Prof. Chow Ying-Foon, Ph.D.
Room 1215, Cheng Yu Tung Building
Department of Finance
The Chinese University of Hong Kong

2017–2018 Second Term

Phone: 3943 7638
Fax: 2603 6586
Email: yfchow@baf.cuhk.edu.hk

FINA 4150 A Quantitative Methods for Financial Derivatives
Suggested Project Topics
Due Date: May 2, 2018

1 Fitting the Extended Vasicek Model to Market

One of the advantage of the extended Vasicek model is the flexibility to accommodate the market prices by choosing the functional form of time-varying parameters, e.g., θ as in the stochastic differential equation

$$dr(t) = \kappa(\theta(t) - r(t))dt + \sigma dB(t)$$

The prices of zero-coupon bonds maturing at date u are given by

$$P(0, \tau) = \exp(-\phi(s)ds - a(\tau) - b(\tau)r(0))$$

where

$$\begin{aligned}
\phi(s) &= \int_0^s e^{-\kappa(s-y)}\theta(y)dy \\
a(\tau) &= -\frac{\sigma^2}{4\kappa^3}(2\kappa\tau - e^{-2\kappa\tau} + 4e^{-\kappa\tau} - 3) \\
b(\tau) &= \frac{1}{\kappa}(1 - e^{-\kappa\tau})
\end{aligned}$$

The project involves collecting data of bond prices (zeros or coupons, 20 plus different bonds), then using the data to fit the model prices. That is to find a function $\theta(s)$ such that

$$P(0, \tau) = P^{\text{mkt}}(0, \tau)$$

where P^{mkt} is the market price and the short rate can be approximated by 1-or 3-month T-bill rate.

A simple way of fitting yield curve in this context is to model $\theta(s)$ or $\phi(s)$ as a piece-wise step function of time, i.e., $\theta(s) = a$, if $0 < s < 1$ and $\theta(s) = b$, if $1 \leq s < 3$, etc. As the the steps are getting smaller (adding more bonds with observed market prices), this step function can converge to a smooth function.

We can then extend the above by including the time-varying $\kappa(t)$. Besides choosing $\theta(s)$ or $\phi(s)$ to fit the bond prices, you first need to find out $\kappa(t)$ to fit the yield volatilities (term structure of volatilities), which you have to estimate from historical yields for several (7+) maturities.

As a comparison, you will fit the general Hull–White model. Now all parameters are time-varying. You have to first fit the model to the yield volatilities by choosing $\kappa(t)$, then to the cap prices (market data for caps can be downloaded from Bloomberg or Reuters Eikon) by choosing $\sigma(t)$, and finally to the bond prices by choosing $\theta(t)$ or $\phi(t)$.

Your report should include but not limit to:

- a fairly detailed description about the data used in your analysis, e.g., the original data themselves, the date of the data, and the data used to estimate κ and σ, etc.;
- the program used to calculate the function θ (VBA or other programming language code, e.g., Octave);
- the results of the fittings, e.g., some graphs of the fitted function θ;
- the fitted yield curve (term structure) itself, and some comparisons with the yield curve reported by data providers.

You are free to use any references. The basic references are [1] and [2].

References

[1] Back, Kerry, 2005, *A Course in Derivative Securities* (Springer).

[2] Hull, John C., 2017, *Options, Futures, and Other Derivatives*, tenth edition (Pearson).

2 Fixing Volatile Volatility

Despite its simplicity and popularity among practitioners, it is commonly recognized that the Black–Scholes–Merton (BSM) model only partly captures the complexity of financial markets and produces a persistent bias in pricing derivatives. A clear misspecification of this model is the assumption that stock returns are normally distributed. Indeed, nowadays there is abundant empirical evidence that return distributions exhibit sizable negative skewness and large kurtosis. Moreover, the assumption of constant volatility is particularly limiting and contrasts with volatility smiles and smirks implicit in market option prices.

Much effort has been put into the research on more realistic approaches that relax the most restrictive assumptions of the BSM model. A large variety of sophisticated models have been proposed by assuming different processes for stock prices, interest rates and market prices of risks. For a review see [3]. The drawback of these models is that closed-form formulae are rarely available and the advantage of a more realistic description can be offset by the costs of implementation and calibration. Among such alternative models, stochastic volatility (SV) models have attracted a great deal of interest. In an SV framework, volatility changes over time according to a random process usually assigned through a suitable stochastic differential equation (s.d.e.). In this manner, SV models manage to reproduce empirical regularities displayed by the risk-neutral density implied in option quotations.

We will follow [4] model and consider an extension allowing for the inclusion of jumps in the stock return dynamics as proposed by [2] and by [1].We aim at testing the ability of the aforementioned models to fit market prices in comparison to the standard BSM model.

