plt.style.use('seaborn-v0_8-ticks')

plt.hist(diabetes,

        width = 0.2,

        edgecolor = "k",

        color = "c",

        bins = [-1, 0, 1, 2])

plt.xlabel('Personnes Diabétiques')
```

```
plt.ylabel('Frequence*100')
plt.title('Frequence des personnes diabétiques')
plt.xticks([-1, 0, 1, 2])
plt.ylim(0, 6500)

plt.show()

# Histogramme : Distribution selon l’Avoir d’Infarctus dans l’Histoire de la Famille
familyHistory = pd.Series(data.FamilyHistory)
plt.style.use('seaborn-v0_8-ticks')

plt.hist(familyHistory,
         width = 0.2,
         edgecolor = "k",
         color = "purple",
         bins = [-1, 0, 1, 2])

plt.xlabel('Personnes ayant un problème cardiaque')
plt.ylabel('Frequence*100')
plt.title('Frequence des personnes ayant un problème cardiaque dans l\'histoire de la famille')
plt.xticks([-1, 0, 1, 2])
plt.ylim(0, 5000)

plt.show()

# Histogramme : Distribution selon le Risque d'Attaque Cardiaque
heartAttackRisk = pd.Series(data.HeartAttackRisk)

plt.hist(heartAttackRisk,
         bins = [-1, 0, 1, 2],
         color = 'blue',
         width = 0.2)

plt.xticks([-1, 0, 1, 2])
plt.ylim(0, 6200)

plt.xlabel('Heart Attack Risk')
plt.ylabel('Frequence')
plt.title('Frequence d\'avoir risque d\'un problème cardiaque')

plt.show()
```

Histogramme : Distribution selon l’Utilisation d’un Médicament

```
medUse = pd.Series(data.MedicationUse)

plt.style.use('seaborn-v0_8-ticks')

plt.hist(medUse,
         width = 0.2,
         edgecolor = "k",
         color = "brown",
         bins = [-1, 0, 1, 2])

plt.xlabel('L\'utilisation de medication')
plt.ylabel('Frequence')
plt.title('Frequence de l\'utilisation de medication')
plt.xticks([-1, 0, 1, 2])

plt.show()

# Diagramme en Violon : Distribution selon l’Utilisation d’un Médicament
plt.figure(figsize=(8, 6))
sns.violinplot(x = 'MedicationUse',
               data = data,
               color = 'limegreen',
               edgecolor = 'k',
               bins = [-1, 0, 1, 2])
plt.title('Distribution de l\'utilisation de medication')
plt.xlabel('Utilisation de medication')
plt.xticks([-1, 0, 1, 2])
plt.show()

# Histogramme : Distribution selon l’Obésité
obesity = pd.Series(data.Obesity)
plt.style.use('seaborn-v0_8-ticks')

plt.hist(obesity,
         width = 0.2,
         edgecolor = "k",
         color = "gold",
         bins = [-1, 0, 1, 2])

plt.xlabel('Obesité')
plt.ylabel('Frequence*100')
plt.title('Frequence des personnes obèses')
plt.xticks([-1, 0, 1, 2])
plt.ylim(0, 5000)
```

```
plt.show()
```

```
# Histogramme : Distribution selon l'avoir des Problèmes Cardiaques Antérieurs
```

```
previousHeartProblems = pd.Series(data.PreviousHeartProblems)
```

```
plt.style.use('seaborn-v0_8-ticks')
```

```
plt.hist(previousHeartProblems,
```

```
        width = 0.2,
```

```
        edgecolor = "k",
```

```
        color = "darkorange",
```

```
        bins = [-1, 0, 1, 2])
```

```
plt.xlabel('Presence de problèmes cardiaques au passé')
```

```
plt.ylabel('Frequence')
```

```
plt.title('Frequence d\'avoir un problème cardiaque au passé')
```

```
plt.xticks([-1, 0, 1, 2])
```

```
plt.show()
```

```
# Diagramme en Violon : Distribution selon l'avoir des Problèmes Cardiaques Antérieurs
```

```
plt.figure(figsize=(8, 6))
```

```
sns.violinplot(x = 'PreviousHeartProblems',
```

```
              data = data,
```

```
              color = 'blue',
```

```
              edgecolor = 'k',
```

```
              bins = [-1, 0, 1, 2])
```

```
plt.title('Présence d\'un problème cardiaque au passé')
```

```
plt.xticks([-1, 0, 1, 2])
```

```
plt.show()
```

```
# Histogramme : Distribution selon être Fumeur ou Non-Fumeur
```

```
smoking = pd.Series(data.Smoking)
```

```
plt.style.use('seaborn-v0_8-ticks')
```

```
plt.hist(smoking,
```

```
        width = 0.2,
```

```
        edgecolor = "k",
```

```
        color = "b",
```

```
        bins = [-1, 0, 1, 2])
```

```
plt.xlabel('Fumeurs')
plt.ylabel('Frequence*100')
plt.title('Frequence des personnes fumeurs')
plt.xticks([-1, 0, 1, 2])
plt.ylim(0, 9000)
```

```
plt.show()
```

```
# Nuage des Points : Fumer ~ Age
age = pd.Series(data.Age)
smoking = pd.Series(data.Smoking)
```

```
plt.scatter(age,
            smoking,
            s = 1,
            marker = '.',
            c = 'b')
```

```
plt.title("Frequence de fumeurs en fonction de l'Age")
plt.xlabel("Age")
plt.ylabel("Frequence de fumeurs")
```

```
plt.yticks([-1, 0, 1, 2])
```

```
plt.show()
```

```
# Diagramme en Bâtons : Distribution selon le Continent
continent = ['North America', 'South America', 'Australia', 'Europe', 'Asia', 'Africa']
```

```
valuesNA = len(data[data['Continent'] == 'North America'])
valuesSA = len(data[data['Continent'] == 'South America'])
valuesAUS = len(data[data['Continent'] == 'Australia'])
valuesEU = len(data[data['Continent'] == 'Europe'])
valuesASIA = len(data[data['Continent'] == 'Asia'])
valuesAFR = len(data[data['Continent'] == 'Africa'])
```

```
values = [valuesNA, valuesSA, valuesAUS, valuesEU, valuesASIA, valuesAFR]
```

```
plt.figure(figsize=(8, 6))
plt.bar(continent,
        values,
        color='skyblue',
        edgecolor = 'k')
```



```
plt.xlabel('Continent')
plt.ylabel('Population')
plt.title('Population dans les continents')

plt.show()

# Diagramme en Bâtons : Distribution selon l’Alimentation
colors = ['red', 'green', 'blue']

dietCounts = data['Diet'].value_counts()
ax = dietCounts.plot(kind = 'bar',
                      color = colors)

ax.set_xticklabels(dietCounts.index,
                  rotation = 0)

plt.ylim(0, 3500)

plt.title('Diet Type Frequences')
plt.xlabel('Diet Type')
plt.ylabel('Frequence*100')

plt.show()
```