SCHOOL OF ECONOMICS, UNIVERSITY OF HYDERABAD

Comparison of Black-Scholes Merton and Binomial Option Pricing Model with Actual Market Price: A Case Study of Netflix Stock Options

Project: Financial Derivatives and Risk Management

Under the guidance of Dr. Alok Mishra.

MAYUR INGOLE (17SEMA07)

11/5/2019

CONTENTS

INTRODUCTION 2
DATA ANALYSIS: SOURCES AND METHODOLOGY 2
BLACK-SCHOLES OPTION PRICING MODEL 6
BINOMIAL OPTION PRICING MODEL 9
COMPARING ALL THREE PRICES 12
EVALUATING PUT-CALL PARITY 14
CONCLUSION 17
REFERENCES 18

INTRODUCTION

Options are derivative contracts which give the bearer of the contract the right, but not obligation, to sell or buy the underlying asset from the writer of the contract. The right can be exercised either at expiration date of the contract (European Option) or at any time till the expiration date of the contract (American Option). A call option gives the bearer of the contract the right to buy, and a put option gives the bearer of the contract the right to sell, the underlying asset. The option contracts are either traded in an exchange in standardized form or they are traded over-the-counter in customized form. When the underlying asset is a stock then the options are known as stock options. Options can either be used to earn money by speculating about the future direction of the value of underlying asset based on market opinion or they can be used to limit losses through hedging. Usually the bearer of the option contract has to pay a premium to the writer of the option contract. The premium is usually paid at the time the contract is signed. The premium is also known as the value of the option or the option price.

The value of an option consists of two parts: intrinsic value and extrinsic value. The intrinsic value is the amount of payoff earned if the option is exercised and the underlying asset is disposed off immediately at the market price. Extrinsic value is intrinsic value plus a premium based on the time to expiration and the volatility of the underlying asset. While the value of option decreases with decreasing time to maturity, the option value increases with increasing volatility. The value of option can be calculated using mathematical models which take into account both the intrinsic as well as extrinsic value of an option.

Our purpose in this assignment is to estimate option value using two such mathematical models for non-dividend paying european options: the Black-Scholes-Merton model and the Binomial model. We will then compare these estimated values with actual market price of the option. We will also examine the Put-Call Parity relationship in all the three cases.

DATA ANALYSIS: SOURCES AND METHODOLOGY

For the purpose of our analysis we will do the case study of Netflix stocks listed on NASDAQ and its widely traded 3 month stock options. We will use an option chain with an expiration date of “08-11-2019” as our unit of analysis. All the data analysis, modelling, and visualization will be done using “R software”. Specifically we will use the “quantmod” package which allows us to import option chain data for currently traded options from “Yahoo! Finance”. We will create functions to estimate Black-

Scholes-Merton(BSM) and Binomial option prices from the given inputs: current stock price(S), strike price(K), risk free rate of interest(r), volatility(**σ**/sigma), time period to maturity(T), and type of option(Call/Put).

Following are the R packages we require:

```
> library(quantmod)
> library(lubridate)
> library(dplyr)
> library(ggplot2)
> library(tidyr)
```

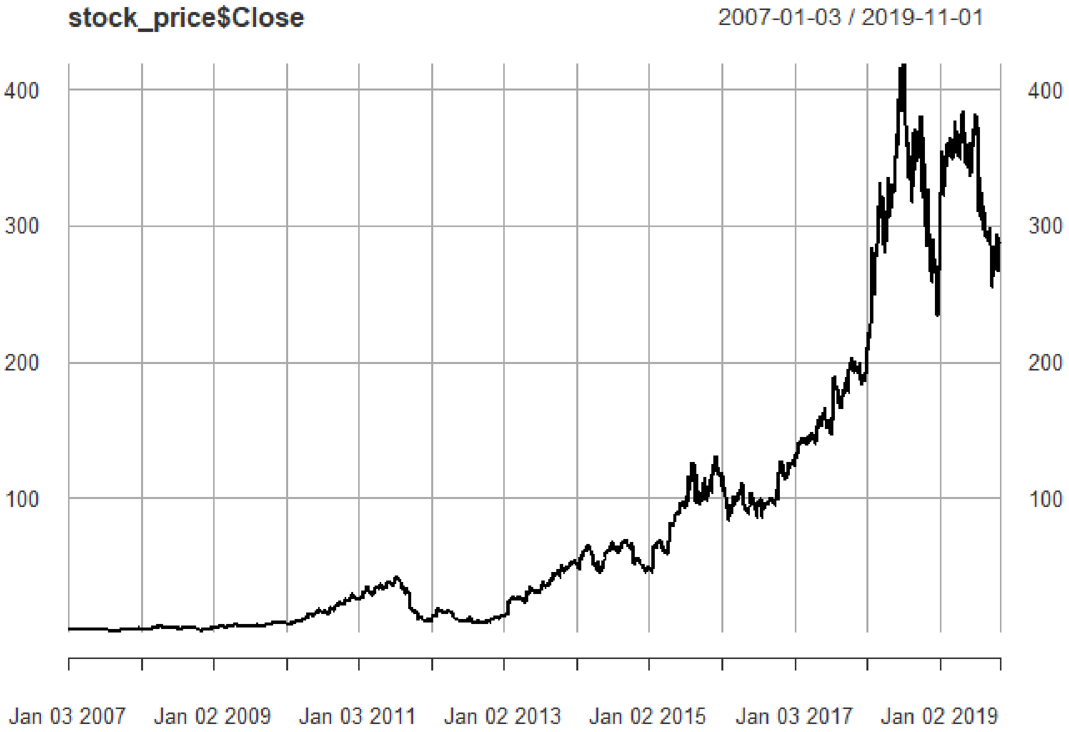
Following is the external customizable input for the code: It includes the ticker for the specific stock, the expiration date for the option, the risk free rate of interest, the number of past years' data to be used to calculate volatility, and the number of steps to used in Binomial tree. We will use the current interest rate on a three-month U.S. Treasury bill as our risk free rate of interest.

```
> # NOTE: check beforehand whether the option chain data for that particular
ticker and expiration date is available on Yahoo Finance
> ticker <- 'NFLX'
> expiration_date <- as.Date('2019-11-08')
> riskfree_rate <- 0.0193
> # number of past years to calculate annualized volatility
> Y <- 5
> # number of steps in Binomial Tree
> N <- 3
```

We will import the option chain and stock price data, and plot it.

