CS 476/676 Assignment 3

Winter 2023

Instructor: Yuying Li   Office: DC3623   *yuying@uwaterloo.ca*

Lecture Times: MW 11:30-12:50   MC 2035

Yuying Li: OH: Thurs 3-4pm

TA: Mohammad Shirazi   DC 3594 (SciCom)   mmkshira@uwaterloo.ca

OH: 10th March: 2-3 PM   DC 3594 (SciCom)

OH: 14th March: 1-2PM   DC 3594 (SciCom)

OH: 16th March: 1-2 PM   DC 3594 (SciCom)

My Web Site: http://www.cs.uwaterloo.ca/~yuying

Due: 11am Mar 20, 2023

Programming Questions

IMPORTANT: In this and in future assignments, most of the marks for programming questions are allocated for explanations of algorithms (e.g. pseudo-code) and discussion of results. If all you hand in is the listing of the "Raw Code" or "Raw Output" by itself, you will get poor marks. All coding should be done in Matlab. All the plots should be appropriately labeled. You should submit all Matlab code used in your assignment. Be sure to document (i.e. add liberal comments) your code. The TA will take off marks for poor documentation.

1 [(5 marks)] (Numerical Scheme for SDE)

Consider the following mean reverting process for variance:

$$dv_t = -\lambda(v_t - \bar{v})dt + \eta\sqrt{v_t}dZ_t$$

where $\lambda > 0$, $\bar{v}$, and $\eta > 0$ are given constants.

(a) Given v_n at t_n, write down the Euler-Maruyama formula for computing v_{n+1} at t_{n+1}. Can v_{n+1} become negative? If yes, provide conditions on the standard normal sample in Euler-Maruyama formula that leads to negative v_{n+1}. Otherwise, explain.

(b) Given v_n at t_n, write down the Milstein method for computing v_{n+1}. Assumming $v_n = 0$, provide an inequality using parameters $\lambda, \bar{v}, \eta$, which guarantees that v_{n+1} is always positive.

.

2 [(12 marks)] (Numerical Scheme for correlated SDEs)

A long term European option on a risky asset S_t has the payoff $P(S_T)$ at the expiry T. Assume that the underlying price process S_t, under the risk neutral probability, and the interest rate r_t process satisfy equations below

$$\frac{dS_t}{S_t} = r_t dt + \sigma_t dZ_t^{(0)}$$

$$d\sigma_t = k_1(\bar{\sigma} - \sigma_t)dt + \eta_1\sqrt{\sigma_t}dZ_t^{(1)}$$

$$dr_t = k_2(\bar{r} - r_t)dt + \eta_2\sqrt{r_t}dZ_t^{(2)}$$

$$\mathbf{E}^Q(dZ_t^{(0)}dZ_t^{(1)}) = \rho\ dt$$

where $k_1, k_2, \bar{r}, \bar{\sigma}, \eta_1, \eta_2, \rho$ are given positive constants. In addition, the initial price $S_0 > 0$, interest rate $r_0 > 0$, and volatility $\sigma_0 > 0$ are given. Furthermore, $dZ^{(0)}$ and $dZ^{(2)}$ are independent and $dZ^{(1)}$ and $dZ^{(2)}$ are independent.

Table 1: Data for Hedging Simulations Using a Binomial Lattice

σ	.20
r	.03
μ	0.15
Time to expiry T	1 year
Initial asset price $S(0)$	$100
Strike Price K_1	$0.9S(0)$
Strike Price K_3	$1.1S(0)$
N	250

(a) (2 marks) Let $dZ^{(i)} = \phi^{(i)}\sqrt{dt}$ and $\phi = [\phi^{(0)}, \phi^{(1)}, \phi^{(2)}]$. What is the covariance matrix for ϕ? Describe how to generate iid samples for $\phi^{(i)}, i = 0, 1, 2$, assuming that you are given a function **randn** which generates independent samples from the standard normal.

