Master Maths en Action
–
Méthodes de Monte-Carlo – TD n°1

Exercice 1

On se propose de calculer l'intégrale

$$\int_0^1 \cos\left(x^3\right)\exp\left(-x\right)dx.$$

1. Proposer une méthode de Monte-Carlo basée sur des tirages de la loi uniforme sur $]0,1[$.
2. Programmer une fonction en `Matlab` réalisant cette méthode :

   ```
   function [I] = MonIntegrale1(N)
   ```

   où N est le nombre de tirages à effectuer. On utilisera la fonction `rand`.
3. Réaliser un script `Matlab` permettant de faire varier N et visualiser la convergence de la méthode.
4. Transformer l'intégrale initiale en une intégrale sur $\mathbb{R}$ et faire ainsi apparaître la densité de probabilité d'une loi classique.
5. En déduire une méthode de Monte-Carlo pour cette nouvelle formulation.
6. Programmer une fonction en `Matlab` réalisant cette nouvelle méthode :

   ```
   function [I] = MonIntegrale2(N)
   ```
7. Programmer un script en `Matlab` comparant visuellement la convergence des deux méthodes en fonction de N.

Exercice 2

On considère ici le carré $C =]-1,1[^2$ dans $\mathbb{R}^2$ et le disque unité $D \subset C$. On rappelle que si X_1 et X_2 suivent une loi uniforme sur $]-1,1[$ alors $X = (X_1, X_2)$ suit une loi uniforme sur C.

1. Vérifier que $\mathbb{E}(\mathbf{1}_{X\in D}) = \pi/4$.
2. Programmer une méthode de Monte-Carlo calculant une approximation de cette valeur :

   ```
   function [I] = ApproxPisur4(N)
   ```

   où `N` sera le nombre de tirages à effectuer et `I` la valeur approchée obtenue.
3. Modifier la fonction précédente pour obtenir une fonction estimant la variance de la méthode selon la formule du cours :

   ```
   function [I, sigma2] = ApproxPisur4bis(N)
   ```

   où `sigma2` est l'estimateur de la variance.
4. Pour finir, modifier encore la fonction pour qu'elle choisisse elle-même le nombre de tirages N (ici N sera pair) de sorte à avoir une estimation de l'espérance prcise à $\delta = 0.01$ près avec une probabilité $1-\alpha = 0.95$.

   ```
   function [I,sigma2,N] = ApproxPisur4ter
   ```

   où `N` est le nombre de tirages qui aura été fait.
5. Généraliser encore la fonction à $\delta > 0$ et $1-\alpha$ quelconques pour obtenir la fonction :

   ```
   function [I,N] = ApproxPisur4Optim(delta,alpha)
   ```