

Mini Projet de Visualisation Python

(Theme Cyberpunk Display)

BAKENGGA BAMBI William

2022-09-29

DATA VISUALISATION

L'intérêt ici est non seulement le test du package `cyberpunk` mais aussi les avantages de la combinaison entre `RMarkdown` et le langage `python`, qui est une option intéressante tant `RMarkdown` offre des capacités de rendus assez intéressantes (to say the least...). De plus coupler les rendus `RMarkdown` avec des graphiques réalisés avec `matplotlib` ou `seaborn` peut être une méthode qui finira par se répandre qui sait ... Le jeu de donnée utilisé se portera sûr les violences à *l'arme à feu de la police* dans les différents états aux **Etats-Unis**.

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import mplcyberpunk
import matplotlib.style as style
```

Ici nous avons importer les packages *python* dont nous allons avoir besoin. `Pandas` va nous permettre d'apporter quelques modifications à notre jeu de données afin de le modéliser tel que nous en aurons besoin. L'essentiel des graphiques sera fourni grâce aux extensions `seaborn` et `matplotlib`. Nous modifieront les thèmes de fond grâce `mplcyberpunk`.

```
data1 = pd.read_csv('C:\\Users\\HP\\Documents\\Cours\\Tableaux_donnees\\fatal-police-shootings-data.csv')
data1.head()
```

```
##      id      name      date ... threat_level      flee      body_camera
## 0     3      Tim Elliot 2015-01-02 ...      attack  Not fleeing      False
## 1     4  Lewis Lee Lembke 2015-01-02 ...      attack  Not fleeing      False
## 2     5  John Paul Quintero 2015-01-03 ...      other  Not fleeing      False
## 3     8   Matthew Hoffman 2015-01-04 ...      attack  Not fleeing      False
## 4     9  Michael Rodriguez 2015-01-04 ...      attack  Not fleeing      False
##
## [5 rows x 14 columns]
```

```
data1.info()
```

```
## <class 'pandas.core.frame.DataFrame'>
## RangeIndex: 5416 entries, 0 to 5415
## Data columns (total 14 columns):
##  #   Column      Non-Null Count  Dtype
## ---  ---
##  0   id          5416 non-null   int64
##  1   name        5416 non-null   object
##  2   date        5416 non-null   object
```

```
## 3   manner_of_death      5416 non-null   object
## 4   armed                5189 non-null   object
## 5   age                  5181 non-null   float64
## 6   gender               5414 non-null   object
## 7   race                 4895 non-null   object
## 8   city                 5416 non-null   object
## 9   state                5416 non-null   object
## 10  signs_of_mental_illness 5416 non-null   bool
## 11  threat_level         5416 non-null   object
## 12  flee                 5167 non-null   object
## 13  body_camera          5416 non-null   bool
## dtypes: bool(2), float64(1), int64(1), object(10)
## memory usage: 518.5+ KB
```

```
list(data1.columns)
```

```
## ['id', 'name', 'date', 'manner_of_death', 'armed', 'age', 'gender', 'race', 'city', 'state', 'signs_
```

Nous allons ici faire un descriptif assez rapide des variables du jeu de donnée, alors :

id : Cela va représenter l'identifiant de la victime
 name : Le nom de la victime
 manner_of_death : La manière dont la victime a été tuée
 armed : Si la victime était armée
 age : l'âge de la victime
 gender : Le sexe de la victime
 race : La race ethnique de la victime
 city : La ville où le meurtre s'est déroulé
 state : L'état où le meurtre s'est déroulé
 signs_of_mental_illness : des signes de troubles mentaux chez la victime
 threat_level : Le niveau de danger que représentait la victime
 flee : si la victime avait un moyen de transport et lequel
 body_camera : si l'officier avait une caméra dans son uniforme

Par Etats

