King's College London

UNIVERSITY OF LONDON

PLACE THIS PAPER AND ANY ANSWER BOOKLETS **in the EXAM ENVELOPE provided**

Candidate No: .... **Desk No:** ....

MSC EXAMINATION

7CCMFM06 NUMERICAL AND COMPUTATIONAL METHODS IN FINANCE

MAY 2019

TIME ALLOWED: TWO HOURS

ALL QUESTIONS CARRY EQUAL MARKS. FULL MARKS ARE AWARDED FOR CORRECT ANSWERS TO FOUR QUESTIONS. IF MORE THAN FOUR ARE ATTEMPTED THEN ONLY THE BEST FOUR WILL COUNT.

YOU ARE PERMITTED TO USE A CALCULATOR.
ONLY CALCULATORS FROM THE CASIO FX83 AND FX85 RANGE ARE ALLOWED.

DO NOT REMOVE THIS PAPER FROM THE EXAMINATION ROOM

TURN OVER WHEN INSTRUCTED

1. (a) What is meant by a European knock-out call option with strike K, maturity T, and barrier B on a stock? You should write a mathematical formula giving the payout of the option in terms of the stock price S_t at times $t \in [0, T]$. **[20%]**

Answer: Bookwork.

$$\text{payoff} = \begin{cases} \max\{S - K, 0\} & \text{when } S_t < B \quad \forall t \in [0, T] \\ 0 & \text{otherwise} \end{cases}$$

(b) Suppose that the market follows the Black–Scholes model: in particular with the stock following geometric Brownian motion with drift μ and volatility σ; zero coupon bonds are available at a risk-free rate of r. Write the MATLAB code to simulate the stock price paths. **[20%]**

Answer: Bookwork.

```
function [ S, times ] = generateBSPaths( ...
T, S0, mu, sigma,nPaths, N  )

dt = T/nSteps;
logS0 = log( S0);
eps = randn( nPaths, nSteps );
dlogS = (mu-0.5*sigma^2)*dt + sigma*sqrt(dt)*eps;
logS = logS0 + cumsum( dlogS, 2);
S = exp(logS);
times = dt:dt:T;

end
```

(c) Describe how you could use such a simulation to estimate the price of a European Knock-Out option by the Monte Carlo method.

Answer: Simulate price paths in the risk-neutral measure. For each price path, compute the payoff of the knockout option and then discount this payoff by multiplying it by e^{-rT}. The risk-neutral price is approximated by the mean discounted payoff. **[20%]**

(d) How would you estimate the Delta and Gamma of a European Knock-Out option using the Monte Carlo method? **[20%]**

Answer: Estimating the Gamma in unseen.

Let $\boldsymbol{\epsilon}$ be a matrix of independent normally distributed random variables whose rows represent different price paths and whose columns correspond to different time points. Let

$$V(S_0, \boldsymbol{\epsilon})$$

be the price of the option as computed by the algorithm above.

The Delta can then be approximated by the central estimate:

$$\frac{V(S_0 + h, \boldsymbol{\epsilon}) - V(S_0 - h, \boldsymbol{\epsilon})}{2h}$$

and the Gamma by the estimate

$$\frac{V(S_0 + h, \boldsymbol{\epsilon}) - 2V(S_0 + h, \boldsymbol{\epsilon}) + V(S_0 - h, \boldsymbol{\epsilon})}{h^2}.$$

The important points are the approximation formula for the derivatives and the use of the same random matrix for each computation of V.

(e) Why might a trader want to know the Gamma of the option? **[10%]**
Answer: Unseen. They may wish to reduce the risk of their position by Gamma hedging.

(f) Explain why the same Monte Carlo method cannot be used to price an American Knock-Out option. What could you do instead to price an American Knock-Out option? **[10%]**
Answer: Bookwork. The Monte Carlo algorithm we have used requires that the payoff of the derivative be a function of the price path S_t. For an American option the payoff depends upon the choice of exercise time and so is not purely a function of the price path. One could use a finite difference method instead.

