Introduction to Data Science 2024
Assignment 1

In Assignment 1 you will work with probability, statistics, and hypothesis testing.

Available from Tuesday, February 6th. Your report should be uploaded to Absalon no later than Monday 19th, at 15:00 latest. Total points: 50. Responsible TA: Antonia Karamolegkou.

Guidelines for the assignment:

- **The assignments in IDS must be completed and written individually.** This means that your code and report must be written by yourself.
- Exercises in blue are coding exercises and will be evaluated as such: do provide a working code! Code files must be handed in a single zip file. If some code templates are provided in Absalon, please use them.
- Upload your report as a single PDF file (no Word .docx file) named `firstname.lastname.pdf`. What do include in the report? Check the deliverables of each exercise!
- Upload your code as `firstname.lastname.zip`. It should consist of one or more Python scripts (text files with `".py"` extension) or Jupyter/IPython notebooks (with `".ipynb"` extension). Do not upload the report and source code in a single zip archive. This makes it impossible to use Absalon's SpeedGrader for annotating the reports.
- We will grade using Python 3.10. In some assignments, the specific package versions might be specified in `requirements.txt`. Other versions may work or break; using these versions is safest. Using Anaconda, create an appropriate `conda env` with:
  `conda create --name IDS-A1 python=3.10`
  then activate it with: `conda activate IDS-A1`,
  then install the packages with: `pip install -r requirements.txt`

1 Academic Code of Conduct

You are welcome to discuss the assignment with other students, but sharing of code is not permitted. Copying code or text from the report directly from other students will be treated as plagiarism. Please refer to the University's plagiarism regulations if in doubt. For questions regarding the assignment, please ask on the Absalon discussion forum.

In short, plagiarism means copying text or ideas from others without acknowledging the underlying sources. Crucially, this does not mean that you are prohibited from building on others' ideas or use external sources, but rather that you have to properly acknowledge

all sources used in your work. This holds for instance for building on code from lectures or lab sessions. If in doubt, we recommend erring on the side of over- rather than under-acknowledging sources.

While guidance on AI assistance usage is not covered in this course, you are welcome to take a look at the Absalon course on Learning resources for Digital Literacy.

You are also welcome to use AI assistance (e.g., ChatGPT, GitHub Copilot) for tutoring purposes, augmenting the TAs for help with questions and issues. However, keep in mind that their output is not guaranteed to be either comprehensive, true or aligned with the course scope and expectations. Always check with the TAs in case of doubt. Importantly, the use of AI assistance while writing the assignment is allowed only for the following purposes:[1]

- As coding tools (e.g., GitHub Copilot): no restrictions.
- As writing tools to improve the writing of original content, i.e. when the prompt you write contain all the ideas to be formulated: no restrictions.
- As search tools to identify related literature: no restrictions. Usual citation requirements apply (see plagiarism note above): you must cite the original work you identify, even if you used an LLM to find it. Just like you do not cite Google Search for papers you find using it, you should not cite ChatGPT for this either. In particular, always make sure that the citations it provides actually exist—LLMs are known to often generate plausible but nonexistent references.
- As generation tools for *new* ideas: generated content must be clearly highlighted even if post-edited by yourself. All prompts/transcripts from the tools used must be included as an appendix at the end of the submission in PDF, after the references.

For all uses of AI assistance, the purpose, tool, and version[2] must be stated in the submission—e.g, in a dedicated section. Here is such a statement for example:

```
ChatGPT August 3 Version was used as a writing assistance tool and as a
serch tool to identify related literature.  GitHub Copilot July 14 Version
was used while developing the code for...
```

Does smoking affect your lung capacity?

It is well known that smoking is not good for your health, but how can we quantify this statistically? In this assignment, you will work with a dataset consisting of information on the lung function, smoking status and demographics of 654 youth and children aged 3-19. See Appendix A below for a detailed description of the data material, and see in particular Appendix B below for a description of the so-called FEV1 measure, which quantifies lung function.

