MTH 9821 Numerical Methods for Finance I

Lecture 5 & 6–Monte Carlo Method

1 Monte Carlo Methods for Evaluating Integrals

$$I = \int_0^1 f(x)dx$$

Note: $\int_0^1 f(x)dx = E\big[f(U)\big]$, where $U = Uniform\big([0,1]\big)$

Procedure:

- Generate independent samples $U_1, U_2, \ldots, U_n$ of U
- Let $X_i = f(U_i)$, then $E(X_i) = E\big(f(U)\big) = I$
  By SLLN,

$$\frac{1}{n}\sum_{i=1}^{n} X_i \xrightarrow{a.s.} E(X) = I$$

Convergence:

If $\int_0^1 \big|f(x)\big|^2 dx < \infty \;\Rightarrow\; var(X) = var(f(U)) < \infty, \qquad \big(f(x) \text{ is a } L^2 \text{ function.}\big)$

By CLT, $\frac{\frac{1}{n}\sum_{i=1}^{n} X_i - I}{\frac{\sigma_X}{\sqrt{n}}} \xrightarrow{d} Z$

Approximation Error:

$$\left|\frac{1}{n}\sum_{i=1}^{n} X_i - I\right| = O\left(\frac{1}{\sqrt{n}}\right)$$

Comments:

- Monte Carlo simulation converges at the rate $\frac{C}{\sqrt{n}}$ where n is the number of sample values, $C = \sigma_X$. Therefore, in order to optimize the speed, we need to generate random variables with smaller variation.
- The convergence of Monte Carlo method is $O\left(\frac{1}{\sqrt{n}}\right)$.
  - Convergence is actually slow.
  - Finite difference methods, on the other hand, converge at rate $O\left(\frac{1}{n^2}\right)$ for two dimensional PDEs (faster).

2 Advantages and Disadvantages of Monte Carlo Method

- **Advantages:**
  - **Simple to code**
  - **Very efficient for path-dependent securities:**
  - **Works well for multi-asset derivative securities.**
  - **Disadvantages:**
    - **Converges slowly**: computationally expensive
    - **Challenging to apply for American options**:
    - **Difficult to compute Greeks**

3 Monte Carlo Method for Non-path-dependent Single Asset Derivative Securities

3.1 Securities Pricing

⋆ Generate independent samples of $S(T)$, denoted $S_1, S_2, \cdots, S_n$
e.g., to valuate derivative security on underlying asset with lognormal distribution: generate $Z_1, Z_2, \cdots, Z_n$ independent samples of Z,

$$S_i = S(0)exp\left((r - q - \frac{\sigma^2}{2})T + \sigma\sqrt{T}Z_i\right), \quad \forall i = 1 : n$$

⋆ Compute

$$V_i \;=\; e^{-rT}V(S_i), \quad \hat{V}(n) \;=\; \frac{1}{n}\sum_{i=1}^{n} V_i$$

⋆ Then

$$|\hat{V}(n) - V(0)| = O(\frac{1}{\sqrt{n}})$$

For example, put option

$$V(0) = e^{-rT}\mathbb{E}_{RN}\big(\max(K - S(T), 0)\big)$$

$$\begin{aligned}
&\star \quad S_i \to S(T): \quad \text{generate } S_i \quad \forall i = 1 : n \\
&\star \quad V_i \to V(0): \quad V_i = e^{-rT}\max\big(K - S_i, 0\big) \\
&\star \quad \hat{V}(n) = \frac{1}{n}\sum_{i=1}^{n} V_i \\
&\quad\;\; |\hat{V}(n) - P_{BS}| = O(\frac{1}{\sqrt{n}})
\end{aligned}$$

Remark:

- There's no rule for convergence when using Monte Carlo Methods for vanilla European options. We can use the same sample of normal random variables for variance reduction.

- Why comparing with Black Scholes? Because BS and MC both follow the lognormal assumption.

3.2 Greeks Computations

3.2.1 Delta

$$\Delta = \frac{\partial V(0)}{\partial S(0)} = \frac{\partial V(0)}{\partial S(T)}\frac{\partial S(T)}{\partial S(0)}$$
$$\text{where} \quad \frac{dS(T)}{dS(0)} = exp\left((r - q - \frac{\sigma^2}{2})T + \sigma\sqrt{T}Z_i\right) = \frac{S(T)}{S(0)}$$

For example, put option

$$V(0) = e^{-rT}\mathbb{E}_{RN}\big(\max(K - S(T), 0)\big)$$

$\star \quad S_i \to S(T): \quad \text{generate } S_i \quad \forall i = 1:n$

$\star \quad V_i \to V(0): \quad V_i = e^{-rT}\max\big(K - S_i, 0\big)$

$\star \quad \text{Thus,} \quad \dfrac{\partial V_i}{\partial S_i} = e^{-rT}\begin{cases} -1, & \text{if } S_i < K \\ 0, & \text{if } S_i > K \end{cases} = e^{-rT}\mathbb{1}(S_i < K)$

