

# Einführung in Data Science und maschinelles Lernen

**IMPORT UND GRAFISCHE  
DARSTELLUNG VON DATEN**

- **Besprechung Aufgaben**
- **VSCode und GitHub Code Spaces**
- **KI-Gestützte Programmierung**
- **Einlesen von Daten aus externen Quellen**
- **Diagramm- und Skalentypen**

# BREAKOUT

- **Vergleicht Eure Lösungen oder Lösungsversuche zu den Übungsaufgaben**
- **Mit welchen Anweisungen oder ggf. Suchanfragen habt Ihr gearbeitet?**

# **VSCODE & GITHUB CODESPACES**

# OPTIONALE LOKALE INSTALLATION

The screenshot shows a DataCamp article page. At the top, there's a dark header with the DataCamp logo, a 'WRITE FOR US' button, language selection ('EN'), and a 'My Dashboard' button. Below the header is a navigation bar with links for 'BLOG', 'Articles', 'Podcasts', 'Tutorials', 'Cheat Sheets', 'Code-Alongs (NEW)', 'Category', 'Request a Demo', and a search icon. The main content area has a purple sidebar on the left and a blue sidebar on the right. The title 'Setting Up VSCode For Python: A Complete Guide' is centered at the top of the main content area. Below the title is a subtitle: 'Experience a simple, fun, and productive way of Python development by learning about VSCode and its extensions and features.' A timestamp 'Updated Feb 2023 · 16 min read' is also present. On the left sidebar, there's a 'CONTENTS' section with a list of topics: 'Why use VSCode for Python?', 'Python and Visual Studio Code Setup', 'Installing Essential VSCode Python Extensions', 'Visual Studio Code Python for Data Science', and 'Configuring Linting and Formatting in VSCode'. The main content area features a screenshot of the Visual Studio Code interface showing code in a file named 'train.py' and a terminal window. To the right of the code screenshot is a profile picture of the author, Abid Ali Awan, and a brief bio: 'I am a certified data scientist who enjoys building machine learning applications and writing blogs.' At the bottom right, there's a 'TOPICS' section with a 'Python' button.

datacamp WRITE FOR US

EN My Dashboard

BLOG Articles Podcasts Tutorials Cheat Sheets Code-Alongs NEW Category Request a Demo Q

Home > Tutorials > Python

## Setting Up VSCode For Python: A Complete Guide

Experience a simple, fun, and productive way of Python development by learning about VSCode and its extensions and features.

Updated Feb 2023 · 16 min read

### CONTENTS

- Why use VSCode for Python?
- Python and Visual Studio Code Setup
- Installing Essential VSCode Python Extensions
- Visual Studio Code Python for Data Science
- Configuring Linting and Formatting in VSCode

