

Einführung in die Programmierung mit Python

Dr. Bernd Zey

Abteilung Forschungsdatenmanagement

TU Dortmund

30.08.2024

- Worum geht es heute hier?
 - Kurz-Einstieg in Python
 - Anwendungsfall
 - Tabellarische Daten laden und auswerten
 - Ergebnisse grafisch darstellen
 - Ergebnisse als Datei speichern
 - Keine Python-/Programmierkenntnisse erforderlich
- Motivation
 - Warum Python?
 - Entwicklungsumgebungen (IDE)
- Ausblick, Take-Home
- Live-Coding
 - JupyterLab

Warum Python?

- **Standard** in Data Science
- **Zahlreiche Module** für die statistische Auswertung und Visualisierung (Open Source)
 - **Pandas**, NumPy, **Matplotlib**, Seaborn, ...
- **Einsteigerfreundliche Programmiersprache**
 - Allgemeine Programmiersprache, Plattform-unabhängig
 - Viele (wissenschaftliche) Open Source-Projekte
 - Einfache Syntax
 - Objektorientierung, Prozedural, ...
 - Schnell umfangreiche Programme möglich
 - Einfaches Datei-Handling
 - Viele Module, u.a. für Machine Learning (TensorFlow, Keras, ...)
 - Online-Hilfe



- Entwicklungsumgebung – IDE (Integrated Development Environment)
 - JupyterLab <https://jupyter.org/>



Entwicklungsumgebungen



Gemeinsam besser
studieren, forschen und lehren

- Entwicklungsumge
- JupyterLab <http://>

The screenshot shows the JupyterLab interface. On the left is a file browser with a search bar and a list of files. The selected file is '2024-03-py...'. The main area shows a notebook with the following content:

```
# Im nächsten Schritt wird das beispielhaft vorgeführt, wenn die Datei im Unterordner "data" liegt
# import os
```

Dateien laden

- csv-Datei (comma separated value-Datei)
- Optional: falls Date im Unterordner: Pfad mit os zusammenbauen

```
[50]: # Dateiname der zu Ladenden Datei
#filename = os.path.join("data", "Titanic-Dataset.csv") # optional: wenn die Datei im Unte
filename = "Titanic-Dataset.csv"
```

```
[51]: # Datei Laden und in Variable "data" speichern
data = pd.read_csv(filename)
```

```
[52]: # Tabelle data anzeigen
# In JupyterLab direkt mit dem Namen möglich
data
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	
[52]:	0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
	1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs)	female	38.0	1	0	PC 17599	71.2833	C85	C

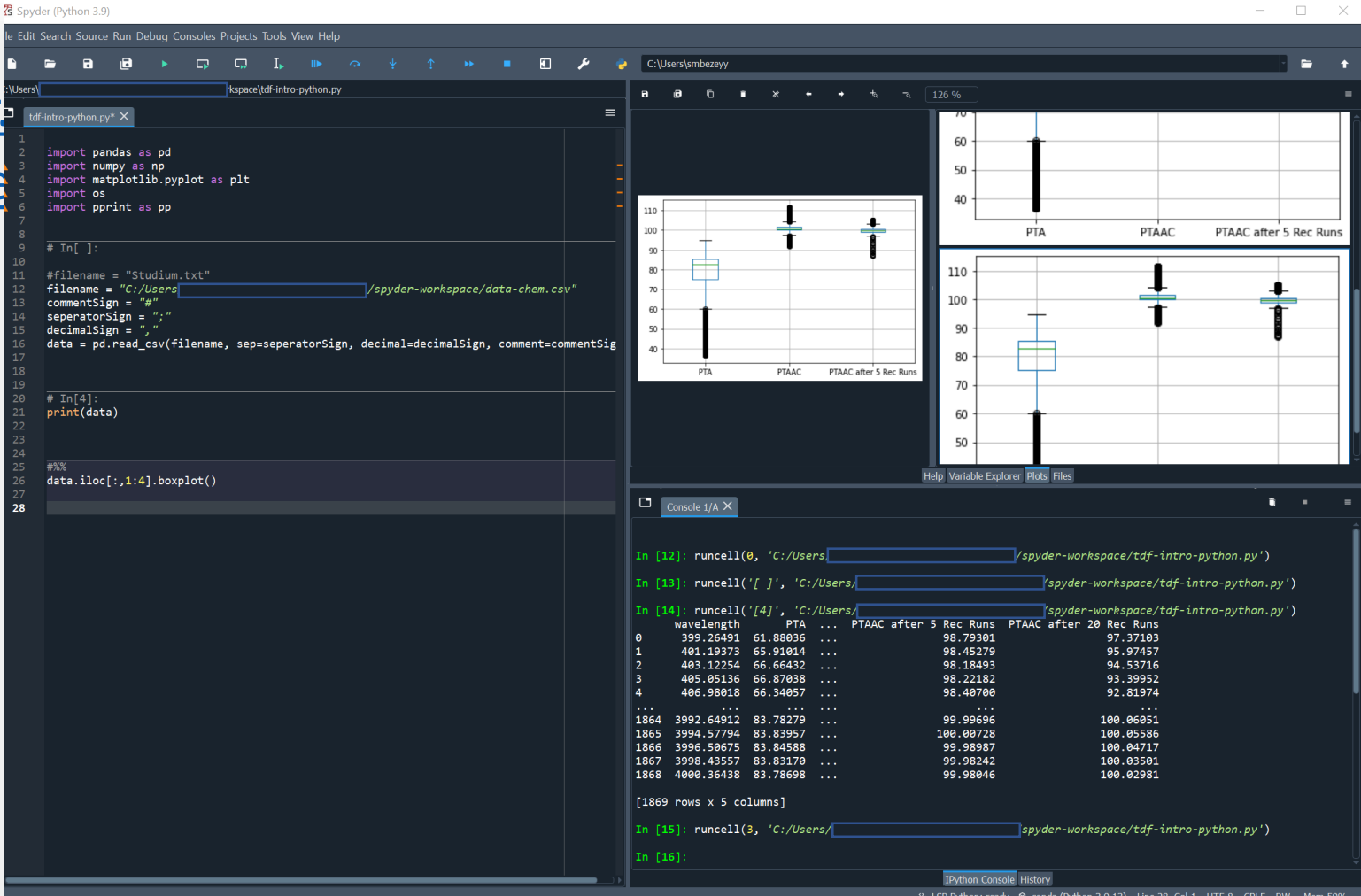
Entwicklungsumgebungen

- Entwicklungsumgebung – IDE (Integrated Development Environment)
 - JupyterLab <https://jupyter.org/>
 - Spyder <https://www.spyder-ide.org/>



Entwicklungsumgebungen

- Entwicklungsumgebung – IDE
 - JupyterLab <https://jupyter.org/>
 - Spyder <https://www.spyder-ide.org/>



