IFT-6561: SIMULATION **Automne 2020**

Prof. Pierre L'Ecuyer

DEVOIR 4

Devoir à remettre le *mercredi 11 novembre 2020, 13h30, avant le début du cours*, par courriel au professeur.

Les devoirs doivent être faits individuellement ou en équipe de deux. Il est très important de bien expliquer tout ce que vous faites et d'expliquer le sens de vos résultats (discussion), pour les expériences numériques et aussi pour les preuves. Dans la correction, on accordera davantage d'importance à la clarté des explications qu'aux programmes informatiques et aux résultats comme tels. Dans ce sens, remettre uniquement un programme qui donne les bons résultats ne suffit pas et vous donnera peu ou pas de points (nous ne corrigeons pas les programmes comme tels). Vous devez quand même envoyer une copie de vos programmes par courriel avec votre devoir, pour que le professeur puisse les vérifier au besoin.

Attention au plagiat: il n'est pas permis de copier et/ou modifier les programmes ou les solutions venant de quelqu'un d'autre.

---

1. (10 points)

Exercice 2.23 (a, b, c).

2. (12 points)

Considérons un processus variance-gamma (VG) $Y = \{Y(t),\, t \geq 0\}$ défini par $Y(t) = X(G(t))$ où le subordinateur G est un processus gamma de paramètres 1 et $\nu = 0.3$ (l'accroissement sur un intervalle de longueur t a une moyenne de t et une variance de $\nu t = 0.3t$) et $X = \{X(t),\, t \geq 0\}$ est un mouvement Brownien de paramètres $\theta = \mu = -0.1436$ et $\sigma = 0.12136$. On a $G(0) = X(0) = 0$. On veut générer les valeurs du processus Y aux instants $t_1 = 0.6$, $t_2 = 0.8$, $t_3 = 1.0$, et $t_4 = 1.2$, puis calculer $S = Y(t_1) + Y(t_2) + Y(t_3) + Y(t_4)$. On va répéter cela $n = 10^4$ fois pour pouvoir calculer un intervalle de confiance à 95% sur $\mathbb{E}[S]$.

Donnez un algorithme (détaillé) qui fait cela pour chacune des trois méthodes BGSS, BGBS, et DGBS, vues en classes et décrites dans les notes (voir l'exemple 6.52, page 495).

Pour BGSS, on génère $\tau_1 = G(t_1)$, $Y(t_1) = X(\tau_1)$, $\tau_2 = G(t_2)$, $Y(t_2) = X(\tau_2)$, $\tau_3 = G(t_3)$, $Y(t_3) = X(\tau_3)$, $\tau_4 = G(t_4)$, $Y(t_4) = X(\tau_4)$, dans cet ordre.

Pour BGBS, on génère $\tau_4 = G(t_4)$, $Y(t_4) = X(\tau_4)$, $\tau_2 = G(t_2)$, $Y(t_2) = X(\tau_2)$, $\tau_1 = G(t_1)$, $Y(t_1) = X(\tau_1)$, $\tau_3 = G(t_3)$, $Y(t_3) = X(\tau_3)$, dans cet ordre.

Pour DGBS, on génère $G^+(t_4)$, $G^-(t_4)$, $G^+(t_2)$, $G^-(t_2)$, $G^+(t_1)$, $G^-(t_1)$, $G^+(t_3)$, $G^-(t_3)$, dans cet ordre.

Implantez ces trois algorithmes et utilisez votre programme pour calculer un intervalle de confiance sur $\mathbb{E}[S]$, pour les valeurs données ci haut. Comparez les trois méthodes.

Pour cet exercice, vous *ne pouvez pas utiliser* la classe `stochprocess.VarianceGammaProcess` de SSJ ou ses sous-classes, car ce serait trop simple. Vous devez plutôt programmer vous-mêmes vos algorithmes. L'objectif est de comprendre comment générer des trajectoires pour ce processus, et non pas seulement appeler une fonction.

3. (12 points)

Supposons que l'on veut générer un vecteur (X, Y) selon la loi binormale standard, tronquée sur le rectangle $[0, a] \times [0, b]$, où a et b sont des constantes positives, par une méthode de rejet. Pour cela, on peut borner la densité conjointe $f(x, y)$ de (X, Y) sur ce rectangle par une constante c, générer des points (X, Y, Z) au hasard uniformément dans la boite rectangulaire $[0, a] \times [0, b] \times [0, c]$, accepter le premier pour lequel $Z \le f(X, Y)$, et retourner le (X, Y) correspondant.

(a) Calculez la plus petite valeur de c pour laquelle $f(x, y) \le c$ sur ce rectangle.

(b) Donner l'algorithme pour votre méthode de rejet et écrivez un programme qui implante cette méthode pour des valeurs générales de a et b.

(c) Quel est le nombre espéré de rejets avant d'accepter le premier (X, Y), en fonction de a et b seulement (donnez une valeur précise).

(d) Avec votre programme, générez $n = 1000$ paires (X, Y) selon la loi tronquée, pour $a = 4$ et $b = 5$. Faites un "scatter plot" de ces points (X, Y).

(e) Si on générait directement X selon la loi normale tronquée sur $[0, a]$ et Y selon la loi normale tronquée sur $[0, b]$, est-ce que (X, Y) suivrait la bonne loi tronquée sur $[0, a] \times [0, b]$? Prouvez votre réponse.

