CFRM505: Monte Carlo Methods in Finance (Winter 2021)

Instructor: Prof. Tim Leung

Midterm Exam (take-home, open-book)

1. Consider a random variable X that has probability density function (PDF)

$$f(x) = \begin{cases} \alpha x + 2\alpha x^3 & 0 \leq x \leq 1 \\ 0 & x < 0 \text{ or } x > 1. \end{cases} \tag{1}$$

   (a) Write down the constant α.

   (b) Let's generate samples of X using the *composition approach*. Describe briefly in words the steps of the simulation algorithm (you can write in the form of psuedo-code if you like). Then, proceed to implement the algorithm in R/Python to generate 10000 (or more) samples of X. Show the histogram, and estimate the mean and variance.

   (c) Now, apply an acceptance-rejection algorithm for generating X with the use of a uniform random variable. Derive analytically the minimum value of a you can use. Then proceed to implement the algorithm in R/Pythonto generate 10000 (or more) samples of X. Show the histogram, and estimate the mean and variance.

2. Let $X \sim \exp(\lambda)$, for some constant $\lambda > 0$. Its CDF is given by $F(x) = 1 - e^{-\lambda x}$, for $x > 0$.

   (a) Compute analytically and give an expression for the probability

$$\mathbb{P}\{X \leq x \,|\, a \leq X \leq b\},$$

   for some constants a, b with $0 < a < b$. You can express your answer using/in terms of $F(\cdot)$.

   (b) Let Y be a *truncated* exponential random variable over the interval $[a, b]$, with the pdf

$$f(x) = C\lambda e^{-\lambda x}, \text{ for } a \leq x \leq b.$$

   Note that it only take values in the interval $[a, b]$.

   Now let $a = 1, b = 3, \lambda = 0.5$. State the value of the constant C (no steps needed). Describe in words the inverse transform method to simulate Y. Then proceed to implement the algorithm in R/Python. Show the histogram, and estimate the mean and variance.

3. Continuing from Q.2, again we have the truncated exponential random variable Y with $a = 1, b = 3, \lambda = 0.5$, but also consider another truncated exponential

random variable $\widetilde{Y}$ over the interval $[2, 5]$ with $\lambda = 1$. We want to generate correlated samples of Y and $\widetilde{Y}$, using the Gaussian copula with bi-variate normals with covariance matrix

$$\Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

Specifically, generate a plot in Python/R to show the relationship between ρ (x-axis) and the sample correlation between Y and $\widetilde{Y}$ (y-axis). You only need to do it for $\rho \in \{-1.0, -0.9, -0.8, -0.7, \ldots, 0, \ldots, 0.7, 0.8, 0.9, 1.0\}$, and use $N = 10000$ for each case.

4. Consider two correlated CIR processes:

$$\begin{aligned} dX_t &= \theta(\mu - X_t)\,dt + \sigma\sqrt{X_t}\,dB_t, \\ d\widetilde{X}_t &= \widetilde{\theta}(\widetilde{\mu} - \widetilde{X}_t)\,dt + \widetilde{\sigma}\sqrt{\widetilde{X}_t}\left(\rho\,dB_t + \sqrt{1-\rho^2}dW_t\right) \end{aligned}$$

with constants $\mu, \theta, \sigma, \widetilde{\mu}, \widetilde{\theta}, \widetilde{\sigma} > 0$, and $\rho \in [-1, 1]$. Here, W_t and B_t are independent standard BMs.

Write an algorithm in R/Python to simulate these two stochastic processes over the time horizon $[0, 2]$ (2 years). As an example, take $\Delta t = 1/250, \theta = 10, \mu = 15, \sigma = 6, \widetilde{\theta} = 16, \widetilde{\mu} = 10, \widetilde{\sigma} = 3$, and $\rho = 0.2$. In R/Python, simulate 1000 (or more) paths for each process, and estimate the mean and variance of X_2 and $\widetilde{X}_2$.

Lastly, generate a plot in Python/R to show the relationship between ρ (x-axis) and the sample correlation between X_2 and $\widetilde{X}_2$ (y-axis). You only need to do it for $\rho \in \{-0.8, -0.2, 0, 0.2, 0.8\}$.

5. Suppose we want to estimate the expectation

$$\mathbb{E}[\,(e^{(V+W)} - 2)^+]$$

with $V \sim U(-0.5, 1.5)$ (uniform r.v.) and $W \sim N(-0.2, 2)$ (normal r.v.). First, write an MC algorithm in R/Python to estimate the expectation. Specifically, report the mean and variance, as well as the confidence interval of your estimate.

Next, apply a control variate to this estimation. To this end, denote by Z the control variate of your choice. Write down your control variate (no proof/steps needed for this).

Now incorporate your control variate and implement an algorithm in R/Python to estimate the expectation again. Specifically, report the mean and variance, as well as the confidence interval of your estimate. Do you achieve variance reduction? If so, quantify it.

Here are the parameters you can use for both estimations (if applicable): $n0 = 1000$ (pilot runs), $N = 50000$ (sample size), $\delta = 0.05$, and $\epsilon = 0.01$.