Univ. de Lille Département de Mathématiques M.Fradon, G.Castellan, M-F.Barme
L2 Maths S4 **Probabilités Discrètes** 2020-2021

Fiche 6 sur ordinateurs

Avertissement : Ceci est un TD de maths. Le but n'est pas d'apprendre Python auprès des matheux ! Vous avez eu droit à trois semestres avec des informaticiens infiniment plus qualifiés pour l'enseigner. Le but est de mettre en pratique vos connaissances sur les variables aléatoires discrètes, et d'approfondir votre compréhension des lois et des théorèmes limites.

Ex 1. Au commencement était la loi uniforme...

1) Lancer Python. Charger la librairie `numpy` en tapant `from numpy.random import *`. Effectuer des tirages avec `random()`. Que fait cette commande ?

2) Effectuer des tirages de `random()<0.3`. Que fait cette commande ?

3) Effectuer des tirages de `int( random()<0.3 )`. Que fait cette commande ? Quel est le nom mathématique de la fonction qu'on a rajoutée ici (transformation par `int` des booléens) ? Quelle est le nom de la loi de probabilité selon laquelle on tire ici ?

4) Effectuer des tirages de `[int(random()<0.5) for i in range(3)]`. on pourra remplacer 3 et 0.5 par les valeurs de son choix. Que fait cette commande ?

5) Effectuer des tirages de `sum([ int(random()<0.5) for i in range(3) ])`. Quelle est le nom de la loi de probabilité selon laquelle on tire ici ?

6) Ce semestre, on a mentionné des variables aléatoires comme *le nombre de fois où un dé fait* 6 *en* 10 *lancers* ou *le nombre de piles obtenus en* 1000 *tirages d'une pièce équilibrée.* Simuler de telles v.a. Constate-t-on empiriquement que le nombre de piles en 1000 lancers a une forte probabilité d'être autour de 500 mais une faible probabilté d'être égal à 500 ? (prouvé en exercice dans la fiche 1).

Que signifient les mots *empirique* et *empiriquement* ?

Ex 2. Ensuite vinrent... beaucoup d'autres lois (mais pas toutes en L2)

On a vu qu'à partir de tirages uniformes sur l'intervalle $[0;1[$ finement discrétisé, on peut faire un tirage d'une Bernoulli, d'une binomiale, etc. En fait *toutes* les lois peuvent être simulées à partir de l'uniforme sur $[0,1[$, d'après un théorème qui n'est pas au programme de L2. A partir de la fonction `random()`, Python peut donc simuler des tirages de toutes les lois classiques. Elles sont préprogrammées.

1) Effectuer des tirages de `binomial(3, 0.5)` et `binomial(3, 0.5, 10)` en variant les paramètres. Que représentent les trois paramètres de cette fonction ? Les résultats sont-ils conformes à ce que vous savez de cette loi, notamment la valeur de son espérance ?

2) Même question pour `poisson(5,10)`.

3) Même question pour `geometric(0.5,10)`.

4) Même question pour `hypergeometric(100,200,9,20)` (attention à la signification des paramètres !).

5) Il n'existe pas de commande `bernoulli()`. Pourquoi ? Par quelle commande fait-on un tirage d'une Bernoulli ?

Ex 3. Histogrammes

L'*histogramme* d'une loi de probabilité discrète est un graphique représentant la loi sous forme de colonnes verticales. Pour dessiner la loi P_X de la v.a. discrète X, on place une colonne de hauteur $P(X = x_k)$ à chaque abcisse $x_k \in X\Omega)$ des valeurs que X peut prendre.

1) Dessiner sur papier les histogrammes des lois $\mathcal{B}er(\frac{3}{10})$ et $\mathcal{B}in(3, \frac{1}{2})$.

2) Importer les commandes graphiques en tapant `import matplotlib.pyplot as plt`. La commande `plt.bar([0,1,2], [0.7, 0.1, 0.2])` génère un graphique. Le visualiser en tapant `plt.show()`. Faire afficher successivement les histogrammes des lois $\mathcal{B}er(\frac{3}{10})$ et $\mathcal{B}in(3, \frac{1}{2})$. On pourra effacer l'écran graphique en tapant `plt.clf()`, ou varier les écrans en changeant le paramètre de `plt.show()`.

