

FACULTÉ DES SCIENCES DE SFAX (FSS)

DÉPARTEMENT MATHÉMATIQUE

Methode de Monte-Carlo

Etudiante :
GHORBEL HANA

Encadrant :
Msr.MASMOUDI AFIF

17 juin 2021

Table des matières

Méthodes de Monte-Carlo	3
0.1 Notations Mathématiques	3
0.2 Introduction	5
0.3 Simulation des variables aléatoires	6
0.3.1 Idée générale	6
0.3.2 Simulation suivant la loi uniforme	6
0.3.3 Simulation par inversion de la fonction de répartition	7
0.3.3.0.1 Exemple 1	8
0.3.3.0.2 Exemple 2	10
0.4 Performance de la méthode de Monte-Carlo	12
0.4.1 Description de la méthode de Monte-Carlo	12
0.4.2 Convergence	12
0.4.3 Validité et comportement de la méthode	15
0.4.4 Intervalle de confiance	18

0.4.5	Vitesse de convergence et estimation de l'erreur	19
-------	--	----

Méthodes de Monte-Carlo

0.1 Notations Mathématiques

X, Y : Vecteurs aléatoires

X_1, \dots, X_n : Échantillon (aléatoire)

x_1, \dots, x_n : Échantillon (observé)

μ : Espérance d'une variable aléatoire

σ^2 : Variance d'une variable aléatoire

$E(X)$: Espérance théorique de la variable aléatoire X

$Var(X)$: Variance théorique de la variable aléatoire X

$N(0, 1)$: Loi Gaussienne centrée réduite

$N(\mu, \sigma^2)$: Loi Gaussienne de paramètre μ et σ^2

$U(a, b)$: Loi Uniforme sur l'intervalle $[a, b]$

$Bin(n, p)$: Loi Binomiale de paramètres n et p

$\varepsilon(\lambda)$: Loi Exponentielle de paramètre λ

$P(\lambda)$: Loi de Poisson de paramètre λ

$\tau(n)$: Loi de Student à n degrés de liberté

$\chi^2(n)$: Loi de Khi-deux à n degrés de liberté

$f_X(.)$: Fonction densité de la variable aléatoire X

$F_X(.)$: Fonction de répartition de la variable aléatoire X

F_X^{-1} : Fonction de répartition réciproque de la variable aléatoire X

\bar{X}_n : Moyenne Empirique $\frac{1}{n} \sum_{i=0}^n X_i$

\bar{x}_n : Réalisation de la moyenne empirique $\frac{1}{n} \sum_{i=0}^n X_i$

$\xrightarrow[n \rightarrow +\infty]{Loi}$: Symbole de convergence en Loi

$\xrightarrow[n \rightarrow +\infty]{P.S}$: Symbole de convergence Presque sûrement

$\xrightarrow[n \rightarrow +\infty]{P}$: Symbole de convergence en Probabilité

$\hat{F}_n(.)$: Fonction de répartition empirique

θ : Paramètre inconnu

$\hat{\theta}(X_1...X_n)$: Estimateur du paramètre inconnu θ basé sur l'échantillon $X_1...X_n$

$\hat{\theta}(x_1...x_n)$: Estimation du paramètre inconnu θ basé sur l'échantillon observé $x_1...x_n$

$IC_{1-\alpha}(\theta)$: Intervalle de confiance (aléatoire), de niveau de confiance $1 - \alpha$, pour θ

$1 - \alpha$: Niveau de confiance d'un intervalle de confiance

i.i.d : Indépendantes et identiquement distribuée.

$1_{[a,b]}$: Vaut 1 si x appartient à $[a, b]$, 0 sinon

0.2 Introduction

De nombreuses approches peuvent être mises en œuvre pour exploiter un modèle mathématique d’une situation réelle et étudier ses propriétés. Il est parfois possible de calculer explicitement les quantités auxquelles on s’intéresse au moyen de formules fermées, ou tout au moins de calculer celles-ci numériquement, de manière exacte si l’on exclut les questions de précision finie des calculs sur ordinateur, pour des valeurs fixées des paramètres.

La simulation, qui consiste à reproduire artificiellement le fonctionnement du modèle étudié, constitue l’une des approches importantes permettant d’exploiter celui-ci. Elle permet notamment de valider ou d’invalidier des hypothèses, d’obtenir des informations quantitatives (qui peuvent venir affiner des informations qualitatives si l’on en possède), de valider certaines approximations, d’évaluer la sensibilité d’un modèle à certaines hypothèses ou à certains paramètres, ou tout simplement d’explorer le comportement d’un modèle lorsque celui-ci est mal connu ou mal compris.

