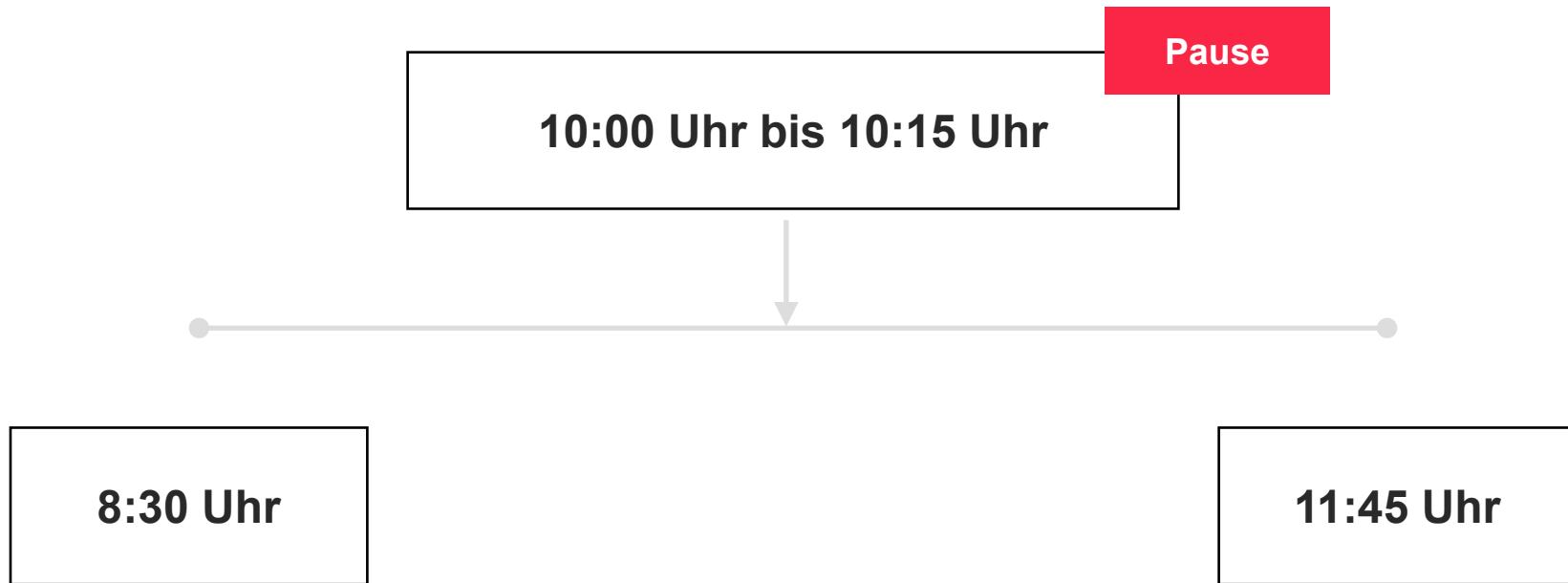


DATA MINING

21. April 2020

Zeitplan



Wer bin ich?

Tin Votan

- Wirtschaftsingenieurwesen / Master of Science
Karlsruher Institut für Technologie
- Wirtschaftsingenieurwesen / Bachelor
Technische Universität Dresden
- Start-up-Gründer
- Machine Learning Engineer

votan@lehre.dhbw-stuttgart.de



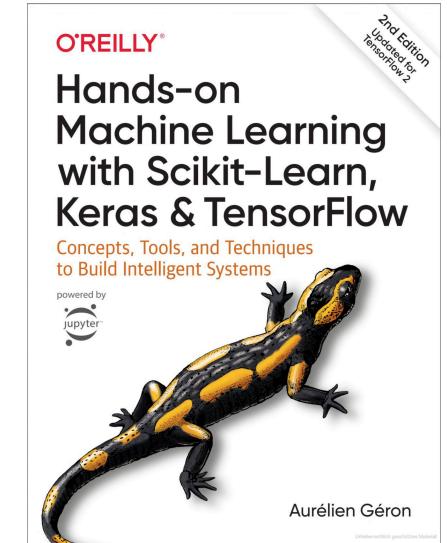
[linkedin.com/in/tinvotan/](https://www.linkedin.com/in/tinvotan/)