$\star \quad \Delta_i = -e^{-rT}\mathbb{1}(S_i < K)\dfrac{S_i}{S(0)}$

$\star \quad \dfrac{1}{n}\displaystyle\sum_{i=1}^{n}\Delta_i \to \Delta_{BS}(p)$

For call option

$$V(0) = e^{-rT}\mathbb{E}_{RN}\big(\max(S(T) - K, 0)\big)$$

$\star \quad S_i \to S(T): \quad \text{generate } S_i \quad \forall i = 1:n$

$\star \quad V_i \to V(0): \quad V_i = e^{-rT}\max\big(S_i - K, 0\big) = e^{-rT}(S_i - K)\mathbb{1}(S_i > K)$

$\star \quad \dfrac{\partial V_i}{\partial S_i} = e^{-rT}\mathbb{1}(S_i > K)$

$\star \quad \Delta_i(c) = e^{-rT}\mathbb{1}(S_i > K)\dfrac{S_i}{S(0)}$

$\star \quad \dfrac{1}{n}\displaystyle\sum_{i=1}^{n}\Delta_i(c) \to \Delta_{BS}(p)$

Remark: This above calculation also satisfies Put-Call Parity.

$$C - P \ = \ Se^{-qT} - Ke^{-rT} \ \Rightarrow \ \Delta(c) - \Delta(p) \ = \ e^{-qT}$$

We can prove that $\frac{1}{n}\sum\limits_{i=1}^{n}\Delta_i(c) - \frac{1}{n}\sum\limits_{i=1}^{n}\Delta_i(p) \ \to \ \Delta(c) - \Delta(p) \ = \ e^{-qT}$

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n}\Delta_i(c) - \frac{1}{n}\sum_{i=1}^{n}\Delta_i(p) &= e^{-rT}\frac{1}{nS(0)}\sum_{i=1}^{n}S_i\big(\mathbb{1}(S_i > K) + \mathbb{1}(S_i < K)\big) \\
&= e^{-rT}\frac{1}{nS(0)}\sum_{i=1}^{n}S_i \\
&\to e^{-rT}\frac{1}{S(0)}E_{RN}\big(S(T)\big) = e^{-rT}\frac{1}{S(0)}S(0)e^{(r-q)T} = e^{-qT}
\end{aligned}$$

3.2.2 Vega

$$\Delta = \frac{\partial V(0)}{\partial \sigma} = \frac{\partial V(0)}{\partial S(T)}\frac{\partial S(T)}{\partial \sigma}$$

$$\text{where} \quad \frac{dS(T)}{d\sigma} = S(T)(-\sigma T + \sqrt{T}Z)$$

For example, put option

$$V(0) = e^{-rT}\mathbb{E}_{RN}\big(\max(K - S(T), 0)\big)$$

- $\star$ $S_i \to S(T):$ generate S_i $\forall i = 1 : n$
- $\star$ $V_i \to V(0):$ $V_i = e^{-rT} \max\big(K - S_i, 0\big)$
- $\star$ Thus, $\dfrac{\partial V_i}{\partial S_i} = e^{-rT} \begin{cases} -1, & \text{if } S_i < K \\ 0, & \text{if } S_i > K \end{cases} = e^{-rT} \mathbb{1}(S_i < K)$
- $\star$ $Vega_i = -e^{-rT} \mathbb{1}(S_i < K)(-\sigma T + \sqrt{T} Z)$
- $\star$ $\dfrac{1}{n} \displaystyle\sum_{i=1}^{n} Vega_i \to Vega_{BS}(p)$

4 Monte Carlo Method for Path-dependent Derivatives

$\star$ Simulate n paths of the underlying asset, each path discretized between $t = 0$ and $t = T$ using m time steps of length $\delta_t = \frac{T}{m}$.

$\star$ Generate $N = m \cdot n$ independent samples of Z, then use $Z_0, Z_1, \ldots, Z_{m-1}$ to generate one path as follows:

$$S(t_{j+1}) \;=\; S(t_j)\exp\left((r-q-\frac{\sigma^2}{2})\delta_t + \sigma\sqrt{\delta_t}Z_j\right), \quad \forall j = 0:(m-1)$$

where $t_j = j\delta_t,\ j = 0:m$

Remark: From $dS = (r-q)Sdt + \sigma SdX$, how should we discretize?

