prediction_cardiaque

November 5, 2024

Projet : Analyse de données scientifiques

0.0.1 PARTIE 1: Importation des données et extraction d'informations (Pandas)

Importer les bibliothèques qui seront nécessaires au projet. Écrire votre code dans la cellule suivante.

[327]:
```
# Votre code ici
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
```

Écrire le code pour charger les données dans le dataframe df_prediction. Écrire votre code dans la cellule suivante.

[328]:
```
# Votre code ici
df_prediction = pd.read_csv('insuffisance_cardiaque.csv')
```

Afficher les noms des colonnes.Écrire le code dans la cellule suivante.

[329]:
```
# Votre code ici
df_prediction.columns
```

[329]:
```
Index(['Age', 'Sex', 'ChestPainType', 'RestingBP', 'Cholesterol', 'FastingBS',
       'RestingECG', 'MaxHR', 'ExerciseAngina', 'Oldpeak', 'ST_Slope',
       'HeartDisease'],
      dtype='object')
```

```
<b>Quel est le nom de la 5e colonne ?</b><br>
```

Écrire votre réponse dans cette cellule : (Double-cliquez dessus pour écrire votre réponse)

Modifier les noms de toutes les colonnes pour les traduire en français, tel qu'indiqué ci-dessous. Écrire le code dans la cellule suivante. **ATTENTION**: Vous devez obligatoirement utiliser les deux listes fournies dans le code: 'noms_actuels' et 'nouveaux_noms' et aussi une boucle.

- **Age** : Âge
- **Sex** : Sexe
- **ChestPainType** : Type de douleur thoracique

- **RestingBP** : Pression artérielle au repos
- **Cholesterol** : Cholestérol
- **FastingBS** : Glycémie à jeun
- **RestingECG** : ECG au repos
- **MaxHR** : Fréquence cardiaque maximale
- **ExerciseAngina** : Angine induite par l'exercice
- **ST_Slope** : Pente du segment ST
- **HeartDisease** : Maladie cardiaque

[330]:
```
# Votre code ici

# Liste des noms actuels des colonnes
noms_actuels = ['Age', 'Sex', 'ChestPainType', 'RestingBP', 'Cholesterol',␣
 ↪'FastingBS', 'RestingECG', 'MaxHR', 'ExerciseAngina', 'ST_Slope',␣
 ↪'HeartDisease']

# Liste des nouveaux noms des colonnes
nouveaux_noms = ['Âge', 'Sexe', 'Type de douleur thoracique', 'Pression␣
 ↪artérielle au repos', 'Cholestérol', 'Glycémie à jeun', 'ECG au repos',␣
 ↪'Fréquence cardiaque maximale', 'Angine induite par l\'exercice', 'Pente du␣
 ↪segment ST', 'Maladie cardiaque']

# Utilisation d'une boucle for pour renommer les colonnes
for i in range(len(noms_actuels)):
    df_prediction = df_prediction.rename(columns={noms_actuels[i]:␣
 ↪nouveaux_noms[i]})
```

Afficher juste les 2 premières lignes du DataFrame, pour confirmer le changement des noms des colonnes. Écrire le code dans la cellule suivante.

[331]:
```
# Votre code ici
df_prediction.head(10)
```

[331]:
```
   Âge Sexe Type de douleur thoracique  Pression artérielle au repos  \
0   40    M                        ATA                           140
1   49    F                        NAP                           160
2   37    M                        ATA                           130
3   48    F                        ASY                           138
4   54    M                        NAP                           150
5   39    M                        NAP                           120
6   45    F                        ATA                           130
7   54    M                        ATA                           110
8   37    M                        ASY                           140
9   48    F                        ATA                           120

   Cholestérol  Glycémie à jeun ECG au repos  Fréquence cardiaque maximale  \
0          289                0       Normal                           172
1          180                0       Normal                           156
```

```
2          283                0           ST                          98
3          214                0       Normal                         108
4          195                0       Normal                         122
5          339                0       Normal                         170
6          237                0       Normal                         170
7          208                0       Normal                         142
8          207                0       Normal                         130
9          284                0       Normal                         120

  Angine induite par l'exercice  Oldpeak Pente du segment ST  \
0                             N      0.0                  Up
1                             N      1.0                Flat
2                             N      0.0                  Up
3                             Y      1.5                Flat
4                             N      0.0                  Up
5                             N      0.0                  Up
6                             N      0.0                  Up
7                             N      0.0                  Up
8                             Y      1.5                Flat
9                             N      0.0                  Up

   Maladie cardiaque
0                  0
1                  1
2                  0
3                  1
4                  0
5                  0
6                  0
7                  0
8                  1
9                  0
```

Quels sont les types des colonnes “Fréquence cardiaque maximale” et “Oldpeak”?Écrire le code permettant d’obtenir les réponses dans la cellule suivante.Écrire vos réponses dans la cellule après celle du code.

