Project 5, deadline December 16, 2020

Black-Scholes equation

Fall semester 2020

Solving the Black-Scholes Equation Numerically

WARNING: this is a somewhat experimental problem given for the first time the Autumn of 2020. Only a very minimal theoretical background is given in this text. Feel free to write to Sebastian for more information.

The briefest introduction to options. A financial derivative is a contract that derives its value from the performance of an underlying asset. Perhaps the most common financial derivative is the option. Such contracts are incredibly old, the first mention of options contracts in history is by greek philosopher Thales from the sixth century.

An option is a right, but not an obligation, to buy or sell and underlying asset[1] at a predetermined price E at or before an expiration time T. Having such an option is valuable, but determining the fair price of an option is a difficult problem.

Options to buy (sell) are commonly referred to as call (put) options. An option that only allows one to excercise this right at the maturity date is called a European option, while an option that can be exercise at any date prior to the maturity date is called an American option. There are many other varieties.

In 1997, the (pseudo-)Nobel prize in economics was awarded to Robert C. Merton and Myron S. Scholes "for a new method to determine the value of derivatives", the Black-Scholes-Merton model.

The Black-Scholes equation. The Black-Scholes equation is a partial differential equation, which describes the price of an option over time. The key insight behind the equation is that one can perfectly hedge the option by buying and selling the underlying asses and the "bank account asset" (cash) in just the right way to eliminate risk.

$$\frac{\partial V}{\partial t} + \frac{1}{2} S^2 \sigma^2 \frac{\partial^2 V}{\partial S^2} + (r - D) S \frac{\partial V}{\partial S} - rV = 0 \qquad (1)$$

[1] We focus on stocks, but it can be anything.

Here $V(S,T)$ is the value of the options, S is the price of the underlying asset, σ is the volatility of the underlying asset, r is the "risk-free" interest rate, and D is the yield (dividend paying rate) of the underlying stock.

The volatility σ stems from an underlying assumption that the stock moves like a geometric Brownian motion,

$$\frac{dS}{S} = \mu dt + \sigma dW. \tag{2}$$

Explicit solutions for the Black-Scholes equation, called The Black-Scholes formulae, are known only for European call and put options. For other derivatives, such a formula doest not have to exist. However, a numerical solution is always possible.

5a: Transformation to Heat Equation/Diffusion equation. Instead of an initial value problem, we have a terminal value problem at time T, i.e. the expiration date or the maturity date. We change to an initial value problem by substituting $\tau = T - t$. This new variable can be interpreted as time remaining to expiration.

The transformed spatial variable is $x = \ln(S/E)$, where E is the exercise price of the option. Now, values of x close to zero correspond to stock prices that are close to the exercise price of the option. Negative values of x correspond to stock prices lower than the exercise price and positive values of x correspond to prices higher than the exercise price.

Just substituting for the variables above leads to a parabolic equation, with constant coefficients. Show that by making a final substitution;

$$u(x,\tau) = e^{\alpha x + \beta \tau} V(S,t) \tag{3}$$

we get the heat (diffusion) equation

$$\frac{\partial u}{\partial \tau} = \frac{\sigma^2}{2}\frac{\partial^2 u}{\partial x^2} \tag{4}$$

What are the correct parameters for α and β?

5b: Create solver(s) for the 1D diffusion equation. You can implement a solver for the diffusion equation inspiration from the project on the diffusion equation. Another very good resource is Langtangen and Linge's book "Finite Difference Computing with PDEs".

It is highly recommended to start with an explicit scheme, and then move to a more sophisticated implicit scheme. You should make up your own mind regarding what scheme is robust enough. Gauss-Seidel (successive over-relaxation) and Crank-Nicholson are some suggestions.

After creating a general diffusion equation solver, you can adapt the program to solve the Black-Scholes equation. Plot V vs S at different values for t (τ).

Special considerations The variable x is unbounded, i,e. $x \in [-\infty, \infty]$. Numerically, we have to pick a bounded interval $[-L, L]$, where L is a sufficiently large number. This interval remain unchanged when considering an option with a different strike price.

You need to impose special boundary conditions for $x = -L$ and $x = L$, corresponding to stock prices that are close to zero and very high, approaching infinity.

You need to pick som values for E, r, D and σ some starting choices can 50, 0.04, 0.12 and 0.4, respectively. Discuss what values would be reasonable - in the present market situation the values suggested are absolutely not!

In order to make reasonable plots, you need to transform the solution of the diffusion equation into the solution for the Black-Scholes equation.

