IEOR 4703 - Monte Carlo Simulation
Solutions for Assignment 3

Enrique Lelo de Larrea

Spring 2018

Problem 1. (a) Observe that $\theta = \mathrm{E}(\mathbb{1}(X_1+X_2 > 8))$, where $X_1 \sim \mathrm{Exp}(2)$ and $X_2 \sim \mathrm{Exp}(1)$. Throughout this problem we adopt the convention that the parameter λ of the exponential distribution corresponds to the rate and not the mean. That is, if $X \sim \mathrm{Exp}(\lambda)$ then $\mathrm{E}(X) = \frac{1}{\lambda}$ and $\mathrm{Var}(X) = \frac{1}{\lambda^2}$. The standard simulation algorithm is the following:

Algorithm 1 Simple Monte Carlo

```
for i = 1 to n do
  generate U_1, U_2 ~ U(0,1)
  set X_1 = -log(U_1)/2 and X_2 = -log(U_2)
  if X_1 + X_2 > 8 then
    set θ_i = 1
  else
    set θ_i = 0
  end if
end for
set θ̂_n = Σ_{i=1}^n θ_i/n
set σ̂_n^2 = Σ_{i=1}^n (θ_i − θ̂_n)^2/(n − 1)
set CI_{1−α} = θ̂_n ± z_{1−α/2} × σ̂_n/√n
return θ̂_n and CI_{1−α}
```

(b) To use conditional Monte Carlo, it is possible to condition on any of the two X_i. However, we will condition on X_1 since it has lower variance than X_2. We then have

$$\begin{aligned}
\theta &= \mathrm{E}(\mathbb{1}(X_1 + X_2 > 8)) \\
&= \mathrm{E}(\mathrm{E}(\mathbb{1}(X_2 > 8 - X_1) \mid X_1)) \\
&= \mathrm{E}(\mathbb{1}(X_1 > 8) + \mathbb{1}(X_1 \le 8)\mathrm{e}^{-(8-X_1)}).
\end{aligned}$$

The conditional Monte Carlo algorithm is:

Algorithm 2 Conditional Monte Carlo

for $i = 1$ **to** n **do**
  generate $U \sim \mathrm{U}(0,1)$
  set $X_1 = -\log(U)/2$
  if $X_1 > 8$ **then**
    set $\theta_i = 1$
  else
    set $\theta_i = \mathrm{e}^{-(8-X_1)}$
  end if
end for
set $\hat{\theta}_n^{CM} = \sum_{i=1}^n \theta_i/n$
set $\hat{\sigma}_n^2 = \sum_{i=1}^n (\theta_i - \hat{\theta}_n^{CM})^2/(n-1)$
set $CI_{1-\alpha} = \hat{\theta}_n^{CM} \pm z_{1-\alpha/2} \times \hat{\sigma}_n/\sqrt{n}$
return $\hat{\theta}_n^{CM}$ and $CI_{1-\alpha}$

(c) Both θ_n and θ_n^{CM} should be close to the true value of $\theta = 2\mathrm{e}^{-8} - \mathrm{e}^{-16} \approx 0.00067$ and the confidence interval with conditional Monte Carlo should be considerably smaller than the one with simple Monte Carlo.

Problem 2. (a) Observe that for each path we need to simulate $S_{\frac{T}{2}}$ and then S_T. The algorithm to price the option using standard Monte Carlo is the following:

Algorithm 3 Option price with simple Monte Carlo

for $i = 1$ **to** n **do**
  generate $Z_1, Z_2 \sim \mathrm{N}(0,1)$
  set $S_{\frac{T}{2}} = S_0 \mathrm{e}^{(r-q-\frac{\sigma^2}{2})\frac{T}{2}+\sigma\sqrt{\frac{T}{2}}Z_1}$
  set $S_T = S_{\frac{T}{2}} \mathrm{e}^{(r-q-\frac{\sigma^2}{2})\frac{T}{2}+\sigma\sqrt{\frac{T}{2}}Z_2}$
  if $S_{\frac{T}{2}} < H$ **then**
    set $\theta_i = \mathrm{e}^{-rT}(\lambda_1 S_{\frac{T}{2}} - S_T)^+$
  else
    set $\theta_i = \mathrm{e}^{-rT}(S_T - \lambda_2 S_{\frac{T}{2}})^+$
  end if
end for
set $\hat{\theta}_n = \sum_{i=1}^n \theta_i/n$
set $\hat{\sigma}_n^2 = \sum_{i=1}^n (\theta_i - \hat{\theta}_n)^2/(n-1)$
set $CI_{1-\alpha} = \hat{\theta}_n \pm z_{1-\alpha/2} \times \hat{\sigma}_n/\sqrt{n}$
return $\hat{\theta}_n$ and $CI_{1-\alpha}$

