TD12

ENCORE DES COUPLES.

Exercice 1

Une urne contient des boules blanches (en proportion p), des boules noires (en proportion q) et des boules rouges (en proportion r). On a donc $p+q+r=1$. On effectue des tirages successifs avec remise dans cette urne. Pour tout $i \in \mathbb{N}^*$, on note T_i la variable aléatoire qui vaut 1 si la i-ème boule tirée est blanche, -1 si elle est noire et 0 si elle est rouge.

Les tirages étant effectués avec remise, les variables T_i sont mutuellement indépendantes.

Partie A.

On note ensuite X_1 la variable égale au numéro du tirage qui amène pour la première fois une boule blanche et X_2 celui qui correspond au tirage où sort pour la première fois la deuxième boule blanche.

Par exemple, si les premiers tirages donnent *Noire, Noire, Rouge, Blanche, Rouge, Blanche,...*, on a $X_1 = 4$ et $X_2 = 6$.

1. Expliciter, pour tout $i \in \mathbb{N}^*$, la loi de T_i. Calculer son espérance et sa variance.
2. Reconnaître la loi de X_1. Rappeler son espérance et sa variance.
3. Compléter la fonction Python suivante afin qu'elle simule les variables aléatoires X_1 et X_2. Pourquoi la fonction ne prend-elle en argument que la proportion p de boules blanches ?

```
import numpy as np
import numpy.random as rd

def simul_X1_X2(p) :
  x1 =
  while ....... :
    x1 = .......
  x2 = .......
  while ....... :
    x2 = .......
  return [x1,x2]
```

4. **a.** Expliciter la loi conjointe du couple (X_1, X_2).

   b. En déduire la loi marginale de X_2.

   c. Montrer que X_2 admet une espérance et que $\mathbb{E}(X_2) = \frac{2}{p}$.
5. On note $U_2 = X_2 - X_1$.

   a. Pour tout $j \in \mathbb{N}^*$, déterminer $\mathbb{P}([U_2 = j])$.
   En déduire que X_1 et U_2 suivent la même loi, puis que U_2 admet une variance et préciser sa valeur.

   b. Montrer que U_2 est indépendante de X_1.

ECG 2 Maths Appliquées, http://louismerlin.fr.

c. Exprimer X_2 en fonction de U_2 et de X_1 et en déduire que X_2 admet une variance que l'on explicitera.

d. Que vaut $\mathrm{cov}(X_1, X_2)$? Les variables X_1 et X_2 sont-elles indépendantes ?

e. On rajoute les instructions Python suivantes à la suite de la fonction précédente. Que peut-on prévoir sur le retour après exécution de celles-ci ?

```python
p = 1/4
L = []; M = [];
for k in range(10000) :
  [X1,X2] = simul_X1_X2(p)
  L = L.append(X1)
  M = M.append(X2)
U = [M(k) - L(k) for k in range(10000)]
print(np.mean(L[k]*U(k) for k in range(10000)]) - np.mean(L)*np.
    mean(U)
```

Partie B.

On note W la variable aléatoire correspondant au nombres de boules rouges obtenues avant l'apparition de la première boule blanche.

6. Pour tout $i \in \mathbb{N}^*$, déterminer la loi conditionnelle de W sachant $[X_1 = i]$.

7. En déduire que la loi de W est donnée par la somme, pour tout $k \in \mathbb{N}$,

$$\mathbb{P}([W = k]) = p\left(\frac{r}{q+r}\right)^k \sum_{i=k+1}^{+\infty} \binom{i-1}{k} \left(\frac{q}{q+r}\right)^{i-1-k} (1-p)^{i-1}.$$

8. Vérifier que pour tout $i \in \mathbb{N}^*$ et pour tout $k \in [\![0, i]\!]$,

$$k\binom{i}{k} = i\binom{i-1}{k-1}.$$

9. En admettant qu'il est licite de permuter les sommes

$$\sum_{k=0}^{+\infty} \sum_{i=k+1}^{+\infty} \cdots = \sum_{i=1}^{+\infty} \sum_{k=0}^{i-1} \cdots,$$

et que W admet une espérance, montrer que

$$\mathbb{E}(W) = \frac{r}{p}.$$

Partie C.

On note, pour tout $n \in \mathbb{N}^*$, $S_n = T_1 + \cdots + T_n$.