3) Quand on fait n tirages indépendants d'une v.a. X on obtient un *n-échantillon* X_1, X_2,...,X_n de la loi de X. Tirer un 10-échantillon de la loi $\mathcal{B}er(\frac{3}{10})$ qu'on nommera `tirage`. On peut le lire avec `list(tirage)`.

4) L'*histogramme empirique* d'un n-échantillon est la représentation de colonnes de hauteur égale à la proportion de valeurs égales à x_k dans l'échantillon :

$$\text{Pour chaque } x_k \in X(\Omega) \quad \text{colonne de hauteur } \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{X_i = x_k} \text{ à l'abcisse } x_k$$

Que fait la commande `plt.hist(tirage)` ? Visualiser ce graphique pour différents tirages.

5) Contrairement à l'histogramme tout court (histogramme théorique) l'histogramme empirique est aléatoire. Faire tourner plusieurs fois le programme pour $n = 10$ et constater les variations. Que se passe-t-il quand $n = 100$? Quand $n = 1\,000$? $n = 10\,000$? 100 000 ? 1 000 000 ?

6) Ce que Python affiche par défaut comme histogramme correspond-il à la définition mathématique ci-dessus ? Pour obtenir des colonnes de largeur 1 centrées en les x_k et de hauteur normalisée, on pourra faire `import numpy as np` et rajouter `bins=np.linspace(`a`,`b`,`nb`, density=True)` pour fixer la borne min, la borne max et le nombre total de bornes des colonnes :

```
plt.hist(tirage, bins=np.linspace(-0.5,1.5,3), density=True)
```

7) Comparer l'histogramme empirique de chaque échantillon à l'histogramme de la loi correspondante, pour différentes tailles d'échantillon. Que constate-t-on ? Prouver qu'il s'agit d'une propriété générale [1]

Ex 4. Convergences

On a vu que les histogrammes empiriques d'un grand nombre de tirages indépendants d'une même loi donnent une représentation approximative (et asymptotiquement exacte) de cette loi.

1) Représenter ainsi la loi $\mathcal{P}ois(5)$.

2) Représenter aussi les lois $\mathcal{B}in(10, \frac{1}{2})$, $\mathcal{B}in(100, \frac{1}{20})$, $\mathcal{B}in(1\,000, \frac{1}{200})$ et $\mathcal{B}in(10\,000, \frac{1}{2\,000})$. Quelle propriété prouvée en cours explique le phénomène constaté ici ?

3) Même quetion concernant les lois $\mathcal{H}ypergeom(30, 10, 9)$, $\mathcal{H}ypergeom(300, 100, 9)$, $\mathcal{H}ypergeom(3\,000, 1\,000, 9)$ et $\mathcal{H}ypergeom(30\,000, 10\,000, 9)$. Quelle propriété démontrée et quelle loi sous-jacente apparaissent ici ?

1. Question à faire plus tard si la loi faible des grands nombres n'a pas encore été vue.

Ex 5. Moyenne empirique

La *moyenne empirique* d'un n-échantillon de la loi P_X d'une v.a. X est la moyenne des valeurs aléatoires obtenues par n tirages répétés indépendants de cette loi :

Moyenne empirique $\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$ avec $X_1, X_2, \ldots, X_n$ indépendantes toutes de même loi que X

1) Rappeler ce qui a été vu en cours sur la moyenne empirique.

2) La commande `mean` de la librairie `numpy` calcule la moyenne empirique d'un échantillon. Calculer plusieurs fois la moyenne empirique d'un 10-échantillon de la binomiale de votre choix. Que constate-t-on ? Faire de même pour des échantillons de plus grande taille. Le phénomène constaté est-il conforme à ce qu'on attend ?

3) Même question pour des échantillons de taille croissante de la loi de Poisson de votre choix, puis de la loi géométrique de votre choix.

4) On a deux v.a. indépendantes $X \sim \mathcal{G}eom(\frac{1}{5})$ et $Y \sim \mathcal{G}eom(\frac{1}{4})$. Trouver informatiquement une valeur approximative de l'espérance des v.a. $\min(X, Y)$ et $\max(X, Y)$.

Indication : utiliser les commandes `min` et `max` de `numpy` sur deux échantillons de X et Y. Que se passe-t-il si on passe en second paramètre 0 ou 1 à ces commandes ?