2. Suppose that a stock price process, S_t, follows the SDE

$$\mathrm{d}S_t = S_t(\mu\,\mathrm{d}t + \sigma\,\mathrm{d}W_t^1)$$

where W_t^1 is a Brownian motion. Suppose that the short-term interest rate r_t follows the SDE

$$\mathrm{d}r_t = a(r_t - b)\mathrm{d}t + c\,\mathrm{d}W_t^2$$

for some constants a, b and c, and where W_t^2 is a Brownian motion independent of W_t^1. You wish to simulate the processes S_t and r_t at discrete times 0, δt, $2\,\delta t$, $\ldots$, $i\,\delta t, \ldots, N\,\delta t$.

(a) Write the difference equation you would use to simulate r_t using the Euler scheme. **[20%]**

Answer: Unseen.

$$r_{t+\delta t} = r_t + a(r_t - b)\delta t + c\sqrt{\delta t}\epsilon_t^2$$

where the ϵ_t^2 are independent, standard normal random variables.

(b) Explain mathematically how you would simulate S_t, explaining why you would not use the Euler scheme for S_t. **[30%]**

Answer: Bookwork. Let $z_t = \log S_t$. By Itô's lemma, z_t satisfies

$$\mathrm{d}z_t = (\mu - \frac{1}{2}\sigma^2)\mathrm{d}t + \sigma \mathrm{d}W_t^1$$

which we solve to find

$$z_{t+\delta t} = z_t + (\mu - \frac{1}{2}\sigma^2)\delta t + \sigma(W_{t+\delta t}^1 - W_t^1).$$

Hence we may simulate z_t by taking

$$z_{t+\delta t} = z_t + (\mu - \frac{1}{2}\sigma^2)\delta t + \sigma\sqrt{\delta t}\epsilon_t^1.$$

where the ϵ_t^1 are independent, standard normal random variables. Then $S_t = \exp(z_t)$.

We do not use the Euler scheme directly because it is an approximation whereas we have actually solved the SDE exactly.

(c) Suppose that a trader incorrectly believes that the interest rate is a constant $r = r_0$. They write a call option with strike K and maturity $T = N\,\delta t$ for the Black–Scholes price P, and then follow the Black–Scholes delta hedging strategy in discrete time. Write down difference equations that you could use to compute their bank balance b_t at time $t = i\,\delta t$. You should assume that over the time interval from $i\,\delta t$ to $(i+1)\delta t$ their bank balance grows at the continuously compounded rate r_t. **[30%]**

Answer: Similar to bookwork, the variable interest rate being unseen. At the first time point

$$b_0 = P - \Delta_0 S_0.$$

At intermediate time points

$$b_{(i+1)\,\delta t} = e^{r_t \delta t} b_{i\delta t} - (\Delta_{(i+1)\delta t} - \Delta_{i\,\delta t}) S_{(i+1)\delta t}.$$

At the final time point

$$b_T = e^{r_t \delta t} b_{(N-1)\delta t} + \Delta_{(N-1)\delta t} S_T - (S - K)^+.$$

(d) Suppose that W_t^1 and W_t^2 are not independent, but are instead correlated with correlation ρ. How would you change your simulation to account for this? **[20%]**

Answer: Simulate r_t using the formula

$$r_{t+\delta t} = r_t + a(r_t - b)\delta t + c\sqrt{\delta t}(\rho \epsilon_t^1 + \sqrt{1-\rho^2}\epsilon_t^2)$$

Here we have used the fact that the Cholesky decomposition of the correlation matrix

$$\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

is

$$\begin{pmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{pmatrix}.$$

3. (a) What is meant by the Value at Risk of a portfolio? **[20%]**

Answer: The d-day, $p\%$ Value at Risk of a position is the $(100 - p)$-th percentile of the loss distribution over a d-day time horizon. (A formal mathematical definition would also be accepted)