La simulation de modèles stochastiques nécessite le recours à des nombres pris au hasard, et est connue sous le nom générique de méthode de Monte-Carlo (par référence aux jeux de hasard des casinos). De nombreux problèmes numériques a priori sans rapport aucun avec le hasard ou les phénomènes aléatoires (évaluation d’intégrales, résolution de systèmes linéaires ou d’équations aux dérivées partielles) peuvent cependant, de manière plus ou moins artificielle, être traduits en termes de modèles stochastiques, et la portée des méthodes de Monte-Carlo dépasse donc très largement le cadre de la modélisation de phénomènes aléatoires.

0.3 Simulation des variables aléatoires

0.3.1 Idée générale

Soit X une variable aléatoire de fonction de répartition F , telle que, $F(x) = P(X \leq x)$.

On simule (génère) une réalisation u de la loi uniforme sur $[0, 1]$:

$$u \sim U[0, 1].$$

Dans la suite, on va calculer $F^{-1}(u) = x$ une réalisation de la variable aléatoire X .

Puis, on répète cette procédure n fois on obtient un échantillon x_1, \dots, x_n qui sont les réalisations d'un échantillon aléatoire X_1, \dots, X_n .

Cette méthode s'appelle la méthode d'inversion.

Il y a d'autres méthodes de simulation telles que la méthode d'acceptation-rejet.

0.3.2 Simulation suivant la loi uniforme

En pratique, l'ordinateur ne sait engendrer que des suites de nombres déterministes, il est incapable de générer une suite réellement aléatoire. Par contre, il est possible de construire des suites de nombres qui se comportent "statistiquement" comme des suites aléatoires.

Les suites les plus courantes produites par les ordinateurs sont calculés à partir d'un nombre N d'entiers $0, \dots, N - 1$. En divisant par N , on obtient ainsi une suite à valeurs dans $[0, 1[$.

Elles sont construites sur la base de récurrences de la forme :

$$u_{n+1} = g(u_n)$$

où g : fonction de $0, \dots, N - 1$ dans lui même, u_0 : appelé graine, est à initialiser dans $0, \dots, N - 1$

On pose alors ; $x_n = \frac{u_n}{N} \in [0, 1[, n \in \mathbb{N}$

0.3.3 Simulation par inversion de la fonction de répartition

La donnée d'un générateur aléatoire uniforme permet, en théorie, de simuler suivant n'importe quelle distribution de variables aléatoires réelles en inversant la fonction de répartition.

Cette méthode est simple et efficace mais ne s'applique pas toujours, elle est basée sur l'utilisation de la fonction de répartition : $F(x) = P(X \leq x)$.

Définition 0.3.1 (Définition des quantiles). *Soit X une variable aléatoire réelle, de fonction de répartition F .*

On appelle quantile de niveau p ou p -quantile, de X (ou de sa loi de probabilité) une valeur x telle que,

$$P[X \leq x] = p$$

Définition 0.3.2 (Inverse généralisée). *Soit X une variable aléatoire réelle de fonction de répartition F .*

On appelle inverse généralisée (ou fonction quantile) de F , notée F^{-1} , la fonction définie pour tout $u \in [0, 1]$ par,

$$F^{-1}(u) = \inf\{x \in \mathbb{R}; F(x) \geq u\}$$

Théorème 0.3.3. *Soit X une variable aléatoire de fonction de répartition continue F . Alors, $F(x) = P(X \leq x)$, alors,*

$$U = F(X) \sim U[0, 1]$$

Théorème 0.3.4. *Soit $U \sim U[0, 1]$, alors,*

$$F^{-1}(U) \text{ et } X \text{ sont de même loi .}$$

Remarque 0.3.5. *L'inverse généralisée ne correspond à l'inverse (au sens bijection) que lorsque F est continue et strictement croissante.*

Lecture graphique de $F^{-1}(u)$ i.e, quantile d'ordre u .

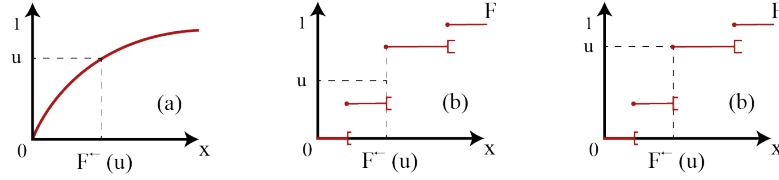


FIGURE 1 – Lecture Graphique

(a) : L'équation $F(x) = u$ a une unique solution.

(b) : L'équation $F(x) = u$ n'a pas de solutions.

(b') : L'équation $F(x) = u$ a une infinité de solutions.

on a :

$$\{x \in \mathbb{R}; F(x) \geq v \geq u\} \subseteq \{x \in \mathbb{R}; F(x) \geq u\} \Rightarrow F^{-1}(u) \leq F^{-1}(v)$$

L'inverse généralisée est donc en particulier mesurable.