(b) (2 marks) Let $t_n = n\Delta t$, $n = 0, 1, \cdots, N$, where $N\Delta t = T$. Assuming that $r(t_n), \sigma(t_n), S(t_n)$ are given, describe how to compute $r(t_{n+1}), \sigma(t_{n+1}), S(t_{n+1})$ using the Euler's formulae?

(c) (8 marks) The fair value V_0 of this European option can be computed based on the risk neutral valuation, i.e.,

$$V_0 = \mathbf{E}^Q \left(e^{-\int_0^T r(t)dt} P(S_T) \right)$$

Write a pseudo code, using M Monte Carlo simulations and N time steps, $N\Delta t = T$, to determine the fair value of the option at $t = 0$ based on (a) and (b). You can ignore handling of potential negative rates and prices.

.

3 [(20 marks)](Compute Delta from a Binomial Lattice)

Assume that the (no dividend) underlying price follows

$$\frac{dS_t}{S_t} = \mu dt + \sigma dZ_t \tag{1}$$

where Z_t is a standard Brownian, μ and σ are constants.

Consider the European Butterfly Spreads, whose payoff is defined by strike prices $K_1 < K_2 < K_3$, where $K_2 = \frac{1}{2}(K_1 + K_3)$, as follows:

$$\text{payoff}(S_T) = \begin{cases} 0 & \text{if } S_T \leq K_1 \\ S_T - K_1 & \text{if } K_1 < S_T \leq K_2 \\ K_3 - S_T & \text{if } K_2 < S_T \leq K_3 \\ 0 & \text{if } S_T > K_3 \end{cases}$$

You are to conduct hedging effectiveness analysis for this power option using MC simulations of (1) for the underlying price so that there is no time discretization error in the simulated price.

In addition, you will compute hedging positions by modifying your code for option pricing under a binomial lattice in Assignment 2.

Assume that the initial option price is computed from your binomial model. Specifically,

(a) Write a Matlab function $[V_0, S, \delta] = \mathbf{binomialDeltaButterfly}(S_0, K_1, K_3, r, \sigma, T, N)$ which returns, the initial option value V_0, S and δ, representing the underlying price and delta hedging positions on the binomial lattice nodes over equally spaced N time periods in $[0, T]$. The interest rate r and volatility σ are constants.

(b) Now consider a vector of underlying price S (not necessarily equal to any binomial lattice price, e.g., from simulations), using linear interpolation to determine the corresponding hedging positions $\delta(S)$ from the binomial lattice price vector S^n and delta δ^n as follows: When a simulated price in S does not equal to any lattice price S^n, interpolate using the hedging values at lattice nodes at each rebalancing time to approximate the hedging position (you can use matlab function **interp1**). To avoid potentially large extrapolation error, reset the hedging position equal to that of the nearest binomial price S^n, if an underlying price exceeds the range of the price S^n. Write a Matlab function $\delta=$**interpDelta**(δ^n, S^n, S) to implement this interpolation.

(c) Using the data in Table 1, **binomialDeltaButterfly**, and **interpDelta**, write a matlab script to compute hedging positions at time t_n, $n = 0.8N$, for a European Butterfly $S = \text{linspace}(0.6S_0, 1.4S_0, 100)$ and plot the computed hedge position δ against S.

(d) Using the parameter values given in Table 1, determine the hedging error using discrete delta hedging for the Butterfly Spread option . Plot the histogram of the relative hedging error P&L, for no hedging, $n = 1 : 1 : N$ (daily), $n = 5 : 5 : N$ (weekly), $n = 20 : 20 : N$ (monthly) respectively. assuming hedging positions are computed using **binomialDeltaButterfly** and **interpDelta**,. The Matlab functions $histc$, bar might prove useful. Use at least 50 bins in your histogram. Comment on your observations.

(e) Write a Matlab function [var,cvar]=dVaRCVaR(P&L, β) which returns VaR and CVaR for a discrete P&L distribution with M independent samples using the procedure described. In Matlab, ordering can be done using **sort**. For the Butterfy, compute and report in a table, mean, standard deviation, VAR (95%), CVAR (95%), of $P\&L$ for no hedging, rebalancing monthly, weekly, and daily. Discuss how hedging performance changes with the rebalancing frequency.