(b) What is meant by the Expected Shortfall of a portfolio? **[10%]**

Answer: For portfolios with continuous loss distribution, the d-day, $p\%$ Expected Shortfall is the expected loss in the worst $p\%$ of cases over a d-day time horizon. (A formal mathematical definition would also be accepted)

(c) If a portfolio has current value P and its value at time T is normally distributed with mean M and standard deviation σ write down a formula to compute the Value at Risk of this portfolio over time horizon T in terms of N^{-1}, the inverse cumulative distribution function of the standard normal distribution. **[20%]**

Answer:

$$P - M - \sigma N^{-1}\left(\frac{p}{100}\right)$$

(d) Suppose that the market contains n assets. The current price of asset i $(1 \leq i \leq n)$ is P_i. At time T, the asset prices follow a multivariate normal distribution with means M_i and with covariance matrix Ω. Give a formula for the standard deviation of a portfolio consisting of α_i units of each asset i at time T. **[10%]**

Answer: Bookwork.

$$\text{standard deviation} = \sqrt{\alpha^T \Omega \alpha}$$

where α is the vector of weights.

(e) An investor has a budget B and wishes to invest this in a portfolio of the n assets in such a way as to make an expected profit of Q at time T while minimizing the portfolio's Value at Risk at time T at confidence level 95%. Give a mathematical formulation of this problem, justifying your answer, and explain how you would solve it in MATLAB. **[30%]**

Answer: Similar to bookwork. Any portfolio in this market will be normally distributed and so its expected shortfall is determined monotonically from its variance. Hence this problem is equivalent to minimizing the variance of the portfolio. We can therefore write the problem as

$$\text{minimize } \alpha^T \Omega \alpha$$

subject to

$$\sum_{i=1}^{n} \alpha_i M_i = Q$$

and

$$\sum_{i=1}^{n} \alpha_i P_i = B.$$

This is a quadratic programming problem so can be solved using the built-in function quadprog.

(f) How would you find a portfolio that minimizes the Expected Shortfall rather than the value at risk? **[10%]**

Answer: In exactly the same way. The portfolio will be normally distributed, so minimizing expected shortfall, minimizing value at risk and minimizing standard deviation are all equivalent problems.

4. (a) Let S be a subset of $[-1,1]^3$. One way to approximate the volume of S is by counting cubes. We divide each interval $[-1,1]$ into n subintervals and so divide the cube $[-1,1]^3$ into n^3 smaller cubes. One can then compute the proportion of cubes whose center lies in S and hence estimate the volume of S. Write MATLAB code to implement this algorithm in the case where S is given by

$$S = \{(x,y,z) \mid x^4 + y^4 + z^4 \leq 1.\}$$

[50%]

Answer: Unseen.

```
function ret = inS(x,y,z)
    ret = (x^4 + y^4 + z^4 \leq 1)
end

function computeVolume( shape, n )
    count = 0;
    stepSize = 2/n;
    for i=1:n
        for j=1:n
            for k-1:n
                x = -1 + stepSize*(i-0.5);
                y = -1 + stepSize*(j-0.5);
                z = -1 + stepSize*(k-0.5);
                if (containedInS(x,y,z))
                    count = count+1;
                end
            end
        end
    end
    return (8 * count)/n^3
end

% One can then calculate the volume using 1000^3 cubes by
computeVolume(@inS, 1000)
```

(b) How would you test your code? **[10%]**
Answer: Unseen. The code has been written so you can use a variety of shapes. We can test it gives the correct answer for a sphere.

(c) For subsets, S of $[-1,1]^3$ with smooth boundary, the error in this algorithm is proportional to $O(n^{-1})$. Describe how you could test this fact numerically, being sure to describe how you would interpret the results. **[20%]**
Answer: Draw a log-log plot of the error in the computation of the volume of a sphere against the n. The result should be approximately a straight line of gradient -1 if the claim is correct.

