Christian-Albrechts-UniversitÃ¤t zu Kiel **SS 2024**
Mathematisches Seminar
Prof. Dr. Mathias Vetter
Henrik Valett, Fan Yu, Ivo Richert, Anton Schellin Sheet 06

Computational Finance

Exercises for all participants

C-Exercise 21 (Greeks of a European option in the Black-Scholes model) (4 points)
In the 'Material' folder of the OLAT you find a python 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 $\sigma > 0$. (This is formula (3.21) from the lecture notes)
The first order greeks for a European option in the Black-Scholes model are given by the first order derivatives

$$\Delta(r,\sigma,S(0),T,g) = \frac{\partial}{\partial S(0)} V_{BS}(r,\sigma,S(0),T,g),$$
$$\nu(r,\sigma,S(0),T,g) = \frac{\partial}{\partial \sigma} V_{BS}(r,\sigma,S(0),T,g),$$
$$\gamma(r,\sigma,S(0),T,g) = \frac{\partial^2}{\partial S(0)\partial S(0)} V_{BS}(r,\sigma,S(0),T,g),$$

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

a) Write a Python function

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

that computes the greeks described above numerically using the approximations

$$\frac{\partial}{\partial x} f(x,y) \approx \frac{f(x+\varepsilon x, y) - f(x,y)}{\varepsilon x},$$
$$\frac{\partial^2}{\partial x \partial x} f(x,y) \approx \frac{f(x+\varepsilon x, y) - 2f(x,y) + f(x-\varepsilon x, y))}{(\varepsilon x)^2}.$$

For this you can use the function `BS_Price_Int`.

b) Plot $\Delta(r,\sigma,S(0),T,g)$ for the European call with payoff function $g(x) = (x-110)^+$ and parameters $r = 0.05$, $\sigma = 0.3$, $T = 1$ for $S(0) \in [60,140]$. Use $\varepsilon = 0.001$.

T-Exercise 22QF (Self-quanto call) (for QF only) (4 points)
The self-quanto call with strike K is an option paying $(S(T)-K)^+$ shares of stock at time T, which means that its payoff equals $(S(T)-K)^+S(T)$. Determine the integral transform representation of the fair option price, i.e. the function $\widetilde{f}$ in equation (4.5) of the lecture notes.

T-Exercise 22Math (Laplace transform approach for digital call) (for math only) (4 points)

a) Compute the Laplace transform for the payoff function $g(x) = 1_{\{x \geq \log(K)\}}$, $K \in \mathbb{R}$ of a digital call option and determine its domain of convergence.

b) Use the results from part a) to determine the fair value $V_g(0) = e^{-rT}\mathbb{E}_Q[g(S_T)]$ of the digital call in the Black-Scholes model.

C-Exercise 23 (4 points)
Write a Python function

```
V0 = BS_EuCall_Laplace (S0, r, sigma, T, K, R)
```

that computes the initial price of a European call option in the Black-Scholes model via the Laplace transform approach. I.e., implement the formula

$$V(t) = \frac{e^{-r(T-t)}}{\pi} \int_0^\infty \mathrm{Re}\left(\tilde{f}(R+iu)\chi_t(u-iR)\right) \mathrm{d}u$$

from the course. Choose an appropriate R and test your function for

$$S(0) = [50:150], \quad r = 0.03, \quad \sigma = 0.2, \quad T = 1, \quad K = 110$$

Plot your results in a common graph.

Useful Python commands: `cmath.exp`, `complex`, `scipy.integrate.quad`, `real`

C-Exercise 24 (4 points)
A perpetual American put option is an option contract without expiry that yields the payoff $g(S(t)) = (K - S(t))^+$ at any future time chosen by the holder of the option. Let $x^* = \frac{2Kr}{2r+\sigma^2}$. One can show that the value function $v(S(t))$ of the perpetual American put option in the Black–Scholes model is independent of time and satisfies the free boundary problem

$$\frac{\sigma^2}{2}x^2\frac{\partial^2}{\partial x^2}v(x) + rx\frac{\partial}{\partial x}v(x) - rv(x) = 0 \quad \text{for } x \geq x^*,$$
$$v(x) = g(x) \quad \text{for } x \leq x^*.$$

Write a Python function `V0 = BS_AmPerpPut_ODE(S_max, N, r, sigma, K)` that computes the price of the perpetual American put option in the Black–Scholes model on the equidistant grid $(0, \frac{S_{\max}}{N}, \frac{2S_{\max}}{N}, \ldots, S_{\max})$ of stock prices by numerically solving the above ordinary differential equation. Plot the option value against the stock price grid using

$$S_{\max} = 200, \quad N = 200, \quad r = 0.05, \quad \sigma^2 = 0.4, \quad K = 100.$$

Useful Python commands: `scipy.integrate.solve_ivp`, `scipy.integrate.ode`, `scipy.integrate.odeint`

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: Thu, 13.06.2023, 12:00