

Maxime Cordier
mcardier@enssat.fr

Malo Louboutin
mloubout@enssat.fr

Module :
Probabilités et statistiques

Promotion :
FISA IAI 2

Devoir maison 2 « Loi des grands nombres et marche aléatoire »

1 Loi des grands nombres

1.a Partie 1

Nous avons choisi de générer des nombres aléatoires pour une loi discrète uniforme et pour une loi continue exponentielle. La figure 1 ci-dessous regroupe des représentations graphiques de génération de nombre aléatoire pour la loi discrète uniforme. Ces nombres ont été créés à l'aide d'une fonction génératrice congruentielle linéaire.

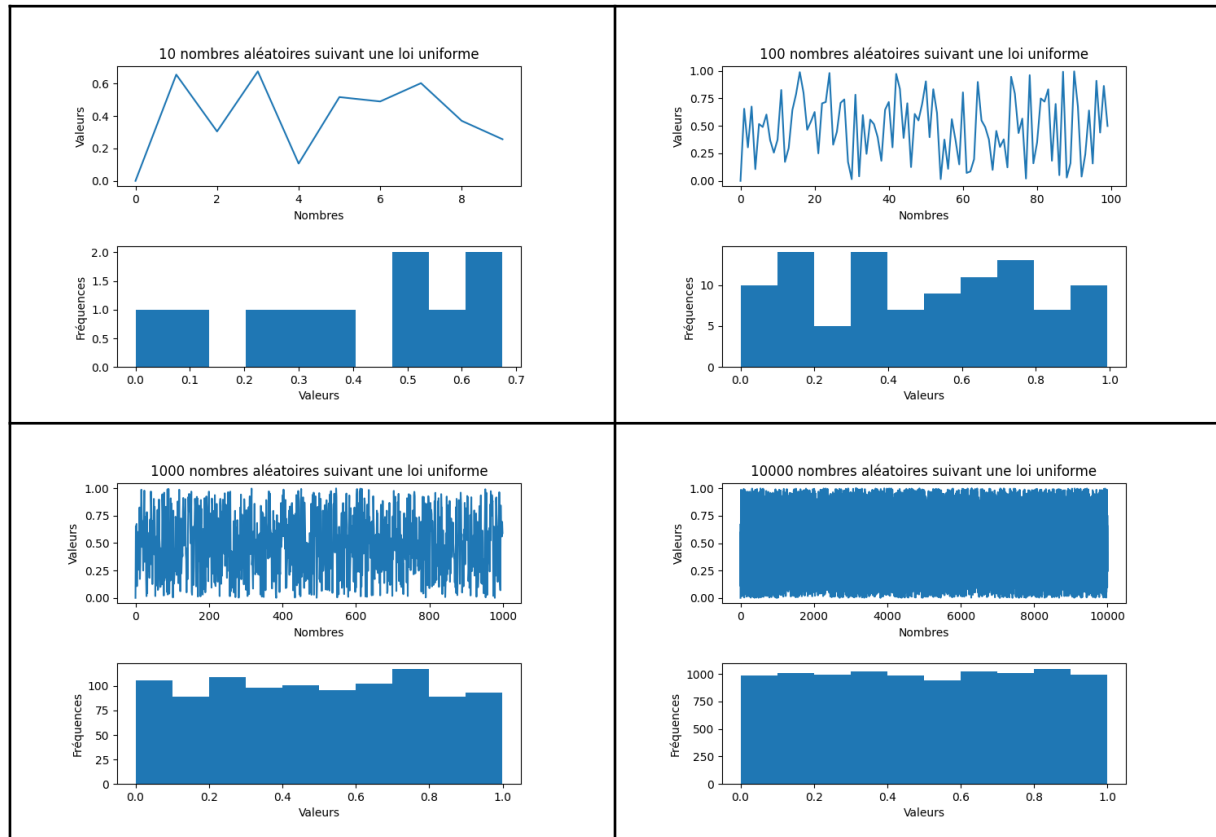


Tableau 1: Génération de N nombres aléatoires suivant une loi uniforme, pour plusieurs valeurs de N.

