Übung 02

Freitag, 21. April 2023 10:32

CompFin_2023_SS_She...

Sheet: 02
Group: QF 27
Group members: [illegible], Robert Hennings

Christian-Albrechts-Universität zu Kiel
Mathematisches Seminar
Prof. Dr. Sören Christensen
Henrik Valett, Fan Yu, Oskar Hallmann

SS 2023

Sheet 02

Computational Finance

Exercises for all participants

Black-Scholes formula

The fair price of a European Call option with strike $K > 0$ in the Black-Scholes model at time $0 \le t \le T$ with stock price $S(t)$ is given by:

$$C(t, S(t), r, \sigma, T, K) = S(t)\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2).$$

Here, Φ denotes the cumulative distribution function of the standard normal distribution and

$$d_1 := \frac{\log\left(\frac{S(t)}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}, \quad d_2 := d_1 - \sigma\sqrt{T-t}.$$

The fair price of a European Put option with strike $K > 0$ in the Black-Scholes model at time $0 \le t \le T$ with stock price $S(t)$ is given by:

$$P(t, S(t), r, \sigma, T, K) = Ke^{-r(T-t)}\Phi(-d_2) - S(t)\Phi(-d_1).$$

Here, Φ denotes the cumulative distribution function of the standard normal distribution and

$$d_1 := \frac{\log\left(\frac{S(t)}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}, \quad d_2 := d_1 - \sigma\sqrt{T-t}.$$

C-Exercise 04 (4 points)
We want to price call options in the CRR-model.

(a) Write a Python function

```
V_0 = CRR_EuCall (S_0, r, sigma, T, M, K)
```

that computes and returns an approximation to the price of an European call option with strike $K > 0$ and maturity $T > 0$ in the Black-Scholes model with initial stock price $S(0) > 0$, interest rate $r > 0$ and volatility $\sigma > 0$ using the CRR-model as presented in the course and $M \in \mathbb{N}$ time steps.
Hint: Use the results from C-Exercise 01 and the process from T-Exercise 05.

(b) We want to compare the CRR model to the true price in the BS-model. To this end implement the BS-Formula for European call options as a Python function:

```
V_0 = BlackScholes_EuCall (t, S_t, r, sigma, T, K)
```

(c) Compare the results by plotting the error of the CRR model against the BS-price in a common graph. Use the following parameters

$S(0) = 100,\ r = 0.03,\ \sigma = 0.3,\ T = 1,\ M = 100,\ K = 70, \ldots, 200.$

T-Exercise 05 (4 points)
We want to price an American put option with strike price $K = 1.2$ and time to maturity being three years. For this purpose we want to utilize a CRR model with $M = 3$ equally spaced time periods, $S(0) = 1$, $\sigma^2 = 0.3$ and an annual interest rate of 5%.

a) Draw and calculate the corresponding CRR model by hand (of course you can still use a calculator) and write beneath each point the corresponding price of the option (please round on four position after the decimal point after each calculation).

b) Calculate the replicating portfolio $\varphi = (\varphi_0, \varphi_1)$ for all time periods.

C-Exercise 06 (Options in the CRR model) (4 points)

(a) Write a Python function

```
V_0 = CRR_AmEuPut (S_0, r, sigma, T, M, K, EU)
```

that computes and returns an approximation to the price of a European or an American put option with strike $K > 0$ and maturity $T > 0$ in the CRR model with initial stock price $S(0) > 0$, interest rate $r > 0$ and volatility $\sigma > 0$. The parameter "EU" is 1 if the price of an European put shall be computed or is 0 in the American case. Use the binomial method as presented in the course with $M \in \mathbb{N}$ time steps.

