

Méthodes de Monte-Carlo

Les méthodes de Monte-Carlo sont des méthodes probabilistes basées sur l'observation d'un grand nombre d'issues. Elles permettent, par le calcul de probabilités, de déterminer des aires.

Soit f une fonction définie sur $[0; 1]$ telle que pour tout $x \in [0; 1]$, $0 \leq f(x) \leq 1$.

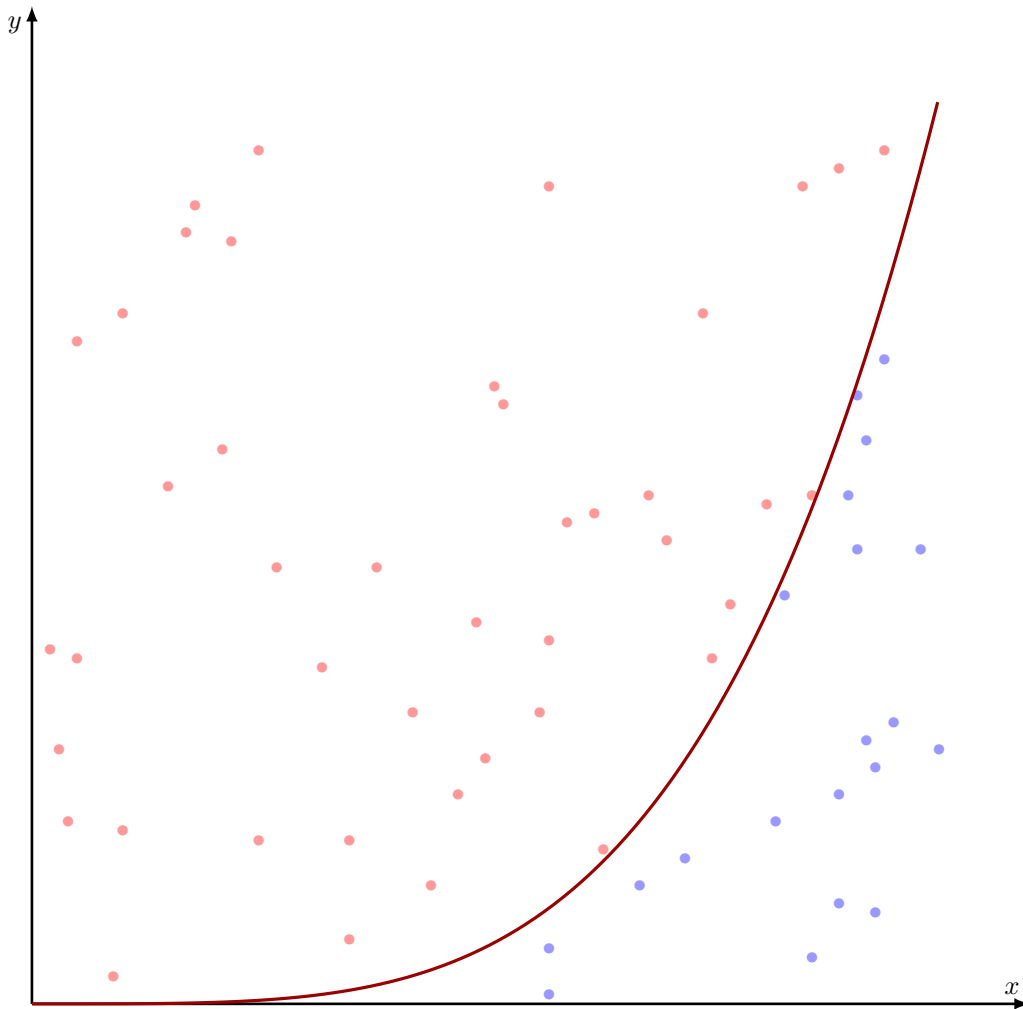
On note $A = \int_0^1 f(x) dx$.

On considère l'épreuve de Bernoulli consistant à choisir au hasard un point $(x; y)$ dans le carré de côté 1.

On appelle Succès l'événement "Le point choisi est sous la courbe représentative de f ."

On admet que la probabilité de cet événement est :

$$p = \frac{\text{aire sous la courbe}}{\text{aire du carré}} = \frac{A}{1} = A$$



1. On considère le schéma de Bernoulli consistant à répéter de façon indépendante l'épreuve précédente. Pour tout entier naturel i non nul, on note X_i la variable aléatoire qui vaut 1 en cas de succès, et 0 sinon, lors de la i -ième épreuve. Donner la loi de probabilité de chaque variable aléatoire X_i .

On note, pour $n \in \mathbb{N}^*$:

$$S_n = \sum_{i=1}^n X_i \quad \text{et} \quad M_n = \frac{S_n}{n}$$

2. Justifier que, pour tout réel $t > 0$, on a :

$$\lim_{n \rightarrow +\infty} \mathbb{P}(|M_n - A| \geq t) = 0$$

3. Compléter la fonction `Bern` ci-dessous, afin qu'elle renvoie 1 si un point de coordonnées (x, y) est sous la courbe représentative de la fonction f et 0 sinon.

```
def f(x):  
    return x**4  
  
from random import random  
  
def Bern(x,y):  
    if ....:  
        X=....  
    else :  
        X=....  
    return X
```

4. (a) Compléter la fonction `Aire` ci-dessous, afin qu'elle renvoie une valeur approchée de l'aire A .

```
def Aire(n):  
    M=0  
    for i in range(....):  
        M=....+....  
    return ....
```

- (b) Estimer, en utilisant la fonction `Aire`, la valeur de $\int_0^1 x^4 \, dx$.
(c) Vérifier le résultat obtenu par le calcul exact de l'intégrale.
5. Modifier l'un des programmes afin de déterminer une valeur approchée de l'intégrale :

$$\int_0^1 e^{-\frac{x^2}{2}} \, dx$$