Christian-Albrechts-Universität zu Kiel **SS 2017**
Mathematisches Seminar
Prof. Dr. Mathias Vetter
Ole Martin, Adrian Theopold Sheet 04

Computational Finance

Exercises for participants of the programme **'Quantative Finance'**

C-Exercise 11 (Greeks of a European option in the Black-Scholes model)
On the OLAT you find a scilab function

```
V0 = BS_Price_Int (r, sigma, S0, T, g)
```

which computes the price of a European option with payoff $g(S(T))$ at maturity $T > 0$ in a Black-Scholes model with initial stock price $S(0) > 0$, interest rate $r > 0$ and volatility $\sigma > 0$. The first order greeks for a European option in the Black-Scholes model are given by the first order derivatives

$$\begin{aligned}
\Delta(r,\sigma,S(0),T,g) &= \frac{\partial}{\partial S(0)} V_{BS}(r,\sigma,S(0),T,g),\\
\nu(r,\sigma,S(0),T,g) &= \frac{\partial}{\partial \sigma} V_{BS}(r,\sigma,S(0),T,g),\\
\rho(r,\sigma,S(0),T,g) &= \frac{\partial}{\partial r} V_{BS}(r,\sigma,S(0),T,g),\\
\Theta(r,\sigma,S(0),T,g) &= -\frac{\partial}{\partial T} V_{BS}(r,\sigma,S(0),T,g),
\end{aligned}$$

where $V_{BS}(r,\sigma,S(0),T,g)$ denotes the Black-Scholes price of the European option.

a) Write a scilab function

```
[Delta,vega,rho,Theta]=BS_Greeks_num(r, sigma, S0, T, g ,eps)
```

that computes the greeks described above numerically using the function `BS_Price_Int` and the approximation

$$\frac{\partial}{\partial x} f(x,y) \approx \frac{f(x+\varepsilon x,y)-f(x,y))}{\varepsilon x}.$$

b) Plot $\Delta(r,\sigma,S(0),T,g)$ for the European put with payoff function $g(x) = (100-x)^+$ and parameters $r = 0.05$, $\sigma = 0.2$, $T = 1$ for $S(0) \in [60,140]$. Use $\varepsilon = 0.001$.

Useful scilab command: `exec`

C-Exercise 12 (Barrier option in the CRR model)
In the binomial model from Section 2.1 with parameters $S(0)$, r, σ, $T > 0$ and $M \in \mathbb{N}$, we denote by V the fair price process of an *up-and-out put option* on the stock S with strike $K > 0$ and barrier $B > K$. I.e., its payoff is given by

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

(a) Explain which line in the algorithm from C-Exercise 06 has to be changed and why.

(b) Implement the change and write a scilab function

```
V0 = UpOutPut_BinMod (S_0, r, sigma, T, K, B, M)
```

that computes and returns the fair value at time $t_0 = 0$ of the up-and-out put option. Test your function with

$$S(0) = 100,\ r = 0.05,\ \sigma = 0.2,\ T = 1,\ K = 100,\ B = 110,\ M = 1000.$$

T-Exercise 13
For $\mu \in R$ and $\sigma, r > 0$ we consider the Black-Scholes market with bond B and stock price process S which evolve according to

$$\begin{aligned} dB_t &= rB_t dt, & B_0 &= 1, \\ dS_t &= \mu S_t dt + \sigma S_t dW_t, & S_0 &> 0. \end{aligned}$$

(a) Calculate the Itô process representation of the logarithmic stock process $X_t := \log(S_t)$ and the associated quadratic variation process $[X,X]_t$.

(b) Consider a self-financing portfolio $\varphi = (\varphi_t^0, \varphi_t^1)_{t\geq 0}$ with initial value $V_0(\varphi) = 1$ that always invests half of the wealth into the stock, i.e. $\varphi_t^1 = \frac{V_t(\varphi)}{2S_t}$. Show that the value process $V_t(\varphi)$ is a geometric Brownian motion.

Please save your solution of each C-Exercise in a file named `Exercise_##.sce`, where ## denotes the number of the exercise. Please include your name(s) as comment in the beginning of the file.

Submit until: Fri, 19.05.2017, 10:00
Discussion: 22./24.05.2017,