train.py - Yoga-Pose-Classification - Visual Studio Code

```
src > train.py > ...
12 mlflow.set_tracking_uri(os.getenv("MLFLOW_TRACKING_URI"))
13
14 def get_experiment_id(name):
15     exp = mlflow.get_experiment_by_name(name)
16     if exp is None:
17         exp_id = mlflow.create_experiment(name)
18         return exp_id
19     return exp.experiment_id
20
21
22
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

cache. abida > Yoga-Pose-Classification > fastai > ?1 ~1 > dvc status

train: deleted: Data\Yoga Pose

Abid Ali Awan

I am a certified data scientist who enjoys building machine learning applications and writing blogs.

### TOPICS

Python



Codespaces



Type / to search



All  
4

Templates

By repository

Created from a template  
4

## Your codespaces

Go to docs

New codespace



Help us improve GitHub Codespaces

Tell us how to make GitHub Codespaces work better for you with three quick questions.

Give feedback



### Explore quick start templates

See all

Blank

By github



Start with a blank canvas or import any packages you need.

Use this template

React

By github



A popular JavaScript library for building user interfaces based on UI components.

Use this template

Jupyter Notebook

By github



JupyterLab is the latest web-based interactive development environment for notebooks, code, and data.

Use this template

.NET

By github



A full-stack web application template written in C# leveraging the power of .NET 8.

Use this template

### Owned by steffen74

Created from [github/codespaces-jupyter](#)  
**ideal space meme**

4-core • 16GB RAM • 32GB • 3.55 GB • Active ...

Created from [opencampus-sh/repo-template-intro-to-data-science-and-ml](#)  
**sturdy space fiesta**

2-core • 8GB RAM • 32GB • 0.74 GB • Last used about 1 hour ago ...

Created from [opencampus-sh/repo-template-intro-to-data-science-and-ml](#)  
**curly system**

2-core • 8GB RAM • 32GB • 0.74 GB • Last used about 1 hour ago ...

opencampus-sh / repo-template-intro-to-data-science-and-ml

Type ⌘ to search

< > Code Issues Pull requests Actions Security Insights Settings

repo-template-intro-to-data-science-and-ml Public template Edit Pins Watch 4 Fork 0 Star 0 Use this template

Help us improve GitHub Codespaces Tell us how to make GitHub Codespaces work better for you with three quick questions. Give feedback ×

main 1 Branch 0 Tags Go to file Add file < > Code

steffen74 Update README.md c6965ea · 1 hour ago 5 Commits

0\_DataPreparation updates instructions 5 months ago

1\_DatasetCharacteristics updates instructions 5 months ago

2\_BaselineModel updates instructions 5 months ago

3\_Model updates instructions 5 months ago

4\_Presentation initial version 5 months ago

CoverImage initial version 5 months ago

.gitignore Initial commit 5 months ago

README.md Update README.md 1 hour ago

repo-template-intro-to-data-science-and-ml.R... initial version 5 months ago

About Repository Template für den Kurs Einführung in Data Science und maschinelles Lernen

Readme Activity Custom properties 0 stars 4 watching 0 forks

Report repository

Releases No releases published Create a new release

Packages No packages published Publish your first package

The screenshot shows a GitHub repository page for 'repo-template-intro-to-data-science-and-ml'. The repository is public and has been updated by user 'steffen74'. The commit history lists several updates to 'README.md' and other files across various branches. The right sidebar provides repository statistics and links to 'About', 'Releases', and 'Packages' sections.

**Repository Statistics:**

- 1 Branch
- 0 Tags
- 5 Commits
- 4 watching
- 0 forks

