TP 3 – Simulations de lois

1 Simulation des lois usuelles

On commencera par importer la bibliothèque `numpy.random`.

Syntaxe 1.

Simuler une fois	*Syntaxe*
$\mathcal{U}(\llbracket a,b\rrbracket)$	`rd.randint(a,b)`
$\mathcal{U}(\llbracket 0,b\llbracket)$	`rd.randint(b)`
$\mathcal{B}(n,p)$	`rd.binomial(n,p)`
$\mathcal{G}(p)$	`rd.geometric(p)`
$\mathcal{P}(\lambda)$	`rd.poisson(lambda)`
$\mathcal{U}([0,1[)$	`rd.random()` ou `rd.rand()`

Syntaxe 2.

N fois	*Syntaxe*
$\mathcal{U}(\llbracket a,b\rrbracket)$	`rd.randint(a,b,N)`
$\mathcal{U}(\llbracket 0,b\llbracket)$	`rd.randint(b,size=N)`
$\mathcal{B}(n,p)$	`rd.binomial(n,p,N)`
$\mathcal{G}(p)$	`rd.geometric(p,N)`
$\mathcal{P}(\lambda)$	`rd.poisson(lambda,N)`
$\mathcal{U}([0,1[)$	`rd.random(N)` ou `rd.rand(N)`

Syntaxe 3.

Matrice $N\times P$	*Syntaxe*
$\mathcal{U}(\llbracket a,b\rrbracket)$	`rd.randint(a,b,(N,P))`
$\mathcal{U}(\llbracket 0,A\llbracket)$	`rd.randint(A,size=(N,P))`
$\mathcal{B}(n,p)$	`rd.binomial(n,p,(N,P))`
$\mathcal{G}(p)$	`rd.geometric(p,(N,P))`
$\mathcal{P}(\lambda)$	`rd.poisson(lambda,(N,P))`
$\mathcal{U}([0,1[)$	`rd.random((N,P))` ou `rd.rand(N,P)`

2 Simulation de lois par `random`

Proposition 1. *On peut simuler la loi uniforme sur $\llbracket a,b\rrbracket$ par la commande suivante.*

```
a + np.floor((b-a+1)*rd.random())
```

Proposition 2. *On peut simuler la loi de Bernoulli $\mathcal{B}(p)$ par la commande suivante.*

```
X = 0
if(rd.random()<p):
    X=1
print(X)
```

Proposition 3. *On peut simuler la loi binomiale $\mathcal{B}(n,p)$ par les commandes suivantes.*

```
X = 0
for k in range(n):
    if rd.random()<p:
        X+=1
print(X)
```

Proposition 4. *On peut simuler la loi géométrique $\mathcal{G}(p)$ par les commandes suivantes.*

```
X=1
while rd.random()>p:
    X+=1
print(X)
```

3 Histogrammes

Syntaxe 4.

Pour tracer l'histogramme analysant la liste de valeurs `valeurs` avec n intervalles de tailles égales :

```
plt.hist(valeurs, n)
plt.show()
```

Pour tracer l'histogramme analysant la liste de valeurs `valeurs` avec les intervalles dont les extrémités sont dans `interv` :

```
plt.hist(valeurs, interv)
plt.show()
```

Exemple 5.

```
val = np.random.randint(1,7,1000)
# simulation de 1000 lancers de dés
inter = np.linspace(0.5,6.5,7)
plt.hist(val,inter)
plt.show()
```

Remarque 6.

Quand on va dans l'aide sur `hist`, on peut trouver tous les paramètres de cette fonction.
On peut noter entre autres `color`, `rwidth`, `cumulative` ou `label`.

4 Comparaison expérimentale de lois

Loi binomiale et loi de Poisson

Proposition 7 (*Rappel*).
Soient $\lambda\in\mathbb{R}_+^*$ et $(X_n)_{n\in\mathbb{N}^*}$ une suite de variables aléatoires suivant la loi binomiale $\mathcal{B}(n,\frac{\lambda}{n})$. Alors

$$\forall k\in\mathbb{N},\quad \lim_{n\to+\infty}\mathbb{P}(X_n=k)=\mathbb{P}(X=k),$$

où X suit une loi de Poisson de paramètre λ.

Exemple 8.

Tester les commandes suivantes pour différentes valeurs de n et p :

```
n, p, N = 1000, 0.1, 10000
Bi = rd.binomial(n,p,N)
Poi = rd.poisson(n*p,N)
inter = np.linspace(0,n/2)
plt.hist([Bi,Poi],inter,label=['Bi','Poi'])
plt.legend()
plt.show()
```

5 Exercices

Exercice 1.

1. Simuler 1 000 lancers d'un dé équilibré à six faces.
2. Pour cette simulation, déterminer le nombre de fois où le dé est tombé sur chacune des faces, et la fréquence associée.
3. Tracer l'histogramme associé.

Exercice 2.

1. Écrire une fonction qui simule le nombre de lancers nécessaires à l'obtention d'un six lors du lancer d'un dé équilibré, sans utiliser `geom`.
2. Faire 10 000 simulations. Combien de lancers sont nécessaires en moyenne ?

Exercice 3.

