OCaml in Quantitative Finance: MS1 Alpha

Authors: Arthur Wayne (asw263), Advay Koranne (ak845),
Maxwell Zweig (maz72), Jack Young (jry28)

Vision:

Currently the vision for our project is to develop an interactive options pricing suite with data visualization capabilities and a user-friendly GUI. We hope to include the Black-Scholes OPM (options pricing model) as well as the Binomial Tree OPM and give the user the ability to choose which model to apply. Our project's overall vision has shifted significantly. We started off having a vision that was extremely pragmatically-based with a pure focus on implementing the mathematics behind the OPMs that we implement and a very small focus on user interactivity. We now are considering more client-facing features such as data visualization and the GUI. We made this change after discussing improvements we could make to our project proposal with our PM. We hope to better accommodate the user experience in doing so.

Summary of progress:

During this sprint, our team was able to complete a variety of tasks that put us closer towards our path to project completion including:

1. Implemented a robust numeric PDF integrator to support our client facing Black-Scholes model.
2. Engineer a terminal UI for the Black-Scholes model to price European options
3. Build a variety of helper functions to deal with time-series data.
4. Created a module to parse time-series CSV data.

Our demo will combine all of these to showcase the Black-Scholes pricing model in the terminal, providing pricing analysis for both short term and LEAP (Long-Term Equity Anticipation Securities) options. We wrote a comprehensive testing suite to test each of our functions, and in showcasing the Black-Scholes model also utilized a variety of different functions, including numerical integration and date handling. Furthermore, we properly set up our coding environments and organized in such a way where we tried to make sure we made things as decoupled as possible and abstracted things so that we could iterate things in the future if needed. We developed different execution and test files so that we can develop our test cases and keep a client facing side as well following a similar approach as A2 and A3. We also developed our in-house math library which contains things such as an integral calculator and different probability distribution functions which will allow us to modify our models and make them more accurate depending on the clients needs and desires.

Activity breakdown:

Regarding our activity breakdown, each team member were assigned the following tasks and completed them in the following amounts of time:

Arthur:

1. Implement a module to read .csv files and convert them to a usable format for calculations in OCaml
2. Cleaned online options data to only include relevant fields and ease data processing
3. Develop test cases for all applications developed
4. Hours spent: 13

Max:

1. Develop a terminal UI to input custom parameters into the Black-Scholes model with Advay
2. Create helper functions to deal with time-series data (date math)
3. Start looking into and setting up files for Monte Carlo Options Pricing
4. Develop test cases for all applications developed
5. Hours Spent: 12

Advay:

1. Implement the Black-Scholes OPM to price European call options
2. Develop test cases for all applications developed
3. Develop a terminal UI to input custom parameters into the Black-Scholes model with Max
4. Hours Spent: 12

Jack:

1. Developed numeric PDF integrator by hand to support the Black-Scholes model calculations as well as support for integration on a variety of non-normal PDFs
2. Develop test cases for all applications developed
3. Hours Spent: 13

Productivity analysis:

Regarding our team's productivity, we were very productive overall and completed what we set out to do. We met on a regular basis (at least twice a week) and communicated with each other via Slack regarding any kind of issues that we ran into with code, git, or general problems. Our in-person meetings were without a doubt more productive and allowed us to be fully engaged in completing our tasks with the added benefit of easy communication. Zoom meetings, on the other hand, were helpful for planning what kind of tasks that we wanted to complete while being a little bit more flexible with scheduling. We collaborated effectively and adhered to the goals and standards we had created in MS0. Overall, we accomplished what we wanted to complete and the goals that we set for ourselves were challenging yet fulfilling once achieved.

Scope grade:

We would give ourselves a very strong good scope grade. Our primary target for excellent scope with MS1 Alpha was to hit 500 lines of code and we fell slightly short at around 400. We knew that it would be difficult to hit this mark, but we still got very close and we are happy with the results and functionality of the current state of our project. Additionally, we had noted in our proposal that we wanted to consider speed of the pricing model for excellent scope, but we no longer saw a practical need for speed considering we wouldn't be doing any live trading with our implemented models. Our shift to focusing on user needs and user experience influenced this decision and the overall direction of the project. At the end of the day, we believe our work reflects good scope and very close to excellent scope.

Goals for next sprint:

1. Satisfactory: Have the basic documentation and understanding of the Binomial Tree OPM and set up the user interface.
2. Good: A basic implementation of the Binomial Tree OPM that uses well documented functions as well as completion of the Black-Scholes OPM but for put options. This implementation should be able to properly price models based on set values and example problems which we will work out by hand and find online. This should be able to accurately price options using the user interface.

3. Excellent: A fully working Binomial Tree OPM and Black-Scholes OPM that can provide an accurate price as well as an interactive GUI for the user to input the parameters for their options with possible data visualization.

Notes:

N.B: in case you run into issues please see https://github.com/advayk/CS3110-Options-Pricing the documents section has our INSTALL.txt file as well.