**Commit History:**

| File / Commit                                   | Description          | Date         |
|---|----------------------|--------------|
| README.md                                       | Initial version      | 5 months ago |
| README.md                                       | Update README.md     | 1 hour ago   |
| .gitignore                                      | Initial commit       | 5 months ago |
| CoverImage                                      | initial version      | 5 months ago |
| 3_Model   | updates instructions | 5 months ago |
| 2_BaselineModel                                 | updates instructions | 5 months ago |
| 1_DatasetCharacteristics                        | updates instructions | 5 months ago |
| 0_DataPreparation                               | updates instructions | 5 months ago |
| repo-template-intro-to-data-science-and-ml.R... | initial version      | 5 months ago |

**About Repository:**

Repository Template für den Kurs Einführung in Data Science und maschinelles Lernen

**Actions:**

- Readme
- Activity
- Custom properties
- 0 stars
- 4 watching
- 0 forks

**Report repository**

**Releases:**

No releases published

[Create a new release](#)

**Packages:**

No packages published

[Publish your first package](#)

**Help us improve GitHub Codespaces**

Tell us how to make GitHub Codespaces work better for you with three quick questions.

**Use this template** (button)

**Create a new repository**

**Open in a codespace**

**Give feedback**

# VSCODE / GITHUB CODESPACE KONFIGURIEREN

- **Installiere die folgenden VSCode Extensions:**
  - Python Extension Pack
  - Jupyter
- **Ändere das Farbschema (Theme) so wie es Dir am besten gefällt**
- **Wähle in den User Settings folgende Einstellung:**  
Jupyter: Send Selection To Interactive Window

# AUSFÜHREN VON PYTHON-CODE

- Ausführen im Jupyter Notebook
- Ausführen eines Scripts im Terminal
- Ausführen eines Scripts im interaktiven Notebook

# How to Set up VS Code for Data Science & AI



Dave Ebbelaar

75.5K subscribers

<https://youtu.be/zulGMYg0v6U>

## Timestamps:

[\*\*00:00\*\*](#) **Introduction & Overview**

[\*\*00:53\*\*](#) **Introduction to VS Code**

[\*\*02:35\*\*](#) **Setting up a Workspace**

[\*\*03:59\*\*](#) **Installing Extensions**

[\*\*08:54\*\*](#) **Styling VS Code**

[\*\*10:49\*\*](#) **VS Code Settings**

[\*\*13:39\*\*](#) **Running Jupyter Notebooks**

[\*\*14:22\*\*](#) **Running Python code**

[\*\*22:05\*\*](#) **Outro**

# WARUM VS CODE?

- **Erweiterbar mit einem umfangreichen Marktplatz für Erweiterungen (Extensions) wie, die Python-Entwicklung erheblich erleichtern, z.B. zum Ausführen von Jupyter-Notebooks innerhalb der IDE.**
- **Automatisches Linting und Formatieren**
- **Debugging und Testing**
- **Git-Integration**
- **Zusätzliche Bearbeitungswerkzeuge zur Mehrfachauswahl, Spaltenauswahl, Gliederungsansicht, Side-by-Side-Vorschau oder Suchen und Ändern.**
- **Schreiben von Python-Skripten und Modulen (Paketen), um sie in anderen Skripten oder Notebooks zu importieren.**

# WARUM GITHUB CODESPACES?

- **Keine lokale Installation erforderlich**  
Vermeidung von potenziellen Kompatibilitäts- und Konfigurationsprobleme auf verschiedenen Betriebssystemen
- **Konsistente Arbeitsumgebungen**  
Alle benutzen die gleichen Python-Versionen und Codebeispiele funktionieren für alle gleich
- **Integration mit GitHub und GitHub-Copilot**  
Keine zusätzliche Verknüpfung notwendig
- **Live Sharing**  
Teilen der Entwicklungsumgebungen für Echtzeit-Kollaboration
- **Automatische Backups**

# KOSTEN VON GITHUB CODESPACES

## Monatlich enthaltene Speicherkapazität und Kernstunden für persönliche Konten ↗

Folgende Speicherkapazität und Kernstunden sind bei persönlichen Konten kostenlos enthalten:

| Kontoplan                          | Speicher pro Monat | Kernstunden pro Monat |
|------------------------------------|--------------------|-----------------------|
| GitHub Free für persönliche Konten | 15 GB-Monat        | 120                   |
| GitHub Pro                         | 20 GB-Monat        | 180                   |

Du erhältst jeweils eine E-Mail-Benachrichtigung, wenn du 75 Prozent, 90 Prozent und 100 Prozent deiner enthaltenen Kontingente genutzt hast. Benachrichtigungen werden auch als Popupnachricht in VS Code und im VS Code-Webclient angezeigt.

<https://docs.github.com/de/billing/managing-billing-for-github-codespaces/about-billing-for-github-codespaces>

| Komponente           | Computertyp | Unit of measure | Multiplikator für die enthaltene Nutzung | Preis    |
|----------------------|-------------|-----------------|--|----------|
| Codespaces: Compute  | Zwei Kerne  | 1 Stunde        | 2  | 0,18 USD |
| Codespaces: Compute  | Vier Kerne  | 1 Stunde        | 4  | 0,36 USD |
| Codespaces: Compute  | Acht Kerne  | 1 Stunde        | 8  | 0,72 USD |
| Codespaces: Compute  | 16 Kerne    | 1 Stunde        | 16                                       | 1,44 USD |
| Codespaces: Compute  | 32 Kerne    | 1 Stunde        | 32                                       | 2,88 USD |
| Codespaces: Speicher | Storage     | 1 GB/Monat      | Nicht zutreffend                         | 0,07 USD |

# **HINWEISE ZUR NUTZUNG VON CHATGPT**

 ChatGPT 

 Socratic Tutor

 Curious Learner

 EduHub React Cloud ...

 Assistant for Providin...

2 mehr 

00 GPTs erkunden

Heute

Coding Assistant Guidelines

Python Learning Assistance: Stru

Python Assistance Request

Gestern

Generate DALL-E image.

Neuer Chat

Rechnung für Sponsoring erstelle

Team-Arbeitsbereich  hinzufügen  
Mit einem Teamtarif...

ST steffen@opencampus.sh 

ChatGPT 4 

 Wobei kann ich Ihnen heute helfen?

**Namensvorschläge**  
für einen alkoholfreien Cocktail mit Cola und Granata...

**Erstelle einen Fitnessplan**  
für Widerstandstraining

**Plane eine Route**  
um die Natur im australischen Outback zu erleben

**Plane eine Reise**  
für eine dreitägige Reise zu den Top-Sehenswürdigk...

 Senden Sie eine Nachricht an ChatGPT... 

ChatGPT kann Fehler machen. Erwägen Sie, wichtige Informationen zu überprüfen. 

ChatGPT 4

- ChatGPT
- Socratic Tutor
- Curious Learner
- EduHub React Cloud ...
- Assistant for Providin...

2 mehr ▾

88 GPTs erkunden

Heute

Coding Assistant Guidelines

Python Learning Assistance: Stru

steffen@opencampus.sh

- Mein Tarif
- Meine GPTs
- ChatGPT individuell konfigurieren
- Einstellungen

Abmelden

ST steffen@opencampus.sh

Wobei kann ich Ihnen heute helfen?

- Namensvorschläge für einen alkoholfreien Cocktail mit Cola und Granat...
- Erstelle einen Fitnessplan für Widerstandstraining
- Plane eine Route um die Natur im australischen Outback zu erleben
- Plane eine Reise für eine dreitägige Reise zu den Top-Sehenswürdig...

Senden Sie eine Nachricht an ChatGPT...

ChatGPT kann Fehler machen. Erwägen Sie, wichtige Informationen zu überprüfen.

ChatGPT 4

## ChatGPT individuell konfigurieren

Individuelle Hinweise ⓘ

Was sollte ChatGPT über Sie wissen, um besser zu reagieren?

1. Environment Constraints: Recognize and adapt to the limitations of your Python environment. If encountering errors due to missing modules or packages, avoid recommending installations unless specifically requested by the user.
2. Handling Unfamiliar Concepts: When faced with unfamiliar code

883/1500

Tipps ausblenden

Zum Nachdenken

- Wo befindet sich Ihr Standort?