Une expérience consiste à effectuer 100 lancers d'une pièce telle que la probabilité d'obtenir Pile vaut 0,3.
On note X la variable aléatoire égale au nombre de piles obtenus.

1. Créer une fonction `X(n)` prenant en entrée un entier n qui réalise n fois cette expérience et renvoie un tableau contenant les n résultats.
2. Tracer l'histogramme des résultats observés lors de la réalisation de 1 000 expériences.

Exercice 4.

Une puce se déplace sur un axe gradué en partant de l'origine. A chaque saut, elle se déplace d'une unité, de manière aléatoire et équiprobable vers la gauche ou vers la droite.

1. Écrire une fonction `puce` d'argument N et donnant la position de la puce au bout de N sauts.
2. On effectue 1 000 fois l'expérience durant laquelle la puce saute 200 fois. Tracer un histogramme correspondant aux différentes positions de la puce pour $N = 200$ sauts.
3. On s'intéresse au numéro du saut donnant le premier retour à l'origine de la puce. Écrire une fonction `PremRetour()` sans argument et donnant le numéro de ce saut.
4. Donner une estimation du nombre moyen de sauts nécessaires pour un premier retour à l'origine. Renouveler plusieurs fois cette opération. Que remarque-t-on ?

Exercice 5.

1. Écrire une suite d'instructions pour approcher la probabilité que dans une classe de 42 élèves, deux élèves au moins soient nés le même jour, en faisant 1 000 simulations. On ne comptera pas les années bissextiles.
2. Écrire une fonction `anniv` d'argument `n` et donnant la fréquence de l'événement dans une classe de n élèves, pour 1 000 simulations.
3. À partir de combien d'élèves cette fréquence est-elle supérieure à 0,5 ?

Exercice 6.

On considère une succession de n lancers d'une pièce équilibrée, et on note X la variable aléatoire égale au nombre de fois où l'on obtient deux tirages consécutifs identiques.

1. Écrire une fonction d'argument n qui réalise la variable aléatoire X.
2. Faire 1000 simulations, et tracer un histogramme des valeurs de X pour $n = 30$.
3. Calculer une estimation de $\mathbb{E}(X)$ et de $\mathbb{V}(X)$.

Exercice 7.

Une urne contient r boules rouges et b boules bleues. On tire successivement et sans remise n boules de l'urne, avec $1 \leqslant n \leqslant r + b$, et on note X le nombre de boules rouges obtenues.

1. Écrire une fonction `urne` qui a pour arguments r, b et n, et qui réalise la variable aléatoire X.
2. On se place dans la situation où $(r, b, n) = (7, 13, 9)$.
   Estimer l'espérance et la variance de X.

Exercice 8.

On lance indéfiniment une pièce telle que la probabilité d'obtenir pile vaut $p \in]0, 1[$.
On note L_1 la longueur de la première série, et L_2 la longueur de la deuxième.
Par exemple, si les lancers ont donné $PPPFFFFPFF$, alors $L_1 = 3$ et $L_2 = 4$.

1. Écrire une fonction `series` d'argument p, qui donne les longueurs des deux premières séries.
2. Estimer les espérances de L_1 et L_2.

Exercice 9.

On lance indéfiniment une pièce telle que la probabilité d'obtenir pile vaut $p \in]0, 1[$.

1. Écrire une fonction `attente` d'arguments p et k, qui donne le nombre de lancers nécessaires pour obtenir le k-ème pile.
2. On réalise cette expérience 10000 fois. Estimer le nombre moyen de lancers nécessaires pour obtenir le k-ème pile.

Exercice 10.

On lance deux dés équilibrés à six faces.

1. Écrire une fonction `des` d'argument n, et renvoyant une matrice à deux colonnes simulant les réalisations de n lancers de ces deux dés.
2. Construire le tableau des sommes de ces deux dés pour chacun des n lancers.
3. Construire et afficher le tableau des fréquences observées de ces sommes.
4. Déterminer et afficher la somme la plus fréquente.

Exercice 11.

1. Soit X une variable aléatoire suivant la loi binomiale de paramètres n et p.
   Créer une fonction `simubin` de paramètres n, p et k, simulant la probabilité $\mathbb{P}(X = k)$.
2. Soit Y une variable aléatoire suivant la loi de Poisson de paramètre λ.
   Créer une fonction `simupoi` de paramètres λ et k, simulant la probabilité $\mathbb{P}(Y = k)$.
3. Créer une fonction `compa` d'arguments n et p, donnant en sortie une matrice avec 2 lignes et $n + 1$ colonnes : la première ligne donne les $\mathbb{P}(X = k)$ pour $k \in [\![0, n]\!]$, et la seconde les $\mathbb{P}(Y = k)$ pour $k \in [\![0, n]\!]$, où $\lambda = np$.
4. Tester la fonction `compa` pour $(n, p) \in \{(30; 0{,}4), (30; 0{,}2), (30; 0{,}1), (30; 0{,}05)\}$.
   Penser à faire des tracés. Que conclure ?
5. Tester la fonction `compa` pour $(n, p) \in \{(10; 0{,}1), (20; 0{,}1), (30; 0{,}1), (50; 0{,}1)\}$.
   Penser à faire des tracés. Que conclure ?