A Python implementation with $n = 20000$ and the given parameters gave the following results:

$$\hat{\theta}_n = 4.1102$$

$$\mathrm{CI}_{99\%} = [3.9383, 4.2822].$$

(b) In order to use conditional Monte Carlo we condition on $S_{\frac{T}{2}}$ to obtain

$$
\begin{aligned}
\theta &= \mathrm{e}^{-rT}\mathrm{E}\left(\mathbb{1}(S_{\frac{T}{2}} < H)(\lambda_1 S_{\frac{T}{2}} - S_T)^+ + \mathbb{1}(S_{\frac{T}{2}} \geq H)(S_T - \lambda_2 S_{\frac{T}{2}})^+\right) \\
&= \mathrm{e}^{-rT}\mathrm{E}\left(\mathrm{E}\left(\mathbb{1}(S_{\frac{T}{2}} < H)(\lambda_1 S_{\frac{T}{2}} - S_T)^+ + \mathbb{1}(S_{\frac{T}{2}} \geq H)(S_T - \lambda_2 S_{\frac{T}{2}})^+ \mid S_{\frac{T}{2}}\right)\right) \\
&= \mathrm{e}^{-r\frac{T}{2}}\mathrm{E}\left(\mathbb{1}(S_{\frac{T}{2}} < H)\mathrm{e}^{-r\frac{T}{2}}\mathrm{E}\left((\lambda_1 S_{\frac{T}{2}} - S_T)^+ \mid S_{\frac{T}{2}}\right) + \mathbb{1}(S_{\frac{T}{2}} \geq H)\mathrm{e}^{-r\frac{T}{2}}\mathrm{E}\left((S_T - \lambda_2 S_{\frac{T}{2}})^+ \mid S_{\frac{T}{2}}\right)\right) \\
&= \mathrm{e}^{-r\frac{T}{2}}\mathrm{E}\left(\mathbb{1}(S_{\frac{T}{2}} < H)\mathrm{Put}_{\mathrm{BS}}\left(S_{\frac{T}{2}}, \lambda_1 S_{\frac{T}{2}}, r, q, \sigma, \frac{T}{2}\right) + \mathbb{1}(S_{\frac{T}{2}} \geq H)\mathrm{Call}_{\mathrm{BS}}\left(S_{\frac{T}{2}}, \lambda_2 S_{\frac{T}{2}}, r, q, \sigma, \frac{T}{2}\right)\right),
\end{aligned}
$$

where $\mathrm{Put}_{\mathrm{BS}}(S_0, K, r, q, \sigma, T)$ and $\mathrm{Call}_{\mathrm{BS}}(S_0, K, r, q, \sigma, T)$ are the Black-Scholes prices for a European put and call respectively. The algorithm is then:

Algorithm 4 Option price with conditional Monte Carlo

---

for $i = 1$ **to** n **do**
  generate $Z \sim \mathrm{N}(0,1)$
  set $S_{\frac{T}{2}} = S_0 \mathrm{e}^{(r-q-\frac{\sigma^2}{2})\frac{T}{2} + \sigma\sqrt{\frac{T}{2}}Z}$
  if $S_{\frac{T}{2}} < H$ **then**
    set $\theta_i = \mathrm{e}^{-r\frac{T}{2}}\mathrm{Put}_{\mathrm{BS}}\left(S_{\frac{T}{2}}, \lambda_1 S_{\frac{T}{2}}, r, q, \sigma, \frac{T}{2}\right)$
  else
    set $\theta_i = \mathrm{e}^{-r\frac{T}{2}}\mathrm{Call}_{\mathrm{BS}}\left(S_{\frac{T}{2}}, \lambda_2 S_{\frac{T}{2}}, r, q, \sigma, \frac{T}{2}\right)$
  end if
end for
set $\hat{\theta}_n^{CM} = \sum_{i=1}^n \theta_i / n$
set $\hat{\sigma}_n^2 = \sum_{i=1}^n (\theta_i - \hat{\theta}_n^{CM})^2/(n-1)$
set $CI_{1-\alpha} = \hat{\theta}_n^{CM} \pm z_{1-\alpha/2} \times \hat{\sigma}_n/\sqrt{n}$
return $\hat{\theta}_n^{CM}$ and $CI_{1-\alpha}$

