Informatique en CPGE (2017-2018)
Exercices : probabilités

Exercice 1 : coefficients binomiaux

1. Ecrire une fonction **factorielle(n)** qui prend en argument un entier naturel n et renvoie la valeur de $n\,!$, en utilisant une simple boucle for.
   Afficher les valeurs de $n\,!$ pour n variant de 0 à 10.
2. Afficher les valeurs de $n\,!$ pour n variant de 0 à 10 en utilisant la fonction **factorial** du module **math**.
3. Ecrire une fonction **combinaisons1(n, k)** qui prend en arguments deux entiers naturels n et k avec $k \leq n$, et renvoie le coefficient $\binom{n}{k} = \dfrac{n!}{k!(n-k)!}$. On utilise ici la définition de $\binom{n}{k}$.
4. Ecrire une fonction **combinaisons2(n, k)** qui prend en arguments deux entiers naturels n et k avec $k \leq n$, et renvoie le coefficient $\binom{n}{k} = \dfrac{n \times (n-1) \times \ldots \times (n-k+1)}{k!}$. On utilise ici une simplification du quotient.
5. Ecrire une fonction récursive **cb(n, k)** qui prend en arguments deux entiers naturels n et k avec $k \leq n$, et renvoie le coefficient $\binom{n}{k}$ en utilisant la propriété : $\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$.
6. Tester et comprendre le code suivant utilisant la fonction **combinations** du module **itertools** :

```
from itertools import combinations
print(list(combinations(range(5),3)))
```

   Tester alors le code suivant :

```
from itertools import combinations
def choix(n,k):
    return len(list(combinations(range(n),k)))

print("\nAvec itertools")
for k in range(11):
    print("n=10,","k="+str(k)+'\t',(choix(10,k)))
```

7. Tester le code suivant utilisant la fonction **comb** du module **scipy.misc** :

```
from scipy.misc import comb
print("\nAvec comb de scipy.misc")
for k in range(11):
    print("n=10,","k="+str(k)+'\t',int(comb(10,k)))
```

8. Tester le code suivant utilisant la fonction **binom** du module **scipy.special** :

```
from scipy.special import binom
print("\nAvec binom de scipy.special")
for k in range(11):
    print("n=10,","k="+str(k)+'\t',int(binom(10,k)))
```

Exercice 2 : loi faible des grands nombres

On considère une variable aléatoire X suivant la loi binomiale $\mathcal{B}(n;p)$ de paramètres $n = 10$ et $p = 0,2$.

1. Tester le code suivant permettant de simuler et d'afficher une réalisation de X :

```
from numpy import random as rd
n,p=10,0.2
print(rd.binomial(n,p))
```

   Pour afficher une liste de 50 simulations par exemple, on écrira :

```
print(list(rd.binomial(n,p,size=50)))
```

2. Ecrire le code permettant de simuler 20 réalisations de la variable aléatoire X puis de calculer et afficher la moyenne $\dfrac{1}{20}\sum x_i$, afin de la comparer avec l'espérance $E(X) = np$.
3. Même question que la précédente mais avec 50 puis 100 et enfin 1000 simulations.
4. Une conséquence de la loi des grands nombres.
   Ecrire le code permettant de simuler 100 réalisations de la variable aléatoire X puis de calculer et afficher, afin de les comparer, les fréquences d'obtention de k pour $k \in \{0, 1, \dots, n\}$ et les probabilités $P(X = k) = \dbinom{n}{k} p^k(1-p)^{n-k}$.
5. Même question que la précédente mais avec 1000 puis 10000 et enfin 100000 simulations.

Exercice 3 : loi binomiale et loi de Poisson

1. Tester le code suivant permettant de simuler et d'afficher une liste de 50 réalisations de X où X une variable aléatoire suivant la loi binomiale $\mathcal{B}(n;p)$ de paramètres $n = 25$ et $p = 0,2$ (on utilise ici la bibliothèque scipy) :

```
from scipy.stats import binom
n,p=25,0.2
va=binom(n,p)
test=list(va.rvs(size=50))
print(test)
```

2. Ecrire le code permettant de simuler et d'obtenir une liste de 1000 réalisations de X puis d'obtenir une figure similaire à la figure 1 présentée ci-dessous. On utilisera le code :

```
plt.plot([i],[k],marker="o",color="blue")
```