On réalise maintenant le test du χ^2 pour la loi uniforme. Pour cela, on prend en compte les nombres X_i générés via notre fonction génératrice congruentielle linéaire et les nombres Y_i générés à l'aide de la bibliothèque random de python. La distance entre ces deux méthodes de générations se mesure par la variable aléatoire D^2 . On a :

$$D^2 = \sum_{i=1}^k \frac{(X_i - Y_i)^2}{Y_i}$$

On compare ensuite cette valeur à la valeur critique de la loi du χ^2 à N-1 degrés de liberté.

Exemple : Pour 10 nombres on s'intéresse à la valeur critique de la loi du χ^2 à un degré de 9. Si D^2 est inférieur à cette valeur critique, alors on accepte l'hypothèse que les 10 nombres générés suivent une loi uniforme. Sinon, on rejette cette hypothèse.

Ensuite, la figure 2 ci-dessous regroupe des représentations graphiques de génération de nombre aléatoire pour la loi continue exponentielle. Ces nombres ont été créés à l'aide d'une fonction génératrice congruentielle linéaire et de la fonction inverse de la fonction de répartition de la loi exponentielle ayant pour paramètre λ .

Cette fonction inverse a pour équation : $F^{-1}(x) = \frac{-\ln(1-x)}{\lambda}$

Pour la génération des nombres aléatoires suivant une loi exponentielle, nous avons choisi de prendre $\lambda = 3$.

