APMA 1720 Computational Project

(1) a) ① $E[Z_1 \mid Z_2] = \mu_1 + \rho \frac{\sigma_1}{\sigma_2}(Z_2 - \mu_2)$

$\rightarrow \mu_1 = \mu_2 = 0 \quad , \quad \sigma_1 = \sigma_2 = 1$

$\therefore E[Z_1 \mid Z_2 = z] = (0) + \rho \frac{(1)}{(1)}((z) - (0)) = \rho z$

② $Var(Z_1 \mid Z_2) = \sigma_1^2(1-\rho^2)$

$\therefore Var(Z_1 \mid Z_2 = z) = (1)^2(1-\rho^2) = (1-\rho^2)$

$\therefore Z_1 \mid Z_2 = z \sim N(\rho z, 1-\rho^2)$

b) Given $Z_2 = z \longrightarrow Y_T$ becomes constant ① $C_2 Y_T - K < 0$; ② $C_2 Y_T - K \geq 0$

① $C_2 Y_T - K < 0$

$c_1 X_T = c_1 X_0 \, e^{(r - \frac{1}{2}\sigma_1^2)T + \sigma_1 \sqrt{T} Z_1}$ ← Find lognormal distribution

$\log(c_1 X_T) = \log(c_1 X_0) + (r - \frac{1}{2}\sigma_1^2)T + \sigma_1 \sqrt{T} Z_1$

$E[Z_1 \mid Z_2 = z] = \rho z$

$\rightarrow$ Mean : $E[\log(c_1 X_T) \mid Z_2 = z] = \underline{\log(c_1 X_0) + (r - \frac{1}{2}\sigma_1^2)T + \sigma_1 \sqrt{T} \rho z}$

$\rightarrow$ Variance: $Var(\log(c_1 X_T) \mid Z_2 = z) = 0 + Var(\sigma_1 \sqrt{T} Z_1 \mid Z_2 = z) = \sigma_1^2 T \, Var(Z_1 \mid Z_2 = z)$

$= \underline{\sigma_1^2 T (1-\rho^2)}$

$\therefore$ Price $= E\left[e^{-rT}(c_1 X_T + \underbrace{c_2 Y_T - K}_{<0})^+ \mid Z_2 = z\right]$

$= E\left[e^{-rT}(c_1 X_T - (\underbrace{K - c_2 Y_T}_{>0}))^+ \mid Z_2 = z\right]$ ← Takes the form $E[(S-K)^+]$

$= e^{-rT} \int_{-\infty}^{\infty} (e^x - (K - c_2 Y_T))^+ \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx$

$= e^{-rT} \int_{\log(K - c_2 Y_T)}^{\infty} (e^x - (K - c_2 Y_T)) \cdot \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx$

$$= e^{-rT} \left\{ \int_{\log(K - c_2 Y_T)}^{\infty} e^x \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx - (K - c_2 Y_T) \int_{\log(K - c_2 Y_T)}^{\infty} \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \right\}$$

$\rightarrow$ Let $\bar{K} = \dfrac{\log(K - c_2 Y_T) - \mu}{\sigma}$ $\qquad x = \mu + \sigma z$

$$\rightarrow \int_{\bar{K}}^{\infty} e^{\mu} e^{\sigma z} \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{z^2}{2}} \sigma \, dz \; - \; (K - c_2 Y_T) \int_{\bar{K}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz$$

$$= e^{\mu + \frac{1}{2}\sigma^2} \int_{\bar{K}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{(z-\sigma)^2}{2}} dz \; - \; (K - c_2 Y_T) \int_{\bar{K}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz$$

$$= e^{\mu + \frac{1}{2}\sigma^2} \int_{\bar{K}-\sigma}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz \; - \; (K - c_2 Y_T) \int_{\bar{K}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz$$

$$= e^{\mu + \frac{1}{2}\sigma^2} \left[1 - \Phi(\bar{K} - \sigma) \right] \; - \; (K - c_2 Y_T)\, \Phi(-\bar{K})$$

$$= e^{\mu + \frac{1}{2}\sigma^2}\, \Phi(\sigma - \bar{K}) \; - \; (K - c_2 Y_T)\, \Phi(-\bar{K})$$