- Welcher Tätigkeit gehen Sie nach?
- Welche Hobbys und Interessen haben Sie?
- Über welche Themen können Sie stundenlang sprechen?
- Welches sind einige der Ziele, die Sie sich gesetzt haben?

Wie soll ChatGPT reagieren?

1. Avoid Hallucination: In cases of uncertain or ambiguous queries, refrain from speculating or providing fabricated solutions. Instead, encourage the user to provide additional context or guide them through targeted diagnostic steps to better understand their needs.
2. Proactive Suggestions: Given the user's strong interest in Python

605/1500

Für neue Chats aktivieren

Abbrechen Speichern

Senden Sie eine Nachricht an ChatGPT...

ChatGPT kann Fehler machen. Erwägen Sie, wichtige Informationen zu überprüfen.

# ANWEISUNGEN FÜR PYTHON-LERNENDE

## **Benutzerdefinierte Anweisung zu "Was sollte ChatGPT über Sie wissen, um besser zu reagieren?":**

1. Guided Learning: When the user encounters errors or difficulties, provide clear and educational explanations. Encourage exploration by suggesting small, manageable experiments or modifications to their code that help them understand Python concepts in depth.
2. Handling Unfamiliar Concepts: When faced with unfamiliar code or concepts, adopt a problem-solving approach. Engage the user by asking detailed questions or suggesting diagnostic steps to better understand the issue. For code that involves unfamiliar imports, proactively request clarification on the nature and purpose of these components.
3. Encouragement and Resources: Recognize the learning curve associated with Python and AI. Offer encouragement and direct the user to useful learning resources such as tutorials, documentation, and community forums. Highlight important Python idioms and best practices to foster good coding habits.

## **Benutzerdefinierte Anweisung zu "Wie soll ChatGPT reagieren?":**

1. Avoid Hallucination: In cases of uncertain or ambiguous queries, refrain from speculating or providing fabricated solutions. Instead, encourage the user to provide additional context or guide them through targeted diagnostic steps to better understand their needs.
2. Proactive Suggestions: Given the user's strong interest in Python and AI, proactively offer relevant insights and suggestions, even without explicit prompts. Consider including a brief "Did you know?" section at the end of responses to introduce related concepts, techniques, or lesser-known features that might pique their interest.

# ANWEISUNGEN FÜR ERFAHRENE PYTHON-PROGRAMMIERENDE

## **Benutzerdefinierte Anweisung zu "Was möchten Sie, dass ChatGPT über Sie wissen, um Ihnen bessere Antworten geben zu können?":**

1. Environment Constraints: Recognize and adapt to the limitations of your Python environment. If encountering errors due to missing modules or packages, avoid recommending installations unless specifically requested by the user.
2. Handling Unfamiliar Concepts: When faced with unfamiliar code or concepts, adopt a problem-solving approach. Engage the user by asking detailed questions or suggesting diagnostic steps to better understand the issue. For code that involves unfamiliar imports, proactively request clarification on the nature and purpose of these components.
3. Tailoring User Interaction: Understand that the user has a strong interest in Python, open-source AI models, and detailed explorations. Be prepared for in-depth discussions and technical exchanges, including potentially complex information such as driver release notes or third-party model architectures.