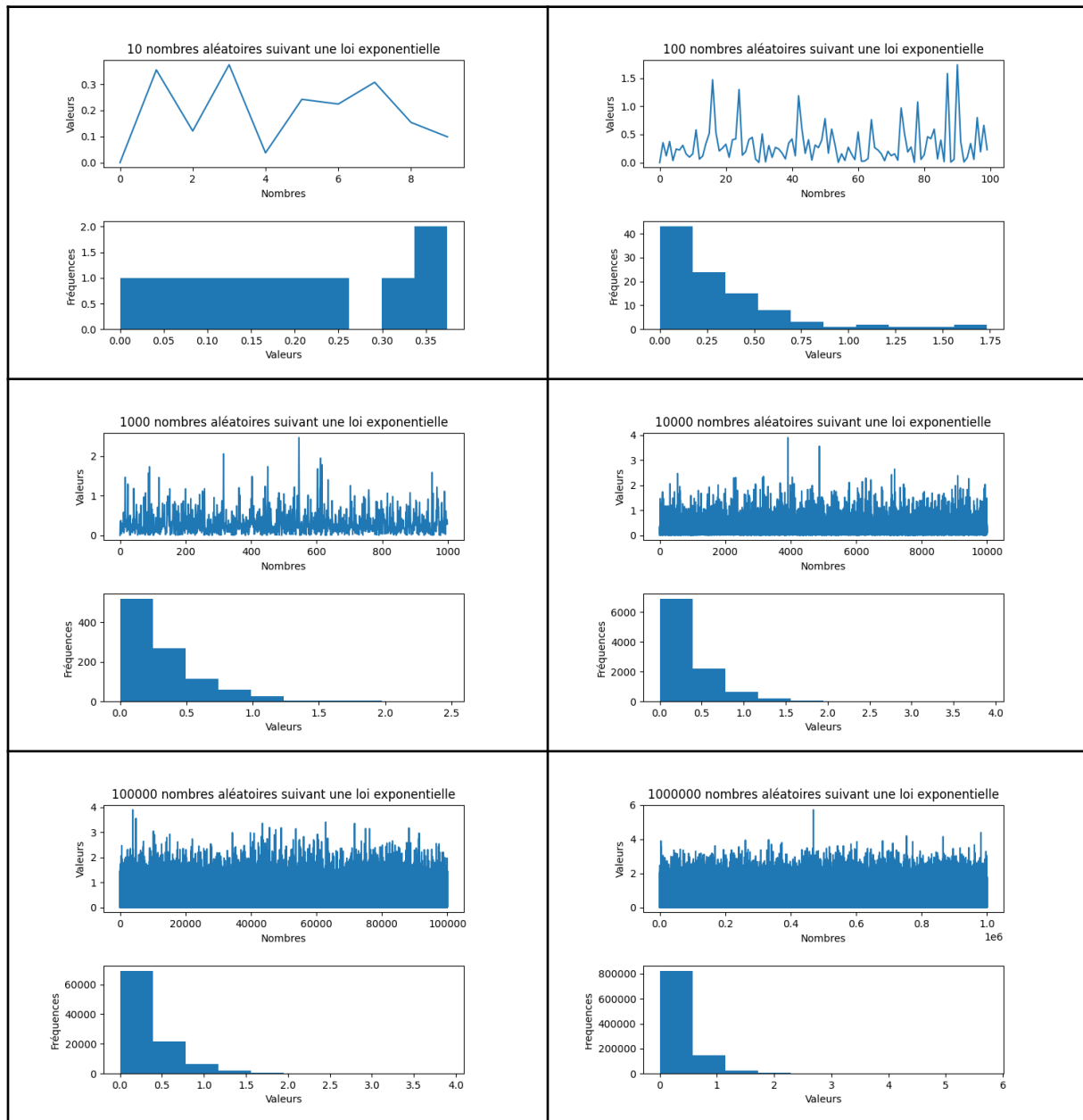


Tableau 2: Génération de N nombres aléatoires suivant une loi exponentielle, pour plusieurs valeurs de N.

Nous réalisons également un test du χ^2 pour la loi exponentielle. Il s'agit de la même méthode que pour la loi uniforme.

Pour conclure, nous avons remarqué que les nombres générés à l'aide du générateur congruenciel linéaire suivent restent constamment les mêmes. Exécuter une nouvelle fois le code ne change rien. Cela est dû au fait que le générateur congruenciel linéaire génère certes, des nombres aléatoires, mais toujours avec le même calcul et les mêmes paramètres. Les nombres générés sont donc tout le temps les mêmes. Ce n'est pas le cas avec les fonctions de la bibliothèque random. En effet, ces fonctions utilisent des paramètres différents à chaque exécution. Les nombres générés sont donc différents à chaque exécution.

1.b Partie 2

Rappel : Soit N variables aléatoires X_i indépendantes de même loi. La moyenne empirique \overline{X}_N est définie par : $\overline{X}_N = \frac{1}{N} \sum_{i=1}^N X_i$

On note l'espérance de \overline{X}_N : $E(\overline{X}_N)$. On a :

$$E(\overline{X}_N) = E\left(\frac{1}{N} \sum_{i=1}^N X_i\right) = \frac{1}{N} \sum_{i=1}^N E(X_i)$$

Or les X_i sont indépendantes et suivent la même loi, donc $E(X_i) = E(X_1)$ pour tout i . On a donc :

$$\frac{1}{N} \sum_{i=1}^N E(X_i) = \frac{1}{N} \sum_{i=1}^N E(X_1) = \frac{1}{N} N E(X_1) = E(X_1)$$

Ainsi $E(\overline{X}_N) = E(X_1)$.

On note l'écart-type de \overline{X}_N : $\sigma(\overline{X}_N)$. Par ailleurs, nous savons que l'écart-type vaut la racine carré de la variance. Déterminons premièrement la variance de \overline{X}_N . On note $V(\overline{X}_N)$ cette variance. On a :

$$V(\overline{X}_N) = V\left(\frac{1}{N} \sum_{i=1}^N X_i\right) = \frac{1}{N^2} \sum_{i=1}^N V(X_i)$$

Or les X_i sont indépendantes et suivent la même loi, donc $V(X_i) = V(X_1)$ pour tout i . On a donc :

$$\frac{1}{N^2} \sum_{i=1}^N V(X_i) = \frac{1}{N^2} \sum_{i=1}^N V(X_1) = \frac{1}{N^2} N V(X_1) = \frac{V(X_1)}{N}$$

Ainsi $V(\overline{X}_N) = \frac{V(X_1)}{N}$.

Pour conclure, $\sigma(\overline{X}_N) = \sqrt{\frac{V(X_1)}{N}}$.

La figure 3 ci-dessous regroupe des représentations graphiques de la moyenne empirique pour plusieurs nombres de variables aléatoires.