0.3.3.0.1 Exemple 1 Dans le cas de la fonction F donnée dans les figures précédentes, nous avons

$$F(x) \begin{cases} \frac{x}{3} & x \in [0, 1[\\ \frac{x}{3} + \frac{1}{3} & x \in [1, 2] \\ 1 & x \geq 2 \\ 0 & x \leq 0 \end{cases}$$

$$F^{-1}(x) \begin{cases} 3x & x \in [0, \frac{1}{3}] \\ 1 & x \in [\frac{1}{3}, \frac{2}{3}] \\ 3x - 1 & x \in [\frac{2}{3}, 1] \end{cases}$$

Programme : Simulation par inversion de la fonction de répartition.

def inv(u) :

```

if ( $u < \frac{1}{3}$ )

 $z = 3u$ 

elif ( $u < \frac{2}{3}$ )

 $z = 1$ 

else

 $z = 3u - 1$ 

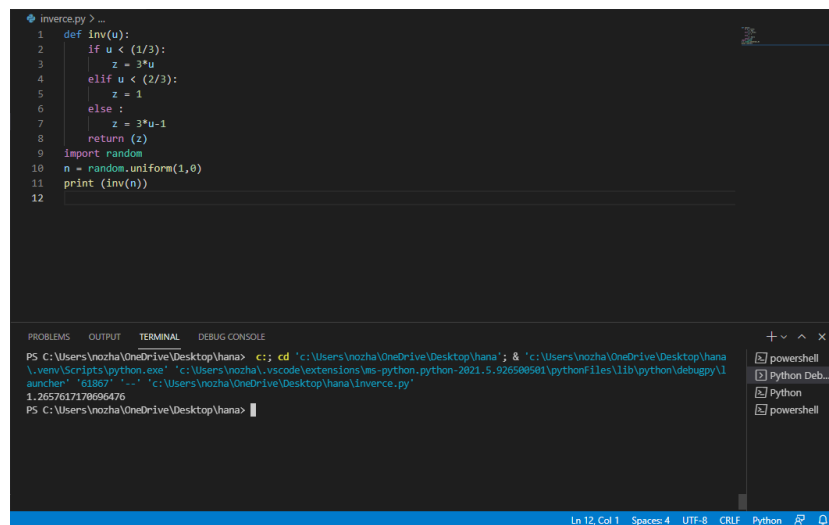
return(z)

import random

 $n = \text{random.uniform}(1,0)$ 

print(inv(n))

```



The image shows a screenshot of a Visual Studio Code editor window. The editor displays a Python script named 'inverse.py' with the following code:

```

1 def inv(u):
2     if u < (1/3):
3         z = 3*u
4     elif u < (2/3):
5         z = 1
6     else:
7         z = 3*u-1
8     return (z)
9
10 import random
11 n = random.uniform(1,0)
12 print (inv(n))

```

Below the editor, the 'TERMINAL' panel is open, showing the command prompt output of running the script. The command executed is:

```

PS C:\Users\nozha\OneDrive\Desktop\hana> c++; cd 'c:\Users\nozha\OneDrive\Desktop\hana'; & 'c:\Users\nozha\OneDrive\Desktop\hana\venv\Scripts\python.exe' 'c:\Users\nozha\.vscode\extensions\ms-python.python-2021.5.926508501\pythonFiles\lib\python\debugpy\launcher' '61867' '-.' 'c:\Users\nozha\OneDrive\Desktop\hana\inverse.py'

```

The output of the script is:

```

1.2657617178696476

```

The terminal also shows the prompt 'PS C:\Users\nozha\OneDrive\Desktop\hana>'.

FIGURE 2 – PYTHON CODE

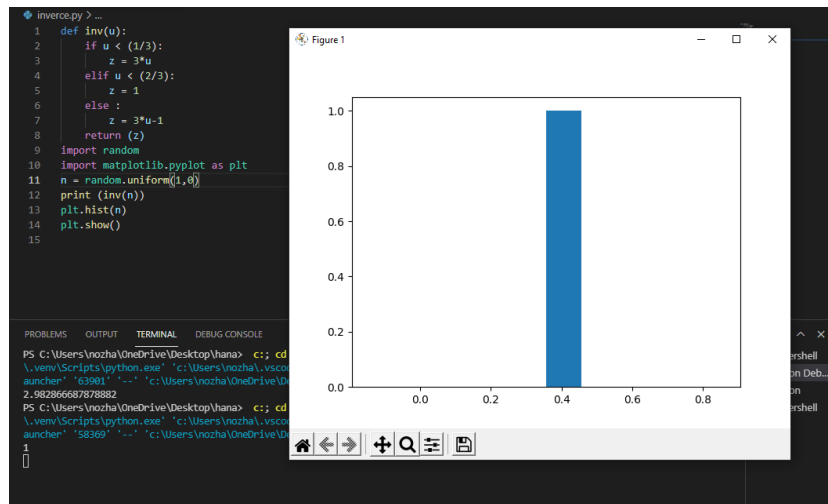


FIGURE 3 – Histogramme

0.3.3.0.2 Exemple 2 [Loi exponentielle]

Rappel : La loi exponentielle a pour densité

$$f(x) = \lambda e^{-\lambda x} 1_{x \geq 0}$$

et pour fonction de répartition

$$F(x) = (1 - e^{-\lambda x}) 1_{x \geq 0}$$

Soit $X \sim E(\lambda)$ une variable aléatoire de loi exponentielle de paramètre $\lambda > 0$.