References

[1] Bakshi, Gurdip S., Charles Cao, and Zhiwu Chen, 1997, Empirical performance of alternative option pricing models, *Journal of Finance* 52(5), 2003–2049.

[2] Bates, David S., 1996, Jumps and stochastic volatility: Exchange rate processes in Deutsche Mark options, *Review of Financial Studies* 9(1), 69–107.

[3] Gatheral, Jim, 2006, *The Volatility Surface: A Practitioner's Guide* (Wiley Finance).

[4] Heston, Steven L., 1993, A closed-form solution for options with stochastic volatility with application to bond and currency options, *Review of Financial Studies* 6(2), 327–343.

3 An "American" Monte Carlo

Several pricing methods for American-style options have been proposed in the specialized literature (see, e.g., [4] for a quick overview). In the case of options written on a large basket of underlying assets, these techniques tend to become quite inefficient and computationally slow. In order to overcome these difficulties, [3] proposed a simulation-based method, which suffers, however, from numerical instability problems. [1] proposed an alternative based on the notion of Malliavin derivative. However, it suffers from a computational drawback in terms of speed for large baskets. [2] suggest a stochastic mesh method for overcoming these issues and their proposal has gained popularity among practitioners.

We shall investigate a method to price American-style options using Monte Carlo simulation as proposed by [6]. This method is an alternative method to the famous simulation-based technique introduced by [5]. [6]'s proposal shares the starting point of the analysis with [5]. They both start with the traditional dynamic programming equation, which conveys the idea that pricing an American-style option is a problem of knowing, at each point of time, whether it is worth to exercise the option immediately or to continue holding the option. As opposed to [5] who attempt to determine an optimal exercise policy, the method in [6] transforms the dynamic programming equation in a dual problem, proposes a financial interpretation of this latter, and finally tries to solve it numerically.

References

[1] Bouchard, Bruno, Ivar Ekeland, and Nizar Touzi, 2004, On the Malliavin approach to Monte Carlo approximation of conditional expectations, *Finance and Stochastics* 8(1), 45–71.

[2] Broadie, Mark, and Paul Glasserman, 1997, Pricing American-style securities using simulation, *Journal of Economic Dynamics and Control* 21, 1323–1352.

[3] Carrière, Jacques F., 1996, Valuation of the early-exercise price for options using simulations and nonparametric regression, *Insurance: Mathematics and Economics* 19(1), 19–30

[4] Lamberton, Damien, and Bernard Lapeyre, 2007, *Introduction to Stochastic Calculus Applied to Finance*, second edition (Chapman & Hall).

[5] Longstaff, Francis A., and Eduardo S. Schwartz, 2001, Valuing American options by simulation: A simple least squares approach, *Review of Financial Studies* 14(1), 113–147.

[6] Rogers, L. C. G., 2002, Monte Carlo valuation of American options, *Mathematical Finance* 12(3), 271–286.

4 Estimating the Risk-Neutral Density

Investors, risk-managers, monetary authorities and other financial operators are confronted with the need to assess market expectations concerning a number of fundamental macroeconomic variables, such as exchange and short-interest rates, stocks, commodities, stock indices. In simple terms, they need to estimate the probability distributions of future events. Information embedded in market prices of derivative assets provides central banks and operators with timely forward-looking information on market expectations regarding the underlying fundamental factors.

[1] presents an overview of the most common methods used to extract risk-neutral density functions from option prices. Market participants' expectations are too heterogeneous and complex to be captured using simple descriptive statistics, such as means and other point estimates. However, this information can be extracted from prices of traded options. Indeed, due to a nonlinearity in their payoffs, option prices across different strikes for a common maturity allow the assigning of probabilities to a wide range of possible values taken by the underlying asset at that maturity. These probabilities represent a synthesis of market expectations about future trends as perceived by operators at a given point in time. Moreover, option prices can provide additional information compared to the one stemming from a time series analysis of the underlying price process.

Consider a European call option stricken at K and maturing in τ years. The fair value of this option is given by the risk-neutral expected value of its discounted payoff. This value reads as an integral over the exercise region:

$$c_t(K,\tau) = e^{-r\tau} \int_K^\infty (x-K) q_{t+\tau}(x) dx$$

Here $q_{t+\tau}$ denotes the risk-neutral density of the underlying asset price at the expiration time $t+\tau$. Similarly, the fair price of a put option with equal features is

$$p_t(K,\tau) = e^{-r\tau} \int_0^K (K-x) q_{t+\tau}(x) dx$$

The inverse problem consisting of the identification of a risk-neutral distribution $q_{t+\tau}$ implied by option prices was first addressed in a seminal paper by [2]. These authors show that the risk-neutral density $q_{t+\tau}$ is recovered from option prices as

$$q_{t+\tau}(x) = e^{r\tau} \left. \frac{\partial^2 c_t(K,\tau)}{\partial K^2} \right|_{K=x}$$

Implementing this formula requires the knowledge of option prices for a continuum of strikes. Of course this is not possible in practice and infinitely many density functions are compatible to any given set of option prices over a finite range of strikes. However, some basic constraints have to be satisfied when constructing a risk-neutral density. For example, a well-defined risk neutral density is nonnegative, integrates to one, and prices exactly all calls and puts.