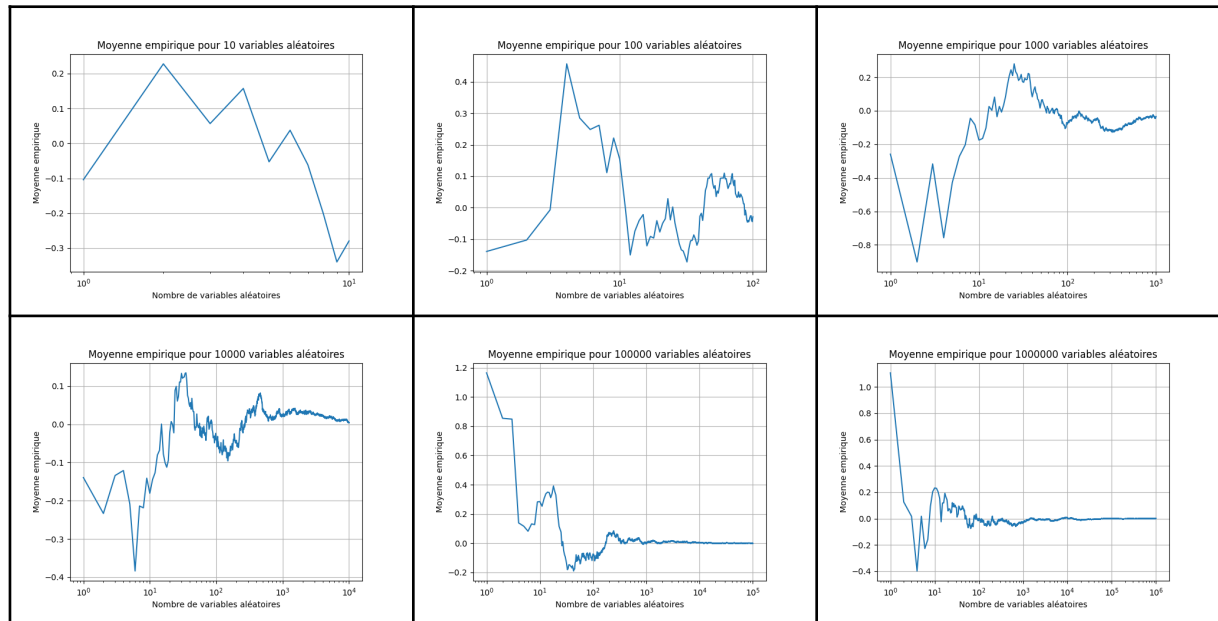


Tableau 3: Moyenne empirique pour N variables aléatoires.

Ces variables aléatoire sont générées avec la fonction `randn` de la bibliothèque `random`. Ces variables suivent alors une loi normale centrée réduite $N(0,1)$.

Pour chaque N , nous avons également calculer l'espérance et l'écart type :

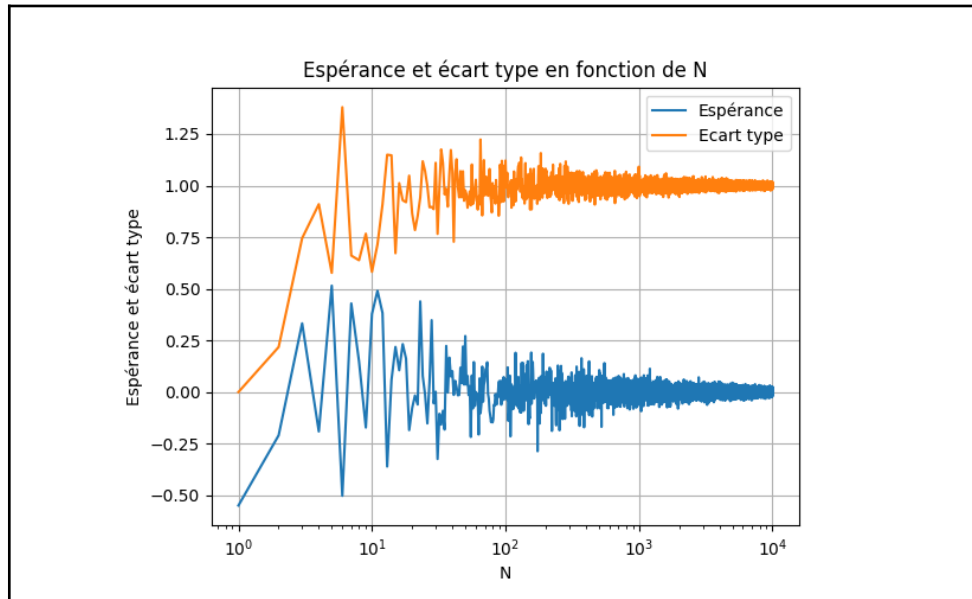


Tableau 4: Espérance et écart-type en fonction de N variables aléatoires.