```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import os
5 import pprint as pp
6
7
8
9 # In[ ]:
10
11 #filename = "Studium.txt"
12 filename = "C:/Users/.../spyder-workspace/data-chem.csv"
13 commentSign = "#"
14 seperatorSign = ";"
15 decimalSign = "."
16 data = pd.read_csv(filename, sep=seperatorSign, decimal=decimalSign, comment=commentSig
17
18
19
20 # In[4]:
21 print(data)
22
23
24
25 #%%
26 data.iloc[:,1:4].boxplot()
27
28
```

	wavelength	PTA	PTAAC	PTAAC after 5 Rec Runs	PTAAC after 20 Rec Runs	PTAAC after 50 Rec Runs
0	399.26491	61.88036	98.79301	98.79301	97.37103	97.37103
1	401.19373	65.91014	98.45279	98.45279	95.97457	95.97457
2	403.12254	66.66432	98.18493	98.18493	94.53716	94.53716
3	405.05136	66.87038	98.22182	98.22182	93.39952	93.39952
4	406.98018	66.34057	98.40700	98.40700	92.81974	92.81974
...
1864	3992.64912	83.78279	99.99696	99.99696	100.00051	100.00051
1865	3994.57794	83.83957	100.00728	100.00728	100.05586	100.05586
1866	3996.50675	83.84588	99.98987	99.98987	100.04717	100.04717
1867	3998.43557	83.83170	99.98242	99.98242	100.03501	100.03501
1868	4000.36438	83.78698	99.98046	99.98046	100.02981	100.02981

- Entwicklungsumgebung – IDE (Integrated Development Environment)

- JupyterLab <https://jupyter.org/>



- Spyder <https://www.spyder-ide.org/>



- PyCharm <https://www.jetbrains.com/pycharm/>



- Visual Studio Code <https://code.visualstudio.com/>