(b) As $M \to \infty$ we would expect convergence of the price in the binomial model towards the price in the Black-Scholes model. To show this implement the BS-Formula for European put options as a Python function:

```
V_0 = BlackScholes_EuPut (t, S_t, r, sigma, T, K)
```

(c) Test your algorithm with

$S(0) = 100,\ r = 0.05,\ \sigma = 0.3,\ T = 1,\ M = 10, \ldots, 500,\ K = 120.$

- Plot the price of a European put option (EU = 1) in the binomial model in dependence on the number of steps M.
- Plot the fair price in the BS-model into the same plot using the same parameters.
- Print the price of an American put option (EU = 0) with the same $S(0), r, \sigma, T, K$ as above and $M = 500$ steps in the console.

Useful Python commands: numpy.maximum

T-Exercise 07 (Barrier options in the CRR model) (for math 4 points; for QF 4 bonuspoints)
In the binomial model from Section 2.1 with parameters $S(0), r, \sigma, T > 0$ and $M \in \mathbb{N}$, we denote by

- V the fair price process of a *European call option* on the stock S with strike $K > 0$, i.e. its payoff is given by $V(T) = (S(T) - K)^+$,
- $\tilde{V}$ the fair price process of a *down-and-out call option* on the stock S with strike $K > 0$ and barrier $B < K$, i.e. its payoff is given by

$$\tilde{V}(T) = 1_{\{S(t_i) > B \text{ for all } i = 0, \ldots, M\}}(S(T) - K)^+,$$

- $\hat{V}$ the fair price process of a *down-and-in call option* on the stock S with strike $K > 0$ and barrier $B < K$, i.e. its payoff is given by

$$\hat{V}(T) = 1_{\{S(t_i) \le B \text{ for one } i = 0, \ldots, M\}}(S(T) - K)^+.$$

Outline (e.g. in pseudo code) an algorithm that computes the initial price $\tilde{V}(0), \hat{V}(0)$ of the barrier options in $O(M^2)$ steps, i.e. there is a constant $C > 0$ independent of M such that the algorithm terminates after less than CM^2 operations. Please explain where your algorithm differs from the algorithm for European call options presented in the lecture and why these changes make sense/are needed.

Hint:

(a) Start with the *down-and-out call option*.

(b) Express the *down-and-in call option* in terms of a *down-and-out call option*.

If we have price of one we can obtain the other one!
Down below → nothing → stopping time like American Option!
→ [illegible] put price then to 0!
→ how many steps?

Please include your name(s) as comment in the beginning of the file.
Do not forget to include comments in your Python-programs.
Submit until: Fri, 28.04.2022, 10:00

T-Exercise 05

a) Draw + calculate CRR tree

Inputs:

American Put Option

$M = 3$

$S(0) = 1$

$\sigma^2 = 0{,}3 \qquad \sigma = \sqrt{0{,}3}$

$r = 0{,}05$

$K = 1{,}2$

$T = 3$

Follow the algorithm:

1) $\Delta t = \frac{T}{M}$

$\Delta t = \frac{3}{3} = 1$

2) $d = \beta - \sqrt{\beta^2 - 1}$ / $d = \frac{1}{u}$

$u = \beta + \sqrt{\beta^2 - 1}$

$\beta = \frac{1}{2} \cdot \left[e^{-r \cdot \Delta t} + e^{(r+\sigma^2)\Delta t}\right]$

$\beta = \frac{1}{2} \cdot \left[e^{-0{,}05 \cdot 1} + e^{(0{,}05 + 0{,}3) \cdot 1}\right]$

