MÉTHODES DE MONTE-CARLO
EXERCICES DU 9 FÉVRIER 2015

par Rémi Peyre

EXERCICE 1 — Mathématiques expérimentales

Soit P la loi normale sur $\mathbb{R}$ de moyenne 1 *et d'écart-type* 1.

1. Écrire une fonction `simulation` qui simule une réalisation de la loi P ; en d'autres termes, cette fonction ne prendra aucun argument et renverra une valeur x réelle suivant la loi P, la valeur de x retournée étant indépendante à chaque appel de la fonction.

Indication : Les lois normales peuvent être simulées à l'aide de la fonction `randn` de Matlab. Consulter « `doc randn` » pour plus de détails.

2. Pour X suivant la loi P, créer un programme `MC_simple.m` évaluant $\mathbb{E}(X^4)$ par la méthode de Monte-Carlo, pour 5 000 000 simulations.

3. Pour X suivant la loi P, justifier que $\mathbb{E}(X^8) < \infty$.

4. Créer une variante du programme `MC_simple.m`, appelée `MC_IC.m` qui, outre l'estimation de $\mathbb{E}(X^4)$, calcule et affiche un intervalle de confiance à 98 %. Conjecturer une valeur exacte pour la valeur à estimer.

On se propose maintenant de tester l'affirmation du cours selon laquelle, notant t le cout de calcul (en l'occurrence assimilé au temps de calcul), la largeur de l'intervalle de confiance décroit comme $t^{-1/2}$.

5. Créer une variante `investigation_MC` de la fonction `MC_IC` qui fera les mêmes calculs, mais prendra en argument le nombre N de simulations demandées, n'affichera rien, et renverra en sortie, d'une part l'estimateur m de $\mathbb{E}(X^4)$, et d'autre part la valeur (estimée) σ de l'écart-type de l'estimateur (qui est par construction directement proportionnelle à la largeur de l'intervalle de confiance), ainsi que le temps t de calcul. Appliquer `MC_IC` pour des valeurs de N égales à $10^6, 2\times10^6, 3\times10^6, ..., 10^7$ (on pourra créer un vecteur-ligne `valeurs_N` recensant toutes ces valeurs), et stocker les résultats obtenus (dans des vecteurs-lignes `valeurs_m`, `valeurs_sigma` et `valeurs_t`). Tracer t en fonction de N, m en fonction de t, σ^{-2} en fonction de t [*]. Les affirmations du cours sont-elles vérifiées sur cet exemple ?

Indication : La commande `plot` de Matlab permet de tracer des graphiques.

Indication : Il peut être utile d'exiger que le bas bord gauche de la fenêtre corresponde aux coordonnées nulles. Une manière de s'assurer cela est d'entrer, juste après avoir exécuté la commande `plot` : « `xlim_ = xlim; xlim ([0, xlim_(2)]); ylim_ = ylim; ylim ([0, ylim_(2)]);` ».

Maintenant nous allons expérimenter quelques aspects de la génération pseudo-aléatoire de Matlab.

6. Lancer deux fois de suite le programme `MC_IC`. Les deux résultats sont-ils identiques ? Pouvait-on le prévoir ? Comparer la largeur des deux intervalles de confiance : qu'observez-vous ? Pouvait-on le prévoir ?

7. Éteindre Matlab ; le rallumer ; lancer la fonction `MC_simple` ; noter le résultat sur une feuille ; rééteindre Matlab ; recommencer. Que constate-t-on ? Comment l'expliquez-vous ? Quels avantages et inconvénients voyez-vous à ce phénomène ?

[*]. Ayez l'obligeance de tracer juste un nuage de points, sans les relier entre eux... !

Nous reprenons maintenant notre objectif de tester numériquement notre conjecture sur la valeur exacte de $\mathbb{E}(X^4)$.

8. À l'aide des résultats de la question 5, évaluer la valeur de N pour laquelle votre programme `MC_IC` mettrait environ deux minutes à tourner.

9. Comment pourrait-on employer que la classe dispose de 14 ordinateurs pour améliorer la précision du calcul de Monte-Carlo tout en respectant la contrainte des deux minutes ? En quoi les constatations de la question 7 vont-elles poser souci ? Comment y remédier ?
Indication : Il est vivement conseillé de se renseigner sur la commande `rng` de Matlab pour répondre à la dernière interrogation...

10. Mettre en application la méthode mise au point lors de la dernière question. La conjecture s'en trouve-t-elle renforcée ?
Indication : Pour obtenir de Matlab un affichage précis, il faut saisir la commande « `format('long');` » dans la console ; quant on utilise la fonction `fprintf`, il faut remplacer le code de format « `%f` » par « `%.15f` ».

11. **(★)** Démontrer la conjecture.

EXERCICE 2 — Probabilité d'un brelan

Au poker (variante dite « fermée »), le joueur reçoit 5 cartes au sein d'un sabot contenant 4 cartes de chacun des 13 « hauteurs » possibles. Puis il décide d'échanger entre 0 et 3 cartes en vue d'améliorer sa main. Une combinaison intéressante est d'obtenir un brelan, c'est-à-dire (au moins) trois cartes de même hauteur.

Question. Évaluer la probabilité qu'un joueur dont le seul but est d'obtenir un brelan atteigne son objectif. On exprimera le résultat sous la forme d'un intervalle de confiance à 95 %.
Indication : On prendra soin de prendre garde à un détail : si le joueur se voit servir une double paire, doit-il ne rejeter que la carte non appariée, ou aussi rejeter une des ses deux paires ?

EXERCICE 3 — Volume d'une boule

1. Évaluer le volume de la boule unité de dimension 6 [†] par la méthode de Monte-Carlo (en prenant la loi d'échantillonnage la plus évidente), en précisant un intervalle de confiance à 90 %. Comparer avec la valeur théorique, égale à $\pi^3/6$.

On veut maintenant utiliser la méthode de Monte-Carlo avec une autre loi d'échantillonnage. Soit Q *la loi sur* $\mathbb{R}^6$ *caractérisée par la densité*

$$\frac{dQ}{dx}(x) = Z^{-1} e^{-4\|x\|_{\ell^1}}, \tag{1}$$

où $\|x\|_{\ell_1} := |x_1| + |x_2| + \cdots + |x_6|$, *et où* Z *est la constante qui fait de* (1) *une densité de probabilité.*

2. Écrire une fonction simulant Q.

3. Calculer la valeur de Z.

4. Reprendre la question 1 en utilisant Q comme loi d'échantillonnage.

5. Comparer l'efficacité de la méthode de Monte-Carlo selon qu'on utilise la méthode de base où l'échantillonnage selon Q.

[†]. Càd. l'ensemble des points de $\mathbb{R}^6$ dont la distance euclidienne à l'origine est inférieure ou égale à 1.