בס"ד

Numerical Methods for Mathematical Finance, 88-636
Moed Aleph, Semester Aleph, 5773 - Solutions

Exam length: 90 minutes
You may use all reference materials and a pocket calculator.
Answer all the questions. All questions carry equal weight.
Explain all your answers thoroughly.

1. The stochastic process $X(t)$ satisfies the stochastic differential equation

$$dX = aX(2-X)dt + \sigma dW$$

and $X(0) = 0$. It is required to compute

$$p(T) = P(\max_{0<t<T} X(t) < 1).$$

(a, σ and T are all positive constants.)

(a) Write down the Euler-Maruyama method for simulating the process $X(t)$, and explain how you would use this in a Monte Carlo simulation to find $p(T)$ for some specific time T.

(b) Do you expect $p(T)$ to be an increasing or decreasing function of the parameters a, σ and T ?

(c) What are sources of error in the calculation of $p(T)$?

(d) How does the calculation need to be modified if the SDE is changed to $dX = -2a\log(1-X)dt + \sigma dW$? (Note $\log(1-X)$ is only defined for $X < 1$.)

(a) The Euler-Maruyama method approximates the continuous stochastic process $X(t)$ by the discrete stochastic process X_n given by

$$X_{n+1} = X_n + ahX_n(2-X_n) + \sqrt{h}\sigma Z_n \ , \qquad n \geq 0$$

where $X_0 = 0$ and $Z_n \sim N(0,1)$. Here h is the discrete time step and X_n is understood as the approximation to $X(nh)$. To use this in a Monte Carlo simulation to approximate $P(T)$, choose a (sufficiently large) number of time steps N and set $h = \frac{T}{N}$, and a (sufficiently large) number of simulations M. Construct M simulations of the process X_n, and for each one check whether $\max_{0\leq n\leq N} X_n < 1$. If m is the number of simulations for which this is true, then $p = \frac{m}{M}$ is an approximation to $P(T)$, with stochastic error estimate $\frac{\sqrt{p(1-p)}}{M}$.

(b) For $X < 1$, the drift term $aX(2-X)$ is positive, and increases with a. Thus increasing a increases the chance of "hitting" $X = 1$ and reduces $P(T)$. Similarly increasing T increases the chance of hitting, and reduces $P(T)$. And increasing σ means more stochastic effects and also increases the chance of hitting, thus reducing $P(T)$.

(c) There are 3 sources of error. First, using the EM method to simulate the stochastic process. Second, we only look if $X(t)$ is above 1 for discrete times t. Third, using the Monte-Carlo method to approximate the expected value by an average. The latter is "stochastic error", the first two are both forms of "deterministic error".

(d) In the original SDE the drift term increases from 0 to a as X increases from 0 to 1. In the new SDE the drift term increases from 0 to ∞. Indeed the drift is not defined for $X \geq 1$. Thus whereas for the first equation we can follow the stochastic process from $t = 0$ to $t = T$, for the second equation, once X has "hit" 1 the process cannot be continued. However, the only real effect of this is to make the programming a little more complicated.

---

2. The price $S(t)$ of a certain asset follows a geometric Brownian motion

$$dS = S(rdt + \sigma dW) .$$

A "can't lose" contract with expiration T and barrier $B \geq S(0)$ pays the holder, at time T, $S(T)$ if $S(T) \geq S(0)$ and $\max_{0<t<T} S(t) \leq B$, and $S(0)$ otherwise.

(a) Explain why the (current) value of this contract rises with B. What is the value if $B = S(0)$? Explain why for large B the value is $S(0)\exp(-rT)$ plus the value of a call with strike $S(0)$.

(b) Explain how you would perform a calculation to determine the value of the contract for fixed r, σ, T and B. You should not write explicit Matlab code, but you should explain all the necessary considerations in writing such a program.

(c) Explain how you would perform a calculation to determine the value B^* of the barrier for which the value of the contract is equal to $S(0)$.

(d) The value of σ is not known exactly. How would you go about estimating the accuracy required in σ for the value of B^* to have an error of no more than 5% ?

(a) The contract pays $S(0)$ if either $S(T) < S(0)$ or the barrier B is hit sometime between $t = 0$ and $t = T$. Otherwise it pays $S(T)$.

The higher the barrier, the less chance it will be hit. Hitting the barrier reduces the payoff, so a higher barrier should give a higher return.

If the barrier is $B = S(0)$ then it is already hit at the start. The return will for sure be $S(0)$ and the value of the contract is $S(0)e^{-rT}$.

If the barrier is very high then it will not be hit. The return is $\max(S(T), S(0)) = S(0) + \max(S(T) - S(0), 0)$. In other words, the return is $S(0)$ plus the return from a regular call option with strike $K = S(0)$, and the value of the contract is $S(0)e^{-rT}$ plus the value of this call.

(b) I would chose a number of simulations to do M, and a number of subintervals N in which to divide the time period T. I would use the standard GBM formula to simulate the price of the asset at the end of the time period from the price at the start

$$S_{n+1} = S_n e^{\left(r-\frac{1}{2}\sigma^2\right)h+\sqrt{h}\sigma Z_n}$$

where $h = \frac{T}{N}$ and $Z_n \sim N(0,1)$. For each simulation I would find the maximum value of the price and decide whether the barrier B is breached or not, and record the corresponding return:

$$\text{return} = \begin{cases} S(T) & \mathrm{max_{0\le t\le B} S(t) \le B \text{ and } } S(T) > S(0) \\ S(0) & \text{otherwise} \end{cases}$$

I would estimate the value of the contract by taking the average value of the return over the M simulations, and compute an estimate of the stochastic error from the standard deviations of the returns, divided by $\sqrt{M}$.

