

# Étude du Caractère Pseudo-aléatoire des Décimales de $e$ et générateur de loi uniforme $[0, 1[$

LAMLIH Houssam

May 31, 2023

## 1 Introduction

Le nombre  $e$  est une constante mathématique importante qui représente la base du logarithme naturel. Les décimales de  $e$  ont été étudiées avec intérêt pour leur nature pseudo-aléatoire. Dans ce rapport, nous utilisons le test du chi-deux et le test du poker pour évaluer si les décimales de  $e$  présentent des propriétés pseudo-aléatoires, par la suite on crée notre propre générateur de loi uniforme en utilisant les décimales de  $e$ .

## 2 Étude des Décimales de $e$

### 2.1 Test du Chi-Deux

Le test du chi-deux est utilisé pour évaluer la similarité entre les fréquences observées et les fréquences attendues. La statistique du test du chi-deux est calculée en sommant les carrés des différences normalisées entre les fréquences observées et les fréquences attendues.

#### 2.1.1 Comptage des chiffres

Nous initialisons un dictionnaire pour compter les occurrences de chaque chiffre de 0 à 9 dans les décimales de  $e$ . En parcourant chaque chiffre de  $e$ , nous mettons à jour les comptes dans le dictionnaire.

#### 2.1.2 Calcul des fréquences attendues

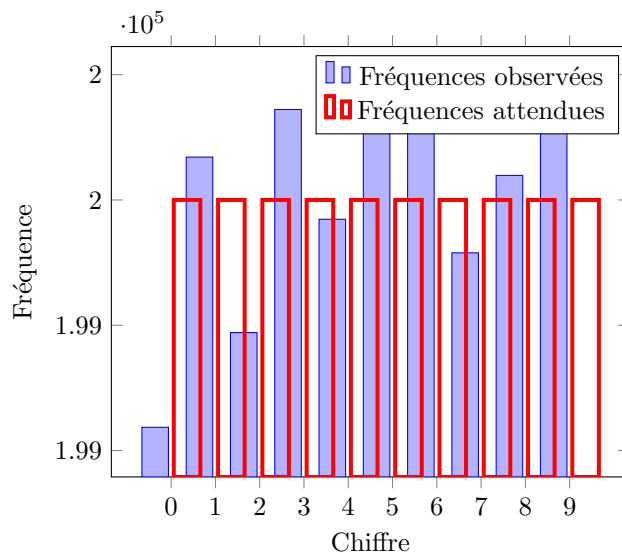
Le nombre total de décimales de  $e$  est calculé. Ensuite, les fréquences attendues pour chaque chiffre sont calculées en supposant une répartition uniforme ( $1/10$  pour chaque chiffre).

## 2.2 Résultats

### 2.2.1 Statistique du test du chi-deux

La statistique du test du chi-deux obtenue pour les décimales de  $e$  est de 8.65376.

### 2.2.2 Graphique des fréquences observées et attendues



Le graphique ci-dessus montre les fréquences observées des chiffres dans les décimales de  $e$  ainsi que les fréquences attendues (uniformes) pour chaque chiffre.

## 2.3 Discussion

Le test du chi-deux nous permet d'évaluer la similarité entre les fréquences observées et les fréquences attendues. Une statistique du test du chi-deux plus élevée suggère une plus grande différence entre les fréquences observées et attendues, indiquant un caractère moins pseudo-aléatoire des décimales de  $e$ .

Dans notre cas, la statistique du test du chi-deux obtenue est 8.65376. Il serait nécessaire d'évaluer cette valeur par rapport à un seuil de signification approprié pour tirer des conclusions plus définitives sur le caractère pseudo-aléatoire des décimales de  $e$ .

Si on considère une erreur de première espèce  $\alpha$  de .05, la valeur critique de la statistique du chi-deux est de 16.919 donc l'hypothèse est acceptée.