Choice 1: (No)

$$\begin{aligned}
& dS \;=\; (r-q)Sdt + \sigma SdX \\
\Rightarrow\; & S(t_{j+1}) - S(t_j) = (r-q)S(t_j)\delta_t + \sigma S(t_j)\big(X(t_{j+1}) - X(t_j)\big) \\
& \text{Note that } X(t_{j+1}) - X(t_j) \sim N(0, t_{j+1} - t_j) \sim N(0, \delta_t) \\
\Rightarrow\; & S(t_{j+1}) - S(t_j) = (r-q)S(t_j)\delta_t + \sigma S(t_j)\sqrt{\delta_t}Z_j \\
\Rightarrow\; & S(t_{j+1}) \;=\; S(t_j)\big[(r-q)\delta_t + \sigma\sqrt{\delta_t}Z_j + 1\big]
\end{aligned}$$

Note that it's possible (though with small possibility) that $\big[(r-q)\delta_t + \sigma\sqrt{\delta_t}Z_j + 1\big] < 0$, which makes $S(t_{j+1}) < 0$.

Choice 2: (Yes)

$$\begin{aligned}
dS \;&=\; (r-q)Sdt + \sigma SdX \\
\Rightarrow d\ln S \;&=\; (r-q-\frac{\sigma^2}{2})dt + \sigma dX \quad \text{(Ito's Lemma)} \\
\Rightarrow \ln\left(\frac{S(t_{j+1})}{S(t_j)}\right) \;&=\; (r-q-\frac{\sigma^2}{2})\delta_t + \sigma\sqrt{\delta_t}Z_j \\
\Rightarrow S(t_{j+1}) \;&=\; S(t_j)\exp\left((r-q-\frac{\sigma^2}{2})\delta_t + \sigma\sqrt{\delta_t}Z_j\right)
\end{aligned}$$

Remark:

Convergence order of MC simulations for path dependent derivative securities depends both on n, the number of simulations as $O\left(\frac{1}{\sqrt{n}}\right)$, and also on δ_t, the time step of the discretization as $O(\delta_t)$.

Order of convergence is $O\left(\max\left(\frac{1}{\sqrt{n}}, \delta_t\right)\right)$.

To achieve optimal convergence speed, we try to make

$$\delta_t \approx \frac{1}{\sqrt{n}} \implies \frac{T}{m} \approx \frac{1}{\sqrt{n}} \implies n \approx \frac{m^2}{T^2}$$

Recall that

$$m \cdot \frac{m^2}{T^2} \approx N \Longrightarrow \frac{T}{m} \approx \sqrt[3]{\frac{T}{N}}$$

e.g., $T = 1$, $N = 1{,}000{,}000$, $m \approx \sqrt[3]{N}T^{\frac{2}{3}} = 100$, it's approximately twice a week.

5 Methods for Generating Standard Normal Samples

1. Inverse Transform Method
2. Acceptance-Rejection Method
3. Box-Muller Method (with Marsaglia-Bray Algorithm)

We first need to generate samples of uniform random variables

Linear Congruential Generator of Uniform Random Variables
The generator takes the form

- Choose x_0, a, c, m positive integers
- Generate $u_1, u_2, \dots,$ form $U[0,1]$ as follows:

for $i = 0 : N$
    $x_{i+1} = (ax_i + c) \ \ mod \ \ m;$
    $u_{i+1} = \frac{x_{i+1}}{m};$
end

Good choice for x_0, a, c, m requires

(1) c, m are relatively prime, i.e., $(c, m) = 1$ (common divisor)

(2) every prime number that divides m divides $a - 1$, i.e., $p|m \Rightarrow p|(a-1)$

(3) $a - 1$ is divisible by 4 if m is, i.e., $4|m \Rightarrow 4|(a-1)$

Linear Congruential Generators are effective since

(1) Period has maximal length (m) if a, m, c are chosen properly

(2) Fast - it requires fewer operations to generate each sample

(3) Portable - it generates the same sequence of random numbers on different platforms

(4) good randomness properties

5.1 Inverse Transform Method

Find samples of the random variable X with cummulative distribution function $F(x)$:
Given $u_1, u_2, \dots, u_n$ samples of $U = Unif[0,1]$,

$$x_i \;=\; F^{-1}(u_i),\ \forall i \geq 1$$

where $x_1, x_2, \dots, x_n$ are samples of X.
We can easily check as follows:
Let $Y = F^{-1}(U)$,

$$\begin{aligned} &\mathbb{P}(Y \leq a) = \mathbb{P}(F^{-1}(U) \leq a) = \mathbb{P}(U \leq F(a)) = F(a) = \mathbb{P}(X \leq a) \\ \Rightarrow\ & Y = F^{-1}(U) = X \end{aligned}$$

e.g., for standard normal random variable,

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-t^2/2} dt$$

solve $F(x) = y$ for x, given y.
(See class handout page 67 69 for more detailed computation)

5.2 Acceptance-Rejection Method

Goal: generate samples of random variable with pdf $f(x)$ using samples of a random variable with pdf $g(x)$ (which we already know how to generate, e.g., by using inverse transform method), and there exists a constant $c \in \mathbb{R}$ s.t. $f(x) \leq cg(x),\ \forall x \in \mathbb{R}$.
Do: generate sample from g and accept it with probability $\frac{f(x)}{cg(x)}$

