Homework #6

Dustin Leatherman

5/17/2020

9

Consider a distribution with the density function as

$$\frac{1}{B(\alpha,\beta)}x^{\alpha-1}(1-x)^{\beta-1}, \quad 0 \le x \le 1$$

Derive a rejection-acceptance method for generating random variables with above pdf with $\alpha = 2, \beta = 2$ from $U[0,1]$ random variables. Generate 1000 such numbers.

With $\alpha = 2,\ \beta = 2$.

$$\frac{1}{B(2,2)}x(1-x)$$

$\frac{1}{B(2,2)} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} = \frac{(4-1)!}{(2-1)!\cdot(2-1)!} = 6$

Thus,

$$6x(1-x)$$

$Q = \frac{f_X(y)}{f_Y(y)} = \frac{6x(1-x)}{1}$

$\frac{\partial Q}{\partial x} = 6 - 12x = 0 \to x = \frac{1}{2}$

There is only one value since the derivative is a first-order polynomial, thus $\frac{1}{c} = \frac{6\cdot 0.5(1-0.5)}{1} = \frac{3}{2} \to c = \frac{2}{3}$

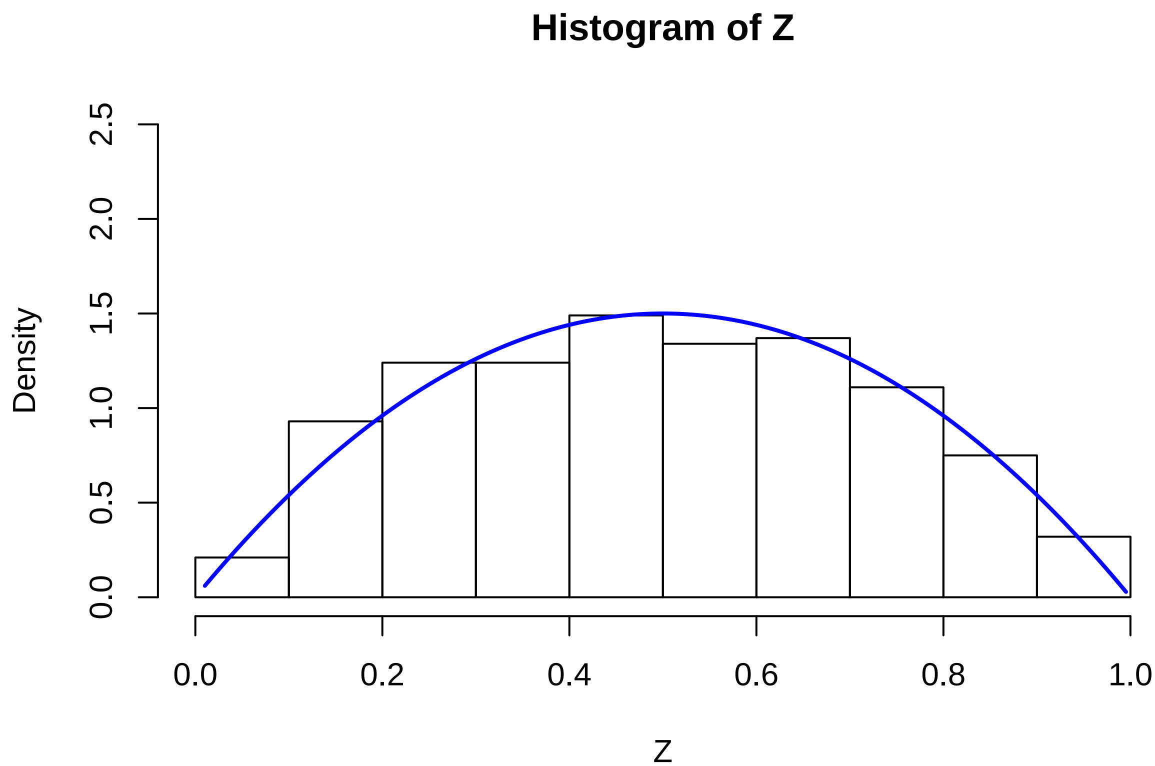
Number simulations required to run

$N = 1.1 \cdot E(N) = \frac{1000}{c} \cdot 1.1 = \frac{3}{2} \cdot 1000 \cdot 1.1 = 1650$

Accept-Rejection Decision

$$W_i \le \frac{2}{3} \cdot 6x(1-x) = 4x(1-x)$$

```
set.seed(23423523)
n<-10^3
efficiency <- (2/3)
N<-ceiling(n/efficiency*1.1)
X <- runif(N)
W <- runif(N)

Z <- X[which(efficiency * 6 * X * (1 - X) >= W)]
Z <- Z[1:n]

hist(Z, freq=F, ylim=c(0,2.5))

xfit <- seq(min(Z), max(Z),length=1000)
lines(xfit, 6 * xfit * (1 - xfit),col='blue',lwd=2)
```



10

Consider the problem of pricing lookback options for a stock modeled by a geometric Brownian motion with an initial price of \$100, a volatility of 40%, and zero interest rate. Let the expiry time be 12 weeks in the future (consider 52 weeks a year), and let the monitoring frequency be weekly.

