

Exercice sur les statistiques

October 9, 2017

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
sns.set(color_codes = True)
from data import *
from scipy import stats

def statistiques(data):
    mini = np.min(data)
    maxi = np.max(data)
    moy = np.mean(data)
    med = np.median(data)
    std = np.std(data)

    print('Valeur minimale : ' + str(mini))
    print('Valeur maximale : ' + str(maxi))
    print('Valeur moyenne : ' + str(moy))
    print('Valeur mediane : ' + str(med))
    print('Ecart-type : ' + str(std))
    return (mini, maxi, moy, med, std)
```

```
In [2]: m0 = 0
sigma0 = 1
sigma1 = 10
N = 10000
m4 = 10
sigma4 = 10
m5 = 0.001
sigma5 = 0.01
```

0.1 Importance de la représentation graphique

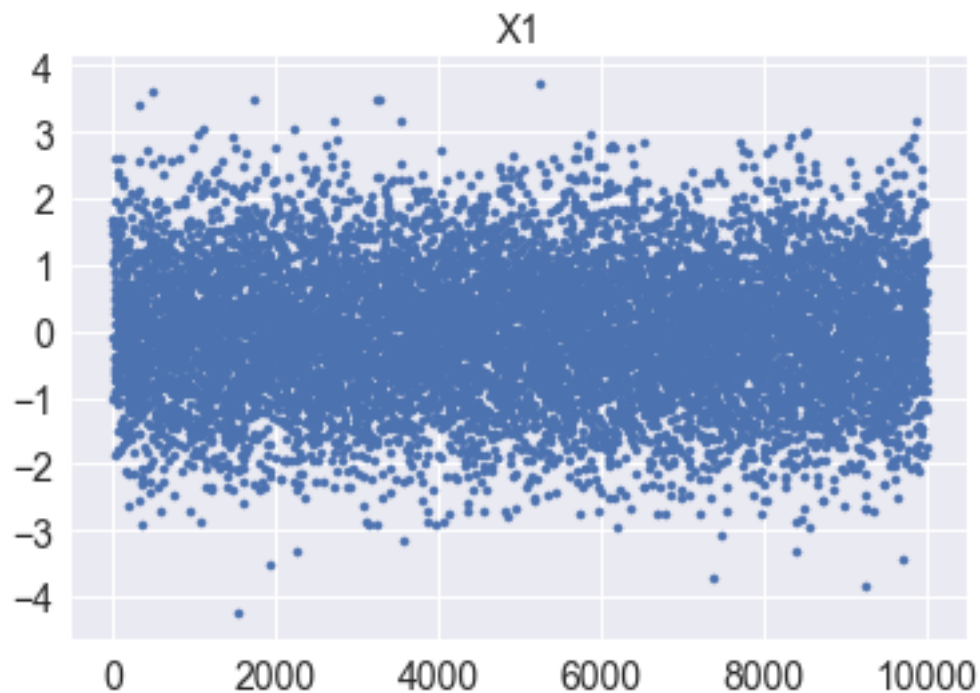
0.1.1 Jeu de données n°1