References

[1] Bahra, Bhupinder, 1997, Implied risk-neutral probability density functions from option prices: Theory and application, Bank of England Working Paper No. 66. Available at SSRN: https://ssrn.com/abstract=77429 or http://dx.doi.org/10.2139/ssrn.77429

[2] Breeden, Douglas T., and Robert H. Litzenberger, 1978, Prices of state-contingent claims implicit in option prices, *Journal of Business* 51(4), 621–651.

5 Vanilla Options

5.1 Geometric Brownian Motion Model

- The purpose of this part is to well understand the Geometric Brownian Motion (GBM) model. As we know, the commonly used constant GBM model for stock prices is given by

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t \quad \text{or equivalently} \quad dS_t = S_t\left(\mu dt + \sigma dW_t\right) \tag{1}$$

  where S_t is the stock price at time t, μ is the expected rate of return of the stock, σ is the volatility of the stock, and W_t is a standard Brownian motion (or Wiener process).

- Its discrete version, which is practically more convenient to use than the continuous model, is written as

$$\Delta S_t = S_t\left(\mu\Delta t + \sigma\sqrt{\Delta t}\varepsilon_t\right) \tag{2}$$

  where Δt is the length of the small time interval, ΔS_t is the stock price change in the time interval (i.e., $\Delta S_t = S_{t+\Delta t} - S_t$), ε_t is a standard Gaussian random variable (i.e., $\varepsilon_t \sim \mathcal{N}(0,1)$). This discretized GBM model can be used to generate future stock prices by Monte Carlo simulation once the two parameters μ and σ are known.

- This part consists of two parts: (1) fit the model using market data; (2) generate future prices by Monte Carlo simulation.

5.1.1 Use Market Data to Estimate Model Parameters μ and σ

1. Choose a specific stock and index you want to study. For instance, 00005.HK, MSFT, IBM, WMT, KO, etc., for individual stock; Dow, S&P500, Nasdaq, HSI for index.

2. Choose a period of one year (or more) from the recent past and download the historical prices of the stock for the chosen period. For example, 01/01/2017 – 02/28/2018, 03/01/2017 – 02/28/2018, etc.

3. Display the prices on a graph. Use the horizontal axis for time, vertical axis for price, and put the asset name in the title of the graph.

4. Estimate the annualized expected rate of return μ and the volatility rate σ from the historical data.

Remark *You must clearly state which asset you choose, the exact time period for which the historical prices you use, where you obtain the data, what method you use for estimating the two parameters (include mathematical formula if necessary).*

5.1.2 Generate Future Stock Prices Using Monte Carlo Simulation

Here you are asked to use the parameter values for μ and σ obtained from above and the discretized model (2) to randomly generate future prices for the stock and then answer questions based on your simulated future price trajectories.

1. Find the stock price on 02/28/2018. Assume that is the most recent end-of-day price of the stock that you can get from the market quote. Use this number as the initial price S_0 in your simulation.
2. Select a small number for the time interval Δt. One day would be a good choice and is recommended for use. However, you can try different values, for example, one hour, one minute, etc.
3. Use simulation to randomly generate 1000 trajectories for the future stock prices for one year. If you use one day for Δt, 250 random prices should be generated for each trajectory; one for each trading day, assuming there are 250 trading days in a year.
4. Display the first 10 trajectories on graph. Use the horizontal axis for time and vertical axis for price. You may plot each trajectory on a separate graph, or put all of them on one graph. For the latter, try to use different color or line type for different trajectories.
5. For each of the trajectory you produced, find/calculate and report the following statistical quantities:
   - Maximum and minimum prices.
   - Average price for the one year period.
   - Average rate of return.
   - Standard deviation of the rate of return.
6. Based on your generated price trajectories and the calculated quantities, answer the following questions:
   - Are the simulated future price trajectories similar to the historical realization you obtained in part 5.1.1?
   - Are the average rates of return and the standard derivations you calculated from the simulated future price trajectories close to or significantly different from the estimated values for μ and σ in part 5.1.1?
   - What comments/explanations do you have?

