

Data Analysis with Python

CorrelAidXKonstanz

Welcome

A brief round of introduction:

- What is your name?
- What do you study?
- Do you have prior coding experience?

What is “Data Analysis”?

Data analysis **the process of systematically collecting, cleaning, transforming, describing, modeling, and interpreting data**, generally employing statistical techniques. Data analysis is an important part of both scientific research and business, where demand has grown in recent years for data-driven decision making. Data analysis techniques are used to **gain useful insights from datasets**, which can then be used to make operational decisions or guide future research. - [Encyclopedia Britannica](#)

What is Python?



Python is a widely used high-level programming language. It is known & liked for its very clear and easy-to-understand syntax, and its versatility: Python is used to build anything from websites to artificial intelligence systems.

It's popular in data analysis due to its rich ecosystem of libraries (like pandas and NumPy) that streamline tasks like data manipulation, visualization, and statistical analysis.

```
# Compute the n-th value in the fibonacci-sequence:

def fibonacci(n):
    if n <= 1:
        return n

    return fibonacci(n - 1) + fibonacci(n - 2)

for i in range(1,11):
    print(fibonacci(i))
```

What is Anaconda?

Anaconda is a “distribution” of Python for scientific computing (Data Analysis, Data Science, Machine Learning, etc. etc.).

It includes the language and a bundle of around 250 packages that extend the functionality of the base language, as well as a way to manage packages and virtual environments.



What are Jupyter Notebooks?

Jupyter Notebooks are interactive coding environments where you can write and run Python code in small sections, called cells, and see the output right away. It's especially useful for data analysis because you can mix code, visualizations, and explanations (like text or equations) all in one place, making it easy to test ideas and document your process as you go.

```
[5]: import matplotlib.pyplot as plt
plt.style.use('classic')
%matplotlib inline
import numpy as np
import pandas as pd
import seaborn as sns
sns.set()
```

```
[6]: rng = np.random.RandomState(0)
x = np.linspace(0, 10, 500)
y = np.cumsum(rng.randn(500, 6), 0)
```

Next step

Now, create a graph.

```
[7]: plt.plot(x, y)
plt.legend('ABCDEF', ncol=2, loc='upper left');
```



Setup time

Part 1: local setup (running code on *your* device)



Download Anaconda:

- [Anaconda](#)
 - You have to give a mail address to download, if you don't want to use your own you can use a [temporary one](#).

You can find the installation instructions here:

- [Windows](#)
- [MacOS](#)
- [Linux](#)

Setup time

Part 2: remote setup (running code remotely, i.e. not on your device)

If you do not have a proper laptop:

Go to [Google Colab](#). Colab is a hosted Jupyter Notebook service that requires no setup to use and provides free access to computing resources, including GPUs and TPUs. Colab is especially well suited to machine learning, data science, and education.

In short, Colab allows you to run code from within your browser (it is run remotely on some Google server and not actually on your device).

Course outline

Outline here

- **Week 1:** Setup & getting started
- **Week 2:** Basic programming in Python
- **Week 3:** Data Analysis with [pandas](#) and [numpy](#) pt. 1
- **Week 4:** Data Visualization
- **Week 5:** Data Analysis with [pandas](#) and [numpy](#) pt. 2
- **Week 6:** Project topics and start of project work

The rest of the sessions will be open coworking & bonus-content sessions, where you can come here to work on your projects. At least one of us will always be here to help you or offer assistance.

There will also be some sessions with bonus content (e.g. web scraping) which we will announce in time.

Final projects

The “examination” for this course will be a data analysis project that you will work on on your own. You can do this either alone or in groups (max. 3 people).

We will provide some topics, but you can also come up with your own ideas. In the end, you are expected to hand in your code (most likely in the form of one or more notebooks) and optimally also a short expose (max. one full page) to explain what you did and what your results are.

The course is ungraded (pass or fail). To receive the ECTS, you have to complete and hand in the project.