[332]:
```
# Votre code ici

df_prediction.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 918 entries, 0 to 917
Data columns (total 12 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Âge                             918 non-null    int64
 1   Sexe                            918 non-null    object
```

```
 2   Type de douleur thoracique     918 non-null    object
 3   Pression artérielle au repos   918 non-null    int64
 4   Cholestérol                    918 non-null    int64
 5   Glycémie à jeun                918 non-null    int64
 6   ECG au repos                   918 non-null    object
 7   Fréquence cardiaque maximale   918 non-null    int64
 8   Angine induite par l'exercice  918 non-null    object
 9   Oldpeak                        918 non-null    float64
 10  Pente du segment ST            918 non-null    object
 11  Maladie cardiaque              918 non-null    int64
dtypes: float64(1), int64(6), object(5)
memory usage: 86.2+ KB
```

Écrire votre réponse dans cette cellule : (Double-cliquez dessus pour écrire votre réponse)
Type de 'Fréquence cardiaque maximale':
Type de 'Oldpeak':

Combien il y a t-il de données manquantes dans le dataFrame ?Écrire le code permettant d'obtenir les réponses dans la cellule suivante.Écrire vos réponses dans la cellule après celle du code.

[333]:
```
# Votre code ici

#df_prediction.isnull()
donnees_manquantes = df_prediction.isna()

print(donnees_manquantes)
```

```
       Âge   Sexe  Type de douleur thoracique  Pression artérielle au repos  \
0    False  False                       False                         False
1    False  False                       False                         False
2    False  False                       False                         False
3    False  False                       False                         False
4    False  False                       False                         False
..     …    …                           …                             …
913  False  False                       False                         False
914  False  False                       False                         False
915  False  False                       False                         False
916  False  False                       False                         False
917  False  False                       False                         False

     Cholestérol  Glycémie à jeun  ECG au repos  Fréquence cardiaque maximale  \
0          False            False         False                         False
1          False            False         False                         False
2          False            False         False                         False
3          False            False         False                         False
4          False            False         False                         False
..           …                …           …                             …
913        False            False         False                         False
914        False            False         False                         False
```

```
915        False            False         False                          False
916        False            False         False                          False
917        False            False         False                          False

     Angine induite par l'exercice  Oldpeak  Pente du segment ST  \
0                             False    False                False
1                             False    False                False
2                             False    False                False
3                             False    False                False
4                             False    False                False
..                              …        …                  …
913                           False    False                False
914                           False    False                False
915                           False    False                False
916                           False    False                False
917                           False    False                False

     Maladie cardiaque
0                False
1                False
2                False
3                False
4                False
..                 …
913              False
914              False
915              False
916              False
917              False

[918 rows x 12 columns]
```

Écrire votre réponse dans cette cellule : (Double-cliquez dessus pour écrire votre réponse) Nombre de données manquantes:

- Définissez une fonction qui prend un dataframe en paramètre et qui retourne le dataframe avec aucune donnée manquante.
- Utilisez une boucle pour effectuer votre calcul dans la fonction.
- Utilisez cette fonction avec le dataframe contenant seulement les personnes atteintes de maladies cardiaques et affichez le résultat avec un `print`.

[334]:
```
# Votre code ici
# Définition de la fonction
def nettoyage_df(donnees_sales):
    #


# Appel de la fonction
```

```
nettoyage_df()    # Compléter

# Confirmation qu'il n'y a plus de données manquantes
df_prediction.isnull()
```

```
  Cell In[334], line 8
    nettoyage_df()    # Compléter
    ^
IndentationError: expected an indented block after function definition on line 
```

Quelle est la moyenne d'âge des patients ?Écrire le code permettant d'obtenir les réponses dans la cellule suivante.Écrire vos réponses dans la cellule après celle du code.