## 2.4 Test du poker

Le test du poker est utilisé pour évaluer la distribution des combinaisons de poker dans une séquence de nombres. Il permet de déterminer si la séquence

présente des caractéristiques de randomité conformes à celles d'un jeu de cartes bien mélangé.

### 2.4.1 Création des groupes

Tout d'abord, nous divisons la séquence de nombres en groupes de cinq éléments. Chaque groupe représente une main de poker.

### 2.4.2 Comptage des classes de poker

Pour chaque main de poker, nous comptons le nombre de classes de poker distinctes présentes. Une classe de poker est définie par le nombre de valeurs uniques dans la main (rien, une paire, un brelan, un carre, un poker). Nous enregistrons les occurrences de chaque classe de poker.

### 2.4.3 Calcul des fréquences attendues

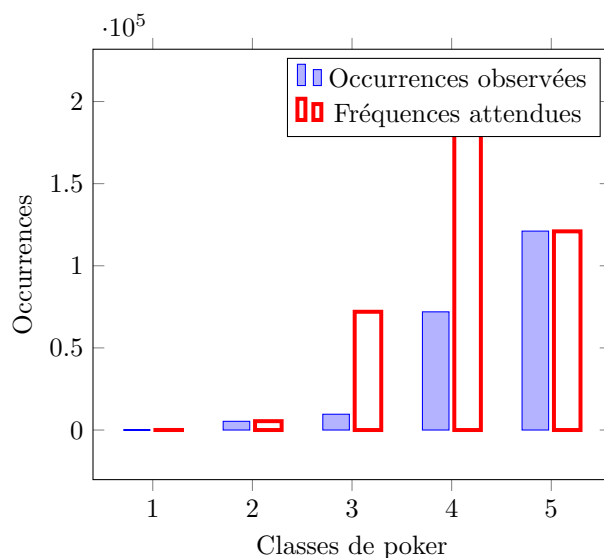
Nous calculons les fréquences attendues pour chaque classe de poker en utilisant les coefficients de Stirling et les facteurs combinatoires. Les fréquences attendues sont déterminées en supposant que les nombres de la séquence suivent une distribution aléatoire uniforme.

## 2.5 Résultats

### 2.5.1 Statistique du test du poker

La statistique du test du poker obtenue pour les décimales de  $e$  est de 2.89

### 2.5.2 Graphique des occurrences observées et attendues



Le graphique ci-dessus montre les occurrences observées des classes de poker dans les décimales de  $e$  ainsi que les fréquences attendues pour chaque class.

## 2.6 Discussion

Le test du poker nous permet d'évaluer la distribution des combinaisons de poker dans une séquence de nombres. Une statistique du test du poker plus élevée suggère une plus grande différence entre les occurrences observées et les fréquences attendues, indiquant un caractère moins pseudo-aléatoire des décimales de  $e$ .

Dans notre cas, la statistique du test du poker obtenue est 2.89. Il serait nécessaire d'évaluer cette valeur par rapport à un seuil de signification approprié pour tirer des conclusions plus définitives sur le caractère pseudo-aléatoire des décimales de  $e$ .

Si on considère une erreur de première espèce  $\alpha$  de 05, la valeur critique de la statistique du poker dépend du nombre de classes de poker ( $k$ ) et de la taille de l'échantillon ( $n$ ). Avec  $k = 5$  et  $n = 400000$ , la valeur critique est d'environ 9.488. Par conséquent, l'hypothèse nulle d'une distribution pseudo-aléatoire des décimales de  $e$  ne peut être rejetée.

## 3 Génération de nombres aléatoires

Ce chapitre explique deux générateurs de nombres implémentés en Python qui utilisent les décimales de la constante mathématique  $e$ . Les générateurs s'appellent LCERandomGenerator et ESimpleRandomGenerator.

### 3.1 LCERandomGenerator

Le LCERandomGenerator est basé sur l'algorithme du Linear Congruential Method (LCM) et utilise les décimales de  $e$  pour générer des nombres aléatoires. Le générateur est initialisé avec une valeur de départ (graine), la valeur de  $e$  utilisée pour générer les nombres aléatoires.