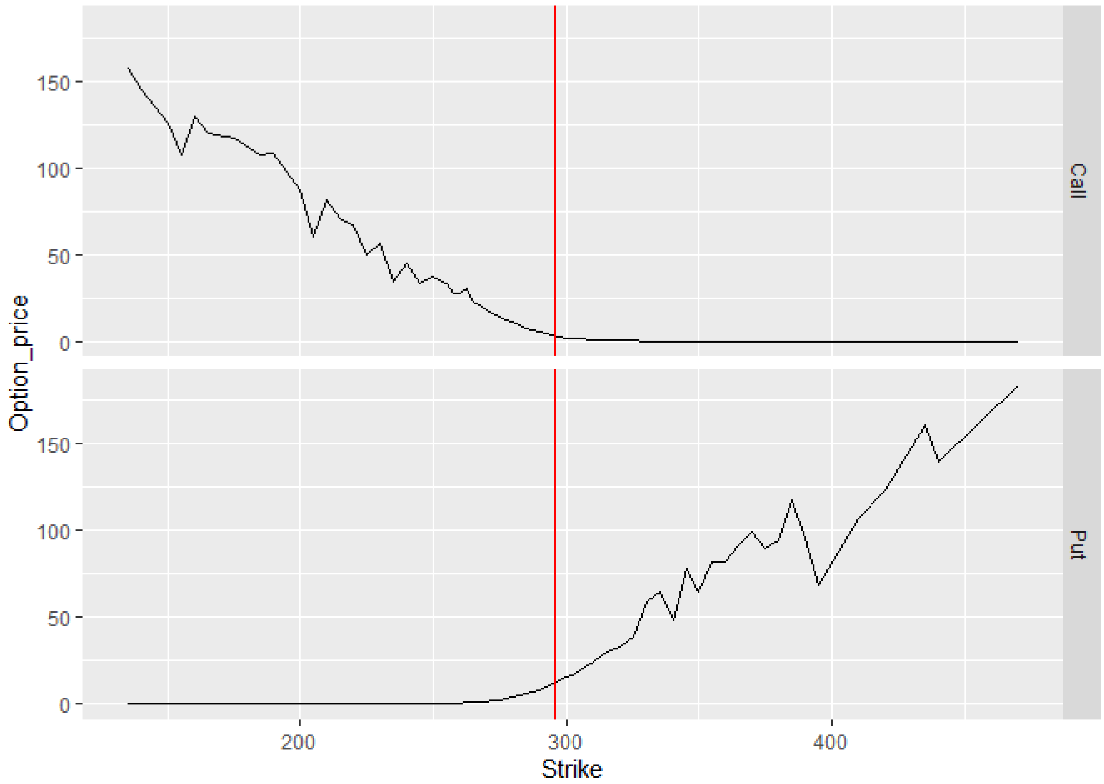
```
> options(timeout= 4000000)
> options_chain_data <- getOptionChain(ticker, Exp = expiration_date)
> # separate call & put; select only strike, volume, last price which we need
to calculate intrinsic value of an option
> calls <- options_chain_data$calls[,c('Strike', 'Last', 'Vol')]
> puts <- options_chain_data$puts[,c('Strike', 'Last', 'Vol')]
> colnames(calls)[2] <- 'Market_price'
> colnames(puts)[2] <- 'Market_price'
>
> stock_price <- getSymbols(ticker, src = 'yahoo', auto.assign = F)
> colnames(stock_price)[4] <- 'Close'
> colnames(stock_price)[6] <- 'Adjusted'
> stock_price$Returns <- diff(log(stock_price$Adjusted))
>
> contract_date <- expiration_date
> month(contract_date) <- month(expiration_date) - 3
> maturity_time <- as.numeric(expiration_date - contract_date)/365
>
> So <- as.numeric(stock_price$Close[contract_date])
```

```
>
> plot(stock_price$Close)
```



The stock prices roughly resemble Geometric Brownian Motion with constant drift and volatility.

```
> calls %>%
+   select(Strike, Call = Market_price) %>%
+   inner_join(puts %>% select(Strike, Put = Market_price), by = 'Strike') %>
%
+   gather(key = 'Option_type', value = 'Option_price', -Strike) %>%
+   ggplot(aes(x = Strike, y = Option_price)) +
+   geom_line() +
+   geom_vline(xintercept = So, col = 'red') +
+   facet_grid(rows = vars(Option_type))
```



The vertical red line indicates the current stock price (So) at the time when the contract was signed.

As we can observe: the market price of call options decreases as strike price increase while the market price of put options increases as strike prices increase. This observation is consistent with our theoretical assumptions about the behavior of option prices.

Now we will exclude observations where market price of the option is less than its intrinsic value which is max[St – K, 0] for call options and max[K –St , 0] for put options since these observations are anomalous with our theoretical assumptions.

```
> calls$intrinsic <- So - calls$Strike
> calls$dummy <- ifelse(calls$intrinsic <= calls$Market_price, 1, 0)
> number_of_inconsistent_calls <- sum(!calls$dummy)
> calls <- calls %>% filter(dummy == 1)
>
> puts$intrinsic <- puts$Strike - So
> puts$dummy <- ifelse(puts$intrinsic <= puts$Market_price, 1, 0)
> number_of_inconsistent_puts <- sum(!puts$dummy)
> puts <- puts %>% filter(dummy == 1)
```

BLACK-SCHOLES OPTION PRICING MODEL

The Black-Scholes-Merton Option Pricing Model was developed by three economists: Fischer Black, Myron Scholes and Robert Merton. It was published, in a 1973 paper titled "The Pricing of Options and Corporate Liabilities," in the Journal of Political Economy.

