Введение в финансовую математику. Лабораторная работа № 3.

Zhdanov Pavel, C41111

Вариант 12

Задание 1

Пусть цена Европейского put опциона с X = 100$ равна 6$, а текущая цена акции равна S = 93$. Найдите цену Американского Call опциона с тем же страйком, если r = 2% и T = 1. Ответ округлите до целых.

Решение

Цена Американского Call опциона равна цене Европейского Call опциона. Цену Европейского Call опциона можно найти из put-call parity:

$$C - P = S - X * e^{-r(T-t)}$$

$$C = S - X * e^{-r(T-t)} + P$$

```
X = 100
P = 6
S = 93
r = 0.02
dt = 1

C = S - X * exp(-r * dt) + P
print(round(C))

## [1] 1
```

Задание 2

Цель задачи - научиться считать цены Американских опционов при помощи биномиальной модели. Для этого необходимо реализовать функции подсчета:
* Цен европейских опционов
* Payoff-ов
* Цены на i-ом шаге цен i+1 шага
* Цен Европейских опционов при помощи биномиальной модели
* Цены Американского Put опциона при помощи биномиальной модели

Постройте графики зависимости цен в биномиальной модели от числа шагов N, используя следующие параметры: S = 110$, X = 110$, T = 1, r = 10%, sigma = 30%, N = [20:400].

Сравнить полученные результаты:
*Для Евпропейских опционов - с точным решением (Black - Scholes)
*Для Американского Put - с аппроксимацией Barone-Adesi and Whaley (разрешается воспользоваться встроенной функцией)

Поменяйте страйк на X = 97$ и постройте график цен для Евпропейского Call опциона. Как изменилось поведение цены? Какие могут быть причины таких изменений?

Переменные

```
X = 110
S = 110
T = 1
r = 0.1
sig = 0.3
N = 100
dt = T / N
```

Необходимые функции для работы

Цены европейских опционов

```
d1 <- function(S){
  ans = (1/(sig * sqrt(T))) * (log(S/X) + (r + (sig^2)/2) * T)
  return (ans)
}

d2 <- function(S){
  ans = d1(S) - sig * sqrt(T)
  return (ans)
}


call_BS_func <- function(spot_price){
  ans = pnorm(d1(spot_price)) * spot_price - pnorm(d2(spot_price)) * X * exp(-r*
T)
  return (ans)
}

put_BS_func <- function(spot_price){
  ans = pnorm(-d2(spot_price)) * X * exp(-r * T) - pnorm(-d1(spot_price)) * spot
_price
  return (ans)
}
```

Paoyff

```
call_payoff_func <- function(spot_price){
  if (spot_price > X){
```

```
    return (spot_price - X)
  }
  return (0)
}

put_payoff_func <- function(spot_price){
  if (spot_price < X){
    return (X - spot_price)
  }
  return (0)
}
```

Цена на i-ом шаге из цен i-1 шага

```
Up <- function(dt){
  return (exp(sig * sqrt(dt)))
}

Down <- function(dt){
  return (exp(-sig * sqrt(dt)))
}

Prob <- function(dt){
  return ((exp(r * dt) - Down(dt))/(Up(dt) - Down(dt)))
}

V_cur <- function(V_pos, V_neg, dt){
  return(exp(-r * dt) * (Prob(dt) * V_pos + (1 - Prob(dt)) * V_neg))
}
```

Цены Европейских опционов с помощью биномиальной модели

```
get_stock_prices <- function(n_step, dt){
  ans = matrix(rep(0.0, n_step), n_step ^ 2, nrow = n_step, ncol = n_step)
  ans[1,1] = S
  for (i in 2:n_step){
    for (j in 1:i){
      ans[i,j] = S * (Down(dt) ^ (i - j)) * (Up(dt) ^ (j - 1))
    }
  }
  return (ans)
}

put_european_by_binomial_tree <- function(n_step, dt){
  prices = get_stock_prices(n_step, dt)
  ans = rep(0.0, n_step)
  for (i in 1:n_step){
    ans[i] = put_payoff_func(prices[n_step, i])
  }
  for (i in (n_step-1):1){
    for (j in 1:i){
      ans[j] = V_cur(ans[j+1], ans[j], dt)
    }
```

```
  }
  return (ans)
}

call_european_by_binomial_tree <- function(n_step, dt){
  prices = get_stock_prices(n_step, dt)
  ans = c()
  for (i in 1:n_step){
    ans = c(ans, call_payoff_func(prices[n_step, i]))
  }
  for (i in (n_step-1):1){
    for (j in 1:i){
      ans[j] = V_cur(ans[j+1], ans[j], dt)
    }
  }
  return (ans)
}


library('fOptions')
```

