CPE Lyon 4ETI 2022-2023

Probabilités discrètes : TP2

Exercice 1 : Simulation des résultats de l'exercice 10 chapitre V du cours de probabilités discrètes

Reprenons l'énoncé de l'exercice 10 du chapitre V du cours de probabilités discrètes :

Soit $n \in \mathbb{N}^*\backslash\{1\}$. Une urne contient n boules numérotées de 1 à n. Un joueur extrait au hasard une poignée de 2 boules. On note X la variable aléatoire égale au plus petit des numéros obtenus.

On a montré que la loi de probabilité de X peut s'écrire :

$$\forall k \in [\![1; n-1]\!], \qquad P(X = k) = \frac{2(n-k)}{n(n-1)}$$

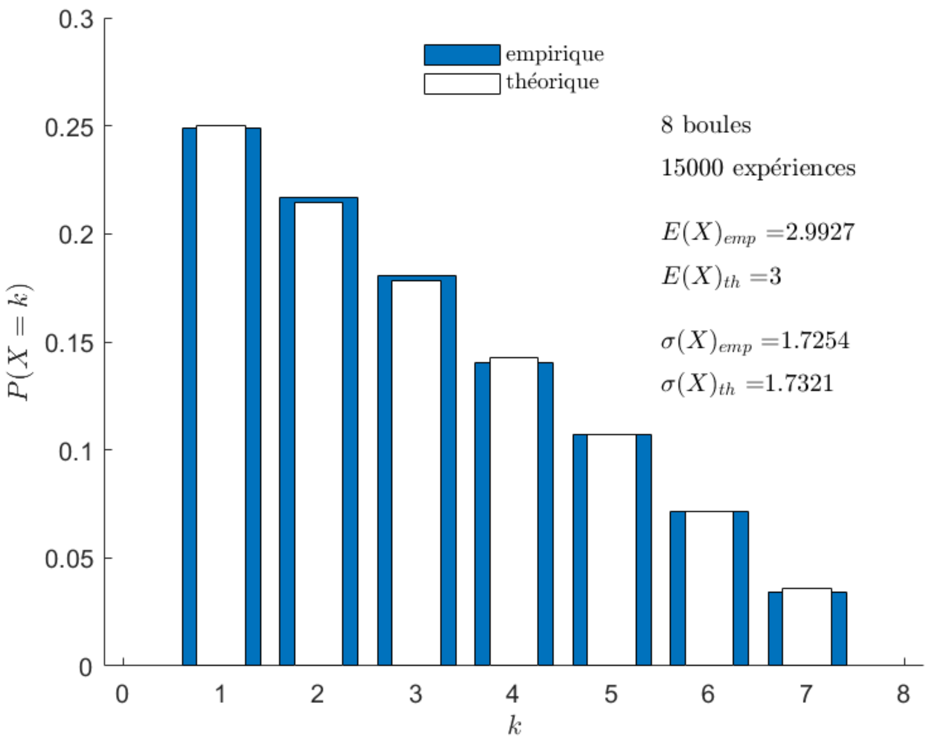
On a également montré que l'espérance mathématique et la variance de X sont :

$$E(X) = \frac{n+1}{3} \qquad \text{et} \qquad V(X) = \frac{n^2 - n - 2}{18}$$

On souhaite faire une simulation de cette loi à l'aide d'un programme Matlab. Ce programme doit permette de :

- simuler la loi de X
- afficher l'histogramme empirique de la loi simulée
- afficher l'histogramme théorique (en superposition de l'histogramme empirique)
- calculer et afficher l'espérance mathématique empirique de X (commande `mean`)
- calculer et afficher l'écart type empirique de X (commande `std`)
- calculer et afficher l'espérance mathématique et l'écart type théoriques de X

Pour $n = 8$ boules, l'exécution du programme doit donner la figure suivante :



Exercice 2 : Simulations des lois de probabilité usuelles discrètes

Objectif : Simuler une variable aléatoire suivant la loi uniforme, la loi de Bernoulli, la loi binomiale et la loi géométrique et comparer avec les formules théoriques.

1) Créer les fonctions suivantes :

▪ Loi uniforme sur l'intervalle $[1, n]$ (`function y=LoiUniforme(n)`)
Cette fonction renvoie un nombre entier aléatoire y compris entre 1 et `n`.