The model's assumptions are:

- European Options
- No Dividend
- Efficient Markets
- No Transaction Cost
- Stock Returns are normally distributed

The formula of the BSM model is:

$$C = S.N(d1) - K.e^{-r.T}.N(d2)$$

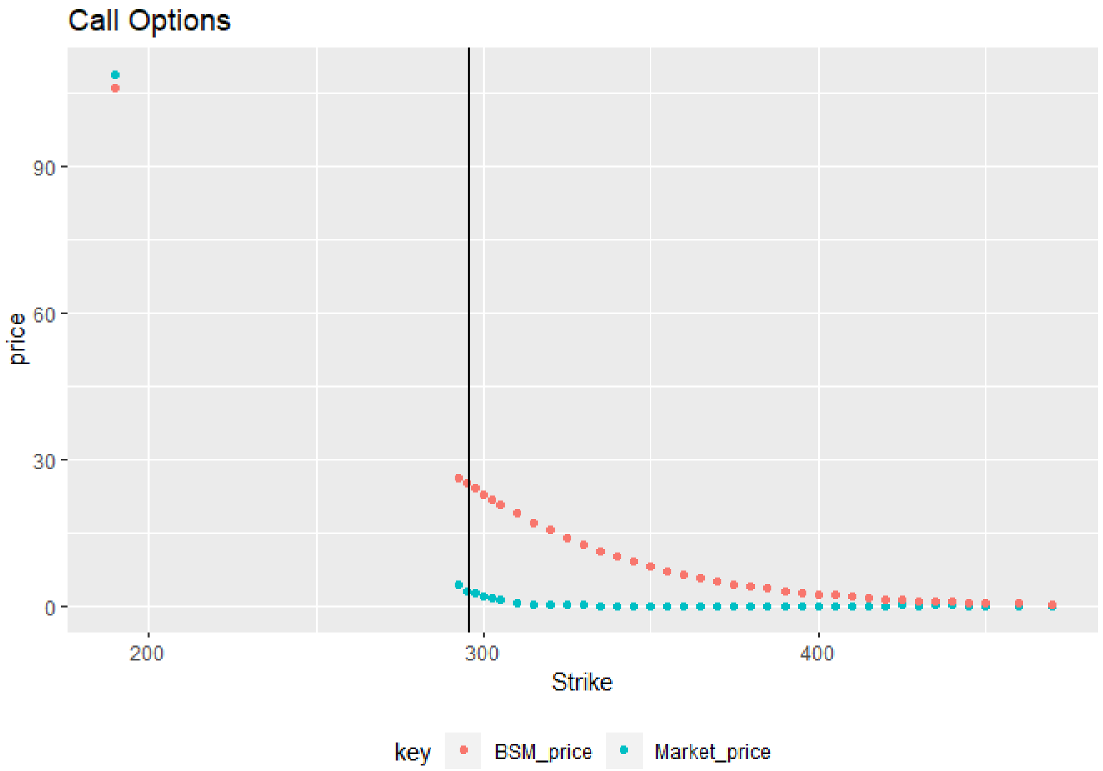
Where

$$d1 = \frac{\left(\ln\left(\frac{S}{K}\right) + \left(r\ +\ \frac{\sigma v^2}{2}\right)T\right)}{(\frac{\sigma s}{\sqrt{T}})}$$

$$d2 = d1 -\ \sigma s\sqrt{T}$$

The volatility is the only unobservable input in BSM model. Therefore we will use historical volatility of the stock as a proxy for the volatility in the stock returns. We will take last five years stock returns and calculate its standard deviation and divide it by square root of 252(number of working day in a year) to calculate the annualized volatility and use it as volatility in the formula.

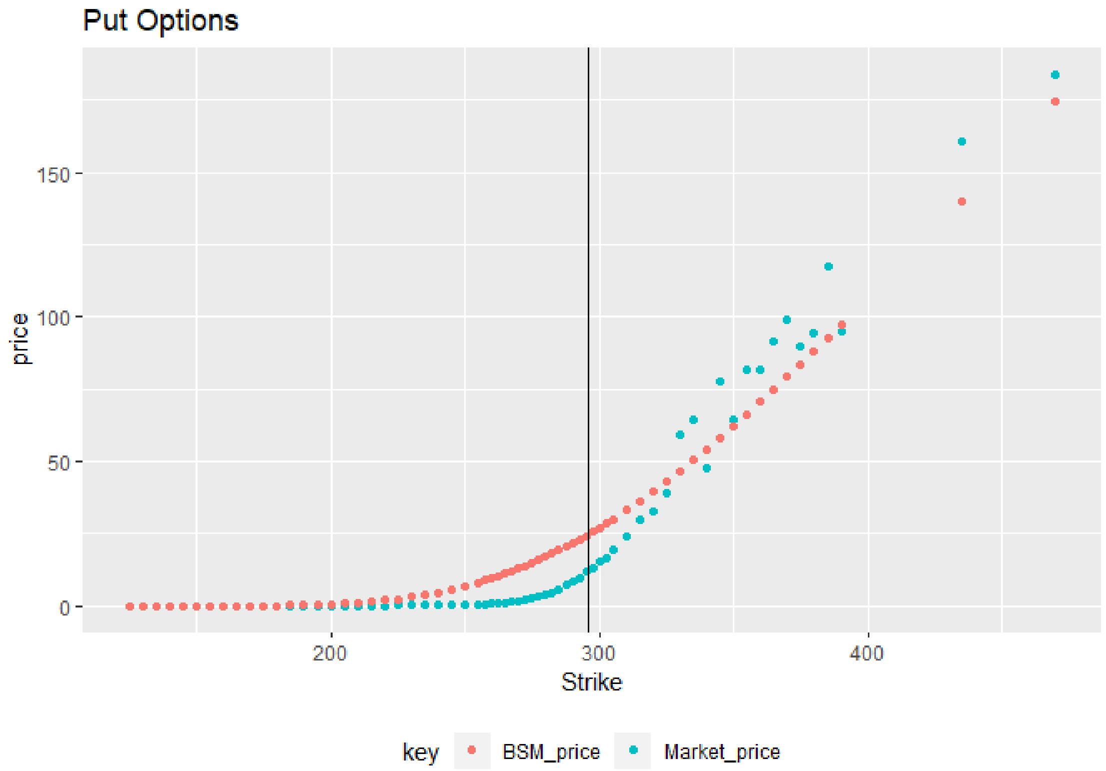
```
> # multiply volatility of returns by 252 to annualize it
> returns_annualized_volatility <- sd(stock_price$Returns[paste(max(year(inde
x(stock_price))) - Y, '/')], na.rm = T)*sqrt(252)
>
> # creating a function to calculate BSM option price
> bs.opm <- function(S,K,T,r,sigma,type){
+   d1<-(log(S/K)+(r+0.5*sigma^2)*T)/(sigma*sqrt(T))
+   d2<-d1-sigma*sqrt(T)
+   if(type=="Call"){
+     opt.val<-S*pnorm(d1)-K*exp(-r*T)*pnorm(d2)
+     }
+   if(type=="Put"){
+     opt.val<-K*exp(-r*T)*pnorm(-d2)-S*pnorm(-d1)
+     }
+   opt.val
+ }
>
> calls$BSM_price <- vapply(X = calls$Strike, FUN = bs.opm, FUN.VALUE = doubl
e(1), S = So, T = maturity_time, r = riskfree_rate, sigma = returns_annualize
d_volatility, type = 'Call')
>
> puts$BSM_price <- vapply(X = puts$Strike, FUN = bs.opm, FUN.VALUE = double(
1), S = So, T = maturity_time, r = riskfree_rate, sigma = returns_annualized_
volatility,  type = 'Put')
>
>
> calls %>%
+   select(Strike, Market_price, BSM_price) %>%
+   gather(key = 'key', value = 'price', -Strike) %>%
+   ggplot(aes(x = Strike, y = price, col = key)) +
+   geom_point() +
+   geom_vline(xintercept = So) +
+   ggtitle('Call Options') +
+   theme(legend.position="bottom")
```