Step 1: Generate X from g

Step 2: Generate $U \sim Unif\big([0,1]\big)$

Step 3: If $U \leq \frac{f(x)}{cg(x)}$, return X; else, go to step 1

Following above steps, we have

$$\begin{aligned} \mathbb{P}(Y \leq \beta) &= \mathbb{P}(X \leq \beta | U \leq \frac{f(x)}{cg(x)}) = \frac{\mathbb{P}\big((X \leq \beta) \cap (U \leq \frac{f(x)}{cg(x)})\big)}{\mathbb{P}\big(U \leq \frac{f(x)}{cg(x)}\big)} \\ \text{where } \ \mathbb{P}\left(U \leq \frac{f(x)}{cg(x)}\right) &= \int_{-\infty}^{\infty} g(x) \left(\int_0^1 \mathbb{1}_{U \leq \frac{f(x)}{cg(x)}} du\right) dx = \int_{-\infty}^{\infty} g(x) \frac{f(x)}{cg(x)} dx = \frac{1}{c} \int_{-\infty}^{\infty} f(x) dx = \frac{1}{c} \\ \mathbb{P}\big((X \leq \beta) \cap (U \leq \frac{f(x)}{cg(x)})\big) &= \int_{-\infty}^{\beta} g(x) \left(\int_0^1 \mathbb{1}_{U \leq \frac{f(x)}{cg(x)}} du\right) dx = \frac{1}{c} \int_{-\infty}^{\beta} f(x) dx \end{aligned}$$

$$\Rightarrow\ \mathbb{P}(Y \leq \beta) = \int_{-\infty}^{\beta} f(x) dx$$

Generate samples of standard normal

Goal:

Generate samples of Z with $f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$.

We use double exponential random variable with $g(x) = \frac{1}{2} e^{-|x|}$.

$\star$ **Determine** c

$$\frac{f(x)}{g(x)} = \frac{\frac{1}{\sqrt{2\pi}} e^{-x^2/2}}{\frac{1}{2} e^{-|x|}} = \sqrt{\frac{2}{\pi}} e^{|x| - \frac{x^2}{2}} = \sqrt{\frac{2}{\pi}} e^{\frac{1}{2}} e^{-\frac{1}{2}(|x|-1)^2} \leq \frac{2e}{\pi}$$

Thus, we choose $c = \frac{2e}{\pi}$, and $\frac{f(x)}{cg(x)} = e^{-\frac{1}{2}(|x|-1)^2}$

$\star$ **Use inverse transformation method to generate double exponential r.v.**

Compute $G(x) = \int_{-\infty}^{x} g(t)dt$

- If $x < 0$,

$$g(x) = \frac{1}{2} e^{-(-x)} = \frac{1}{2} e^{x} \;\Rightarrow\; G(x) = \frac{1}{2} e^{x},\; G(0) = \frac{1}{2}$$

- If $x > 0$,

$$G(x) = G(0) + \int_0^x \frac{1}{2} e^{-t} dt = \frac{1}{2} + \frac{1}{2}(1 - e^{-x}) = 1 - \frac{1}{2} e^{-x}$$

Solve $G(x) = y$

- $x < 0 \Rightarrow y < \frac{1}{2}$, solve $\frac{1}{2} e^{x} = y \Longrightarrow x = \ln(2y)$

- $x > 0 \Rightarrow y > \frac{1}{2}$, solve $1 - \frac{1}{2} e^{-x} = y \Longrightarrow x = -\ln(2(1-y))$

$$G^{-1}(y) = \begin{cases} \ln(2y), & y < \frac{1}{2} \\ -\ln(2(1-y)), & y > \frac{1}{2} \end{cases} \sim \begin{cases} \ln U, & 0 < U < 1 \\ -\ln U, & 0 < U < 1 \end{cases}$$

choose $c = \sqrt{\frac{2e}{\pi}}$.

Step 0: Generate U_1, U_2, U_3 from $U\big([0,1]\big)$

Step 1: $X = -\ln(U_1)$ //generate only positive samples

Step 2: If $U_2 > \exp\big(-\frac{1}{2}(|x|-1)^2\big)$, go to step 0 //acceptance-rejection
else, generate U_3 from $U\big([0,1]\big)$
If $U_3 \leq \frac{1}{2}$, $X = -X$
Return X

5.3 The Box-Muller Method

Generate a sample from the bivariate normal distribution where each component is a univariate standard normal.

$$\textbf{Uniform } U_1, U_2 \Longrightarrow \textbf{Exponential } R \Longrightarrow \textbf{Independent Standard Normals } Z_1, Z_2$$

If Z_1, Z_2 are independent standard normals, then $R = Z_1^2 + Z_2^2$ is exponential with mean 2.