[ ]:
```
# Votre code ici

df_prediction.describe()
```

[ ]:

	Âge	Pression artérielle au repos	Cholestérol	Glycémie à jeun	\
count	918.000000	918.000000	918.000000	918.000000	
mean	53.510893	132.396514	198.799564	0.233115	
std	9.432617	18.514154	109.384145	0.423046	
min	28.000000	0.000000	0.000000	0.000000	
25%	47.000000	120.000000	173.250000	0.000000	
50%	54.000000	130.000000	223.000000	0.000000	
75%	60.000000	140.000000	267.000000	0.000000	
max	77.000000	200.000000	603.000000	1.000000	

	Fréquence cardiaque maximale	Oldpeak	Maladie cardiaque
count	918.000000	918.000000	918.000000
mean	136.809368	0.887364	0.553377
std	25.460334	1.066570	0.497414
min	60.000000	-2.600000	0.000000
25%	120.000000	0.000000	0.000000
50%	138.000000	0.600000	1.000000
75%	156.000000	1.500000	1.000000
max	202.000000	6.200000	1.000000

Écrire votre réponse dans cette cellule : (Double-cliquez dessus pour écrire votre réponse) Âge moyen des patients:

Sélection de la population **atteinte** de maladies cardiaques

Filtrez les personnes atteintes de maladie cardiaque. Nous voulons un dataframe avec seulement les patients (Hommes et Femmes) atteints de maladies cardiaques. Indices: - Filtrer la colonne 'Maladie cardiaque' - Un patient est atteint de maladies cardiaques si la valeur de la colonne 'Maladie cardiaque' est égale à 1

Combien il y a t-il de personnes atteintes de maladies cardiaques ?Écrire le code permettant

d'obtenir les réponses dans la cellule suivante.Écrire la réponse avec la fonction 'print' écrite dans la cellule code.

[ ]:
```
# Complétez le code ci-dessous
df_cardiaques = df_prediction[(df_prediction['Maladie cardiaque'] == 1)]
nombre_de_lignes = len(df_cardiaques)
print(f"Il y a {nombre_de_lignes} personnes atteintes de maladies cardiaques")
```

```
Il y a 508 personnes atteintes de maladies cardiaques
```

Sélection de la population féminine **atteinte** de maladies cardiaques

Filtrez les personnes féminines atteintes de maladie cardiaque. Nous voulons un dataframe avec seulement les patientes de sexe féminin et cardiaques. Indices: - Filtrer les colonnes 'Sexe' et 'Maladie cardiaque' - Un patient est atteint de maladies cardiaques si la valeur de la colonne 'Maladie cardiaque' est égale à 1

Combien il y a t-il de femmes atteintes de maladies cardiaques ?Écrire le code permettant d'obtenir les réponses dans la cellule suivante.Écrire la réponse avec la fonction 'print' écrite dans la cellule code.

[ ]:
```
# Complétez le code ci-dessous
df_F_cardiaques = df_prediction[(df_prediction['Sexe'] == 'F') &␣
 ↪(df_prediction['Maladie cardiaque'] == 1)]
nombre_de_lignes = len(df_F_cardiaques)
print(f"Il y a {nombre_de_lignes} femmes atteintes de maladies cardiaques")
```

```
Il y a 50 femmes atteintes de maladies cardiaques
```

Sélection de la population féminine **non atteinte** de maladies cardiaques

Filtrez les personnes féminines non atteintes de maladie cardiaque. Nous voulons un dataframe avec seulement les patientes de sexe féminin et non cardiaques. Indices: - Filtrer les colonnes 'Sexe' et 'Maladie cardiaque' - Une patiente n'est pas atteinte de maladies cardiaques si la valeur de la colonne 'Maladie cardiaque' est égale à 0

Combien il y a t-il de femmes non atteintes de maladies cardiaques ?Écrire le code permettant d'obtenir les réponses dans la cellule suivante.Écrire la réponse avec la fonction 'print' écrite dans la cellule code.

[ ]:
```
# Complétez le code ci-dessous
df_F_non_cardiaques = df_prediction[(df_prediction['Sexe'] == 'F') &␣
 ↪(df_prediction['Maladie cardiaque'] == 0)]
nombre_de_lignes = len(df_F_non_cardiaques)
print(f"Il y a {nombre_de_lignes} femmes non atteintes de maladies cardiaques")
```

```
Il y a 143 femmes non atteintes de maladies cardiaques
```

Que pouvez-vous conclure par rapport au nombre de femmes atteintes versus celles qui ne sont pas atteintes de maladies cardiaques ?Écrire la réponse dans la cellule suivante.

Écrire votre réponse dans cette cellule : (Double-cliquez dessus pour écrire votre réponse) Conclusion (F atteintes vs non atteintes):

Les patientes atteintes de maladies cardiaques sont moins nombreuses que celles sans maladies cardiaques.

Quelle est la moyenne du cholestérol pour les personnes (Hommes et Femmes) atteintes de maladies cardiaques ?Écrire le code permettant d'obtenir la réponse dans la cellule suivante. Pour ce faire: - Définissez une fonction qui prend un dataframe en paramètre et qui retourne la moyenne du cholestérol pour ce dataframe. - Utilisez une boucle pour effectuer votre calcul dans la fonction. - Utilisez cette fonction avec le dataframe contenant seulement les personnes atteintes de maladies cardiaques et affichez le résultat avec un `print`.