```
plt.rcParams.update({'font.sans-serif': 'Helvetica'})

plt.style.use("cyberpunk")

sns.set_style({'font.family' : 'serif', 'font_serif' : ['Times New Roman']})

les_6_etats = data1['state'].value_counts().nlargest(6)

plt.figure(figsize = (18,9))
plt.title("Les 6 états subissant le plus de violences policières"
        , size = 20
        , color = 'grey'
        , fontweight="bold"
        )
plt.xlabel("Les noms des différents états"
        , size = 16
        , color = 'grey'
        , fontweight="bold")
plt.xticks(rotation = 90
        , color = 'white'
        , fontsize = 14
        , fontweight = 'bold')
```

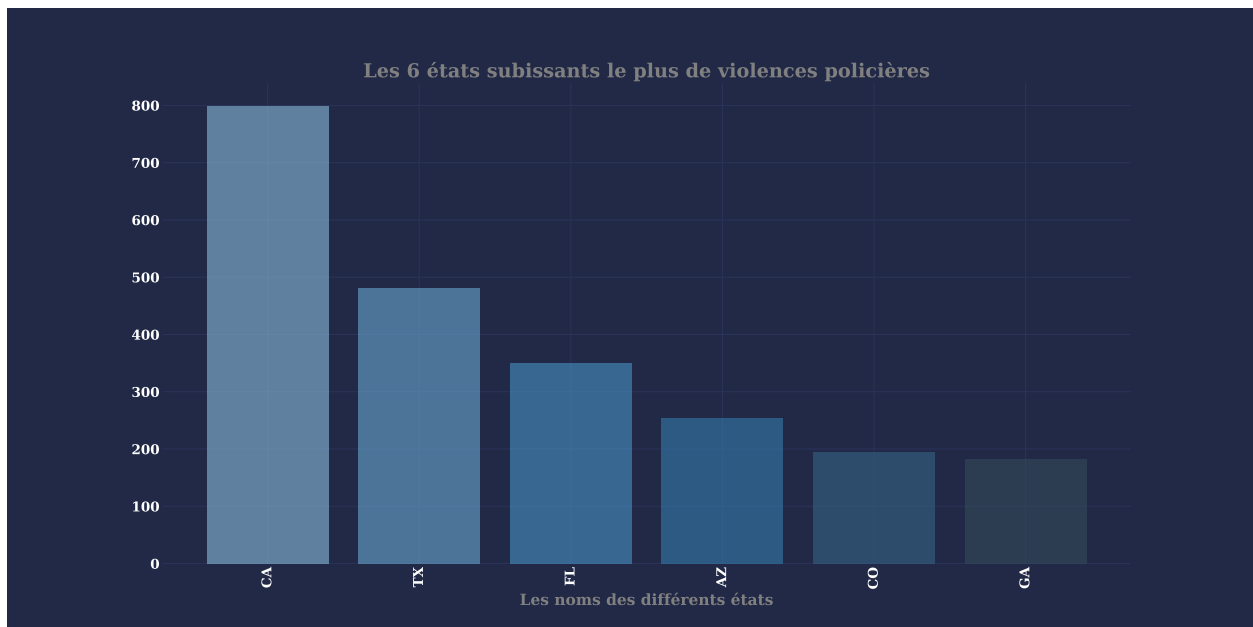
```
## (array([0. , 0.2, 0.4, 0.6, 0.8, 1. ]), [Text(0.0, 0, '0.0'), Text(0.2, 0, '0.2'), Text(0.4, 0, '0.4
```

```
plt.yticks(color = 'white'
           , fontsize = 14
           , fontweight = 'bold')

## (array([0. , 0.2, 0.4, 0.6, 0.8, 1. ]), [Text(0, 0.0, '0.0'), Text(0, 0.2, '0.2'), Text(0, 0.4, '0.4')])
plt.bar(x = les_6_etats.index
       , height = les_6_etats
       , color=sns.color_palette("Blues_d")
       , alpha = 0.6
       )