F est bijective (la bijection réciproque et l'inverse généralisée coïncident) et pour tout $x \in [0, 1]$,

$$F^{-1}(u) = -\frac{1}{\lambda} \log(1 - u)$$

from numpy import

import numpy as np

import matplotlib.pyplot as plt

$n = 1000$

$l = 2$

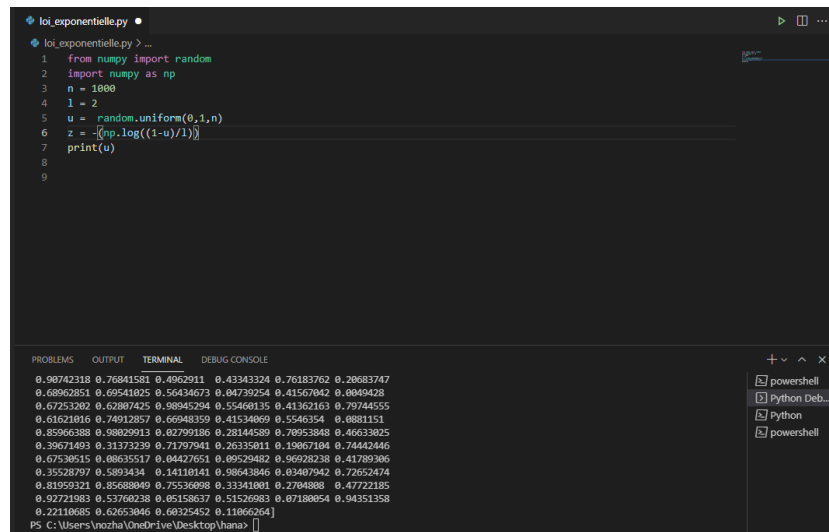
```
u = random.uniform(0,1,n)
```

```
z = -(np.log((1-u)/1))
```

```
print(u)
```

```
plt.hist(z)
```

```
plt.show()
```



```
loi_exponentielle.py
1 from numpy import random
2 import numpy as np
3 n = 1000
4 l = 2
5 u = random.uniform(0,1,n)
6 z = -(np.log((1-u)/1))
7 print(u)
8
9
```

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

```
0.90742319 0.76841581 0.4962911 0.43343324 0.76183762 0.20683747
0.68962851 0.69541825 0.56434673 0.04739254 0.41567042 0.0049428
0.67253202 0.62807425 0.98945294 0.55460135 0.41362163 0.79744555
0.61621016 0.74912857 0.66948359 0.41534069 0.5546354 0.8881151
0.85966388 0.98029913 0.02799186 0.28144589 0.70953848 0.46633825
0.30671493 0.31373239 0.71797941 0.26335911 0.19867104 0.74442846
0.67530515 0.08635517 0.04427651 0.09520482 0.96928238 0.41789306
0.35528797 0.5893434 0.14110141 0.98643846 0.03407942 0.72652474
0.81959321 0.85688049 0.75536098 0.33341001 0.2704888 0.47722185
0.92721983 0.53760238 0.05158637 0.51526983 0.07180854 0.94351358
0.22110685 0.62653046 0.60325452 0.11066264]
PS C:\Users\mno2ha\OneDrive\Desktop\Umana>
```