(d) Describe briefly how you could compute the volume of S by a Monte Carlo method. **[10%]**

Answer: Choose N uniformly generated values for $(x,\ y{,}z)$ in $[-1,1]^3$. The volume of S is equal to the proportion of these points that lie in S times the volume of $[-1,1]^3$ (which is 8).

(e) Which is a more efficient algorithm: counting cubes or the Monte Carlo method? Justify your answer. **[10%]**

Answer: (Unseen) By the central limit theorem, the standard error in the Monte Carlo method will be of order $O(N^{-\frac{1}{2}})$ where N is the number of test points. For counting cubes the error is of order $O(N^{-\frac{1}{3}}) = O(n^{-1})$. So the Monte Carlo method is more efficient.

5. (a) The Black–Scholes PDE is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0.$$

Describe the change of variables is required to transform this to the backwards heat equation

$$\frac{\partial U}{\partial t} = -\frac{1}{2}\sigma^2 \frac{\partial^2 U}{\partial x^2} \quad .$$

Justify your answer. [**30%**]

Answer: Bookwork. The Black–Scholes PDE is given by applying the Feynman–Kac theorem to V, the discounted expectation of the payoff of a derivative where the payoff of the derivative is given in terms of the final stock price S_T. This motivates us to define

$$U = e^{r(T-t)}V$$

so that U is equal to the expectation of the payoff. We define

$$x_t = (\log S_t) - \left(\mu - \frac{1}{2}\sigma^2\right) t$$

so that x_t follows driftless arithmetic Brownian motion with volatility σ. Applying Feynman–Kac to obtain a PDE for U, we will find that U satisfies the backwards heat equation.

(b) A digital call option of strike K which has payoff at maturity time T given by

$$\text{payoff} = \begin{cases} 1 & \text{when } S_T \geq K \\ 0 & \text{otherwise.} \end{cases}$$

Suppose that you wish to price this derivative by applying the implicit finite difference method to the backwards heat equation.

(i) Over what region of the (t, x)-plane would you perform the calculation? Justify your answer. [**20%**]

Answer: Bookwork. An appropriate region would be

$$[0, T] \times [x_0 - 8\sigma\sqrt{T}, x_0 + 8\sigma\sqrt{T}]$$

since it is unlikely that the stock price will change by as much as 8 standard deviations, so any errors introduced by using approximations on the boundary shouldn't have too great an impact on the accuracy of the answers.

(ii) What boundary conditions would you use? **[20%]**
Answer: Unseen. Along the bottom boundary ($x = x_0 - 8\sigma\sqrt{T}$) we may approximate the value of the option by assuming that it is sure to end up out of the money, so $U = 0$. Along the top boundary ($x = x_0 + 8\sigma\sqrt{T}$) we may approximate the value of the option by assuming that it is sure to end up in the money so the value is given by

$$V = e^{-r(T-t)} \times 1$$

and hence $U = 1$ along the top boundary. At time T we have

$$W = V = \begin{cases} 1 & \text{when } S_T \geq K \\ 0 & \text{otherwise.} \end{cases}$$

(iii) Derive the difference equations used for this finite difference method at points away from the boundary. **[30%]**
Answer: Bookwork. Write $U_{i,j} = U(i\,\delta t, j\,\delta x)$ where δx is the grid size in the t-direction, and δx is the grid size in the x-direction. Using the forward estimate difference for the derivative in t we find:

$$\frac{U_{i+1,j} - U_{i,j}}{\delta t} = -\frac{1}{2}\sigma^2 \frac{U_{i,j-1} - 2U_{i,j} + U_{i,j+1}}{\delta x^2}$$

Rearranging:

$$U_{i+1,j} = -\lambda U_{i,j-1} + (1 + 2\lambda)U_{i,j} - \lambda U_{i,j+1}$$

where $\lambda = \frac{\delta t}{2\delta x^2}$.

Final Page