You should compare to the analytic solution, i.e. the Black-Scholes Formula:

$$C(S_t, t) = N(d_1)S_t - N(d_2)PV(E), \tag{5}$$

where the present value of the exercise price is given by

$$PV(E) = Ee^{-rt}, \tag{6}$$

furthermore we have parameters d_1,

$$d_1 = \frac{1}{\sigma\sqrt{T-t}}\left[\ln\left(\frac{S_t}{E}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t),\right] \tag{7}$$

and d_2,

$$d_2 = d_1 - \sigma\sqrt{t}, \tag{8}$$

while N is the cumulative normal distribution function.

5c: Compute values for first-order Greeks. The "Greeks" measure sensitivity of the value of the derivative of a portfolio to changes in parameter value while holding other parameters fixed. There are different ways of hedging a portfolio agains risk by eliminating the movement in one or several of these parameters.

Delta: $\Delta = \frac{\partial V}{\partial S}$

Gamma: $\gamma = \frac{\partial^2 V}{\partial V}$

Vega[2]: $\nu = \frac{\partial V}{\partial \sigma}$

Theta: $\Theta = -\frac{\partial V}{\partial \tau}$

Rho: $\rho = \frac{\partial V}{\partial r}$

Make a representation of how these parameters move over time. Why are the Greeks interesting? Is your computed results reasonable?

[2]Notice that this is not really a greek letter.

5d: Find data and compute implied volatility. There are several market-traded standardised options. On the Oslo Stock Exchange (OSE) there are standardised options for several of the larger listed companies such as Equinor, Norsk Hydro, Telenor, Mowi, Orkla etc. See for instance `https://www.oslobors.no/markedsaktivitet/#/list/derivatives/quotelist/false`

Since there the market already prices the options, one can use these options prices to derive the value of implicit variables. A common practice is to use the market-given prices of options to derive the implied volatility of the option.

To do this; pick a large company with a high turnover of stock, and whose stock has seen several actual option trades on OSE lately. Find a reasonable estimate for the risk-free interest rate r. This would typically be given by short-term government bonds of the same currency as the stock. In NOK, you can find "Statskassaveksler" - these are bonds issued by the Norwegian Government with a shorter maturity than 12 months. Find the effective, real, annualised interest of these and use that as an estimate for r (these days it would be very close to zero). Norges Bank publishes data on this. You can also find quotes on OSE. The dividend yield D of the stock can be computed by taking the sum of all dividends paid py the company per stock for the year and dividing by the stock price. Then you should be able to tune the volatility in your solver to fit the call prices given by the market. You can do this by trial and error, but also use a root-finder like Newton's method or Brent's method.

Introduction to numerical projects

Here follows a brief recipe and recommendation on how to write a report for each project.

- Give a short description of the nature of the problem and the eventual numerical methods you have used.
- Describe the algorithm you have used and/or developed. Here you may find it convenient to use pseudocoding. In many cases you can describe the algorithm in the program itself.
- Include the source code of your program. Comment your program properly.
- If possible, try to find analytic solutions, or known limits in order to test your program when developing the code.
- Include your results either in figure form or in a table. Remember to label your results. All tables and figures should have relevant captions and labels on the axes.
- Try to evaluate the reliability and numerical stability/precision of your results. If possible, include a qualitative and/or quantitative discussion of the numerical stability, eventual loss of precision etc.
- Try to give an interpretation of you results in your answers to the problems.

- Critique: if possible include your comments and reflections about the exercise, whether you felt you learnt something, ideas for improvements and other thoughts you've made when solving the exercise. We wish to keep this course at the interactive level and your comments can help us improve it.

- Try to establish a practice where you log your work at the computerlab. You may find such a logbook very handy at later stages in your work, especially when you don't properly remember what a previous test version of your program did. Here you could also record the time spent on solving the exercise, various algorithms you may have tested or other topics which you feel worthy of mentioning.

Format for electronic delivery of report and programs

The preferred format for the report is a PDF file. You can also use DOC or postscript formats or as an ipython notebook file. As programming language we prefer that you choose between C/C++, Fortran2008 or Python. The following prescription should be followed when preparing the report:

- Use **Canvas** to hand in your projects, log in at `https://www.uio.no/english/services/it/education/canvas/` with your normal UiO username and password.

- Upload **only** the report file! For the source code file(s) you have developed please provide us with your link to your github domain. The report file should include all of your discussions and a list of the codes you have developed. Do not include library files which are available at the course homepage, unless you have made specific changes to them. Alternatively, you can just upload the address to your GitHub or GitLab repository.

- In your git repository, please include a folder which contains selected results. These can be in the form of output from your code for a selected set of runs and input parameters.

- In this and all later projects, you should include tests (for example unit tests) of your code(s).

- Comments from us on your projects, approval or not, corrections to be made etc can be found under your **Canvas** domain and are only visible to you and the teachers of the course.

Finally, we encourage you to work two and two together. Optimal working groups consist of 2-3 students. You can then hand in a common report.