

# **Analyse de données – Lab 1**

**M. Sébastien MASCHA**

**ISEP Paris - 17 Septembre 2019**

**M. Sublime**

# Lab 1 – Analyse de données

# Lab 1 – Analyse de données

## SOMMAIRE

**I. Title 1** \_\_\_\_\_ *Erreur ! Signet non défini.*

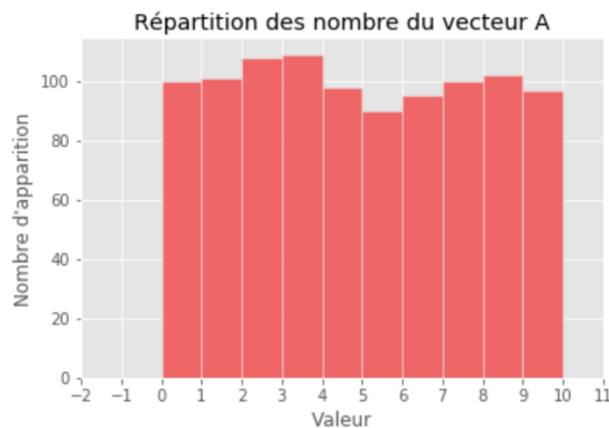
# Lab 1 – Analyse de données

## I. SERIES DISCRETES

1. Question 1 - Générez un vecteur contenant 1000 valeurs aléatoires incluses entre 0 et 10 :

`A = np.random.randint(0,10,1000)`

```
hist = plt.hist(A, bins = range(11), edgecolor='#E6E6E6', color='#EE6666')
plt.title("Répartition des nombre du vecteur A")
plt.xlabel("Valeur")
plt.ylabel("Nombre d'apparition")
plt.xticks(range(-2, 12), )
plt.show()
```



*Fig. 1 : Histogramme du vecteur A*

2. Questions 3 à 5 - Déterminez le mode, la moyenne et la médiane de ces données sans utiliser les fonctions Python prédéfinies.

Le mode de A est 1. On retrouve aussi cette valeur avec la fonction prédéfinie stats de la librairie scipy.

La médiane est 4,674.

```
moyenne = sum(A)/len(A)
```

On remarque que la mediane s'écarte de la moyenne d'autant plus que le jeu de données est grand.

3. Question 6 - Calculez le domaine de définition, l'écart-type et la variance de ces données

Le domaine de définition de A est : [ 0 , 9 ]

Son écart-type est de : 2.8737130989366317

# Lab 1 – Analyse de données

La variance est de 8.25822697499998

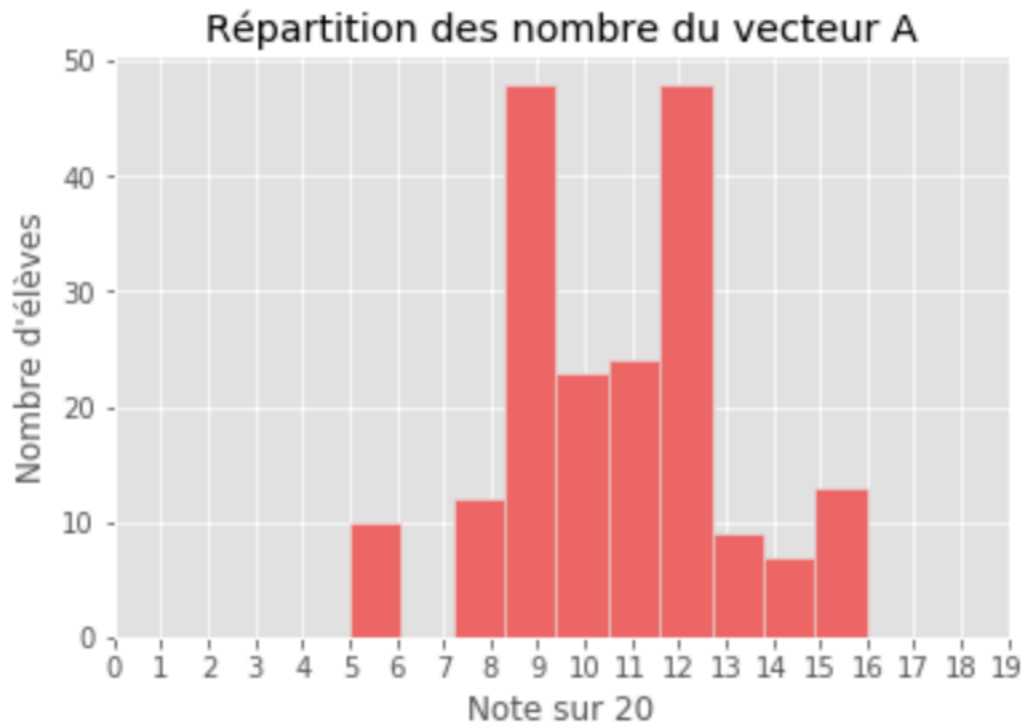
Le domaine de définition sera toujours le même, il est unique.