10. Quelle est la loi de S_1 ? Préciser son espérance et sa variance.

11. Expliciter l'espérance et la variance de S_n.

12. Soit $t > 0$. On pose $V_n = t^{S_n}$.

a. Déterminer la loi, l'espérance et la variance de V_1

b. En déduire l'expression de $\mathbb{E}(V_n)$.

Exercice 2

Soit n un entier naturel supérieur ou égal à 3. Une urne contient une boule noire non numérotée et $n-1$ boules blanches dont $n-2$ portent le numéro 0 et une porte le numéro 1. On extrait ces boules au hasard, une à une, sans remise jusqu'à l'apparition de la boule noire.

Pour chaque $i \in [\![1, n-1]\!]$, on note B_i l'événement : "le i-ème tirage donne une boule blanche" et on pose $\overline{B_i} = N_i$. Enfin, on note X la variable aléatoire égale au rang d'apparition de la boule noire et Y la variable aléatoire qui vaut 1 si la boule numérotée 1 a été piochée lors de l'expérience précédente et 0 sinon.

Partie I - Simulation informatique.

1. Compléter la fonction Python suivante afin qu'elle simule l'expérience aléatoire décrite dans cet exercice et renvoie les valeurs prises par X et Y.

   On conviendra que la boule noire est désignée par le nombre `nB+1` où `nB` désigne le nombre de boules blanches.

```
import numpy as np
import numpy.random as rd

def silul_XY(n) :
  nB = n-1
  x = 1
  y = .......
  u = rd.randint(1,nB+2)
  while (u < nB+1) :
    nB = ......
    if u == 1 :
      y = .......
    u = rd.randint(.......,.......)
    x = .......
  return .......
```

2. Écrire une suite de commandes permettant, pour un n quelconque, d'obtenir la *covariance empirique* d'un échantillon de taille 1000 du couple (X, Y).
3. En rentrant $n = 4$, le programme affiche `0.449475`. Que peut-on conjecturer.

Partie II - Loi conjointe de X et Y.

4. Donner l'ensemble $X(\Omega)$ des valeurs que peut prendre la variable X.
5. **a.** Pour tout i de $[\![2, n-1]\!]$, justifier que
$$P_{B_1\cap\cdots\cap B_{i-1}}(B_i) = \frac{n-1}{n-i+1}.$$
   b. En déduire $\mathbb{P}([X = k])$ pour tout $k \in X(\Omega)$.

   c. Reconnaître la loi de X et donner son espérance et sa variance.
6. **a.** Pour tout k de $X(\Omega)$, montrer que
$$\mathbb{P}([X = k] \cap [Y = 0]) = \frac{n-k}{n(n-1)}.$$
   b. En déduire $\mathbb{P}([Y = 0])$.

   c. Reconnaître la loi de Y et donner son espérance et sa variance.

   d. Les variables X et Y sont-elles indépendantes ?

Partie III - Un calcul de covariance.

7. **a.** On considère deux nombres entiers naturels k et n tels que $n \geqslant k$. Rappeler le lien entre les nombres
$$\binom{k}{j}, \quad \binom{k+1}{j+1} \quad \text{et} \quad \binom{k}{j+1}.$$
   b. Établir alors la formule
$$dsum_{k=j}^{n}\binom{k}{j} = \binom{n+1}{j+1}.$$
   c. En faisant $j = 2$, en déduire une expression factorisée de la somme suivante
$$\sum_{k=2}^{n} k(k-1).$$

8. Montrer que

$$\mathbb{E}(XY) = \sum_{k=1}^{n} k\mathbb{P}\left(crochet X = k \cap [Y=1]\right) = \frac{n+1}{3}.$$

9. En déduire la valeur de $\operatorname{cov}(X,Y)$.

Exercice 3 Librement inspiré de **HEC/ESSEC** 2022 sujet 0.

Dans tout cet exercice, n désigne un entier supérieur ou égal à 2 et p un réel de l'intervalle $]0,1[$. Soit $S = \{s_1, \cdots, s_n\}$ un ensemble fini de n sommets. On s'intéresse dans cet exercice à des graphes aléatoires construits à partir de l'ensemble S.

Plus précisément, pour tout couple (i,j) de $[\![1,n]\!]^2$ avec $i<j$, on introduit les variables $T_{i,j}$ mutuellement indépendantes de même loi $\mathcal{B}(p)$. Les arêtes du graphe sont les paires de sommets $\{s_i, s_j\}$ pour lesquelles $T_{i,j} = 1$.

On dit qu'un sommet est isolé s'il n'y a aucune arête incidente à ce sommet.