Nous pouvons remarquer que l'écart type tend vers 1. L'espérance tend par contre vers 0 comme la moyenne empirique. On vient de mettre en évidence la loi des grands nombres qui est : plus le nombre d'observations augmente, plus la moyenne empirique se rapproche de l'espérance théorique de la variable aléatoire.

2 Marche aléatoire

2.a Partie 1

Dans le contexte d'une marche aléatoire où un jet de pièce équilibrée détermine le déplacement à droite (+s) ou à gauche (-s), la variable aléatoire X_i peut être modélisée comme une variable aléatoire suivant la loi de Rademacher.

La loi de Rademacher est définie par la distribution de probabilité suivante :

$$P(X_i = +1) = \frac{1}{2}$$

$$P(X_i = -1) = \frac{1}{2}$$

Dans cette formulation, la valeur +1 représente un déplacement vers la droite (face), et la valeur -1 représente un déplacement vers la gauche (pile).

Ainsi, la loi de la variable aléatoire X_i dans ce cas est la loi de Rademacher.

L'erreur ici serait de penser à une loi de Bernoulli, qui ne comptabilise que les succès (0 et 1, et non pas -1 et 1)

Pour déterminer la loi de X , qui représente la position du marcheur à l'instant nT , nous devons prendre en compte la somme des déplacements individuels X_i jusqu'à cet instant.

Comme mentionné précédemment, chaque déplacement X_i suit une loi de Rademacher avec des probabilités égales de +s et -s, toutes deux égales à $\frac{1}{2}$.

La variable $X(n * T, \omega)$ peut être exprimée comme la somme des déplacements individuels jusqu'à l'instant nT :

$$X(n * T, \omega) = X_1 + X_2 + \dots + X_n$$

La somme de variables aléatoires indépendantes et identiquement distribuées (i.i.d.) suit une autre loi de probabilité. Dans ce cas, la somme des déplacements X_i suit une loi de distribution normale, également appelée distribution gaussienne.

Si on prend $s = T = 1$, on peut représenter X pour $n=20,100,1000$.

