

Computational Finance

Exercises for participants of mathematical programmes

C-Exercise 01

Write a scilab function

```
Vn = interest_yield (V0, r, n, c)
```

that computes and returns the capital V_n if an interest of $r > 0$ has been paid on the initial endowment $V_0 > 0$ for $n \in \mathbb{N}$ years. If $c = 0$, the variable r refers to a simple rate (i.e., $V_n = V_0(1 + r)^n$), and if $c = 1$, it refers to a continuous rate (i.e., $V_n = V_0e^{rn}$). Test your function for

$$V_0 = 1000, \quad r = 0.03, \quad n = 10, \quad c = 1.$$

Useful scilab commands: if, elseif, else, exp

C-Exercise 02

We are given two vectors $x = (x_1, \dots, x_n)^\top$ and $y = (y_1, \dots, y_n)^\top$ of the same length, where $^\top$ denotes the transpose of a vector or matrix. We consider the problem of finding numbers ϑ_1 , ϑ_2 and ϑ_3 such that the approximation

$$y_k \approx \vartheta_1 x_k^2 + \vartheta_2 x_k + \vartheta_3, \quad k = 1, \dots, n,$$

holds as closely as possible. The solution $\hat{\vartheta} = (\hat{\vartheta}_1, \hat{\vartheta}_2, \hat{\vartheta}_3)^\top$ to this problem is given by $\hat{\vartheta} = (X^\top X)^{-1} X^\top y$ with

$$X := \begin{pmatrix} x_1^2 & x_1 & 1 \\ \vdots & \vdots & \vdots \\ x_n^2 & x_n & 1 \end{pmatrix}.$$

The approximated y -values are then given by $\hat{y} := X \hat{\vartheta}$. Write a scilab function

```
[y_hat, theta_hat] = quadratic_regression (x, y)
```

that computes $\hat{\vartheta}$ and \hat{y} for vectors $x = (x_1, \dots, x_n)^\top$ and $y = (y_1, \dots, y_n)^\top$. Test your function for

$$x = (0, 1, 2, 3, 4)^\top, \quad y = (1, 0, 3, 5, 8)^\top.$$

Plot x against y and \hat{y} in a common graph for this example.

Useful scilab commands: ones, inv, plot, title, xlabel, ylabel, legend

C-Exercise 03

Let s_1, \dots, s_N denote a time series of, e.g., stock prices on days $1, \dots, N$. The *logarithmic return* (log-return) of the stock on day $n \in \{2, \dots, N\}$ is given by

$$l_n := \log\left(\frac{s_n}{s_{n-1}}\right) = \log(s_n) - \log(s_{n-1}).$$

Assuming 250 trading days per year, the *annualized empirical mean* of log-returns is given by

$$\hat{\mu} = \frac{250}{N-1} \sum_{k=2}^N l_k$$

and the *annualized empirical standard deviation* of log-returns is given by

$$\hat{\sigma} = \sqrt{\frac{250}{N-2} \sum_{k=2}^N \left(l_k - \frac{\hat{\mu}}{250}\right)^2}.$$

Write a scilab function

```
function lr = compute_log_returns (data)
```

that computes and returns the time series of log-returns for the time series given in data.

On the OLAT entry of the course you find the file `time_series_dax.csv` containing a time series of daily DAX data. Import this time series and test your function with it. Visualize the time series of log-returns in a plot. Compute and display the annualized empirical mean and standard deviation of the log-returns.

Useful scilab commands: `diff`, `log`, `csvRead`, `mean`, `variance`, `disp`

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: Friday, 22.04.2016, 08:30
Discussion: in tutorials on Mon, 25.04.2016