$\beta = \frac{1}{2} \cdot \left[e^{-0{,}05} + e^{0{,}35}\right] = 1{,}1851$ ✓

$u = 1{,}1851 + \sqrt{1{,}1851^2 - 1}$

$u = 1{,}8211$

$d = \frac{1}{u} = \frac{1}{1{,}8211} = 0{,}5491$

$q = \frac{e^{r \cdot \Delta t} - d}{u - d}$

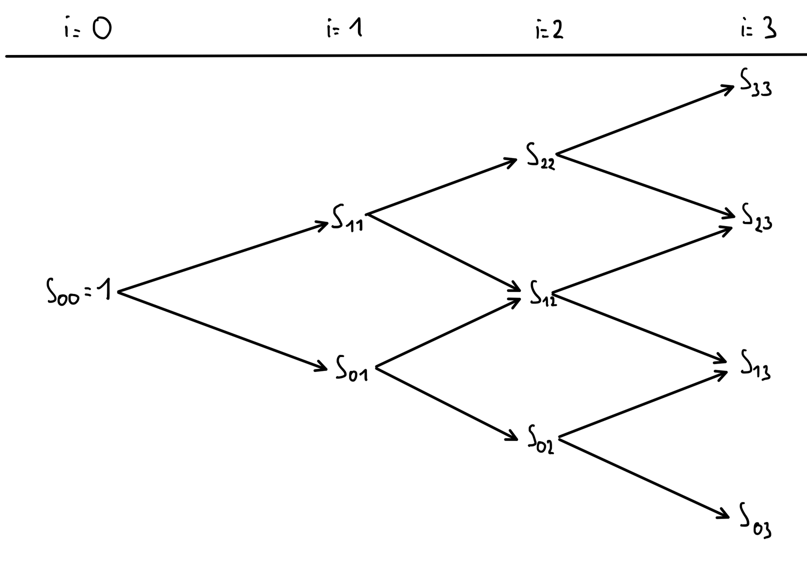
$q = \frac{e^{0{,}05 \cdot 1} - 0{,}5491}{(1{,}8211 - 0{,}5491)}$

$q = 0{,}3948$

$(1-q) = 0{,}6052$

3) $S_{00} = 1$

4) Calculate the tree with $\Delta t = 1$; $S_{ji} = S_{00} \cdot u^j \cdot d^{i-j}$



From section 1.5)

Algorithm to compute Option prices (European + American) in the BSM model based on the binomial model approximation:

Needed Input parameters: $r, \sigma, S(0), T, K$, Call / Put, European / American, M

Single computation steps:

1) $\Delta t := \frac{T}{M}$ number of time increments "continuousization"

2) Set u, d, q according to 1.4 - 1.7

$d = \beta - \sqrt{\beta^2 - 1}$

$u = \beta + \sqrt{\beta^2 - 1}$

$q = \frac{e^{r \cdot \Delta t} - d}{u - d}$

$\beta = \frac{1}{2} \cdot \left[e^{-r \cdot \Delta t} + e^{(r+\sigma^2)\Delta t}\right]$

3) Set $S_{00} := S(0)$

4) $S_{ji} = S_{00}\, u^j \cdot d^{i-j}$ for $i = 1, \ldots, M$ and $j = 0, \ldots, i$

j upward, $i-j$ downs, i: period

5) V_{jM} according to

$$V_{jM} = \begin{cases} (S_{jM} - K)^+ & \text{for a call} \\ (K - S_{jM})^+ & \text{for a put} \end{cases}$$

6) V_{ji} according to

In the European case then: $V_{ji} = e^{-r \cdot \Delta t} \cdot \left[q \cdot V_{j+1,i+1} + (1-q) \cdot V_{j,i+1}\right]$

In the American case then:

$$V_{ji} = \begin{cases} \max\left\{(S_{ji} - K)^+,\ e^{-r \cdot \Delta t} \cdot \left(q \cdot V_{j+1,i+1} + (1-q) \cdot V_{j,i+1}\right)\right\} & \text{for a call} \\ \max\left\{(K - S_{ji})^+,\ e^{-r \cdot \Delta t} \cdot \left(q \cdot V_{j+1,i+1} + (1-q) \cdot V_{j,i+1}\right)\right\} & \text{for a put} \end{cases}$$