```
In [3]: X1 = np.random.normal(m0, sigma0, N)
```

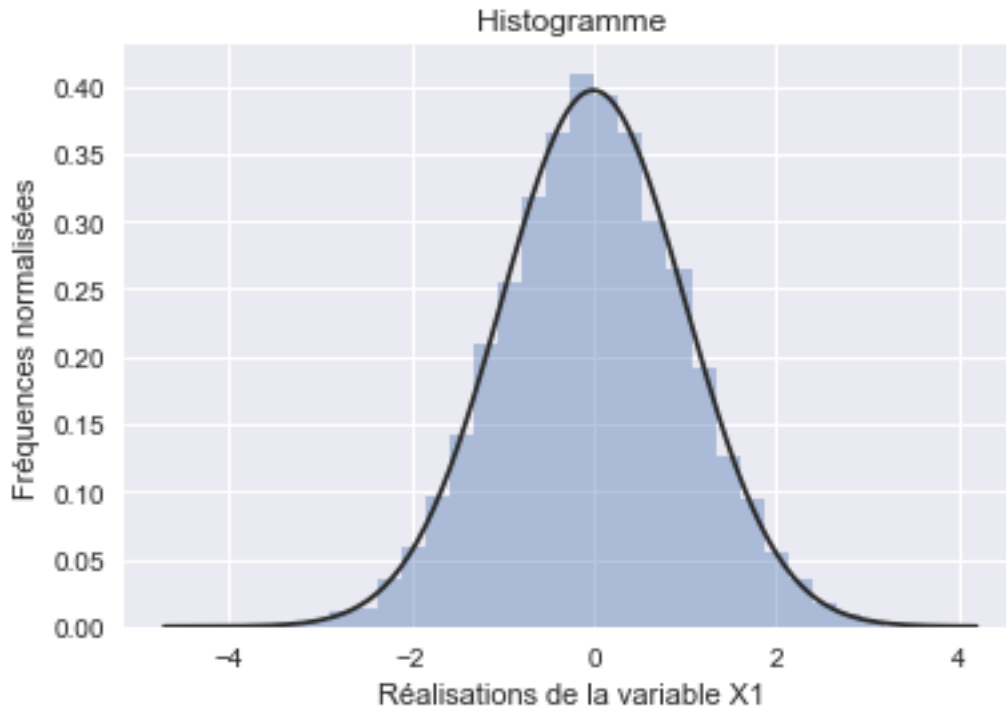
```
In [4]: (mini1, maxi1, moy1, med1, std1) = statistiques(X1)
```

Valeur minimale : -4.22884801517
Valeur maximale : 3.71570123866
Valeur moyenne : -0.00517175489011
Valeur mediane : -0.0104175686256
Ecart-type : 1.00493869437

```
In [5]: plt.figure()  
plt.scatter(np.arange(N), X1, s= 10)  
plt.title('X1', fontsize=14)  
plt.xticks(fontsize=14)  
plt.yticks(fontsize=14)  
plt.show()
```



```
In [6]: y1 = pd.Series(X1, name = "variable X1")  
sns.distplot(y1, bins=30, fit = stats.norm, kde= False);  
plt.title('Histogramme')  
plt.xlabel('Réalisations de la variable X1')  
plt.ylabel('Fréquences normalisées')  
plt.show()
```



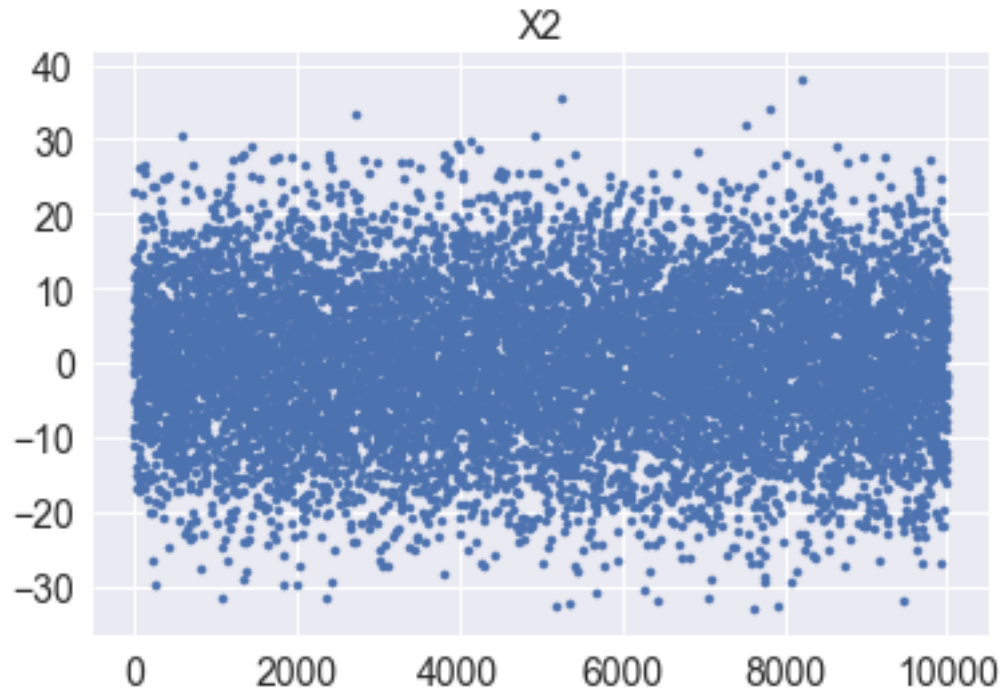
0.1.2 Jeu de données n°2

