Assignment 2

Group 3

Yijie Wu, CID: 01894265
Qian Zhang, CID: 01939418
Dennis Wen, CID: 01973771

1. Option Pricing in the Binomial Model

a) We can calculate the stock price at each node with binomial tree model:

$$S_{t+1,1} = u * S_t$$
$$S_{t+1,2} = d * S_t$$

Then, we have the binomial tree of stock price:

period	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
stock	100.00	103.95	108.06	112.32	116.76	121.37	126.17	131.15	136.33	141.72	147.31	153.13	159.18	165.47	172.01	178.80
		96.20	100.00	103.95	108.06	112.32	116.76	121.37	126.17	131.15	136.33	141.72	147.31	153.13	159.18	165.47
			92.54	96.20	100.00	103.95	108.06	112.32	116.76	121.37	126.17	131.15	136.33	141.72	147.31	153.13
				89.03	92.54	96.20	100.00	103.95	108.06	112.32	116.76	121.37	126.17	131.15	136.33	141.72
					85.65	89.03	92.54	96.20	100.00	103.95	108.06	112.32	116.76	121.37	126.17	131.15
						82.39	85.65	89.03	92.54	96.20	100.00	103.95	108.06	112.32	116.76	121.37
							79.26	82.39	85.65	89.03	92.54	96.20	100.00	103.95	108.06	112.32
								76.25	79.26	82.39	85.65	89.03	92.54	96.20	100.00	103.95
									73.35	76.25	79.26	82.39	85.65	89.03	92.54	96.20
										70.56	73.35	76.25	79.26	82.39	85.65	89.03
											67.88	70.56	73.35	76.25	79.26	82.39
												65.30	67.88	70.56	73.35	76.25
													62.82	65.30	67.88	70.56
														60.43	62.82	65.30
															58.14	60.43
																55.93

We then need to calculate the probability of q and (1-q):

$$q = \frac{R-d}{u-d} = \frac{1.000333 - 0.9620}{1.0395 - 0.9620} = 0.49461$$

$$1 - q = 1 - 0.49461 = 0.50539$$

<u>To calculate the American call option price:</u>

we can calculate the payoff at period t by:

$$C_T = \max\,(S_T - K, 0)$$

And we work backwards to get the option payoff at each node:

$$C_t = \frac{1}{R} E_t^Q [C_{t+1}]$$

Finally, at each node, we need to compare:

$$\max\left(\frac{1}{R} E_t^Q [C_{t+1}], S_t - K\right)$$

Therefore, we get the binomial tree of American call option. The call option price at time 0 is equal to 2.68

period	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
call	2.68	3.85	5.42	7.48	10.12	13.37	17.25	21.70	26.64	31.94	37.50	43.28	49.29	55.54	62.04	68.80
		1.54	2.31	3.40	4.91	6.94	9.59	12.90	16.89	21.47	26.51	31.86	37.42	43.21	49.22	55.47
			0.79	1.24	1.92	2.92	4.36	6.35	9.01	12.41	16.55	21.30	26.44	31.79	37.35	43.13
				0.35	0.58	0.94	1.52	2.41	3.75	5.69	8.37	11.91	16.28	21.22	26.37	31.72
					0.12	0.22	0.38	0.65	1.11	1.86	3.06	4.91	7.65	11.45	16.20	21.15
						0.03	0.06	0.11	0.20	0.37	0.69	1.25	2.24	3.94	6.80	11.37
							0.00	0.01	0.02	0.03	0.07	0.14	0.28	0.57	1.15	2.32
								0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
									0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
										0.00	0.00	0.00	0.00	0.00	0.00	0.00
											0.00	0.00	0.00	0.00	0.00	0.00
												0.00	0.00	0.00	0.00	0.00
													0.00	0.00	0.00	0.00
														0.00	0.00	0.00
															0.00	0.00
																0.00

b) **To calculate the American put option price:**

we can calculate the payoff at period t by:

$$P_T = \max\,(K - S_T, 0)$$

And we work backwards to get the option payoff at each node:

$$P_t = \frac{1}{R} E_t^Q [P_{t+1}]$$

Finally, at each node, we need to compare:

$$\max\left(\frac{1}{R} E_t^Q [P_{t+1}], K - S_t\right)$$

Therefore, we get the binomial tree of American put option. The put option price at time 0 is equal to 12.23

period	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
put	12.23	9.45	6.93	4.74	2.97	1.64	0.75	0.26	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00
		14.96	11.92	9.07	6.49	4.27	2.51	1.24	0.46	0.10	0.00	0.00	0.00	0.00	0.00	0.00
			17.94	14.72	11.60	8.66	6.00	3.75	2.00	0.82	0.20	0.00	0.00	0.00	0.00	0.00
				21.10	17.79	14.48	11.26	8.21	5.46	3.15	1.43	0.39	0.00	0.00	0.00	0.00
					24.35	21.03	17.65	14.26	10.91	7.73	4.84	2.45	0.78	0.00	0.00	0.00
						27.61	24.35	20.98	17.54	14.04	10.56	7.18	4.08	1.54	0.00	0.00
							30.74	27.61	24.35	20.97	17.46	13.87	10.22	6.56	3.06	0.00
								33.75	30.74	27.61	24.35	20.97	17.46	13.80	10.00	6.05
									36.65	33.75	30.74	27.61	24.35	20.97	17.46	13.80
										39.44	36.65	33.75	30.74	27.61	24.35	20.97
											42.12	39.44	36.65	33.75	30.74	27.61
												44.70	42.12	39.44	36.65	33.75
													47.18	44.70	42.12	39.44
														49.57	47.18	44.70
															51.86	49.57
																54.07