$c_2 Y_T - K < 0$:

$$\therefore \text{Price} = \underbrace{e^{-rT}}_{\frac{1}{R}} \left[e^{\mu + \frac{1}{2}\sigma^2}\, \Phi(\sigma - \bar{K}) - (K - c_2 Y_T)\, \Phi(-\bar{K}) \right]$$

where $\quad \mu = \log(c_1 X_0) + (r - \frac{1}{2}\sigma_1^2) T + \sigma_1 \sqrt{T} \rho z$

$\sigma^2 = \sigma_1^2 T (1 - \rho^2)$

$\bar{K} = \dfrac{\log(K - c_2 Y_T) - \mu}{\sigma}$

② $c_2 Y_T - K \geq 0$

this quantity will always be positive when $c_2 Y_T - K \geq 0$

$$\therefore \text{Price} = E\left[e^{-rT}\big(\underbrace{c_1 X_T + c_2 Y_T - K}_{>0}\big)^+ \,\Big|\, Z_2 = z\right]$$

$$= E\left[e^{-rT}\left(c_1 X_T + c_2 Y_T - K\right) \,\Big|\, Z_2 = z\right]$$

$$= E\left[e^{-rT} c_1 X_T \,\Big|\, Z_2 = z\right] + E\Big[\underbrace{e^{-rT}\left(c_2 Y_T - K\right)}_{\text{constant}} \,\Big|\, Z_2 = z\Big]$$

$$= E\left[e^{-rT} c_1 X_T \,\Big|\, Z_2 = z\right] + e^{-rT}\left(c_2 Y_T - K\right)$$

Expectation of lognormal variable

$$= e^{-rT} \cdot e^{\mu + \frac{1}{2}\sigma^2} + e^{-rT}\left(c_2 Y_0\, e^{(r - \frac{1}{2}\sigma_2^2)T + \sigma_2\sqrt{T} z} - K\right)$$

$c_2 Y_T - K \geq 0$:

From ①

$$\therefore \text{Price} = e^{-rT}\Big[e^{\mu + \frac{1}{2}\sigma^2} + c_2 \underbrace{Y_0\, e^{(r-\frac{1}{2}\sigma_2^2)T + \sigma_2\sqrt{T} z}}_{Y_T} - K\Big]$$

where $\mu = \log(c_1 X_0) + (r - \frac{1}{2}\sigma_1^2)T + \sigma_1\sqrt{T}\rho z$

