Jacobs University
Fall 2019

November 18, 2019

Stochastic Methods + Lab

Assignment Sheet 10

Due on November 25, 2019

Problem 1 [6 points]

A system of linear equations of the form

$$\begin{pmatrix} b_1 & c_1 & & \cdots & 0 \\ a_1 & b_2 & c_2 & & \\ & a_2 & b_3 & \ddots & \\ \vdots & & \ddots & \ddots & c_{n-1} \\ 0 & \cdots & & a_{n-1} & b_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_n \end{pmatrix},$$

where the $n \times n$ matrix on the left-hand side is tridiagonal (i.e., the matrix entries are zero except for the main diagonal and the diagonal above and below), can be solved in $O(n)$ steps.

(a) Write out the expressions which arise when performing Gaussian elimination on this system.

(b) Write a tridiagonal solver as a Python function.

(c) Scipy has a build-in banded matrix solver:

```
from scipy.linalg import solve_banded
```

Look up the documentation and use it to compare the result and the computing time with your tridiagonal solver from b) for the case of $a_i = c_i = 1$ for $i = 1, \ldots, n-1$, $b_i = -2$ for $i = 1, \ldots, n$, n large, and the right-hand side some random vector.

Problem 2 [2 points]

Consider the Black-Scholes partial differential equation for the price $C(S,t)$ of a European call option as a function of the current stock price S and time t,

$$\frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2} + r\,S\,\frac{\partial C}{\partial S} - r\,C = 0\,,$$

where σ is the volatility of the underlying stock and r the risk-free interest rate. Use the chain rule to show that under the change of variable $S = \exp(X)$ and $V(X,t) = C(S,t)$, the Black-Scholes equation turns into the constant coefficient drift-diffusion equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 \frac{\partial^2 V}{\partial X^2} + \left(r - \frac{1}{2}\sigma^2\right)\frac{\partial V}{\partial X} - r\,V = 0\,.$$

Problem 3 [12 points]

(a) Using the conventions from Problem 2, the explicit form of the finite difference approximation to the Black-Scholes equation reads

$$\frac{V_n^m - V_n^{m-1}}{\Delta t} + \frac{\sigma^2}{2}\,\frac{V_{n-1}^m - 2\,V_n^m + V_{n+1}^m}{\Delta X^2} + \left(r - \frac{\sigma^2}{2}\right)\frac{V_{n+1}^m - V_{n-1}^m}{2\,\Delta X} - r\,V_n^m = 0\,.$$

Write a code which uses the explicit finite difference scheme to price a European call option. Make sure to include a discussion of the boundary conditions.

(b) Show that the explicit code becomes unstable unless the time step Δt is much smaller than ΔX.

(c) Modify your code to use the implicit finite difference scheme

$$\frac{V_n^{m+1} - V_n^m}{\Delta t} + \frac{\sigma^2}{2}\,\frac{V_{n-1}^m - 2\,V_n^m + V_{n+1}^m}{\Delta X^2} + \left(r - \frac{\sigma^2}{2}\right)\frac{V_{n+1}^m - V_{n-1}^m}{2\,\Delta X} - r\,V_n^m = 0\,.$$

Show that it is stable even when the time step Δt is large.

(d) Demonstrate the order of convergence of the implicit finite difference method with the same number of meshpoints in the t and in the X direction.