c) It is optimal to early exercise the American put option, if at some nodes:

$$\max\left(\frac{1}{R} E_t^Q [P_{t+1}], K - S_t\right) = K - S_t$$

For American put option in part (b), we can early exercise it.

d) Based on the logic mentioned in part (c), we can get the optimal earliest period to exercise the American put option is at period 7.

period	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
put																0.00
																0.00
																0.00
																0.00
																0.00
																0.00
																0.00
								33.75	30.74	27.61	24.35	20.97	17.46	13.80	10.00	6.05
									36.65	33.75	30.74	27.61	24.35	20.97	17.46	13.80
										39.44	36.65	33.75	30.74	27.61	24.35	20.97
											42.12	39.44	36.65	33.75	30.74	27.61
												44.70	42.12	39.44	36.65	33.75
													47.18	44.70	42.12	39.44
														49.57	47.18	44.70
															51.86	49.57
																54.07

e) Since American options allow early exercise, put-call parity will not hold for American options unless they are held to expiration. Early exercise will result in a departure in the present values of the two portfolios.

2. Pricing Futures Contracts

Since a future contract always worth 0, we have

$$E_k^Q\left[\frac{F_{k+1}-F_k}{R}\right]=0$$

Because R is constant, we have

$$E_k^Q\left[\frac{F_{k+1}-F_k}{R}\right]=E_k^Q[F_{k+1}-F_k]=E_k^Q[F_{k+1}]-E_k^Q[F_k]=0$$

Therefore,

$$E_k^Q[F_{k+1}]=E_k^Q[F_k]$$

Since at $\mathrm{t}=\mathrm{k}+1,$ F_k is known, we have

$$E_k^Q[F_{k+1}]=E_k^Q[F_k]=F_k$$

Therefore, for $0\leq k\leq n$:

$$F_k=E_k^Q[F_{k+1}]=E_k^Q[E_{k+1}^Q[F_{k+2}]]=\cdots=E_k^Q[E_{k+1}^Q[\ldots E_{n-1}^Q[F_n]]]$$

According to tower property, for any variable x and $u\leq t$:

$$E_u[E_t(x)]=E_u[x]$$

Therefore,

$$F_k=E_0^Q[F_n]$$

Since for a future contract, $F_n=S_n$

Hence,

$$F_k = E_0^Q[S_n]$$

Since there is no dividend paid,

$$S_{k+1} = S_k R$$
$$S_n = S_0 R^n$$

So, we have

$$F_k = E_0^Q[S_0 R^n]$$

Since S_0 and R^n are both constant,

$$F_k = E_0^Q[S_n] = R^n S_0$$

3. Convergence of Binomial Model Option Prices to Black-Scholes Prices

a) Compute the prices of European call options using Black-Scholes model

```
#build a Black-Scholes function to calculate the European option prices
BlackScholes <- function(S, sig, r, c, T, t, K, type){

  if(type=="C"){
  d1 <- (log(S/K) + (r - c + sig^2/2)*(T-t)) / sig*sqrt(T-t)
  d2 <- d1 - sig*sqrt(T-t)

  call_bsm <- exp(-c*(T-t))*S*pnorm(d1) - exp(-r*(T-t))*K*pnorm(d2)
  return(call_bsm)}

  if(type=="P"){
  put_bsm <- BlackScholes(S, sig, r, c, T, t, K, "C") + K*exp(-r*(T-t)) - S*exp(-c*(T-t))
  return(put_bsm)}
}

#substitute parameters into Black-Scholes function
bsm_call <- BlackScholes(S=100, sig=0.3, r=0.02, c=0, T=1, t=0, K=100, type="C")
bsm_call
```

```
## [1] 12.82158
```