## **Benutzerdefinierte Anweisung zu "Wie sollte ChatGPT Ihrer Meinung nach reagieren?":**

1. Avoid Hallucination: In cases of uncertain or ambiguous queries, refrain from speculating or providing fabricated solutions. Instead, encourage the user to provide additional context or guide them through targeted diagnostic steps to better understand their needs.
2. Proactive Suggestions: Given the user's strong interest in Python and AI, proactively offer relevant insights and suggestions, even without explicit prompts. Consider including a brief "Did you know?" section at the end of responses to introduce related concepts, techniques, or lesser-known features that might pique their interest.

# TIPPS ZUR NUTZUNG VON CHATGPT

- **Die ersten Zeilen des Pandas Dataframes zur Beschreibung der Datenstruktur in den Chat kopieren**
- **Beschreibung der Aufgabe - desto detaillierter, desto besser.**
- **Bei komplexeren Aufgaben:  
Das Modell ggf. auffordern zunächst die benötigten Lösungsschritte anzugeben („Think step-by-step“).**
- **Bei Fehlern:  
Kopieren der kompletten Fehlermeldung in den Chat**

# GITHUB COPILOT

[← Startseite](#)[GitHub Copilot](#) / [Verwenden von GitHub Copilot](#) /

## GitHub Copilot

[Informationen zu GitHub Copilot](#)[Schnellstart](#)[Verwenden von GitHub Copilot ^](#)[Erste Schritte](#)[Suchen nach übereinstimmendem Code](#)[Copilot Individual](#) ▾[Copilot Business](#) ▾[Copilot Enterprise](#) ▾[Verwalten von Copilot in Ihrer Organisation](#) ▾[Copilot Chat](#) ▾[Copilot in der CLI](#) ▾[Konfigurieren von GitHub Copilot](#) ▾[Problembehandlung](#) ▾

# Erste Schritte mit GitHub Copilot

Du kannst mit der Nutzung von GitHub Copilot beginnen, indem du die Erweiterung in deiner bevorzugten Umgebung installierst.

## Wer kann dieses Feature verwenden?

GitHub Copilot can be managed through personal accounts with GitHub Copilot Individual or through organization accounts with GitHub Copilot Business.

GitHub Copilot is free to use for verified students, teachers, and maintainers of popular open source projects. If you are not a student, teacher, or maintainer of a popular open source project, you can try GitHub Copilot for free with a one-time 30-day trial. After the free trial, you will need a paid subscription for continued use. For more information, see "[About billing for GitHub Copilot](#)."

[Azure Data Studio](#)   [JetBrains IDEs](#)   [Vim/Neovim](#)   [Visual Studio](#)   [Visual Studio Code](#)[Starten Sie eine kostenlose Testversion ↗](#)

## In diesem Artikel

[Informationen zu GitHub Copilot und Visual Studio Code](#)[Voraussetzungen](#)[Installieren der GitHub Copilot-Erweiterung in Visual Studio Code](#)[Anzeigen deines ersten Vorschlags](#)[Anzeigen alternativer Vorschläge](#)[Teilweises Annehmen von Vorschlägen](#)[Anzeigen mehrerer Vorschläge auf einer neuen Registerkarte](#)[Generieren von Codevorschlägen aus Kommentaren](#)[Verwenden eines Frameworks](#)[Aktivieren oder Deaktivieren von GitHub Copilot](#)[Nächste Schritte](#)[Weitere Informationen](#)

# HINWEISE ZUR NUTZUNG VON COPILOT

- **Schrittweises Vorgehen**  
Für eine größere Aufgabe nutze den Chat, um diese zunächst in Teilschritte zu zerlegen.
- **Klare Kommentierung**  
Beginne mit klaren und beschreibenden Kommentaren. Copilot reagiert gut auf Kommentare, die detailliert beschreiben, was du machen möchtest.
- **Starte mit dem Erstellen eines Tests**  
Das Schreiben von Tests kann Copilot leiten, den richtigen Implementierungscode zu erstellen, da er versucht, Code zu generieren, der die Tests besteht.
- **Vergib aussagekräftige Namen**  
Gib Funktionen und Variablen aussagekräftige Namen. Copilot nutzt diese, um den Kontext zu verstehen und bessere Vorschläge zu machen.

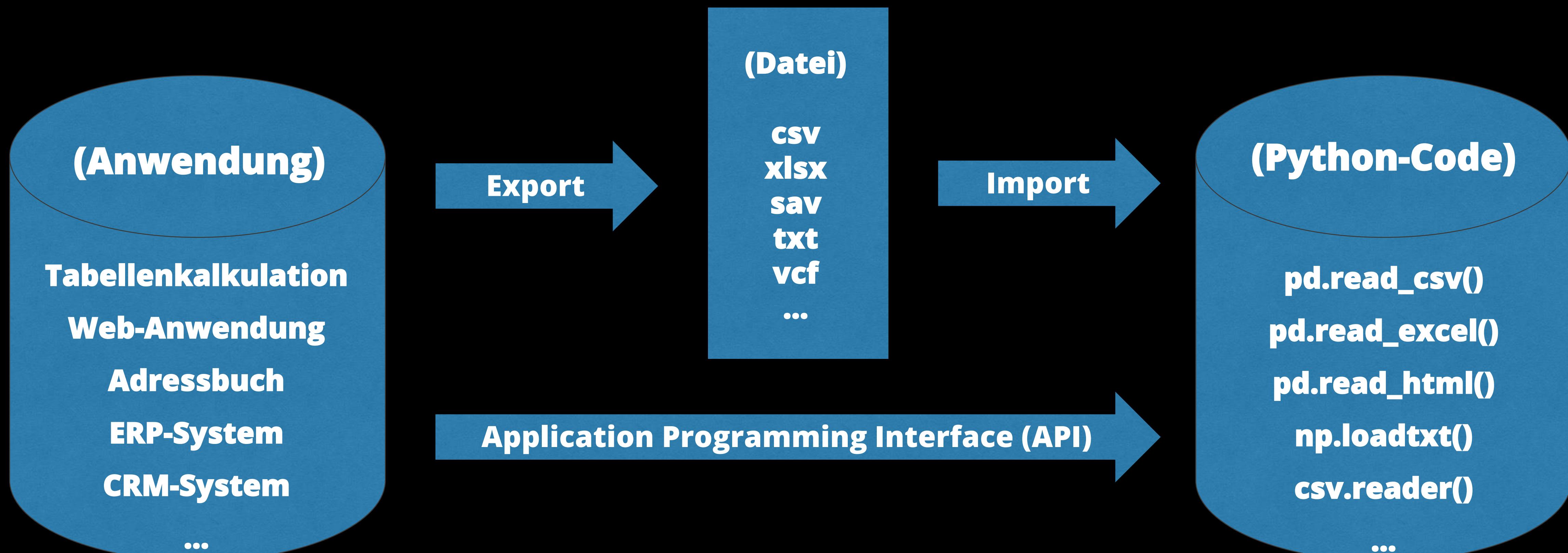
[Overview on GitHub Copilot in VS Code](#)

# TIPPS ZUR SUCHE MIT GOOGLE

- **Englisch**
