1 Introduction

On souhaite simuler en `python` des variables aléatoires à densité.

Soit $X : \Omega \longrightarrow \mathbb{R}$ une variable aléatoire à densité, et f_X une densité de probabilité pour X. En notant F_X la fonction de répartition X, alors pour tout $x \in \mathbb{R}$:

$$F_X(x) = P(X \leqslant x)$$

$$f_X(x) = F_X'(x) \quad \text{et} \quad \int_{-\infty}^{+\infty} f_X(s)ds = 1$$

$$\Longrightarrow \; P(X \leqslant x) = \int_{-\infty}^{x} f_X(s)ds$$

En particulier, pour tout couple $(a, b) \in \mathbb{R}^2$:

$$P(a \leqslant X \leqslant b) = P(a < X \leqslant b) = P(a \leqslant X < b) = P(a < X < b) = \int_a^b f_X(s)ds$$

2 Générateur pseudo aléatoire `random()` et simulation de loi uniforme

La fonction `random()` du module `random` génère un nombre (de type `float`) pseudo-aléatoire dans l'intervalle $[0,1[$ selon une loi uniforme. Soit la variable aléatoire :

$$X \hookrightarrow \mathcal{U}([0,1])$$

Il est facile à l'aide de `random()` de simuler le résultat de $X \hookrightarrow \mathcal{U}([0,1])$ pour une expérience aléatoire :

```
from random import random
def varUni():
    return random()
```

2.1 Simulation du résultat de N expériences aléatoires

Simulons la fréquence d'apparition de chaque résultat sur N expériences aléatoires.

Pour cela on discrétise l'intervalle $[0,1]$ par une subdivision régulière en $n+1$ points, c'est à dire que donné le nombre n d'intervalles, on considère la subdivision régulière de l'intervalle $[0,1]$:

$$(x_k)_{0 \leqslant k \leqslant n} \quad \text{,définie par :} \quad \forall\, k \in [[0,n]],\; x_k = \frac{k}{n}$$

On considère la liste de longueur n :

```
FreqX = [0] * n
```

et `X[i]` contiendra à l'issue de la simulation le nombre de fois où l'expérience a réalisé l'évènement $(x_i \leqslant X < x_{i+1})$.

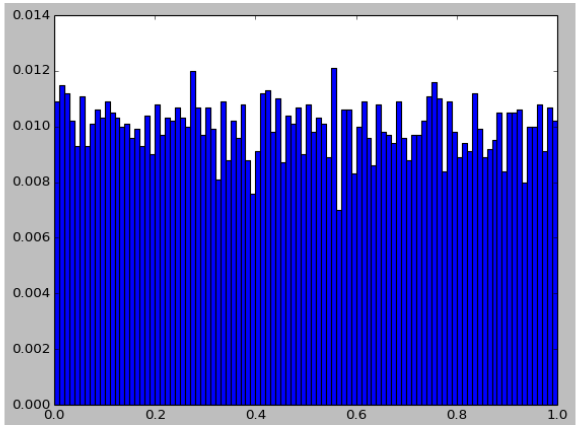
Pour que la simulation soit fidèle il faut prendre $n \leqslant N$!

• Code pour la simulation :

```
def simulUni(N,n):
    FreqX = [0] * n
    for k in range(N):
        x = varUni()
        i = int(x * n)
        FreqX[i] += 1
    return [x/N for x in FreqX]
```

• Et son tracé de l'histogramme :

```
import numpy as np
import matplotlib.pyplot as plt
N = 10000
n = 100
I = np.arange(0,1,1/n)    # liste des x_k
FreqX = simulUni(N, n)
plt.clf()
plt.bar(I,FreqX,width = 1/n)
plt.show()
```

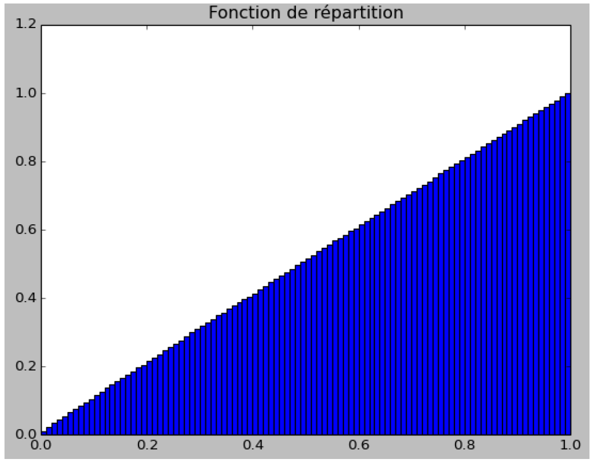


2.2 Simulation et tracé de la fonction de répartition et de la densité

• A partir de la simulation réalisée il est facile de simuler et tracer la fonction de répartition : $F_X(x) = P(X \leqslant x) \approx Freq(X \leqslant x)$.