$S(0) = 100$

$\sigma = 0.4$

$r = 0$

$T = 12/52$

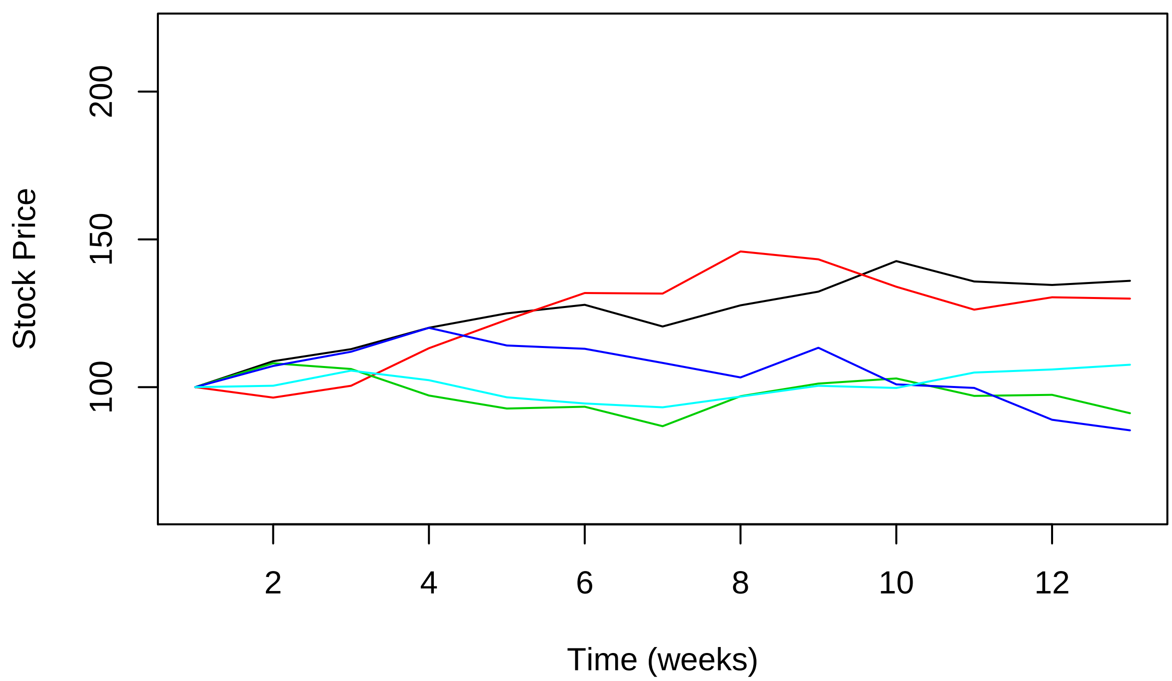
$d = 12$

a

Find the fair price of both the put and call options

```
s0 <- 100
sigma <- 0.4
r <- 0
t <- (12/52)
d <- 12
```

```r
n <- 5
delta <- t/d
grid <- seq(delta, t, length.out = d)

#create a matrix to store asset prices
S <- matrix(rep(0, n * (d + 1)), nrow = n)

#generate nxd pseudo-random normal numbers
x <- matrix(rnorm(n * d), nrow = n)

#generate n sample paths of Brownian Motion
BM <- sqrt(delta) * t(apply(x, 1, cumsum))

S <- cbind(rep(s0, n), s0 * exp(sweep(sigma * BM, MARGIN = 2, (r - sigma^2 / 2) * grid, '+')))

plot(1:(d+1),S[1,],type='l',ylim=c(60,220),col=1,ylab='Stock Price',xlab='Time (weeks)')
for (i in 2:n){
    points(1:(d+1),S[i,],type='l',col=i)
}
```




```r
fair.call <- mean(rep(s0, n) - apply(S, 1, min))

fair.put <- mean(apply(S, 1, max) - rep(s0, n))
```

The Fair Price for Call option under these terms is \$7.63

The Fair Price for Put option under these terms is \$24.86

b

Does the put or the call have a higher price? What is a possible intuitive explanation?

The Put option has a higher price. The risk free interest rate is 0 which means that volatility has a more significant effect on stock price. This means that there is more risk so an option to buy at a future date would be priced higher than an option to sell since there is less guarantee that the money will be made back.