```
In [7]: X2 = np.random.normal(m0, sigma1, N)
```

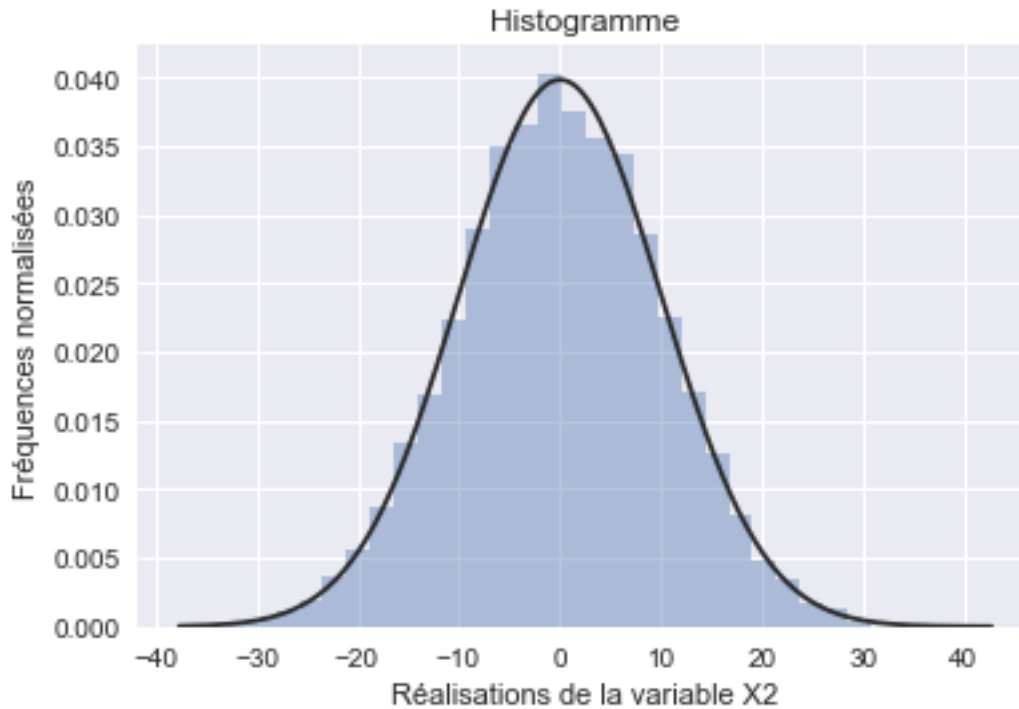
```
In [8]: (mini2, maxi2, moy2, med2, std2) = statistiques(X2)
```

```
Valeur minimale : -32.8968994457
Valeur maximale : 37.9710076558
Valeur moyenne : 0.0293498102543
Valeur mediane : -0.0292998659019
Ecart-type : 10.0191516487
```

```
In [9]: plt.figure()
plt.scatter(np.arange(N), X2, s= 10)
plt.title('X2', fontsize=14)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
plt.show()
```