Recall: Exponential distribution with mean α
⋆ pdf: $f(x) = \frac{1}{\alpha}e^{-\frac{x}{\alpha}}$, for $x > 0$; CDF: $F(x) = 1 - e^{-\frac{x}{\alpha}}$
⋆ Inverse function of F: $x = -\alpha \ln(1-y)$

It's enough to show that $P(R \le a) = P(Z_1^2 + Z_2^2 \le a)$:

$$P(R \le a) = F(a) = 1 - e^{-\frac{a}{2}}$$

$$P(Z_1^2 + Z_2^2 \le a) = \int_{\mathbb{R}}\int_{\mathbb{R}} \mathbb{1}_{x_1^2+x_2^2\le a} \frac{1}{\sqrt{2\pi}}e^{-\frac{x_1^2}{2}} \frac{1}{\sqrt{2\pi}}e^{-\frac{x_2^2}{2}} dx_1 dx_2 = \frac{1}{2\pi}\int_{\mathbb{R}}\int_{\mathbb{R}} \mathbb{1}_{x_1^2+x_2^2\le a} e^{-\frac{x_1^2+x_2^2}{2}} dx_1 dx_2$$

Let $x_1 = r\cos\theta,\ x_2 = r\sin\theta$

$$= \frac{1}{2\pi}\int_0^{2\pi}\int_0^{\infty} \mathbb{1}_{r^2\le a} e^{-\frac{r^2}{2}} r dr d\theta = \frac{1}{2\pi}\int_0^{2\pi}\int_0^{\sqrt{a}} e^{-\frac{r^2}{2}} r dr d\theta$$

$$= \left(-e^{-\frac{r^2}{2}}\right)\Bigg|_0^{\sqrt{a}} = 1 - e^{-\frac{a}{2}}$$

5.3.1 Implement:

Given R, (Z_1, Z_2) is uniformly distributed on the circle of center O and radius $\sqrt{R}$, first generate R, then choose a point uniformly on the circle of radius $\sqrt{R}$.

Generate $U_1,\ U_2 \sim Unif\big([0,1]\big)$;
$R = -2\ln(U_1)$;
(Note that U_1 and $1 - U_1$ has same distribution.)
$V = 2\pi U_2$;
$Z_1 = \sqrt{R}\cos V,\ Z_2 = \sqrt{R}\sin V$;
return $Z_1,\ Z_2$;

5.3.2 Marsaglia-Bray Algorithm

An improvement of Box-Muller by avoiding trigonometric functions.

- Let $U_1,\ U_2 \sim Unif\big([0,1]\big)$
- For U_1, U_2 inside circle $D(0,1)$, $X = U_1^2 + U_2^2 \sim Unif\big([0,1]\big)$

$$P(a \le X \le b) = P(a \le U_1^2 + U_2^2 \le b) = \iint_{D(0,1)} \mathbb{1}_{\{a \le u_1^2 + u_2^2 \le b\}} \frac{1}{\pi} du_1 du_2$$
$$= \frac{1}{\pi} \int_0^{2\pi} \int_{\sqrt{a}}^{\sqrt{b}} r dr d\theta = \frac{1}{\pi} \dot{2}\pi \int_{\sqrt{a}}^{\sqrt{b}} r dr = r^2 \Big|_{\sqrt{a}}^{\sqrt{b}} = b - a$$
$$\Big((U_1, U_2) \text{ subject to } X < 1 \text{ is uniformly distributed on circle } D(0,1) \text{ with pdf } \frac{1}{\pi}\Big)$$

- $X \sim Unif\big([0,1]\big) \ \Rightarrow\ R = -2\ln(1-X) = -2$
  $(1-X) \sim Unif\big([0,1]\big) \ \Rightarrow\ R = -2\ln(X) = -2$, $R \sim Exponential(2)$
- Look for $Z_1 = U_1 Y$, $Z_2 = U_2 Y$ on the circle $(0, \sqrt{R})$, i.e., $Z_1^2 + Z_2^2 = R$.
  $R = -2\ln(X) = Z_1^2 + Z_2^2 = (U_1^2 + U_2^2)Y^2 = XY^2$.
- Thus, $Y^2 = -2\frac{\ln(X)}{X} \ \Rightarrow\ Y = \sqrt{-2\frac{\ln(X)}{X}}$
  We get $Z_1 = U_1 Y$, $Z_2 = U_2 Y$.

