# Q1) Monte Carlo Algorithm for Rainbow Option

## a)

For $d-$correlated assets the prices at maturity are $(S_T^{(j)})_{i=1,\ldots,d}$. Under $\mathbb{Q}$, $T=1$, and the Black Scholes model, the solution to the price at maturity for all $j=1,2,\ldots,d$ is:

$$S_T^{(j)} = S_0^{(j)} \exp(r - (\sigma^{(j)})^2/2 + \sigma^{(j)} W_T^{(j)}$$

Where for each stock $j=1,2,\ldots,d$:

- $S_0^{(j)}$ is its initial price of stock
- $\sigma^{(j)}$ is its standard deviation
- $W_T^{(j)}$ is is the brownian motion at maturity such that for another stock $i$ we have correlated brownian motion processes $Cor(W_t^{(j)}, W_t^{(i)}) = \rho$.

Define the correlation matrix.

$$\Sigma = \begin{bmatrix} 1 & \rho & \ldots & \rho \\ \rho & 1 & \ldots & \rho \\ \vdots & \vdots & \ddots & \vdots \\ \rho & \rho & \ldots & 1 \end{bmatrix} \in \mathbb{R}^{d\times d}$$

We want $X = (W_T^{(1)}, \ldots, W_T^{(d)}) \sim N_d(0, \Sigma)$, so let $Z^j \sim N(0,1)$ for $j = 1, \ldots d$ such that for a matrix $A \in \mathbb{R}^{d\times d}$ satisfying $\Sigma = AA^T$ we have $X = AZ$ (equal in distribution). Let $A$ be the **Cholesky factor** of $\Sigma$.

---

**Algorithm**

1. Given $S_0^{(j)}, \sigma^{(j)}$ for all assets $j = 1, \ldots, d$
2. Initialize $V_T :: \text{float}[d] \leftarrow 0 \in \mathbb{R}^d$
3. Compute $A \leftarrow \text{cholesky}(\Sigma)$
4. for $i = 1, \ldots, n$ do:
    - Sample $Z_1, \ldots, Z_d \sim^{iid} N(0,1)$ and let $Z = (Z_1, \ldots, Z_d)$
    - Let $X = AZ \in \mathbb{R}^d$ (matrix multiplication)
    - Let $S_{MAX} = -1$
    - for $j = 1, \ldots, d$:
        - Let $S_T^{(j)} = S_0^{(j)} exp\{r - (\sigma^{(j)})^2/2 + \sigma^{(j)} X[j]\}$
        - If $(S_T^{(j)} > S_{MAX}) : S_{MAX} = S_T^{(j)}$
    - Set $V_T[i] = \exp(-r) \times \max(S_{MAX} - K, 0)$
5. return $\text{mean}(V_T) \pm \frac{1.96\times\text{sd}(V_T)}{\sqrt{n}}$

**b)**

```
getAmatrix <-function(rho, d){
  ## Construct correlation matrix Sigma for d-RV's with pairwise correlation rho.
  Sigma <- matrix(rep(rho, d^2), nrow = d, ncol = d) # populate with rho
  diag(Sigma) <- 1 # Set diagonal equal to 1
  ## Get matrix A such that AA^T = Sigma
  t(chol(Sigma))
}

partA <- function(d, S0s, sigmas, rho, K, r, n) {
  A <- getAmatrix(rho, d)
  vals <- rep(0, n) # values at maturity for each simulation
  for (i in 1:n){
    Z <- rnorm(d)
    X <- A%*%Z
    Smax <- -1
    for (j in 1:d){
      STj <- S0s[j] * exp((r-sigmas[j]^2/2) + sigmas[j]*X[j])
      if (STj > Smax) {
        Smax = STj
      }
    }
    vals[i] <- max(exp(-r)*(Smax-K), 0)
  }
  MCest <- mean(vals)
  MCsd <- sd(vals)
  alpha <- qnorm(.975)
  lo <- MCest-alpha*MCsd/sqrt(n)
  hi <- MCest+alpha*MCsd/sqrt(n)
  ans <- c(MCest, lo, hi)
  ## names(ans) <- c("estimate", "low", "high")
  return(ans)
}

partB <- function(rho) {
  set.seed(123)
  ans<-partA(d= 2, S0s = c(100, 100), sigmas = c(.2, .3), rho=rho, K=110, r=0.05, n=100000)
  return(ans[1])
}
```

```r
rhos <- seq(-0.9, 0.9, 0.1)
estimates <- sapply(rhos, partB)
df <- data.frame(rhos, estimates)
print(df)
```

```
##    rhos estimates
## 1  -0.9  16.10186
## 2  -0.8  16.04699
## 3  -0.7  15.95444
## 4  -0.6  15.83054
## 5  -0.5  15.68365
## 6  -0.4  15.51868
## 7  -0.3  15.33255
## 8  -0.2  15.12776
## 9  -0.1  14.90380
## 10  0.0  14.66019
## 11  0.1  14.39500
## 12  0.2  14.10670
## 13  0.3  13.79058
## 14  0.4  13.44057
## 15  0.5  13.05431
## 16  0.6  12.62352
## 17  0.7  12.14263
## 18  0.8  11.59120
## 19  0.9  10.93301
```

```r
plot(x = rhos, y = estimates, main = "Correlation vs MC price estimates",
     ylab = "estimate", xlab = "pairwise correlation")
```