FLUCTUATION D’ÉCHANTILLONNAGE 17.1

I) Une probabilité difficile à déterminer

On cherche la probabilité p que la somme obtenue en lançant trois dés soit supérieure ou égale à 13.
Peut-on facilement déterminer p à l’aide d’un tableau ? d’un arbre ?

II) Première simulation

Ne sachant pas calculer facilement la probabilité p ci-dessus, on décide de faire des essais. On pourrait pour cela prendre 3 dés et les lancer un certain nombre de fois en notant les sommes obtenues : Possible mais fastidieux…
Pour gagner du temps, on décide de créer un script Python en simulant les lancers de dés avec la fonction « randint(1,6) »

1) Cette fonction randint() fait partie du module « random » de Python. Il faut donc taper en début de script : « from random import * » pour que randint() soit reconnue par Python.
   Écrire un script affichant le résultat d’un lancer de dé et exécuter ce script plusieurs fois. Vérifier que l’on obtient bien des entiers compris entre 1 et 6.
2) Y a-t-il une différence entre les deux instructions suivantes :
   « randint(1,6) + randint(1,6) + randint(1,6) » et « 3×randint(1,6) » ?
   Laquelle de ces deux instructions permet de simuler correctement une somme de 3 lancers de dés ?
3) Écrire un algorithme permettant de simuler 20 fois le lancer de trois dés et d’afficher les sommes obtenues.
4) Modifier cet algorithme pour qu'il affiche plutôt en fin d'exécution la fréquence f des lancers dont la somme aura été supérieure ou égale à 13.
5) En exécutant cinq fois cet algorithme, on obtient cinq échantillons de 20 lancers de trois dés. Noter les valeurs de f obtenues. Sont-elles proches les unes des autres ? Permettent-elles d’avoir une bonne idée de la valeur de p ?

Remarque : la valeur de f « fluctue » selon les échantillons. Ce phénomène est appelé « fluctuation d’échantillonnage ».
Plus la fluctuation est forte, moins la valeur de f obtenue pour un échantillon pris au hasard nous donne une approximation fiable de la valeur de p !

III) Deuxième simulation

On se demande si la taille des échantillons a une influence sur la fluctuation entre les échantillons.
Pour cela on étudie le script ci-dessous qui simule une série de 10 échantillons :

```
from random import *
n=100
m=1
M=0
for i in range(10):
    a=0
    for j in range(n):
        s=randint(1,6)+randint(1,6)+randint(1,6)
        if s>=13:
            a=a+1
    f=a/n
    print(f)
    if f<m:
        m=f
    if f>M:
        M=f
print(m,M)
```

Répondre aux questions 1) à 3) sans lancer le script !!

1) A chaque lancer de dés, que contient la variable « s » ?
2) Pour chaque échantillon, que contiennent « n » et « f » ?
3) En langage Python, « m » et « M » désignent deux variables différentes. On voit dans le bas du script que ces deux variables prennent la valeur de « f » dans certaines conditions. Que contiennent-elles en fin de script ?

Tester le script plusieurs fois et vérifiez votre réponse à la question 3

4) Essayer le script en modifiant la taille des échantillons : 100, 1000, 10000.
   La taille des échantillons a-t-elle une influence sur la fluctuation ?
5) Proposer une estimation de p (valeur et précision)

IV) Bilan

Le calcul exact de p donne : $p=\frac{56}{6^3}$. Vous étiez vous trompé de beaucoup ?