TPB3 - Introduction à la méthode de Monte-Carlo

Dans tout le TP, on importe les modules suivants :

```python
import numpy as np
import numpy.random as rd
import matplotlib.pyplot as plt
```

1 Première approche

Le but de ce TP est d'étudier la méthode de Monte-Carlo. C'est une technique d'estimation de certaines quantités de manière probabiliste. Avec cette méthode, on obtient donc une valeur probable (mais pas certaine et qui peut être plus ou moins fausse) de la quantité que l'on chercher à estimer.
Le principe de la méthode de Monte-Carlo est le suivant : pour évaluer une certaine quantité I, on va chercher des situations probabilistes que l'on va pouvoir simuler qui donne cette quantité. Regardons un exemple.

Exercice 1 - Premier exemple ⋆

On va estimer numériquer $I = \int_{-\infty}^{+\infty} \frac{x^2}{\sqrt{2\pi}} \mathrm{e}^{-\frac{x^2}{2}} \mathrm{d}x$.

1. On pose $f(x) = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{x^2}{2}}$. Reconnaître la densité f.
2. Soit X une v.a.r. de densité f. Exprimer I en fonction d'une espérance construire à partir de X.
3. Si $(X_n)_{n\in\mathbb{N}}$ est une suite de variables i.i.d. de même loi que X, justifier alors que $\overline{X_n} \xrightarrow{P} E(X)$.
4. On part du code Python suivant :

```python
def mc_mgauss(n):
  donnees = rd.normal(0,1,n)
  t = ...
  return np.mean(t)
```

Compléter la fonction pour qu'elle renvoie une estimation de I.

Exercice 2 - Un pi-aller ⋆⋆

On va chercher à estimer π. Travaillons dans l'univers $\Omega = [0,1]^2$. On admet que l'on peut munir Ω d'une tribu naturelle et d'une probabilité telle que la probabilité d'un événement soit proportionnelle à sa surface. On note :

$$A = \{\omega = (x, y) \in \Omega,\ x^2 + y^2 \leq 1\}.$$

1. Représenter dans le plan l'univers Ω puis l'événement A.
2. On pose la variable aléatoire $X = \mathbb{1}_A$. Rappeller la définition de $\mathbb{1}_A$.
3. En vous appuyant sur la représentation de la première question, que vaut $P(A)$? En déduire la loi de X et son espérance.
4. On part du code Python suivant :

```python
def mc_pi(n):
  donnees = np.zeros(n)
  for i in range(n):
    x = rd.random()
    y = rd.random()
    if ... :
      donnees[...] = ...
  return ...
```

(a) Quel est le rôle de `donnees = np.zeros(n)` ?
(b) Quelle loi est simulée pour `x` ?
(c) Compléter la condition et le remplissage de `donnees`.
(d) Que faut-il renvoyer à la fin de la fonction pour avoir une estimation de π ? Compléter le code en conséquence.

2 Évaluation d'une fonction de répartition

La méthode de Monte-Carlo s'applique particulièrement bien lorsque la quantité à évaluer est directement tirée de la théorie des probabilités. Nous allons dans cette partie voir un tel exemple avec la fonction de répartition de la loi normale.

Exercice 3 ⋆⋆

Soit $X \hookrightarrow \mathcal{N}(0,1)$.

1. Donner la densité continue sur $\mathbb{R}$ de X.
2. Donner une expression intégrale de la fonction de répartition Φ de X.
3. On effecture $n \in \mathbb{N}$ tirages simulant X. Soit $x \in \mathbb{R}$. Comment peut-on estimer $\Phi(x) = P(X \leq x)$?
4. On part du code suivant :

```
def mc_Phi(tableau_x,n):
  simulations = rd.normal(0,1,n)
  tableau_Phi = np.zeros(len(tableau_x))
  for i in range(len(tableau_x)):
    ...
    tableau_Phi[i] = ...
  return tableau_Phi
```

   Compléter le code en s'inspirant du TP précédent pour que `tableau_Phi[i]` contienne une estimation de $\Phi(x)$ où x désigne `tableau_x[i]`.

5. Utiliser `np.linspace` pour générer un tableau de valeurs pour x régulièrement espacée entre -5 et 5. Puis utiliser le code suivant :

```
tableau_Phi = mc_Phi(tableau_x,1000)
plt.plot(tableau_x,tableau_Phi)
plt.show()
```

   Que fait ce code ? Commenter la courbe qui s'affiche.

3 Une application

Exercice 4 - Calcul d'intégral ⋆⋆

On va désormais appliquer la méthode de Monte-Carlo à l'évaluation de :

$$I = \int_0^1 f(x)\mathrm{d}x$$

où f est une fonction continue sur $[0,1]$. Pour cela, on pose $X \hookrightarrow \mathcal{U}([0,1])$.

1. Justifier que $f(X)$ admet une espérance. Donner une expression de $E(f(X))$.
2. Appliquer ce qui a été vu jusqu'à présent pour estimer : $I = \int_0^1 \frac{\mathrm{e}^{-t}}{1+t^4}\mathrm{d}t$.