FIGURE 4 – PYTHON CODE

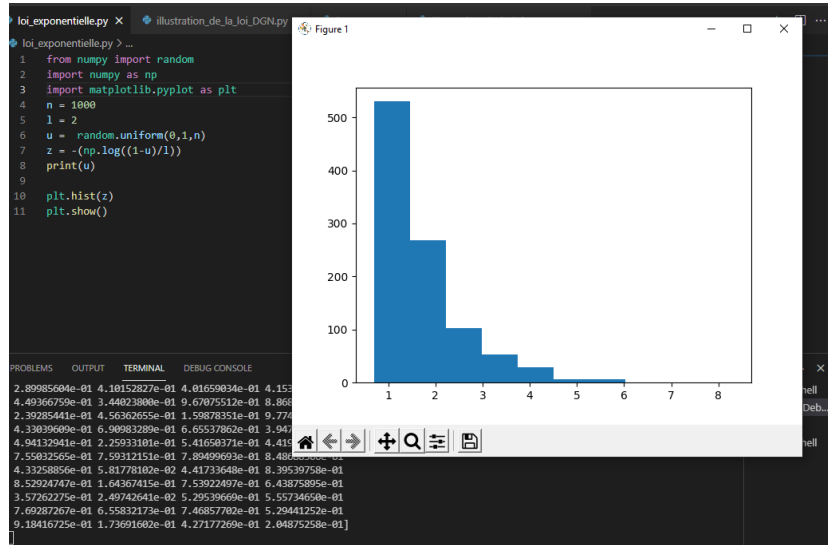


FIGURE 5 – Histogramme

0.4 Performance de la méthode de Monte-Carlo

0.4.1 Description de la méthode de Monte-Carlo

On appelle méthode de Monte-Carlo toute méthode visant à calculer numériquement une valeur approchée d'une espérance en se basant sur la loi des grands nombres.

La méthode de Monte-Carlo permet d'estimer la valeur d'une espérance en utilisant la simulation des variables aléatoires.

On suppose que l'on veuille calculer une intégrale I . En premier lieu, il faut la mettre sous forme d'une espérance $E(g(X))$ avec $g(X)$ est une variable aléatoire intégrable.

0.4.2 Convergence

Théorème 0.4.1 (Loi forte des grands nombres). *Soit $(X_n)_{n \in \mathbb{N}}$ une suite de variables aléatoires i.i.d à valeurs dans \mathbb{R}^d avec $d \in \mathbb{N}^*$.*

On suppose que $X \in L^1$, alors,

$$\frac{X_1 + \dots + X_n}{n} \xrightarrow[n \rightarrow +\infty]{P.S.} E(X)$$

Autrement dit,

$$\frac{1}{n} \sum_{i=1}^n X_i \approx E(X), \text{ pour } n \text{ assez grand.}$$

Ceci nous permet alors d'approcher

$$I \simeq \frac{g(X_1) + g(X_2) + \dots + g(X_n)}{n}, \text{ pour } n \text{ assez grand.}$$

L'objectif de la méthode de Monte-Carlo, est d'obtenir une meilleure approximation de l'intégrale I .

Soient X une variable aléatoire et g une fonction réelle mesurable telles que $g(X) \in L^1$.

Notre objectif est d'approcher $I = E(g(X))$.

La méthode de Monte-Carlo est formée alors par deux étapes :

Étape1 : Simuler n variables aléatoires i.i.d X_1, \dots, X_n de même loi que X (avec n assez grand).

Étape2 : Calculer l'estimateur \hat{I} de I :

$$\hat{I} = \frac{1}{n} \sum_{i=1}^n g(X_i)$$

Pour évaluer la performance de \hat{I} , nous considérons l'erreur quadratique moyenne d'un estimateur \hat{I}

Définition 0.4.2 (L'erreur quadratique). *L'erreur quadratique moyenne d'un estimateur \hat{I} d'un paramètre I (la notation en anglais MSE) est une mesure caractérisant la précision de cet estimateur.*

Elle est définie par :

$$MSE(\hat{I}, I) = E((\hat{I} - I)^2)$$

Théorème 0.4.3.

$$MSE(\hat{I}, I) = \text{Biais}(\hat{I})^2 + \text{Var}(\hat{I})$$

avec,

$$\text{Biais}(\hat{I}) = E(\hat{I}) - I$$

Preuve.

$$MSE(\hat{I}, I) = E[(\hat{I} - I)^2] = E[(\hat{I} - E(\hat{I}) + E(\hat{I}) - I)^2]$$

avec

$$\text{Biais}(\hat{I}) = E(\hat{I}) - I$$

Donc,

$$MSE(\hat{I}, I) = E[(\hat{I} - E(\hat{I}) + \text{Biais}(\hat{I}))^2] = E[(\hat{I} - E(\hat{I}))^2 + 2(\hat{I} - E(\hat{I}))\text{Biais}(\hat{I}) + \text{Biais}(\hat{I})^2] = E[(\hat{I} - E(\hat{I}))^2] + 2E[\hat{I} - E(\hat{I})]\text{Biais}(\hat{I}) + \text{Biais}(\hat{I})^2$$

d'où,

$$MSE(\hat{I}, I) = \text{Var}(\hat{I}) + 2(E(\hat{I}) - E(\hat{I}))\text{Biais}(\hat{I}) + \text{Biais}(\hat{I})^2 = \text{Var}(\hat{I}) + \text{Biais}(\hat{I})^2$$

□

Propriété 0.4.4. *Un estimateur est dit sans biais, si $\text{Biais}(\hat{I}) = 0$, autrement dit, $E(\hat{I}) = I$.*

Quand un estimateur est sans biais, l'erreur quadratique est égale à la variance.

$$MSE(\hat{I}, I) = E((\hat{I} - I)^2) = \text{Var}(\hat{I})$$

Proposition 0.4.5. *Si $MSE(\hat{I}, I)$ tend vers 0, quand n tend vers l'infini, alors \hat{I} est un estimateur convergent dans L^2 vers I .*

$$E((\hat{I} - I)^2) = \|\hat{I} - I\|_2^2 \xrightarrow{n \rightarrow +\infty} 0$$

Soient \hat{I}_1 et \hat{I}_2 deux estimateurs de I .