3. Ecrire le code permettant de simuler et d'obtenir une liste de 10000 réalisations de Y, une variable aléatoire suivant la loi binomiale $\mathcal{B}(n;p)$ de paramètres $n = 100$ et $p = 0,05$.
4. Ecrire le code permettant d'obtenir une figure similaire à la figure 2 présentée ci-dessous. En bleue, les réalisations de la variable X, en rouge celles de la variable Y et en vert la courbe correspondant à la loi de Poisson de paramètre $\lambda = 5$. (Attention au choix de l'échelle en ordonnées).

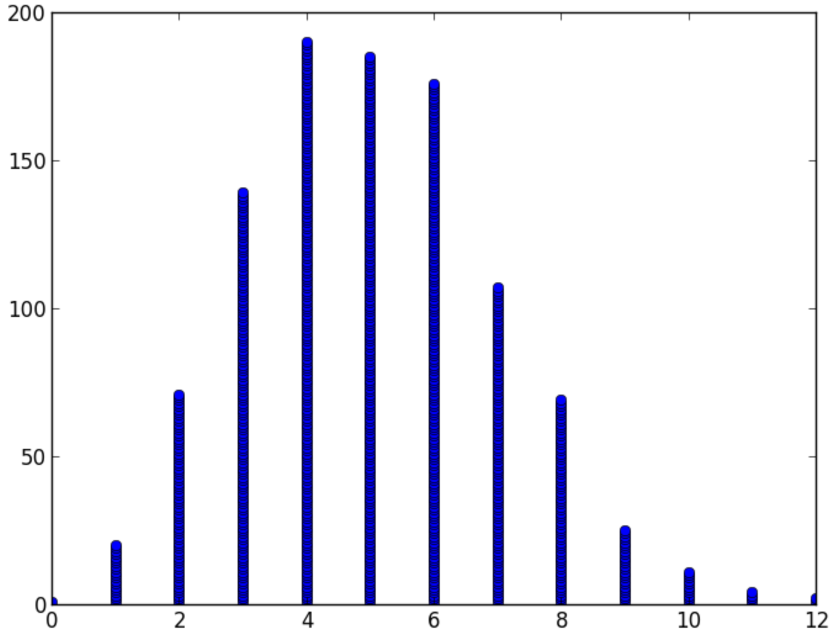

FIGURE 1 – 1000 réalisations de la loi binomiale B(25,0.2)

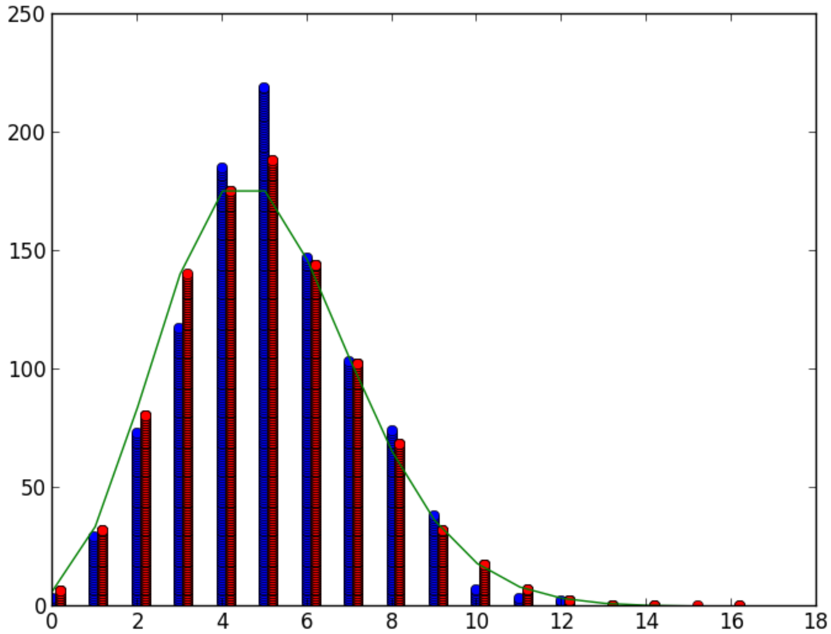

FIGURE 2