Quellen

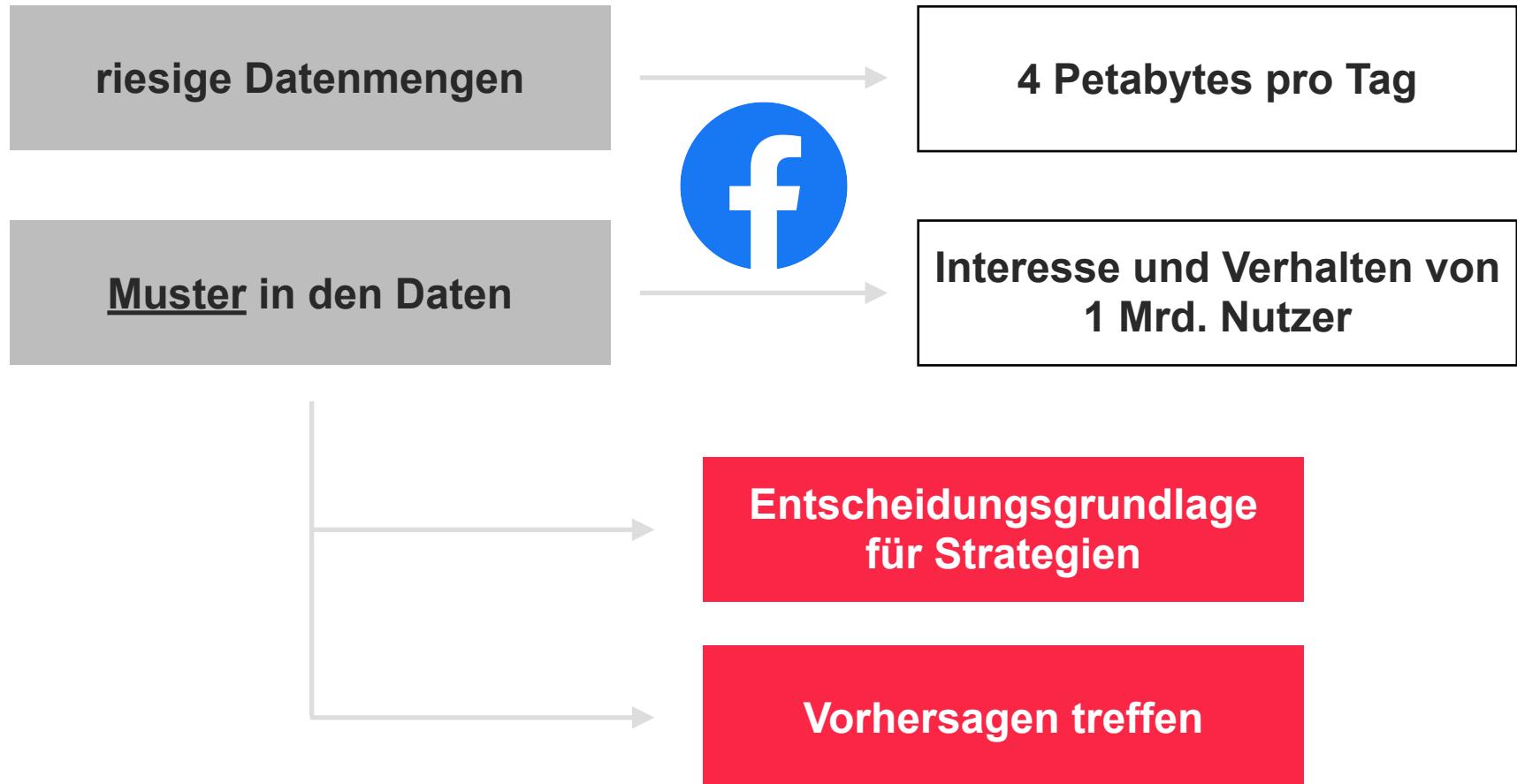
Géron, Aurélien (2019): *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 2. Edition, Sebastopol/CA (United States of America): O'Reilly Media Inc.