$\sigma^2 = \sigma_1^2 T (1 - \rho^2)$

c)

```
function [price, SE] = basketCallPricePlain(r, sigma1, sigma2, rho, X0, Y0, T, K, c1, c2, n)
    X = zeros(n, 1);

    for i = 1:n
        Z1 = normrnd(0, 1);
        Z2 = normrnd(0, 1);

        U1 = 1 * Z1;
        U2 = rho * Z1 + sqrt(1 - rho^2) * Z2;

        XT = X0 * exp((r - 0.5 * sigma1^2) * T + sigma1 * sqrt(T) * U1);
        YT = Y0 * exp((r - 0.5 * sigma2^2) * T + sigma2 * sqrt(T) * U2);

        payoff = max(c1 * XT + c2 * YT - K, 0);
        X(i) = exp(-r * T) * payoff;
    end

    price = mean(X);
    SE = sqrt((mean(X .* X) - price^2) / (n - 1));
end
```

```matlab
function [price, SE] = basketCallPriceConditioning(r, sigma1, sigma2, rho, X0, Y0, T, K, c1, c2, n)
    X = zeros(n, 1);

    for i = 1:n
        % Generate Z2 = z so we know YT
        Z2 = normrnd(0, 1);
        YT = Y0 * exp((r - 0.5 * sigma2^2) * T + sigma2 * sqrt(T) * Z2);

        if (c2 * YT - K) < 0
            mu = log(c1 * X0) + (r - 0.5 * sigma1^2) * T + sigma1 * sqrt(T) * rho * Z2;
            sigma = sqrt(sigma1^2 * T * (1 - rho^2));
            Kbar = (log(K - c2 * YT) - mu) / sigma;

            payoff = exp(mu + 0.5 * sigma^2) * normcdf(sigma - Kbar) - (K - c2 * YT) * normcdf(-Kbar);
            X(i) = exp(-r * T) * payoff;
        end
        if (c2 * YT - K) >= 0
            mu = log(c1 * X0) + (r - 0.5 * sigma1^2) * T + sigma1 * sqrt(T) * rho * Z2;
            sigma = sqrt(sigma1^2 * T * (1 - rho^2));

            payoff = exp(mu + 0.5 * sigma^2) + c2 * YT - K;
            X(i) = exp(-r * T) * payoff;
        end
    end

    price = mean(X);
    SE = sqrt((mean(X .* X) - price^2) / (n - 1));
end
```

```matlab
r = 0.1;
sigma1 = 0.2;
sigma2 = 0.3;
rho = 0.7;
X0 = 50;
Y0 = 50;
T = 1;
K = 55;
c1 = 0.5;
c2 = 0.5;
n = 10000;

[plain_price, plain_SE] = basketCallPricePlain(r, sigma1, sigma2, rho, X0, Y0, T, K, c1, c2, n)
[cond_price, cond_SE] = basketCallPriceConditioning(r, sigma1, sigma2, rho, X0, Y0, T, K, c1, c2, n)
```

```
plain_price =

    4.6477


plain_SE =

    0.0780


cond_price =

    4.7533


cond_SE =

    0.0746
```

(2) a) $dS_t = r S_t \, dt + \theta_t S_t \, dW_t$ $\quad$ $d\theta_t = \alpha(\Theta - \theta_t)\, dt + \beta \, dB_t$

$V = e^{-rT} E[(S_T - K_1)^+ - (S_T - K_2)^+]$ $\quad$ $0 < K_1 < K_2$

$\rightarrow$ Let $Y_t = \log S_t$

$\therefore dY_t = (r - \frac{1}{2}\theta_t^2)\, dt + \theta_t \, dW_t$

① $Y_{t_{i+1}} = Y_{t_i} + (r - \frac{1}{2}\theta_{t_i}^2)(t_{i+1} - t_i) + \theta_{t_i}\sqrt{t_{i+1} - t_i}\, Z_{t_i}$, $Z_{t_i} \sim N(0,1)$

② $\theta_{t_{i+1}} = \theta_{t_i} + \alpha(\Theta - \theta_{t_i})(t_{i+1} - t_i) + \beta\sqrt{t_{i+1} - t_i}\, X_{t_i}$, $X_{t_i} \sim N(0,1)$

$\therefore S_T = e^{Y_T}$

$\longrightarrow$ Assume n simulations

$\therefore$ Average Payoff $= \frac{1}{n}\sum_{i=1}^{n}[(S_T - K_1)^+ - (S_T - K_2)^+]$

price $= e^{-rT} \cdot$ Average Payoff

$\therefore$ price $= \frac{1}{ne^{rT}}\sum_{i=1}^{n}[(S_T - K_1)^+ - (S_T - K_2)^+]$

b) ① For the control variates method, I set the constant volatility σ equal to $\Theta = 0.2$. In theory, σ can be arbitrary, but we select a typical value of θ to achieve a higher level of variance reduction. Therefore, it is reasonable to let $\sigma = \Theta$, which is the long run average of θ.

