COMPUTING IN FINANCE: Class Project: Fall 2010

Forming Teams:

You are required to do this class project in **groups of up to 4 members**. You should elect one member of your group to be the team leader. **Groups once formed cannot be changed midway through the project**, **as this is extremely disruptive.**
Since you will be spending many hours and many (programming) nights together, you should form a group

a) That will work well together,
b) Where members have minimal schedule conflicts
c) That collectively houses key competencies i.e. Math, Finance, Computer Science.

Although not required, you are welcome give your team a "name" such as "Houston Rocket Scientists", "Courant Quant-Sox" , "Martingale Madrid" ….etc.
Your team leader should email names of your team members to the TA and Kishor and cc each member of your team. This should be done by **9am October 01**.

Roles & Responsibilities:

Your team LEAD is responsible to help facilitate the planning of the project and help resolve any conflicts. The entire team will plan the project under the guidance of your team leader. **Planning** involves identifying *what* should be done (tasks), *who* should do it (resources), *when* tasks should be done (time frames) and *how* tasks are best sequenced (dependencies).
In addition, your team leader will submit the teams **"self evaluation"** of the project. The criteria for self evaluation will be provided to you separately by Thanksgiving.

In many ways, this project will expose you to aspects of a typical work experience. However, one common practice used in the industry will be *completely absent* in this project and it is *outsourcing*. All work for this project should be done entirely by the group i.e. no outsourcing.

As you work together, you should expect to challenge each others' ideas and find differences of opinions on various topics. Remember to be constructive, objective and understanding without personalizing your comments. Conflicts are best resolved by the group but those that cannot be resolved by the group should be directed to the TA or Kishor.

For each section in the project, you will work in pairs with one team member designated to the each of the following roles.

a) "DEVELOPMENT": to design and implement the solution.
b) "VALIDATION". to validate the solution and find areas of improvement.

Other than the above, your team will collectively agree on how you will operate.

Final Submission:

You will submit your code and report tar/zipped via email. The final projects are due at 9 am on December 15.You will write a 2 page executive summary of your work and a detailed project report. In your report, you will

a) Write about the specific requirements outlined in each section.
b) For each section, name the members responsible for Development and Validation.
c) For each section, the Development unit should write a brief description of their design approach, design choices and implementation specifics.
d) For each section, the testing unit should write a brief assessment, which should indicate how validation was done, the test cases used and the pitfalls discovered.
e) Include the self evaluation of the project.

Project Review:

You are encouraged submit/show your work in its state "as-is" in the week of Nov 7 for the TA to review. This is entirely optional and earns no credit. It helps review your design, documents and code with the TA and streamline your process of submission. In addition, you should review any questions/issues with the TA/Kishor on a regular as-needed basis.

Getting the data you need

Although you may be provided input and test data for the project in some cases, you should plan on learning how to get the data you need. This involves using the Bloomberg terminal available to you as well as getting familiar with other sources of data on the internet.

Input and output of data and information to your code should be based entirely on ASCII-text files. There is no requirement to build an application/web/excel interface but you are welcome to do whatever is needed to make validation easier.

Project Description

The sections that follow describe the project.
They constitute minimum requirements – so should you choose to pursue one or more sections in more depth or detail, feel free to do so and let me know.
All sections must be done in C++ unless otherwise specified.
Sections A, B, C, and G must be done in C++ and each team member should implement (which means be the "DEVELOPMENT" person) for at least one section in C++. Sections D, E, F, and H can be done in R/MATLAB or C++
Section I can be done for extra credit. Any extra credit awarded will help your grade in the project only and will not affect the grade in your final exam.

A) Building objects for yield curves and underlying asset for stock.

(i) Yield Curve: You will write an object that represents the behavior of the term structure of interest rates with a view towards using it as a market parameter needed for pricing derivative securities.
The object will be identified by a name and can be thought of as a collection of yield-points.

A yield point is defined by

Rate, term (years), type (cash or swap), and dayCount (30/360, ACT/365….).

The object should be able to provide methods for

spotRate (t) spot rate at time t
discountFactor (t) discount factor at time t
forwardRate (t,T) forward rate at time t for period T

Your methods will use a method of interpolation of your choice (linear, cubic etc.) Your design should be such that your interpolator can be easily switched in a plug-n-play fashion
Use the bootstrapping method to compute your discount function.
To test your object, build a US-Dollar yield curve by using LIBOR rates up to 2 years and swap rates from 2-30 years.
As part of your report, document your test case for the US Dollar LIBOR Curve and plot a graph of

1) Discount Factor vs T
2) 6 month Forward Rate vs T

(ii) Stock Underlying: You will write an object that represents the characteristics and behavior of a common stock with a view towards using it as a market parameter needed for pricing derivative securities.
At the very least, the following attributes should be included in a stock object

Currency of Denomination
Yield Curve
Dividend Schedule: This is a list of dividend amounts paid on specific dates

Your stock object should have methods that provide users of the object with

Market Price
Forward Price, F (T)

(iii) Rate Curve Extensions: You are required to design objects to build curves for Treasury bonds, UK Gilts and US Municipal Bonds. Indicate if your object framework would accommodate these requirements. Do you find portions that are reusable? Would you choose to reuse this object design or build a new one from scratch? Would your answers change if implementing a high performance solution is an imperative?