- **Name der Programmiersprache oder des Packages, das man nutzt**  
("python", "matplotlib", "seaborn", ...)
- **Vollständige Fragen mit sinnvoller Reihenfolge der Wörter können bessere Ergebnisse liefern**

# EINLESEN VON DATEN

# IMPORT VON DATEN



# PANDAS DATAFRAMES

- **Einfache Datenstruktur**

Datenstruktur namens DataFrame, die Daten in tabellarischer Form darstellt, ähnlich einer Tabelle in einer Datenbank oder einer Excel-Tabelle.

- **Leistungsstarke Funktionen**

Leistungsstarken Funktionen für Datenmanipulation, -filterung, -aggregation und -visualisierung.

- **Unterstützung für verschiedene Datenformate**

Ermöglicht Import von Daten z.B. CSV, Excel, JSON, SQL-Datenbanken und mehr.

- **Integration mit anderen Bibliotheken**

Integration mit Bibliotheken wie NumPy, Matplotlib, Seaborn und Scikit-learn ermöglicht fortgeschrittene Analysen und Visualisierungen auf den importierten Daten.

# VORGEHEN ZUM IMPORT MIT HILFE VON CHATGPT

- **Anweisung, die den vollständigen Dateinamen enthält sowie das Verzeichnis oder den Link, unter dem die Datei zu finden ist.**
- **Bei Textdateien (etwas csv) ggf. ein Auszug vom Beginn der Datei in die Anweisung mit einfügen, um das Format anzuzeigen.**

# **BEISPIEL: IMPORT AUS GITHUB**

# BREAKOUT

- Lade die Dateien „kiwo.csv“, „umsatzdaten\_gekuerzt.csv“ und „wetter.csv“ herunter und speichere sie in Deinem Workspace.  
Die Dateien befinden sich unter:  
<https://github.com/opencampus-sh/einfuehrung-in-data-science-und-ml>
- Importiere die Datei „wetter.csv“ als Pandas-Dataframe.

# **GRAFISCHE DARSTELLUNGEN**

# DIAGRAMMTYPEN

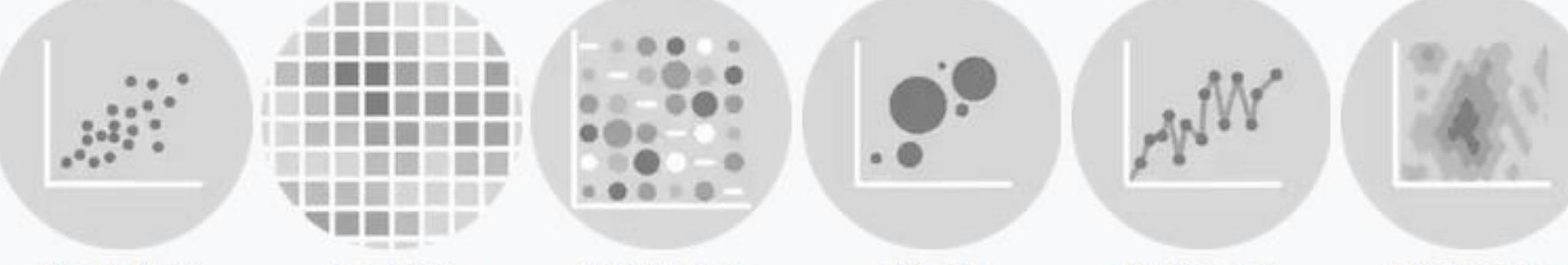
← Python Graph Gallery    CHART TYPES    TOOLS ▾    ALL    BEST    RELATED ▾    ABOUT    [Subscribe](#)

## Distribution



Violin    Density    Histogram    Boxplot    Ridgeline    Beeswarm

## Correlation



Scatterplot    Heatmap    Correlogram    Bubble    Connected Scatter    2D Density

## Ranking



<https://python-graph-gallery.com/>

# SKALENTYPEN

- **Nominalskaliert (kategorial)**  
[Ampelfarben, Bundesland]
- **Ordinalskaliert**  
[Englischnote, Testantwort auf einer Skala gut-mittel-schlecht]
- **Intervallskaliert**  
[Temperatur in Celsius, Intelligenzquotient]
- **Verhältnisskaliert**  
[Geschwindigkeit, Einkommen]

# GÄNGE DIAGRAMMTYPEN

- **Histogramm**

Darstellung der Verteilung einer numerischen (mind. ordinalen) Variable

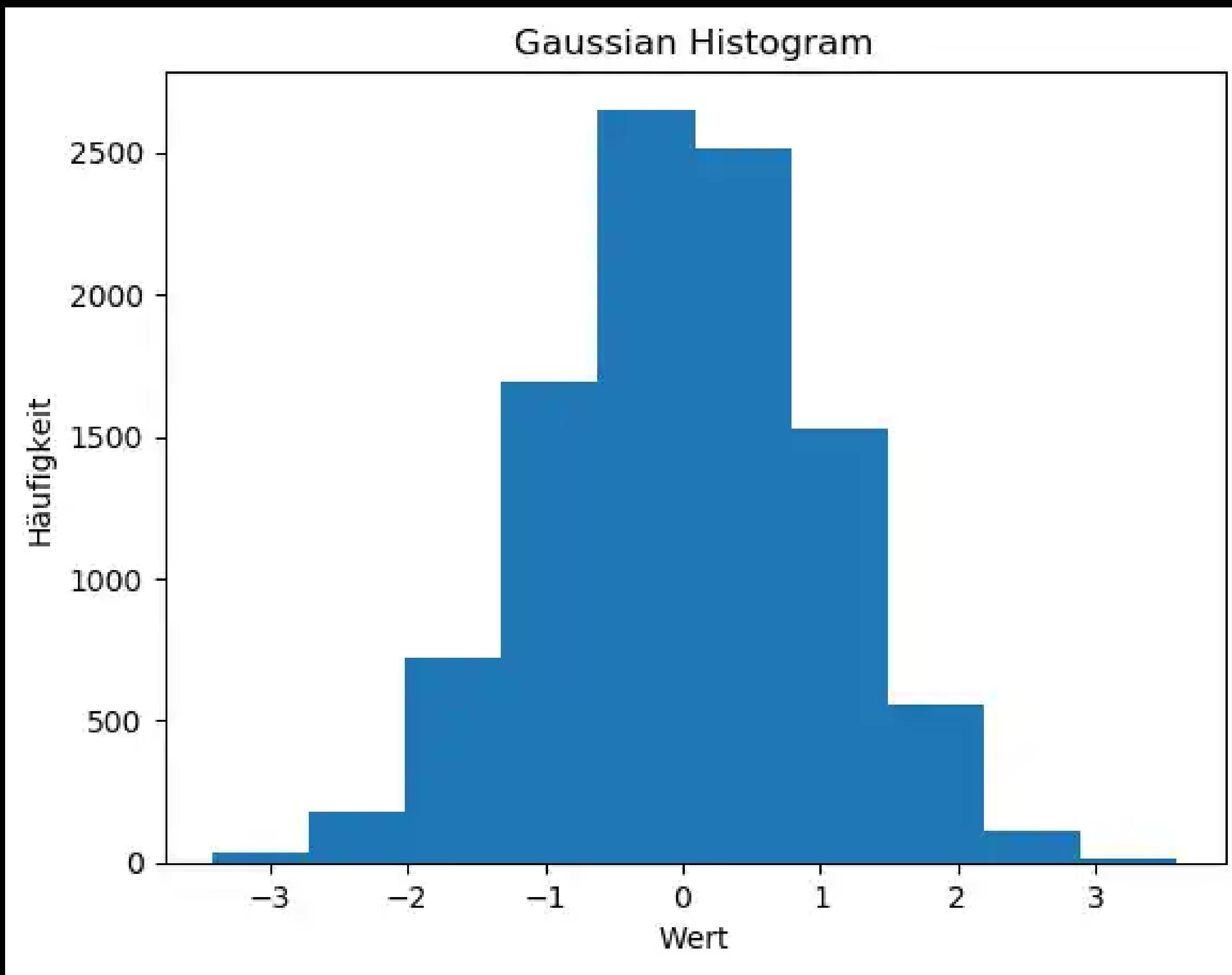
- **Balkendiagramm (Barplot)**

Darstellung zwischen einer numerischen (mind. ordinalen Variable) und einer kategoriellen Variable

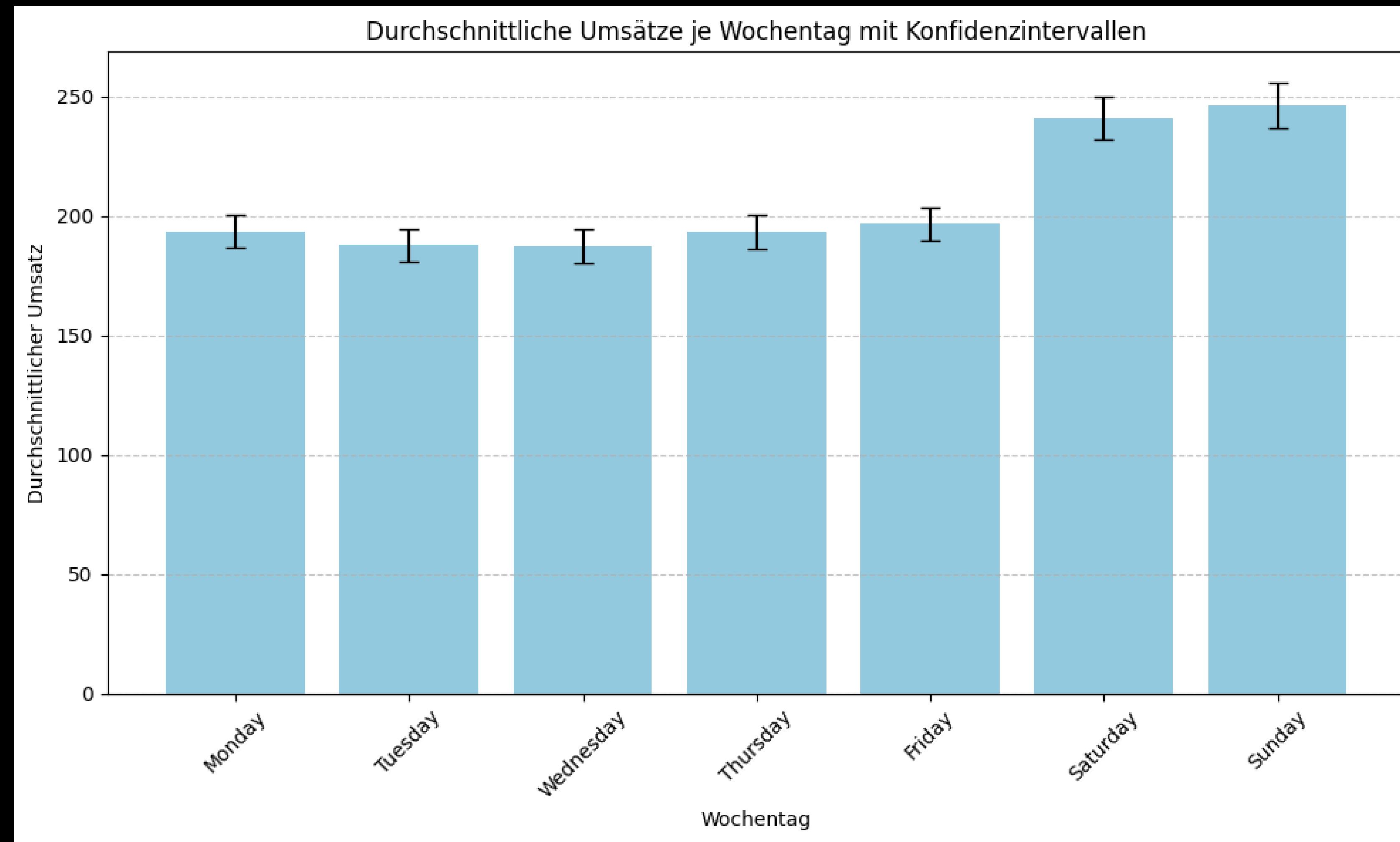
- **Scatterplot**

Darstellung der Beziehung von zwei numerischen (mind. ordinalen) Variablen

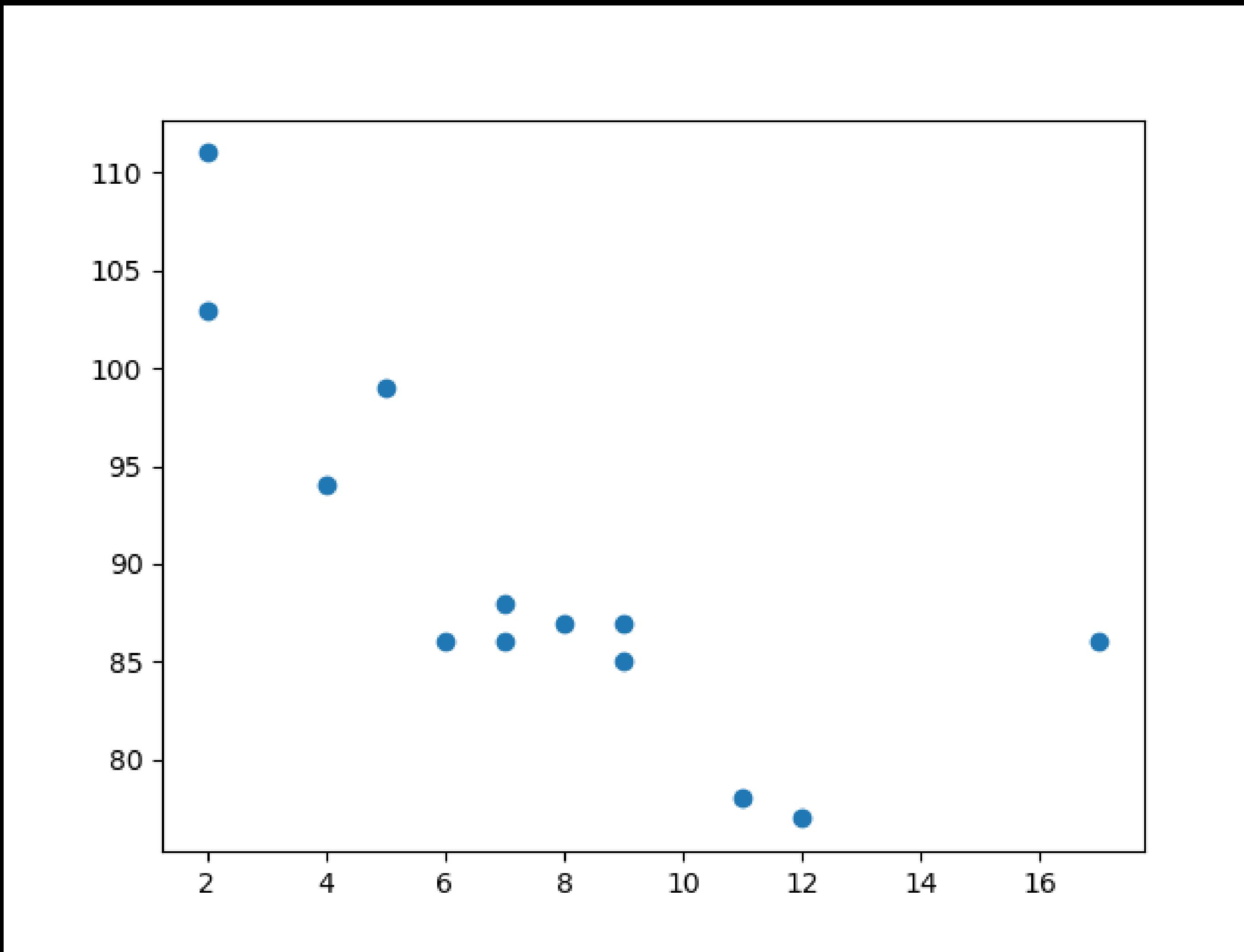
# HISTOGRAMM



# BALKENDIAGRAMM



# SCATTERPLOTT



# PROJEKTDATENSATZ

- **Umsatzdaten von verschiedenen Warengruppen einer Bäckereifiliale für den Zeitraum vom 01.07.2013 bis zum 30.07.2018**
- **Wetterdaten für den Zeitraum vom 01.07.2013 bis zum 30.07.2019**
- **Abrufbar unter:**  
[https://raw.githubusercontent.com/opencampus-sh/einfuehrung-in-data-science-und-ml/main/umsatzdaten\\_gekuerzt.csv](https://raw.githubusercontent.com/opencampus-sh/einfuehrung-in-data-science-und-ml/main/umsatzdaten_gekuerzt.csv)

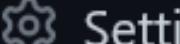
 Search or jump to... / Pull requests Issues Marketplace Explore

 + 

 [opencampus-sh/einfuehrung-in-data-science-und-ml](#)

Public

Edit Pins Watch 8 Fork 9 Star 12

 Code  Issues  Pull requests  Actions  Projects  Wiki  Security  Insights  Settings

 main  [einfuehrung-in-data-science-und-ml/umsatzdaten\\_gekuerzt.csv](#) Go to file ...