Bizer, Christian (2020): *Data Mining - Introduction and Organization*. (Vortrag, 12.02.2020). Mannheim: Universität Mannheim - Data and Web Science Group.

<https://github.com/ageron/handson-ml2> (Abruf: 19.04.2020)



Was ist Data Mining?



<https://www.brandwatch.com/blog/facebook-statistics/>

Definition

Erkundung und Analyse von sehr großen Datenmengen um bedeutungsvolle Muster zu erkennen

Nicht-unbedeutende Extraktion von:

- impliziten,
- vorher unbekannten und
- potenziell nützlichen Informationen aus Daten

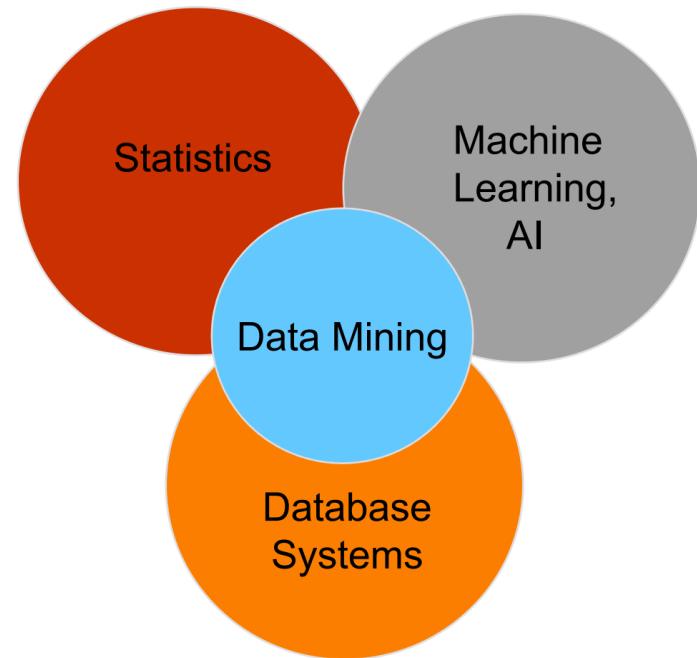
Data-Mining Methodik

1. Interessante Muster in großen Datenmengen erkennen
2. Entscheidungen / Strategien aufgrund der abgebildeten Entscheidungen treffen
3. Vorhersagen über mögliche Ereignisse basierend auf den erkannten Mustern treffen

Ursprünge

Unzulänglichkeiten der traditionellen Verfahren überwinden

- große Datensätze
- hohe Dimensionalität der Daten
- Heterogenität und komplexe Struktur von Daten
- explorative (erkenntnisreiche) Analyse fern vom “*Hypothese stellen und testen*” - Paradigma



Vorgehensweise

1. Daten sammeln

2. Daten erkunden

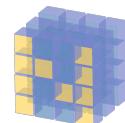
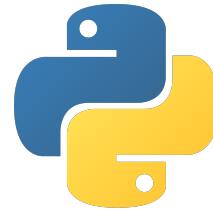
3. Daten aufbereiten

4. Algorithmus auf Daten trainieren

5. Algorithmus evaluieren

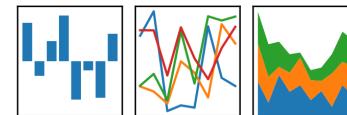
Die heutige Veranstaltung

Vorhersage von der Immobilienpreise in den einzelnen
Distrikten von Kalifornien 1990



NumPy

pandas
 $y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



matplotlib

JupyterHub: Anmeldung auf AWS Client

`http://ec2-35-158-133-141.eu-central-1.compute.amazonaws.com/hub/login`



The screenshot shows a 'Sign in' page for JupyterHub. A yellow warning box at the top left states: 'Warning: JupyterHub seems to be served over an unsecured HTTP connection. We strongly recommend enabling HTTPS for JupyterHub.' Below the warning, there are fields for 'Username:' and 'Password:', both of which have been filled with placeholder text. To the right of the password field is a red button labeled '...'. At the bottom left is an orange 'Sign In' button.