▪ Loi de Bernoulli de paramètre p (`function x=LoiBernoulli(p)`)
Cette fonction doit simuler le comportement d'une variable aléatoire prenant la valeur 1 si l'évènement A de probabilité `p` est réalisé et la valeur 0 sinon. Pour cela, on pourra déterminer au hasard un nombre compris entre 0 et 1 à l'aide de la commande `rand` et affecter 1 à `x` si le nombre obtenu est inférieur à `p`.

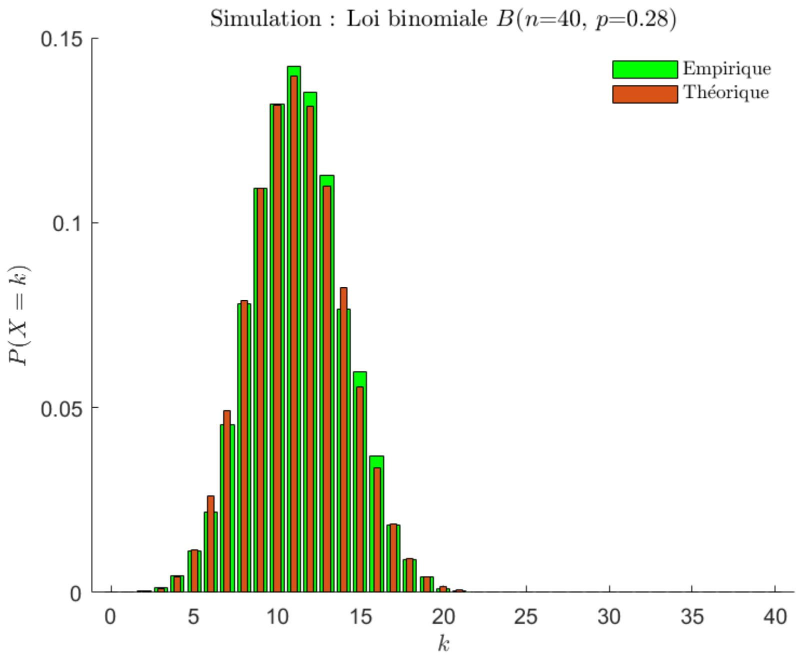
▪ Loi binomiale de paramètres n et p (`function nbSucces=LoiBinomiale(n,p)`)
Cette fonction renvoie le nombre de succès d'un évènement de probabilité `p` répété `n` fois (on utilisera la fonction précédente `LoiBernoulli(p)`).

▪ Loi géométrique de paramètre p (`function n=LoiGeometrique(p)`)
Cette fonction renvoie le nombre d'évènements ayant eu lieu avant que le 1er succès de probabilité `p` ne se produise (on utilisera la fonction `LoiBernoulli(p)`).

2) Tests

On répète un grand nombre de fois l'expérience associée à chacune des lois afin de comparer les fréquences empiriques obtenues avec les fréquences théoriques.
Par exemple, pour tester la loi uniforme sur l'intervalle $[1, n]$, on peut appeler 10000 fois la fonction `LoiUniforme(n)` et faire « compter » les effectifs correspondants à 1, les effectifs correspondants à 2, etc … jusqu'aux effectifs correspondant à n. Pour cela on pourra utiliser la commande Matlab `histogram`.

L'utilisateur doit choisir la loi à tester, et pour chaque loi, il doit pouvoir choisir le ou les paramètre(s) associé(s) à la loi.
Le programme doit afficher sur un même graphique les résultats empiriques et les résultats théoriques (voir figure page suivante) afin de pouvoir faire aisément la comparaison et apprécier si la simulation est correcte. Utiliser la commande `bar` pour afficher les histogrammes des lois théoriques.
Enfin, on affichera les valeurs théoriques et empiriques de l'espérance mathématique et de l'écart type (utiliser les commandes `mean` et `std` pour les valeurs empiriques).


Simulation : Loi binomiale $B(n{=}40,\ p{=}0.28)$
Empirique
Théorique
$P(X=k)$
k
0.15
0.1
0.05
0
0
5
10
15
20
25
30
35
40