For $n = 20000$ we obtained:

$$\hat{\theta}_n^{CM} = 4.1454$$

$$\mathrm{CI}_{99\%} = [4.1166, 4.1743].$$

As expected, due to variance reduction, the length of the confidence interval obtained with conditional Monte Carlo is considerably smaller than the one obtained with simple Monte Carlo.

Problem 3. (a) It is simple to compute the exact price of the option using the Black-Scholes formula. We obtain

$$\theta \approx 1.4101.$$

To estimate the price using standard simulation we use the following algorithm:

Algorithm 5 Put price with simple Monte Carlo

for $i = 1$ **to** n **do**
  generate $Z \sim \mathrm{N}(0,1)$
  set $S_T = S_0 \mathrm{e}^{(r-q-\frac{\sigma^2}{2})T + \sigma\sqrt{T}Z}$
  set $\theta_i = \mathrm{e}^{-rT}(K - S_T)^+$
end for
set $\hat{\theta}_n = \sum_{i=1}^{n} \theta_i / n$
set $\hat{\sigma}_n^2 = \sum_{i=1}^{n} (\theta_i - \hat{\theta}_n)^2/(n-1)$
set $CI_{1-\alpha} = \hat{\theta}_n \pm z_{1-\alpha/2} \times \hat{\sigma}_n/\sqrt{n}$
return $\hat{\theta}_n$ and $CI_{1-\alpha}$

For $n = 20000$, we obtained:

$$\hat{\theta}_n = 1.3385$$
$$\mathrm{CI}_{99\%} = [1.2350, 1.4420].$$

(b) For this problem, our payoff function is

$$h(z) = \mathrm{e}^{-rT}(K - S_T)^+ = \mathrm{e}^{-rT}\left(K - S_0 \mathrm{e}^{(r-q-\frac{\sigma^2}{2})T + \sigma\sqrt{T}z}\right)^+,$$

and the likelihood is

$$f(z) = \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{z^2}{2}}.$$

A natural importance sampling distribution would be $g \sim \mathrm{N}(\mu_*, 1)$ where μ_* maximizes the product $h(z)f(z)$ (you can use a different one). This can be done efficiently via numerical methods. The likelihood ratio in this case is

$$l(z) \triangleq \frac{f(z)}{g(z)} = \mathrm{e}^{\frac{\mu_*^2}{2} - \mu_* z}.$$

The pseudocode is the following:

Algorithm 6 Put price with importance sampling

compute μ_*
for $i = 1$ **to** n **do**
  generate $Z \sim \mathrm{N}(\mu_*, 1)$
  set $S_T = S_0 \mathrm{e}^{(r-q-\frac{\sigma^2}{2})T + \sigma\sqrt{T}Z}$
  set $l(Z) = \mathrm{e}^{\frac{\mu_*^2}{2} - \mu_* Z}$
  set $\theta_i = l(Z) \times \mathrm{e}^{-rT}(K - S_T)^+$
end for
set $\hat{\theta}_n^{IS} = \sum_{i=1}^{n} \theta_i / n$
set $\hat{\sigma}_n^2 = \sum_{i=1}^{n} (\theta_i - \hat{\theta}_n^{IS})^2/(n-1)$
set $CI_{1-\alpha} = \hat{\theta}_n^{IS} \pm z_{1-\alpha/2} \times \hat{\sigma}_n/\sqrt{n}$
return $\hat{\theta}_n^{IS}$ and $CI_{1-\alpha}$

For $n = 20000$, we obtained:

$$\hat{\theta}_n^{IS} = 1.4157$$

$$\text{CI}_{99\%} = [1.3928, 1.4385].$$

As expected, the confidence interval obtained via importance sampling is narrower than the one obtained via simple Monte Carlo.

Problem 4. (a) See Problem 3 (a).