Цены Американского Put опциона с помощью биномиальной модели

```
put_american_by_binomial_tree <- function(n_step, dt){
  prices = get_stock_prices(n_step, dt)
  ans = c()
  for (i in 1:n_step){
    ans = c(ans, put_payoff_func(prices[n_step,i]))
  }
  for (i in (n_step-1):1){
    for (j in 1:i){
      ans[j] = max(V_cur(ans[j+1], ans[j], dt), put_payoff_func(prices[i,j]))
    }
  }
  return (ans)
}
```

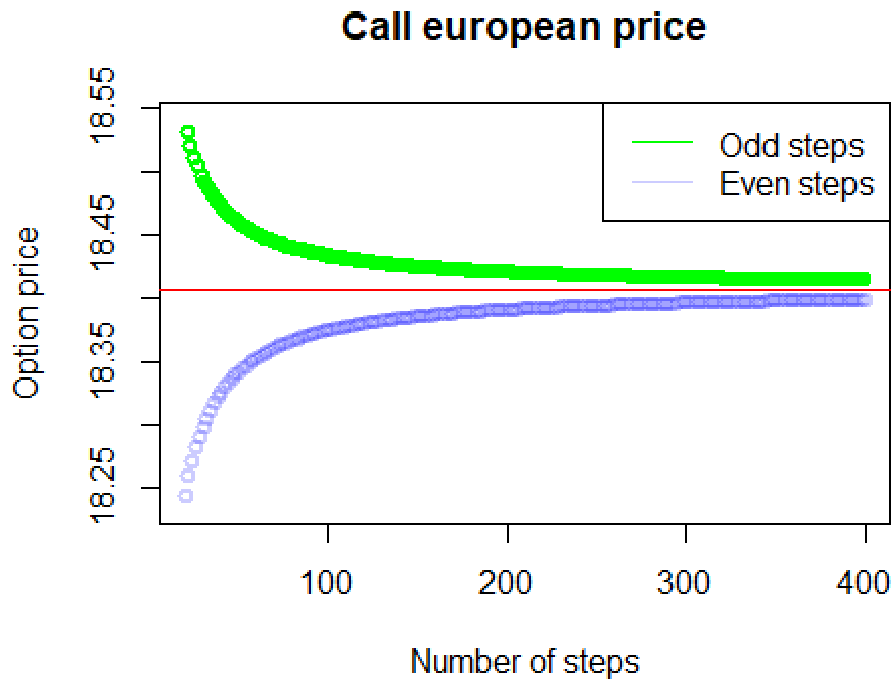
Проверить в цикле, насколько хорошо аппроксимируется модель для European Call опционов

```
N_arr = seq(20, 400)
call_european_approximation = c()
certain_solution_call_european = call_BS_func(S)
for (i in 1:length(N_arr)){
  N = N_arr[i]
  dt = T/N
  n_step = N + 1
  cur_appr = call_european_by_binomial_tree(n_step, dt)
  cur_appr = cur_appr[1]
  call_european_approximation <- c(call_european_approximation, cur_appr)
}
```

Визуализация для European Call опционов

```
N_odd = c()
N_even = c()
call_european_approximation_odd = c()
call_european_approximation_even = c()
for (i in 1:length(N_arr)){
  cur_n = N_arr[i]
  if (cur_n %% 2 == 1){
    N_odd <- c(N_odd, cur_n)
    call_european_approximation_odd <- c(call_european_approximation_odd, call_e
uropean_approximation[i])
  } else {
    N_even <- c(N_even, cur_n)
    call_european_approximation_even <- c(call_european_approximation_even, call
_european_approximation[i])
  }
}

y_start = call_european_approximation_even[1] - 0.01
y_finish = call_european_approximation_odd[1] + 0.01
plot(N_odd, call_european_approximation_odd, col=rgb(0.0,1.0,0.0,1.0), ylim = c(
y_start, y_finish),
     lwd=2.0, type = 'p', ylab = 'Option price', xlab='Number of steps', main='C
all european price')
lines(N_even, call_european_approximation_even, type = 'p', col=rgb(0.0,0.0,1.0,
0.2), lwd=2.0);
abline(h = certain_solution_call_european, col='red')
legend("topright", legend = c("Odd steps", "Even steps"),
       col = c(rgb(0.0,1.0,0.0,1.0), rgb(0.0,0.0,1.0,0.2)), lty=1:1)
```



Проверить в цикле, насколько хорошо аппроксимируется модель для European Put опционов

