CS 476 Assignment 4

Jeongseop Yi (Patrick), j22yi

Q4

a)

The vector $x = (x_1, x_2)$ denote units of the underlying and the bond to hold. As we are running Monte Carlo simulations with M simulations, there are total of M stock prices at time T from each simulations.

We would like to know the optimal hedging portfolio x^*. The portfolio is constructed using the underlying and the bond with weights of x_1, x_2 at time T. For each simulation i, the portfolio $\Pi_i = S_i^T \cdot x_1 + e^{rT} \cdot x_2$. The bond price is compounded continuously at rate r until time T.

Therefore, the matrix A can be constructed as $A_i = \begin{bmatrix} S_i^T & e^{rT} \end{bmatrix}$ where $i = 1 \ldots M$

The matrix b corresponds to the payoff in the expected quadratic error $\mathrm{E}((\Pi_T - \text{payoff})^2)$. Therefore $b_i = \begin{bmatrix} \text{ButterflyPayoff}(S_i^T) \end{bmatrix}$ where $i = 1, 2, \ldots, M$ which denotes payoff at time T for each M simulation.

$$A = \begin{bmatrix} S_1^T & e^{rT} \\ S_2^T & e^{rT} \\ S_3^T & e^{rT} \\ \vdots & \vdots \\ S_M^T & e^{rT} \end{bmatrix} b = \begin{bmatrix} \text{ButterflyPayoff}(S_1^T) \\ \text{ButterflyPayoff}(S_2^T) \\ \text{ButterflyPayoff}(S_3^T) \\ \vdots \\ \text{ButterflyPayoff}(S_M^T) \end{bmatrix}$$

The butterfly payoff function is given in Q1.

b)

The pseudocode to calculate A, b, c is as follows:

```
def Q4B(S0, M, N, T, r, ...):
    A: M x 2 matrix, b: M x 1 matrix, S: M x N matrix, c: 2 x 1 Matrix
    # MC_simulation returns M x N matrix for all M simulations for N timesteps
    S = MC_simulation(S0, M, N, T, r, ...)
    # First column of A is the stock prices at time T
    # Second column of A is the bond prices at time T
    A[:, 0] = S[-1], A[:, 1] = e^(r*T)
    # b is the vector of butterfly payoffs for stock prices at time T
    b = ButterflyPayoff(A[0])
    # c is just (S0, 1)
    c = (S0, 1)

    return A, b, c
```

c)

The Lagranigan of the optimization problem is as follows:

$$L(x, v) = \frac{1}{M}||Ax - b||_2^2 + v^T(c^T x - V_0^B)$$

As x^* and v^* are the optimal solution of the optimization problem and the Lagrangian multiplier associated with the problem, both x^* and v^* must satisfy KKT.

We will look at the gradient requirement of KKT to find the values of x^* and v^*.

By KKT, the following holds for the Lagrangian at (x^*, v^*):

$$\nabla_x L(x^*, v^*) = 0$$

$$\nabla_x L(x^*, v^*) = \frac{2}{M}(Ax^* - b)^T A + (v^*)^T(c^T) = 0$$
$$2(Ax^* - b)^T A + M(v^*)^T(c^T) = 0$$
$$(Ax^* - b)^T A = \frac{-M}{2}(v^*)^T(c^T)$$
$$(x^*)^T A^T A - b^T A = \frac{-M}{2}(v^*)^T(c^T)$$
$$(x^*)^T A^T A = \frac{-M}{2}(v^*)^T(c^T) + b^T A$$
$$(x^*)^T = (\frac{-M}{2}(v^*)^T(c^T) + b^T A)(A^T A)^{-1}$$
$$x^* = ((\frac{-M}{2}(v^*)^T(c^T) + b^T A)(A^T A)^{-1})^T$$

Is $A^T A$ invertible? Yes!

$$A^T A = \begin{bmatrix} \Sigma(S_i^T)^2 & e^{rT}\Sigma S_i^T \\ e^{rT}\Sigma S_i^T & Me^{2rT} \end{bmatrix}$$

where $i = 1...M$.

Then determinant of the matrix is:

$$det(A^T A) = Me^{2rT}\Sigma(S_i^T)^2 - e^{2rT}(\Sigma S_i^T)^2 = (M-1)e^{2rT}\Sigma(S_i^T)^2 + ... \neq 0$$

where $i, j = 1...M$ and $i \neq j$, as all stock prices at time T from M simulations cannot be all 0 at the same time.

Find v^* in terms of V_0^B by plugging in above x^* to the constraint requirement $c^T x^* - V_0^B = 0$

Once v^* is found, the value of x^* can be also calculated using $x^* = ((\frac{-M}{2}(v^*)^T(c^T) + b^T A)(A^T A)^{-1})^T$.