(b) We define $m = 10$ strata Δ_j for $Z \sim \text{N}(0,1)$ as follows (there are many options): $(-\infty, -4]$, $(-4, -3]$, $(-3, -2]$, $(-2, -1]$, $(-1, 0]$, $(0, 1]$, $(1, 2]$, $(2, 3]$, $(3, 4]$ and $(4, \infty)$. Observe that if $\Delta_j = (a, b]$ then $p_j \triangleq \text{P}(Z \in \Delta_j) = \Phi(b) - \Phi(a)$. Also it is easy to simulate Z given $Z \in \Delta_j$ by simulating $U \sim \text{U}(\Phi(a), \Phi(b))$ and then setting $Z = \Phi^{-1}(U)$. The algorithm is:

Algorithm 7 Put price with sub-optimal stratified sampling

---

for $j = 1$ **to** m **do**
  compute p_j
  set $n_j = p_j \times n$
  for $i = 1$ **to** n_j **do**
    generate $Z \sim \text{N}(0,1)$ given $Z \in \Delta_j$
    set $S_T = S_0 \text{e}^{(r-q-\frac{\sigma^2}{2})T + \sigma\sqrt{T}Z}$
    set $\theta_i^{(j)} = \text{e}^{-rT}(K - S_T)^+$
  end for
  set $\theta_j = \sum_{i=1}^{n_j} \theta_i^{(j)} / n_j$
  set $\hat{\sigma}_j^2 = \sum_{i=1}^{n_j} (\theta_i^{(j)} - \theta_j)^2 / (n_j - 1)$
end for
set $\hat{\theta}_n^{SS} = \sum_{j=1}^{m} p_j \theta_j$
set $\hat{\sigma}_n^2 = \sum_{j=1}^{m} p_j^2 \hat{\sigma}_j^2 / n_j$
set $CI_{1-\alpha} = \hat{\theta}_n^{SS} \pm z_{1-\alpha/2} \times \hat{\sigma}_n$
return $\hat{\theta}_n^{SS}$ and $CI_{1-\alpha}$

For $n = 20000$, we obtained:

$$\hat{\theta}_{n,sub}^{SS} = 1.4533$$
$$\text{CI}_{99\%} = [1.4036, 1.5030].$$

Observe that not much variance reduction is achieved using stratified sampling with sub-optimal allocation.

(c) It is necessary to run a pilot simulation in order to estimate σ_j for each stratum. The pseudo-code would be:

Algorithm 8 Put price with optimal stratified sampling

for $j = 1$ **to** m **do**
  estimate $\hat{\sigma}_j^{(pilot)}$ {pilot simulation}
end for
for $j = 1$ **to** m **do**
  compute p_j
  set $n_j = \left(\frac{p_j \hat{\sigma}_j^{(pilot)}}{\sum_{k=1}^m p_k \hat{\sigma}_k^{(pilot)}} \right) \times n$
  for $i = 1$ **to** n_j **do**
    generate $Z \sim \mathrm{N}(0, 1)$ given $Z \in \Delta_j$
    set $S_T = S_0 \mathrm{e}^{(r-q-\frac{\sigma^2}{2})T+\sigma\sqrt{T}Z}$
    set $\theta_i^{(j)} = \mathrm{e}^{-rT}(K - S_T)^+$
  end for
  set $\theta_j = \sum_{i=1}^{n_j} \theta_i^{(j)}/n_j$
  set $\hat{\sigma}_j^2 = \sum_{i=1}^{n_j} (\theta_i^{(j)} - \theta_j)^2/(n_j - 1)$
end for
set $\hat{\theta}_n^{SS} = \sum_{j=1}^m p_j \theta_j$
set $\hat{\sigma}_n^2 = \sum_{j=1}^m p_j^2 \hat{\sigma}_j^2/n_j$
set $CI_{1-\alpha} = \hat{\theta}_n^{SS} \pm z_{1-\alpha/2} \times \hat{\sigma}_n$
return $\hat{\theta}_n^{SS}$ and $CI_{1-\alpha}$

For $n = 20000$, we obtained:

$$\hat{\theta}_{n,opt}^{SS} = 1.4122$$

$$\mathrm{CI}_{99\%} = [1.3929, 1.4315].$$

Observe that, with optimal allocation, the stratified sampling method reduces variance considerably.