(iv) Underlying Asset Extensions: You are required to design objects to build commodities and currencies as underlying assets. Indicate if your object framework would accommodate these requirements. Do you find portions that are reusable? Would you choose to reuse this object design or build a new one from scratch? Would your answers change if implementing a high performance code is an imperative?

B) Building a volatility surface.
In this section, you will write an object that represents the characteristics and behavior of volatility surfaces. You will validate your volatility surface by using it to price options and variance swaps.

(i) Volatility surface: A volatility surface is associated with assets such as stocks, interest rates, FX rates or baskets of assets. You will find that there are numerous volatility surface models and approaches that you can implement. Each approach has its own assumptions, methodology, technique and applicability. For the purposes of this assignment, you are free to choose any well-accepted approach of your choice. The simplest representation is a quadratic parameterization.

Your volatility object should offer the following methods

Variance (S, K, t)	variance at time t for underlying S, strike K
Volatility(S, K, t)	volatility at time t for underlying S, strike K
Forward Volatility (S, K, t, T)	volatility at time t and period T for underlying S, strike K

You should construct a volatility object for the S&P500 and the index level I.

As part of you report you should describe
- Assumptions, data and technique used
- Method for interpolation
- Plots of the volatility surface for T=0 to T=2 years and S/K=0.3 to S/K=2

(ii) European Options: Write a model that prices European options on stock using the Black Scholes formula. You should write methods that give you the following results for your option
a) Price, delta, gamma, vega, theta

Use Monte Carlo simulation to validate your option prices.

Test your model for options on the S&P Index using your yield curve and volatility objects as market parameters. Options should include those with the following strikes (90%, 100% and 110% of spot) and maturities (3 month, 6 month, and 12 months from today).

(iii) Option Strategies: Extend your model for a portfolio of European options and use it to provide the same results for the following option strategies.
a) Long Call Spread
b) Long Straddle

(iv) Variance Swaps: Extend your model to compute the value of a variance swap. Write methods that compute the fair value of the variance swap for the data provided.

(v) VIX: Use your variance swap model to compute the level of a one-month variance swap. Compare your results with the VIX level for the day.

C) Derivatives using Monte Carlo
In this section, you will design and write a Monte Carlo based framework that will allow you to price a variety of exotic products. Your framework will generate simulated paths for one (or more) asset, for every month for five years. Once you have these paths you will apply a corresponding "payoff formula" on the set of paths to get a price for European style products.

(i) Barriers: Single asset products will include the following Barrier Options (calls and puts).

Up and out.
Down and out.
Up and in.
Down and in.

(ii) Basket Options: Extend your simulation framework to handle more than one asset and basket options.

Basket Option n-asset max {(W1*S1 + W2*S2…+ WnSn - K), 0}

You can, but not required to, choose your assets from ETFs for which you have historical data. You should test your work for both calls and puts on baskets with 2, 5, and 8 assets.

(iii) Scenario Analysis: Show how the value of each of the products above depends on changes in

a) Spot price
b) Volatility
c) Volatility skews
d) Correlation

Validate your model using closed form solutions wherever applicable.
As part of your report, describe your test cases, results and a graph the difference between prices obtained using closed form and prices obtained using simulation.

D) Trading Strategies: Implied Volatility
Using data for historical implied volatility you are asked to assess if the data supports the relationships below. To substantiate your view on these relationships you should choose SPX and 5 ETFs for your analysis. You should pick the most convincing combination of statistical tests, measures such as correlation, regression analysis and plots. Your investigation should culminate with

a) An opinion on each relationship below along with
b) A possible empirical or intuitive explanation and
c) Possible trading strategies that exploit these relationships

The relationships are

- Long term skew is flatter than short term skew. Here skew is defined as the difference between 90% volatility and ATM-volatility.
- A sharply upward sloping ATM-Volatility term structure indicates an impending decline in the underlying index level.
- Short term curvature is greater than long term curvature.
- ATM-volatility is negatively correlated to the underlying index level.
- ATM-volatility increases on days leading up to FOMC meetings.