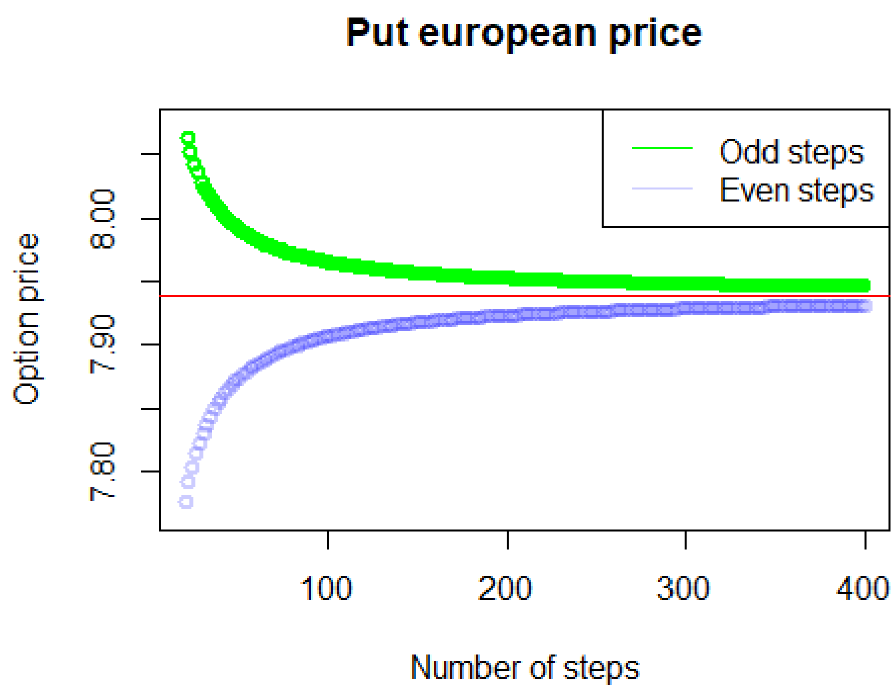
```
N_arr = seq(20, 400)
put_european_approximation = c()
certain_solution_put_european = put_BS_func(S)
for (i in 1:length(N_arr)){
  N = N_arr[i]
  dt = T/N
  n_step = N + 1
  cur_appr = put_european_by_binomial_tree(n_step, dt)
  cur_appr = cur_appr[1]
  put_european_approximation <- c(put_european_approximation, cur_appr)
}
```

Визуализация для European Put опционов

```
N_odd = c()
N_even = c()
put_european_approximation_odd = c()
put_european_approximation_even = c()
for (i in 1:length(N_arr)){
  cur_n = N_arr[i]
  if (cur_n %% 2 == 1){
    N_odd <- c(N_odd, cur_n)
    put_european_approximation_odd <- c(put_european_approximation_odd, put_euro
pean_approximation[i])
  } else {
    N_even <- c(N_even, cur_n)
    put_european_approximation_even <- c(put_european_approximation_even, put_eu
```

```
ropean_approximation[i])
  }
}

y_start = put_european_approximation_even[1] - 0.01
y_finish = put_european_approximation_odd[1] + 0.01
plot(N_odd, put_european_approximation_odd, col=rgb(0.0,1.0,0.0,1.0), ylim = c(y
_start, y_finish),
     lwd=2.0, type = 'p', ylab = 'Option price', xlab='Number of steps', main='P
ut european price')
lines(N_even, put_european_approximation_even, type = 'p', col=rgb(0.0,0.0,1.0,0
.2), lwd=2.0);
abline(h = certain_solution_put_european, col='red')
legend("topright", legend = c("Odd steps", "Even steps"),
       col = c(rgb(0.0,1.0,0.0,1.0), rgb(0.0,0.0,1.0,0.2)), lty=1:1)
```



Проверить в цикле, насколько хорошо аппроксимируется модель для European Put опционов и сравнить с приближением Barone-Adesi and Whaley