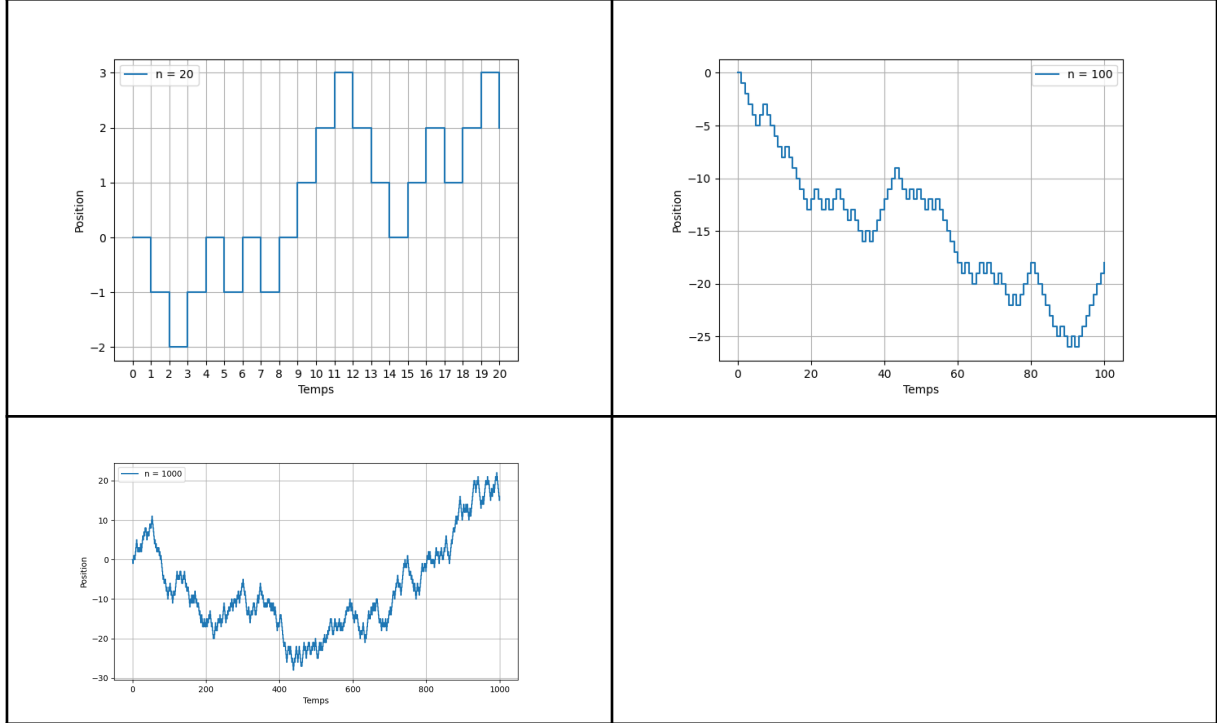


Tableau 5: Représentation de X pour $n=20,100,1000$.

Le code qui a permis de générer ces représentation de X est disponible en annexe (1).

Pour calculer l'espérance de $X(n * T, \omega)$, nous pouvons utiliser la propriété de linéarité de l'espérance.

On va de plus normaliser $\{-s; +s\}$ à $\{-1; +1\}$ pour nous simplifier la tâche.

$$E[X(n * T, \omega)] = E[X_1 + X_2 + \dots + X_n]$$

Comme les déplacements normalisés X_i suivent une distribution de Rademacher avec une probabilité de $+1$ et -1 égales à $\frac{1}{2}$ chacune, l'espérance de chaque déplacement individuel est de 0 :

$$E[X_i] = (+1) * \frac{1}{2} + (-1) * \frac{1}{2} = 0$$

Puisque les déplacements normalisés ont une espérance de 0, la somme des déplacements normalisés jusqu'à l'instant nT donne également une espérance de 0 :

$$E[X(n * T, \omega)] = E[X_1 + X_2 + \dots + X_n] = E[X_1] + E[X_2] + \dots + E[X_n] = 0 + 0 + \dots + 0 = 0$$

Ainsi, dans le cas d'une marche aléatoire symétrique avec des déplacements normalisés à $+1$ et -1 , la variable aléatoire $X(n * T, \omega)$ a une espérance de 0

Pour calculer la variance de X , nous devons tenir compte des déplacements individuels X_i .

Comme nous l'avons précédemment établi, chaque déplacement X_i suit une loi de Rademacher définie par :

$$P(X_i = +1) = \frac{1}{2}$$

$$P(X_i = -1) = \frac{1}{2}$$

La variance de $X(n * T, \omega)$ peut être calculée en utilisant la propriété de linéarité de la variance. Comme les déplacements individuels X_i sont indépendants, la variance totale est la somme des variances des déplacements individuels :

$$V[X(n * T, \omega)] = V[X_1 + X_2 + \dots + X_n]$$

Comme les déplacements X_i sont i.i.d., nous pouvons simplifier davantage :

$$V[X(n * T, \omega)] = V[X_1] + V[X_2] + \dots + V[X_n]$$

Chaque déplacement X_i suit une loi de Rademacher, et la variance de la loi de Rademacher est définie par $E[X_i^2] - (E[X_i])^2$. Dans ce cas, l'espérance des déplacements X_i est de 0 (comme nous l'avons déterminé précédemment).

Ainsi, nous avons :

$$V[X_i] = E[X_i^2] - (E[X_i])^2 = E[X_i^2] = (+\sigma)^2 * \frac{1}{2} + (-\sigma)^2 * \frac{1}{2} = \sigma^2 * \frac{1}{2} + \sigma^2 * \frac{1}{2} = \sigma^2$$

Puisque chaque déplacement X_i a une variance de σ^2 , la variance totale de $X(n * T, \omega)$ est la somme des variances des déplacements individuels :

$$V[X(n * T, \omega)] = V[X_1] + V[X_2] + \dots + V[X_n] = \sigma^2 + \sigma^2 + \dots + \sigma^2 \text{ (n termes au total)} = n * \sigma^2$$

Ainsi, la variance de la variable aléatoire $X(n * T, \omega)$ est $n * \sigma^2$ dans le cas d'une marche aléatoire symétrique.