Ex 6. Marche aléatoire

Une particule se trouve à l'instant 0 à la position X_0. A l'instant 1 elle saute au hasard vers la droite (position $X_1 = X_0 + 1$) avec probabilité α ou vers la gauche (position $X_1 = X_0 + 1$) avec probabilité $1 - \alpha$. Elle recommence à l'instant suivant, indépendamment de ce qui précède, et continue ainsi indéfiniment. On suppose qu'elle se déplace dans $\{-n, -n+1, -n+2, \ldots, n-1, n\}$. Si elle doit sauter à droite quand elle est en n elle reste sur place. De même si elle doit sauter vers la gauche depuis $-n$.

Selon ce schéma, elle occupe des positions aléatoires successives $X_0, X_1, X_2, X_3, X_4, \ldots$ qui forment une suite de v.a. indexée par le temps $X = (X_k)_{k\in\mathbb{N}}$. La position dépend du hasard et du temps. Un tel objet X (fonction du hasard et du temps) s'appelle un *processus stochastique*, et dans le cas où chaque transition vers l'état suivant est indépendante des transitions précédentes, on parle de *chaine de Markov*.

Le programme ci-dessous simule sous forme d'histogramme l'évolution d'une (ou plusieurs) particule(s) le long de la chaine de Markov. Vous pouvez le recopier (sans les nombreux commentaires) ou vous en inspirer.

```
### Marche aléatoire sur un intervalle (D.Redon, fév 2021)

from numpy.random import *
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.animation as animation

N_ETATS = 10 # mouvement sur {-N_ETATS,...,N_ETATS}
N_PARTS = 1 # nombre de particules indépendantes en mouvement

etats = [0 for _ in range(N_PARTS)] # position initiale des particules
```

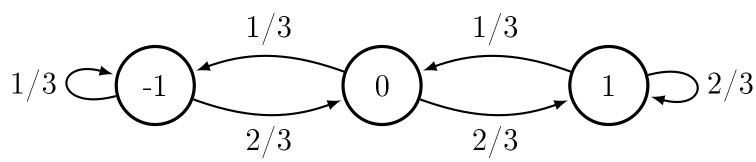
```
def suivant(etat): # calcul de la position suivante de chaque particule
    return min( max( -N_ETATS, etat + choice((-1,1),p=(1/3,2/3)) ), N_ETATS )

def dessine(tour):
    global etats
    plt.clf() # efface le graphique précédent
    # plt.gca().set_ylim((0,N_PARTS/N_ETATS*2)) # fixe l'axe vertical de l'histogramme
    plt.hist(etats, bins=np.linspace(-N_ETATS-0.5,N_ETATS+0.5,2*N_ETATS+2)) # histogramme
    etats = [suivant(e) for e in etats] # tirage de la position suivante des particules

anim = animation.FuncAnimation(plt.gcf(), dessine, interval=1, repeat_delay=500)
plt.show()
```

1) Que constate-t-on pour une particule ? Pour un grand nombre de particules indépendantes ? Le comportement est-il le même pour une marche aléatoire symétrique ($\alpha = 1/2$) ou non symétrique ($\alpha \neq 1/2$) ? En quoi dépend-il de la position initiale X_0 ?

2) Dans le cas où $n = 1$ et $\alpha = 2/3$, la *matrice de transition* et le *graphe de transition* de la marche aléatoire sont



$$A = \begin{pmatrix} 1/3 & 2/3 & 0 \\ 1/3 & 0 & 2/3 \\ 0 & 1/3 & 2/3 \end{pmatrix}$$

Quel lien y a-t-il entre eux ? Si $\mu_k = (P(X_k = -1),\ P(X_k = 0),\ P(X_k = 1))$ alors $\mu_{k+1} = \mu_k A$ pour chaque k de $\mathbb{N}$: justifier cette affimation par une formule du cours.

3) Pour un grand nombre de particules, la situation en temps long se stabilise, tout en gardant des fluctuations aléatoires autour d'une répartition qui dépend visiblement de α. Comparer cette répartition à ce qu'on obtient en résolvant l'équation $\mu = \mu A$ pour $\mu = (a, b, c)$ inconnue.

4) Justifier la formule $\mu_k = \mu_0 A^k$. Calculer la limite de la suite des puissances de A. En déduire la limite de la suite des μ_k. En quoi dépend-elle de μ_0 ? Conclusion ?

Sous des hypothèses assez larges, les chaines de Markov ont des propriétés analogues à celles prouvées ici.