The vertical black line indicates the current stock price (So) at the time when the option contract was signed.

As we can observe that there are very few observations for call options below the current stock price. For the strike price above the current stock price, the option price calculated from the BSM model converges as the strike price increases. This does indicate that the market is expecting for the stock price to rise.

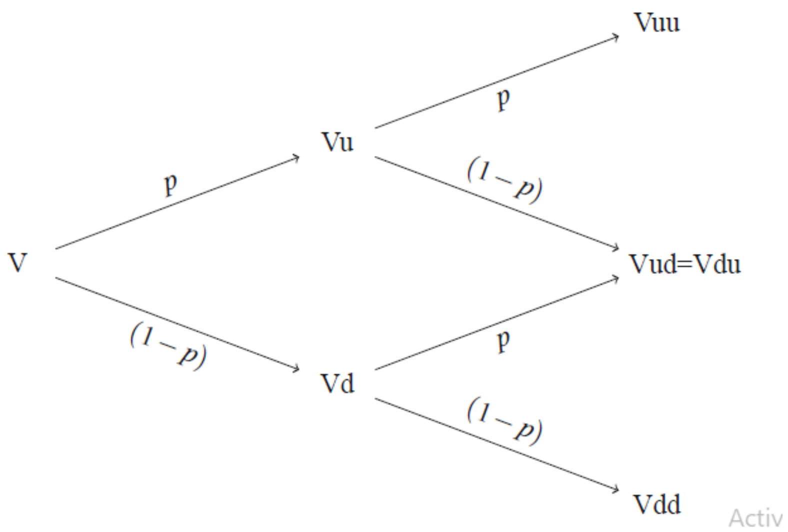
```
> puts %>%
+   select(Strike, Market_price, BSM_price) %>%
+   gather(key = 'key', value = 'price', -Strike) %>%
+   ggplot(aes(x = Strike, y = price, col = key)) +
+   geom_point() +
+   geom_vline(xintercept = So) +
+   ggtitle('Put Options') +
+   theme(legend.position="bottom")
```



As compared to the call options the put options are fairly evenly distributed around the current stock price. For strike price below current stock price the market price converges to BSM option price as strike price decreases. For the strike price above current stock price the market price is scattered around BSM option price.

BINOMIAL OPTION PRICING MODEL

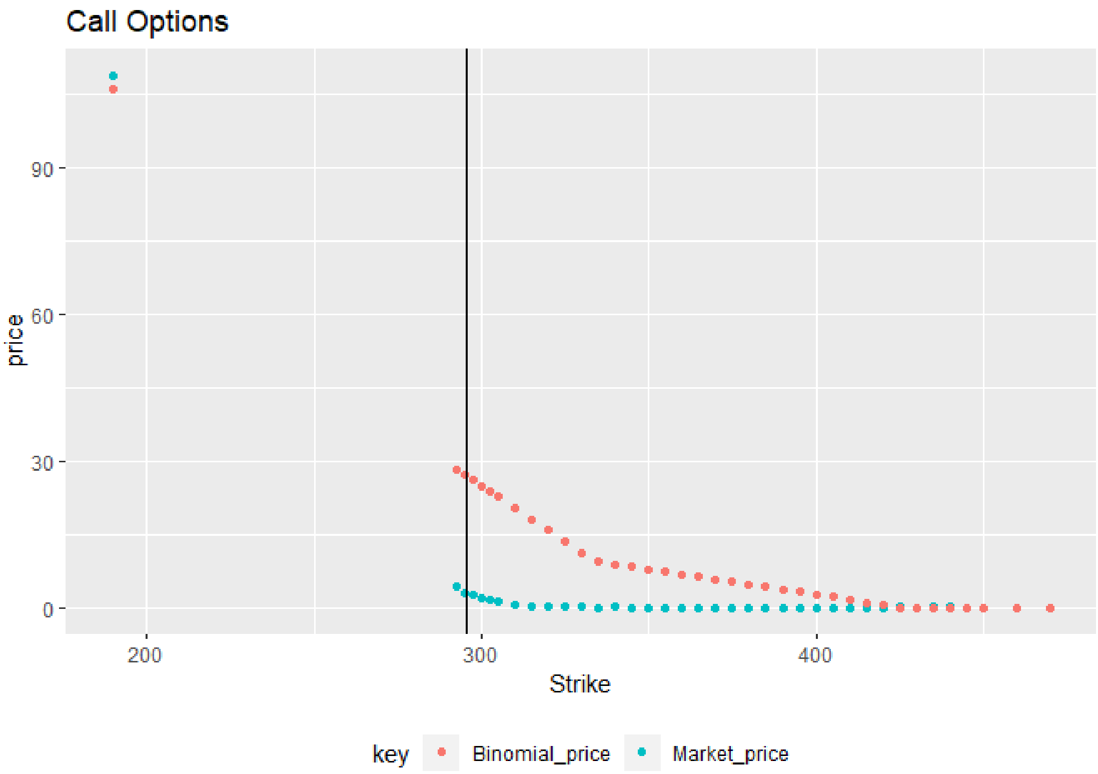
The Binomial Model is a tree based model which models the path of an asset price in discrete time steps. If we assume that V is the current stock price, and its probability of going up is “p” and we assume that the stock price move is discrete steps where either it moves up by a factor of “u” or it moves down by a factor of “d” then the path followed by the stock after two steps will be:



Now we will create a function using two for loops which replicates this process and which takes the number of steps (n) in the binomial tree as its additional input.