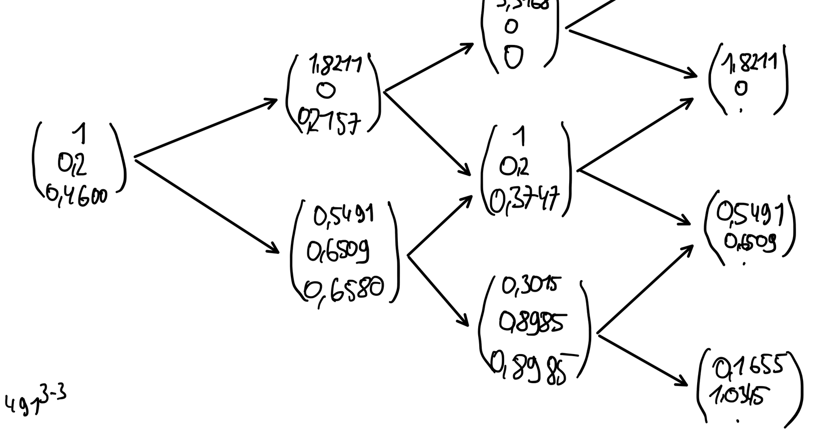
for $i = M-1, \ldots, 0$ and $j = 0, \ldots, i$

7) Output V_{00}

↳ desired approximation for initial option price in the Black Scholes model

i = 0 i = 1 i = 2 i = 3

$$\begin{pmatrix} S_{ji} \\ V_{jM} \\ V_{ji} \end{pmatrix}$$



Stock price evolution:

i = 1

j = 1
$S_{11} = 1 \cdot 1{,}8211^1 \cdot 0{,}5491^{1-1}$
$S_{11} = 1{,}8211$

j = 0
$S_{01} = 1 \cdot 1{,}8211^0 \cdot 0{,}5491^{1-0}$
$S_{01} = 0{,}5491$

i = 2

j = 2
$S_{22} = 1 \cdot 1{,}8211^2 \cdot 0{,}5491^{2-2}$
$S_{22} = 3{,}3168$

j = 1
$S_{12} = 1 \cdot 1{,}8211^1 \cdot 0{,}5491^{2-1}$
$S_{12} = 1$

j = 0
$S_{02} = 1 \cdot 1{,}8211^0 \cdot 0{,}5491^2$
$S_{02} = 0{,}3015$

i = 3

j = 3
$S_{33} = 1 \cdot 1{,}8211^3 \cdot 0{,}5491^{3-3}$
$S_{33} = 6{,}0406$

j = 2
$S_{23} = 1 \cdot 1{,}8211^2 \cdot 0{,}5491^{3-2}$
$S_{23} = 1{,}8211$

j = 1
$S_{13} = 1 \cdot 1{,}8211^1 \cdot 0{,}5491^{3-1}$
$S_{13} = 0{,}5491$

j = 0
$S_{03} = 1 \cdot 1{,}8211^0 \cdot 0{,}5491^{3-0}$
$S_{03} = 0{,}1655$

5) Compute the value process:

V_{jM} according to (j upward moves) $i, \ldots, M$

$$V_{jM} = \begin{cases} (S_{jM} - K)^+ & \text{for a call} \\ (K - S_{jM})^+ & \text{for a put} \end{cases} \qquad K = 1{,}2$$

i = 00

j = 0
$V_{00} = (K - S_{00})^+$
$= (1{,}2 - 1)$
$V_{00} = 0{,}2$

i = 1

j = 1
$V_{11} = (K - S_{11})^+$
$V_{11} = (1{,}2 - 1{,}8211) = 0$

j = 0
$V_{01} = (1{,}2 - 0{,}5491)^+$
$V_{01} = 0{,}6509$

i = 2

j = 2
$V_{22} = (1{,}2 - 3{,}3168)^+$
$V_{22} = 0$

j = 1
$V_{12} = (1{,}2 - 1)^+$
$V_{12} = 0{,}2$

j = 0
$V_{02} = (1{,}2 - 0{,}3015)^+$
$V_{02} = 0{,}8985$

i = 3

j = 3
$V_{33} = (1{,}2 - 6{,}0406)^+$
$V_{33} = 0$

j = 2
$V_{23} = (1{,}2 - 1{,}8211)^+$
$V_{23} = 0$

j = 1
$V_{13} = (1{,}2 - 0{,}5491)^+$
$V_{13} = 0{,}6509$

j = 0
$V_{03} = (1{,}2 - 0{,}1655)^+$
$V_{03} = 1{,}0345$

6) Compute further, based on the Option type,

V_{ji} according to

In the European case then: $V_{ji} = e^{-r \cdot \Delta t} \cdot \left[q \cdot V_{j+1,i+1} + (1-q) \cdot V_{j,i+1}\right]$

