Prof. Dr. E. Kostina
M. Sc. H. Stibbe

Ruprecht-Karls-Universität
Institut für Angewandte Mathematik

# Linear Optimization - WS 2017/2018

## Excercise 1

## MATLAB-EXCERCISES

### M1

Define the following matrices and vectors in Matlab:

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 3 \end{pmatrix}, \quad B = \begin{pmatrix} 4 & 6 & 8 \\ 11 & 2 & 12 \\ 3 & 5 & 16 \end{pmatrix}, \quad u = \begin{pmatrix} 4 & 6 & 8 \end{pmatrix}.$$

Now calculate the following products:

a) $c = uB$,

b) $d = Bu^T$,

c) $e = cd$,

d) $F = dc$,

e) $G = (c^T d^T)^T$.

### M2

The linear system of equations is given as:

$$\begin{aligned} 2x_1 + 4x_2 - 2x_3 &= 2 \\ 4x_1 + 9x_2 - 3x_3 &= 8 \\ -2x_1 - 3x_2 + 7x_3 &= 10 \end{aligned}$$

Formulate the system of equations in the notation $Ax = y$. Use the Matlab function `lookfor` to find the function for matrix inversion and calculate $x = A^{-1}y$.

### M3

We consider the quadratic function

$$f(x) = a_1 x^2 + a_2 x + a_3.$$

The zeros of this function can be calculated by solving an equation of the form $x^2 + px + q = 0$ using the so called $pq$-formula

$$x_{1,2} = -\frac{p}{2} \pm \sqrt{\frac{p^2}{4} - q}.$$

Formulate the quadratic function with $a_1 = 2,\ a_2 = 4,\ a_3 = -6$ as above and calculate the zeros using the $pq$-formula. Compare your results with the zeros that you get using the Matlab-function `roots`.

### M4

*Euler's number* is $e = 2,718281828459\ldots$. It is the limit of the following series:

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \cdots = \sum_{n=0}^{\infty} \frac{1}{n!}.$$

a) Calculate an approximative value $e$ with a `for`-loop using the first five coefficients of the power series.

b) Calculate the value of Euler's number $e_{Mat}$ using the Matlab-function `exp`. What is the relative error

$$\eta_{rel} = \left| \frac{e - e_{Mat}}{e_{Mat}} \right| \cdot 100$$

of $e$ from a) compared with the exact value $e_{Mat}$?

c) Calculate $e$ using a `while`-loop such that the relative error stays below a given tolerance TOL=$10^{-6}$, i.e. $\eta_{rel}$ <TOL.

d) Generalize your programm for arbitrary power series

$$e^x = 1 + \frac{x^1}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots = \sum_{n=0}^{\infty} \frac{x^n}{n!}.$$

**M5**

We consider a quadratic function as in M3 $f(x) = a_1x^2 + a_2x + a_3$ and the calculation of the zeros using the $pq$-formula. The value of the discriminant $D := \frac{p^2}{4} - q$ is determined by the number of real zeros:

$$\begin{aligned} D < 0 &: \text{ no real zero} \\ D = 0 &: \text{ one real zero} \\ D > 0 &: \text{ two real zeros} \end{aligned}$$

Write a function `function`$[x_1, x_2]$ `= nullst(`$a$`)`, where $a = \begin{pmatrix} a_1 & a_2 & a_3 \end{pmatrix}$ is the vector of coefficients of the function $f(x)$.

a) Distinguish the cases $a_1 \neq 0$ and $a_1 = 0,\ a_2 \neq 0$ respectively using an `if`-statement.

b) For the case $a_1 \neq 0$ calculate the values of $p$, $q$ and $D$ and determine the number of real solutions using an `if`-statement. Write another function `disp`, that can display

$$D = 0 : \text{A real zero:}$$

In the case of existence also calculate the zeros and display them.

c) In the case of $a_1 = 0,\ a_2 \neq 0$ the quadratic equation becomes a linear equation. Calculate the zeros for this case and display them.

d) Test your problem with the following coefficients:

(i) $a_1 = 1,\ a_2 = 2,\ a_3 = -3$

(ii) $a_1 = 0,\ a_2 = 2,\ a_3 = 4$

(iii) $a_1 = 1,\ a_2 = 2,\ a_3 = 1$

(iv) $a_1 = 1,\ a_2 = 2,\ a_3 = 8$

## M6

Write a function `function` $c$ `=` `vectorprod(`$a, b$`)` for calculating the vector product

$$c = a \times b = \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix}$$

and test it with the vectors $a = \begin{pmatrix} 4 \\ 1 \\ 0 \end{pmatrix}$ and $b = \begin{pmatrix} -2 \\ -2 \\ 0 \end{pmatrix}$.

## M7

Use the operator `:` to construct the matrix

$$A = \begin{pmatrix} 4 & 6 & 8 \\ 11 & 2 & 12 \\ 3 & 5 & 16 \end{pmatrix}$$

from

$$B = \begin{pmatrix} 11 & 2 \\ 3 & 5 \end{pmatrix}, \quad u = \begin{pmatrix} 4 \\ 6 \\ 8 \end{pmatrix}, \quad v = \begin{pmatrix} 8 \\ 12 \\ 16 \end{pmatrix}.$$

## M8

The Newton-method for calculating a zero works with the iteration

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \qquad n = 0, 1, 2, \ldots, \qquad f'(x_n) \neq 0.$$

a) For the function $f(x) = x^2 + 2x - 3$ determine the first three iterates with 6 decimal places starting with the inital value $x_0 = 0$.

b) Write a function `function` $x$ `=` `newton(`$f, df, x0, TOL, NMAX$`)`, where the functions $f$ and $f'$ are declared as `inline`. Iterate in the function with a `while`-loop until either the value is below the stop tolerance TOL, i.e. $|x_{n+1} - x_n| < TOL$ or the maximum number of iterations is reached, i.e. $n > NMAX$.

c) Check the number of input parameters using the Matlab-function `nargin`. Expand your program such that it uses $NMAX = 20$ if there are 4 input parameters and $NMAX = 20$ and $TOL = 10^{-6}$ if there are 3 input parameters. If there are less than 3 input parameters it should display a error message.

d) Write a helptext that explains the functionality of your program when you use the command `help newton`.

e) Test your program for the function in a) with the parameters $x_0 = 0$, $NMAX = 20$ for different accuracies $TOL \in [10^{-9}, 10^{-3}]$ and in each case display the number of required iterations as well as the running time of the programm using the Matlab function `tic` and `toc`.

f) Plot your results over the interval $[x^* - 5, x^* + 5]$ using the Matlab command `plot`, where $x^*$ is the detected zero.

**M9**

Define the following vectors and matrices in Matlab:

$a = \texttt{zeros}(1,5), \quad B = \texttt{zeros}(2), \quad c = \texttt{ones}(1,5), \quad D = \texttt{ones}(2), \quad E = \texttt{eye}(5), \quad F = \texttt{eye}(2).$

First think about which results the following expressions will provide, and then check your assumption with Matlab.

a) $a + c$

b) $c' * c$

c) $E(2,:) + c$

d) $a. * c$

e) $c * c'$

f) $E(:,3) + c$

g) $a' * c$

h) $E * c$

i) $D + F$

j) $a'. * c$

k) $c * E$

l) $D. * F$

m) $D * F$

n) $B. * D$

o) $B + D(1,1)$