> **while** $X > 1$ **do**
>   Generate $U_1,\ U_2 \sim Unif\big([0,1]\big)$;
> $U_1 = 2U_1 - 1$, $U_2 = 2U_2 - 1$; $\big(U_1,\ U_2 \sim Unif\big([-1,1]\big)\big)$
> $X = U_1^2 + U_2^2$;
> **end**
> $Y = \sqrt{-2\frac{\ln X}{X}}$;
> $Z_1 = U_1 Y$, $Z_2 = U_2 Y$;
> return Z_1, Z_2;

6 Variance Reduction Technique

6.1 Control Variates

Recall: Linear Regression of Y_i as X_i, $i = 1 : n$

$$\begin{aligned}
&\text{Find } c_1, c_2 \text{ such that } Y_i \approx c_1 X_i + c_2 \\
\Rightarrow\ & \min ||\tilde{Y} - c_1\tilde{X} - c_2||^2 = \min \sum_{i=1}^{n} (Y_i - c_1 X_i - c_2)^2 \\
\Rightarrow\ & c_2 = \hat{Y}(n) - c_1 \hat{X}(n), \\
& c_1 = \frac{cov(\tilde{X}, \tilde{Y})}{var(\tilde{X})} = \frac{\sum\limits_{i=1}^{n} \big(X_i - \hat{X}(n)\big)\big(Y_i - \hat{Y}(n)\big)}{\sum\limits_{i=1}^{n} \big(X_i - \hat{X}(n)\big)^2}
\end{aligned}$$

Alternatively, we need to find

$$\begin{pmatrix} X_1 & 1 \\ X_2 & 1 \\ \vdots & \vdots \\ X_n & 1 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \approx \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}$$

From the fact that the error of regression should be orthogonal to $\mathbb{1}^t$, we need that

$$\begin{aligned}
& \mathbb{1}^t \big(\tilde{Y} - c_1\tilde{X} - c_2\mathbb{1}\big) = 0 \\
\Rightarrow & \mathbb{1}^t\tilde{Y} - c_1 \mathbb{1}^t \tilde{X} - c_2 \mathbb{1}^t \mathbb{1} = 0 \\
& \Big(\text{Denote } \hat{X}(n) = \frac{1}{n}\sum_{i=1}^{n} X_i,\ \hat{Y}(n) = \frac{1}{n}\sum_{i=1}^{n} Y_i\Big) \\
\Rightarrow & n\hat{Y}(n) - c_1 n\hat{X}(n) - c_2 n = 0 \\
\Rightarrow & \mathbf{c_2 = \hat{Y}(n) - c_1\hat{X}(n)} \\
& \tilde{Y} \approx c_1\tilde{X} + c_2 = c_1\tilde{X} + \hat{Y}(n) - c_1\hat{X}(n) \\
\Rightarrow & \tilde{Y} - \hat{Y}(n) = c_1 \left(\tilde{X} - \hat{X}(n)\right) \\
\Rightarrow & \left(\tilde{X} - \hat{X}(n)\right)^t \left(\tilde{Y} - \hat{Y}(n) - c_1\left(\tilde{X} - \hat{X}(n)\right)\right) = 0 \\
\Rightarrow & \mathbf{c_1} = \frac{\left(\mathbf{\tilde{X} - \hat{X}(n)}\right)^{\mathbf{t}} \left(\mathbf{\tilde{Y} - \hat{Y}(n)}\right)}{||\mathbf{\tilde{X} - \hat{X}(n)}||^{\mathbf{2}}} = \frac{\mathbf{cov(\tilde{X}, \tilde{Y})}}{\mathbf{var(\tilde{X})}}
\end{aligned}$$

Control Variates Method

Generate $Y_1, Y_2, \cdots, Y_n$ n independent replications of Y

- In Monte Carlo, we use $\hat{Y}(n) = \frac{1}{n}\sum_{i=1}^{n} Y_i$ to approximate Y;
- Alternatively, every time we generate a replication Y_i of Y, also generate a replication X_i of another variable X, whose exact expected value is known.
  And instead of Y_i, use
  $$\tilde{Y_i} = Y_i - b\,(X_i - E(X))$$
  where b is a constant chosen carefully to approximate Y.

Example: Generate $S_1, S_2, \cdots, S_n$ samples of $S(T)$, with $E(S(T)) = e^{rT}S(0)$
Use S_i to approximate V_i: $\tilde{V_i} = V_i - b\left(S_i - e^{rT}S(0)\right)$

$Y_{CV}(n)$ **is an unbiased estimation of** Y

$$\begin{aligned} Y_{CV}(n) &= \frac{1}{n}\sum_{i=1}^{n}\tilde{Y_i} = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - b(X_i - E(X))\right) \\ &= \frac{1}{n}\sum_{i=1}^{n} Y_i - \left(\frac{b}{n}\sum_{i=1}^{n} X_i - \frac{b}{n}nE(X)\right) \\ &= \hat{Y}(n) - b\left(\hat{X}(n) - E(X)\right) \end{aligned}$$

$$E\left(Y_{CV}(n)\right) = E(\hat{Y}(n)) - b\left(E(X) - E(X)\right) = E(\hat{Y}(n)) = E(Y)$$