On dit que \hat{I}_1 est meilleur que \hat{I}_2 en terme de risque quadratique minimale si

$$MSE(\hat{I}_1, I) < MSE(\hat{I}_2, I)$$

Proposition 0.4.6 (Convergence Presque sûrement). *Une suite $(X_n)_{n \in \mathbb{N}}$ de variables aléatoires converge presque sûrement vers X ,*

$$\text{si } P\{\omega \in \Omega / \lim_{n \rightarrow +\infty} X_n(\omega) = X(\omega)\} = 1$$

On note :

$$X_n \xrightarrow[n \rightarrow +\infty]{P.S.} X$$

Soit X_1, \dots, X_n un n -échantillon aléatoire de même densité f . Alors,

$$\hat{I} = \frac{1}{n} \sum_{i=1}^n g(X_i) \xrightarrow[n \rightarrow +\infty]{P.S.} E(g(X_i)) = I$$

0.4.3 Validité et comportement de la méthode

La convergence de la méthode est assurée par la loi des grands nombres. Ce théorème nous donne sous certaines conditions la validité de l'approximation de la méthode de Monte-Carlo.

Exemple 0.4.7. *On souhaite à calculer*

$$I = \int_{-\infty}^{+\infty} x^4 \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}} dx$$

Illustration de la loi des grands nombres

```
from numpy import random
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
n = 1000
```

```
y = np.zeros ((n-2))
```

```
j = np.linspace(2, n-1 , n-2)
```

```
i = 1
```



```
while (y[i] >= 0 ) and (i+1 < len(y)) :  
    x = random.normal(0,1,i)  
    y[i] = np.mean(x**4)  
    i = i + 1  
d = np.mean(y)  
print(d)  
plt.plot(j,y)  
plt.show()
```