[ ]:
```
# Complétez le code ci-dessous
def calcul_moyenne_cholesterol(donnees):
    somme = 0
    for cas in donnees["Cholestérol"]:
        somme += cas

    return somme / len(donnees)

# Appel de la fonction
moyenne_cholesterol = calcul_moyenne_cholesterol(df_cardiaques)
print(f"La moyenne du cholestérol des patients atteints de maladies cardiaques␣
 ↪est de {round(moyenne_cholesterol,2)} mg/dl")
```

```
La moyenne du cholestérol des patients atteints de maladies cardiaques est de
175.94 mg/dl
```

Vérifiez votre résultat avec la fonction 'describe()'. Écrire votre code dans la cellule suivante.

[ ]:
```
# Votre code ici

df_atteints.describe()
```

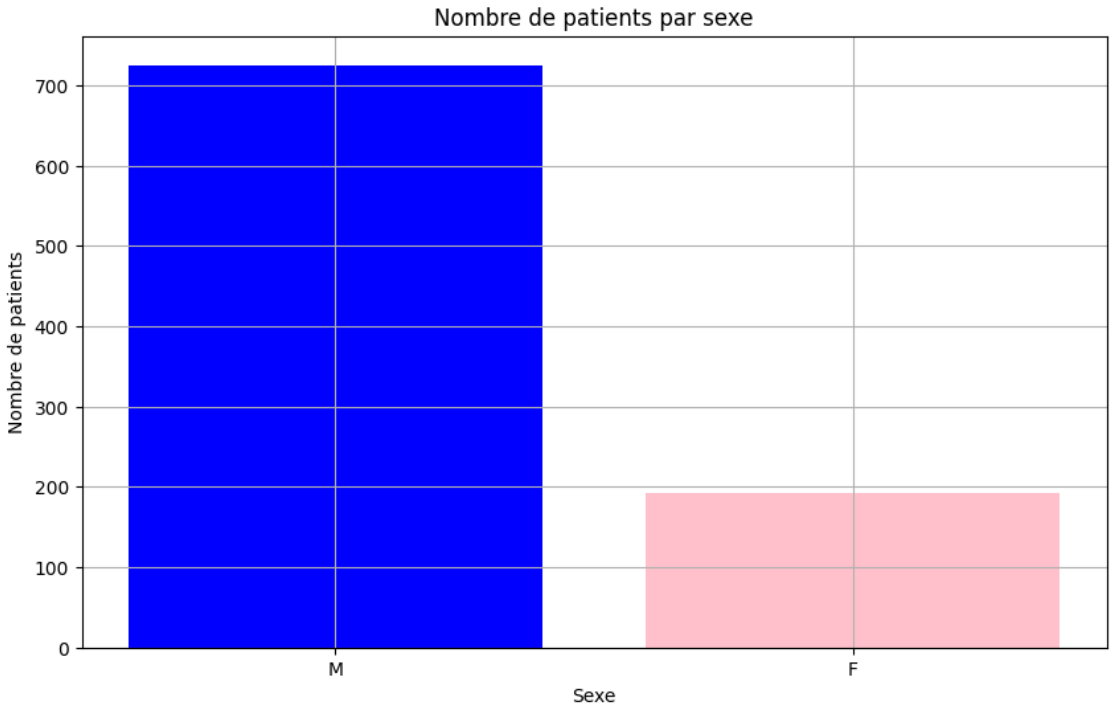
[ ]:
```
              Âge  Pression artérielle au repos  Cholestérol  Glycémie à jeun  \
count  508.000000                    508.000000   508.000000       508.000000
mean    55.899606                    134.185039   175.940945         0.334646
std      8.727056                     19.828685   126.391398         0.472332
min     31.000000                      0.000000     0.000000         0.000000
25%     51.000000                    120.000000     0.000000         0.000000
50%     57.000000                    132.000000   217.000000         0.000000
75%     62.000000                    145.000000   267.000000         1.000000
max     77.000000                    200.000000   603.000000         1.000000

       Fréquence cardiaque maximale     Oldpeak  Maladie cardiaque
count                    508.000000  508.000000              508.0
mean                     127.655512    1.274213                1.0
std                       23.386923    1.151872                0.0
min                       60.000000   -2.600000                1.0
25%                      112.000000    0.000000                1.0
50%                      126.000000    1.200000                1.0
```

75%	144.250000	2.000000	1.0
max	195.000000	6.200000	1.0

0.0.2 Partie 2. Visualiser graphiquement les données des patients (Matplotlib)

Entre les hommes et les femmes quel genre est plus nombreux parmis les patients ?Écrire le code permettant de créer un un diagramme à barres pour montrer le nombre de patients masculins et féminins., dans la cellule suivante.