Sign in

Warning: JupyterHub seems to be served over an unsecured HTTP connection. We strongly recommend enabling HTTPS for JupyterHub.

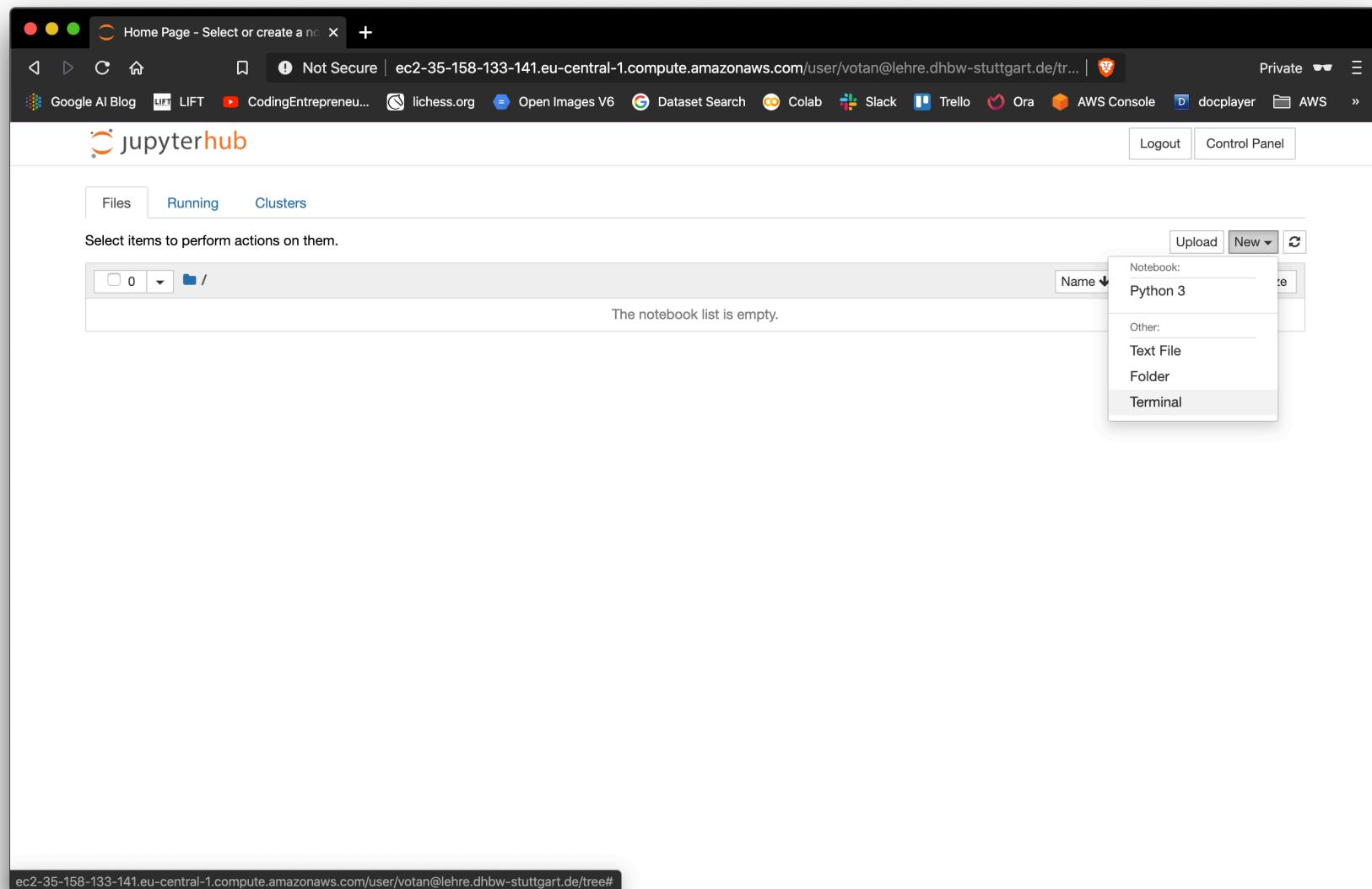
Username:

Password:

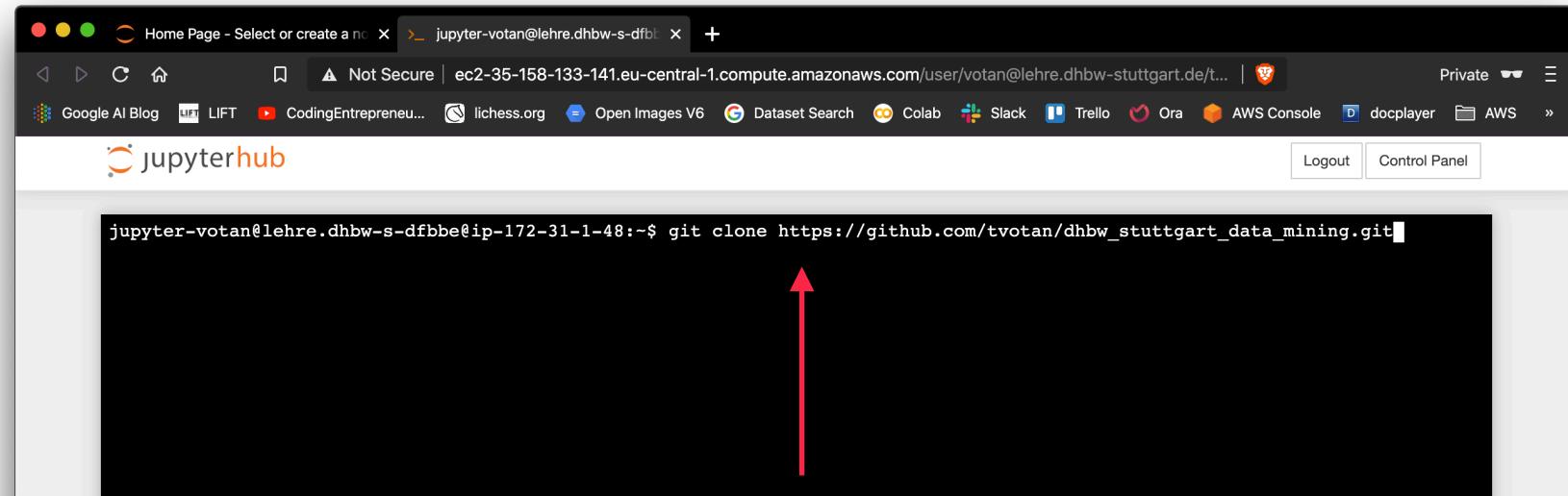
Sign In

...
wiw17XXX@lehre.dhbw-stuttgart.de

JupyterHub: Terminal öffnen



JupyterHub: Repository herunterladen



A screenshot of a JupyterHub terminal window. The title bar shows "Home Page - Select or create a notebook" and the URL "jupyter-votan@lehre.dhbw-stuttgart.de". The terminal window displays the command:

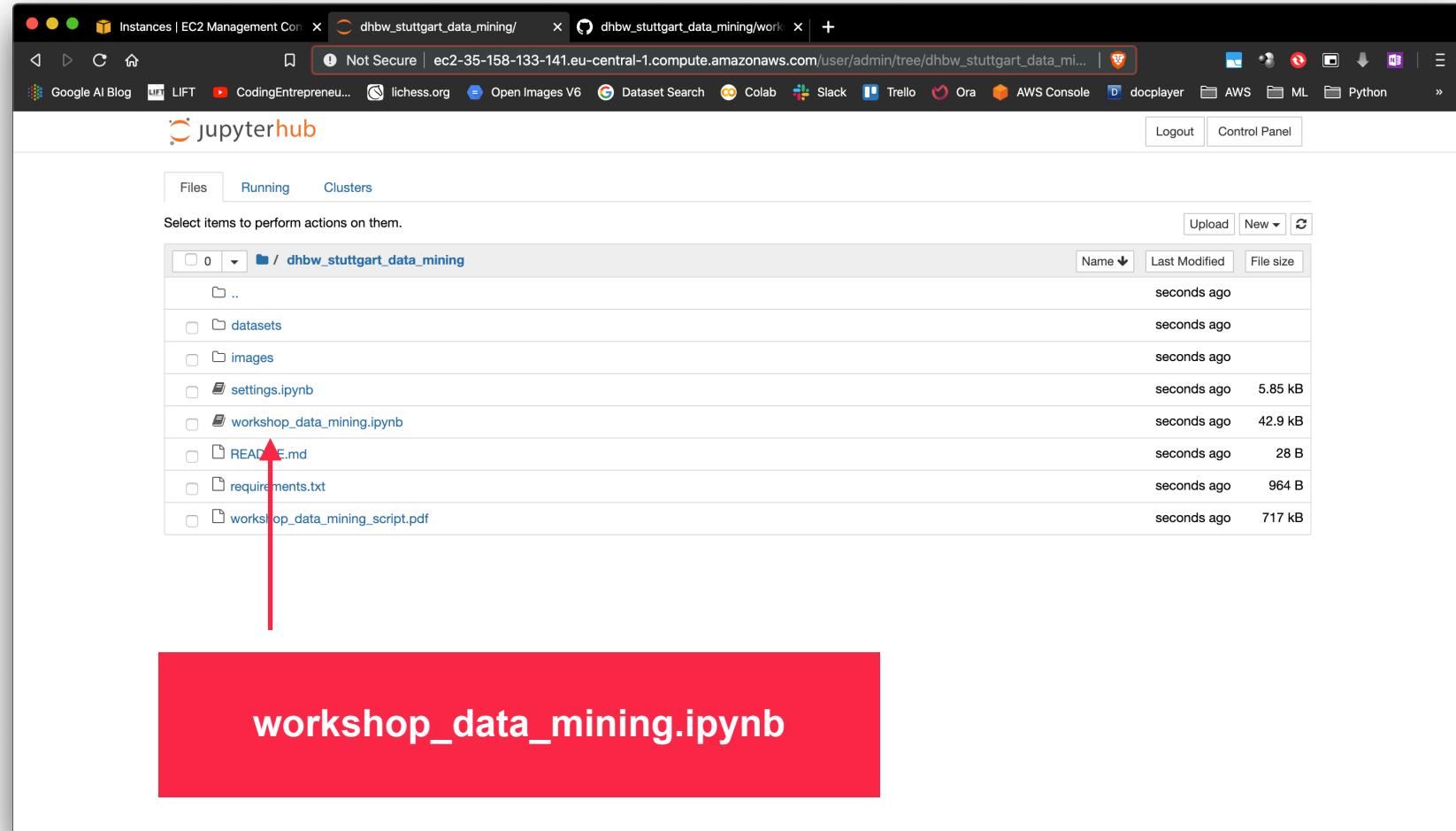
```
jupyter-votan@lehre.dhbw-stuttgart.de:~$ git clone https://github.com/tvotan/dhbw_stuttgart_data_mining.git
```

A red arrow points upwards from the bottom of the slide towards the terminal window.