(f) In general, a hedging rebalancing time t_n^{rb} may not coincide with the discrete time t_n of the lattice. How would you change your code to implement the hedging analysis?

.

4 [(10 marks)]. (Trading simulation)

You think that you have discovered an arbitrage dynamic trading strategy. Does it work?

Simulation can be used to analyze your strategy. Consider the following dynamic trading strategy

Let's assume that it is possible to trade continuously in time, without transaction cost. For simplicity, we also assume that the interest $r = 0$.

Let S_t and C_t denote time t values of the stock and a European call with strike K and expiry T.

Suppose that today you sell the European call option with strike K and expiry T. Assume that $S_0 < K$, i.e. the initial asset price is less than K.

Assume that you then implements the following simple hypothetic hedging strategy: As soon as $S_t > K$, you borrow from cash account and buy the stock at t. As soon as $S_t < K$, you sell the stock at t.

Let $\delta_t S_t + B_t$ denote the value of the hedging portfolio of the underlying and bond. At $t = 0$, the cash account is set such that $-C_0 + \delta_0 S_0 + B_0 = 0$.

Assume that the interest $r = 0$.

(a) What is the value of B_0? Write down a mathematical expression for the writer's trading strategy δ_t, $0 \le t < T$.

(b) Under the above continuous trading assumption, the value of hedging portfolio $\delta_t S_t + B_t$ is either $S_t - K + C_0$ or C_0 for any $0 < t < T$. When does the value $\delta_t S_t + B_t$ equals to $S_t - K + C_0$? When does the value $\delta_t S_t + B_t$ equals to C_0? Explain your answers.

(c) What is the relative P&L for the writer $(-C_T + \delta_T S_T + B_T)/C_0$? What are the mean and variance of this relative P&L?

3

(d) The answers to the above questions seem to suggest the existence of an arbitrage. Perform the following computation to investigate whether this is true. Assume that the underlying follows the following Black-Scholes model. Simulate writer's strategy above, assuming $S_0 = 95$, $K = 105$ and other parameters in Table 1. Compute the relative P&L $(-C_T + \delta_T S_T + B_T)/C_0$ by MC simulations. Assume C_0 equals the initial call value computed by Black-Scholes formula. Use about $80,000$ simulations. Use $100, 200, 400, 800$ rebalancing times. Show a table of results (mean, standard deviation, 95% VAR and CVAR) versus number of rebalancing times. Generate plot of the probability density of the relative P&L for the case of 800 rebalancing times.

What do you observe about the mean and variance of the hedging error?

(e) **Bonus**. Explain the discrepancy in the variance hedging error between the theoretical value under the continuous hedging assumption and the computed values using MC simulations.

.

5 [(12 marks)] (MC Pricing Under Jump Model)

Assume that Z_t is a standard Brownian motion. Let q_t denotes a Poisson counting process where the change dq_t is defined as

$$dq_t = \begin{cases} 0 & \text{with probability } 1 - \lambda dt \\ 1 & \text{with probability } \lambda dt, \end{cases}$$

where $\lambda > 0$ denotes the jump intensity. Consider the following risk neutral jump-diffusion process

$$\frac{dS(t)}{S(t^-)} = (r - \lambda \kappa)dt + \sigma dZ + (J - 1)dq_t \tag{2}$$

where the value $S(t^-)$ is the price just before jump and $\kappa = E[J - 1]$. Note that, if $dq_t = 1$, then after the jump

$$S(t) = S(t^-)J$$

where J corresponds to the random jump amplitude. In Merton's jump model, $\log J$ has a normal distribution, see §9, 9.1, 9.2, 9.4, 9.5 in the course notes for discussions on the Merton's jump diffusion model and its implementation.