(c) In principle the procedure described in the previous section defines a function $f(B)$ which gives the price for a given B. For $B = S(0)$ we will have $f(B) = S(0)e^{-rT}$ and as B increases, so does $f(B)$. We want to find the solution of the equation

$$f(B) = S(0) \ .$$

This can be done, for example, by Newton's method. BUT: (1) For different choices of the random numbers used, different values of $f(B)$ are obtained. In solving the equation it is essential to use the same common random numbers every time. (2) The value of f is not accurate, but has a stochastic error, say q. So in fact it makes more sense to solve the 2 equations

$$f(B) = S(0) - q \ , \qquad f(B) = S(0) + q$$

This will give an interval $B_1 \le B \le B_2$ in which B^* must lie — we can take B^* to be the middle of this interval, and the error estimate for B^* to be half the length of the interval.

(d) We need to compute the sensitivity of B^* to changes in σ, i.e. we need to find

$$\frac{\partial B^*}{\partial \sigma} \ .$$

As usual we can approximate this by a finite difference:

$$\frac{\partial B^*}{\partial \sigma} \approx \frac{1}{2\Delta}\left(B^*(\sigma+\Delta) - B^*(\sigma-\Delta)\right)$$

for a suitable, not too small, Δ. Once again, the 2 calculations of B^* must be done with common random numbers to get a sensible result. The stochastic error in calculation of B^* may make the above result rather inaccurate, but it should be sufficient to get an idea of what is needed. Once the sensitivity is known, the change $\Delta\sigma$ in σ that gives a given change ΔB in B can be found from

$$\Delta B \approx \frac{\partial B^*}{\partial \sigma} \Delta\sigma \ .$$

---

3. Give a brief explanation of the Euler and Crank Nicolson methods for solution of

$$u_t = u_{xx} + (2 + \sin x) u_x \ , \qquad 0 < x < 2\pi, \quad t > 0$$

assuming Dirichlet boundary conditions $u(0,t)$ and $u(2\pi,t)$ are specified.
If instead of Dirichlet boundary conditions, *periodic boundary conditions*, i.e. that $u(0,t) = u(2\pi,t)$ and $u_x(0,t) = u_x(2\pi,t)$, are specified, how would you implement these in the Euler method? Why is the Crank Nicolson method ruined?

In both E and CN methods, the solution function $u(x,t)$ is approximated at grid points $x = ih$, $t = jk$ where h, k are the step sizes in the x and t directions respectively, and $0 \le i \le N$, where $h = \frac{2\pi}{N}$.
For Euler we have

$$\frac{u_{i,j+1} - u_{i,j}}{k} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + (2 + \sin ih)\,\frac{u_{i+1,j} - u_{i-1,j}}{2h}$$

or

$$u_{i,j+1} = \left(\frac{k}{h^2} + \frac{k(2+\sin ih)}{2h}\right) u_{i+1,j} + \left(1 - 2\frac{k}{h^2}\right) u_{i,j} + \left(\frac{k}{h^2} - \frac{k(2+\sin ih)}{2h}\right) u_{i-1,j}$$

For CN we have

$$\begin{aligned}\frac{u_{i,j+1} - u_{i,j}}{k} &= \frac{1}{2}\left[\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + (2+\sin ih)\,\frac{u_{i+1,j} - u_{i-1,j}}{2h}\right. \\ &\quad \left. + \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} + (2+\sin ih)\,\frac{u_{i+1,j+1} - u_{i-1,j+1}}{2h}\right].\end{aligned}$$

or

$$\begin{aligned}&-\left(\frac{k}{2h^2} + \frac{k(2+\sin ih)}{4h}\right) u_{i+1,j+1} + \left(1 + \frac{k}{h^2}\right) u_{i,j+1} - \left(\frac{k}{2h^2} - \frac{k(2+\sin ih)}{4h}\right) u_{i-1,j+1} \\ &= \left(\frac{k}{2h^2} + \frac{k(2+\sin ih)}{4h}\right) u_{i+1,j} + \left(1 - \frac{k}{h^2}\right) u_{i,j} + \left(\frac{k}{2h^2} - \frac{k(2+\sin ih)}{4h}\right) u_{i-1,j}\end{aligned}$$

For the equations of both E and CN the index i runs from 1 to $N-1$, and every occurence of $u_{0,j}$ or $u_{N,j}$ should be replaced by the relevant boundary condition (assuming Dirichlet boundary conditions). For E, the equation allows direct determination of the $u_{i,j+1}$ given $u_{i,j}$. For CN, the equations give a tridiagonal system to determine the $u_{i,j+1}$.

If periodic boundary conditions are imposed then we no longer know $u_{0,j}$ and $u_{N,j}$. However we have $u_{0,j} = u_{N,j}$ and

$$\frac{u_{1,j} - u_{0,j}}{h} = \frac{u_{N,j} - u_{N-1,j}}{h}$$

giving

$$u_{0,j} = u_{N,j} = \frac{1}{2}\left(u_{1,j} + u_{N-1,j}\right)$$

which in E allows the boundary conditions to be determined at each level. Unfortunately in CN this relation wrecks the tridiagonality.