`git clone https://github.com/tvotan/dhbw_stuttgart_data_mining.git`



JupyterHub: Notebook öffnen



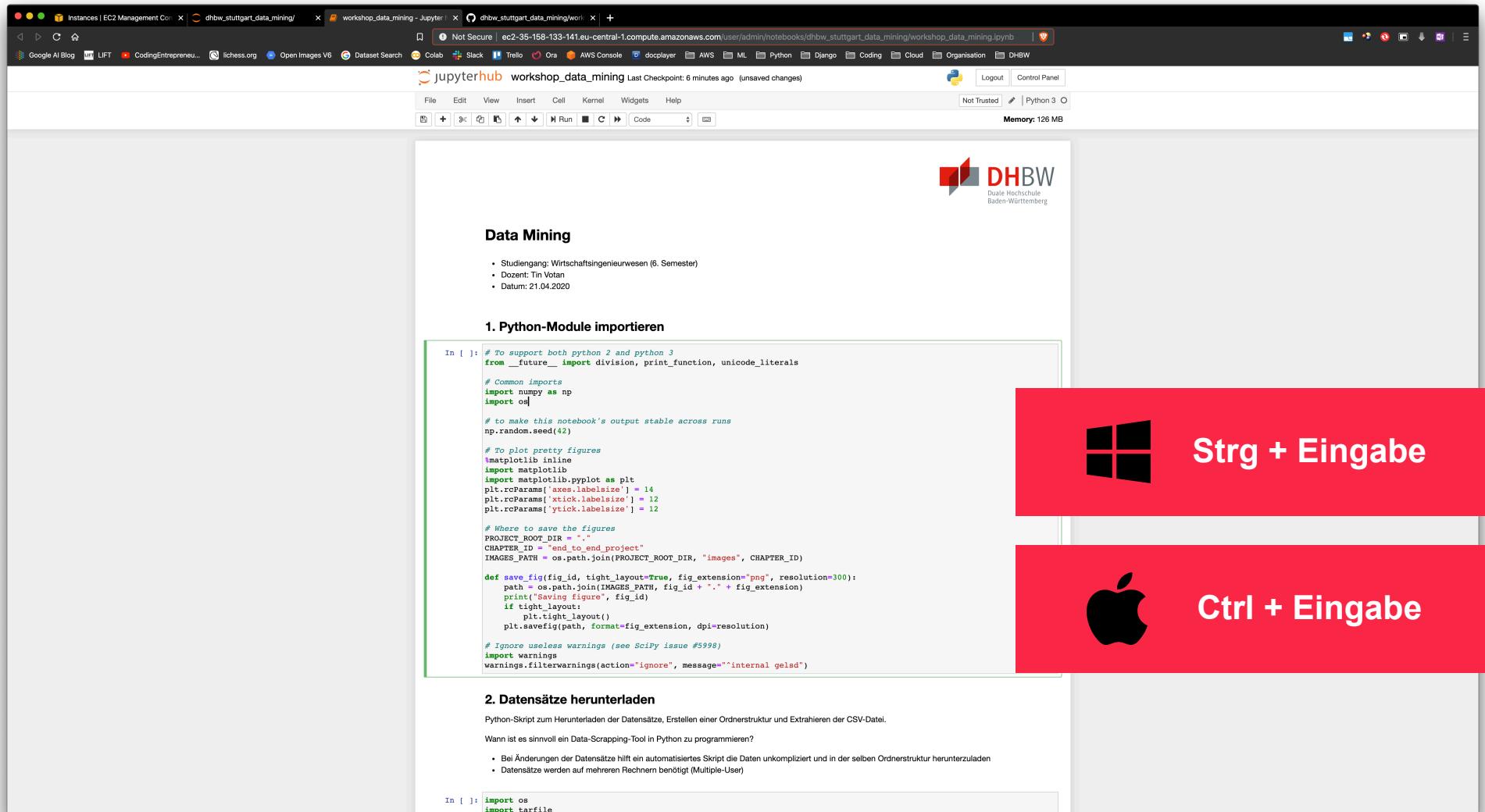
The screenshot shows a web browser window with the following details:

- Address Bar:** Not Secure | ec2-35-158-133-141.eu-central-1.compute.amazonaws.com/user/admin/tree/dhbw_stuttgart_data_mi... | +
- Toolbar:** Instances | EC2 Management Con, dhbw_stuttgart_data_mining/, dhbw_stuttgart_data_mining/work, +
- Header:** jupyterhub, Logout, Control Panel
- File List:** Shows a directory structure under "dhbw_stuttgart_data_mining".