b **should be chosen carefully: variance of** $Y_{CV}(n)$ **is minimum**

$$\begin{aligned} var[Y_{CV}(n)] &= var[\frac{1}{n}\sum_{i=1}^{n}\tilde{Y_i}] = \frac{1}{n}var[\tilde{Y_i}] \\ &= \frac{1}{n}var[Y_i - b(X_i - E(X))] = \frac{1}{n}var[Y - bX] \\ &= \frac{1}{n}[var(Y) - 2bcov(Y,X) + b^2 var(X)] \\ &\text{the minimum is obtained with } b^* = \frac{cov(Y,X)}{var(X)} \\ var^*\left[Y_{CV}(n)\right] &= \frac{1}{n}\left[var(Y) - 2\frac{cov(Y,X)}{var(X)}cov(Y,X) + \frac{cov(Y,X)^2}{var(X)^2}var(X)\right] = \frac{1}{n}\left[var(Y) - \frac{cov(Y,X)^2}{var(X)}\right] \\ &= \frac{1}{n}\left[\sigma_Y^2 - \frac{\sigma_X^2\sigma_Y^2\rho_{XY}^2}{\sigma_X^2}\right] = \frac{\sigma_Y^2}{n}\left[1 - \rho_{XY}^2\right] \end{aligned}$$

Therefore,

$$\frac{var\left[Y_{CV}(n)\right]}{var\left[\hat{Y}(n)\right]} = 1 - \rho_{XY}^2$$

We want to find X such that ρ_{XY} close to 1 or to -1.

If X, Y are uncorrelated ($\rho_{XY} = 0$), then the control variates doesnot help.

In practice, we use sample values of b^*

$$\hat{b}(n) = \frac{\sum_{i=1}^{n} \left(X_i - \hat{X}(n)\right)\left(Y_i - \hat{Y}(n)\right)}{\sum_{i=1}^{n} \left(X_i - \hat{X}(n)\right)^2}$$

Remark:
⋆ Here we use $\hat{X}(n)$ rather than $E(X)$, which is also known.

How would the correlation matter?

ρ_{XY}	$1-\rho_{XY}^2$	n/n_{CV}
0.95	$\approx$0.1	$\approx$10
0.9	$\approx$0.2	$\approx$5
0.75	04375	$\approx$2.5

- Monte Carlo error is to the order of $\sigma_Y/\sqrt{n}$.
- Control variate error is to the order of $\sigma_{Y_{CV}}/\sqrt{n_{CV}}$.

If $1-\rho_{XY}^2 = 0.1, \quad \Rightarrow \quad \frac{\sigma_{Y_{CV}}^2}{\sigma_Y^2} = 1-\rho_{XY}^2 = 0.1 \quad \Rightarrow \quad \sigma_{Y_{CV}} = \sigma_Y/\sqrt{10}$
Therefore, for errors to be similar,

$$n \approx 10 \cdot n_{CV}$$

i.e., control variate method needs 10 times less simulations than Monte Carlo method.

When is the derivative security value perfectly correlated with the underlying?
For example, European call option

$$\tilde{c_i} = c_i - b(n)(S_i - e^{rT}S(0))$$

We estimated correlation $\tilde{\rho}$ between $S(T)$ and $\max(S(T)-K,0)$ for different value of K in Monte Carlo method $(S(0)=50,\ \sigma=0.3,\ T=0.25)$:

K	40	45	50	55	60	65	70
$\tilde{\rho}^2$	0.99	0.94	0.8	0.59	0.36	0.19	0.08

We can see that for deep ITM options, the correlation is high, for deep OTM options, the correlation is pretty low.

We can thus valuate a different derivative security as

$$\tilde{V_i} = V_i - b(n)(C_i - C_{BS})$$

e.g. Payoff = max $\left(S_T^2 - K, 0\right)$
More specificly, we first generate S_i, and use them to simulate both V_i and C_i.

Weighted Monte Carlo
In Control Variate,

$$\begin{aligned}
\tilde{Y}_i &= Y_i - b(n)(X_i - E(X)) \\
Y_{CV}(n) &= \frac{1}{n}\sum_{i=1}^{n}\tilde{Y}_i = \hat{Y}(n) - b\left(\hat{X}(n) - E(X)\right) \\
\hat{b}(n) &= \frac{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)\left(Y_i - \hat{Y}(n)\right)}{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)^2} \\
&= \sum_{i=1}^{n} Y_i \frac{X_i - \hat{X}(n)}{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)^2} - \hat{Y}(n)\frac{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)}{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)^2} \\
&\left(\text{where } \sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right) = 0\right) \\
&= \sum_{i=1}^{n} Y_i \frac{X_i - \hat{X}(n)}{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)^2} \\
Y_{CV}(n) &= \sum_{i=1}^{n} Y_i \left(\frac{1}{n} - \frac{(X_i - \hat{X}(n))(\hat{X}(n) - E(X))}{\sum_{i=1}^{n}\left(X_i - \hat{X}(n)\right)^2}\right)
\end{aligned}$$

Weighted MC:

$$\begin{aligned}
Y_{WMC}(n) &= \sum_{i=1}^{n}\omega_i Y_i \\
\text{s.t.} \quad & \sum_{i=1}^{n}\omega_i = 1, \quad \sum_{i=1}^{n}\omega_i X_i = E(X) \\
\text{minimize} \quad & \sum_{i=1}^{n}\omega_i^2 \\
\text{and thus} \quad & var\left(Y_{WMC}(n)\right) = var(Y)\sum_{i=1}^{n}\omega_i^2
\end{aligned}$$

6.2 Moment Matching

Idea: transform the paths to match known moments of a random variable.