On introduit la variable aléatoire N_n égale au nombre d'arêtes du graphe. Pour tout $k \in [\![1,n]\!]$, on note D_k la variable aléatoire qui prend pour valeur le degré du sommet s_k (c'est-à-dir le nombre d'arêtes incidentes à ce sommet).

Enfin on note X_k la variable aléatoire qui vaut 1 si le sommet s_k est isolé et 0 sinon, puis Z_n celle égale au nombre de sommets isolés du graphe.

1. Recopier et compléter le fonction Python ci-dessous qui renvoie la liste d'adjacence d'un tel graphe aléatoire en prenant pour argument la liste des sommets S et p.

```python
import numpy as np
import numpy.random as rd

def list_adj(S,p) :
  n = .......
  l = [ [] for k in range(n)]
  for i in range(n-1) :
    for j in range(i+1,n) :
      if rd.random() <= p :
        l[i].append(.......)
        l[j].append(.......)
  return l
```

2. On exécute alors la commande suivante qui

```python
print(list_adj('abcdef', 1/3))
```

qui génère l'affichage suivant

```
>>>
[['b','e','f'], ['a','c','f'], ['b'], [], ['a'], ['a','b']]
```

a. Donner une représentation graphique du graphe aléatoire généré par cette exécution.

b. Ce graphe contient-il des sommets isolés ? Si oui, combien ?

c. Expliciter la matrice d'adjacence de ce graphe. Quelle propriété a-t-elle ? Pourquoi ?

3. Écrire une fonction d'en-tête def nb_som_is(l) qui prend en argument la liste l d'adjacence d'un graphe renvoie le nombre de sommets isolés de celui-ci.

4. On dispose de la fonction mystère suivante

```python
def mystere(S,p) :
  return np.mean([nb_som_is(list_adj(S,p)) for k in range(1000)])
```

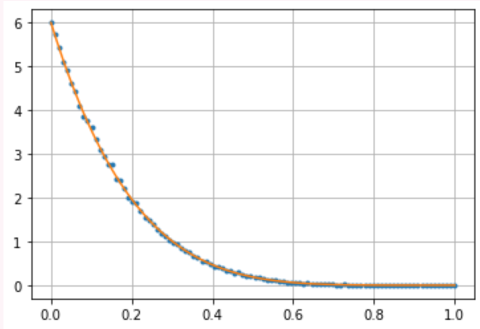
a. Que renvoie-t-elle ?

b. On ajoute les instructions suivantes dont l'exécution produit la figure ci-contre. Interpréter et émettre une conjecture.

```
import matplotlib.pyplot as plt

S = 'abcdef'
x = np.linspace(0,1,100)
y = [mystere(S,p) for p in x]
w = [len(S)*(1-p)**(len(S)-1) for p in x]
plt.grid()
plt.plot(x,y,'.')
plt.plot(x,w)
plt.show()
```

Affichage Python



5. **a.** Justifier que $N_n(\Omega) = [\![0, \binom{n}{2}]\!]$.

b. Montrer que

$$\mathbb{P}([N_n = 0]) = \mathbb{P}\left(\bigcap_{i=1}^{n-1} \bigcap_{j=i+1}^{n} [T_{i,j} = 0]\right) = (1-p)^{\frac{n(n-1)}{2}}.$$

c. Que vaut $\mathbb{P}\left(\left[N = \binom{n}{2}\right]\right)$?

6. **a.** Justifier que pour tout $k \in [\![1, n]\!]$, on a

$$D_k = \sum_{i=1}^{k-1} T_{i,k} + \sum_{i=k+1}^{n} T_{k,i}.$$

En déduire la loi de D_k.

b. Montrer que pour tout $1 \leqslant k < l \leqslant n$, on a

$$\operatorname{cov}(D_k, D_l) = p(1-p).$$

Les variables D_k et D_l sont-elles indépendantes ?

c. Déterminer $\mathbb{E}(Z_n)$. Est-ce cohérent avec la conjecture précédente ?

7. **a.** Montrer que pour tous $i < j$,

$$\mathbb{P}([X_i = 1] \cap [X_j = 1]) = (1-p)^{2n-3}.$$

b. En observant que

$$Z_n^2 = \sum_{i=1}^{n} X_i + 2 \sum_{1 \leqslant i < j \leqslant n} X_i X_j,$$

montrer que

$$\mathbb{E}(Z_n^2) = n(1-p)^{n-1} + n(n-1)(1-p)^{2n-3}.$$