2.b Partie 2

Ici une représentation de X pour n fixé à 100 et T de plus en plus petit.

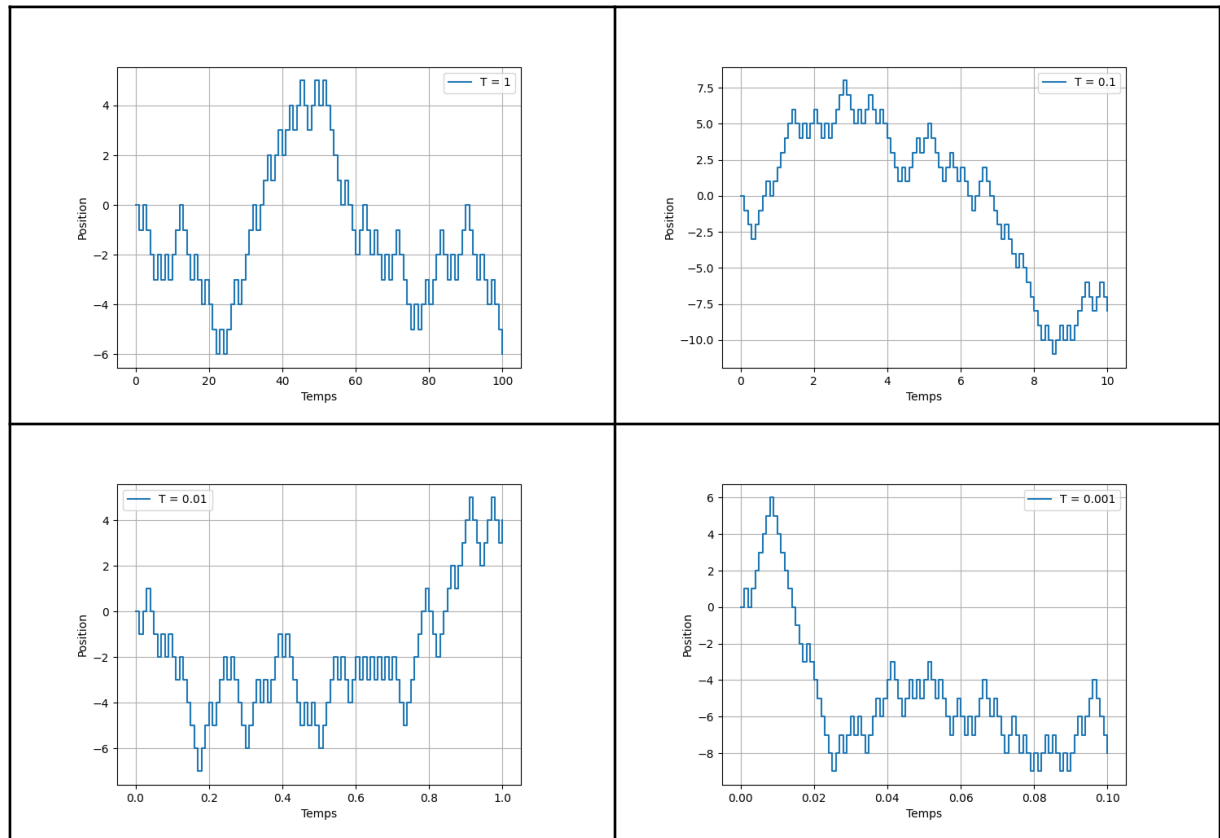


Tableau 6: Représentation de X pour n=100 et T=1,0.1,0.01,0.001

Le code qui a permis de générer ces représentation de X est disponible en annexe (2).

Pour démontrer que $X(t)$ suit une loi normale de moyenne 0 et d'écart-type \sqrt{t} , nous allons utiliser le théorème central limite.

Dans notre cas, chaque déplacement individuel suit une loi de Rademacher avec une probabilité de +1 ou -1, et chaque déplacement est de longueur s .