```
In [10]: y2 = pd.Series(X2, name = "variable X2")
sns.distplot(y2, bins=30, fit = stats.norm, kde= False);
plt.title('Histogramme')
plt.xlabel('Réalizations de la variable X2')
plt.ylabel('Fréquences normalisées')
plt.show()
```



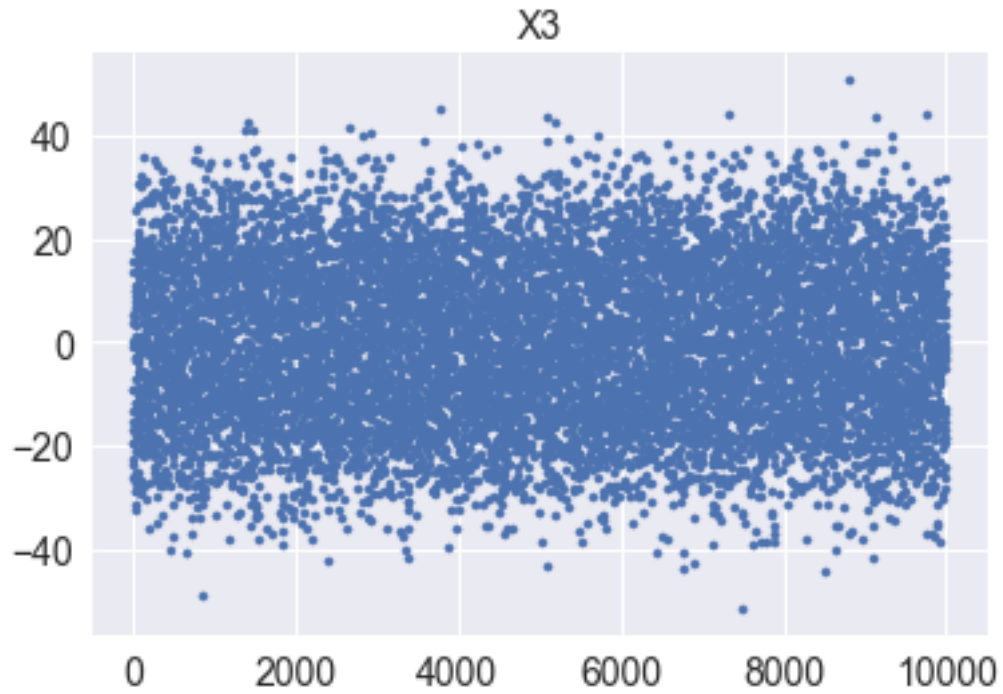
0.13 Jeu de données n°3

```
In [11]: X3 = np.zeros(N)
          X3[:int(N/2)] = np.random.normal(m0 - 12, sigma1, int(N/2))
          X3[int(N/2):] = np.random.normal(m0 + 12, sigma1, int(N/2))
          np.random.shuffle(X3)
```

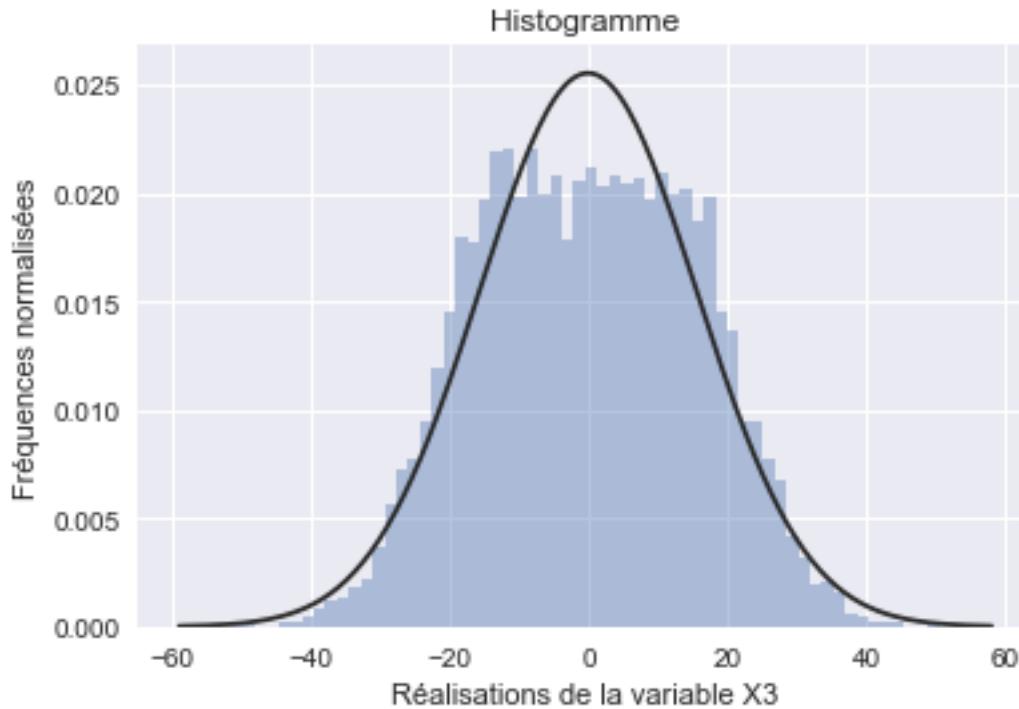
```
In [12]: (mini3, maxi3, moy3, med3, std3) = statistiques(X3)
```