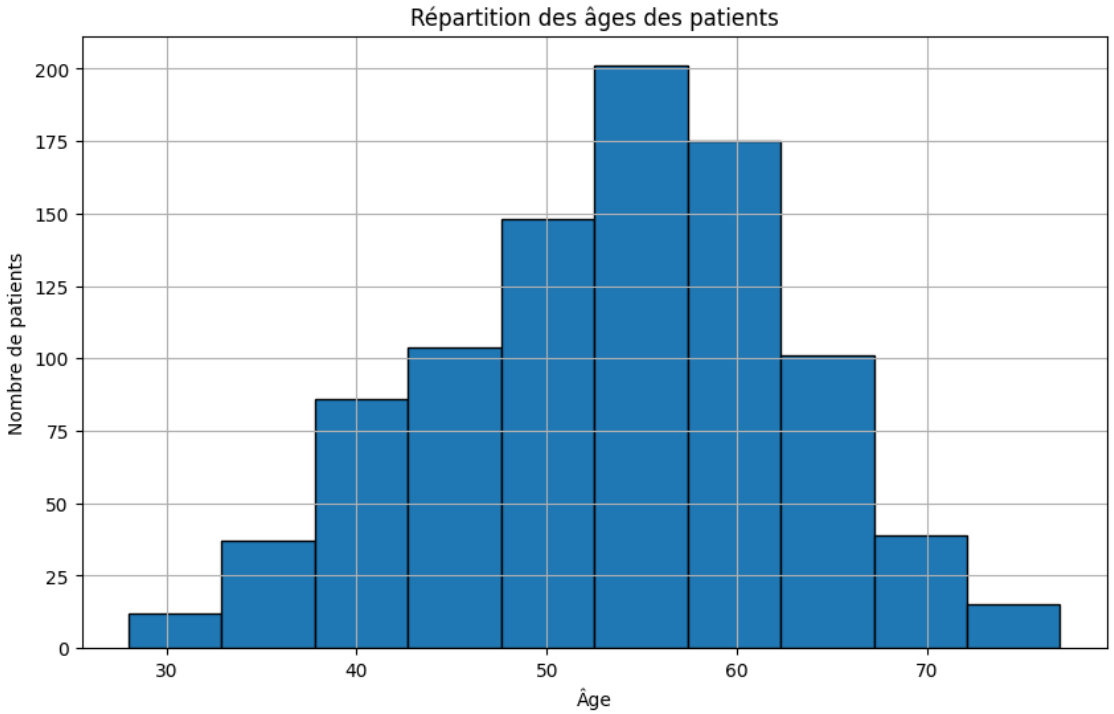
[ ]:
```
# Votre code ici
# Graphique 1: Diagramme à barres du nombre de patients par sexe

# Compter le nombre de patients par sexe
sex_counts = df_prediction['Sexe'].value_counts()

# Créer le diagramme à barres
plt.figure(figsize=(10, 6))
plt.bar(sex_counts.index, sex_counts.values, color=['blue', 'pink'])
plt.xlabel('Sexe')
plt.ylabel('Nombre de patients')
plt.title('Nombre de patients par sexe')
plt.grid(True)
plt.show()
```



Quel est la tranche d'âges de la majorité des patients ?Écrire le code permettant de créer un

histogramme pour visualiser la répartition des âges des patients, dans la cellule suivante.

