IFT-6561 **SIMULATION** **Automne 2014**

Prof. Pierre L'Ecuyer

Travail de fin de session

Travail final, à remettre au plus tard le **vendredi 19 décembre 2014 à midi**, dans le casier du professeur au secrétariat du DIRO (attention, le secrétariat ferme souvent un peu avant midi) ou à l'assistant du cours Tien Mai, local AA-3375. Vous devez remettre une copie imprimée. Le jour de la remise, envoyez aussi une copie de vos programmes par courriel au correcteur, pour qu'il puisse les regarder ou essayer si nécessaire: `maianhti@iro.umontreal.ca`. Ceci dit, ce sont surtout la clarté et l'exhaustivité de vos explications qui seront notées, pas les programmes. Le nombre de points accordé à chaque question est indiqué entre parenthèses. Attention au plagiat: il n'est pas permis de copier et/ou modifier les programmes ou les solutions d'un(e) autre étudiant(e).

— 1 —

(50 points)

(a) Dans l'exemple du réseau d'activités stochastique des notes de cours (Voir les exemples 1.4, 1.20, 1.47, 1.49, 1.50, 6.3, 6.21, and 6.36), combinez la méthode CMC utilisée dans l'exemple 6.21 avec l'utilisation de variables de contrôle, comme suit. Prenez chaque $Y_j \in \mathcal{B}$ comme variable de contrôle, pour estimer $\mu = \mu_2 = \mathbb{P}[T > x]$ pour $x = 90$. Prenez $n = 10,000$. Estimez le vecteur $\boldsymbol{\beta}^*$ des huit coefficients optimaux sans faire d'expérience pilote, estimez la variance de l'estimateur de μ, et calculez un intervalle de confiance pour μ, avec et sans les variables de contrôle. Discutez et comparez.

Pour les opérations matricielles servant à calculer votre estimateur de $\boldsymbol{\beta}^*$, vous pouvez utiliser `cern.colt.matrix.linalg.Algebra` dans la librairie Colt (voir le site web de SSJ). Le programme `CallCenterWithCV.java` sur la page du cours est un exemple de programme utilisant cette librairie (dans la méthode `ApplyCV`). Il existe aussi plusieurs autres librairies d'algèbre linéaire en Java (ou autre) que vous pouvez utiliser.

(b) Pour le même modèle, dans l'exemple 1.47 on a vu comment estimer la sensibilité de $\mu_1 = \mathbb{E}[T]$ par rapport à un changement dans la valeur des paramètres θ_2 et θ_4 (les moyennes) des lois de probabilité des durées des activités 2 et 4, en utilisant des variables aléatoires communes (VAC) avec T comme estimateur de base de $\mathbb{E}[T]$. Supposons maintenant que l'on veut estimer la *dérivée* de $\mathbb{E}[T]$ par rapport à chacun de ces deux paramètres θ_2 et θ_4. Expliquez comment faire cela en utilisant les dérivées stochastiques de l'estimateur $X = T$ et prouvez que vos estimateurs sont sans biais. Pour cela, vous pouvez utiliser (si vous voulez) le corollaire 6.6 des notes.

(c) Supposons maintenant que l'on veut estimer les dérivées de $\mu_2 = \mathbb{P}[T > x]$ par rapport aux deux mêmes paramètres. Expliquez pourquoi on ne peut pas utiliser la dérivée stochastique combinée avec l'estimateur de base $X = \mathbb{I}[T > x]$ dans ce cas.

(d) Montrez que l'on peut par contre utiliser la dérivée stochastique combinée avec l'estimateur CMC de l'exemple 6.21 pour estimer les deux dérivées en (c). Prouvez que cet estimateur est sans biais.

(e) Implantez les estimateurs en (b) et (d), essayez-les avec $x = 90$ et $n = 10,000$, estimez la variance, calculez des intervalles de confiance à 95% sur les dérivés, et discutez vos résultats.

(f) Supposons maintenant que l'on veut estimer la dérivée de $F(x) = \mathbb{P}[T \le x]$ par rapport à x, qui est en fait la fonction de densité de T évaluée à x, disons $f(x)$. Expliquez comment faire cela en utilisant CMC et la dérivée stochastique. Expliquez aussi comment on peut estimer cette densité $f(x)$ en plusieurs valeurs de x simultanément en simulant le réseau n fois et en utilisant *les mêmes n simulations pour toutes les valeurs de x*. Implantez ceci pour $\{x = x_j = x_0 + j\epsilon \text{ pour } j = 0, 1, \ldots, k\}$ avec $x_0 = 80$, $\epsilon = 1$, $k = 20$ et $n = 10,000$. Faites un graphique de la densité $f(x)$ estimée en fonction de x. Comparez avec l'histogramme de la figure 1.4 des notes de cours.