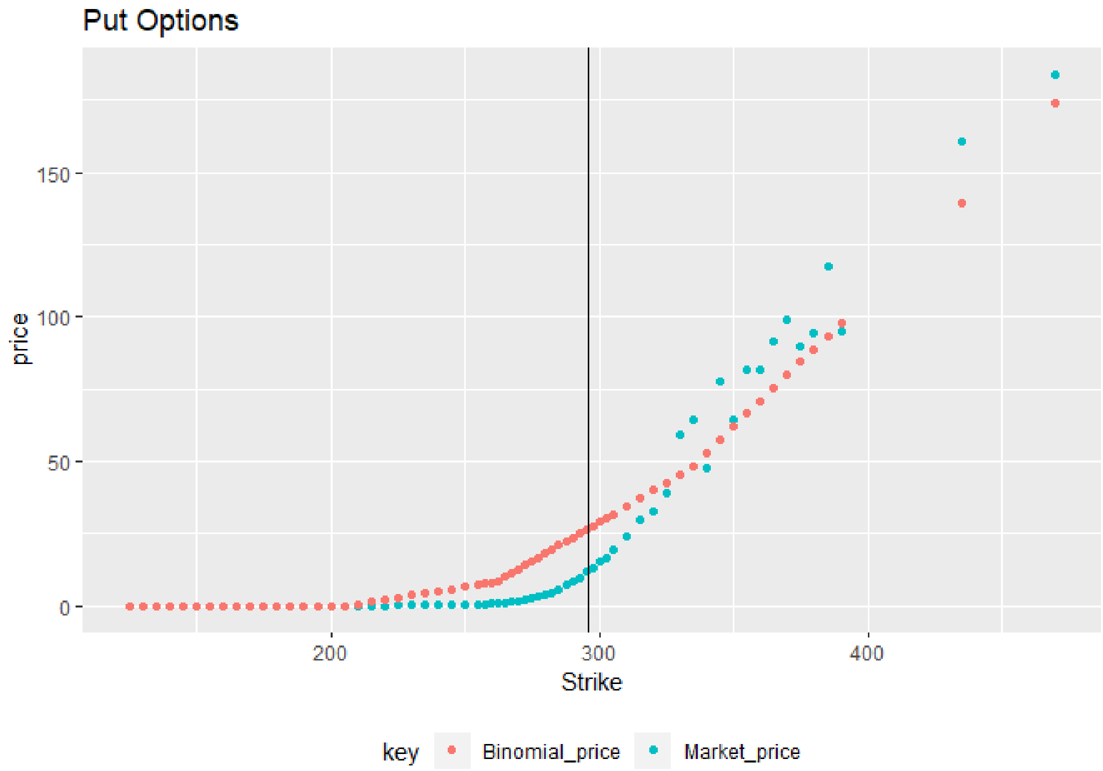
```
> # Binomial OPM function
> Binomial_OPM <- function(S,K,T,r,sigma,n,type) {
+   x=NA
+   if (type=="Call") x=1
+   if (type=="Put") x=-1
+   if (is.na(x)) stop("Option Type can only be call or put")
+   dt=T/ n
+   u=exp(sigma*sqrt(dt))
+   d=1/ u
+   p=((1+r*dt)-d)/ (u-d)
+   disc<- (1+r*dt)
+   OptVal<- x*(S*u^(0:n)*d^(n:0)-K)
+   OptVal=ifelse(OptVal<0,0,OptVal)
+   for (j in seq(from=n-1,to=0,by=-1))
+     for (i in 0:j)
+       OptVal[i+1]=(p*OptVal[i+2]+(1-p)*OptVal[i+1])/disc
+   value=OptVal[1]
+   value
+ }
>
> calls$Binomial_price <- vapply(calls$Strike, FUN = Binomial_OPM, FUN.VALUE
= double(1), S = So, T = maturity_time, r = riskfree_rate, sigma = returns_an
nualized_volatility, n = N, type = 'Call')
>
> puts$Binomial_price <- vapply(puts$Strike, FUN = Binomial_OPM, FUN.VALUE =
double(1), S = So, T = maturity_time, r = riskfree_rate, sigma = returns_annu
alized_volatility, n = N, type = 'Put')
>
```

```
> calls %>%
+   select(Strike, Market_price, Binomial_price) %>%
+   gather(key = 'key', value = 'price', -Strike) %>%
+   ggplot(aes(x = Strike, y = price, col = key)) +
+   geom_point() +
+   geom_vline(xintercept = So) +
+   ggtitle('Call Options') +
+   theme(legend.position="bottom")
```



As we can observe there are less observation for strike price below current stock price, and for strike price above current strike price the market price and the Binomial price converge as the stock price increases.

```
> puts %>%
+   select(Strike, Market_price, Binomial_price) %>%
+   gather(key = 'key', value = 'price', -Strike) %>%
+   ggplot(aes(x = Strike, y = price, col = key)) +
+   geom_point() +
+   geom_vline(xintercept = So) +
+   ggtitle('Put Options') +
+   theme(legend.position="bottom")
```