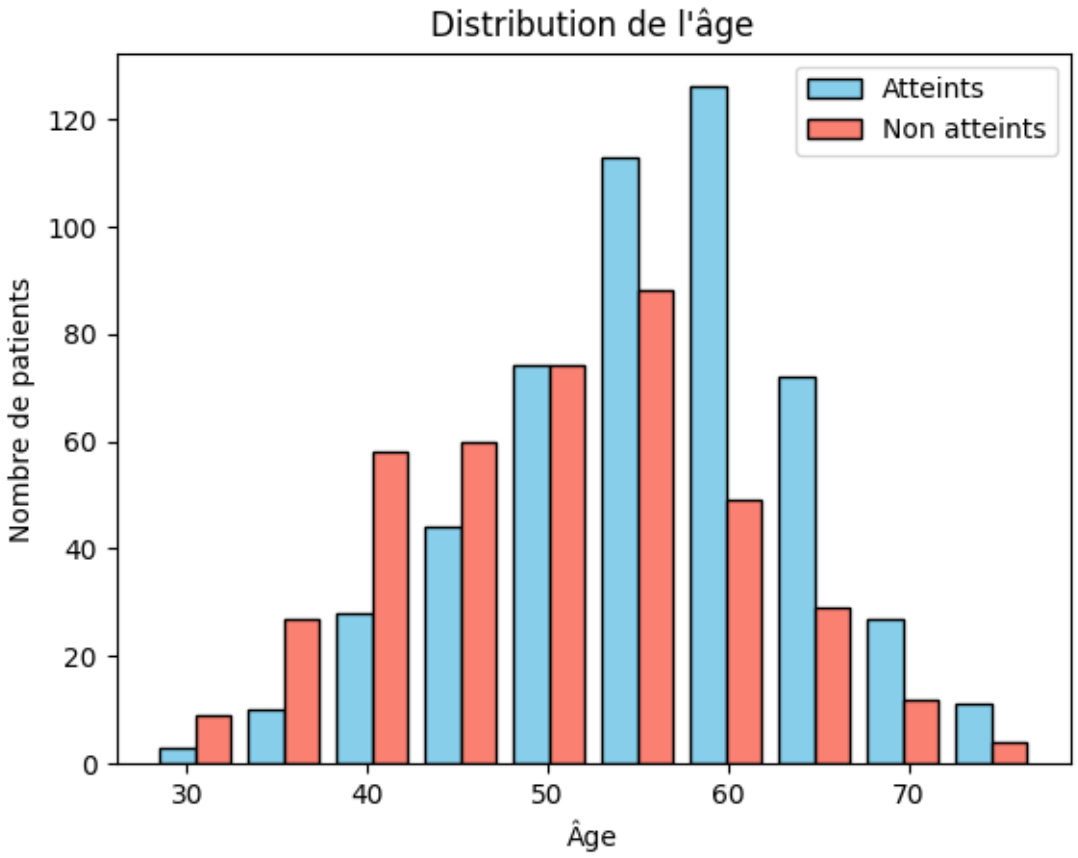
[ ]:
```
# Votre code ici

# Graphique 2: Histogramme de la répartition des âges

plt.figure(figsize=(10, 6))
plt.hist(df_prediction['Âge'], bins=10, edgecolor='black')
plt.xlabel('Âge')
plt.ylabel('Nombre de patients')
plt.title('Répartition des âges des patients')
plt.grid(True)
plt.show()
```



Quelle est la distribution des âges, entre les patients atteints et ceux pas atteints de maladies cardiaque ?Écrire le code permettant de créer un un histogramme pour montrer la distribution des patients atteints vs non atteints., dans la cellule suivante.

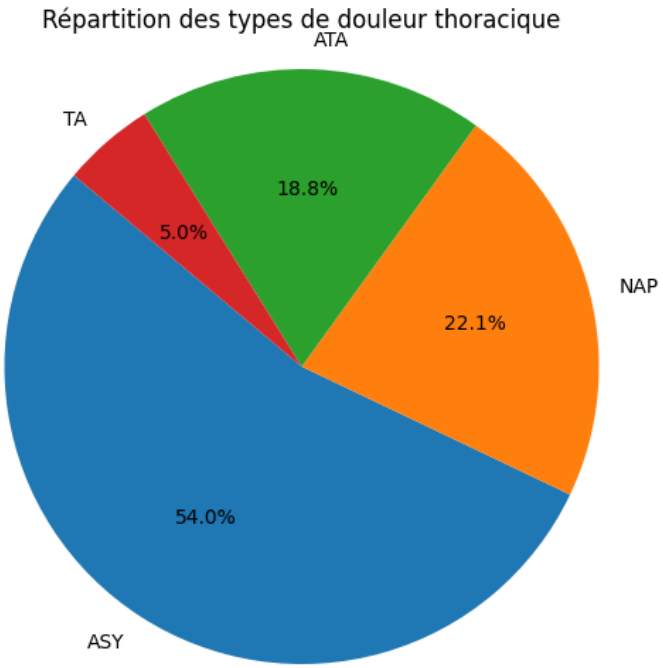
[ ]:
```
# Graphique 3: Histogramme de la distribution des âges selon qu'ils sont␣
↪atteints ou non

# Créer une liste contenant l'age des patients atteints
liste_age_atteints = df_atteints['Âge'].values.tolist()
```

```
# Créer une liste contenant l'age des patients non atteints
liste_age_non_atteints = df_non_atteints['Âge'].values.tolist()

# Créer l'histogramme
plt.hist([liste_age_atteints, liste_age_non_atteints], bins=10,␣
 ↪color=['skyblue', 'salmon'], edgecolor='black', label=['Atteints', 'Non␣
 ↪atteints'])

# Ajouter des titres et des labels
plt.title('Distribution de l\'âge')
plt.xlabel('Âge')
plt.ylabel('Nombre de patients')
plt.legend()

# Afficher le graphique
plt.show()
```



Quel est le type de douleur thoracique le plus fréquent parmis les patients ?Écrire le code permettant de créer un un diagramme en secteurs pour visualiser la répartition des types de douleur thoracique parmi les patients., dans la cellule suivante.