#### 3.1.1 Initialisation

Le LCERandomGenerator est initialisé avec les paramètres suivants :

- **seed (int)** : La valeur de départ (graine) du générateur.
- **e\_value (str)** : La valeur de  $e$  utilisée pour générer les nombres aléatoires.
- **precision (int)** : La précision du générateur, déterminant le nombre de décimales de  $e$ .

Le générateur stocke la graine initiale, la dernière valeur générée, le dernier indice calculé, le nombre de décimales à considérer à partir de  $e$  et la différence entre la précision et le nombre de décimales.

### 3.1.2 Calcul des paramètres optimaux

Pour maximiser le cycle du générateur, des paramètres optimaux sont calculés en fonction de la précision et du nombre de décimales. Les paramètres sont calculés comme suit en appliquant le théorème de Hull et Dobel:

1. Initialiser le paramètre  $c$  à 2.
2. Incrémenter  $c$  jusqu'à ce qu'il soit premier avec la précision ou que  $c$  dépasse la précision.
3. Calculer les diviseurs premiers de la précision à l'aide de la fonction `get_prime_dividers`.
4. Calculer le paramètre  $a$  en multipliant tous les diviseurs premiers calculés et en ajoutant 1.
5. Si la précision est divisible par 4 et que  $a$  n'est pas divisible par 4, multiplier  $a$  par 4 (ou par 2 si  $a$  est divisible par 2).

Le générateur utilise ensuite les valeurs calculées de  $a$ ,  $c$  et la précision pour générer des nombres aléatoires.

### 3.1.3 Comparaison avec `random.random` de Python

Ce chapitre présente une comparaison entre le générateur LCM de nombres aléatoires personnalisé et le générateur aléatoire fourni par le module `random` de Python. L'objectif est d'évaluer la performance et la qualité de vos générateurs de nombres aléatoires par rapport à la référence du générateur de Python. Pour cette comparaison nous allons générer 100 nombres aléatoires entre 0 et 1 avec la seed 42 pour notre générateur personnalisé.

On va effectuer des tests statistiques pour comparer la distribution des nombres aléatoires générés par les deux générateurs :

#### Test de Kolmogorov-Smirnov

Ce test produit une statistique  $D$ , qui mesure la plus grande différence entre les fonctions de répartition empiriques des deux échantillons. Plus la valeur de  $D$  est grande, plus les distributions sont différentes.

#### Résultats

Les résultats du test de Kolmogorov-Smirnov sont les suivants :

- Générateur personnalisé : Valeur de  $D = 0.134$
- Générateur aléatoire de Python : Valeur de  $D = 0.0867$

## Discussion

La valeur de  $D$  obtenue pour le générateur personnalisé est de 0.134, tandis que le générateur aléatoire de Python a une valeur de  $D$  de 0.0867. Cela indique que la différence entre les distributions empiriques des deux générateurs est plus grande pour le générateur personnalisé.

## Test du chi-deux

### Résultats

Les résultats du test du chi-carré sont les suivants :

- Générateur personnalisé : Valeur de la statistique du test = 4
- Générateur aléatoire de Python : Valeur de la statistique du test = 6.6

## Discussion

La valeur de la statistique du test obtenue pour le générateur personnalisé est de 4, tandis que le générateur aléatoire de Python a une valeur de 6.6. Cela indique que la différence entre les distributions des deux générateurs est plus grande pour le générateur aléatoire de Python.

## 3.2 ESimpleRandomGenerator

L'ESimpleRandomGenerator est un générateur de nombres aléatoires basé sur les décimales de  $e$ . Il est initialisé avec une valeur de départ (graine), la valeur de  $e$  utilisée pour générer les nombres aléatoires et la précision.

### 3.2.1 Initialisation

L'ESimpleRandomGenerator est initialisé avec les paramètres suivants :

- **seed (int)** : La valeur de départ (graine) du générateur.
- **e\_value (str)** : La valeur de  $e$  utilisée pour générer les nombres aléatoires.
- **precision (int)** : La précision du générateur, déterminant le nombre de décimales de  $e$ .

