**Take-home exam**

**Personal analysis**

The goal of this project is to present a small research implemented with computational methods

and tools taught throughout the course. You may choose any subject or question, preferably one that is connected to your research or for which you have available data.

You should use python code to perform the processing and analysis of the data. You may use any python module you see fit or any external code, as long as the main body of the code is original.

You should **submit the topic for approval** **along with the general workflow**, during the last session of the semester **(28.06)** we will briefly consider the topic with you to verify that the scope of the project is reasonable.

Final submissions of the project will be done through the Moodle website. **Due date is august 10th.**

**Project submission**

The final submission will include the following files in a zip file containing:

**1. Code**

**One main script file** with all the code used for generating, processing and analysing the data. The code should be readable and clear, including comments (the comments within the code must be in English) and function documentation. It is important that the code will run with no errors and the results will be reproducible. **Do not include code that is not necessary** to reproduce the analysis and results (i.e., old versions or redundant sections). **More detailed instructions are below in ‘Grading’ section.**

**2. README file**

Attach a **short** text file called “README.txt” containing the code usage, i.e., how to run your code (for example: “run the script main.py and make sure the input.csv file is in the same directory”).

Following these instructions on pc with python3.9 and anaconda installed should result in no errors. If you want to use any library that was not used in the course you should specify this here (along with the specific versions).

**3. Paper**

**A word / PDF file containing:**

* Overview: what is the background and context of your project? What is the research question? Write as clear and concise as possible to a reader that is not familiar with the field.
* Data: describe the data you will analyse using python.
* Methods: present the workflow of the analysis performed **in your code**, how did you load/organize/process/etc. the data.
* Results: explain briefly the results shown in the figures.

All sections should be concise and clear.   
**This part should be 1-2 pages. No more than 2 pages.**

**4. Data**

The original data on which analyses were performed. Make sure that by using the code on the provided data, the results will be reproduced.

**Grading**

**Only the code that you submit will be evaluated** (however, take into account the supporting sections contribution to code readability).The grading of this part will be based on the quality of the pythonic programming: proper and correct use of python code, usage of the right tools and modules, and correct usage of data structures: strings, lists, tuples, dictionaries, numpy arrays, pandas, etc.

The project must contain, and will be evaluated based on, the next three sections:

1. a. Handling and generating data:

Accessing and parsing data (from CSV, txt, online data, or other source): your data should be easily read and processed. Generating and organizing the data in a convenient pythonic data structure (Pandas, Numpy, etc.). Make sure to use data frames, dictionaries, arrays, lists, tuples, and strings in an elegant way that fits your work.

1. b. Main part of the project:
2. \* Statistical analysis using Numpy, Scipy, scikit-learn and setatsmodels (regression, optimization, hypothesis testing, linear algebra, normalization, GLM, PCA etc.)
3. \* Simulations
4. \* String analysis using regular expressions (regex)
5. \* Machine learning using scikit-learn, tensorflow, keras, xgboost
6. \* These are only examples, be creative!
7. c. Visualization of the data:
8. prepare 2 **different** plots (and different types) of your data that summarize the process: heatmaps, histograms, scatter plots, etc.   
   The plots should be in publication-quality. namely - clear, aesthetic, containing suitable titles, axis labels, legends (if needed) etc. Make sure you adjust all the needed relevant parameters.
9. d. Correct coding practices:
10. your project will also be evaluated on usage of correct Python coding practices, and code readability.   
    Pay attention to the following: variable names, function names, code documentation (comments), line spacing, avoiding excessive if/loop nesting, appropriate error handling, and code compartmentalization using functions (and/or classes).
11. **Additional requirements**

* The code needs to be able to run on a windows PC that is not your own, using python3.9 (test this before submission – you can use the computer lab). if you want to use libraries we did not use during the course – specify which version you are using.
* In your code you must use objects/functions from **at least** one of the next packages: pandas, numpy, scipy, statsmodels, scikit-learn, xgboost, tensorflow, regex.
* The code should consist 100-250 lines of code (not including comments / empty lines).

**Help**

Questions regarding the project are more than welcome. Use the course forum or consult one of us for questions about the instructions or the programming involved. You are also encouraged to consult your adviser or anyone else; however, **the actual code should be written by you alone**. Please ask your questions early, so we can attend all issues.

**GOOD LUCK!**