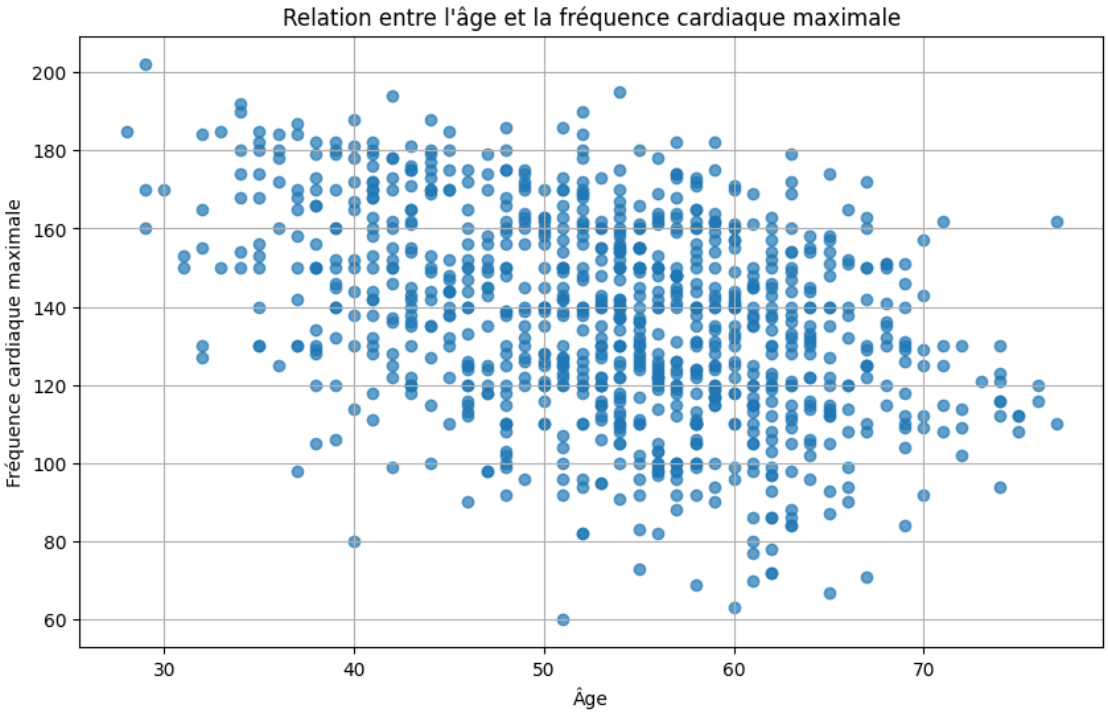
[ ]:
```
# Votre code ici

# Graphique 4: Diagramme en secteurs du type de douleur thoracique
# Compter le nombre de patients par type de douleur thoracique
chest_pain_counts = df_prediction['Type de douleur thoracique'].value_counts()

# Créer le diagramme en secteurs
plt.figure(figsize=(10, 6))
plt.pie(chest_pain_counts.values, labels=chest_pain_counts.index, autopct='%1.
↪1f%%', startangle=140)
plt.title('Répartition des types de douleur thoracique')
plt.axis('equal')  # Assurer que le diagramme est circulaire
plt.show()
```



Quel est la tranche d'âge des paptients qui ont la fréquence cardiaque maximale parmis les patients ?Écrire le code permettant de créer un un Nuage de points de la fréquence cardiaque maximale en fonction de l'âge., dans la cellule suivante.

[ ]:
```
# Votre code ici

# Graphique 5: # Créer le nuage de points

plt.figure(figsize=(10, 6))
```

```
plt.scatter(df_prediction['Âge'], df_prediction['Fréquence cardiaque␣
 ↪maximale'], alpha=0.7)
plt.xlabel('Âge')
plt.ylabel('Fréquence cardiaque maximale')
plt.title('Relation entre l\'âge et la fréquence cardiaque maximale')
plt.grid(True)
plt.show()
```



Quel est la tranche d'âge des patients qui ont la un taux de cholestérol élevé parmis les patients ?Écrire le code permettant de créer un nuage de points pour visualiser la relation entre l'âge et le cholestérol chez les patients atteints., dans la cellule suivante.