## <BarContainer object of 6 artists>

plt.show()
```



L'idée ici a été de prendre les états les **plus concernée** par les violences et d'en faire un tableau de donnée. On peut voir que la médaille d'or de la catégorie revient à la **californie** qui est quand même bien au dessus en terme de valeur que son second qui est le **Texas**. Le **Texas** lui a des valeurs et donc une quantité de rime plus proche des *états* qui le suivent. Ici le barplot du package `matplotlib` montre une hiérarchie de manière assez efficace des *6 états les plus touchés*. On voit que la **Floride**, L'**Arizona**, le **Colorado**, qui paraît plus étonnant, et la **Géorgie** sont les états les plus **touchés**. Remarquons l'*absence étonnante* pour ma part de l'état de *New York* qui a longtemps été très connu pour être très agité. Concernant l'aspect du graphique, on voit l'effet de l'arrière plan modifié grâce au package `mplcyberpunk`, pour pouvoir accorder les `barplot` avec ce *package*, j'ai décidé d'utiliser la palette *Blues_d* de l'extension `seaborn`. Et enfin, les différents titres sont colorés en *grey* qui est une couleur assez basique.

```
les_8_villes_Plus = data1['city'].value_counts().nlargest(8)

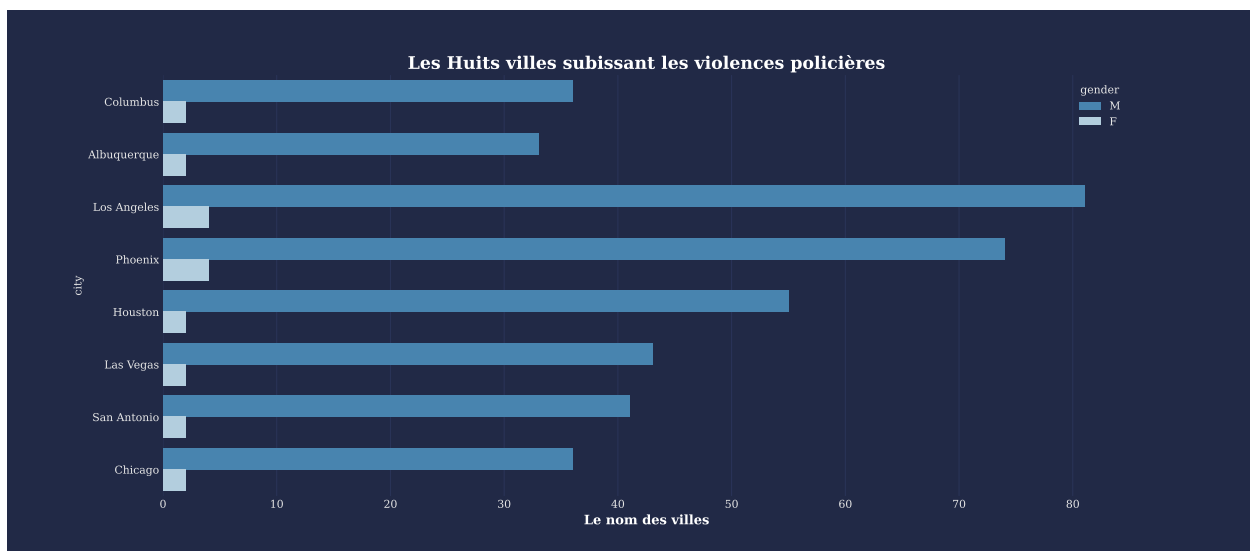
data2 = data1[data1['city'].isin(les_8_villes_Plus.index)]

style.use("cyberpunk")
sns.set_style({'font.family' : 'serif', 'font_serif' : ['Times New Roman']})
plt.figure(figsize = (16,7))
```