```
Valeur minimale : -51.4554549113
Valeur maximale : 50.5926535554
Valeur moyenne : -0.0394216157701
Valeur mediane : 0.0116972078409
Ecart-type : 15.6255878452
```

```
In [13]: plt.figure()
          plt.scatter(np.arange(N), X3, s= 10)
          plt.title('X3', fontsize=14)
          plt.xticks(fontsize=14)
          plt.yticks(fontsize=14)
          plt.show()
```



```
In [14]: y3 = pd.Series(X3, name = "variable X3")
sns.distplot(y3, bins=60, fit = stats.norm, kde= False);
plt.title('Histogramme')
plt.xlabel('Réalizations de la variable X3')
plt.ylabel('Fréquences normalisées')
plt.show()
```



0.1.4 Jeu de données n°4

```
In [15]: X4 = np.random.uniform(-17.35, 17.35, N)
```

```
In [16]: (mini4, maxi4, moy4, med4, std4) = statistiques(X4)
```

Valeur minimale : -17.347077572

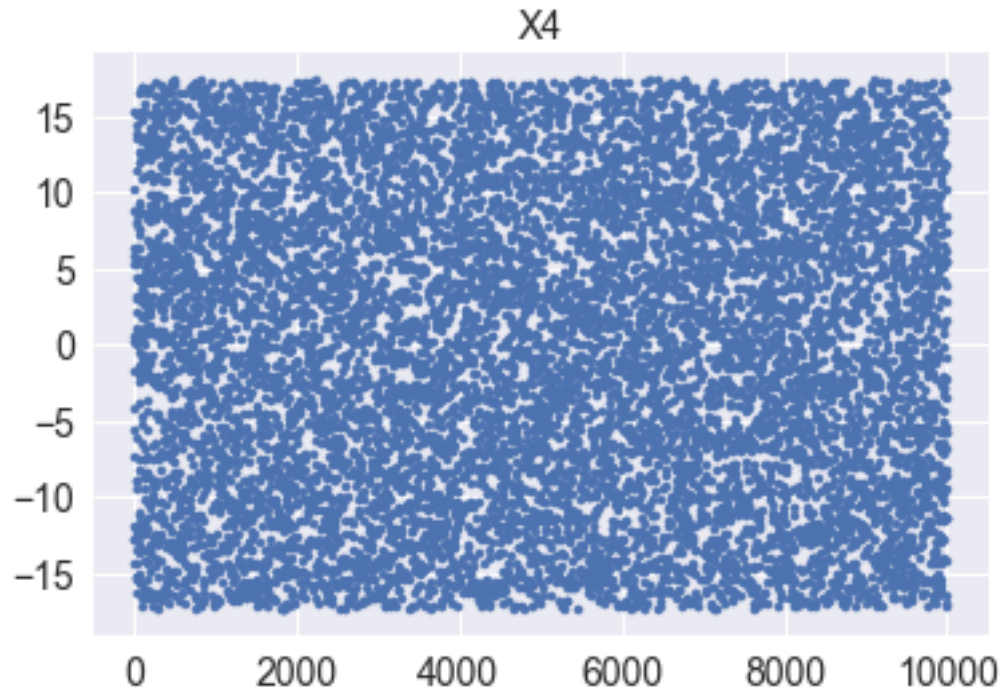
Valeur maximale : 17.34916038

Valeur moyenne : 0.100725924573

Valeur mediane : 0.125318567902

Ecart-type : 9.99251145512