In contrast to the original Merton's jump process, here we assume, instead, a double exponential jump process. Specifically, assume that, when jump occurs, the up-jump occurs with a probability p_u and down-jump occurs with a probability $1 - p_u$ and the density function of the jump amplitude J is a double exponential, i.e., the density function $f(y)$ of $y = \log(J)$ is

$$f(y) = p_u \mu_u \exp(-\mu_u y) \mathbf{1}_{y \geq 0} + (1 - p_u)\mu_d \exp(\mu_d y) \mathbf{1}_{y < 0} \tag{3}$$

where $\mathbf{1}_{y \geq 0}$ is an indicator which equals 1 if $y \geq 0$ and 0 otherwise.

(a) Assume that $\mu_u > 1$. Verify that

$$\kappa = E[J - 1] = \frac{p_u \mu_u}{\mu_u - 1} + \frac{(1 - p_u)\mu_d}{\mu_d + 1} - 1$$

(b) Assuming the underlying follows a jump diffusion process (2) with

$$\kappa = \frac{p_u \mu_u}{\mu_u - 1} + \frac{(1 - p_u)\mu_d}{\mu_d + 1} - 1$$

using Monte Carlo method with Euler time stepping., write a Matlab function to compute the fair value of a European Butterfly Spreads

$$\text{val} = \text{Butterfly}(S_0, r, \sigma, p_u, \mu_u, \mu_d, \lambda, K_1, K_3, T)$$

4

Table 2: Model Parameter Data

σ	.15
r	.05
Time to expiry	1 years
Initial asset price $S(0)$	$1
Strike Price K_1	$0.9S(0)$
Strike Price K_3	$1.1S(0)$
μ_u	**3.04**
μ_d	**3.08**
p_u	0.34
Intensity of Poisson Process λ	.1
Δt	1/1000

with strikes K_1, K_3 and expiry T, the payoff of a European Butterfly Spreads (in Question 3) is as below: let $K_1 < K_2 < K_3$, where $K_2 = \frac{1}{2}(K_1 + K_3)$, as follows:

$$
\text{payoff}(S_T) = \begin{cases} 0 & \text{if } S_T \leq K_1 \\ S_T - K_1 & \text{if } K_1 < S_T \leq K_2 \\ K_3 - S_T & \text{if } K_2 < S_T \leq K_3 \\ 0 & \text{if } S_T > K_3 \end{cases} \tag{4}
$$

Using parameter values in Table 2,, plot the computed option values with the number of (timesteps, simulations) = $(800, 25000)$ for $K_1 = (0.6 : 0.1 : 0.9)$ and $K_3 = 2S(0) - K_1$, hence $\frac{K_3 - K_1}{2} = 0.4 : -0.1 : 0.1$. How does the computed option value depend on $\frac{K_3 - K_1}{2}$? Explain why your observation is reasonable. You can use the matlab function **exprnd** which generates a random sample from an exponential distribution in $[0, +\infty)$ with a probability density function

$$
f_X(x) = \begin{cases} \frac{1}{\mu} e^{-\frac{x}{\mu}} & \text{if } x \in [0, +\infty) \\ 0 & \text{otherwise} \end{cases} \tag{5}
$$

where $\mu > 0$ is a parameter.

(c) Show that the Butterfly Spread in (4) becomes a standard call option when $K_3 = +\infty$. Use Matlab function **blsimpv** to plot the implied volatility of the standard option price using your Butterfly$(S_0, r, \sigma, p_u, \mu_u, \mu_d, \lambda, K_1, K_3, T)$, by setting $K_3 = +\infty$, with (timesteps, simulations) = $(800, 25000)$ against the strike, e.g., K=linspace$(.7, 1.2, 20)$ and $T = 1$.

(d) Discuss observed characteristics of the implied volatility from the assumed jump model.