```

p = sns.countplot(y = 'city'
                  , data=data2
                  , hue = "gender"
                  , palette="Blues_r"
                  , orient = "h"
                  )
plt.title("Les Huits villes subissant les violences policières"
         , color = 'white'
         , size = 16
         , fontweight="bold")
plt.xlabel("Le nom des villes"
         , size = 12
         , color = 'white'
         , fontweight='bold')
plt.show(p)

```



Ici, même si la tendance n'est *pas exactement identique* à ce qu'on a pu observer précédemment, on peut voir qu'il y a quand même des *éléments qui concorde* avec le *premier graphique*. Tout d'abord, nous pouvons voir que **Los Angeles** qui est une ville de l'état de la **Californie** est en tête, suivi de **Phoenix** ville en **Arizona**, **Houston** et **San Antonio** (**Texas**), ce qui rejoint la tendance *précédente*. Par contre **Albuquerque** qui est situé au **nouveau Mexique**, **Las Vegas** dans le **Nevada** et **Chicago** dans l'**illinois** ne sont pas situés dans des *états que nous avons précédemment étudié*. Cependant nous en pouvons pas dire que c'est une surprise tant **Chicago** est connu pour être un états générant pas mal d'effusion de sang. On peut donc quand même être satisfait car les concordances ici valide le graphique précédent malgré quelques divergeances mais cela s'explique aussi car si on venait à augmenter l'échantillon de ville, on pourrait largement voir apparaître plusieurs villes des états du premier graphique. Cette fois, pour l'aspect, nous avons utilisé le "countplot" de **seaborn** avec encore une fois la palette **Blues_r** et nous nous sommes intéressé à un aspect pas encore étudié, le sexe. On observe qu'il y a beaucoup moins de femmes touchées par les meurtres.

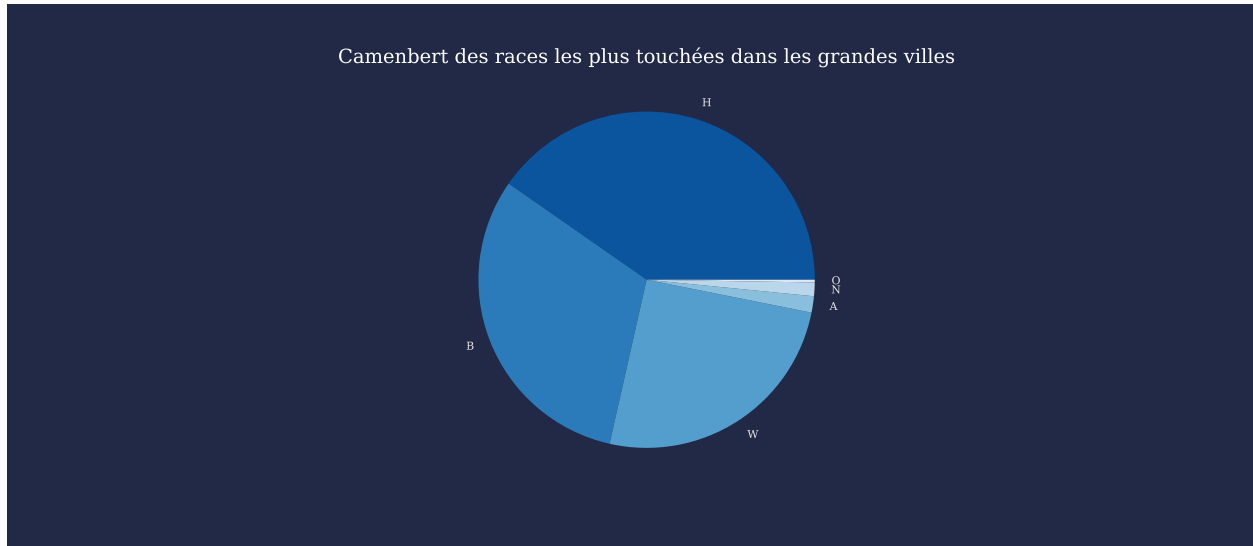
```

style.use("cyberpunk")
sns.set_style({'font.family' : 'serif', 'font_serif' : ['Times New Roman']})
plt.figure(figsize = (16,7))
plt.title("Camenbert des races les plus touchées dans les grandes villes"
         , fontsize = 18
         , color = 'white')
plt.pie(data2['race'].value_counts())

```