• Code pour la simulation de la fonction de répartition :

```
Fx = [0] * n
for k in range(n-1):
    Fx[k+1] = Fx[k] + FreqX[k]
plt.figure(2)
plt.title("Fonction de répartition")
plt.bar(I,Fx,width = 1/n)
plt.show()
```

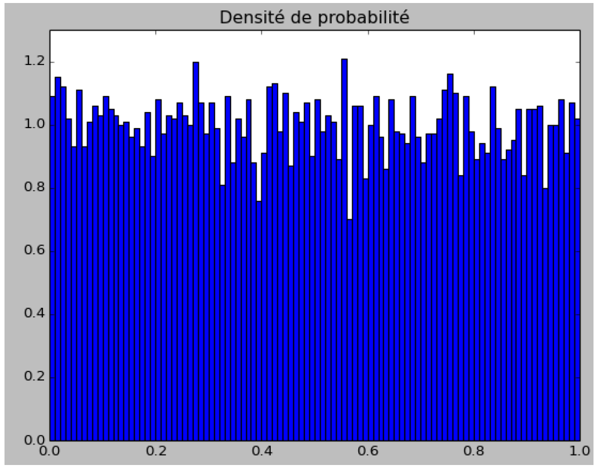


• Pour le tracé de la fonction de densité f_X :

$$f_X(x) = F'_X(x) \implies f_X(x_k) \approx \frac{F_X(x_{k+1}) - F_X(x_k)}{x_{k+1} - x_k} \approx \frac{P(x_k \leqslant X < x_{k+1})}{x_{k+1} - x_k}$$

Puisque pour tout k, $x_{k+1} - x_k = (b-a)/n = 1/n$ (ici !), on en déduit pour notre simulation, en stockant les valeurs prises par f_X aux points $(x_k)_k$ dans une liste `fx` :

```
fx[k] = FreqX[k] * n/(b-a) = FreqX[k] *n
```



Exercice 1

Refaire la simulation pour une variable aléatoire X suivant une loi uniforme sur l'intervalle $[a, b]$:

$$X \hookrightarrow \mathcal{U}([a, b])$$

On adaptera tout le code précédent à ce contexte et l'on effectuera la simulation avec les valeurs $a = 1$, $b = 2.5$.

3 Simulation de variables aléatoires continues non uniformes

On souhaite simuler une VAR continue X de fonction de répartition F_X strictement croissante sur son support.

3.1 Méthode de la fonction inverse

Theorème. *Si X a une fonction de répartition F_X strictement croissante sur son support $S = \{x \in \mathbb{R} \mid F_X(x) > 0\}$, alors la VAR $F_X \circ X$ suit la loi uniforme sur $[0, 1]$:*

$$F_X \circ X \hookrightarrow \mathcal{U}([0, 1])$$

Preuve. Puisque F_X est strictement croissante elle réalise une bijection de son support S sur son image directe $I = F_X(\mathbb{R})$; $]0, 1[\subset I \subset [0, 1]$. Soit $y \in I$ et $x \in S$ tel que $y = F_X(x)$; donc $x = F_X^{-1}(y)$.

Par définition $F_X(x) = P(X \leqslant x)$, et donc :

$$\begin{aligned} & F_X(F_X^{-1}(y)) = P(X \leqslant F_X^{-1}(y)) \\ \Longrightarrow \quad & y = P(F_X \circ X \leqslant y) \end{aligned}$$

car F_X étant strictement croissante les évènement $(X \leqslant F_X^{-1}(y))$ et $(F_X \circ X \leqslant y)$ sont équivalents. En effet, pour tout $\omega \in \Omega$:

$$X(\omega) \leqslant F_X^{-1}(y) \quad \text{si et seulement si } F_X \circ X(\omega) \leqslant F_X \circ F_X^{-1}(y) = y$$

Ainsi pour tout $y \in I$: $P(F_X \circ X \leqslant y) = y$.

De plus puisque $]0,1[\subset I \subset [0,1]$, par passage à la limite, à droite en 0 : $P(F_X \circ X \leqslant 0) = 0$ et à gauche en 1 : $P(F_X \circ X \leqslant 1) = 1$ (rappelons qu'une fonction de répartition étant croissante, elle admet en tout point une limite à droite ainsi qu'une limite à gauche). Ainsi pour tout $y \in [0,1]$: $P(F_X \circ X \leqslant y) = y$.

On reconnait la fonction de répartition de la loi uniforme sur $[0,1]$. □

Application : Méthode de la fonction inverse.

Pour simuler une variable aléatoire continue X ayant un fonction de répartition F_X strictement croissante sur son support $S = \{x \in \mathbb{R} \mid F_X(x) > 0\}$, il suffit de :

1. Déterminer la fonction réciproque F_X^{-1} de F_X restreinte à S.
2. Simuler une loi uniforme $U \hookrightarrow \mathcal{U}([0,1])$ et obtenir X par composition :

$$X = F_X^{-1}(U)$$

3.2 Simulation de la loi exponentielle

Soit $\mathcal{E}(\lambda)$ la loi exponentielle de paramètre $\lambda > 0$. On rappelle que sa densité de probabilité est donnée par :

$$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ \lambda e^{-\lambda x} & \text{si } x \geqslant 0 \end{cases}$$

et sa fonction de répartition est :

$$F(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 - e^{-\lambda x} & \text{si } x \geqslant 0 \end{cases}$$

Elle est strictement croissante sur $\mathbb{R}^+$ et pour tout $x \geqslant 0$:

$$F(x) = 1 - e^{-\lambda x} \quad \Longleftrightarrow \quad F^{-1}(y) = -\frac{1}{\lambda}\ln(1-y)$$

Exercice 2

1. Ecrire une fonction `varExp(param)` qui retourne le résultat d'un VAR X suivant la loi exponentielle de paramètre `l` pour la simulation d'une expérience aléatoire.
2. Ecrire une fonction `simulExp(N,n,param)` sur le modèle du paragraphe 2.1, qui retourne la fréquence des résultats de X pour N simulations d'expériences aléatoires. On ne tiendra compte que des résultats de X strictement inférieur à `Max = 15` (on pourra définir la variable globale `Max`).
3. Dans la suite on prendra `param = 0.5`, `N= 10000` et `n=100`. Effectuer le tracé de l'histogramme des fréquences obtenues pour cette simulation.
4. Tracer pour cette simulation les histogrammes de la fonction de répartition et de la densité de probabilité.
5. Sur les mêmes graphiques tracer les courbes des solutions théoriques de la fonction de répartition et de la densité de probabilité.