MA417 Computational Methods in Finance - Project

2013/14

1 Guidelines

Your project must be submitted to the general office COL 4.01 by

Thursday 22th May 2014, 2pm.

All consequences regarding late submission can be found on the School's website

`http://www.lse.ac.uk/resources/calendar/academicRegulations/regulationsForTaughtMastersDegrees.htm`

There you find in particular, that "five marks out of 100 will be deducted for coursework submitted within 24 hours of the deadline and a further five marks will be deducted for each subsequent 24-hour period (working days only) until the coursework is submitted. After five working days, coursework will only be accepted with the permission of the Chair of the Sub-Board of Examiners".

- Please submit **two spiral bound copies** of your project report. In addition, **submit your report as a PDF-file together with all files containing your `C++` code by email to r.batey@lse.ac.uk.**
  For the electronic submission please only submit one zip-file that contains all your files. Your zip-file should have the following name: If your candidate number is 123456 then name your zip-file CN123456.zip
  This zip-file must contain your project report (as a pdf-file) which you should name according to your candidate number, e.g., CN123456report.pdf and all your `C++` code including makefiles and .dev-files (where necessary)).
- When you submit the project you must hand in a completed and signed copy of the Plagiarism Statement (available from the Departmental Office and on Moodle).
  You are required to read the information on plagiarism on the following website:

  `http://www.lse.ac.uk/resources/calendar/academicRegulations/RegulationsOnAssessmentOffences-Plagiarism.htm`

  Note in particular the first paragraph on this website:
  "All work for classes and seminars as well as scripts (which include, for example, essays, dissertations and any other work, including computer programs) must be the student's own work. Quotations must be placed properly within quotation marks or indented and must be cited fully. All paraphrased material must be acknowledged. Infringing this requirement, whether deliberately or not, or passing off the work of others as the work of the student, whether deliberately or not, is plagiarism."

  Note that all reports will be submitted to Turnitin for textual similarity review and the detection of plagiarism.
- The contents of your work must remain anonymous, so do not write your name on anything except the Plagiarism Statement. Instead, you must identify your work with your Examination Candidate Number. You can check your candidate number on 'LSE for You'.

- The project consists of **two parts** which both contribute to the final mark:

  1. **A written report containing all results and their derivation, interpretation and discussion.**
     Note that you are asked to write a report and not just a question-answer style exercise set solution. You should aim for approximately 20 pages in length on single sided A4 paper with text font point 12 single-spaced. Your report must not exceed 30 pages. Pages will be counted from the first page containing text until the beginning of the appendix. Hence material in the appendix (such as the `C++` code), bibliography, title pages etc. do not count towards the page count.
  2. **`C++` programs.**
     The only acceptable programming language for this project is `C++` . You are, however, allowed to use any software of your choice for graphical illustration of the output of your `C++` program(s).
     You should implement all routines from scratch (using libraries such as iostream, stdlib, cmath, ...) and hence you must not use special scientific/numerical `C++` libraries. There are two exceptions to this rule:
     (a) You may use `C++` code from the MA417 Programming Sessions provided on the MA417 Moodle page. If you decide to do so just write a comment in your `C++` file saying, e.g., the function X or the class Y was taken from the solutions to Worksheet Z.
     (b) You may use the GNU Scientific Library (GSL) for minimizing functions.

     In the appendix please provide codes for all your `C++` programs with all relevant documentation and instructions to the examiners (such as how to run your programs, usage of parameters etc.).

     Please note that your `C++` program(s) should NOT ask the user to enter variables needed for the computation via the standard input stream. Choose reasonable default parameters yourself (e.g., as global variables) and make clear in your instructions what the meaning and the names of the variables are such that the examiners can test several examples.

     Note that your code will be compiled and tested using Bloodshed Dev-C++ (version 4.9.9.2) which is available on the LSE PCs. You are strongly advised to use this programming environment to produce and test your code.

     If you decide to split up your code into several files you are required to submit the corresponding project file (the .dev file) to enable direct compilation within Bloodshed Dev-C++.

- The project consists of 4 problems. You must solve **ALL problems**.

- The project will be marked in line with the departmental assessment criteria which are available on:

`http://www.lse.ac.uk/maths/Courses/PDFs/`
`LSE-Mathematics-MSc-Assessment-Criteria-2012.pdf`

2 Project Description

2.1 The Heston Model

The Heston model assumes the following dynamics for the asset price S under the risk-neutral probability measure:

$$(1)\quad dS(t) = S(t)(rdt + \sqrt{v(t)}dW^S(t)), \qquad S(0) = S_0 > 0,$$

$$(2)\quad dv(t) = \alpha(\beta - v(t))dt + \gamma\sqrt{v(t)}dW^v(t), \qquad v(0) = v_0 > 0,$$

where W^S, W^v are Brownian motions under the risk-neutral probability measure with

$$dW_t^S dW_t^v = \rho dt,$$

and $v_0 > 0$ is the initial squared volatility, $\beta \geq 0$ the long-run squared volatility, $\alpha \geq 0$ the mean reversion rate of the squared volatility and $-1 \leq \rho \leq 1$ is the so-called *leverage* parameter.
In the Heston model the squared volatility v is a Cox-Ingersoll-Ross (CIR) process and therefore no-longer constant as in the Black-Scholes model.