In the American case then:

$$V_{ji} = \begin{cases} \max\left\{(S_{ji} - K)^+,\ e^{-r \cdot \Delta t} \cdot \left(q \cdot V_{j+1,i+1} + (1-q) \cdot V_{j,i+1}\right)\right\} & \text{for a call} \\ \max\left\{(K - S_{ji})^+,\ e^{-r \cdot \Delta t} \cdot \left(q \cdot V_{j+1,i+1} + (1-q) \cdot V_{j,i+1}\right)\right\} & \text{for a put} \end{cases}$$

for $i = M-1, \ldots, 0$ and $j = 0, \ldots, i$ $\quad M = 3 \quad i = 2, 1, 0$

$q = 0{,}3948$

$(1-q) = 0{,}6052$

i = 2

$\boxed{j = 2}$

$$V_{22} = \max\begin{cases} e^{-0{,}05} \cdot \left[0{,}3948 \cdot 0 + 0{,}6052 \cdot 0\right] \\ (K - S_{ji})^+ \quad (1{,}2 - S_{22}) \end{cases}$$

$$V_{22} = \max\begin{cases} 0 \\ 0 \end{cases}$$

$\boxed{j = 1}$

$$V_{12} = \max\begin{cases} e^{-0{,}05} \cdot \left[0{,}3948 \cdot 0 + 0{,}6052 \cdot 0{,}6509\right] \\ (K - S_{ji})^+ \quad (1{,}2 - S_{12}) \end{cases}$$

$$V_{12} = \max\begin{cases} 0{,}3747 \\ 0{,}2 \end{cases}$$

$\boxed{j = 0}$

$$V_{02} = \max\begin{cases} e^{-0{,}05} \cdot \left[0{,}3948 \cdot 0{,}6509 + 0{,}6052 \cdot 1{,}0345\right] \\ (K - S_{ji})^+ \quad (1{,}2 - S_{02}) \end{cases}$$

$$V_{02} = \max\begin{cases} 0{,}84 \\ 0{,}8985 \end{cases}$$

i = 1

$\boxed{j = 1}$

$$V_{11} = \max\begin{cases} e^{-0{,}05} \cdot \left[0{,}3948 \cdot 0 + 0{,}6052 \cdot 0{,}3747\right] \\ (K - S_{ji})^+ \quad (1{,}2 - S_{11}) \end{cases}$$

$$V_{11} = \max\begin{cases} 0{,}2157 \\ 0 \end{cases}$$

$\boxed{j = 0}$

$$V_{01} = \max\begin{cases} e^{-0{,}05} \cdot \left[0{,}3948 \cdot 0{,}3747 + 0{,}6052 \cdot 0{,}8985\right] \\ (K - S_{ji})^+ \quad (1{,}2 - S_{01}) \end{cases}$$

$$V_{01} = \max\begin{cases} 0{,}6580 \\ 0{,}6509 \end{cases}$$

i = 0

$\boxed{j = 0}$

$$V_{00} = \max\begin{cases} e^{-0{,}05} \cdot \left[q \cdot V_{1,1} + (1-q) \cdot V_{0,1}\right] \\ (K - S_{00})^+ \end{cases}$$

$$V_{00} = \max\begin{cases} e^{-0{,}05} \cdot \left[0{,}3948 \cdot 0{,}2157 + 0{,}6052 \cdot 0{,}6580\right] \\ (K - S_{ji})^+ = (1{,}2 - 1)^+ \end{cases}$$

$$V_{00} = \max\begin{cases} 0{,}4600 \\ 0{,}2 \end{cases}$$

7)