Remark *You must clearly state which specific day the stock price you use as S_0, what value you choose for Δt in your simulation, how the Gaussian random numbers are generated (if you use a built-in function from the software you use, mention the function), how the future prices are generated (include mathematical equations if necessary), what method/formula you use to calculate the required statistical quantities.*

5.2 Black–Scholes–Merton Option Pricing Model

- The purpose of this part is to understand the Black–Scholes–Merton (BSM) model for option pricing and to be able to calculate the risk-neutral prices for standard options. To successfully complete the question, you need to go over the BSM model as well as the algorithms thoroughly.

- You need to specify the options first. A list of requirements is stated below that you should follow to specify the options you are going to study.

- The stock that you have chosen in part 5.1 should be used as the underlying asset of the options. Choose option parameters in the way specified as following:

  1. **Stock Price** S. Set $S = S_0$ and use the 1000 trajectories that you have generated in part 5.1.
  2. **Strike Price** K. Choose from (a) $K = S$, (b) $K = 1.1S$, (c) $K = 1.2S$, (d) $K = 0.9S$, (e) $K = 0.8S$, where S is the stock price.
  3. **Volatility** σ. Use the annualized volatility that you have estimated from the historical data in part 5.1.
  4. **Maturity** T. Choose from (a) $T = 3$ months, (b) $T = 6$ months, (c) $T = 9$ months, (d) $T = 1$ year, (e) $T = 2$ years.
  5. **Risk-Free Interest Rate** r. Use $r = 3\%$ for all options.
  6. **Dividend Yield** q. Choose from (a) $q = 0$ (no dividend), (b) $q = 1\%$.
  7. **Option Type**. Call and Put, European and American.

  This would produce totally 200 different options (100 European and 100 American options) for you to use in the project.

- This part consists of three parts: (1) valuing European options using the BSM formula; (2) valuing European options using Monte Carlo simulations; (3) valuing both European and American options using binomial trees.

5.2.1 Calculate Option Price Using the BSM Formula

Calculate the prices of 4 European call options and 4 European put options using the BSM formula. Among the 8 options you select:

- At least one option should have a non-zero dividend yield (i.e., $q = 1\%$), at least one option should be at-the-money (i.e., $K = S$);

- At least one option should be in-the-money (i.e., $K < S$ for call and $K > S$ for put);

- At least one option should be out-of-the-money (i.e., $K > S$ for call and $K < S$ for put).

Remark *For this part you can either work out the calculations in Excel, or write a macro using VBA. You must clearly state the specifications for each option you use and report your calculated option value. A neat way would be using a table.*

5.2.2 Calculate Option Price Using Binomial Method

Use the following values for the number of steps m of the binomial tree: $m = 4, 8, 16, 32$ (**Optional:** $m = 100, 500, \ldots$)

1. For each m, determine the length of step Δt, the risk-neutral probability p, the up move factor u, and the down move factor d for the binomial tree. Report your calculation results.

2. Calculate the prices of the same two European options you have used in part 5.2.3 by the Monte Carlo method for different m as specified. Report and compare your results.

3. Change the option type from European to American and keep all other parameters to be exactly the same. Calculate the prices of the two American options by the binomial tree method for different m as specified. Report and compare your results. Also compare the prices between the European option and the corresponding American option.

Remark *For this part you can either work out the calculations in Excel, or write a macro using VBA.*

5.2.3 Calculate Option Price Using Monte Carlo Simulation

- Write a program (e.g., macro using VBA) to implement the Monte Carlo simulation algorithm for pricing European options. In your implementation, set the number of independent sample paths of the underlying stock prices as a variable (say n).

- Use your program to calculate the price of one European call option and one European put option. For the purpose of comparison, the two options you use here should be among those already valued in part 5.2.1 by BSM formula. Calculate each option price using the following values for n : $n = 100, 1000, 5000, 10000$ (**Optional**: $n = 50000, 100000, 500000, 1000000, \ldots$). Report the results and compare them with the exact price obtained by BSM formula. You should observe that as n increases, the Monte Carlo approximations become closer and closer to the exact value.

Remark *The same procedure used in part 5.1 for generating future stock prices can be used here to generate each sample path. However, for option pricing, the GBM model should be the **risk-neutral model**, not the real world model.*

5.2.4 Important Note

Softwares that implement these standard option valuation methods can be easily found on Web (for example, you can download Hull's DerivaGem program from his web). By simply typing the option parameters, you could get the correct results that are expected in this project. Obtaining option prices in this way is **NOT ACCEPTABLE**. You **MUST** implement the methods yourself. However, you may use these tools to check the results obtained from your implementations.