b) Compute the prices of European call options using binomial model

```
#establish the stock tree
stock_tree <- function(S, sig, del_t, n) {
  tree = matrix(0, nrow=n+1, ncol=n+1)
  u = exp(sig*sqrt(del_t))
  d = exp(-sig*sqrt(del_t))
  for (i in 1:(n+1)) {
    for (j in 1:i) {
      tree[i, j] = S * u^(j-1) * d^((i-1)-(j-1))
    }
  }
  return(tree)
}

#calculate the probability of q
q_prob <- function(r, del_t, sig) {
  u = exp(sig*sqrt(del_t))
  d = exp(-sig*sqrt(del_t))
  return((exp(r*del_t)-d)/(u-d))
}
```

```r
#obtain the option's payoff and price at each node by using binomial tree
binomial_option_price <- function(S, sig, del_t, r, K, type, n) {
  #use the function of stock_tree
  tree=stock_tree(S, sig, del_t, n)
  #use the function of q
  q = q_prob(r, del_t, sig)
  #calculate the payoff of a call/put option
  option_tree = matrix(0, nrow=nrow(tree), ncol=ncol(tree))
  if(type=="C") {
    option_tree[nrow(option_tree),] = pmax(tree[nrow(tree),]-K, 0)
  }
  else if(type=="P") {
    option_tree[nrow(option_tree),] = pmax(K-tree[nrow(tree),], 0)
  }
  #work backward to calculate the option price at each node
  for (i in (nrow(tree)-1):1) {
    for(j in 1:i) {
     option_tree[i,j]=(q*option_tree[i+1,j+1] + (1-q)*option_tree[i+1,j])/exp(r*del_t)
    }
  }
  return(option_tree)
}

#combine all of newly-established functions to get the option price
binomial_option <- function(S, sig, r, T, K, type, n) {
  q = q_prob(r=r, del_t=T/n, sig=sig)
  tree = stock_tree(S=S, sig=sig, del_t=T/n, n=n)
  option = binomial_option_price(S=S, sig=sig, del_t=T/n, r=r, K=K, type=type, n=n)
  return(option[1,1])
}

#substitute parameters into binomial function
binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=10)
```

```
## [1] 12.52997
```

c) As the number of period n increases, the option price calculated by the binomial model will gradually move closer to the option price obtained via Black-Scholes model. As you can see in the graph below, the option price when n=10 is furthest from the Black-Scholes model's option price, while the option price is very similar to that of from Black-Scholes when n=1000. That is to say, the option price calculated by binomial model will converge to the Black-Scholes option price.

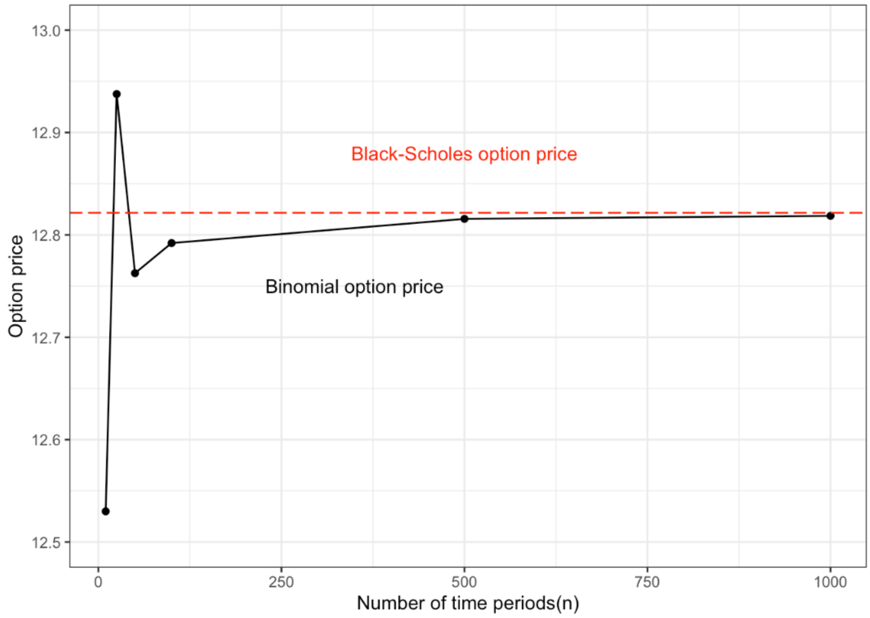
```r
#Change the number of periods (n) to compute different European call option prices
#by using binomial model
option_price_n10 <- binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=10)
option_price_n25 <- binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=25)
option_price_n50 <- binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=50)
option_price_n100 <- binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=100)
option_price_n500 <- binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=500)
option_price_n1000 <- binomial_option(S=100, sig=0.3, r=0.02, T=1, K=100, type="C", n=1000)

#build a data frame to show different number of periods (n)
#and corresponding European call option price
all_price <- as.data.frame(list(n=c(10, 25, 50, 100, 500, 1000), option_price=c(option_price_n10, option_price_n2
5, option_price_n50, option_price_n100, option_price_n500, option_price_n1000)))

all_price
```

```
##      n option_price
## 1   10     12.52997
## 2   25     12.93763
## 3   50     12.76258
## 4  100     12.79204
## 5  500     12.81567
## 6 1000     12.81862
```

```r
#plot the European call option prices calculated by Black-Scholes model and binomial model
library(ggplot2)
ggplot(data=all_price, aes(x=n, y=option_price)) + geom_point() + geom_line() + labs(x="Number of time periods (n)", y="Option price") + geom_hline(yintercept=bsm_call, linetype="longdash", color="red") + ylim(12.5, 13) + theme_bw() + annotate(geom="text", x=500, y=12.88, label="Black-Scholes option price", color="red") + annotate(geom="text", x=350, y=12.75, label="Binomial option price")
```



4. Dynamic Hedging in the Black-Scholes Model