Code de partage avec Capytale : b68a-1487501

Echauffement - partie de badminton

Au cours d'une partie de badminton, on suppose que, lors de chaque échange, le joueur qui a le service emporte le point avec une probabilité $\frac{2}{3}$ et le perd avec une probabilité $\frac{1}{3}$ On suppose que c'est le jouer A qui a le service lors du premier échange. Ensuite, selon les règles de ce jeu, celui qui emporte l'échange marque un point et obtient le service pour l'échange suivant.

On commence par jouer un point, compléter le programme ci-contre pour qu'il renvoie 1 dans deux cas sur trois et 2 sinon :

On modélise ici l'aléa avec une fonction `random()` qui donne un nombre aléatoire sur l'intervalle $[0, 1[$, il y a donc deux chances sur trois que ce nombre soit strictement inférieur à $\frac{2}{3}$

```
import numpy.random as rd

def point():
    a=rd.random()
    if a<2/3 :
      return 1
    else :
      return 2
```

puis compléter le programme suivant pour simuler le déroulement d'une manche :

On voit entre autres que le premier `if` et le `else` correspondant ont des rôles symétriques (pour A et pour B).
On peut éventuellement ajouter une condition dans le `while` pour prendre en compte les deux points d'écart : `or np.abs(s_a-s_b)<2` (il faut importer `numpy` dans ce cas pour la valeur absolue.

```
s_a=0 # score du joueur A au début
s_b=0 # score du joueur B au début
service=-1 # A sert, passe à 1 si B sert
while max([s_a, s_b])<21 # on continue tant qu'aucun joueur n'a atteint 21
    p=point()
    if service<0 : # si A sert
      if p==1 : # i.e. si le serveur gagne le point
        s_a=s_a+1 # on augmente le score de A
      else : # i.e. si le receveur gagne le point
        s_b=s_b+1 # on augmente le score de B
        service=-service # le service change de main
    else :  # si B sert
      if p==1 : # i.e. si le serveur gagne le point
        s_b=s_b+1 # on augment le score de B
      else : # i.e. si le receveur gagne le point
        s_a=s_a+1 # on augmente le score de A
        service=-service # le service change de main
if s_a== 21 : # on peut aussi écrire if s_a>s_b
  print("le joueur A a gagné")
else :
  print("le joueur B a gagné")
print([s_a,s_b]) # pour voir le score
```

Exercice 1 - à partir de Ecricome 2015

N est un entier supérieur ou égal à 3. Une urne contient une boule noire et $N-1$ boules blanches. On effectue des tirages **sans remise** dans l'urne jusqu'à l'obtention de la boule noire.
On note X la variable aléatoire qui prend pour valeur le nombre de tirages nécessaires pour l'obtention de la boule noire. On notera pour tout entier naturel i non nul :

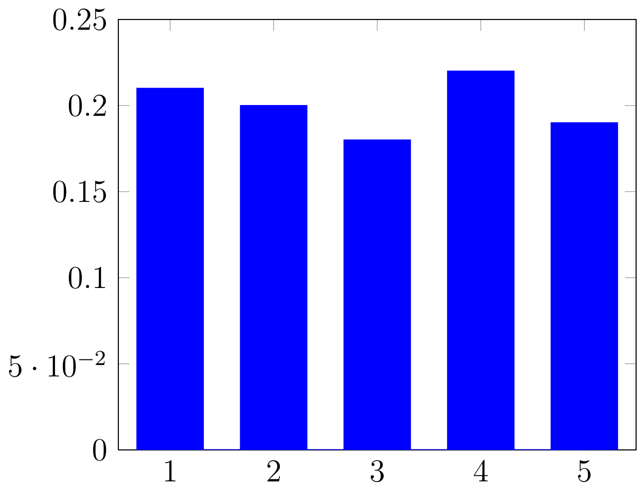
- N_i l'événement « on tire une boule noire lors du i-ième tirage » ;
- B_i l'événement « on tire une boule blanche lors du i-ième tirage ».

1. On simule 10 000 fois cette expérience aléatoire.
   Recopier et compléter le programme Python suivant pour qu'il affiche l'histogramme donnant la fréquence d'apparition du rang d'obtention de la boule noire :

   Il faut modéliser le fait que les tirages continuent tant que la boule noire n'est pas tirée. S'il reste `M` boules, la probabilité de tirer une boule blanche est $\dfrac{M-1}{M}$. On le modélise par **`randint(1,M)>1`**, donc si le nombre entier aléatoire est compris entre 2 et M, on continue (on considère que la boule noire est la boule n°1).

```
import numpy as np
import numpy.random as rd
import matplotlib.pyplot as plt
N = int(input('Donner un entier naturel non nul'))
S = np.zeros(N)
for k in range(0, 10000) :
    i = 0 # compteur du nombre de tirages
    M = N # nombre total de boules, qui évolue
    while rd.randint(1,M+1)>1 : # proba (M-1)/M de tirer une blanche
        i = i + 1 # le nombre de tirages augmente d'une unité
        M = M-1 # on enlève une boule
    S[i] = S[i]+ 1 # on augmente la fréquence du tirage i
print(S / 10000) #
x=np.arange(1,N+1,1) # il faut des abscisses pour les histogrammes
plt.bar(x,S/10000) # on représente les fréquences sous forme d'histogramme
plt.show()
```

2. On exécute le programme complété ci-dessus. On entre 5 au clavier et on obtient l'histogramme suivant (*donné dans l'énoncé*).
   Quelle conjecture pouvez-vous émettre sur la loi de la variable aléatoire X ?



   On obtient des valeurs très proches pour les fréquences d'apparition de la boule noire pour chaque tirage (du premier au cinquième). On peut donc supposer qu'il s'agit d'une loi uniforme (nous n'avons pas encore vu ce terme, c'est une équiprobabilité pour chaque valeur possible de la variable aléatoire), ce que nous allons démontrer.

3. Retrouver ce résultat en calculant $P(X=1), P(X=2), P(X=3), \dots$
   En notant N_k : « On tire une boule blanche au $k^{\text{ème}}$ », on a :
   $P(X=1) = P(N_1) = \dfrac{1}{N}$
   $P(X=2) = P(N_2) = P(\overline{N_1})P_{\overline{N_1}}(N_2) = \dfrac{N-1}{N} \times \dfrac{1}{N-1} = \dfrac{1}{N}$
   $P(X=k) = P(N_k) = P(\overline{N_1})P_{\overline{N_1}}(\overline{N_2})\dots P_{\overline{N_1}\cap\dots\cap\overline{N_{k-1}}}(N_k) = \prod_{i=1}^{k-1}\left(\dfrac{N-i}{N-i+1}\right) \times \dfrac{1}{N-k+1}$
   donc $P(X=k) = \dfrac{1}{N}$, donc la probabilité est bien uniforme.