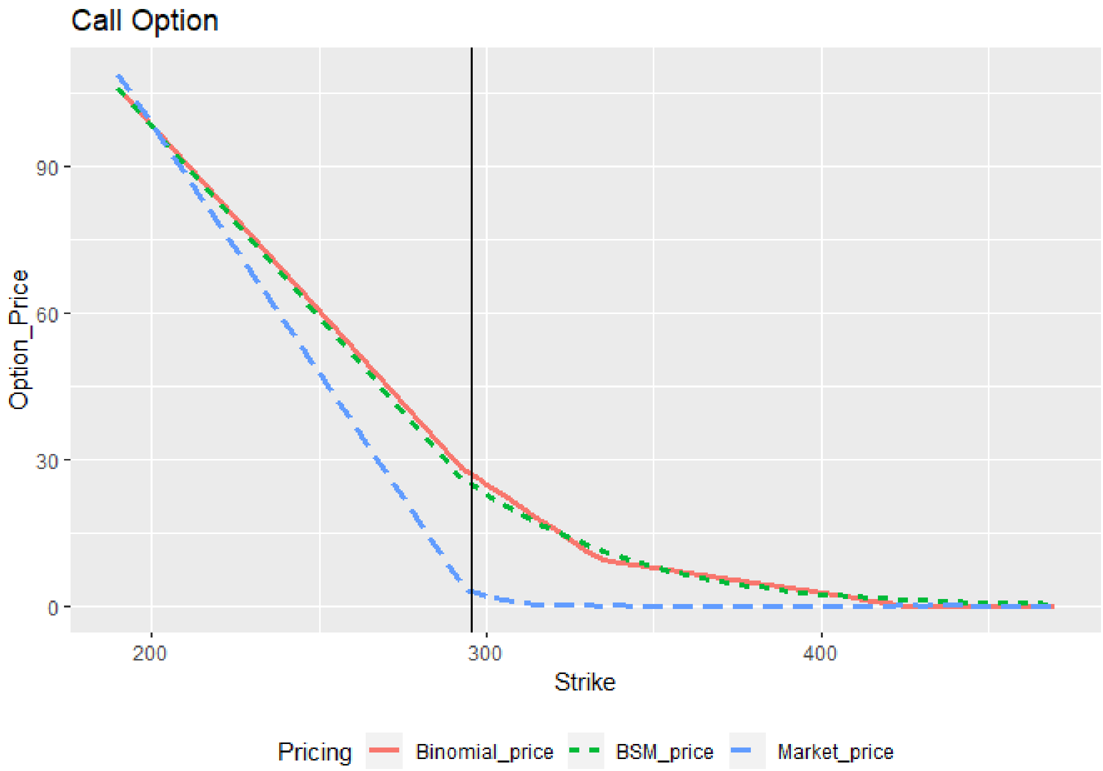


As compared to the call options the put options are fairly evenly distributed around the current stock price. For strike price below current stock price the market price converges to Binomial price as strike price decreases. For the strike price above current stock price the market price is scattered around Binomial option price.

COMPARING ALL THREE PRICES

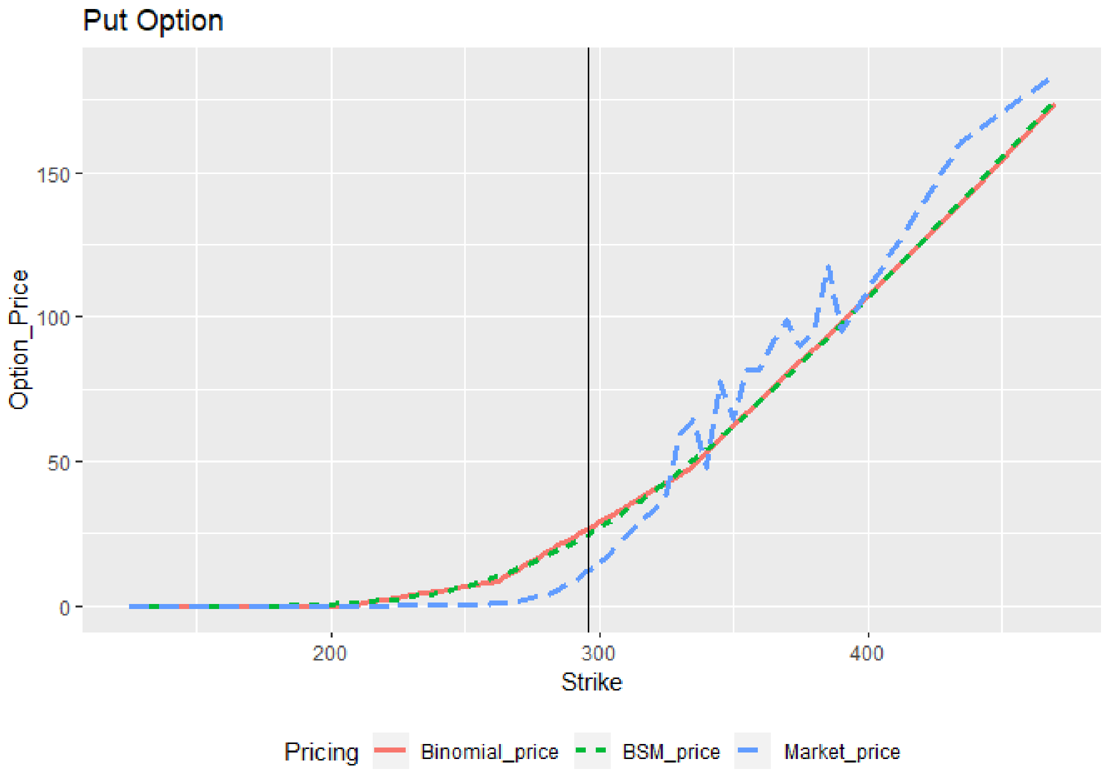
Now we will plot all the three prices (market, BSM, Binomial) on a single graph and compare them.

```
> calls %>%
+   select(Strike, Market_price, BSM_price, Binomial_price) %>%
+   gather(key = 'Pricing', value = 'Option_Price', -Strike) %>%
+   ggplot(aes(x = Strike, y = Option_Price, col = Pricing, linetype = Pricin
g)) +
+   geom_line(size = 1.2) +
+   geom_vline(xintercept = So) +
+   ggtitle('Call Option') +
+   theme(legend.position="bottom")
```



As we can observe the line for Binomial option price and the BSM option price are almost same. While the market price converges to Binomial and BSM option price as the strike price moves away from the current stock price.

```
> puts %>%
+   select(Strike, Market_price, BSM_price, Binomial_price) %>%
+   gather(key = 'Pricing', value = 'Option_Price', -Strike) %>%
+   ggplot(aes(x = Strike, y = Option_Price, col = Pricing, linetype = Pricin
g)) +
+   geom_line(size = 1.2) +
+   geom_vline(xintercept = So) +
+   ggtitle('Put Option') +
+   theme(legend.position="bottom")
```



Similar to call options, the put option prices of BSM and Binomial model are almost same. For strike price below current option price the market price converge to Binomial and BSM option price as the strike price decreases while for strike price above current stock price the market price is scattered around Binomial and BSM option price.

