TP 9 : Autour de la loi faible des grands nombres

Calcul approché de la probabilité d'un événement A.

Protocole :

Je vous donne une pièce truquée. Quelle expérience pourriez-vous faire pour trouver une valeur approchée de la probabilité d'obtenir "pile" ?

Formalisation mathématique :

Pour tout $i \in \mathbb{N}^*$, on considère la variable aléatoire X_i qui vaut 1 si le i^e lancer donne pile, et 0 sinon. Alors pour tout $i \in \mathbb{N}^*$, X_i suit la même loi   , et les variables (X_i) sont mutuellement indépendantes, puisque les lancers le sont.

On montrera bientôt le résultat suivant (appelé loi faible des grands nombres) :

Si (X_n) est une suite de variables aléatoires indépendantes, de même loi et admettant une variance, alors

$$\frac{X_1 + X_2 + ... + X_n}{n} \underset{n\to+\infty}{\longrightarrow} E(X_1)$$

Le sens de cette limite sera défini à ce moment là.

Dans le contexte précédent, comme les variables X_i ne prennent que la valeur 1 ou 0, la variable $X_1+X_2+...+X_n$ est égale au nombre de "pile" obtenus. Donc $\frac{X_1+X_2+...+X_n}{n}$ est la variable égale à la fréquence d'apparition de l'événement pile, et on a $E(X_1) = p$.

L'intuition qui a guidé le protocole est validée mathématiquement.

> Plus généralement, si on cherche la probabilité d'un événement A lors d'une expérience aléatoire, on va réaliser un grand nombre (à choisir) d'expériences et on va compter le nombre d'expériences où l'événement A se réalise.
> La fréquence d'apparition de l'événement A fournira une valeur approchée de $P(A)$.

Exercice 1:

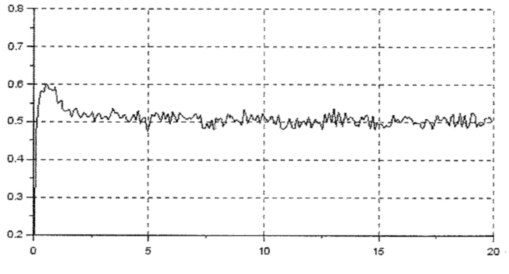
Soit $X \hookrightarrow \mathcal{G}(\frac{1}{3})$.

1. Ecrire une fonction python qui renvoie une simulation de X.
2. Ecrire une fonction python qui prend en entrée un entier N strictement positif, qui simule N réalisations de X, puis qui renvoie la fréquence d'apparition de l'événement $(X \geq 4)$.
3. Exécuter plusieurs fois votre fonction avec $N = 100$. Le résultat varie-t-il beaucoup ?
   Faire de même avec $N = 1000$, $N = 10000$, puis $N = 1000000$, $N = 10000000$. Quel est le bon compromis ?
4. Calculer alors la valeur théorique de $P(X \geq 4)$. Comparer avec les résultats ci-dessus.

Exercice 2:

Soit $a \in \mathbb{R}_+^*$. On considère deux variables aléatoires indépendantes X et Y de même loi de Poisson $\mathcal{P}(a)$.

1. Écrire une fonction d'en-tête `def estime(a):` qui simule un grand nombre de fois les variables X et Y, et renvoie une estimation de $P([X = Y])$.
   Aide python : la syntaxe `np.random.poisson(a)` permet de simuler une variable aléatoire de loi de Poisson de paramètre a, une fois la bibliothèque numpy importée avec l'alias np.
2. Grâce à la fonction précédente, on trace, en fonction de a, une estimation de $\sqrt{\pi a}P([X = Y])$ pour $a \in]0; 20]$.
   À la vue de ce graphe, proposer un équivalent de $P([X = Y])$ lorsque a tend vers $+\infty$.



3. Montrer que $P(X = Y) = e^{-2a} \sum_{k=0}^{+\infty} \frac{a^{2k}}{(k!)^2}$.

Exercice 3:

Deux joueurs effectuent chacun 10 lancers d'une pièce équilibrée de manière indépendante. On notera respectivement, X et Y la variable aléatoire égale au nombre de piles obtenus.

Donner une valeur approchée de la probabilité qu'ils obtiennent le même nombre de pile.

Calcul approché d'une espérance :
On continue d'utiliser la loi faible des grands nombres : avec ce résultat, la visualisation de l'espérance comme d'un nombre moyen (ou d'une moyenne) prend tout son sens.

Exercice 4:
On lance une pièce équilibrée jusqu'à obtenir pour la première fois la configuation "Pile,Pile,Face" dans cet ordre. On note alors X le nombre de lancers effectués.

1. Ecrire une fonction Python sans argument qui simule l'expérience et renvoie le nombre de lancers effectués.
2. A l'aide de python, émettre une conjecture quant à l'existence et la valeur éventuelle de l'espérance de X.

Exercice 5:
Une urne contient au départ une boule blanche et une boule noire. On effectue des tirages d'une boule avec remise jusqu'à l'obtention d'une boule noire, en rajoutant à chaque tirage une boule blanche supplémentaire.
On note X la variable égale au nombre de tirages effectués.

1. Ecrire une fonction python permettant de simuler X.
2. Déterminer la loi de X sur votre feuille afin de vérifier que X n'admet pas d'espérance.
3. Est-ce-possible de le conjecturer avec Python ?
4. A l'aide de python, conjecturer que $\sqrt{X}$ admet une espérance, et donner une valeur approchée de celle-ci.

Exercice 6: *Extrait d'un oral Agro Veto 2023*
On considère une urne contenant n boules numérotées de 1 à n et on effectue n tirages successifs d'une boule avec remise. On note X la variable aléatoire représentant le nombre de numéros distincts obtenus.

1. Écrire une fonction Python d'argument n qui simule l'expérience et renvoie la liste des numéros tirés.
2. Écrire une fonction Python d'argument n qui simule la variable X. On pourra obtenir l'ensemble des valeurs d'une liste L avec la commande `set(L)` et obtenir le cardinal d'un ensemble s avec la commande `len(s)`.
   Bonus : sans cette commande `set`, comment aurait-on pu faire ?
3. Écrire une fonction Python d'argument n qui calcule une valeur approchée de l'espérance de X.

Exercice 7: *Inspiré d'un oral Agro Veto 2021*
Deux amis Anna et Benoît jouent au jeu suivant : ils possèdent une machine qui, à chaque sollicitation, leur donne aléatoirement un entier naturel X selon la loi uniforme $[\![0,8]\!]$.
Si cet entier X est impair, Anna donne X euros à Benoît, on considère que Benoît a gagné.
Si X est nul, on considère que la manche est nulle.
Si X est pair non nul, Benoît donne X euros à Anna, on considère que Anna a gagné.
On pose G le gain algébrique de Anna.

1. Ecrire une fonction permettant de simuler la variable aléatoire G.
2. Ecrire un programme python qui estime la probabilité que Anna gagne, ainsi que la probabilité que Benoit gagne. Le jeu vous semble-t-il équilibré ?
   bonus : Vérifier par le calcul vos résultats.
3. Ecrire alors un programme qui donne une estimation de l'espérance du gain de Anna, ainsi que celle de Benoit. Commentez.
4. *bonus mathématique :* inventer une loi pour X de façon à ce que la probabilité que Anne gagne soit beaucoup plus petite que celle de Benoit, mais telle que l'espérance du gain de Anne soit strictement positive.
   On pourra choisir un $X(\Omega)$ tout petit pour mieux maitriser les paramètres.