

TP Monte Carlo

Au cours de ce TP, nous cherchons à estimer la valeur de π en s'appuyant sur la méthode de Monte Carlo.

Idée générale de la méthode : en plaçant des points aléatoirement dans un domaine d'aire connue, lorsque le nombre de points tend vers l'infini, la proportion de points tombés dans un sous-domaine permet de déterminer son aire.

$$\text{aire_sous_domaine} = \frac{\text{nb_points_dans_sous_domaine}}{\text{nb_points_total}} * \text{aire_domaine}$$

Cela nécessite :

- de pouvoir tirer des points aléatoirement ;
- de pouvoir compter ceux ayant atterri dans le sous domaine.

Solution étudiée : Dans un repère orthonormé :

- l'aire du carré de côté 1 vaut 1 ;
- le quart de cercle de côté 1 inclus dans ce carré a pour aire $\pi / 4$.

1. Ecrivez une fonction de tirage aléatoire de points dans un rectangle. Par défaut, si les coordonnées du rectangle ne sont pas renseignées, elles devront avoir pour valeur $x_{\min}=0$, $x_{\max}=1$, $y_{\min}=0$ et $y_{\max}=1$.
2. Ajoutez une fonction permettant de tirer n points aléatoires et de retourner leur coordonnées dans une liste. Testez votre fonction.
3. Quelle condition permet de déterminer si un point (x,y) est à l'intérieur du cercle de centre (x_0,y_0) et de rayon r ?
4. Ecrivez une fonction `est_dans_cercle(x, y, x0, y0, r)` retournant le booléen `True` si le point (x,y) est dans le cercle de centre (x_0,y_0) et de rayon r , et `False` sinon.
5. Ajoutez les valeurs par défaut suivantes à votre fonction : $x_0=0$, $y_0=0$ et $r=1$
6. Pour revenir à notre problème d'estimation de π , prévoyez une fonction `nb_dans_sous_domaine(points)` prenant en paramètre une liste de points et retournant le nombre à l'intérieur du quart de cercle de rayon 1 centré sur un coin du carré.
7. Utilisez votre dernière fonction pour estimer π en tirant 10, 100, 1000, ... 10000000 points.
8. Quels sont selon vous les points forts / points faibles de la méthode ?

Si vous avez terminé les questions précédentes, essayez de représenter graphiquement les tirages de points aléatoires à l'aide de la librairie `matplotlib`.

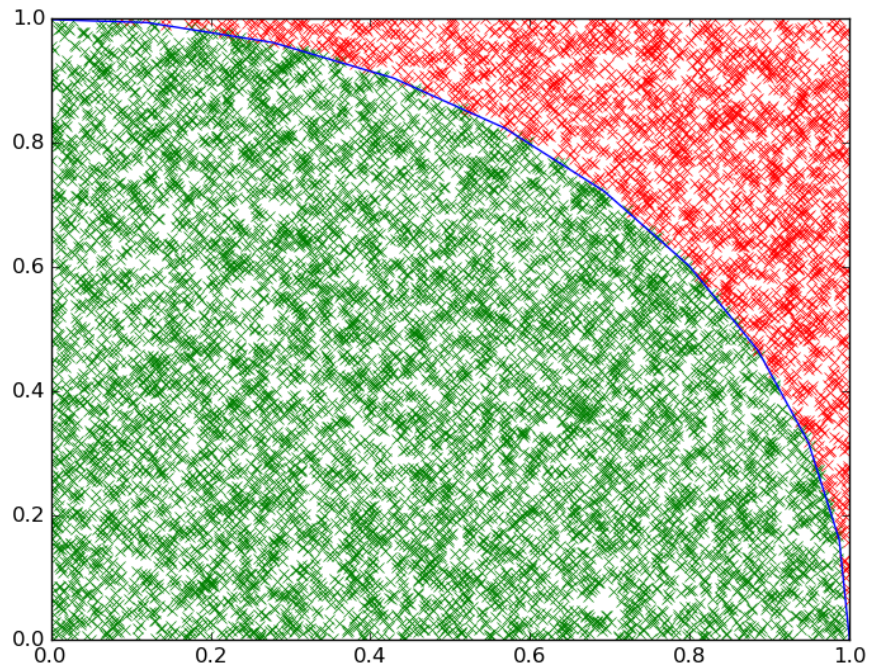


FIGURE 1 – Estimation de Pi à l'aide de la méthode de Monte Carlo

Les instructions suivantes pourront être utiles :

- `import numpy as np` et `import matplotlib.pyplot as plt` pour importer les modules numpy et matplotlib
- `plt.plot(x, y, 'rx')` pour afficher une croix rouge sous le point (x, y)
- `plt.show()` pour afficher un graphique matplotlib
- `np.linspace(bi, bf, n)` pour découper un intervalle [bi, bf] en n éléments de même taille