```
, colors = sns.color_palette("Blues_r")
, labels = data2['race'].value_counts().index)
```

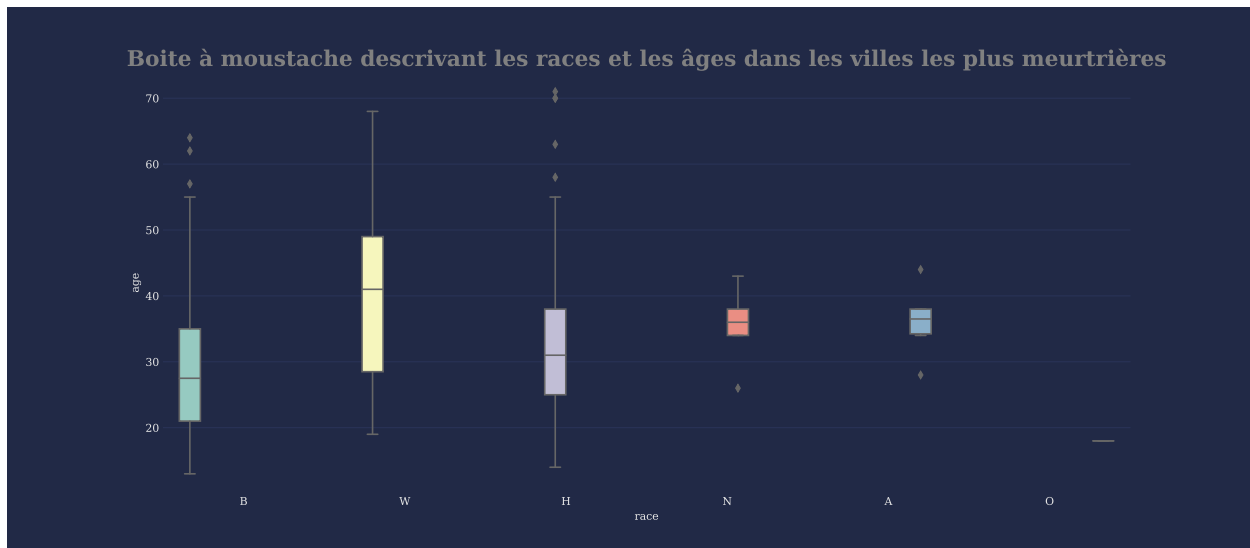
```
## ([<matplotlib.patches.Wedge object at 0x000001FF42AE0940>, <matplotlib.patches.Wedge object at 0x000001FF42AE0940>])
plt.show()
```



Ici, même si ce n'est pas l'une de mes visualisation préférées, on utilise le pie plot car dans ce cas il est possible d'à peu près visualiser les proportions. Ici le but était de s'intéresser aux *rac*es les plus touchées par les meurtres à l'arme à feu des policiers. On peut observer que les plus touchés sont issus de la communauté **Hipanique** suivie de la communauté **noire** puis de la communauté **blanche**. On peut aussi se demander si la proportion d'habitants de chaque communauté dans ces états n'y joue pas un rôle et si il faut absolument conclure que c'est absolument représentatif dans le pays dans sa totalité, tant ceci n'est qu'une représentation. Par contre, on peut largement se faire des idées grâce à cette proportion et se poser la question d'une récurrence. Dans l'aspect visuel, ce graphique est réalisé avec `matplotlib`, sauf la palette qui, elle, vient de `seaborn`.

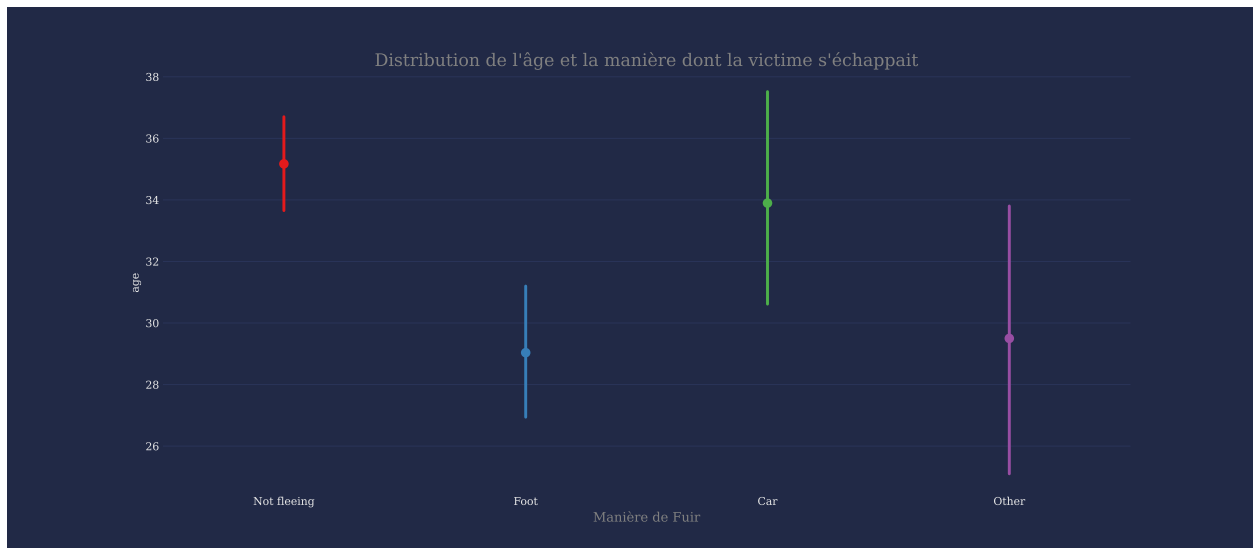
```
style.use("cyberpunk")
sns.set_style({'font.family' : 'serif', 'font_serif' : ['Times New Roman']})
plt.figure(figsize = (16,7))
plt.title("Boite à moustache décrivant les races et les âges dans les villes les plus meurtrières"
, size = 20
, color='grey'
, fontweight="bold")
sns.boxplot(x = "race"
, y = "age"
, palette="Set3"
, hue = "race"
, data = data2)
plt.legend([], [])
, frameon = False
)