[ ]:
```
# Créer une liste contenant le taux de cholestérol des patients atteints
liste_cholesterol_atteints = df_atteints['Cholestérol'].values.tolist()

# Créer le nuage de points
plt.scatter(liste_age_atteints, liste_cholesterol_atteints, color='red')

# Ajouter des titres et des labels
plt.title('Âge vs Cholestérol')
plt.xlabel('Âge')
plt.ylabel('Cholestérol')

# Afficher le graphique
```

```
plt.show()
```

0.0.3 Partie 3. Analyse statistiques (NumPy)

Calcul de la moyenne et de l'écart-type de l'âge des patientsÉcrire le code NumPy dans la cellule suivante.

[ ]:
```
# Votre code ici

# Extraire les âges des patients
ages = df_prediction['Âge'].values

# Calculer la moyenne et l'écart-type
moyenne_age = np.mean(ages)
ecart_type_age = np.std(ages)

print(f"Moyenne de l'âge des patients : {round(moyenne_age,2)}")
print(f"Écart-type de l'âge des patients : {round(ecart_type_age,2)}")
```

```
Moyenne de l'âge des patients : 53.51
Écart-type de l'âge des patients : 9.43
```

Calcul de la pression artérielle moyenne au repos par sexeÉcrire le code NumPy dans la cellule suivante.

[ ]:
```
# Votre code ici

# Extraire les pressions artérielles au repos par sexe
pression_homme = df_prediction[df_prediction['Sexe'] == 'M']['Pression␣
 ↪artérielle au repos'].values
pression_femme = df_prediction[df_prediction['Sexe'] == 'F']['Pression␣
 ↪artérielle au repos'].values

# Calculer la moyenne pour chaque sexe
moyenne_pression_homme = np.mean(pression_homme)
moyenne_pression_femme = np.mean(pression_femme)

print(f"Pression artérielle moyenne au repos (Hommes) :␣
 ↪{round(moyenne_pression_homme,2)}")
print(f"Pression artérielle moyenne au repos (Femmes) :␣
 ↪{round(moyenne_pression_femme,2)}")
```

```
Pression artérielle moyenne au repos (Hommes) : 132.45
Pression artérielle moyenne au repos (Femmes) : 132.21
```

Calcul de la médiane du cholestérolÉcrire le code NumPy dans la cellule suivante.

[ ]:
```
# Votre code ici

# Extraire les valeurs de cholestérol
cholesterol = df_prediction['Cholestérol'].values

# Calculer la médiane
medianne_cholesterol = np.median(cholesterol)

print(f"Médiane du cholestérol des patients : {medianne_cholesterol}")
```

```
Médiane du cholestérol des patients : 223.0
```

Calcul de la fréquence cardiaque maximale moyenne pour les patients avec et sans maladie cardiaque.Écrire le code NumPy dans la cellule suivante.

[ ]:
```
# Votre code ici

# Extraire les fréquences cardiaques maximales pour chaque groupe
hr_max_avec_maladie = df_prediction[df_prediction['Maladie cardiaque'] ==␣
 ↪1]['Fréquence cardiaque maximale'].values
```

```
hr_max_sans_maladie = df_prediction[df_prediction['Maladie cardiaque'] ==␣
 ↪0]['Fréquence cardiaque maximale'].values

# Calculer la moyenne pour chaque groupe
moyenne_hr_max_avec_maladie = np.mean(hr_max_avec_maladie)
moyenne_hr_max_sans_maladie = np.mean(hr_max_sans_maladie)

print(f"Fréquence cardiaque maximale moyenne (avec maladie cardiaque) :␣
 ↪{round(moyenne_hr_max_avec_maladie,2)}")
print(f"Fréquence cardiaque maximale moyenne (sans maladie cardiaque) :␣
 ↪{round(moyenne_hr_max_sans_maladie,2)}")
```

```
Fréquence cardiaque maximale moyenne (avec maladie cardiaque) : 127.66
Fréquence cardiaque maximale moyenne (sans maladie cardiaque) : 148.15
```

Calcul de la proportion de patients ayant une angine induite par l'exercice.Écrire le code NumPy dans la cellule suivante.

[ ]:
```
# Votre code ici

# Extraire les valeurs d'angine induite par l'exercice
angine_par_exercice = df_prediction['Angine induite par l\'exercice'].values

# Calculer la proportion
proportion_angine = np.sum(angine_par_exercice == 'Y') /␣
 ↪len(angine_par_exercice)

print(f"Proportion de patients ayant une angine induite par l'exercice :␣
 ↪{round(proportion_angine,2)}")
```

```
Proportion de patients ayant une angine induite par l'exercice : 0.4
```