La variable aléatoire $X(t)$ est la somme de n déplacements effectués au cours des n intervalles de temps.

Le déplacement total à l'instant t est donné par $X(t) = X_1 + X_2 + \dots + X_n$.

L'espérance de chaque déplacement individuel est :

$$E(X_i) = 1 * P(X_i = 1) + (-1) * P(X_i = -1) = 1 * \frac{1}{2} + (-1) * \frac{1}{2} = 0.$$

La variance de chaque déplacement individuel est :

$$V(X_i) = E(X_i^2) - (E(X_i))^2 = \left(1^2 * \frac{1}{2} + (-1)^2 * \frac{1}{2}\right) - 0^2 = \frac{1}{2} + \frac{1}{2} = 1.$$

La moyenne de $X(t)$ est :

$$E(X(t)) = E(X_1 + X_2 + \dots + X_n) = E(X_1) + E(X_2) + \dots + E(X_n) = 0 + 0 + \dots + 0 = 0.$$

La variance de $X(t)$ est :

$$V(X(t)) = V(X_1 + X_2 + \dots + X_n) = V(X_1) + V(X_2) + \dots + V(X_n) = 1 + 1 + \dots + 1 = n.$$

Maintenant, nous devons tenir compte du fait que $n = \frac{t}{T}$, où T est l'intervalle de temps entre les déplacements et t est le temps total.

Par conséquent, $V(X(t)) = n = \frac{t}{T}$.

En utilisant la définition de la variance, nous pouvons réécrire :

$$V(X(t)) = \frac{t}{T}.$$

Pour obtenir l'écart-type, nous prenons la racine carrée de la variance :

$$\text{Écart-type}(X(t)) = \sqrt{\frac{t}{T}} = \frac{\sqrt{t}}{\sqrt{T}}.$$

Donc, l'écart-type de $X(t)$ est $\frac{\sqrt{t}}{\sqrt{T}}$.

Si nous prenons $T = 1$, nous obtenons :

$$\text{Écart-type}(X(t)) = \sqrt{t}.$$

Ainsi, $X(t)$ suit une loi normale de moyenne 0 et d'écart-type \sqrt{t} , correspondant au mouvement brownien.

3 Annexe

(1)

```
import numpy as np
import matplotlib.pyplot as plt
import time

# Paramètres
s = 1 # Longueur du pas
T = 1 # Intervalle de temps
n_values = [20, 100, 1000] # Valeurs de n

def marche_aleatoire(n, s):
    positions = np.zeros(n+1) # Tableau pour stocker les positions
    for i in range(1, n+1):
        deplacement = np.random.choice([-s, s]) # Déplacement aléatoire (-1 pour
gauche, 1 pour droite)
        positions[i] = positions[i-1] + deplacement
    return positions

# Trajectoires
for n in n_values:
    positions = marche_aleatoire(n, s)
    time.sleep(T)
    temps = np.arange(0, n+1) * T
    plt.step(temps, positions, where='post', label=f'n = {n}')

# Paramètres de la figure
plt.xlabel('Temps')
plt.ylabel('Position')
plt.legend()
plt.grid(True)

# Affichage de la figure
plt.show()
```

(2)

```
import numpy as np
import matplotlib.pyplot as plt

def marche_aleatoire(n, T):
    positions = np.zeros(n+1) # Tableau pour stocker les positions
    for i in range(1, n+1):
        deplacement = np.random.choice([-1, 1]) # Déplacement aléatoire (-1 pour
gauche, 1 pour droite)
        positions[i] = positions[i-1] + deplacement
    return positions

# Paramètres
s = 1 # Longueur du pas
n = 100 # Nombre d'intervalles de temps fixé
T_values = [1, 0.1, 0.01, 0.001] # Valeurs de T de plus en plus petites

# Trajectoires
for T in T_values:
    positions = marche_aleatoire(n, T)
```

```
temps = np.arange(0, n+1) * T
plt.step(temps, positions, where='post', label=f'T = {T}')

# Paramètres de la figure
plt.xlabel('Temps')
plt.ylabel('Position')
plt.legend()
plt.grid(True)

# Affichage de la figure
plt.show()
```