Pour l'écart type et la variance les valeurs peuvent changer en fonction de la méthode de calcul utilisée par les fonctions pré-définies.

# Lab 1 – Analyse de données

## II. PARTIE B - SERIES DISCRETES PAR FREQUENCE

1. Question 1 - Créez le jeu de données correspondant et représentez le sous forme d'un histogramme.



## Lab 1 – Analyse de données

```
index = []
for i in range (0, len(numbers)-1):
    if numbers[i] == np.max(numbers):
        index.append(marks[i])

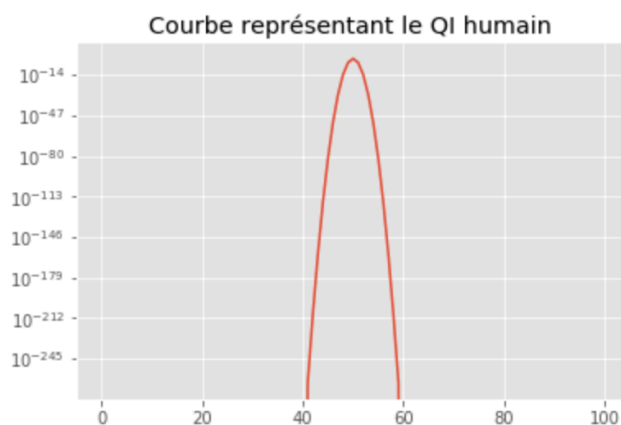
print('Mode = ', index)
print('Moyenne = ', np.mean(list_with_weights))
print('Mediane = ', np.median(list_with_weights))
print('Variance = ', np.var(list_with_weights))
print('Ecart-Type = ', np.std(list_with_weights))
```

```
Mode = [9, 12]
Moyenne = 10.675257731958762
Mediane = 11.0
Variance = 5.848150706770114
Ecart-Type = 2.4182949999473005
```

### III. PARTIE C - DISTRIBUTION GAUSSIENNES

```
sigma_human_iq = np.sqrt(225)
mean = 100

p = (1 / sigma_human_iq * np.sqrt(2*math.pi)) * math.exp(-(0-mean)**2 / 2 * sigma_human_iq)
result = []
for i in range (50, 150):
    result.append(1/sigma_human_iq*np.sqrt(2*(math.pi))*math.exp(-((i-mean)**2)/2*sigma_human_iq))
plt.semilogy(result)
plt.title('Courbe représentant le QI humain')
plt.show()
```



# Lab 1 – Analyse de données

Question 2 - Générez un échantillon de taille 100.000 suivant cette loi et affichez l'histogramme correspondant.

```
data = np.random.normal(100, np.sqrt(225), 100000)
print("Moyenne = ", np.mean(data))
print("Variance = ", np.var(data))

minus_sixty = 0
over_130 = 0
for i in range(0, len(data)-1):
    if data[i] < 60:
        minus_sixty+=1
    if data[i] > 130:
        over_130+=1

print("Pourcentage de QI < 60 = ", minus_sixty/len(data))
print("Pourcentage de QI > 130 = ", over_130/len(data))

print("Intervalle de confiance a 95 pourcent = [", mean-1.96*sigma_human_iq, ',', mean+1.96*sigma_human_iq, ']')

Moyenne = 100.0519926349823
Variance = 225.6182246237175
Pourcentage de QI < 60 = 0.00372
Pourcentage de QI > 130 = 0.02342
Intervalle de confiance a 95 pourcent = [ 70.6 , 129.4 ]
```