(g) Refaites (f), mais en utilisant l'estimateur CMC combiné avec RQMC. Pour RQMC, prenez un ensemble de $n = 2^{14}$ points de Sobol' en $s = 8$ dimensions, randomisés par un décalage aléatoire digital. Discutez le gain d'efficacité obtenu en remplaçant les nombres aléatoires par les points RQMC.

— 2 —

(50 points)

Vous allez simuler le même modèle VG qu'à l'exemple 6.52 des notes de cours (voir aussi la section 2.16.6), avec les mêmes valeurs des paramètres. On a aussi $t_j = jT/d$ pour $j = 0, \ldots, d$, et $d = 16$. SSJ fournit les outils requis pour simuler de tels processus. Vous pouvez utiliser les valeurs exactes données dans l'exemple pour vérifier votre programme.

Pour générer les processus gamma via "bridge sampling", utilisez la classe `GammaProcessSymmetricalBridge`, qui est beaucoup plus rapide que `GammaProcessBridge`.

(a) Écrivez un programme permettant de reproduire à peu près les résultats du tableau 6.12 des notes et refaites les expériences pour la colonne $n = 2^{14}$, avec $m = 32$ répétitions, pour les quatre méthodes RQMC indiqués dans le tableau. Bien sûr, vous n'obtiendrez pas exactement les mêmes facteurs, car ces valeurs sont des estimations bruitées.

(b) Supposons maintenant que $d = 1$, de sorte que $\bar{S} = S(t_1) = S(1)$. Supposons aussi que $K \gg 100$, de sorte que le "payoff" de l'option sera rarement positif. On voudra alors utiliser l'importance sampling (IS). Si on simule le processus par DGBS, il semble raisonnable d'augmenter la moyenne du processus G^+ et de diminuer celle du processus G^-, de manière à augmenter la valeur de $G^+(1) - G^-(1)$.

Pour cela, on peut utiliser la stratégie heuristique suivante. On applique une torsion exponentielle ("exponential twisting") à la densité de chacune des deux v.a. gamma: on multiplie la densité $g^+(x)$ de $G^+(1)$ par $e^{\theta x}$ et la densité $g^-(y)$ de $G^-(1)$ par $e^{-\theta y}$, pour une constante $\theta \geq 0$ qui reste à choisir, puis on normalise les densités.

Pour $\theta \geq 0$ donné, quelles seront les nouvelles densités? Comment peut-on générer des variables aléatoires selon ces nouvelles densités? Et quel sera le rapport de vraisemblance associé?

(c) Notons que l'option commence à payer lorsque $S(1) \geq K$, i.e., lorsque $r + \omega + G^+(1) - G^-(1) = \ln[S(1)/S(0)] \geq \ln(K/S(0))$. Il apparait alors raisonnable de choisir θ tel que l'on ait $\mathbb{E}[G^+(1) - G^-(1)] = \ln(K/S(0)) - r - \omega$ sous les nouvelles densités. En écrivant l'espérance en fonction de θ, obtenez une fonction monotone de θ dont la racine est précisément cette valeur de θ. On pourra alors trouver cette racine en utilisant une méthode telle que celle de Brent-Dekker, disponible dans SSJ et ailleurs. Implantez cette stratégie IS, et utilisez-là pour estimer la valeur de l'option avec une erreur relative d'au plus 1% pour $K = 140$ et pour $K = 180$. Comparez la valeur et la variance de votre estimateur IS avec celui obtenu sans IS, dans les deux cas.

(d) Une seconde approche pourrait consister à trouver θ qui minimise la valeur maximale du rapport de vraisemblance lorsque l'estimateur est non négatif, qui est en fait atteinte lorsque $S(1) = K$. Cela minimisera la borne sur la variance donnée par la proposition 6.26 des notes. Trouvez une expression pour ce θ et comparez-le à celui utilisé en (c).

(e) On peut essayer d'améliorer l'estimateur défini en (c) en s'inspirant de l'exemple 1.42, comme suit. On génère d'abord $G^-(1)$ via IS avec le θ choisi. Ensuite, on génère $G^+(1)$ selon sa loi gamma originale, mais conditionnelle à ce que $G^+(1) \geq G^-(1) + \ln(K/S(0)) - r - \omega$ (une loi gamma tronquée). De cette manière, le revenu observé ne sera jamais nul. Implantez cet algorithme, faites le même type d'expérience qu'en (c), et comparez les résultats. Essayez ensuite de l'améliorer davantage en essayant d'optimiser θ empiriquement.

(f) Combinez votre estimateur IS obtenu en (e) avec une méthode RQMC qui utilise un "Sobol' net" constitué des $n = 2^{14}$ premiers points de la suite de Sobol', randomisé par un "left matrix scramble" suivi d'un décalage aléatoire digital (`LMScrambleShift` dans SSJ). Estimez la valeur de l'option avec ce nouvel estimateur pour les valeurs de K données en (c), donnez un intervalle de confiance approximatif, et estimez aussi le facteur additionnel de réduction de variance obtenu en appliquant IS avec RQMC par rapport à celui où on utilise IS seulement.