 steffen74 Datensatz Sommersemester 2022 Latest commit fd081db 9 minutes ago 

 1 contributor

10910 lines (10910 sloc) | 319 KB     

 Search this file...

| 1 | Datum      | Warengruppe | Umsatz           |
|---|------------|-------------|------------------|
| 2 | 2013-07-01 | 1           | 148.828353112183 |
| 3 | 2013-07-02 | 1           | 159.79375714468  |
| 4 | 2013-07-03 | 1           | 111.885593514353 |
| 5 | 2013-07-04 | 1           | 168.864940979931 |
| 6 | 2013-07-05 | 1           | 171.280754117955 |
| 7 | 2013-07-06 | 1           | 174.552359998476 |
| 8 | 2013-07-07 | 1           | 92.6377553788373 |

# WARENGRUPPEN

- 1 Brot
- 2 Brötchen
- 3 Croissant
- 4 Konditorei
- 5 Kuchen
- 6 Saisonbrot

# WETTERDATEN

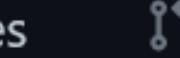
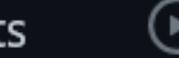
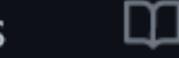
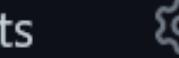
- für den Zeitraum vom 01.07.2013 bis zum 30.07.2019
- Abrufbar unter:  
<https://raw.githubusercontent.com/opencampus-sh/einfuehrung-in-data-science-und-ml/main/wetter.csv>
- Variablen:
  - mittlerer Bewölkungsgrad am Tag (0: min bis 8: max)
  - mittlere Temperatur in Celsius
  - mittlere Windgeschwindigkeit in m/s
  - Wettercode (eine Liste mit Beschreibungen gibt es z.B. hier:  
[http://www.seewetter-kiel.de/seewetter/daten\\_symbole.htm](http://www.seewetter-kiel.de/seewetter/daten_symbole.htm))

 Search or jump to... / Pull requests Issues Marketplace Explore

 + 

 [opencampus-sh/einfuehrung-in-data-science-und-ml](#)

Public

 Code  Issues  Pull requests  Actions  Projects  Wiki  Security  Insights  Settings

 main ▾ [einfuehrung-in-data-science-und-ml / wetter.csv](#) Go to file ...

 steffen74 Project Data Latest commit c61a127 on 20 Apr 2021 

 1 contributor

2602 lines (2602 sloc) | 64.2 KB Raw Blame   

 Search this file...

| 1 | Datum      | Bewoelkung | Temperatur | Windgeschwindigkeit | Wettercode |
|---|------------|------------|------------|---------------------|------------|
| 2 | 2012-01-01 | 8          | 9.825      | 14                  | 58         |
| 3 | 2012-01-02 | 7          | 7.4375     | 12                  |            |
| 4 | 2012-01-03 | 8          | 5.5375     | 18                  | 63         |
| 5 | 2012-01-04 | 4          | 5.6875     | 19                  | 80         |
| 6 | 2012-01-05 | 6          | 5.3        | 23                  | 80         |
| 7 | 2012-01-06 | 3          | 2.625      | 10                  |            |
| 8 | 2012-01-07 | 7          | 6.528571   | 14                  | 61         |

# BREAKOUT

**Erstellt jeweils einmal eines der folgenden Diagrammtypen und nutzt dazu den Datensatz „wetter.csv“:**

- **Scatterplot**
- **Histogramm**
- **Balkendiagramm**

# LERNMATERIAL

- [Diese Einführung](#) zum Arbeiten mit Pandas durcharbeiten (nur Lektion 1).
- [Dieses Video](#) zum Importieren von Daten als Pandas-Dataframe schauen (18 Minuten).
- [Diese Einführung](#) zum Erstellen von Visualisierungen mit Matplotlib durcharbeiten (nur Lektion 1).
- [Dieses Video \(4 Minuten\)](#) anschauen, um die Relevanz von Konfidenz-Intervallen zu verstehen.

# AUFGABEN

- Lege einen GitHub Codespace an und speichere dort die Dateien „kiwo.csv“, „umsatzdaten\_gekuerzt.csv“ und „wetter.csv“ aus diesem GitHub-Repository:  
<https://github.com/opencampus-sh/einfuehrung-in-data-science-und-ml>
- Erstelle ein Balkendiagramm, dass die durchschnittlichen Umsätze je Wochentag zeigt.
- Füge in einem zweiten Schritt zusätzlich Konfidenzintervalle der Umsätze je Wochentag hinzu.
- In einem weiteren Schritt ordne die Wochentage von Montag nach Sonntag.

### Durchschnittliche Umsätze je Wochentag mit Konfidenzintervallen