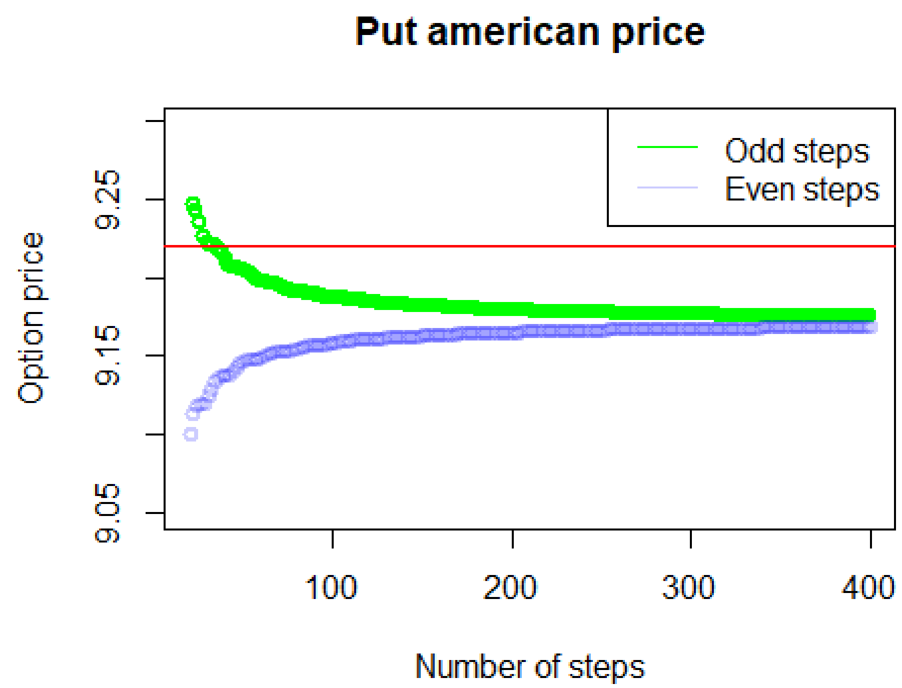
```
N_arr = seq(20, 400)
put_american_approximation = c()
certain_solution_put_american = BAWAmericanApproxOption('p', S, X, T, r, r, sig)
certain_solution_put_american = certain_solution_put_american@price
for (i in 1:length(N_arr)){
  N = N_arr[i]
  dt = T/N
  n_step = N + 1
  cur_appr = put_american_by_binomial_tree(n_step, dt)
  cur_appr = cur_appr[1]
```

```
  put_american_approximation <- c(put_american_approximation, cur_appr)
}
```

Визуализация для American Put опционов

```
N_odd = c()
N_even = c()
put_american_approximation_odd = c()
put_american_approximation_even = c()
for (i in 1:length(N_arr)){
  cur_n = N_arr[i]
  if (cur_n %% 2 == 1){
    N_odd <- c(N_odd, cur_n)
    put_american_approximation_odd <- c(put_american_approximation_odd, put_amer
ican_approximation[i])
  } else {
    N_even <- c(N_even, cur_n)
    put_american_approximation_even <- c(put_american_approximation_even, put_am
erican_approximation[i])
  }
}

y_start = put_american_approximation_even[1] - 0.05
y_finish = put_american_approximation_odd[1] + 0.05
plot(N_odd, put_american_approximation_odd, col=rgb(0.0,1.0,0.0,1.0), ylim = c(y
_start, y_finish),
     lwd=2.0, type = 'p', ylab = 'Option price', xlab='Number of steps', main='P
ut american price')
lines(N_even, put_american_approximation_even, type = 'p', col=rgb(0.0,0.0,1.0,0
.2), lwd=2.0);
abline(h = certain_solution_put_american, col='red')
legend("topright", legend = c("Odd steps", "Even steps"),
       col = c(rgb(0.0,1.0,0.0,1.0), rgb(0.0,0.0,1.0,0.2)), lty=1:1)
```



Поменять strike price на 97$ и построить график цен для Европейского Call опциона

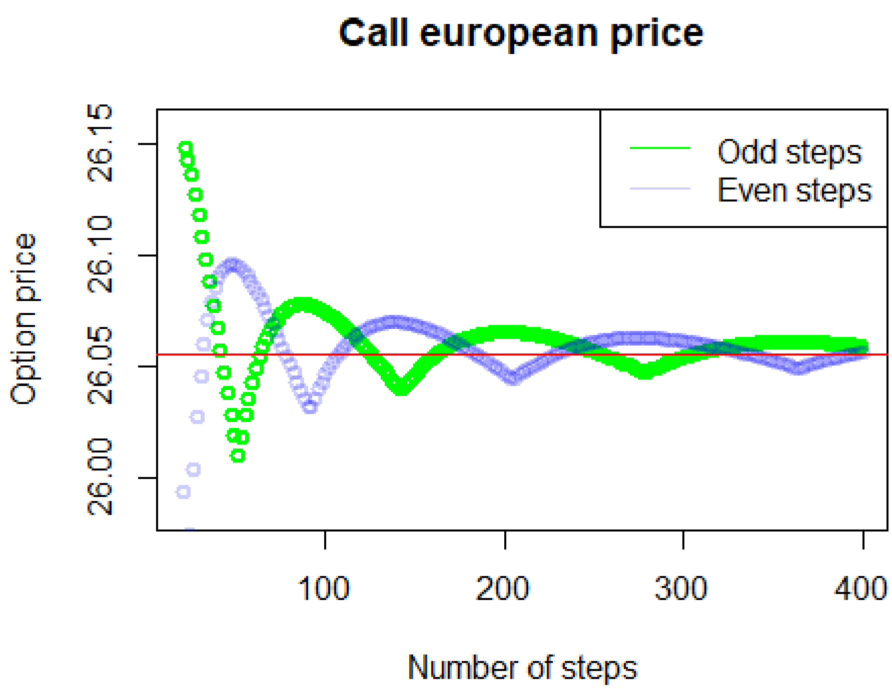
```
X = 97
N_arr = seq(20, 400)
call_european_approximation = c()
certain_solution_call_european = call_BS_func(S)
for (i in 1:length(N_arr)){
  N = N_arr[i]
  dt = T/N
  n_step = N + 1
  cur_appr = call_european_by_binomial_tree(n_step, dt)
  cur_appr = cur_appr[1]
  call_european_approximation <- c(call_european_approximation, cur_appr)
}
```