E) *i*-buddy

Your favorite close friend is thrilled that you are now taking courses at Courant. As the Gods of the alphabet would have it, she invites you out and asks you to be her ***i*-buddy,** a trusted advisor for her personal investments.
She executes securities transactions with a broker who tells her that she should be long the US equity market (S&P500) over the next year. She is in agreement with that view. Her broker is suggesting three ways of achieving this long exposure

1) Buy SPY, the ETF as a proxy for S&P500
2) Buy SSO, the ETF Ultra S&P500
3) Buy ELN, a one-year equity linked over the counter note whose payoff is as follows
   If the Ending Value of SPX is greater than the Starting Value,
      3-to-1 upside with 18% cap
   Else
      1-to-1 downside

She is looking to her ***i*-buddy** to provide her guidance based on the sound analysis of these three investment choices, and help answer questions such as

a) How would these have performed in if the investment was made Jan 1 every year for the last 10 years? How would you characterize the distribution of returns?

b) What questions would you ask her to understand her preferences as an investor about target return, downside protection, capping performance, dividends, leverage, investment horizon, volatility, skew and kurtosis, correlations?

c) Which investment is the best a risk-return basis given her investment preferences? What type of investor, as characterized by their investment preferences, would find the other (remaining) two choices more attractive and suitable?

d) If the macroeconomic forecasts suggest that we are in a period of low growth and high volatility, which is a better investment for such an environment?

e) Which investment is most susceptible to tail risk?

f) She realizes that is more interested in 80% principal protection and is willing to take a 2:1 upside. What would be the fair value of the cap using the market data in project sections A, B?

Using historical data and your simulation framework, you should address the above questions and any other considerations that you may think are important. Document your analysis and use diagrams/charts as needed to illustrate concepts.

F) Reverse Convertibles

A reverse convertible is a typical yield enhancement product. The value of a reverse convertible bond can be thought of as a "combination" of a zero coupon bond value and a short down-and-in put. The issuer has the right to convert the bond to shares at maturity date. In exchange, the issuer pays an attractive coupon.

You are part of a structured derivatives team and have been asked to structure a reverse convertible with one year maturity for the key investor types in your classification. You can choose following parameters

a) Underlying ETF
b) Coupon
c) Conversion level (80% of initial price).

You should make reasonable classification of investor preferences based on target return, downside protection, capping performance, dividends, leverage, investment horizon, volatility, skew and kurtosis

(i) Conduct a back test for the any 5 ETFs as underlying securities for at least 5 years. Your back test should include a payoff profile; return distributions, measures of downside risk, volatility of returns, skew, kurtosis and any other useful measures of investment performance.

(ii) How would your results change if your product

a) Used an Equally weighted basket of the same 5 ETFs as an underlying.
b) Was a worst of reverse convertible with the same 5 ETFs

Indicate how each of these is sensitive to correlations.

(iii) You should also show how your results change if the conversion level is

a) Lower (70%) and
b) Higher (90%) than in your initial analysis.

(iv) You should make your own macroeconomic forecast/assumptions for the next year. Given your market view, you have been asked to find a methodology to choose which ETFs constitute the most attractive candidates on a risk return basis for different investor types.

Document your approach for back testing and results using graphs/charts as appropriate.

This section can be done in R/MATLAB or C++.

G) Convertible Bonds

The value of a convertible bond can be thought of as a "combination" of the straight bond value, value of the conversion option, value of the issuer call and the value of any put. In this section you will use a binomial/trinomial tree to compute the value of the convertible bond. You can assume that the stock price is the only stochastic variable in your model. At maturity, the holder either receives par, or the conversion value of the underlying shares. If it is optimal to put, the value of the convertible is the put value. If it is optimal for the issuer to call, the value of the convertible is the greater of the conversion value and the call price.
Write methods that compute the following

- Price (fair value of the convertible)
- Parity (today's conversion ratio *stock price)
- Parity Delta (change in bond value/change in parity)
- Interest Rate Delta (change in bond value/change in interest rates)
- Parity Gamma (change in parity delta/change in parity)

Test your model for a simple convertible bond of your choice.
This section must be done in C++.

H) Volatility and Correlations.

You have been provided a wide spectrum of ETFs that can help you get exposure to broad equity markets, sectors, country markets, commodities and currencies. Using data for these ETFs as a proxy for the market you have been asked to identify relationships between these markets/assets. Specifically, using a 2-year look back window you have been asked to determine the following

a) Correlations between these ETFs
b) Betas between these ETFs
c) Beta of these ETFs to SPX

Extending back to as far as the data goes using multiple 2 year look back windows determine if the correlations and betas are volatile and how they change over time. Substantiate your findings with empirical observations.
Based on your results you should validate, reject or qualify the following beliefs used by some while choosing investments
a) Choosing assets with consistently high correlations to each other does not help reduce risk.
b) Choosing assets with consistently low correlations to each other does help reduce risk.
c) Assets with consistently high betas can be used as hedges to each other.

I) Extra Credit: High Performance Computing: GPU

Implement the section "Derivatives using Monte Carlo" using GPUs available at Courant.
You should use the C like language appropriate for those GPUs.
Indicate the nature of performance improvements you observe.
Explain how and why your code needed to adapt to make it GPU-ready.
Note: This exercise involves exploration in a 'new' high performance computing technology. Feel free to explore as much as you can. You can earn partial credit even if you do not do it all.