While this assignment is not very heavy on implementation, we want nevertheless you to get familiar with Python, `numpy`, `pandas`, `scipy`, `seaborn` and `matplotlib`. You are not allowed to use any other libraries; you do not need to use all of these libraries (e.g. if you want to use pure `matplotlib` instead of `seaborn`, that's fine.)

[1] Note that evaluating LLMs as models on task data is not considered "AI assistance" and is not restricted or affected by the rules here.

[2] If multiple version have been used throughout the assignment, list all of them. In the ChatGPT web interface, the version is specified at the bottom of the screen, under the text input field.

Exercise 1 (Reading and processing data / 10 points)**.**

a) Read the data from the file `smoking.csv`, and divide the dataset into two groups consisting of smokers and non-smokers. Write a script that computes the average lung function, measured in FEV1, among smokers and non-smokers.

b) Report your computed average FEV1 scores. Are you surprised?

Deliverables. The average lung functions and a brief description of the plot/your observation.

Exercise 2 (Boxplots / 10 points)**.**

a) Make box plots of the FEV1 for the two groups.

b) What do you see? Are you surprised? What makes box plots a good choice for visualizing FEV1 for smokers and non-smokers?

Deliverables. One figure with two boxplots, a brief description of the plot, your findings, and the reason why a boxplot is a suitable visual display. Tip: Check what people typically describe in box plots.

Exercise 3 (Hypothesis testing / 10 points)**.**
Next, we will perform a *hypothesis test* to investigate the difference between the FEV1 level in the two populations *smokers* and *non-smokers*.

a) Write a script that performs a two-sided t-test whose null hypothesis is that the two populations have the same mean. Use a significance level of $\alpha = 0.05$, and return a binary response indicating acceptance or rejection of the null hypothesis. You should try to implement it by yourself – though not the CDF of the t-distribution, use `scipy`'s `stats.t.cdf`. If you can't, you may use `scipy`'s `stats.ttest_ind`.

b) Report your result and discuss it. Are you surprised?

Deliverables. The value of the t-statistic and the degrees of freedom ν, the returned p-value, whether or not you rejected the hypothesis, and a short discussion of the result. Tip: You need to do Welch's t-test. Can you explain why?

Exercise 4 (Correlation / 10 points)**.**

a) Compute the correlation between age and FEV1. Make a scatter plot of age versus FEV1 where non-smokers appear in one color and smokers appear in another.

b) What do you see? What makes a scatter plot a good choice for visualizing these variables?

Deliverables. The scatter plot, the correlation, a brief discussion interpreting the correlation/your results, and the reason why a scatter plot is a suitable visual display.

Exercise 5 (Histograms / 10 points)**.**

a) Create a histogram over the age of subjects in each of the two groups, *smokers* and *non-smokers*.

b) What do you see? Does this explain your results on lung function (FEV1) in the two groups? What makes histograms a good choice in this case?

Deliverables. One figure with the two histograms, and a couple of lines of discussion.

A The data material

The file `smoking.csv`, which can be found in Absalon, contains a 654×6 matrix, where each column corresponds to (in the given order):

- age – a positive integer (years)
- FEV1 – a continuous value measurement (liter)
- height – a continuous value measurement (inches)
- gender – binary (female: 0, male: 1)
- smoking status – binary (non-smoker: 0, smoker: 1)
- weight – a continuous value measurement (kg)

This data is collected from 654 youth and children and each row in the matrix can thus be considered as an observation describing one child/youth.

NB. The `smoking.csv` file does not contain headers, so you can use either numpy or pandas to read the data. If you use pandas, you can add headers by adding arguments to `read_csv(header=None, names=['column1name', 'column2name', ...])`. Also, values in the file are tab-separated, which means you need to specify `sep='\t'` setting in `read_csv` function call.

B Measurement of lung function

Figure 1: Illustration of a spirometry test.

Lung function can be measured using a *spirometry* test, where the person blows into an apparatus as illustrated in Figure 1, and several parameters are computed based on the result. One of these parameters is the *forced expiratory volume in one second* (FEV1), which measures the volume that a person can exhale in the first second of a forceful expiration after a full inspiration. This measure will be used as an indicator of lung function in this assignment. A decrease in FEV1 generally indicates a decrease in lung function.