- Mein Tipp zum Einstieg: Anaconda



- Python-Distribution

- Liefert mehrere IDEs (u.a. alle oben genannten)

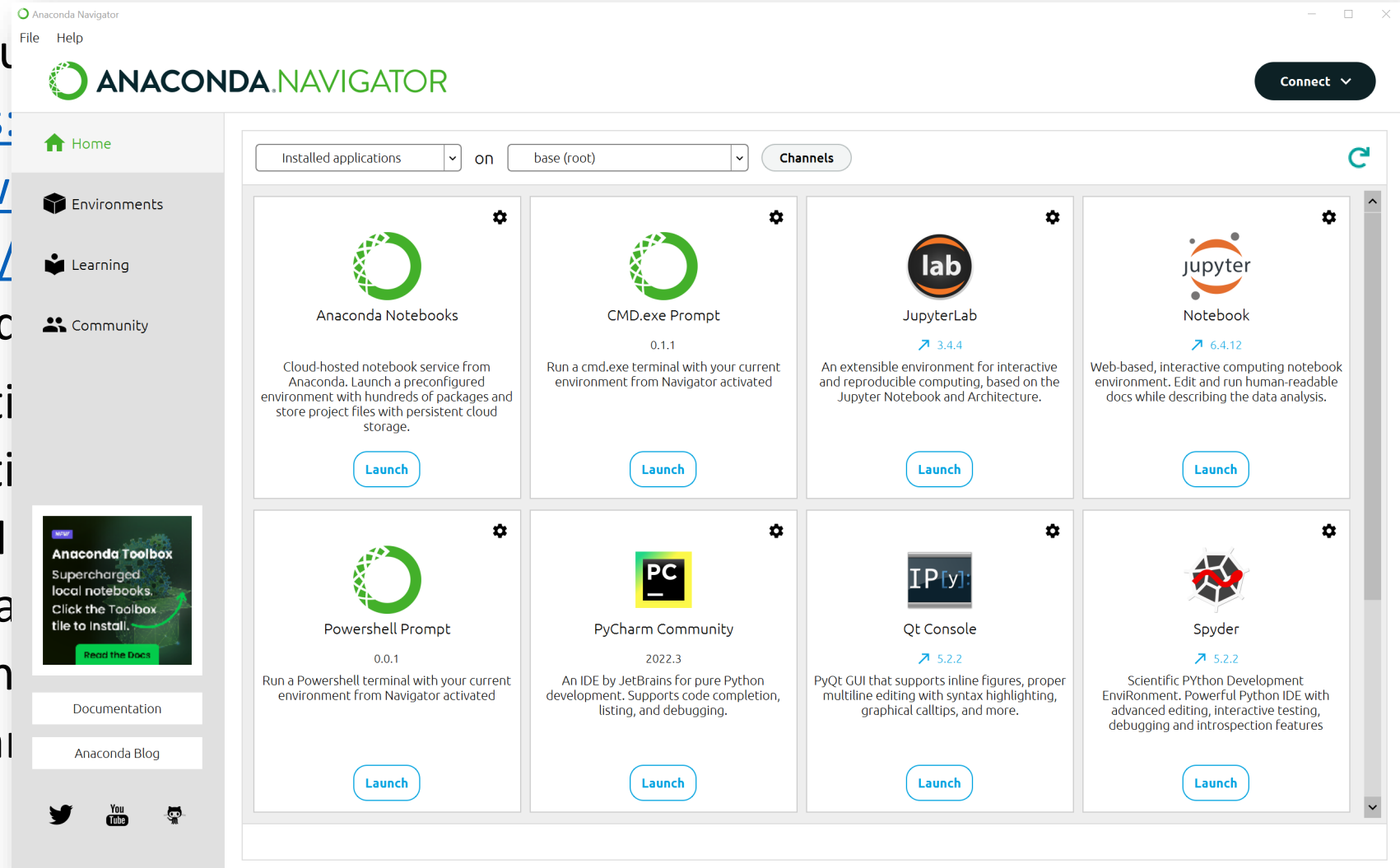
- Anaconda Navigator

- Installiert Python

- Einfache Environment- und Paket-Verwaltung

Entwicklungsumgebungen

- Entwicklungsumgebungen
 - JupyterLab <https://jupyterlab.readthedocs.io/en/latest/>
 - Spyder <https://www.spyder-ide.org/>
 - PyCharm <https://www.jetbrains.com/pycharm/>
 - Visual Studio Code
- Mein Tipp zum Einstellen
 - Python-Distributionen
 - Liefert mehrere Installationen
 - Anaconda Navigator
 - Installiert Python
 - Einfache Environment



- JupyterLab, Live Coding
 - Abschnittsweise Ausführung
 - Markdown-Dokumentation und -Beschreibung
 - Direkte Ausgabe der Diagramme
- Code verfügbar auf Homepage
 - <https://fdm.tu-dortmund.de/informationen/workshop-einfuehrung-in-python/>
 - Lizenz: MIT
- Verwendete Daten
 - Datensatz der Titanic, Lizenz: CC0
 - Siehe FDM-Homepage
 - <https://www.kaggle.com/datasets/yasserh/titanic-dataset>

Ausblick, Take-Away

- Python *einfache* Programmiersprache
 - Automatisierung
- Mächtige Module (heute: Pandas) für Data Science
- Online gute Hilfe (Diskussionen, Code-Schnipsel)
 - z.B. <https://stackoverflow.com>
 - Websuche → „python pandas *thema*“
- Zum Einstieg: Anaconda <https://www.anaconda.com/download>
 - Entwicklungsumgebung JupyterLab (in Anaconda enthalten)