	Name	Last Modified	File size
<input type="checkbox"/>	..	seconds ago	
<input type="checkbox"/>	datasets	seconds ago	
<input type="checkbox"/>	images	seconds ago	
<input type="checkbox"/>	settings.ipynb	seconds ago	5.85 kB
<input type="checkbox"/>	workshop_data_mining.ipynb	seconds ago	42.9 kB
<input type="checkbox"/>	README.md	seconds ago	28 B
<input type="checkbox"/>	requirements.txt	seconds ago	964 B
<input type="checkbox"/>	workshop_data_mining_script.pdf	seconds ago	717 kB
- Callout:** A red callout box highlights the "workshop_data_mining.ipynb" file.

workshop_data_mining.ipynb

JupyterHub: Ausführen von Python-Code



The screenshot shows a JupyterHub interface with the following details:

- Title Bar:** Instances | EC2 Management Con... | dhbw_stuttgart_data_mining | workshop_data_mining - Jupyter | dhbw_stuttgart_data_mining/workshop_data_mining.ipynb
- Header:** Not Secure | ec2-35-158-133-141.eu-central-1.compute.amazonaws.com/user/admin/notebooks/dhbw_stuttgart_data_mining/workshop_data_mining.ipynb | Logout | Control Panel | Not Trusted | Python 3 | Memory: 126 MB
- Content Area:**
 - DHBW Logo:** Duale Hochschule Baden-Württemberg
 - Section 1: Data Mining**
 - Studiengang: Wirtschaftsingenieurwesen (6. Semester)
 - Dozent: Tin Votan
 - Datum: 21.04.2020
 - Section 2: 1. Python-Module importieren**

```
In [ ]: # To support both python 2 and python 3
from __future__ import division, print_function, unicode_literals

# Common imports
import numpy as np
import os

# to make this notebook's output stable across runs
np.random.seed(42)

# to plot pretty figures
import matplotlib.pyplot as plt
plt.rcParams['axes.labelsize'] = 14
plt.rcParams['xtick.labelsize'] = 12
plt.rcParams['ytick.labelsize'] = 12

# Where to save the figures
PROJECT_ROOT_DIR = '.'
CHAPTER_ID = "end_to_end_project"
IMAGES_PATH = os.path.join(PROJECT_ROOT_DIR, "images", CHAPTER_ID)

def save_fig(fig_id, tight_layout=True, fig_extension="png", resolution=300):
    path = os.path.join(IMAGES_PATH, fig_id + "." + fig_extension)
    print("Saving figure", fig_id)
    if tight_layout:
        plt.tight_layout()
    plt.savefig(path, format=fig_extension, dpi=resolution)

# Ignore useless warnings (see SciPy issue #5998)
import warnings
warnings.filterwarnings(action="ignore", message="internal gelsd")
```
 - Section 3: 2. Datensätze herunterladen**

Python-Skript zum Herunterladen der Datensätze, Erstellen einer Ordnerstruktur und Extrahieren der CSV-Datei.

Wann ist es sinnvoll ein Data-Scraping-Tool in Python zu programmieren?

 - Bei Änderungen der Datensätze hilft ein automatisiertes Skript die Daten unkompliziert und in der selben Ordnerstruktur herunterzuladen
 - Datensätze werden auf mehreren Rechnern benötigt (Multiple-User)

```
In [ ]: import os
import tarfile
```



Strg + Eingabe



Ctrl + Eingabe