Christian-Albrechts-Universität zu Kiel **SS 2023**
Mathematisches Seminar
Prof. Dr. Sören Christensen
Henrik Valett, Fan Yu, Oskar Hallmann, Nele Rothert Sheet 09

Computational Finance

Exercises for all participants

T-Exercise 32 (Transforming the Black-Scholes PDE to the heat equation) (4 points)
Let $v(t,x)$ denote the solution of the Black-Scholes PDE (3.26). Consider the variables $\tilde{x} = \log(x/K)$, $\tilde{t} = \sigma^2(T-t)/2$, $q = 2r/\sigma^2$ and define the function $y(\tilde{t},\tilde{x})$ via

$$y(\tilde{t},\tilde{x}) = v\Big(T - \frac{2\tilde{t}}{\sigma^2}, K\exp(\tilde{x})\Big)K^{-1}\exp\left(\frac{1}{2}(q-1)\tilde{x} + \Big(\frac{1}{4}(q-1)^2 + q\Big)\tilde{t}\right).$$

Prove that $y(\tilde{t},\tilde{x})$ solves the heat equation

$$\frac{\partial y}{\partial \tilde{t}} = \frac{\partial^2 y}{\partial \tilde{x}^2}$$

and that the initial conditions (6.2) resemble the terminal conditions for the European call and the European put, respectively.

C-Exercise 33 (Valuation of a European Call using the explicit finite difference scheme) (4 points)
Write a Python function

```
V0 = BS_EuCall_FiDi_Explicit (r, sigma, a, b, m, nu_max, T, K)
```

that approximates the option values $v(0,x_1),\dots,v(0,x_{m-1})$ of a European call option with strike $K > 0$ and maturity $T > 0$ in the Black-Scholes model using the explicit finite difference scheme. Here, $x_i = K\exp(a + i\frac{b-a}{m})$ denote the initial stock prices and a, b, m, ν_{max} are the parameters of the algorithm presented in the course. Test your function for

$$r = 0.05, \quad \sigma = 0.2, \quad a = -0.7, \quad b = 0.4, \quad m = 100, \quad \nu_{max} = 2000,$$
$$T = 1, \quad K = 100.$$

Compare your result with the exact solution using the BS-formula by plotting the difference between the finite difference approximation and the exact option price for all underlying initial stock prices.

C-Exercise 34 (Inversion of tridiagonal matrices) (4 points)
Write a Python function

```
x = TriagMatrix_Inversion(alpha,beta,gamma,b)
```

that calculates the solution $x \in \mathbb{R}^n$ of the equation $Ax = b$ for tridiagonal $A \in \mathbb{R}^{n\times n}$, i.e.

$$\begin{pmatrix} \alpha_1 & \beta_1 & 0 & \dots & 0 \\ \gamma_2 & \alpha_2 & \beta_2 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \gamma_{n-1} & \alpha_{n-1} & \beta_{n-1} \\ 0 & \dots & 0 & \gamma_n & \alpha_n \end{pmatrix} \begin{pmatrix} x_1 \\ \\ \vdots \\ \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \\ \vdots \\ \\ b_n \end{pmatrix},$$

with $\alpha, \beta, \gamma, b \in \mathbb{R}^n$ according to section 6.2 in the lecture notes. That means your function should take n-dimensional vectors as an input but γ_1 and β_n will be irrelevant to your algorithm. Do not use prefabricated python solutions for your function (like `numpy.linalg.solve`) but write the code explicitly according to section 6.2.
Test your function by solving the linear equation system

$$\begin{aligned} x_1 + 3x_2 + x_3 &= 1, \\ x_1 + 2x_2 &= 3, \\ x_2 + 2x_3 &= 3. \end{aligned}$$

Hint: If you are unsure about your results you may use `numpy.linalg.solve` (or calculate the solution by hand) to confirm your algorithm.

T-Exercise 35 (for math only) (4 points)
Let $A \in \mathbb{R}^{d\times d}$ be a symmetric matrix. Show that the following properties are equivalent:

(a) $\lim_{\nu\to\infty} A^\nu z = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$ for any $z \in \mathbb{R}^d$.

(b) $\lim_{\nu\to\infty} (A^\nu)_{ij} = 0$ for any $i, j \in 1, \dots, d$.

(c) The spectral radius of A satisfies $\rho(A) < 1$.

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, 23.06.2023, 10:00