(2) The control variable used was the discounted payoff of an artificial vertical spread. This process $\bar{Y}$ is nearly identical to $\hat{Y}$ but differs with the use of a constant volatility σ which we have defined as equal to Θ. Since we are looking at expected payoffs of a vertical spread, choosing a control variable strongly correlated with the outcome variable allows us to arrive at a more stable and accurate estimate. This is because the variability common to both the main simulation and the control variable will cancel out, leading to a more precise estimate.

```
function [price, SE] = straightEulerVerticalSpread(r, a, Theta, beta, S0, theta0, T, K1, K2, rho, m, n)
    dt = T / m;
    X = zeros(n, 1);

    for k = 1:n
        Yt = log(S0);
        theta_t = theta0;

        for i = 1:m
            % Generate IID samples Z and U from N(0,1)
            Z = normrnd(0, 1);
            U = normrnd(0, 1);
            R = rho * Z + sqrt(1 - rho^2) * U;

            % Update time processes
            Yt = Yt + (r - 0.5 * theta_t^2) * dt + theta_t * sqrt(dt) * Z;
            theta_t = theta_t + a * (Theta - theta_t) * dt + beta * sqrt(dt) * R;
        end

        % Store the discounted payoff
        ST = exp(Yt);
        payoff = max(ST - K1, 0) - max(ST - K2, 0);
        X(k) = exp(-r * T) * payoff;
    end

    price = mean(X);
    SE = sqrt((mean(X .* X) - price^2) / (n - 1));
end
```

```
function [price, SE] = straightEulerControlVerticalSpread(r, a, Theta, beta, S0, theta0, T, K1, K2, rho, m, n)
    sigma = Theta;  % set sigma = Theta (long run average of theta)
    dt = T / m;

    X = zeros(n, 1);
    Q = zeros(n, 1);

    for k = 1:n
        Y_hat = log(S0);
        Y_bar = Y_hat;
        theta_hat = theta0;

        for i = 1:m
            % Generate IID samples Z and U from N(0,1)
            Z = normrnd(0, 1);
            U = normrnd(0, 1);
            R = rho * Z + sqrt(1 - rho^2) * U;

            % Update the time processes and control variable
            Y_hat = Y_hat + (r - 0.5 * theta_hat^2) * dt + theta_hat * sqrt(dt) * Z;
            Y_bar = Y_bar + (r - 0.5 * sigma^2) * dt + sigma * sqrt(dt) * Z;
            theta_hat = theta_hat + a * (Theta - theta_hat) * dt + beta * sqrt(dt) * R;
        end

        ST = exp(Y_hat);
        STbar = exp(Y_bar);

        % Payoffs of actual process and control variable
        payoff_actual = max(ST - K1, 0) - max(ST - K2, 0);
        payoff_control = (max(STbar - K1, 0) - max(STbar - K2, 0));

        % Find expected value of the vertical spread option using BLS
        bls_call_1 = BlackScholes(S0, K1, r, T, sigma);
        bls_call_2 = BlackScholes(S0, K2, r, T, sigma);
        bls_spread = bls_call_1 - bls_call_2;

        % Store discounted actual payoff and variance of the discounted
        % control variable
        X(k) = exp(-r * T) * payoff_actual;
        Q(k) = exp(-r * T) * payoff_control - bls_spread;
    end

    % Let b = 1 per the given assumption in the problem
    b = 1;
    H = X - b * Q;

    price = mean(H);
    SE = sqrt((mean(H .* H) - price^2) / (n - 1));
end


function price = BlackScholes(S0, K, r, T, sigma)
    % Black-Scholes formula for call option
    tmp = log(K / S0) / (sigma * sqrt(T)) + (0.5 * sigma - r / sigma) * sqrt(T);
    price = S0 * normcdf(sigma * sqrt(T) - tmp) - exp(-r * T) * K * normcdf(-tmp);
end
```

```
r = 0.1;
a = 3;
Theta = 0.2;
beta = 0.1;
S0 = 20;
theta0 = 0.25;
T = 1;
K1 = 20;
K2 = 22;
rho = 0.5;
m = 50;
n = 10000;

[straight_price, straight_SE] = straightEulerVerticalSpread(r, a, Theta, beta, S0, theta0, T, K1, K2, rho, m, n)
[control_price, control_SE] = straightEulerControlVerticalSpread(r, a, Theta, beta, S0, theta0, T, K1, K2, rho, m, n)
```

```
straight_price =

    0.9652


straight_SE =

    0.0085


control_price =

    0.9624


control_SE =

    0.0024
```