Worcester Polytechnic Institute
Department of Mathematical Sciences
Professor: Stephan Sturm
Teaching Assistant: Qingyun Ren

Spring 2017

MA 573

Computational Methods of Mathematical Finance

Assignment 8

due on Thursday, March 30, in class

1. Consider the following finite difference schemes for some sufficiently smooth function $f : (t, x, y) \mapsto f(t, x, y)$:

$$\frac{3f(t,x,y) - 4f(t-\Delta t, x, y) + f(t - 2\Delta t, x, y)}{2\Delta t} \tag{1}$$

$$\frac{-f(t, x+3\Delta x, y) + 4f(t, x+2\Delta x, y) - 5f(t, x+\Delta x, y) + 2f(t,x,y)}{\Delta x^2} \tag{2}$$

$$\frac{f(t, x+\Delta x, y) - f(t, x+\Delta x, y-\Delta y) - f(t,x,y) + f(t,x,y-\Delta y)}{\Delta x \Delta y} \tag{3}$$

   (a) Find for every scheme to which limit it converges when the increments (Δ-terms) tend to zero.

   (b) Find for each scheme its accuracy, i.e., determine the order of the error term.

   Context: Such more complicated schemes are also sometimes used, e.g., when implementing boundary conditions.

2. Derive a system of ordinary differential equations (in vector-matrix form $v' = Av$) that approximates the Black–Scholes PDE

$$-v_t(t,s) + \frac{1}{2}\sigma^2 s^2 v_{ss}(t,s) + rsv_s(t,s) = rv(t,s)$$

   by using a space discretization with finite differences

$$v_s(t, s_i) \approx \frac{v(t, s_{i+1}) - v(t, s_i)}{\Delta s}$$
$$v_{ss}(t, s_i) \approx \frac{v(t, s_{i+1}) - 2v(t, s_i) + v(t, s_{i-1})}{\Delta s^2}$$

on the equally spaced grid $\{s_1, s_2, \ldots, s_I\}$, $s_{i+1} - s_i = \Delta s$. Specifically, derive (the coefficients of) the matrix A explicitly. (You may ignore the issue of the boundary condition, i.e., the first and last row of the matrix for the moment).

3. Calibrate the Black–Scholes model

$$dS_t = rS_t\,dt + \sigma S_t\,dW_t^1, \qquad S_0 = s$$

to market data (choose a realistic risk-free rate from data and use call options). Use as data options on Apple stock with maturity May 19, 2017. Discard all strikes that have a trading volume of less then 10 and use always the marked mid-price (average between bid and ask). Calibrate $\Theta = \{\sigma\}$ by minimizing the sum of the squared errors

$$\underset{\Theta}{\operatorname{argmin}} \sum_i \left| C_i^\Theta - C_i \right|^2.$$

This can be done numerically, using a numerical optimization procedure , e.g., `curve_fit` from `scipy.optimize`,
https://python4mpia.github.io/fitting_data/least-squares-fitting.html

   (a) Calculate the Black-Scholes price simulating the path of the stock price and using Monte-Carlo integration. Use different starting values for the least square fit, e.g., $\sigma = 0.2, 0.3, 0.02, 3$.

   (b) Do know the same as in (a), but use know the Black-Scholes formula instead of Monte Carlo integration.

   (c) Compare the results from (a) and (b) and try to understand the difference. Explore the reasons why one of the techniques works well, and one not. To do this, it might be helpful to know that `curve_fit` uses internally the Levenberg-Marquardt algorithm and do an online search to understand how this algorithm works and why it might not work well with one of the pricing techniques.

Note: This problem intends to clarify the issues that lead to the problems encountered in problem 6 on assignment 6.

8 points per problems