Le générateur stocke la graine initiale, la dernière valeur générée, le dernier indice calculé, le nombre de décimales à considérer à partir de  $e$  et la précision.

### 3.2.2 Comparaison avec `random.random` de Python

Ce chapitre présente une comparaison entre le générateur simple de nombres aléatoires personnalisé et le générateur aléatoire fourni par le module `random` de Python. L'objectif est d'évaluer la performance et la qualité de vos générateurs de nombres aléatoires par rapport à la référence du générateur de Python. Pour cette comparaison nous allons générer 100 nombres aléatoires entre 0 et 1 avec la seed 42 pour notre générateur personnalisé.

On va effectuer des tests statistiques pour comparer la distribution des nombres aléatoires générés par les deux générateurs :

#### Test de Kolmogorov-Smirnov

##### Résultats

Les résultats du test de Kolmogorov-Smirnov sont les suivants :

- Générateur personnalisé : Valeur de  $D = 0.0936$
- Générateur aléatoire de Python : Valeur de  $D = 0.0932$

##### Discussion

La valeur de  $D$  obtenue pour le générateur personnalisé est de 0.0936, tandis que le générateur aléatoire de Python a une valeur de  $D$  de 0.0932. Cela indique que la différence entre les distributions empiriques des deux générateurs est plus grande pour le générateur personnalisé.

#### Test du chi-deux

##### Résultats

Les résultats du test du chi-carré sont les suivants :

- Générateur personnalisé : Valeur de la statistique du test = 13
- Générateur aléatoire de Python : Valeur de la statistique du test = 13.8

##### Discussion

La valeur de la statistique du test obtenue pour le générateur personnalisé est de 13, tandis que le générateur aléatoire de Python a une valeur de 13.8. Cela indique que la différence entre les distributions des deux générateurs est plus grande pour le générateur aléatoire de Python.

## 4 Conclusion

Dans ce rapport, nous avons étudié le caractère pseudo-aléatoire des décimales de la constante mathématique  $e$  en utilisant le test du chi-deux et le test du poker. Le test du chi-deux nous a permis d'évaluer la similarité entre les fréquences observées des chiffres dans les décimales de  $e$  et les fréquences attendues. Nous avons obtenu une statistique du test du chi-deux de 8.65376, ce qui suggère une légère différence entre les fréquences observées et les fréquences attendues.

Le test du poker nous a permis d'évaluer la distribution des combinaisons de poker dans les décimales de  $e$ . Nous avons obtenu une statistique du test du poker de 2.89, ce qui indique une certaine différence entre les occurrences observées des classes de poker et les fréquences attendues.

Cependant, ces différences ne sont pas significatives lorsque nous les comparons à des seuils de signification appropriés. Pour une erreur de première espèce de 0.05, les valeurs critiques des statistiques du test du chi-deux et du test du poker sont respectivement de 16.919 et 9.488. Ainsi, nous acceptons l'hypothèse nulle selon laquelle les décimales de  $e$  sont pseudo-aléatoires.

De plus, nous avons présenté deux générateurs de nombres aléatoires basés sur les décimales de  $e$  le `LCERandomGenerator` et l'`ESimpleRandomGenerator`. Nous avons comparé leurs performances et leur qualité avec le générateur aléatoire de Python en utilisant des tests statistiques. Les résultats ont montré que les générateurs personnalisés étaient légèrement différents du générateur de Python, mais les différences étaient négligeables.

En conclusion, bien que les décimales de  $e$  ne soient pas parfaitement pseudo-aléatoires, elles présentent des caractéristiques suffisamment proches d'une distribution aléatoire pour être utilisées dans des générateurs de nombres aléatoires. Cependant, il est important de noter que ces générateurs ne doivent pas être considérés comme des générateurs cryptographiques, car leur qualité et leur sécurité ne sont pas garanties.