Визуализация для измененного European Call опциона

```
N_odd = c()
N_even = c()
call_european_approximation_odd = c()
call_european_approximation_even = c()
for (i in 1:length(N_arr)){
  cur_n = N_arr[i]
  if (cur_n %% 2 == 1){
    N_odd <- c(N_odd, cur_n)
    call_european_approximation_odd <- c(call_european_approximation_odd, call_e
uropean_approximation[i])
  } else {
    N_even <- c(N_even, cur_n)
    call_european_approximation_even <- c(call_european_approximation_even, call
```

```
_european_approximation[i])
  }
}

y_start = call_european_approximation_even[1] - 0.01
y_finish = call_european_approximation_odd[1] + 0.01
plot(N_odd, call_european_approximation_odd, col=rgb(0.0,1.0,0.0,1.0), ylim = c(
y_start, y_finish),
     lwd=2.0, type = 'p', ylab = 'Option price', xlab='Number of steps', main='C
all european price')
lines(N_even, call_european_approximation_even, type = 'p', col=rgb(0.0,0.0,1.0,
0.2), lwd=2.0);
abline(h = certain_solution_call_european, col='red')
legend("topright", legend = c("Odd steps", "Even steps"),
       col = c(rgb(0.0,1.0,0.0,1.0), rgb(0.0,0.0,1.0,0.2)), lty=1:1)
```



Выводы.

В данной работе рассматривалась биномиальная модель для поиска цен Европейских и Американских опционов. Биномиальная модель выглядит следующим образом:
1. Пусть текущая цена базового актива равна S. Предположим, что за dt времени цена может подняться до uS(u > 1) с вероятностью p или опуститься до dS(d < 1) с вероятностью (1-p).
2. Для выбора констант u, d, p воспользуемся следующим условием: у биномиального блуждания и логнормального блуждания должны быть равны среднее и волатильность. Также воспользуемся условием Cox-Ros-Rubenstein(CRR): ud = 1.
3. Двигаясь вперед по времени найдем цену базового актива через N шагов во время исполнения T.

4. Двигаясь обратно по времени, найдем цену опциона для предыдущего момента времени по формуле:

$$V = e^{-rt} * (V_+ * p + V_- * (1 - p))$$

Несмотря на то, что цену Европейских опционов можно аналитически найти с помощью уравнения Блэка - Шоулза, можно также воспользоваться и аппроксимацией в виде биномиальных деревьев. Поскольку для Американских Put опционов не существует аналитического решения, то решение для них можно приближенно найти с помощью биномиальной модели. В данной работе решение, полученное с помощью биномиальной модели для Американских Put опционов сравнивалось с аппроксимацией Barone-Adesi and Whaley.

Для всех видов опционов можно заметить, что при уменьшении шага по времени, дискретное решение сходится к аналитическому. Но в случае, если решение можно найти аналитически, лучше не искать его приближенно, поскольку дискретные модели вычислительно трудозатратны.

Также при изменении strike price для Европейского Call опциона изменился характер сходимости дискретного решения к точному. Поскольку цены европейских Call опционов в биномиальной модели задаются следующей формулой:

$$c = e^{-rT} * \sum_{i=0}^{N} C_n^i p^i (1 - p)^{n-i} max[Su^i d^{n-i} - X, 0]$$

, то уменьшение strike price увеличит число ненулевых слагаемых в модели, что скажется на изменении характера сходимости.