EVALUATING PUT-CALL PARITY

The Put-Call Parity condition for the same underlying asset, strike price, and expiration date is:

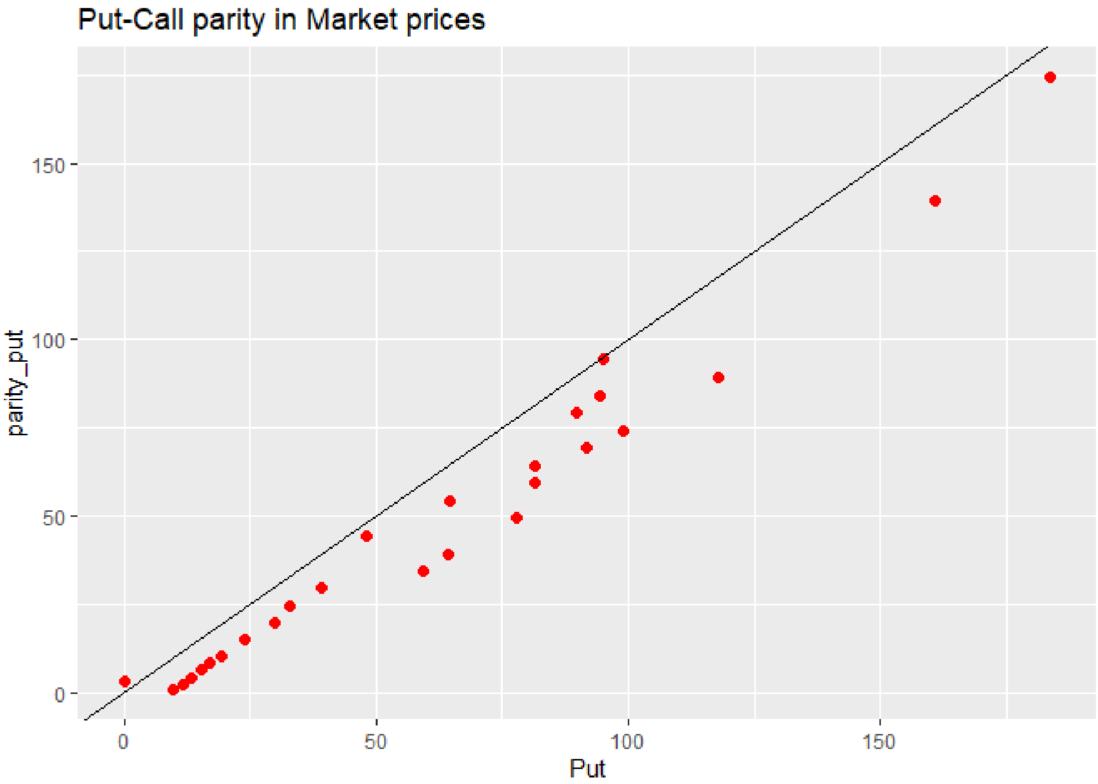
$$p + S = c + K.e^{-r.T}$$

Where "p" and "c" are call and put option price respectively.

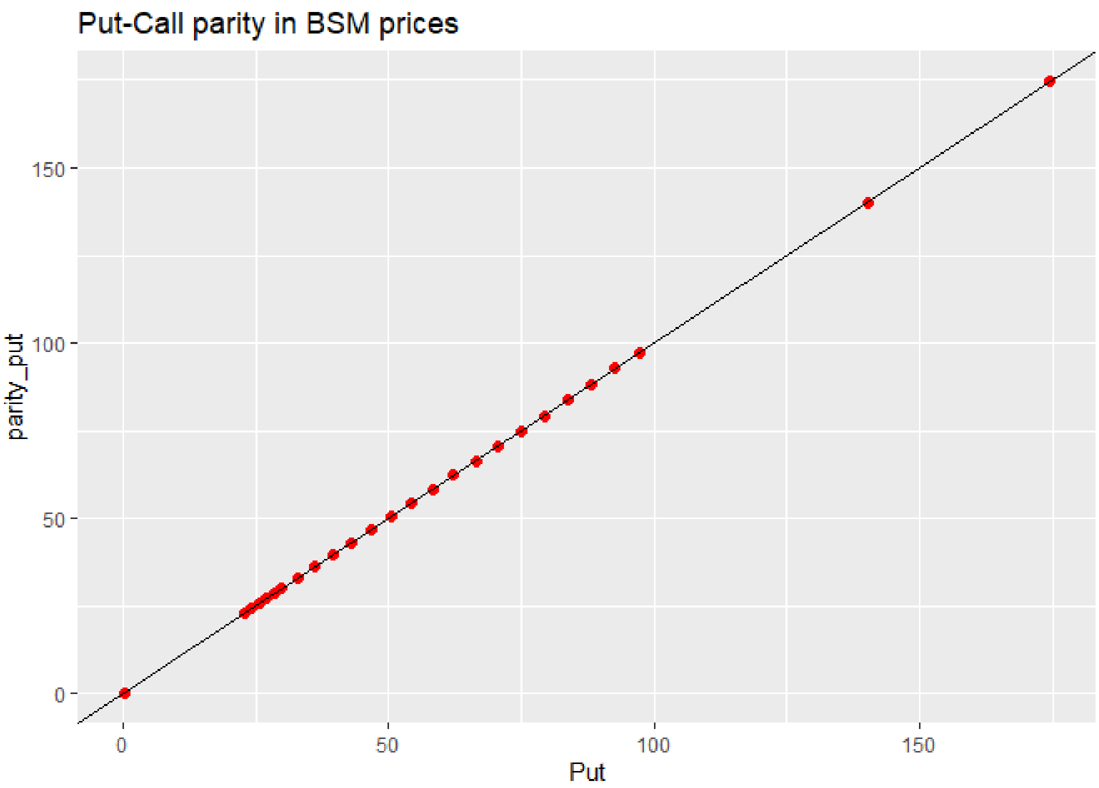
The condition is based on the assumption of no arbitrage opportunity in the market and return at risk free rate of interest.

In the below graph the actual option put option price is on the x-axis and the put option price calculated by using the condition of put-call parity and the call option price is on the y-axis. The red dots indicate our observations and the black line is the 45^0 line indicating the equality between x and y values. Thus if the red dots lie on the black line then we can say that the put-call parity condition is fulfilled.