6 [(Graduate Student Question)] (10 marks)

Practitioner's BS delta hedging consists of a portfolio $\Pi(t)$ with the following components at rebalancing time $t = t_n$: an option position $-V_n$, δ_n shares at price S_n, and the dollar amount B_n in a risk-free cash account, where

$$
\delta_n = \frac{\partial V_{BS}}{\partial S}(S_n, t_n; \breve{\sigma}_n) \tag{6}
$$

and $\frac{\partial V_{BS}}{\partial S}(S_n, t_n; \breve{\sigma}_n)$ is the BS delta at t_n using the implied volatility $\breve{\sigma}_n$ computed from the market option price V_n at the rebalancing time t_n. In [Hull and White(2017)], hedging position function for the underlying is learned from the market price data [1]. More specifically, the mean variance delta, δ_{MV} is modelled in a quadratic form

$$
\delta_{MV} = \delta_{BS} + \frac{\text{VEGA}_{BS}}{S\sqrt{T}}\left(a + b \cdot \delta_{BS} + c \cdot \delta_{BS}^2\right), \tag{7}
$$

---

[1] *Optimal delta hedging for options*, Journal of Banking and Finance, 82, 180-190, 2017.

see (5) in [Hull and White(2017)]. Your task is to read this paper to understand their proposed method and then conduct a comparative computational study on the data below.

You are given the S&P 500 index option data, in the file RawData.mat, from 01-Jan-2009 to 31-Dec-2010 as the training data set and the data from 01-Jan - 2011 to 31-Dec-2011 as the testing data set. In RawData.mat, values of following additional variables are also provided for you to use for training and testing:

- CSTrain: Daily change of the market index price for the training set.
- CVTrain: Daily change of the market index option price for the training set.
- CSTest: Daily change of the market index price for the testing set.
- CVTest: Daily change of the market option price for the testing set.
- DeltaTrain: BS delta δ_{BS} computed using implied volatility σ_{imp} for the training set.
- DeltaTest: BS delta δ_{BS} computed using implied volatility σ_{imp} for the testing set.
- VegaTrain: BS vega VEGA_{BS} computed using implied volatility σ_{imp} for the training set.
- VegaTest: BS vega VEGA_{BS} computed using implied volatility σ_{imp} for the testing set.
- STrain: market index price, S, for the training set.
- STest: market index price, S, for the testing set.
- TauTrain: time to expiry, τ, for the training set.
- TauTest: time to expiry, τ, for the testing set.

Consider daily hedging error $\Delta V - \delta \Delta S$, where $\Delta V = V(S(t_{n+1}), t_{n+1}) - V(S(t_n), t_n)$, $\Delta S = S(t_{n+1}) - S(t_n)$, δ is the position of the underlying at t_n. Using the provided data, you are to

(a) For practitioner's BS delta hedging defined in (6), plot the histogram of the hedging error $\Delta V - \delta \Delta S$, tabulate its mean, standard deviation, 95% VaR and CVaR when the hedging position $\delta = \delta_{BS}$ using given training data and testing data respectively.

(b) Learn the MV hedging model (7), see (5) in [Hull and White(2017)] , using the training data.

   (1) Report the ratio GAIN defined in (3) in [Hull and White(2017)] for the learned MV hedging model (7) using the training data and testing data respectively.

   (2) In addition, plot the histogram of the hedging error $\Delta V - \delta \Delta S$, tabulate its mean, standard deviation, 95% VaR and CVaR with respect to the training data and testing data respectively, when the hedging position δ comes from the MV model (7) learned from the training data. How does the testing performance compare to training performance?

(c) Assume now that the hedging position function δ at time t has a different parametric form given below

$$\delta = \delta_{BS} + c_1 \cdot S + c_2 \cdot \text{VEGA}_{BS} + c_3 \cdot \text{VEGA}_{BS} \cdot \delta_{BS} \tag{8}$$

where δ_{BS} denotes the BS delta, VEGA_{BS} denotes the BS vega. Learn this parametric function for the δ position by minimizing the MV error as defined in [Hull and White(2017)].

Repeat the same performance assessment in (b) using the position in the underlying δ learned from training data but assuming the parametric form (8). Compare results from (c) with the parameterizations in (b). What do you observe? Is the observation reasonable? Why?

You can use Matlab function **blsprice**, **blsimpv**, **blsdelta**.