4. (16 points)

Soit $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ un graphe connexe non orienté ayant un nombre fini de sommets. Un k-*coloriage* du graphe est un vecteur qui donne une couleur à chaque sommet du graphe, en utilisant au plus k couleurs. Pour un graphe de d sommets, il y a k^d possibilités. Un k-coloriage est dit *admissible* s'il n'y a pas deux sommets adjacents qui ont la même couleur. Pour un graphe quelconque, trouver le plus petit k pour lequel il existe un k-coloriage admissible, ou calculer le nombre de k-coloriages admissibles, ou encore générer au hasard un k-coloriage selon la loi uniforme sur l'ensemble $\mathcal{C}$ de tous les coloriages admissibles, sont des problèmes connus comme très difficiles, et qui ont beaucoup d'applications.

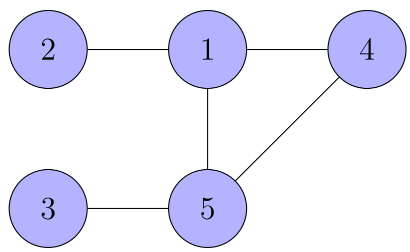
Pour générer des k-coloriages approximativement selon la loi uniforme sur $\mathcal{C}$, pour un k fixé, on peut penser à utiliser MCMC. L'idée est de construire une chaîne de Markov dont l'état est un élément de $\mathcal{C}$ et les probabilités d'état stationnaire correspondent à la loi uniforme sur l'ensemble $\mathcal{C}$.

(a) Expliquez comment on peut faire cela via l'échantillonnage de Gibbs selon chacune des deux approches suivantes: (1) en ré-échantillonnant la couleur d'un seul sommet choisi au hasard à chaque itération; et (2) avec balayage systématique, en ré-échantillonnant à chaque itération la couleur de chacun des sommets du graphe, un sommet à la fois, disons toujours dans le même ordre. Comparez les deux approches, en discutant les avantages et les inconvénients de chacune.

(b) Si k est trop petit, il se peut qu'il n'existe pas de k-coloriage admissible, ou bien encore qu'il en existe, mais que la chaîne construite via Gibbs ne soit pas irréductible sur $\mathcal{C}$, de sorte qu'on ne puisse pas atteindre tous les k-coloriages admissibles à partir d'un k-coloriage initial. Pour voir cela, considérez un graphe en triangle (3 sommets et 3 arêtes). Expliquez ce qui se passe avec l'échantillonnage de Gibbs si $k = 2$, puis si $k = 3$, puis si $k = 4$.

Si Δ est le degré maximum d'un sommet du graphe, on peut prouver que si $k \geq \Delta + 2$, alors la chaîne construite par Gibbs est irréductible sur $\mathcal{C}$, et donc la loi stationnaire est uniforme sur $\mathcal{C}$. Par contre l'inverse n'est peut-être pas toujours vrai.

(c) Considérons le petit graphe ci-bas, repris de mes diapos. Pour $k = 3$, combien y a-t-il d'éléments dans $\mathcal{C}$ pour ce graphe? Est-ce que Gibbs donne une chaîne irréductible sur $\mathcal{C}$? Mêmes questions pour $k = 4$. Bien sûr, il faut prouver vos réponses.



(d) Pour tirer au hasard un coloriage selon la loi uniforme dans $\mathcal{C}$, on pourrait penser à utiliser une méthode de rejet très simple: on génère au hasard un coloriage sur l'ensemble de toutes les d^k possibilités; s'il est admissible on le retient, sinon on recommence. Dans l'exemple en (c), pour $k = 4$, quelle est la probabilité d'accepter le coloriage? Et pour un graphe de $d = 1000$ sommets, disons dont les degrés vont de 2 à 6, et avec 8 couleurs, que devrait-t-il se passer avec cet algorithme de rejet?

(e) Implantez la méthode de Gibbs avec balayage systématique pour le graphe donné en (c), pour $k = 4$. Faites $n = 216,000$ itérations de la méthode et comptez le nombre de fois que chaque k-coloriage de $\mathcal{C}$ est visité. Vous avez calculé en (c) le nombre $|\mathcal{C}|$ de coloriages admissibles pour $k = 4$. Si la chaine est irréductible, sa loi stationnaire devrait être uniforme sur ces $|\mathcal{C}|$ coloriages, donc chacun devrait apparaître à peu près le même nombre de fois. Est-ce que vos résultats sont en accord avec cela?

Une façon simple de tester si c'est en accord est de faire un test de chi-deux sur les résultats, pour tester l'hypothèse que la distribution est uniforme sur l'ensemble des coloriages admissibles. Calculez le chi-deux, puis la p-valeur du test. Discutez vos résultats.

Rappel: S'il y a κ possibilités, et si N_j est le nombre observé d'observations et e_j le nombre espéré d'observations pour la possibilité j, pour $j = 1, \ldots, \kappa$, alors le chi-deux est défini par

$$\chi^2 = \sum_{j=1}^{\kappa} \frac{(N_j - e_j)^2}{e_j}$$

et cette variable aléatoire suit approximativement la loi de chi-deux avec $\kappa - 1$ degrés de liberté. Si on observe $\chi^2 = x$, alors la p-valeur du test est $p = \mathbb{P}[X > x]$ où X est une variable aléatoire qui suit la loi du chi-deux à $\kappa - 1$ degrés de liberté. Lorsque p est trop petit, on peut mettre en doûte l'hypothèse. Dans ce cas-ci, on a $\kappa = |\mathcal{C}|$ et $e_j = n/|\mathcal{C}|$ pour tout j.

Formellement, le test du chi-deux est valide sous l'hypothèse que les n observations sont indépendantes. Ici, elles ne le sont pas, car ce sont les états visités par une chaine de Markov, mais la loi du chi-deux est quand même une bonne approximation quand n est grand, et le test peut détecter si la loi dévie beaucoup de l'uniformité, par exemple.

Pour plus de détails sur cette classe de problèmes, vous pouvez consulter: cet article.
