



Méthode de Monte-Carlo

Méthode de Monte-Carlo

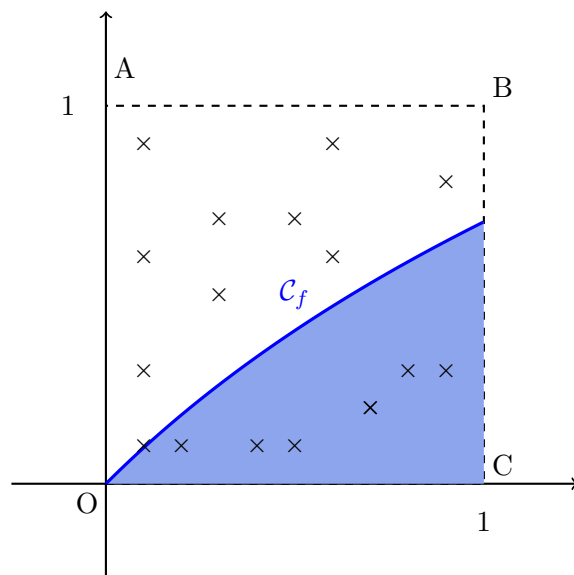
La méthode de Monte-Carlo consiste à approcher une solution d'une équation, ou toute valeur numérique, en utilisant des expériences aléatoires et donc les probabilités. Le nom de cette méthode fait référence aux jeux de hasard du quartier de Monte-Carlo (à Monaco). Cette méthode a été inventée en 1947 par Nicholas Metropolis, et publiée en 1949 dans un article scientifique avec Stanislaw Ulam lors du développement de l'arme nucléaire.

On souhaite déterminer une valeur approchée de

$$\int_0^1 \ln(x+1) dx.$$

Le principe de cette méthode, probabiliste, est le suivant :

- On génère au hasard n points dans le carré OABC.
- On compte le nombre de points k situés sous la courbe représentative de f , entre les points O et C et au dessus de l'axe des abscisses.
- La fréquence $\frac{k}{n}$ est une valeur approchée de l'intégrale recherchée.



Exercice n°1

1. Quelle fonction de python permet de générer un nombre réel compris entre 0 et 1 de façon aléatoire ?
2. Quel sera le test pour vérifier si un point se situe dans la zone voulue, c'est à dire sous la courbe représentative de f , entre les points O et C et au dessus de l'axe des abscisses.
3. Programmer cette méthode sous Python et déterminer une approximation de $\int_0^1 \ln(x+1) dx$.

> Correction des exercices

Exercice n°1

1. Il s'agit de la fonction « random ».
2. Si on note x l'abscisse du point générer aléatoirement et y son ordonnée, il faut que $y \leq f(x)$.
3. Programme Python :

```
1 from math import*
2 from random import*
3 def f(x):
4     return log(x+1)
5 def MonteCarlo(n):
6     k=0
7     for i in range(n):
8         x=random()
9         y=random()
10        if y<=f(x):
11            k=k+1
12    print(k/n)
13 MonteCarlo(1000)
```

Out[] : 0.389