Bonus points sheet

Winter term 2020/2021

Computational Economics

(version: 15. Oktober 2020)

1. The deadline for this sheets ends on **January 31, 2021** (end of day). Submission is via Ilias.
2. **Handwritten program code is not allowed! Submission must be a PDF with proper R-Code!**
3. This sheet contains in total **7 questions**. All questions cover in total **7 pages**.
4. You receive **9 points** (i.e., 10 %) as bonus points for the exam if **50 % (or more)** are correctly answered. However, these points are only added if you pass the exam without bonus points.**The bonus points cannot prevent you from failing the exam!**
5. Group work is not allowed. **Plagiarism can lead to exclusion from the exam.**
6. **Please label the submission document with your name, matriculation number, and program of study.**

Good luck!

Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5	Ex. 6	Ex. 7	Total	Pass
7	8	15	4	8	4	9	55	

Exercise 1: Visualization

a) Create a colored plot with contours of the function $f(x, y) = \frac{x^2}{2} + \sin x + \frac{y^2}{2} - \cos y$. Both x- and y-dimension should range from -10 to 10. 4 3

b) Prepare a 3D plot of the function $f(x, y) = \sin^2 x - \cos^2 y$.

Exercise 2: Dataframes

8

a) Load the csv file *persons2.csv* into a dataframe.

- Insert a column *bmi* which indicates the body mass index (BMI) of the person which is defined as $bmi = \frac{weight\ in\ kg}{(height\ in\ m)^2}$
- A BMI between 19 and 24 is considered to be normal. Which person does not have a normal BMI?
- Create a histogram of the BMI with step size one. Choose reasonable scales and labels for x and y-axis.

Exercise 3: Control Flow

5

a) Write a for-loop that calculates the following sum

$$s(n) = \sum_{i=0}^{n} \frac{1}{2^i}.$$

Test your loop with $n = 10$.

10

b) Write a function that tests if a positive integer n can be written as sum of two squared integers, i. e. $n = i^2 + j^2$ with $i, j \in \mathbb{N}_{>0}$. Test your function for the integer $n = 100$.

Exercise 4: Linear Algebra

4

a) The solution of a linear system

$$A\boldsymbol{x} = \boldsymbol{b}$$

is often rewritten as minimizing a function

$$F(\boldsymbol{x}) = \|A\boldsymbol{x} - \boldsymbol{b}\|^2.$$

We can calculate the gradient to

$$\nabla F(\boldsymbol{x}) = 2A^T(A\boldsymbol{x} - \boldsymbol{b}).$$

Write a matrix expression in R to compute $\nabla F(\boldsymbol{x})$ for

$$A = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 5 \\ 4 \end{bmatrix} \quad \text{and} \quad \boldsymbol{x} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}.$$

Exercise 5: Numerical Analysis

1

7

a) Convert the number 243043 from base 10 into the hexadecmal. State the corresponding R code.

b) Visualize the function $f(x) = e^x + e^{-x}$ and its Taylor approximation for

$$x_0 = 1$$

Use an x-axis ranging from -5 to 5 and y-axis from 0 to 20.

Exercise 6: Optimality Conditions

4

a) Verify that $\boldsymbol{x} = \begin{bmatrix} -\frac{9}{22} \\ -\frac{2}{11} \end{bmatrix}$ is a minimum of the function

$$f(x_1, x_2) = 3 + 2x_1 + 3x_2 + 2x_1^2 + 2x_1x_2 + 6x_2^2.$$

State the corresponding R code.

Exercise 7: Optimization

9

a) Implement the one-dimensional Newton's method to find a local extreme point of a function $f(x)$ given a starting point $x^{(0)}$ and a tolerance ε. Stop the algorithm, when $|x^{(i)} - x^{(i-1)}| < \varepsilon$. Complete the following R code.

```
library(Deriv)
f = function(x) x**2 * sin(x-3) * exp(-0.5*x)
f.prime = # TODO
f.double.prime = # TODO

newton = function(f.prime, f.double.prime, x0, tol) {
  # TODO
}
newton(f.prime, f.double.prime, 5, 10**-6)
```