plt.show()
```



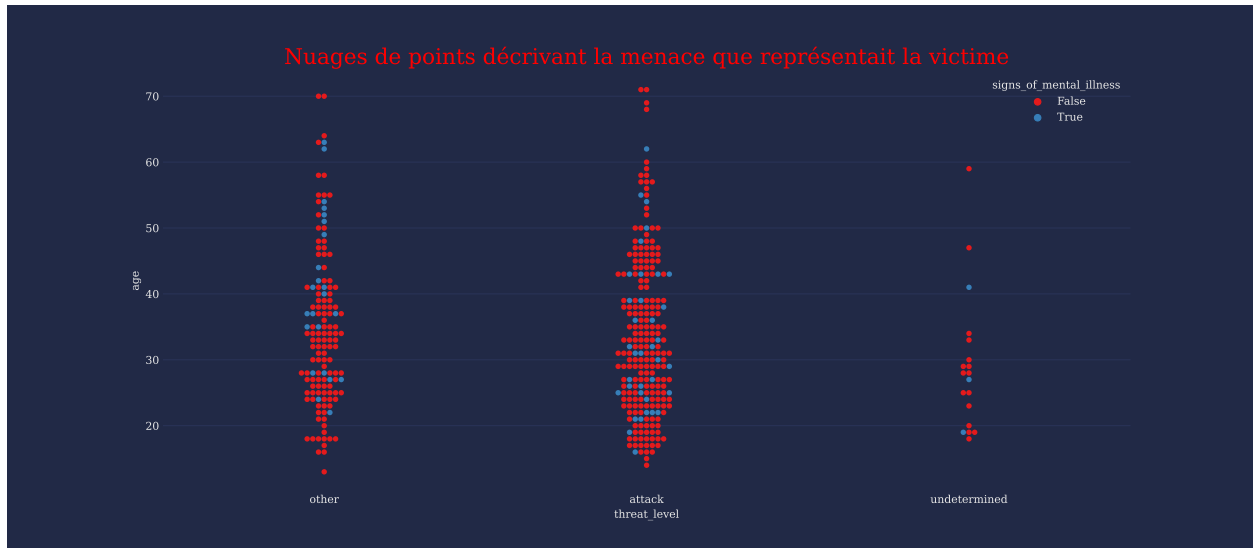
Nous avons ici effectué une boîte à moustache qui permet de voir la distribution des âges parmi les différentes communautés. On peut observer que les victimes de la communauté **noire** sont généralement **assez jeune** avec une médiane en *dessous* de **30 ans**, ce qui est vraiment **jeune**, ce chiffre est suivi quand même de très près par la communauté **hispanique** qui elle a une *médiane* d'âge de **32 ans** donc la moitié des victimes ont moins de **32 ans**. La communauté **blanche**, elle, a une *médiane* bien plus *haute* qui va s'élever à **40 ans**. Notons que la *variabilité* est d'ailleurs plus *visible* tant les valeurs semblent plus éloignées les une des autres. Par exemple le premier quartile est à moins de 30 ans, donc un quart des victimes ont moins de 30 ans, ce qui suggère que l'on peut plus fréquemment trouver des victimes de tout *âge* dans la *communauté blanche*. Visuellement, le graphique a été réalisé avec le *boxplot* de **seaborn** qui est, je trouve, très qualitatif. Encore une fois le thème de fond avec *cyberpunk* et la police est du *Times New Roman*.