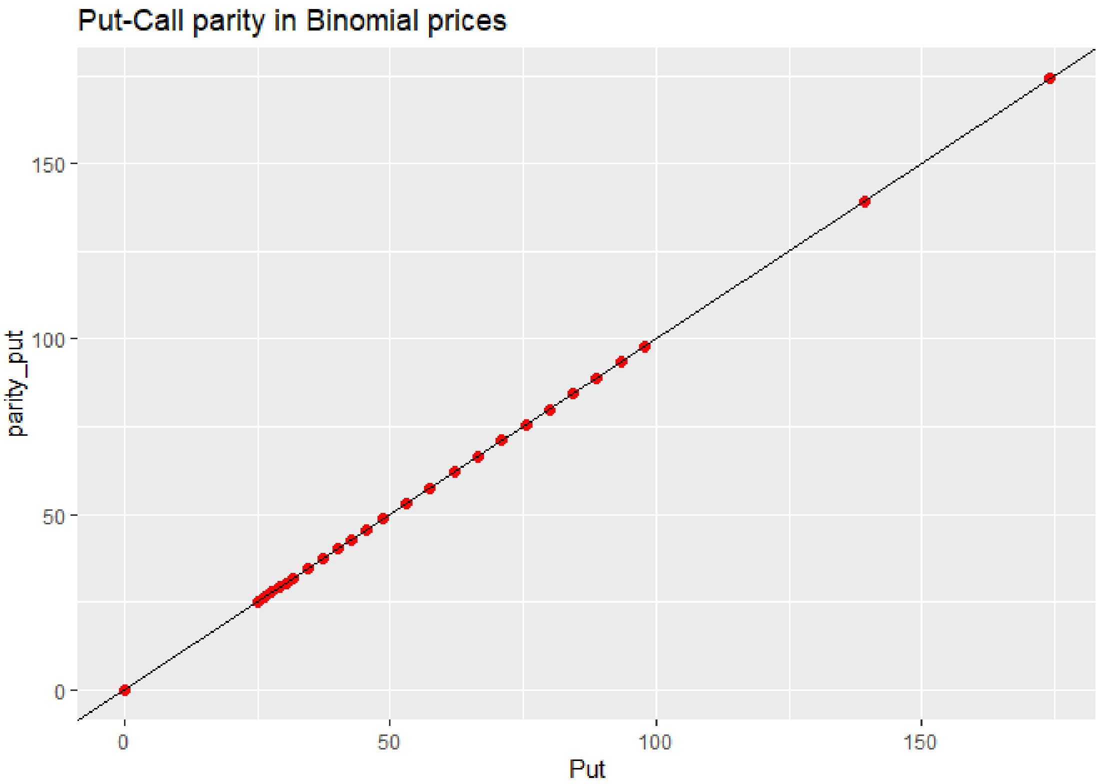
```
> # for Market price
> parity_data_market <- puts %>%
+   select(Strike, Put = Market_price) %>%
+   inner_join(calls %>% select(Strike, Call = Market_price), by = 'Strike')
>
> parity_data_market$parity_put <- parity_data_market$Call + parity_data_mark
et$Strike*exp(-riskfree_rate*maturity_time) - So
>
> parity_data_market %>%
+   ggplot(aes(x = Put, y = parity_put)) +
+   geom_point(col = 'red', size = 2) +
+   geom_abline() +
+   ggtitle('Put-Call parity in Market prices')
```



```
> # for BSM price
> parity_data_BSM <- puts %>%
+   select(Strike, Put = BSM_price) %>%
+   inner_join(calls %>% select(Strike, Call = BSM_price), by = 'Strike')
>
> parity_data_BSM$parity_put <- parity_data_BSM$Call + parity_data_BSM$Strike
*exp(-riskfree_rate*maturity_time) - So
>
> parity_data_BSM %>%
+   ggplot(aes(x = Put, y = parity_put)) +
+   geom_point(col = 'red', size = 2) +
+   geom_abline() +
+   ggtitle('Put-Call parity in BSM prices')
```



```
> # for Binomial price
> parity_data_Binomial <- puts %>%
+   select(Strike, Put = Binomial_price) %>%
+   inner_join(calls %>% select(Strike, Call = Binomial_price), by = 'Strike'
)
>
> parity_data_Binomial$parity_put <- parity_data_Binomial$Call + parity_data_
Binomial$Strike*exp(-riskfree_rate*maturity_time) - So
>
> parity_data_Binomial %>%
+   ggplot(aes(x = Put, y = parity_put)) +
+   geom_point(col = 'red', size = 2) +
+   geom_abline() +
+   ggtitle('Put-Call parity in Binomial prices')
```



As we can observe from the above three graphs: all the red dots for the BSM and Binomial option price lie on the black line which indicates the fulfillment of the put-call parity condition. This is expected result since both these option pricing models are theoretical model which are built on the implicit assumption of put-call parity condition.

For the market prices almost all the actual put option prices lie below the put option price calculated from the call option prices using the put-call parity condition. This can be considered as an indication of market inefficiency and there exist lot of arbitrage opportunities in the market.

CONCLUSION

Black-Scholes-Merton (BSM) option pricing model is theoretical model based on the assumption of returns being distributed normally. While the Binomial model is a simple decision tree based non-parametric model which doesn't assume anything about the distribution of returns. We did a case study of Netflix stock options to study these models. In the BSM model we used the annualized standard deviation of stock returns for last five year as our volatility. We estimated the Binomial option prices using a 3 step decision tree. In our estimation of BSM option prices and Binomial option prices we found them to be almost same. For the call option, the number of observation for strike price below current stock price at the start of contract was very less. From this we inferred that the market is expecting the stock price to go up. For call options with strike price above current stock price and for put options with strike price below current stock price, the market price converges to BSM as well as Binomial option price as the strike price moves away from current stock price. For put options with strike price above current stock price, the market option price is scattered around BSM as well as Binomial option price. When we plotted all the three option prices on the same graph we observed that: for call options the market price is always below BSM as well as Binomial option price, which are almost same and all three converge as the strike price moves away from the current stock price; for put option with strike price above current stock price, the market option price is scattered around BSM as well as Binomial option price and for strike price below current stock price the three prices converge as strike price decrease. We then examined the put-call parity condition in all three prices. We observed, as expected, that the condition holds true for BSM as well as Binomial option prices, since these models are based on the assumption of no arbitrage. The market option price doesn't obey the put-call parity condition. This indicates that there is still arbitrage opportunity in the market which can be exploited to earn profit above risk free rate of interest. It also means the market is inefficient.

REFERENCES

1. John C. Hull; Options, Futures, and Other Derivatives - Pearson (2017); Chapter 13, 15, 21.

2. Clifford S. Ang; Analyzing Financial Data and Implementing Financial Models Using R - Springer (2015); Chapter 9.

3. https://in.finance.yahoo.com/

4. https://www.investopedia.com/