The following Euler scheme for the log-asset price in the Heston model is frequently considered. We denote by $\log(\hat{S})$ and $\hat{v}$ the approximations of $\log(S)$ and v. Let $h > 0$, then consider the following Euler scheme

$$(3)\quad \begin{aligned} \hat{v}(t+h) &= \hat{v}(t) + \alpha(\beta - \hat{v}(t)^+)h + \gamma\sqrt{\hat{v}(t)^+}Z_v\sqrt{h}, \\ \log(\hat{S}(t+h)) &= \log(\hat{S}(t)) + (r - \frac{1}{2}\hat{v}(t)^+)h + \sqrt{\hat{v}(t)^+}Z_s\sqrt{h}, \end{aligned}$$

where Z_v, Z_s are standard normally distributed random variables with correlation ρ and $x^+ = \max\{x, 0\}$. (Here log denotes the natural logarithm.) (Hint: You can simulate Z_v, Z_s by sampling two independent standard normally distributed random variables Z_1, Z_2 and setting $Z_v = Z_1$, $Z_s = \rho Z_1 + \sqrt{1-\rho^2}Z_2$.)

Problem 1. 1. *State the* Feller condition *for the CIR process given in (2). Describe briefly the dynamics of a CIR process.*

2. *Describe how a Monte Carlo estimator of a European Call option written on an asset following the Heston model can be computed based on the Euler scheme (3).*

3. *Implement a C++ program which uses the Euler scheme (3) to compute the time-*0 *price of a European Call option with maturity T and strike price K.*

4. *Use your C++ program to compute the time-*0 *price of a European Call option with maturity $T = 1$, strike $K = 10$ initial asset price $S_0 = 10$, interest rate $r = 0.0$ and the following values of the parameters of the process v:*
   $v_0 = 0.04$, $\alpha = 2$, $\beta = 0.04$, $\gamma = 0.2$, $\rho = -0.2$.

2.2 Calibrating the Heston Model

In the following we consider the problem of calibrating the Heston model to market data. Suppose you have $n \in \mathbb{N}$ empirical market prices $p_1^{\text{market}}, \dots, p_n^{\text{market}}$ of European Call options with maturities $T_1, \dots, T_n$ and strike prices $K_1, \dots, K_n$. We denote by $p^{\text{Heston}}(\alpha, \beta, \gamma, \rho, v_0; T, K, S_0, r)$ the price of a European Call option with maturity T, strike price K where the underlying asset price is given by the Heston model (1) and (2). One possible way to calibrate the Heston model to market data is to minimize the function

$$(4)\quad \sum_{i=1}^{n}(p_i^{\text{market}} - p^{\text{Heston}}(\alpha, \beta, \gamma, \rho, v_0; T_i, K_i, S_0, r))^2$$

over $\alpha, \beta, \gamma, \rho, v_0$.

Problem 2. 1. *Suppose you use the criterion stated in (4) to calibrate the Heston model to market data, describe how the minimization can be carried out by using an algorithm that is available in the GNU Scientific Library. Discuss possible problems that can occur when minimizing this function and make suggestions how to overcome these problems.*

2. *Discuss how you can deal with constraints on the parameters (e.g., non-negativity constraints, Feller condition) in the minimization problem discussed in the first part of this question.*

3. *Suggest at least two alternative criteria which are different from criterion (4) for calibrating the Heston model to empirical data. Discuss advantages and disadvantages of the different criteria.*

Problem 3. 1. *Write a* `C++` *program that implements a calibration routine of the Heston model to empirical data using criterion (4). To do so, use the GNU Scientific Library for obtaining a minimisation routine.*

2. *Consider the following market data, observed at time* $t = 0$ *of European Call options with maturity* $T = 1$ *and* $S_0 = 14.97$ *for different strike prices. The interest rate is assumed to be* $r = 0.0$*:*

strike prices K	*10*	*12.5*	*15*	*17.5*	*20*	*22.5*	*25*
option prices p^{market}	*5.43*	*3.35*	*1.85*	*0.9*	*0.4*	*0.18*	*0.09*

Use your `C++` *program from the first part of Problem 3 to calibrate the Heston model to the empirical data above and answer the following questions:*

(a) *What is the optimal choice of model parameters that you obtain?*

(b) *What are the resulting option prices based on these optimal model parameters?*

(c) *How well does the Heston model fit the empirical data?*

3. *Write a* `C++` *program that implements a calibration routine of the Heston model to empirical data using a criterion of your choice that is different from criterion (4). Use this code to calibrate the Heston model to the data provided above and compare your results to those obtained as answer to the second part of Problem 3.*

2.3 Pricing Exotic Options in the Heston Model

Problem 4. 1. *Write* `C++` *code that computes the price of a barrier option and the price of an Asian option of your choice assuming that the underlying asset can be described by the Heston model.*

2. *Use the empirical data provided in Problem 3 together with your calibration results to answer the following question: For which price(s) would you be willing to buy or sell the barrier option and the Asian option of your choice? Justify your answer.*