```
style.use("cyberpunk")
sns.set_style({'font.family' : 'serif', 'font_serif' : ['Times New Roman']})
plt.figure(figsize = (16,7))
p1 = sns.pointplot(x = "flee"
                  , y = "age"
                  , hue = "flee"
                  , data = data2
                  , palette="Set1"
                  )
plt.title("Distribution de l'âge et la manière dont la victime s'échappait"
        , size = 16
        , color = "grey")
plt.xlabel("Manière de Fuir"
        , size = 12
        , color = "grey")
plt.legend([], [], frameon = False)
plt.show(p1)
```



On a trouvé qu'il serait très intéressant de se pencher sur la tranche d'âge des victimes qui essayaient de s'enfuir. On peut voir que les victimes en dessous de 30 ans avaient un *mode de fuite* qui consistait plutôt à **courir** (ou bien même *marcher*), là où d'autres plutôt proches de 35 et 40 ans soit n'essayaient **pas** de s'enfuir, soit le faisait *véhiculé*, notamment en voiture. Ce graphique a été réalisé avec `pointplot` de `seaborn` qui est assez parlant pour le coup, on peut facilement distinguer les différents modes opératoires. Cette différenciation est d'autant plus marquée avec la palette de `seaborn Set1`. La chose intéressante serait de voir si justement le fait que les plus jeunes victimes cherchent à s'enfuir à pied n'augmente pas le nombre de victimes de cette démographie.

```
plt.figure(figsize = (16,7))
sns.set_style({'font.family' : 'serif', 'font_serif' : ['Times New Roman']})
p2 = sns.swarmplot(x = "threat_level"
                  , y = "age"
                  , data = data2
                  , hue = "signs_of_mental_illness"
                  , palette = "Set1")
plt.title("Nuages de points décrivant la menace que représentait la victime"
        , size = 20
        , color = "red")
plt.show(p2)
```



Ici on a un **swarmplot** de la librairie **seaborn** avec la *palette Set1* de la même librairie. Ce graphique nous permet de voir la menace que représentait les victimes en fonction des différents âges ainsi qu'une distinction entre ceux qui avait des signes de troubles mentaux ou non. On peut observer que beaucoup de victime ayant entre 20 ans et 30 ans, étaient en train d'attaquer possiblement un policier mais que contrairement à ce qu'on pourrait penser, au final peu d'entre eux avait des signes de troubles mentaux. Très peu ont été classé dans un niveau de menace indéterminé, ce qui peut aussi parfois laisser place à l'interprétation, ou bien la victime n'attaquait pas, u bien les évènements se sont produits de manière trop breves pour pouvoir statuer ou bien ce sont des éléments manquants dans le rapport de police et dans ce cas, il est plus simple de ne pas déterminer le niveau de danger. Toujours est-il que finalement beaucoup de ces victimes dans les grandes villes étaient jugés comme dangereux et étaient possiblement armés.

Cette petite visualisation peut globalement nous permettre de récupérer pas mal d'information sûr les meurtres au gun par les policiers que l'on ne peut pas voir immédiatement. Le but est d'ailleurs que les graphiques soient très représentatifs de ce que l'on voudrait montrer, dans le cas présent on pu étudier plusieurs angles pour pouvoir trouver des causes, ou des éléments déclancheurs qui peuvent être évident pour certains et plus suspect pour d'autres.