```
In [18]: plt.figure()
plt.scatter(np.arange(N), X4, s= 10)
plt.title('X4',fontsize=14)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
plt.show()
```



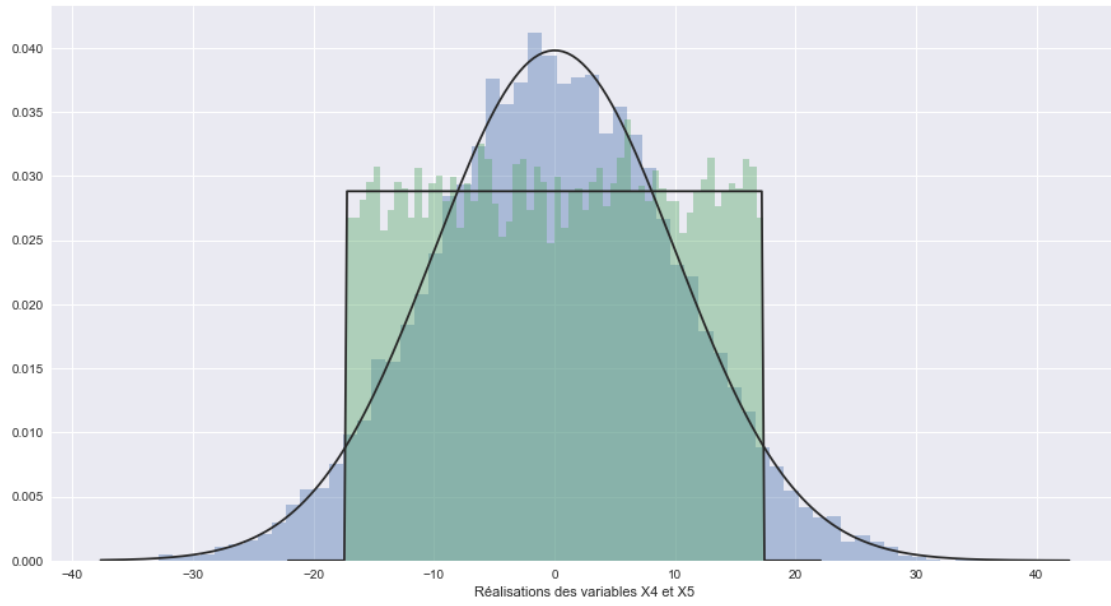
```
In [19]: y4 = pd.Series(X4, name = "variable X4")
sns.distplot(y4, bins=60, kde= False, norm_hist = True, fit = stats.uniform);
plt.title('Histogramme')
plt.xlabel('Réalisations de la variable X4')
plt.ylabel('Fréquences normalisées')
plt.show()
```




0.1.5 Représentation des deux jeux de données

In [20]: `fig, ax = plt.subplots(figsize = (15, 8))`

```
sns.distplot(y2, ax=ax, kde=False, norm_hist = True, fit = stats.norm, bins = 60)
sns.distplot(y4, ax=ax, kde=False, norm_hist = True, fit = stats.uniform, bins = 60)
plt.xlabel('Réalizations des variables X4 et X5')
plt.show()
```



In []:

In []: