Semester project (January 2020)
Estimation of call price using Monte Carlo simulation and variance reduction

Contents

1 Simulation of Brownian Motion

Presented by: D.V.S. Abhijit

Brownian Motion gets its name from the botanist Robert Brown who studied irregular motion of pollen particles suspended in water. Mathematically Brownian motion is modelled by a continuous-time stochastic process called the Wiener Process.

Definiton 1.1 The family of Random variable $\{\mathrm{B}_t\}_{t\geq 0}$ *is called Standard Brownian Motion if* :

1. $\mathrm{B}_0 = 0$.

2.$t \to B_t$ *is almost surely continuous, i.e.*

$P\{\omega \in \Omega | the\, path\, t \to B_t(\omega)\, is\, a\, continuous\, Function\} = 1$

3.B_t, has independent increments i.e. $\forall\, t_1, t_2 ..., t_n$,$\mathrm{B}_{t_2} - B_{t_1}, ...B_{t_n} - B_{t_{n-1}}$,are independent random variables

4.$for\; 0 \leq s \leq t\, \mathrm{B}_t - B_s$ is independent of B_u,for u$\in$ $[0, s]$

5.$B_t - B_s \sim N(0, t-s)\; for\, 0 \leq s \leq t$

1.1 Simulating an approximation to the Standard Brownian Motion

Firstly we generate a list of standard normal numbers using methods such as the Box-Muller method. The following algorithm is used to generate the discrete time approximation of the Standard Brownian Motion.
A small value h>0,is chosen for the time increment.
A sequence $(r_n)_{n\in\mathbb{N}}$,of random numbers from an N(0,h) distribution which can be generated by simple scale transformation of standard normal random numbers. Brownian Motion is then defined as:
$B_{t+h} = B_t + r_n, \forall t, \in \{0, h, 2h, ...\} r_n \in (r_n)\; and$,$\mathrm{B}_{t+h} - B_t \sim N(0, mh)$ since $B_{t+h} - B_t$ is a sum of m IID variables of the form N(0,h).
The above algorithm provides us with discrete points.
Since Brownian Motion is almost surely continuous we/software joins the discrete successive points by straight lines for a continuous map.

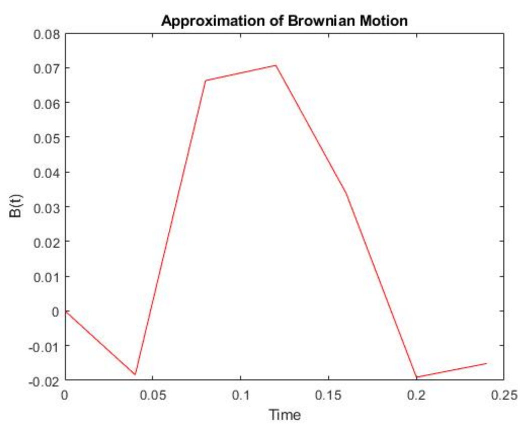


Figure 1: Simulation of Brownian Motion

2 Geometric Brownian Motion

Presented by: Malavika Biju

A geometric Brownian motion (GBM) (also known as exponential Brownian motion) is a continuous-time stochastic process in which the logarithm of the randomly varying quantity follows a Brownian motion (also called a Wiener process) with drift. It is an important example of stochastic processes satisfying a stochastic differential equation (SDE).

Definiton 2.1 Geometric Brownian Motion, S(t), is a continuous-time stochastic process in which the logarithm of the randomly varying quantity follows a Brownian motion (also called a Wiener process) with drift. It is defined by

$$S(t) = S_0 e^{X(t)}$$

where $X(t) = \sigma B(t) + \mu t$ is BM with drift and $S(0) = S_0 > 0$ is the intial value. Taking logarithms yields back the BM. $ln(S(t)) = ln(S_0) + X(t)$ is normal with mean $\mu + ln(S_0)$, and variance $\sigma^2 t$; thus, for each t, S(t) has a lognormal distribution.

Letting $S(t) = S_0 e^{X(t)}$, where $X(t) = \sigma B(t) + \mu t$ is BM with drift μ, and variance σ^2, we solve for new values for μ and σ (denoted by μ', σ'), under which the pricing is "fair", that is, such that the discounted prices e^{rt} S(t) : $t \geq 0$ form a martingale, which here means that $E(S(t)) = e^{rt}$ S_0, $t \geq 0$. But we know that $E(S(t)) = e^{Rt}$ S_0, where $R = \mu + \sigma^2/2$ so we conclude that we need $\mu + \sigma^2/2 = r$. This is accomplished by, $\sigma' = \sigma$, but changing the drift term μ to μ'=$r - \sigma^2/2$ (the risk-neutral drift).In other words, when pricing the option we need to replace S(t) by its risk-neutral version $S'(t) = S_0 e^{X'(t)}$, where $X'(t) = \sigma B(t) + \mu' t = \sigma B(t) + (r - \sigma^2/2)$t.

2.1 GBM in Finance

Most economists prefer Geometric Brownian Motion as a simple model for market prices in contrast to Brownian Motion, even Brownian Motion with drift. In particular, in mathematical finance, it is used to model stock prices in the Black Scholes model.Some of the arguments for using GBM to model stock prices are:

1.Under GBM the return to the holder of the stock in a small period of time is normally distributed and the returns in two non-overlapping periods are independent, which agrees with what we would expect in reality.

2.A GBM process only assumes positive values, just like real stock prices.

3.In the stochastic differential equation for Geometric Brownian Motion, the relative change is a combination of a deterministic proportional growth term similar to inflation or interest rate growth plus a normally distributed random change similar to the kind of 'roughness' in the paths seen in real stock prices.

4.GBM also can (in a limited and approximate sense) be justified from basic economic principles as a reasonable model for stock prices in an "ideal" non-arbitrage world. In this scenario, no one should be able to make a profit with certainty, by observing the past values of the stock. The stock prices generated through GBM offers no such arbitrage opportunities.

2.2 Drawbacks of GBM

1.In real stock prices, volatility changes over time (possibly stochastically), but in GBM, volatility is assumed constant. To make GBM more realistic as a model for stock prices, one can drop the assumption that the volatility (σ) is constant. If we assume that the volatility is a deterministic function of the stock price and time, this is called a local volatility model. If instead we assume that the volatility has a randomness of its own—often described by a different equation driven by a different Brownian Motion—the model is called a stochastic volatility model.

2.In real life, stock prices often show jumps caused by unpredictable events or news, but in GBM, the path is continuous (no discontinuity). Also other assumptions such as no arbitrage opportunities is not applicable in real life.

2.3 Simulating an approximation of GBM

For simulating GBM, we use the formula $S_{t+h} = S_t(1 + \mu h + \sigma r)$. Here h is the length of the small time interval and r is a normal random variable from the distribution N(0,h).Any small time interval h can be used in the simulation. In the limit as h goes to 0, a perfect description of the stochastic process is obtained. The following algorithm is used to generate the discrete time approximation of Geometric Brownian Motion.We input a small value $h > 0$ for the time increment. A sequence $(\mathrm{r}_n), n \in \mathbb{N}$ of random numbers from an N(0,h) distribution. The discrete points in the path of the motion are then obtained using the formula $S_{t+h} = S_t(1 + \mu h + \sigma r)$ $\forall t \in \{0, h, 2h, ...\} r_n \in (r_n)$ Since GBM is a continuous time Markov process, we use a line plot to obtain a continuous path.

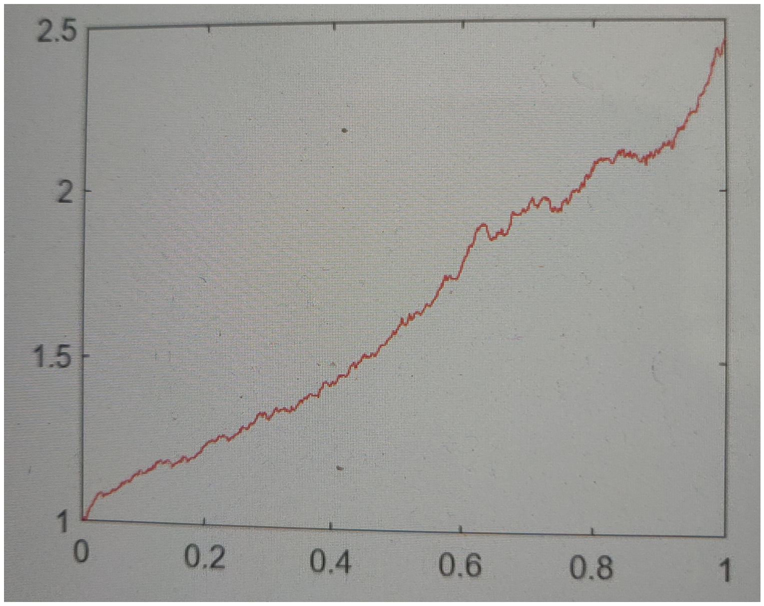


Figure 2: Simulation of GBM (On X axis: Time; On Y axis: S(t))

3 A Variance Reduction technique - Antithetic variates method

Written by : Shruthi Ravindra Bharadwaj

Let X_1,X_2,....,X_n have a given distribution and suppose, we are aiming to compute $\pi = E[\mathrm{g}(X_1,X_2,....,X_n)]$ where g is some function of interest. However, computing this π may not always be analytically feasible and we have to figure out other ways to estimate π. One such way is to use the Monte Carlo simulation technique.

Monte Carlo simulation technique is one of the widely used computational technique which on the basis of repeated random sampling obtains numerical results which are otherwise difficult to obtain analytically. Considering the direct Monte Carlo simulation technique in our scenario, it is done as follows:

First we generate $X_1^{(1)}$, $X_2^{(1)}$,....,$X_n^{(1)}$ which has the same distribution as X_1,X_2,....,X_n and we set

$Y_1 = \mathrm{g}(X_1^{(1)}, X_2^{(1)},....,X_n^{(1)})$

Similarly, repeat this simulating of random variables for k-1 more times, where k is predetermined and set

$Y_i = \mathrm{g}(X_1^{(i)}, X_2^{(i)},....,X_n^{(i)})$

for each i, where i ranges from 2 to k. All these k sets should be generated independent of each other. Now, we have k independent and identically distributed random variables, Y_1, Y_2,....,Y_k, each having the same distribution of g(X_1,X_2,....,X_n). Now letting $\overline{Y}$ denote the mean of these k random variables, then mean being an unbiased estimator of population mean,

$E[\ \overline{Y}\ ] = \pi,$

$E[\ (\overline{Y}-\pi)^2\ ] = \text{Var}(\overline{Y})$

Hence, $\overline{Y}$ can be always used as an estimator for π. But, we are not only looking for an estimator of π but also, we want that estimator to have as low variance as possible.For $\overline{Y}$ being used as an estimator for π, Var($\overline{Y}$) = Var(Y_i)/k. However, Var(Y_i) has to be usually estimated from the the generated values as its value is not always known. Also, we need to note that the error in this Monte Carlo simulation, called the Monte Carlo error(standard deviation of Monte Carlo estimator) has square root convergence and hence, a large number of samples(large k) is required to obtain the accurate results. Now, to overcome this problem, we may try to find other methods to obtain an estimator with lower variance. Antithetic variates method is one such variation reduction technique used in Monte Carlo methods.

3.1 Antithetic Variates method

Now, let us remove the assumption of independence from Y_i's ie Y_1, Y_2,....,Y_k are identically distributed but need not be independent. Now, let k be even and k = 2m for some m≥1 and define, Z_i = (Y_i+Y_{m+i})/2 for i in the range 1 to m. Then both $\overline{Y}$(k) and $\overline{Z}$(m) (k and m in parenthesis indicate the number of points over which the average was calculated) are the estimators of π and are also equal.

Now, let us denote Z = (Y_i+Y_{m+i})/2 to be the generic Z_i. Computing the variances,

$Var(Z) = \frac{Var(Y_i)+Var(Y_{m+i})+2Cov(Y_i,Y_{m+i})}{4}$

$= \frac{Var(Y)+Cov(Y_i,Y_{m+i})}{2}$

In the earlier case, when Y_i's were iid, Cov(Y_i , Y_{m+i}) = 0 and hence, Var(Z) = Var(Y)/2 which further yields,

$Var(\overline{Z}) = \frac{Var(Y)}{k}$

However if, Cov(Y_i , Y_{m+i})<0, then Var(Z)<Var(Y)/2 and hence,

Var($\overline{Z}$) $< \frac{Var(Y)}{k}$

and hence, the variance is reduced. Hence, if we achieve to create some negative correlation within each pair (Y_1,Y_{m+1}), (Y_2,Y_{m+2}),....,(Y_m,Y_{2m}) but keep the pairs iid so that Z_i's are iid and hence, central limit theorem still applies but Var($\overline{Z}$) is reduced than what we were getting taking Y_i's to be iid. To see how this desired negative correlation can be achieved, let us suppose the random variables X_1,X_2,....,X_n are independent and, in addition, that each is simulated via the inverse transform technique. That is, X_i is simulated from $F_i^{-1}(U_i)$ where U_i is a random number and F_i is the distribution of X_i. Hence, Y_1 can be expressed as

Y_1 = g($F_1^{-1}(U_1)$, $F_2^{-1}(U_2)$,....,$F_n^{-1}(U_n)$)

As 1-U is also uniform over (0,1), it follows that $Y_m + 1$ defined by

Y_{m+1} = g(F_1^{-1}(1-U_1), F_2^{-1}(1-U_2),....,F_n^{-1}(1-U_n)),

follows the same distribution as Y_1. And as U and 1-U are negatively correlated, Y_1 and Y_{m+1} are negatively correlated and similarly doing this for all the pairs (Y_1,Y_{m+1}), (Y_2,Y_{m+2}),....,(Y_m,Y_{2m}) will lead to Var($\overline{Z}$) have lower variance than if each Y_i were generated by a new set of random numbers.

An important theorem proving that there will be reduction in variance with the use of antithetic variables whenever g is a monotone function.

Theorem 3.1 If X_1,X_2,....,X_n are independent, then, for any increasing functions f and g of n variables ,

E[f($\mathbf{X}$)g($\mathbf{X}$)] $\geq$ E[f($\mathbf{X}$)]E[g($\mathbf{X}$)]

where $\mathbf{X}$=(X_1,X_2,....,X_n).

Proof: We can prove this by induction on n. For the base case, ie n=1, suppose f, g are increasing functions of f, then for any x and y,

(f(y) - f(x))(g(y) - g(x)) ≥ 0 because if $y \geq x$ ($x \geq y$) , then both the factors are either non-negative (non-positive). As this is true for any x and y, for any random variables X and Y,

(f(Y) - f(X))(g(Y) - g(X)) ≥ 0 holds true and hence,

E(f(Y) - f(X))(g(Y) - g(X)) ≥ 0. Simplifying in the case when X and Y are iid , we get the result for $n = 1$ ie

E[f(X)g(X)] $\geq$ E[f(X)]E[g(X)]

Now, assuming the theorem is true for n-1 variables, let X_1,X_2,....,X_n are independent and f and g are increasing functions, then taking the conditional expectation of f($\mathbf{X}$)g($\mathbf{X}$) given $X_n = x_n$,

E[f($\mathbf{X}$)g($\mathbf{X}$)| X$_n = x_n$] = E[f(X_1,X_2,....,x_n) g(X_1,X_2,....,x_n) | $X_n = x_n$]

=E[f(X_1,X_2,....,x_n) g(X_1,X_2,....,x_n)] (as X_i's are independent)

$\geq$ E[f(X_1,X_2,....,x_n)] E[g(X_1,X_2,....,x_n)] (by induction hypothesis)

= E[f($\mathbf{X}$)| X$_n = x_n$]E[g($\mathbf{X}$) | $X_n = x_n$]

Now, upon expectations on both sides and using the result for base case, we obtain the required result ie E[f($\mathbf{X}$)g($\mathbf{X}$)] $\geq$ E[f($\mathbf{X}$)]E[g($\mathbf{X}$)]

Corollary: If U_1,U_2,....,U_n are independent, and k is either an increasing or decreasing function, then,

Cov(k(U_1,U_2,....,U_n),k(1-U_1,1-U_2,....,1-U_n)) ≤ 0

Remarks: There are other variance reduction techniques such as stratification which splits up the domain of sample points into separate regions and takes a sample of points from each such region, and combine the results to estimate E(g(X));variance reduction by conditioning which uses the idea that we can do part of the problem in closed form, and then do the rest of it by Monte Carlo or some other numerical method; control variates method which uses the analytic solution to a similar and simpler problem to improve the solution. Depending on the suitability, one can choose these variance reduction techniques and get the better estimators of interest.

4 Black-Scholes formula and Call option price for European Options using Geometric Brownian Motion

Presented by: Shruthi Ravindra Bharadwaj

4.1 Introduction

Black-Scholes model is a standard pricing model which is used to determine the theoretical value for a call or a put option. Based on the previous works of Louis Bachelier,Sheen Kassouf and Ed Thorp among others, Fischer Black and Myron Scholes formulated this model by demonstrating that dynamic revision of portfolio can remove the expected return of security and thus, making this model risk-free. Subsequently, Robert Merton expanded the mathematical understanding of this model and formulated the Black-Scholes option pricing formula. For this work,Merton and Scholes received the Nobel Memorial Prize in Economic Sciences in 1997. Although Black was ineligible for the Nobel prize due to his death in 1995,the Swedish academy mentioned him as a contributor. Let us first define some relevant terms of option pricing theory.

Definiton 4.1 An option is an agreement (contract) which gives owner the right, but not obligation to buy (call) or sell (put) an underlying asset or financial instrument at a specifed strike price (K) at maturity time of the contract (T), which is a specifed date.

4.2 Determinants of option value

1.*Current value of the underlying asset.* Options derive their value from an underlying asset and hence,changes in the value of this underlying asset affects the value of options on that asset. As call options gives the right to buy the underlying asset at a specifed strike price, an increase in the value of the asset increases the value of the call option. However, for Put options, an increase in the value of the asset decreases the value of the option.

2.*Variance in value of the underlying asset.* The buyer of an option has the right to buy or sell the underlying asset at a fixed price. The value of the option(both call option and put option) increases as variance in the value of the underlying asset increases. This is because buyers of options at maximum, lose the price they pay for them; but, they have the potential to earn significant returns due to heavy price movemnts.

3.*Strike price of the option.* As the call option holder acquires the right to buy at a specified strike price, the call option value decreases as the strike price increases.However, for put options, where the holder has the right to sell at a fixed price, the put option value increases as the strike price increases.

4.*Time to expiration on the option.* As time to expiration increases, more time is available for the value of the underlying asset to move and hence, increases the value of both call and put options.

5.*Dividends paid on the underlying asset.* To understand how dividend payments affect the value of the option, consider an option on the traded stock. When a call option is in-the-money, exercising the call option will provide the holder with the stock and also entitle him or her to the dividends on the stock in subsequent periods. Failing to exercise the option will mean that these dividends are forgone.Hence, the value of the underlying asset is expected to decrease if dividend payments are made on the asset during the life of the option. This impies that the call option value decreases with increase in the size of expected dividend payments, and the put option value increases with increase in the size of expected dividend payments.

6.*Riskless interest rate corresponding to life of the option.* Since the buyer of an option pays the price of the option up front, an opportunity cost is involved. This cost depends on the level of interest rates and the time to expiration of the option. The riskless interest rate also comes to picture of the valuation of options when the present value of the exercise price is calculated, since the exercise price does not have to be paid (received) until expiration on calls (puts). Increases in the interest rate will increase the value of calls and reduce the value of puts.

4.3 Black-Scholes' Call option price formula

Moving back to discussion on Black-Scholes model,the stock prices, $\{S_t\}_{t\geq 0}$ in BSM model follow Geometric Brownian motion whose dynamics is given by :

$dS_t = \mu S_t \, dt + \sigma S_t \, dW_t, S - 0 \geq 0$

where W_t is a Wiener process or Brownian motion,and μ ('the percentage drift') and σ ('the percentage volatility') are constants.

For the above SDE, the solution is given by:

$S_t = S_0 \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right).$

The Black-Scholes equation is a PDE, which describes the price of the option over time and also Black-Scholes model is only for the European style of option pricing. If at time t the stock price $S_t = S$ and $\phi(S; t)$ is the price of derivative as a function of stock price and time, r is the annualized risk-free interest rate, continuously compounded, μ is the drift parameter, volatility σ is the standard deviation of stock returns per unit time, then ϕ satisfies the following PDE:

$$\frac{\partial\phi}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2\phi}{\partial S^2} + rS\frac{\partial\phi}{\partial S} - rS = 0.$$

The corresponding solution, known as Black-Scholes formula for call option price with strike price K is,

$$C(S_t, t) = N(d_1)S_t - N(d_2)K\exp(-r(T-t)),$$

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

$$d_2 = d_1 - \sigma\sqrt{T-t}$$

where $N(.)$ is the CDF of Standard Normal distribution, T-t is the time to maturity, S_t is the spontaneous stock price used for immediate settlement, and r and σ are as mentioned above.

4.4 Assumptions and drawbacks of BSM model

The BSM model make certain assumptions:

1.The option is European which can only be exercised at expiration.

2.The underlying security does not pay a dividend.

3.Markets are efficient (i.e., market movements cannot be predicted).

4.No transaction costs in buying the option.

5.The stock price follows a GBM with constant drift and volatility.

6.No arbitrage opportunity.

7.It is possible to buy and sell any amount, even fractional, of stock.

These assumptions themselves lead to drawbacks of the model:

1.This model is not suitable for pricing style other than European call options.

2. It assumes that the underlying factors are known and constant throughout the life of option. However, this is not true in the real world scenario where the underlying riskfree rate, volatility, and dividends are unknown and may have high variance leading to change in their values in short span and thus option prices fluctuates very largely.

3.It also assumes the stock prices to have continuous sample paths,and thus, cannot model the opening gaps and jumps.

Inspite of these drawbacks, the BSM model is widely popular as it is simple and easily gives the value of the option.

Now, let us compare the Call option value we obtain from this Black-Scholes formula and the expected value of call option we obtain from simulating stock prices that follow Geometric brownian motion.

4.5 Comparison of the Call option value from Black-Scholes formula and the expected value of call option obtained from simulating stock prices following Geometric brownian motion

Firstly,we need to understand that we expect the values of call options obtained from these two methods to be the same and the reason for this expectation lies in the assumption of BSM model which assumes that the stock price follows a GBM with constant drift and volatility.

Risk neutral version of the S_t, where $\{S_t\}_{t\geq 0}$ follows GBM with drift , and variance σ^2, is S_t^* which is given as:

$S_t^* = S_0\text{exp}(\text{X}_t^*), where$

$\text{X}_t^* = \sigma W_t + (r - \sigma^2 \frac{}{2})\text{t}.$

Hence, we expect the discounted-expected-value form for the option price:

$C_0 = \text{exp}(-r\text{T})E^*(S_T$ - K$)^+$

$=\text{exp}(-rT)E(S_T^*$ - K$)^+$

$=\text{exp}(-r\text{T})E(\text{S}_0\text{exp}(\sigma W_t + (r - \frac{\sigma^2}{2})\text{T}$ -K$)^+$,

We can prove this by carrying out the integration to evaluate $E^*(S_T$ - K$)^+$ above and hence, yielding the Black-Scholes call option pricing formula. However, we can also check it in a simpler way by comparing the value obtained from the formula and the estimate of discounted expected value mentioned as above.

We use direct Monte Carlo method and also antithetic variates method to find the estimate of Call option price and then compare the results for Call option price obtained from the two algorithms for stock prices following GBM with constant drift and volatility.We expect the variance of the estimator obtained from antithetic variates method is lesser than that obtained from direct monte carlo method. Let us see how the Theorem 3.1 helps it explain.

We know that discounted expectation for call option price is given as:

$C_0 = \text{exp}(-r\text{T})\ E[S_0\text{exp}(\sigma W_T + (r - \frac{\sigma^2}{2})\text{T})\ \text{-K}]^+$

Here, notice that keeping S_0, K, T, σ, r fixed,

$\text{g}(W_T) = \text{exp}(-r\text{T})\ (S_0\text{exp}(\sigma W_T + (r - \frac{\sigma^2}{2})\text{T})\ \text{-K}),$

is an increasing function of W_T ($\sigma \geq 0$) and hence,

$\text{f}(W_T) = \text{exp}(-r\text{T})\ (S_0\text{exp}(-\sigma W_T + (r - \frac{\sigma^2}{2})\text{T})\ \text{-K})$

is an decreasing function of W_T. Hence, -f is increasing function of W_T and hence, by the theorem, Cov(-f(W_T), g(W_T))$\geq$0 and hence, Cov(f(W_T), g(W_T))$\leq$0 and hence, our goal of reducing variance is achieved if we generate new random variables, W_T only for half the number of data points we use in each simulation to calculate mean call option price and for the other half, using the negative of random numbers generated in the first half.

4.6 Algorithm to calculate Call option price using Black-Scholes formula

To begin with, take the current Stock price(S), Strike price(K), Risk-free interest rate(r),Time to maturity(T) and Price volatility(σ) as the inputs and then, calculate ln(S/K), (r+σ^2/2)T and σ-$\sqrt{T}$. Using these, calculate d_1 and d_2 in the Black-Scholes formula and then calculate the normal cdf of these d_1 and d_2 and substitute all these calculated values in the Black-Scholes formula to get the value of call option price.

4.7 Algorithm to calculate Call option price by direct Monte carlo method and comparing with the Call option price from BS formula

In this, we use the same value of variables as in the previous algorithm because our aim is the comparison between the two. For this, however, we input l,an even number = 2n for some n$\geq$1, the number of data points in each simulation to estimate mean call option price and m, the number of times simulation needs to be done or number of times the estimate for mean call option price has to be found. Then,for each j in [1,m], you calculate the risk-neutral version of $S_T(w_i^j)$ for each i in [1,l] by just substituting the input values and for $W_T(w_i^j)$, you generate the normal random number with mean 0 and variance T. Here,(w_i^j) is a point in the sample space. Then, you consider max(0,$S_T(w_i^j)$ - K) for each i in [1,l] and each j in [1,m]. Then, you obtain a mean of call option price using these l values of max(0,$S_T(w_i^j)$-K), i runs from 1 to l, multiplied with exp(-rT). You repeat this for m number of times,ie for each j, so as to get idea of distribution of the estimator(by direct monte carlo method) of discounted expectation. You achieve this by plotting a histogram of these m values.To compare Call option price obtained from BS formula and from the above simulations, you vertically plot the value obtained from BS formula on the histogram obtained.

4.8 Algorithm to calculate Call option price using Antithetic variates method

Algorithm by: Shrushti K Patel

In this, we follow the same steps as for direct Monte carlo method except that,for each j in [1,m] for $W_T(w_i^j)$, you generate the normal random number with mean 0 and variance T only for i in [1,n] and obtain $S_T(w_i^j)$ for i in [1,n] and for the remaining n data points,ie for each i in [n+1,2n], we obtain, $S_T^\theta(w_i^j)$,notation for Stock price at time T by taking the negative of the $W_T(w_i^j)$'s obtained for i in [1,n] as $W_T^\theta(w_i^j)$, for each i in [n+1,2n] without changing the order.That is if an k in [1,n] has $W_T(w_k^j)$ as a, then $W_T^\theta(w_i^j)$, for i=n+k, will be -a. Again like in the direct Monte carlo method, to get an idea of distribution of the estimator(by antithetic variates method) of discounted expectation, histogram is again plotted which is merged with the one plotted for direct Monte carlo method.

4.9 Results

As expected, current value of underlying asset, time to expiration on the option, riskless interest rate, variance in the value of underlying asset have positive impact on the value of call option and strike price has negative impact on the value of call option. To understand the distribution of estimator of discounted expectation, we plotted a histogram and we see an almost symmetric distribution(for both direct Monte carlo method and antithetic variates method) if the number of times this estimation is done is very large(500 is seen to be large enough). We see that the means of call option price obtained from both direct Monte carlo method and antithetic method are nearly same and are close to Call option price obtained from Black Scholes formula. However, we see estimator obtained from antithetic variates method has lower variance than that obtained from direct Monte carlo method . We can see the results for a particular set of input values here.

For the given inputs of S = 100 , K = 90 , $r = 0$, T = 0.25 , $\sigma = 0.1$, the Call option price from Black-Scholes formula is 10.0301. We observed that the mean of Call option price for the direct Monte Carlo method is 10.032(upto 3 decimal places) and estimate of standard deviation for this estimator is 0.227(upto 3 decimal places). However, using antithetic variates method, the mean of Call option price is 10.031(upto 3 decimal places) and estimate of standard deviation for this estimator is 0.022(upto 3 decimal places) which is around 10 times lower than direct Monte carlo method though the means are almost the same. The standardised histogram for this input parameters can be seen in the Figure 3 where it can be seen that the sampling distribution for the direct Monte Carlo method (blue) has a wider range of values than for Antithetic variates method(orange).Checking this for various inputs of parameters, it is always observed that standard deviation of estimator from Antithetic variates method is always lower than that from direct Monte Carlo simulation method and the means are almost equal always and close to value obtained from BS formula.

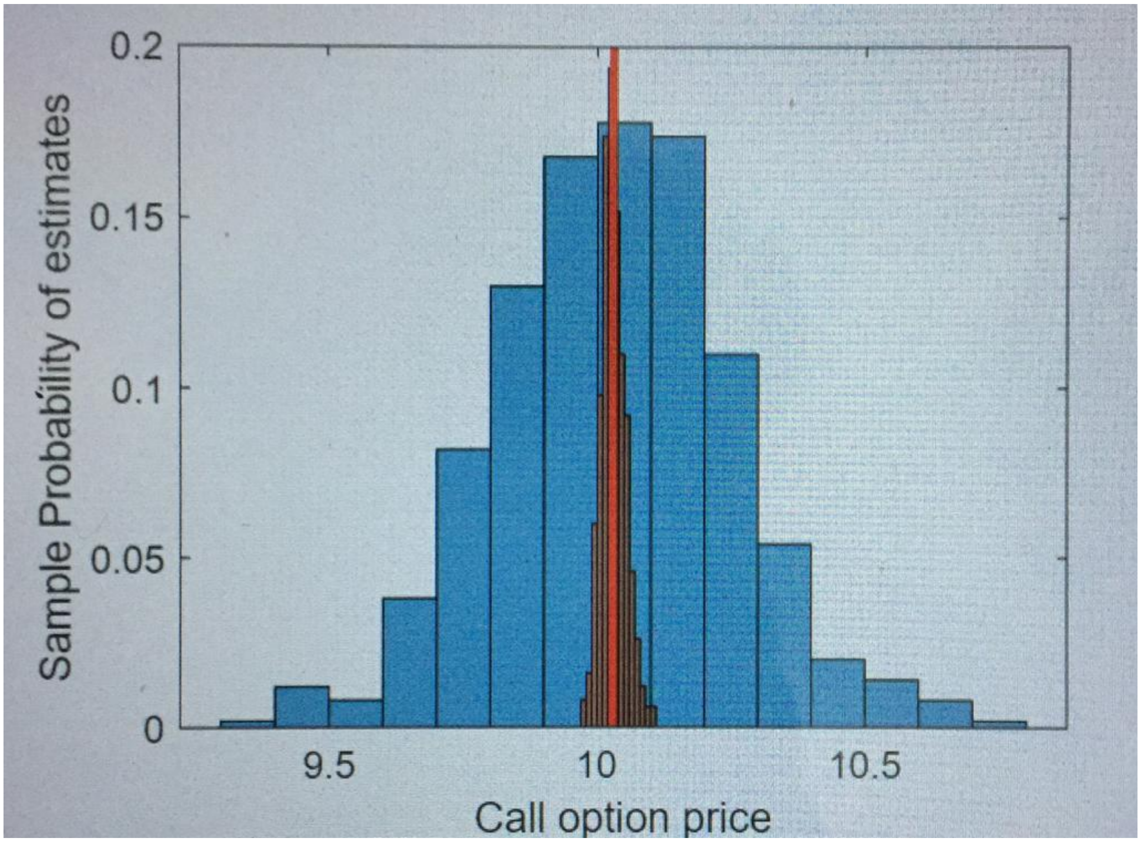


Figure 3: Comparison of Call option price from-Direct Monte Carlo (Blue),Antithetic Variates Method (Orange), BS formula(red)

5 Simulation of Markov Chains

Presented by: Shrushti K Patel

5.1 Introduction

Markov chains was developed by Russian-mathematician Andrei Markov in 1906. Markov-chains are important tool in stochastic-processes. A state-space in Markov-chain is a space of all possible outcomes in a system. Markov-chains are used in computing probabilities of events occurring as states transitioning into other states or into same state. All states are one of two types : (1)Absorbing & (2)Non-absorbing, if process remains in a state permanently it is said to be in Absorbing state and Non-absorbing otherwise.In Markov-process, the current state of a system depends only on its immediately preceding state.We will study two types of Markov-chains and simulate states of Markov-chain : (1) Discrete Time Markov-chain and (2) Continuous Time Markov-chain.There are many applications of Markov-chains.They are used in music recognition, speech recognition & mapping of animal life populations. The page-rank of web-page as used by Google is also defined through Markov-chain.In Finance & Economics, chains help in predicting market trends like bull-market,stagnant-market and bear-market. We can calculate option prices and credit risks using Markov-chains.

5.2 Discrete Time Markov Chain

Definition 5.1 *A stochastic process $X(t) : t \geq 0$ is called discrete-time Markov chain if for all time $n \geq 0$ and all states $i_0, i_1, i_2, i_3...., i_k, ..., i_n - 1, i_n, ..(i_k \epsilon S)$, where is S is state-space of Markov-chain.*

$$P(X_n+1=j|X_n=i, X_n-1=i_n-1, ..., X_0=i_0) = P(X_n+1=j|X_n=i)$$
$$= P_{ij}$$

where, P_{ij}= The probability that the chain in state i, moves next into state j and is called Transition-Probability.
The square matrix $P = P_{ij}$, where $i, j \epsilon S$ is called the Transition-Probability Matrix (TPM). Each row of Transition matrix must sum to 1.

$$\sum_{j\epsilon S} P_{ij} = 1$$

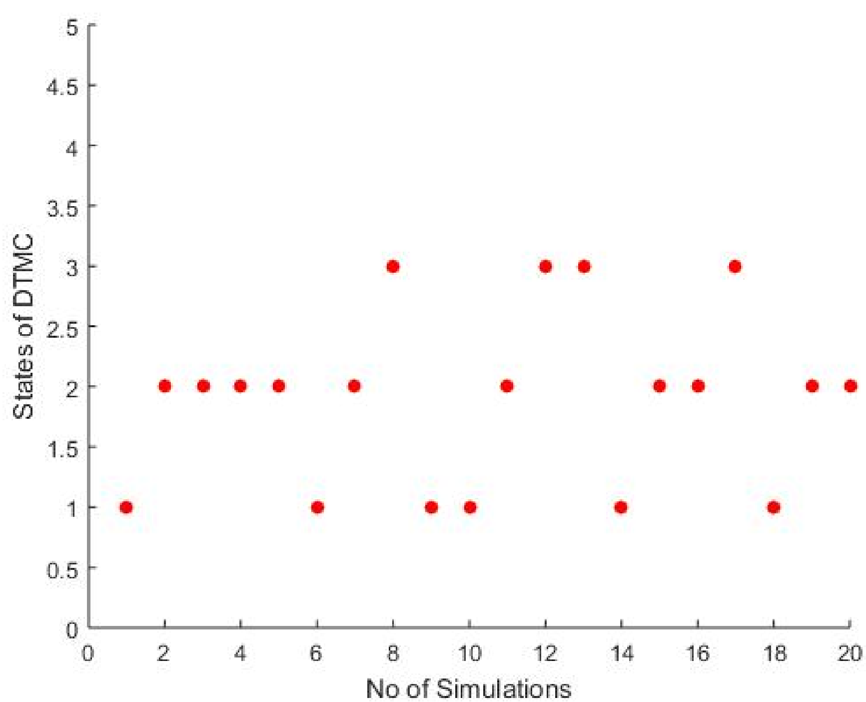
5.3 Simulation of Discrete-Time Markov Chain States

We will simulate Discrete-Time Markov-chain upto N times given a Transition Probability Matrix (TPM_{nxn}) , initial state (x_0) & state-space $S = \{1, 2, ..., k\}$. To simulate states using normal random numbers, we will use following function

$$F(r,i) = \begin{cases} 0 & , if\ r \leq P_{i0} \\ 1 & , if\ P_{i0} < r \leq P_{i0} + P_{i1} \\ 2 & , if\ P_{i0} + P_{i1} < r \leq P_{i0} + P_{i1} + P_{i2} \\ . \\ . \end{cases} \tag{2}$$

If generated normal random variable (r) is less than cumulative sum of state at particular state, then there will be no transition. If r is greater than cumulative sum of state at particular state then there will be transition to next state.
Algorithm:- To simulate states of Discrete Time Markov-Chain
Steps:-
(1). Input Transition probability matrix $(TPM)_{nxn}$, initial state : x_0 & n : no of times we want to simulate states.
(2). for $i = 1$ to n
$r \sim N(0, 1)$,Normal random number
$x_i = F(r, x_{i-1})$
end.

Figure 4: Simulation of states of Discrete-Time Markov Chain



5.4 Continuous-Time Markov-Chain

Continuous-time Markov chain is a stochastic process having the Markov property that the conditional distribution of the future state given present state & past state depends only on the present state and is independent of the past state.

Suppose, we have a continuous-time stochastic-process $\{X(t), t \geq 0\}$. Then, the process $\{X(t), t \geq 0\}$ is a continuous-time markov chain if for all $s, t \geq 0$ & non-negative integers $i, j, x(u), 0 \leq u < s$

$$P\{X(t+s) = j | X(s) = i, X(u) = x(u), 0 \leq u < s\} = P\{X(t+s) = j | X(s) = i)\}$$

$$where, \quad X(t+s): \; future \; state \;, \quad X(s): \; present \; state \;, \quad X(u): \; past \; state$$

If continuous-time markov chain is independent of present state then the markov-chain is said to have stationary or homogeneous transition probability. Here T_i, the amount of time that the process stays in state i before making a transition into a different state is random. Thus T_i, random variable is memory-less and must be exponentially distributed. We know in birth and death process the duration time between two consecutive events is random time (independent random variable) and cdf of random time depends on the rate parameters i.e, birth rate or death rate. Since, continuous-time markov chain can change it's state at any real time, birth or death rates are used in making transitions.

5.4.1 Properties of Continuous Time Markov Chain

(a) The amount of time it spends in particular state before making a transition into a different state is exponentially distributed with mean , say $\frac{1}{\lambda_i}$.
(b) When the process leaves state i, it next enters state j with some probability, say P_{ij},

$$P_{ii} = 0, \forall i$$

$$\sum_j P_{ij} = 1, \forall i$$

5.4.2 Simulation of states of Continuous Time Markov Chain

We will simulate states of Continuous-time Markov-chain given, Square-Matrix $TPM_{(nxn)}$, mean-rate parameter λ_n , initial state x_0 , T:Time-Horizon and δ_t: Time-increment.
Note: States of continuous-time Markov-chain is Non-absorbing state.
Algorithm:- To simulate states of Continuous Time Markov-Chain
Steps:-
(1). Input transition probability matrix $P_{(nxn)}$, T : Time-Horizon and δ_t time-increment.
(2). For each state i, input λ_i where, $\frac{1}{\lambda_i}$ is the mean holding time at state i

$$\lambda = (\lambda_1, \lambda_2, \lambda_3 \;, \; . \; . \; . \; . \; . \; . \; . \; , \lambda_k)$$

(3). Simulate states of Discrete-Time Markov chain using algorithm given earlier.

$$X = (X_1, X_2, X_3, \; . \; . \; . \; . \; . \; . \; , X_n)$$

(4). Now, simulate the holding time at state i

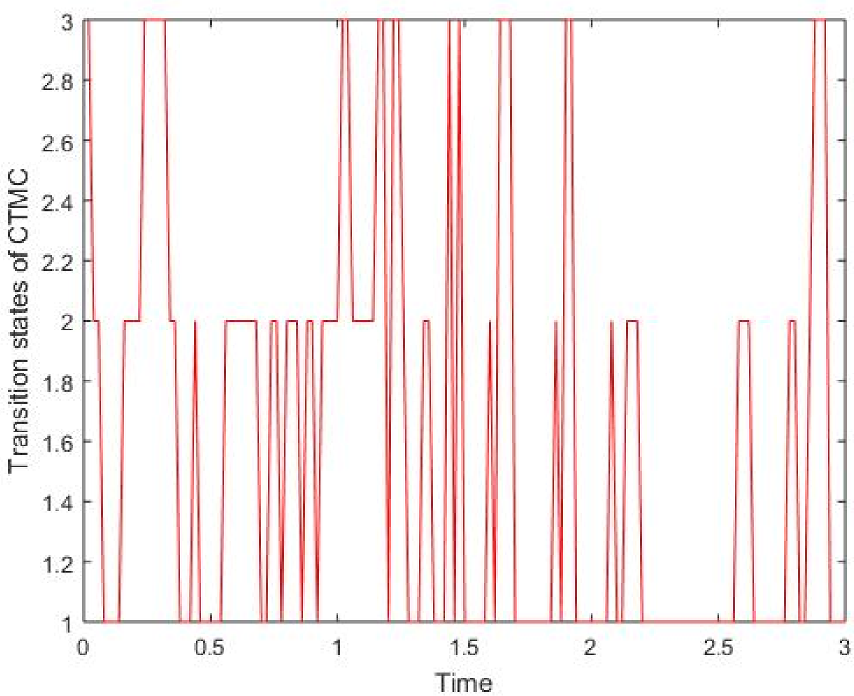
$$T_1 \sim exp\left(\frac{1}{\lambda_{X_1}}\right)$$
$$T_2 \sim exp\left(\frac{1}{\lambda_{X_2}}\right)$$
$$.$$
$$.$$
$$.$$
$$T_{n-1} \sim exp\left(\frac{1}{\lambda_{X_{n-1}}}\right)$$
$$T_n \sim exp\left(\frac{1}{\lambda_{X_n}}\right)$$

(5). To define the Continuous-time Markov chain states M with time granularity δ_t

$$
\begin{aligned}
&a = 0\\
&for \quad j = 1 \; to \; n-1\\
&\qquad b = a + T_j\\
&\qquad for \quad i = \lceil \frac{a}{\delta_t} \rceil + 1 \; to \quad \lceil \frac{b}{\delta_t} \rceil\\
&\qquad\qquad M_i = X_{T_{j-1}}^{-} = X_j\\
&\qquad end\\
&\qquad a = b
\end{aligned}
$$

(6). Plotting states of continuous-time markov chain v/s Time-Horizon, we get following graph.

Figure 5: Simulation of States of Continuous-Time Markov-chain



5.5 Markov-Modulated Geometric-Brownian Motion

Presented by: Shrushti K Patel

5.5.1 Introduction

Markov-Modulated Geometric-Brownian Motion model,also known as Regime-switching model is important model in econometrics and finance. Recent research in financial economics has documented the usefulness of regime-switching models in the study of various economic time-series.Here model assumes, the economic-state of the world is modelled by Markov-chain evolving continuous time. The valuation of European options within the framework of a market , as modelled by the interest rate and asset price's volatility processes switching between a finite number of states. Models with regime-switching for American options in the simple case of 2-state Markov chain are discussed in Elliot and Buffington while closed-form solutions for perpetual American put options in a regime-switching framework are presented in a more recent study of Guo and Zhang .

5.5.2 Model

Let $\mathcal{T}$ be the time-index set $\{0, 1, 2, 3, ...\}$ of the economy. $(\Omega, \mathcal{F}, \mathcal{P})$: complete probability space , where $\mathcal{P}$ is a real-world probability. We suppose that the uncertainties due to the fluctuations of market prices and the economic states are described by the probability space $(\Omega, \mathcal{F}, \mathcal{P})$

Let $\mathcal{X} = \{1, 2, 3, ...\}$ be the state-space of an irreducible Markov-chain $\{X_t, t \geq 0\}$ with transition rule

$$P(\ X_{t+\delta_t}|X_t = i)\} = \lambda_{ij}\delta_t + o(\delta_t), i \neq j$$

$$where \quad \lambda_{ij} \geq 0 \quad for \quad i \neq j \ \ \& \ \ \lambda_{ii} = \sum_{j \neq i}^{k} \lambda ij$$

$$p_{ij} = \frac{\lambda_{ij}}{|\lambda_{ij}|} \ are\ the\ transition\ probabilities\ from\ state$$

i to state j

By definition , Geometric Brownian Motion follows Stochastic Differential Equation (SDE),

$$dS_t = \mu S_t dt + \sigma S_t dB_t$$

$$where,\ B_t : Brownian\ Motion,\ \mu : Drift\ of\ risky\ asset,\ \sigma : Volatility\ of\ risky\ asset$$

The solution of this SDE and the stock-price is ,

$$S_t = S_0 e^{(\mu - \sigma^2/2)t + \sigma B_t)}$$

But , Markov Modulated Geometric Brownian Motion doesn't assume μ & σ as constants. They are assumed to be function of Continuous-Time Markov-chain & so the SDE of the Markov Modulated Geometric Brownian Motion becomes,

$$dS_t = \mu(X_t)S_t dt + \sigma(X_t)S_t dB_t$$

Thus,
Now we will describe the Markov Modulated process for the price dynamics of the underlying risky asset.

We assume that the interest rate of the bank account, the drift , and the volatility of the risky-asset switch over time according to the states of the economy modeled by X.

Let r_t be the market interest rate of the bank account in the t^{th} period

$$r_t = r(X_t)$$

the price dynamic $B = \{B_t\}_{t \epsilon \mathcal{T}}$ of the bank-account is given by

$$B_t = B_{t-1}e^{r_t}\ , B_0 = 1$$

Let $S = \{S_t\}_{t\,\epsilon\,\mathcal{T}}$ be the price process of the risky-stock. For each $t\ \epsilon\ \mathcal{T}$,be the logarithmic return in the t^{th} period. μ_t & σ_t be the drift & the volatility respectively. of the risky asset in the t^{th} period.

$$\mu_t = \mu(X_t) \quad \& \quad \sigma_t = \sigma(X_t)$$

Thus, solution (a) is given by

$$S_{t_{i+1}} = S_{t_i} e^{((\mu_i - (\frac{\sigma_i^2}{2})dt) + \sigma_i (B_{t_i+dt} - B_{t_i}))}$$

where,

$$S_{t_0} > 0 \, , dt = t_{i+1} - t_i \;: Time - increment$$

$$\mu_i = \mu(X_{t_i})$$

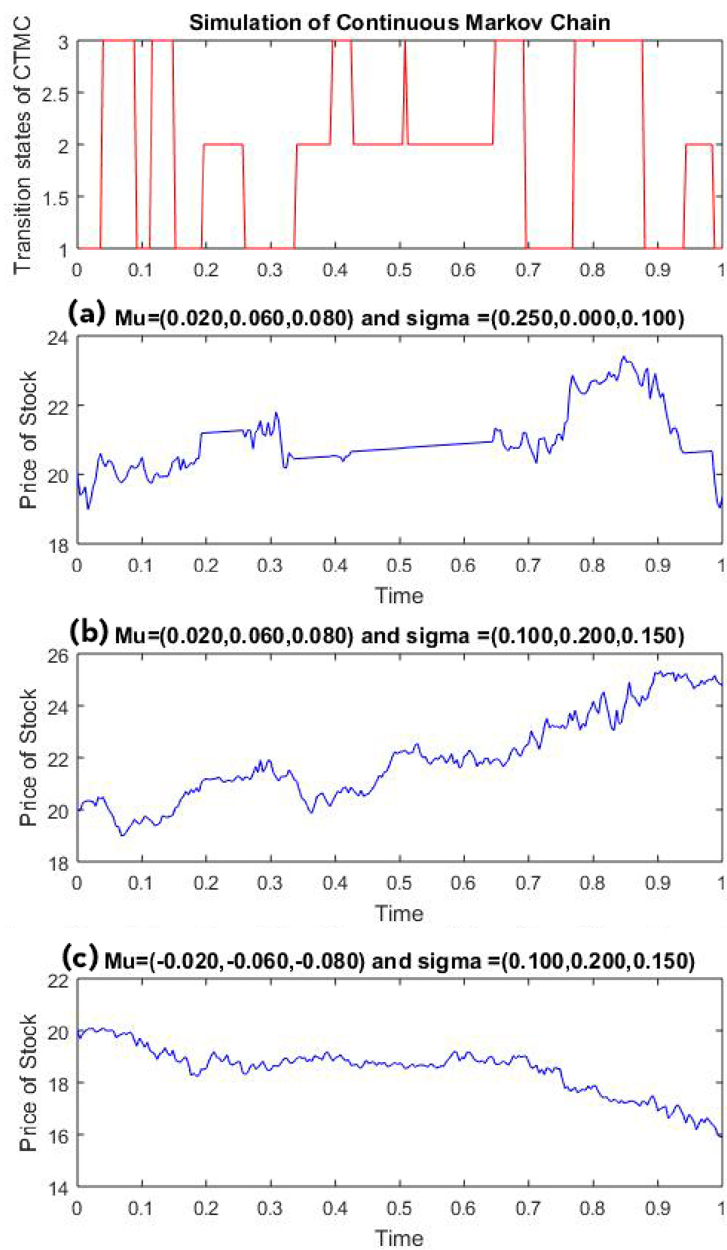
$$\sigma_i = \sigma(X_{t_i})$$

$\{S_{t_i}\}_{i=1}^{N}$ is the discrete simulation of stochastic process $\{S_t\}_{t\geq 0}$

5.5.3 Simulation of Markov-Modulated Geometric Brownian Motion

We will use eqn (b) to simulate Markov-Modulated Geometric Brownian Motion model , given S_0:initial Stock-price,dt:time-increment,T: Time to maturity, μ :Drift & σ :Volatility.

Figure 6: Stochastic process using Markov-Modulated GBM



RESULTS : For the given inputs in Markov-Modulated Geometric-Brownian motion ,initial stock price $S0 = 20$, $Time = 1$, $\delta t = 0.004 = \frac{1}{252}$(Stock-market working days in a year), Transition probability matrix= $\begin{pmatrix} 0 & 0.8 & 0.2 \\ 0.6 & 0 & 0.4 \\ 0.7 & 0.3 & 0 \end{pmatrix}$ and $\lambda = (12, 15, 20)$.Here we simulate states of continuous Markov-chain and stochastic process of Markov-Modulated Geometric-Brownian motion. We observe that according to states of continuous Markov-chain, volatility(σ) and drift(μ) as they are functions of continuous Markov-chain. To understand, behaviour of σ in Markov-Modulated Geometric-Brownian motion we take $\sigma = (0.25, 0, 0.10)$ in figure(a).When model is in state 1 , $\sigma(1) = 0.25$ we notice high-fluctuations in stock-price,in state 2, $\sigma(2) = 0$ stock-price remains constant & in state, $\sigma(3) = 0.10$ stock-price has negligible fluctuation.Thus, when volatility is highly positive or highly negative, there is a high sensitivity to changes in volatility. If sigma-value is close to zero or zero, volatility have very little effect on value of position of stock. (NOTE: In real-life,value of volatility in stock-market is never zero.) Now, we observe behaviour of drift. In figure(b)$\mu = (0.02, 0.06, 0.08)$,slope of stock price graph is increasing & in figure(c)$\mu = (-0.02, -0.06, -0.08)$,slope of stock-price graph is decreasing. Thus,if value of drift is negative then stock price graph has negative slope. Therefore, if value of drift is positive then stock-price are likely to increase else if negative then stock-price will likely decrease. In figure(b), Stock-price graph captures model's behaviour similar to Real-life stock-market.

6 Antithetic and Monte Carlo simulation of Markov Modulated Brownian Motion

Presented by: D.V.S. Abhijit

In this section we intend to obtain the price of a call option of a stock using the model of Markov Modulated Geometric Brownian Motion discussed in the earlier section.
The option price is obtained in two ways namely, the antithetic method and the Monte Carlo method similar to the methods used in the case of Geometric Brownian Motion.

Definiton 5.1 The stock price in a Markov Modulated Brownian motion is given as follows:

Let $\{X_t\}_{t\geq 0}$be a Markov process independent of the • stock price.
Let the initial stock price at t=0,be $S_0 = s$.

Let $\mu(X_t), \sigma^2(X_t)$ be the Markov dependent drift and volatility of the stock repectively.

$\{W_t\}_{t\geq 0}$ is a Brownian Motion.
Let r be the risk free interest rate that does not vary with the states of the Markov chain.

Then the stock price at time t=T ,time to maturity , is given as, $S_T = S_0 exp\{[\int_0^T \mu(X_t) - \sigma^2(X_t)/2)]dt + \int_0^T \sigma(X_t)dW_t\}$ (5.1)

Let $B_t = exp(rt)$ for $t \geq 0$

Let K be the strike price of the stock.

To obtain the discounted call option price at t=0 we use the following formula:

C(0,s)= exp(-rT)E[$(S_T - K)^+ | S_0 = s$]

6.1 Comparison of Monte Carlo method and Antithetic Variates method to calculate Call option from MMGBM

Algorithm to calculate call option price using Monte Carlo Method

We initially select 'Ft' the time horizon/maturity date of the option, 'dt', the time step required, 'l'(An even number), the number of data points required to calculate the estimator of the expectation and 'n', the number of estimators required.

Generating a Markov Chain A Markov chain $X_t$is generated from 0 to Ft in steps of dt using the algorithm mentioned before.

Generating the Brownian motion We generate the brownian motion W_t from 0 to T using a time step of dt using the algorith

Let m=Ft/dt

The difference between both the methods is as follows.

Let 'j' be a number between 1 to n and 'i' be a number between 1 to l, then $S_T(\omega_i{}^j)$ be the ith data point in obtaining the jth estima

$\omega_i{}^j$ is a point in the sample space In the anithetic method,Step 1 :for each 'i',for 'j' from 1 to l/2 we obtain the brownian motionW_i

Step 2 :for each 'i', for 'j' from (l/2)+1 to l the standard normal random numbers for brownian motion to obtain${S_T}^\theta(\omega_i{}^j)$are the ne

In the Monte Carlo method Step 2 of the antithetic method is not considered and normal random variables are obtained independe

We obtain $C_t, D_t$where,

$C_t = \mu(X_t) - \sigma^2(X_t)/2$

$D_t = \sigma(X_t)(W_t - W_{t-1})$

$W_0 = 0$

$t \in [1, m]$

We perform trapezoidal integration on $C_t, D_t,$ from 0 to T and put them in formula (5.1) to obtain S_T.

We obtain 'l' such S_T and calculate $\max(S_T - K, 0)$ for all of the 'l' values and obtain an estimate of the expectation and multiply l

$C^\theta = 1/n(\sum_{n=1}^{l/2}(S_T(\omega_i{}^j) - K)^+ + \sum_{n=l/2}^{l}(S_T{}^\theta(\omega_i{}^j) - K)^+)$.

The estimator in the Monte Carlo method is obtained as: $C^m = 1/n(\sum_{n=1}^{l}(S_T(\omega_i{}^j) - K)^+)$

The sampling distribution of, C^θ, C^m

Results

The parameters considered in the simulation are:

1. A Markov chain with three states 1,2 and 3 and corresponding transition parameters

2. Time horizon,T=0.25,time step,dt=0.04

3. Initial stock price,S=100,Interest rate,r=0

4. Markov chain dependent drift and volatility are considered to be same for both Antithetic and Monte Carlo Methods

   It is observed that the mean of the call option price for antithetic method and Monte Carlo method are same upto four decimal places at 10.8879. But the estimate of standard deviation of the estimators in antithetic method is 0.0617 while that of the Monte Carlo method is 0.3119, which shows that the standard deviation estimate in the antithetic method is 0.197 times of that in the Monte Carlo method while the means are the same upto four decimal places. The sampling distribution for the Monte Carlo method(Red) has a wider range of values than the antithetic method(Blue) as observed from the standardized histogram (Figure(5)).From these observations we can imply that using the antithetic method reduces variance of the estimate without affecting the expectation of the estimator.

   Figure(5):Antithetic method(Blue) vs Monte Carlo method(Red).

   X axis is the call option price. Y axis represents the sample probabilities

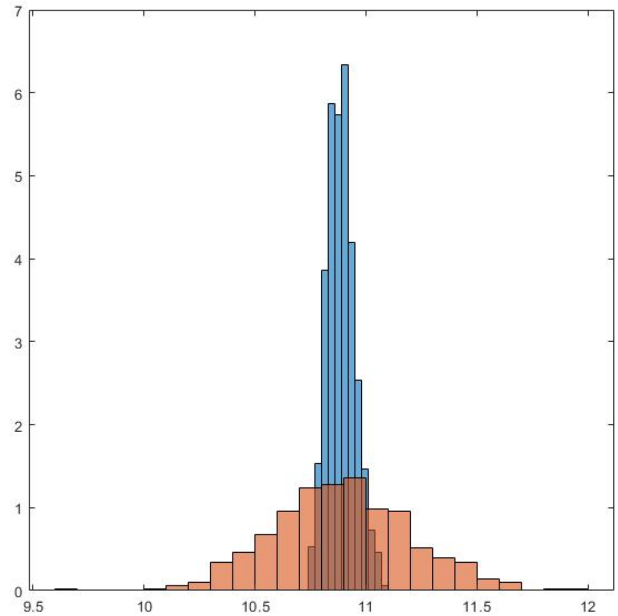


Figure 7: Comaprison of Monte Carlo and Antithetic Methods

7 Call Option Price for Asian Options using Geometric Brownian Motion

Presented by: Malavika Biju

7.1 Introduction

Asian options are path dependent derivatives whose payoffs depend on the average of the underlying asset prices during the option life. Asian Options are different from the case of the usual European options and American options (collectively known as vanilla style options) , where the payoff of the option contract depends on the price of the underlying instrument at exercise. Asian options are thus one of the basic forms of exotic options.

7.2 Types of Asian Option

There are two types of Asian options:

1.Fixed strike : In this style of Asian option, the average of the asset prices is used in place of the price of the underlying security and the strike price is fixed. Fixed strike Asian call and put option payoff is given by

$C(T) = max(A(0,T) - K, 0)$ and $P(T) = max(K - A(0,T), 0)$ respectively.

where A(0,T) denotes the average price for the period [0, T], and K is the strike price.

2. Floating strike : In this style of Asian option, the average of the asset prices is used in place of the strike price and

the value of the underlying security is it's price at the time of maturity.Floating strike Asian call and put option payoff is given by

$C(T) = max(S(T) - A(0,T), 0)$ and $P(T) = max(A(0,T) - S(T), 0)$ respectively .

where S(T) is the price at maturity.

7.3 Types of Averaging used for Asian Options

1. Continuous arithmetic average

$$A(0,T) = \frac{1}{T}\int_0^T S(t)dt$$

2. Discrete arithmetic average Asian call or put with

$$A(0,T) = \frac{1}{m+1}\sum_{n=0}^{m} S(iT/m)$$

3. Continuous geometric average Asian call or put with

$$A(0,T) = exp(\frac{1}{T}\int_0^T log(S(t))dt)$$

4. Discrete geometric average Asian call or put with

$$A(0,T) = exp(\frac{1}{m+1}\sum_{n=0}^{m} log(S(iT/m)))$$

7.4 Pricing of Asian Options

The pricing problem for geometric Asian options can be solved using various approaches to obtain a closed form solution. Assuming that the price of the asset follows a Geometric Brownian Motion, it has a log-normal distribution continuous in time, and the product of log-normally distributed random variables is also log-normally distributed, however the sum is not. Hence, we could expect that the pricing of geometric average Asian options should be easy to deal with, while for arithmetic average ones it may be relatively more complicated to handle. In fact, the pricing formula of geometric average Asian options can be derived in the Black-Scholes framework. But to find the value of arithmetic Asian options we are obliged to use numerical computations.Since we know that the geometric average Asian options can be simply priced according to the Black–Scholes framework whereas the arithmetic ones cannot, we focus on the case of arithmetic average Asian options and on the pricing of this kind by using some other kind of pricing methods.

7.5 Advantages of Asian Options

1.A bond issuer might prefer Asian options to standard European options for several reasons. The most significant reason is probably the protection against price manipulation which an AV-option affords. This is of special importance where thinly-traded assets, like crude oil are concerned, since they are vulnerable to such manipulation when they are traded as standard European options.

2.The Asian options also enhances the ability of bondholders to share in the profits of a firm when those profits depend on the price level of an underlying asset. If, for example, a standard European call option is based on an asset which remains low in price during a large part of the final time period and rises significantly at maturity, the firm would not have been able to generate sufficient revenues to pay the high premium to the option holders.

3.Another advantage of Asian options involves the relative cost of Asian options compared to European or American options. Because of the averaging feature, Asian options reduce the volatility inherent in the option, therefore, Asian options are typically cheaper than European or American options. This can be especially of advantage for corporations that are subject laws which require that they expense employee stock options.

7.6 Algorithm

Our aim is to simulate stock prices over a fixed period of time using GBM and calculate the call option price for Arithmetic Asian options with fixed strike price using both Monte Carlo and Antithetic methods.Now that we have reached the conclusion that no explicit formula for the value of an Arithmetic Asian option can be found, we need to express the time variable t in discrete periods. We may thus approximate A(0,T) in the equation for continuous arithmetic Asian call option price as follows:

$$A(0,T) = \frac{1}{m+1}\sum_{n=0}^{m} S(T_i)$$

where $whereT_i = \frac{iT}{m}$. If m is large enough this formula is a satisfactory approximation of our original equation. We can therefore substitute the expression for A(0,T) in the equation for call option price and thus find the following numerical approximation of the value of an Asian option at time T:

$$C(T) = exp(-rT)E(max(\sum_{n=0}^{m} \frac{S(T_i)}{m+1} - K, 0))$$

To implement this, we input the number of data points in each simulation,l, and the number of times the estimate has to be calculated,n . Then we simulate stock prices as in GBM model for each of the methods. Then we calculate the average of the stock prices over the period of time inputted by the user. We then calculate max(0,avg (stock prices) - K) . Then we obtain an estimate of the call option price using these l values of max(0, avg (stock prices) - K) multiplied by exp(-rt) . We repeat this for n number of times so as to get a distribution of the estimator. We plot two histograms, one for each estimator (Monte Carlo and Antithetic) to get an idea of each of these probability distributions.

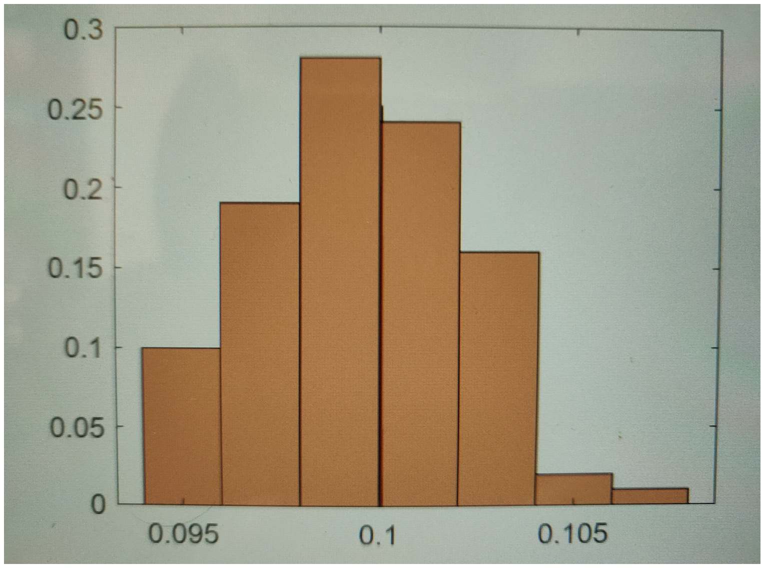


Figure 8: Asian Call Option Price using Monte Carlo and Antithetic methods

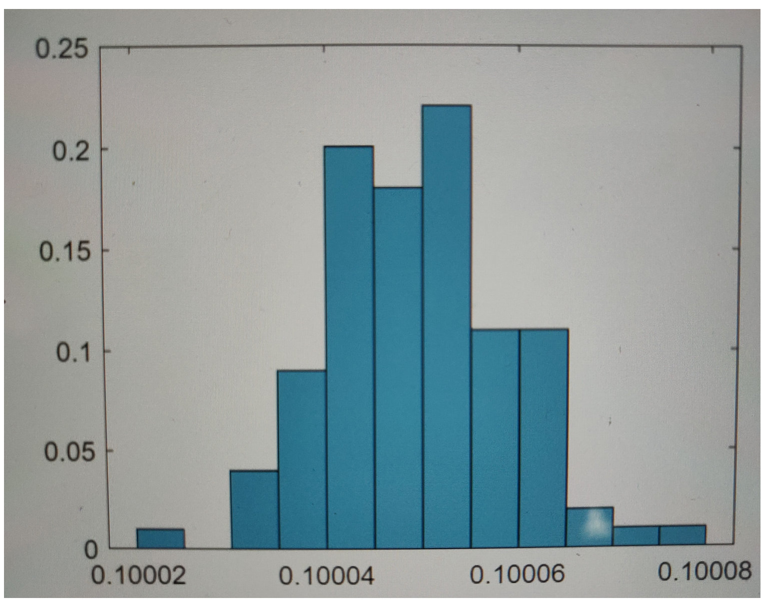


Figure 9: Asian Call Option Price using Antithetic method

7.7 Results

1. The following are the inputs to the program: Time horizon (in years) = 0.25 ; Time increment = 0.001 ; l = 50 ; n = 100 ; Initial Stock price = 100 ; Fixed strike price = 90 ; Interest rate = 0 ; mu = 1 ; sigma = 0.1.

2. It was observed that for Monte Carlo method, the mean Call Option Price is 0.099696 and Antithetic method ,the mean Call Option Price obtained is 0.100050. But the standard error for Antithetic method is 9E-6 whereas for Monte Carlo method, it is 0.001883. From these observations , we can conclude that , although both the methods give very similar values for the call option price, antithetic method shows significantly less variance than Monte Carlo method.

3. Figure 8 shows the normalized histograms for both Monte Carlo and Antithetic methods. The histogram for Monte Carlo is given in orange colour in Figure 8. Since Monte Carlo method simulates a much wider range of values in comparison to Antithetic method, the histogram for Antithetic method is shown approximately in the form of a straight line in Figure 8. In figure 8, the X axis gives the Call option price and the Y axis represents the sample probabilities.

4. In order to get an idea of the probability distribution of the Antithetic estimator, a separate normalized histogram is plotted. This figure is shown by figure 9. As in the previous figure, the X axis represents the call option price and the Y axis represents the sample probabilities.

8 Appendix

1. Brownian Motion Code:

```
clc
clear all
n=input('\n Enter required maxmimum time for process :\n');
h=input('\n Enter the timestep required:\n');
m=floor(n/h);
b=zeros(1,m+1);
t=0:h:m*h
for i=1:m
b(1)=0;
r(i+1)=normrnd(0,h);
b(i+1)=b(i)+r(i+1);
fprintf('\n Random normal number is %f and B(%d) is %f\n',r(i),i,b(i));
end
plot(t,b,'r')
xlabel('Time')
ylabel('B(t)')
title('Approximation of Brownian Motion')
```

2. Geometric Brownian Motion Code

```
clear all ;
n=input ( '\n Enter required maxmimum time for process(in years): \n ' ) ;
h=input ( '\n Enter the time step required : \n ' );
mu=input ( '\n Enter the value of drift : \n ' ) ;
sigma=input ( '\n Enter the volatility : \n ' ) ;
m=floor ( n/h ) ;
S=zeros ( 1 ,m+1);
t =0:h :m*h ;
S(1)=1 ;
for i =1:m
    r(i +1)=normrnd(0,sqrt(h));
    S(i +1)=S(i)*(1+ mu*h+ sigma*r(i +1));
    fprintf('\n Random normal number is %f and S(%d) is %f \n', r(i), i, S(i));
end
plot(t,S,'r');
xlabel('Time')
ylabel('S(t)')
```

3. Black-Scholes Formula and Call Option price using Geometric Brownian Motion

```
    "Black Schloes formula for Call option price"
S=input('\n Enter the current Stock price :\n');
K=input('\n Enter the Strike price :\n');
r=input('\n Enter the Risk-free interest rate :\n');
T=input('\n Enter the Time to maturity :\n');
v=input('\n Enter the Price volatility :\n');
x=log(S/K);
y=(r+(v*v/2))*T;
z=v*(sqrt(T));
d1=(x+y)/z;
d2=d1-z;
F1=normcdf(d1);
F2=normcdf(d2);
Call_OP=(S*F1)-(K*F2*exp(-r*T));
```

```
"Simulations"
"From direct Monte Carlo method"
l=input('\n Enter the required no. of data points for estimating mean COP(even number):\n');
m=input('\n Enter the no. of times estimation of mean COP to be done :\n');
ST_MC=zeros(1,l);
COP_MC=zeros(1,l);
Est_COP_MC=zeros(1,m);
for i=1:m
    for j=1:l
        ST_MC(j)=S*(exp(((r-((v^2)/2))*T)+v*normrnd(0,sqrt(T))));
        COP_MC(j)=max(0,ST_MC(j)-K);
    end
    Est_COP_MC(i) = (exp(-r*T))*mean(COP_MC);
end
"From Antithetic Variates method"
ST_AV=zeros(1,l);
COP_AV=zeros(1,l);
Est_COP_AV=zeros(1,m);
Randnos_AV=zeros(1,l/2);
for i=1:m
    for j=1:l
        if j<=l/2
            Randnos_AV(j)=normrnd(0,sqrt(T));
            ST_AV(j)=S*(exp(((r-((v^2)/2))*T)+v*Randnos_AV(j)));
        else
            ST_AV(j)=S*(exp(((r-((v^2)/2))*T)-v*Randnos_AV(j-(l/2))));
        end
        COP_AV(j)=max(0,ST_AV(j)-K);
    end
    Est_COP_AV(i) = (exp(-r*T))*mean(COP_AV);
end
fprintf('\n Call Option price(up to 4 decimal places) obtained from Black-Schloes formula is %f\n',round(Call_OP,4));
EST_OF_EST_MC=mean(Est_COP_MC);
EST_OF_EST_AV=mean(Est_COP_AV);
```

```
error_MC=std(Est_COP_MC);
error_AV=std(Est_COP_AV);
fprintf('\n Call Option price(upto 4 decimal places) obtained from Black-Schloes formula is %f\n',round(Call_OP,4));
fprintf('\n Call Option price(up to 3 decimal places) obtained from est. mean of estimates using direct Monte Carlo method is %f\n',round(EST_OF_EST_MC,3));
fprintf('\n Call Option price(up to 3 decimal places) obtained from est. mean of estimates using Antithetic variates method is %f\n',round(EST_OF_EST_AV,3));
fprintf('\n Standard deviation of estimator using direct Monte Carlo method is %f\n', round(error_MC,3));
fprintf('\n Standard deviation of estimator using Antithetic variates method is %f\n', round(error_AV,3));
"Histogram"
histogram(Est_COP_MC,'Normalization','probability');
xlabel('Call option price')
ylabel('Sample Probability of estimates )
xlim([9 11]);
hold on;
histogram(Est_COP_AV,'Normalization','probability');
hold on;
line([Call_OP,Call_OP ], ylim, 'LineWidth', 2, 'Color', 'r');
```

4. Simulation of Discrete-Time Markov Chain States

```
clc
clear all
TPM=[0 0.8 0.2;0.6 0 0.4;0.7 0.3 0];%input('\nTransition square matrix')
initialstate=1;
simulength =input('\nEnter how many times you wish to simulate:');
X= zeros(1,simulength);
X(1)=initialstate;
    for i=2:simulength
        stepdist =TPM(X(i-1),:);
        cumdist = cumsum(stepdist);
        r = rand();
        X(i)=find(cumdist>r,1);
```

```
    end
time=1:1:simulength;
plot(time,X)
```

5. Simulating Continuous-Time Markov Chain using given transition-probability matrix and Rate parameter Lambda(L)

```
clc
clear all
TPM=[0 0.8 0.2;0.6 0 0.4;0.7 0.3 0];%input('\nTransition square matrix');
CPM=transpose(cumsum(transpose(TPM)));
L=[20 30 40];%input('\nEnter n value of Lamda :')
initialstate=1;
Ft=input('\nEnter Time Horizon\n');
dt=input('\nEnter time increment: ');
simulength=floor(Ft/dt);
X= zeros(1,simulength);
X(1)=initialstate;
    for j=2:simulength
        cumdist=CPM(X(j),:);
        r = rand();
        X(j+1)=find(cumdist>r,1);
    end
  %For Fix Time
  a=0;
  m=ceil(Ft/dt);
  T=zeros(1,simulength);
    for j=2:simulength
        T(j)=exprnd(1/L(X(j)));
        b=a+T(j);
        for i=ceil(a/dt)+1:min(ceil(b/dt),m)
            M(i)=X(j);
        end
            a=b;
    end
  time=dt:dt:Ft;
  plot(time,M,'r')
  xlabel('Time')
```

```
    ylabel('Transition states of CTMC')
```

6. Simulation of Markov-Modulated Geometric Brownian Motion

```
clc
clear all
TPM=[0 0.8 0.2;0.6 0 0.4;0.7 0.3 0];
CPM=transpose(cumsum(transpose(TPM)));
L=[12 15 20];%input('\nEnter  n value of Lamda :')
Ft=1;%input('\nEnter Time Horizon\n');
dt=0.004;%input('\nEnter time increment:  ');
X(1)=1;
a=0;
m=ceil(Ft/dt);
j=0;
while ceil(a/dt)<m
     j=j+1;
     T(j)=exprnd(1/L(X(j)));
     b=a+T(j);
     for i=ceil(a/dt)+1:min(ceil(b/dt),m)
          M(i)=X(j);
     end
     if ceil(b/dt)>m
          break
     else
          a=b;
          cumdist=CPM(X(j),:);
          r = rand();
          X(j+1)=find(cumdist>r,1);
     end
end
time=dt:dt:Ft;
subplot(2,1,1)
plot(time,M,'r')
%xlabel('Time')
ylabel('Transition states of CTMC')
title('Simulation of Continuous Markov Chain')
```

```
%pricing of markov modulated gbm
sigma=[0.1 0.20 0.15];
Mu=[0.02 0.06 0.08];
S=zeros(1,m);
B=zeros(1,m);
S(1)=20;%Initial Stock Price
for i=1:m
    B(1)=0;
    B(i+1)=B(i)+sqrt(dt)*normrnd(0,1);
    S(i+1)=S(i)*exp((Mu(M(i))-0.5*(sigma(M(i))^2))*dt+sigma(M(i))*(B(i+1)-B(i)));
end
time=0:dt:Ft;
subplot(2,1,2)
plot(time,S,'b')
xlabel('Time')
ylabel('Price of Stock')
str=sprintf('Stochastic process using Regime-Switching Geometric Brownian Motion with Mu=(%5.3f,%5.3f,%5.3f) and sigma =(%5.3f,%5.3f,%5.3f)',Mu(1),Mu(2),Mu(3),sigma(1),sigma(2),sigma(3));
title(str)
```

7. Antithetic and Monte Carlo simulation of Markov Modulated Brownian Motion

```
clc
clear all
TPM=[0 0.8 0.2;0.6 0 0.4;0.7 0.3 0];
CPM=transpose(cumsum(transpose(TPM)));
L=[12 15 20];%input('\nEnter  n value of Lambda :')
Ft=input('\nEnter Time Horizon (Even Number)\n');
dt=input('\nEnter time increment: ');
l=input('\n Enter the required no. of data points for estimating mean COP :\n');
n=input('n\ Enter no. of estimators required\n');
X(1)=1;
a=0;
m=ceil(Ft/dt);
j=0;
```

```
while ceil(a/dt)<m
    j=j+1;
    T(j)=exprnd(1/L(X(j)));
    b=a+T(j);
    for i=ceil(a/dt)+1:min(ceil(b/dt),m)
        M(i)=X(j);
    end
    if ceil(b/dt)>m
        break
    else
        a=b;
        cumdist=CPM(X(j),:);
        r = rand();
        X(j+1)=find(cumdist>r,1);
    end
end
% time=dt:dt:Ft;
% subplot(2,1,1)
% plot(time,M)
% xlabel('Time')
% ylabel('Transition states of CTMC')
% title('Simulation of Continuous Markov Chain')

%Antithetic pricing of markov modulated gbm
sigma=[0.05 0.10 0.20];
Mu=[0.02 0.08 0.04];
S_a=zeros(1,m);
B_a=zeros(1,m);
K=90;
r=0;
for k=1:n
    for i=1:l
        S_a(i,1,k)=100;
        S_m(i,1,k)=100;
    end
end    %Initial Stock Price
```

```
%Generating Antithetic Random Nos.

for k=1:n
    for i=1:l
        for j=1:m
            if i<=l/2
                Randnos_AV(i,j,k)=normrnd(0,1);
            else
                Randnos_AV(i,j,k)=(-(Randnos_AV(i-(l/2),j,k)));

            end
            B_a(i,1,k)=0;
            B_a(i,j+1,k)=B_a(i,j,k)+sqrt(dt)*Randnos_AV(i,j,k);
            A_a(i,j,k)=B_a(i,j+1,k)-B_a(i,j,k);

            C_a(i,j,k)=(Mu(M(j))-0.5*(sigma(M(j))^2))*dt;
            D_a(i,j,k)=(sigma(M(j))*A_a(i,j,k));
            C_a_int(k,i) = trapz(C_a(i,:,k));
            D_a_int(k,i)=trapz(D_a(i,:,k));
        end
          C_a_int(k,i) = trapz(C_a(i,:,k));
          D_a_int(k,i)=trapz(D_a(i,:,k));
          ST_a(k,i)=S_a(i,1,k)*exp(C_a_int(k,i)+D_a_int(k,i));
          Mk_a(k,i)=max(0,ST_a(k,i)-K);
    end
end



for k=1:n
  F(k)=(exp(-r*Ft))*mean(Mk_a(k,:));

end
Est_Est_a=mean(F);
```

```
%Generating Monte carlo Random Nos.

for k=1:n
    for i=1:l
        for j=1:m

                Randnos_MC(i,j,k)=normrnd(0,1);




            B_m(i,1,k)=0;
            B_m(i,j+1,k)=B_m(i,j,k)+sqrt(dt)*Randnos_MC(i,j,k);
            A_m(i,j,k)=B_m(i,j+1,k)-B_m(i,j,k);

            C_m(i,j,k)=(Mu(M(j))-0.5*(sigma(M(j))^2))*dt;
            D_m(i,j,k)=(sigma(M(j))*A_m(i,j,k));

        end
        C_m_int(k,i) = trapz(C_m(i,:,k));
        D_m_int(k,i)=trapz(D_m(i,:,k));
        ST_m(k,i)=S_m(i,1,k)*exp(C_m_int(k,i)+D_m_int(k,i));
        Mk_m(k,i)=max(0,ST_m(k,i)-K);

    end
end

for k=1:n
    for i=1:l

      ST_m(k,i)=S_m(i,1,k)*exp(C_m_int(k,i)+D_m_int(k,i));
    end
end

for k=1:n
```

```
    G(k)=(exp(-r*Ft))*mean(Mk_m(k,:))  ;

end

Est_Est_m=mean(F);

error_F=std(F);
error_G=std(G);
histogram(F,'Normalization','pdf')
hold on
histogram(G,'Normalization','pdf');
```

8. Antithetic and Monte Carlo simulation of Asian Option Using GBM

```
%pricing of Asian options

clc
clear
Ft=input('\nEnter Time Horizon (Even Number)\n');
dt=input('\nEnter time increment: ');
l=input('\n Enter the required no. of data points for estimating mean COP(Even number)
    :\n');
n=input('\n Enter no. of estimators required\n');
m=ceil(Ft/dt);
sigma=0.1;
Mu=1;
B_a=zeros(1,m);
K=100;
r=0;

%Generating Antithetic Random Nos.

for k=1:n
    for i=1:l
        for j=1:m
            if i<=l/2
```

```
                Randnos_AV(i,j,k)=normrnd(0,1);
            else
                Randnos_AV(i,j,k)=(-(Randnos_AV(i-(l/2),j,k)));

            end
            B_a(i,1,k)=0;
            B_a(i,j+1,k)=B_a(i,j,k)+sqrt(dt)*Randnos_AV(i,j,k);
            A_a(i,j,k)=B_a(i,j+1,k)-B_a(i,j,k);


            C_a(i,j,k)=(Mu-0.5*(sigma^2))*dt;
            D_a(i,j,k)=(sigma*A_a(i,j,k));
            ST_a(i,j,k)=100*exp(C_a(i,j,k)+D_a(i,j,k));
        end
        ST_avg(k,i)= mean(ST_a(i,:,k));
        Mk_a(k,i)=max(0,ST_avg(k,i)-K);
    end


end



for k=1:n
    F(k)=(exp(-r*Ft))*mean(Mk_a(k,:));
end

Est_Est_a=mean(F);

%Generating Monte carlo Random Nos.

for k=1:n
    for i=1:l
        for j=1:m

            Randnos_MC(i,j,k)=normrnd(0,1); B_m(i,1,k)=0;
            B_m(i,j+1,k)=B_m(i,j,k)+sqrt(dt)*Randnos_MC(i,j,k); A_m(i,j,k)=B_m(i,j+1,k
```

```
)-B_m(i,j,k);

            C_m(i,j,k)=(Mu-0.5*(sigma^2))*dt;
            D_m(i,j,k)=(sigma*A_m(i,j,k));
            ST_m(i,j,k)=100*exp(C_m(i,j,k)+D_m(i,j,k));
        end
        ST_mavg(k,i)=mean(ST_m(i,:,k));
        Mk_m(k,i)=max(0,ST_mavg(k,i)-K);

    end

end


for k=1:n
    G(k)=(exp(-r*Ft))*mean(Mk_m(k,:))  ;
end

Est_Est_m=mean(G);
error_F=std(F);
error_G=std(G);
histogram(F,'Normalization','probability')
hold on
histogram(G,'Normalization','probability')
```

9 References

(a) Introduction to probability Models by Sheldon Ross

(b) Theory and estimation of Drift Dynamics of Asset Price Process(MS Thesis) by Akash Verma

(c) Kemna, A. G. Z. & Vorst, A. C. F., 1990. "A pricing method for options based on average asset values," Journal of Banking & Finance

(d) Paul Wilmott on Quantitative Finance by Paul Wilmott

(e) www.stat.columbia.edu

(f) www.ntu.edu.sg

(g) www.utexas.edu

(h) www.wikipedia.org

(i) www.albany.edu

(j) people.stern.nyu.edu

(k) www.optionstrading.org

(l) theoptionprophet.com

(m) www.investopedia.com

10 Acknowledgement

We are extremely grateful to Prof.Dr. Anindya Goswami for giving us an opportunity to take up this project in which we have gained a lot of knowledge in simulating random variables and the intricacies involved in it which would benefit us greatly in the future. We are indebted to his exemplary guidance, constant encouragement, and careful monitoring throughout the project. We would also like to thank each other for providing assistance whenever required,supporting each other and helping each of us to learn the importance of team spirit.