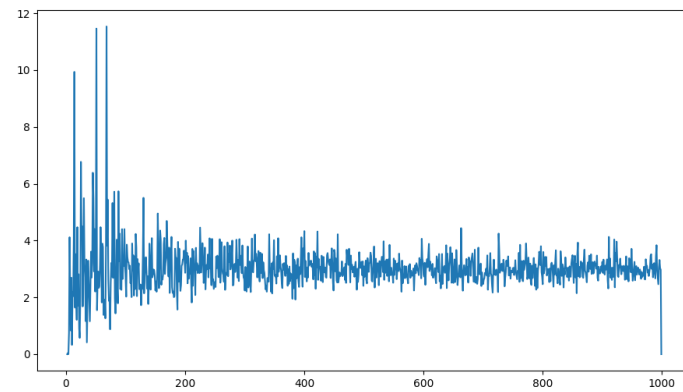


FIGURE 6 – Courbe

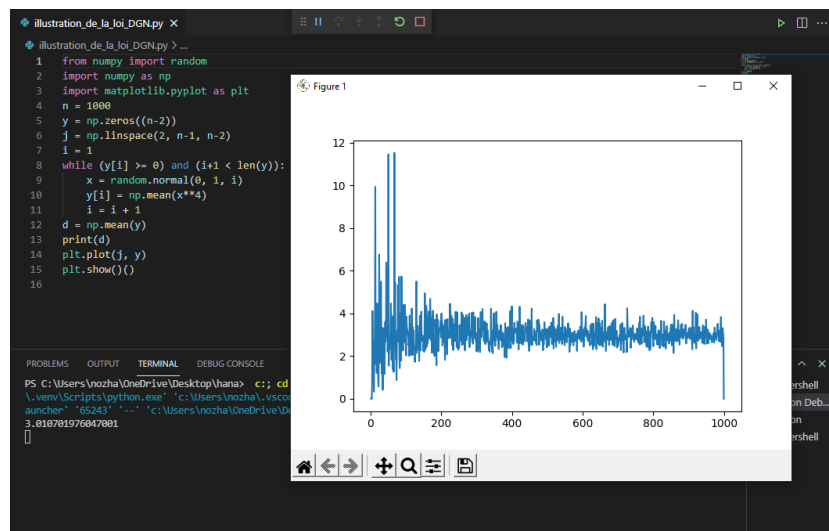


FIGURE 7 – PYTHON CODE

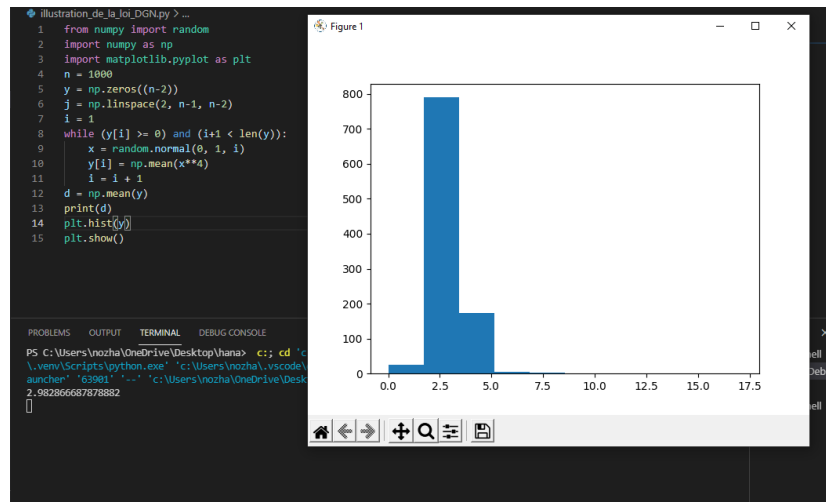


FIGURE 8 – Histogramme

0.4.4 Intervalle de confiance

Soit (X_1, \dots, X_n) un n - échantillon aléatoire de même loi P_θ , avec $\theta \in \Theta$ un paramètre inconnu.

Définition 0.4.8. Soit $\alpha \in]0, 1[$.

S'il existe des variables aléatoires réelles $\theta_{\min}(X_1, \dots, X_n)$ et $\theta_{\max}(X_1, \dots, X_n)$ telles que :

$$P(\theta \in [\theta_{\min}(X_1, \dots, X_n), \theta_{\max}(X_1, \dots, X_n)]) \geq 1 - \alpha.$$

On dit alors que $[\theta_{\min}(X_1, \dots, X_n), \theta_{\max}(X_1, \dots, X_n)]$ est un intervalle de confiance, avec un risque α . On le note $IC_{1-\alpha}(\theta)$.

Dans la pratique, on peut prendre par exemple $\alpha = 5\%$, ce qui nous donne un $IC_{95\%}$.

Cela signifie qu'il y a 95% de chance que la valeur inconnue θ soit comprise entre $\theta_{\min}(X_1, \dots, X_n)$ et $\theta_{\max}(X_1, \dots, X_n)$.

0.4.5 Vitesse de convergence et estimation de l'erreur

Définition 0.4.9 (Convergence en loi). Soit $(X_n)_{n \in \mathbb{N}}$ une suite de variables aléatoires.

On dit que X_n converge en loi vers une variable aléatoire X , si pour toute fonction continue, bornée,

$$E(g(X_n)) \xrightarrow{n \rightarrow \infty} E(g(X))$$

Théorème 0.4.10 (Théorème de la limite centrale TCL). Soit $(X_n)_{n \in \mathbb{N}}$ une suite de variables aléatoires i.i.d et à valeurs dans \mathbb{R} . On suppose que $E(|X_1|^2) < +\infty$.

On note $\sigma^2 = \text{Var}(X_1)$ et $\mu = E(X_1)$.

Alors,

$$\sqrt{n}(\bar{X}_n - \mu) \xrightarrow[n \rightarrow +\infty]{\text{Loi}} N(0, \sigma^2)$$

C'est à dire, pour tout $a < b$,

$$P(a \leq \frac{\sqrt{n}}{\sigma}(\frac{X_1 + \dots + X_n}{n} - \mu) \leq b) \xrightarrow[n \rightarrow +\infty]{} \int_a^b \frac{e^{-\frac{x^2}{2}}}{\sqrt{2\pi}} dx$$

Exemple 0.4.11. Si nous cherchons à calculer une intégrale de la forme

$$I = \int_{\mathbb{R}} g(x)f(x)dx$$

avec f est une densité de probabilités, alors nous pouvons écrire

$I = E(g(X))$ avec X est une variable aléatoire de densité f .

En appliquant TCL,

$$\sqrt{n}(\hat{I} - I) \xrightarrow[n \rightarrow +\infty]{\text{Loi}} N(0, \sigma^2)$$

avec

$$\sigma^2 = \text{Var}(g(X_1)) = \int_{\mathbb{R}} (g(x))^2 f(x)dx - I^2$$

Ainsi, \hat{I} est asymptotiquement normal et

$$I \in [\hat{I} - \frac{\hat{\sigma}}{\sqrt{n}} z_{1-\frac{\alpha}{2}}, \hat{I} + \frac{\hat{\sigma}}{\sqrt{n}} z_{1-\frac{\alpha}{2}}]$$

Intervalle de confiance de I pour un risque α .