Example: In our derivative-underlying asset case,

$$S_i \longrightarrow V_i \longrightarrow \hat{V}(n) = \frac{1}{n}\sum_{i=1}^{n} V_i$$

$$E\big(S(T)\big) = e^{rT}S(0) \quad \text{(forward price)}$$

$$\text{while} \quad \frac{1}{n}\sum_{i=1}^{n} S_i \neq e^{rT}S(0)$$

To match the first moment of S, we want to change S_i into $\tilde{S}_i$ s.t.

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{S}_i = e^{rT}S(0)$$

Possible ways to accomplish the above transformation:

- Method 1:

$$\tilde{S}_i = S_i + e^{rT}S(o) - \hat{S}(n)$$

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{S}_i = \frac{1}{n}\sum_{i=1}^{n} S_i + \frac{1}{n}\cdot n\big(e^{rT}S(0) - \hat{S}(n)\big) = \hat{S}(n) + e^{rT}S(0) - \hat{S}(n) = e^{rT}S(0)$$

Remark: Big problem here is that such $\tilde{S}_i$ could have negative values.

- Method 2:

$$\tilde{S}_i = S_i \frac{e^{rT}S(0)}{\hat{S}(n)}$$

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{S}_i = \frac{1}{n}\sum_{i=1}^{n} S_i \frac{e^{rT}S(0)}{\hat{S}(n)} = e^{rT}S(0)$$

If we use moment matching method, put-call parity will be satisfied.

$$\begin{aligned}
\hat{c}(n) &= \frac{1}{n}\sum_{i=1}^{n} c_i = \frac{1}{n}\sum_{i=1}^{n} e^{-rT}\max\big(S_i - K, 0\big) \\
\hat{p}(n) &= \frac{1}{n}\sum_{i=1}^{n} p_i = \frac{1}{n}\sum_{i=1}^{n} e^{-rT}\max\big(K - S_i, 0\big) \\
\hat{c}(n) - \hat{p}(n) &= \frac{1}{n}e^{-rT}\sum_{i=1}^{n}\Big(\max\big(S_i - K, 0\big) - \max\big(K - S_i, 0\big)\Big) \\
&= \frac{1}{n}e^{-rT}\sum_{i=1}^{n}(S_i - K, 0) = e^{-rT}\hat{S}(n) - e^{-rT}K \\
&= S(0) - e^{-rT}K \quad \text{if } \hat{S}(n) = e^{rT}S(0)
\end{aligned}$$

Remark: We can use control variate method and moment matching method together.

6.3 Antithetic Variables

Reducing variance by introducing negative dependence between pairs of replication.

- Generate $U_1, U_2, \ldots, U_n \sim Unif(0,1)$;
  Also use $1-U_1, 1-U_2, \ldots, 1-U_n \sim Unif(0,1)$;
- Inverse Transform, note that $N(-a) = 1 - N(a)$
  $Z_{1,i} = F^{-1}(U_i) \to X_i$
  $Z_{2,i} = F^{-1}(1-U_i) = -Z_{1,i} \to Y_i$
  Remark: **X_i and Y_i have the same distribution but not independent.**

$\star$ **Monte Carlo,** $\hat{X}(n) = \frac{1}{n}\sum\limits_{i=1}^{n} X_i$;

$\star$ **Antithetic Variables,** $Y_{AV}(n) = \frac{1}{n}\sum\limits_{i=1}^{n} \frac{X_i+Y_i}{2}$

$$\begin{aligned}
var\big(\hat{X}(2n)\big) &= \frac{var(X)}{2n} = \frac{\sigma_X^2}{2n} \\
var\big(Y_{AV}(n)\big) &= \frac{var(X+Y)}{4n} \\
&\qquad var(X+Y) = \sigma_X^2 + 2\sigma_X\sigma_Y\rho_{XY} + \sigma_Y^2 = 2\sigma_X^2(1+\rho_{XY}) \quad \text{since } \sigma_X = \sigma_Y \\
var\big(Y_{AV}(n)\big) &= \frac{\sigma_X^2(1+\rho_{XY})}{2n}
\end{aligned}$$

We want

$$\begin{aligned}
& var\big(Y_{AV}(n)\big) \;\leq\; var\big(\hat{X}(2n)\big) \\
\Rightarrow\quad & \frac{\sigma_X^2}{2n} \;\leq\; \frac{\sigma_X^2(1+\rho_{XY})}{2n} \\
\Rightarrow\quad & \rho_{XY} < 0
\end{aligned}$$