$V_{00} = 0$

b) calculating the hedge / self-financing portfolio

1.12) $B(t_i) = e^{r \cdot t_i}$

$$\varphi_1(t_i) = \frac{V(S(t_{i-1}) \cdot u, t_i) - V(S(t_{i-1}) \cdot d, t_i)}{S(t_{i-1}) \cdot (u-d)}$$

1.13)

$$\varphi_0(t_i) = \frac{V(S(t_{i-1}), t_{i-1}) - \varphi_1(t_i) \cdot S(t_{i-1})}{B(t_{i-1})}$$

$u = 1{,}8211$

$d = \frac{1}{u} = \frac{1}{1{,}8211} = 0{,}5491$

$t_{i=1}$

$V_{11} = 0{,}2157$

$V_{01} = 0{,}6580$

$S(t_0) = 1$

$t_{i=2}$

$V_{22} = 0$

$V_{12} = 0{,}3747$ } upper path

$V_{02} = 0{,}8985$ } lower path

$S(t_1) = \begin{cases} S_{11} = 1{,}8211 \\ S_{01} = 0{,}5491 \end{cases}$

→ value process depends on payoff function!

$$\varphi_1(t_{i=1}) = \frac{V(S(t_{1-1}) \cdot u, t_{i=1}) - V(S(t_{1-1}) \cdot u, t_{i=1})}{S(t_{1-1}) \cdot (u-d)}$$

$$\varphi_1(t_{i=1}) = \frac{V(S(t_0) \cdot u, t_{i=1}) - V(S(t_0) \cdot d, t_{i=1})}{S(t_0) \cdot (u-d)}$$

$$\varphi_1(t_{i=1}) = \frac{V_{11} - V_{01}}{1 \cdot (1{,}8211 - 0{,}5491)} = \frac{0{,}2157 - 0{,}6580}{1{,}2720}$$

$\varphi_1(t_{i=1}) = -0{,}3477$

$$\varphi_0(t_{i=1}) = \frac{V(S(t_{1-1}), t_{1-1}) - \varphi_1(t_1) \cdot S(t_{1-1})}{B(t_{1-1})}$$

$$= \frac{V(S(t_0), t_0) - \varphi_1(t_1) \cdot S(t_0)}{B(t_0)}$$

$$= \frac{0{,}46 - (-0{,}3477) \cdot 1}{e^{0{,}05}}$$

$\varphi_0(t_{i=1}) = 0{,}7683$

Upper path

$$\varphi_1(t_{i=2}) = \frac{V(S(t_{2-1}) \cdot u, t_{i=2}) - V(S(t_{2-1}) \cdot d, t_{i=2})}{S(t_{2-1}) \cdot (u-d)}$$

$$\varphi_1(t_{i=2}) = \frac{V(S(t_1) \cdot u, t_{i=2}) - V(S(t_1) \cdot d, t_{i=2})}{S(t_1) \cdot (u-d)}$$

$$\varphi_1(t_{i=2}) = \frac{0 - 0{,}3747}{1{,}8211 \cdot (1{,}2720)}$$

$\varphi_1(t_{i=2}) = -1{,}6176$

$$\varphi_0(t_{i=2}) = \frac{V(S(t_1), t_1) - \varphi_1(t_2) \cdot S(t_1)}{e^{0{,}05 \cdot 2}}$$

$$= \frac{0{,}2157 - (-1{,}6176) \cdot 1{,}8211}{e^{0{,}05 \cdot 2}}$$

$\varphi_0(t_{i=2}) = 2{,}8607$

Lower path

$$\varphi_1(t_{i=2}) = \frac{0{,}3747 - 0{,}8985}{0{,}5491 \cdot (1{,}2720)}$$

$\varphi_1(t_{i=2}) = -0{,}7499$

$$\varphi_0(t_{i=2}) = \frac{0{,}6580 - (-0{,}7499) \cdot 0{,}5491}{e^{0{,}05 \cdot 2}}$$

$\varphi_0(t_{i=2}) = 0{,}9680$

T-Exercise 07