$z_{1-\frac{\alpha}{2}}$: quantile de $N(0, 1)$ d'ordre $1 - \frac{\alpha}{2}$

($z_{1-\frac{\alpha}{2}} = F^{-1}(1 - \frac{\alpha}{2})$ avec F est la fonction de répartition de la loi $N(0, 1)$)

et

$$\tilde{\sigma}_n^2 = \frac{1}{n} \sum_{k=1}^n X_k^2 - \hat{I}^2$$

avec $\tilde{\sigma}_n^2$ est un estimateur de σ^2 par la méthode de Monte-Carlo.

Remarque 0.4.12. Notons que l'ordre de convergence de la méthode de Monte-Carlo est égale à $\frac{1}{\sqrt{n}}$

Exemple 0.4.13. On peut donner un intervalle de confiance pour le résultat.

Sous les hypothèses du TCL, on pose

$$\varepsilon_n = \frac{g(X_1) + \dots + g(X_n)}{n} - I$$

on cherche à approcher I à 0,01 près avec un niveau de confiance de 95%.
Donc,

$$P(|\varepsilon_n| \geq 0,01) \leq 0,05$$

équivalent à,

$$P(|\varepsilon_n| \leq 0,01) \geq 0,95.$$

Ainsi,

$$P(|\varepsilon_n| \leq 0,01) = P(-0,01 \leq \frac{g(X_1) + \dots + g(X_n)}{n} - I \leq 0,01)$$

$$\begin{aligned}
&= P(-0,01 \frac{\sqrt{n}}{\sigma} \leq (\frac{g(X_1) + \dots + g(X_n)}{n} - I) \frac{\sqrt{n}}{\sigma} \leq 0,01 \frac{\sqrt{n}}{\sigma}) \\
&= 2(\Phi(0,01 \frac{\sqrt{n}}{\sigma}) - 1),
\end{aligned}$$

pour n assez grand.

avec $(\Phi(0,01 \frac{\sqrt{n}}{\sigma}) = \int_{-\infty}^{0,01 \frac{\sqrt{n}}{\sigma}} \frac{e^{-\frac{t^2}{2}}}{\sqrt{2\pi}} dt$

On cherche la valeur de n telle que,

$$2(\Phi(0,01 \frac{\sqrt{n}}{\sigma}) - 1) \geq 0.95$$

donc,

$$2(\Phi(0,01 \frac{\sqrt{n}}{\sigma}) \geq 1.95$$

d'où,

$$(\Phi(0,01 \frac{\sqrt{n}}{\sigma}) \geq 0.975$$

D'après le tableau de la loi normale, il suffit d'avoir

$$0,01 \frac{\sqrt{n}}{\sigma} \geq \Phi^{-1}(0.975) = 1.96$$

.

Donc,

$$n \geq \frac{(1,96 * \sigma)^2}{0,01^2}$$

On remarque que le nombre de tirages nécessaires n pour atteindre un certain niveau d'erreur avec une certaine confiance qui est une fonction linéaire de σ^2 .

Dans la pratique σ^2 peut ne pas être connu mais on peut l'estimer par la méthode de Monte-Carlo.

Sous les hypothèses du TCL, la variance de l'erreur

$$\frac{g(X_1) + \dots + g(X_n)}{n} - I$$

est asymptotiquement d'ordre $\frac{\sigma^2}{n}$.

La variance σ^2 est aussi la variance des variables i.i.d dont on fait la moyenne empirique.

Bibliographie

- [1] JEAN PHILIPPE *Méthode de Simulation avec les variables antithétiques*, juin 2007
- [2] JULIEN STOEHR *Méthodes de Monte Carlo*, M1 2020-2021
- [3] FÉDÉRIC LEGRAND *Principe des méthodes de Monte Carlo*, 2020-2021
- [4] SYLVAIN RUBENTHALER *Méthodes de Monte Carlo*, M1 2018-2020
- [5] ALEXANDRE POPIER *Méthodes de Monte Carlo*, 2020-2021
- [6] RÉMI PEYRE *Méthode de Monte-Carlo et application aux Processus Aléatoires*, Février 2016
- [7] LOGICIEL R *Notations Mathématiques*.
- [8] ARON COURVILLE *Propriétés des estimateurs*.
- [9] J. MALET *Quantiles et simulation*, CEA 2017
- [10] WIKIPEDIA *Variable antithétique*.
- [11] WIKISTAT *Estimations et intervalle de confiance*.
- [12] MARBLESCIENCE *Monte Carlo Simulation*.
- [13] WIKIPEDIA *Erreur quadratique moyenne*.