

Einführung in Data Science und maschinelles Lernen

**IMPORT UND GRAFISCHE
DARSTELLUNG VON DATEN**

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

BREAKOUT

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

MUSTERLÖSUNG

HINWEISE ZUR VERWENDUNG VON KI-TOOLS

Gängige Vorgehensweise

- 1. Möglichst detaillierte der Aufgabe Beschreibung in KI-Tool (z.B. Claude) kopieren und ausführen**
- 2. Bei Fehlern: Fehlermeldung in KI-Tool eingeben und Lösung implementieren**
- 3. Schritte wiederholen bis Code funktioniert**

Potenzielle Probleme und Lösungsansätze

1. Mangelndes Codeverständnis

Problem: Code funktioniert, aber Funktionsweise unklar

Lösung:

- Code schrittweise vom KI-Tool erklären lassen
- Kontrollausgaben zur Überprüfung einzelner Schritte einbauen

2. Datenqualität

Problem: Ergebnisse können durch Datenfehler verfälscht sein

Lösung:

- Systematische Überprüfung der Eingabedaten
- Identifikation möglicher Anomalien (z.B. Sensorausfälle)
- Plausibilitätsprüfungen implementieren

BEST PRACTICES

- 1. Code immer mit Kommentaren erklären lassen**
- 2. Kontrollausgaben für wichtige Zwischenschritte einbauen**
- 3. Datenqualität systematisch prüfen, z.B.:**
 - Ausreißeranalyse („Outlier Detection“)
 - Prüfung auf fehlende Werte
 - Identifikation von Fehlern in der Datenerhebung (z.B. Messfehlern)

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, a navigation bar includes 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 above a description: 'Experience a simple, fun, and productive way of Python development by learning about VSCode and its extensionsn and features.' Below the description, it says 'Updated Feb 2023 · 16 min read'. On the left, a 'CONTENTS' sidebar lists several sections: '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 'train.py' file and the 'Extensions' sidebar. To the right of the code screenshot is a profile picture of the author, Abid Ali Awan, with his name 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 extensionsn 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
```

EXTENSIONS Search Extensions in Marketplace

- INSTALLED
- DVC
- GitLens — Git supercha...
- indent-rainbow
- isort
- Julia

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

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**

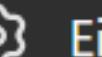
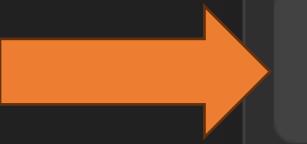
[**13:39**](#) **Running Jupyter Notebooks**

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

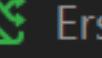
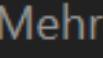
[**22:05**](#) **Outro**

HINWEISE ZUR NUTZUNG VON CHATGPT

Meine GPTs

 ChatGPT individuell konfigurieren Einstellungen Plus verlängern Abmelden

Wie kann ich dir helfen?

 Sende eine Nachricht an ChatGPT Erstelle ein Bild Analysiere Daten Erstelle einen Plan Mehr



ChatGPT ▾



ChatGPT

Socratic Tutor

Curious Learner

EduHub React Cloud ...

Assistant for Providin...

GPTs erkunden

Vorherige 30 Tage

Ghost Server Redirection Error

DNS Information Request

Transparent Background Edit

Oktober

Transparent Delete Button Style

Surboard ausschneiden und verg

Hasura Array Mutation Issues

September

Zuckerschoten auf Französisch

User Intent Clarification

Rappel téléphonique demandé

August

Check Hasura Time Fields

PDF zu TXT Konverter

Plus verlängern

ChatGPT individuell konfigurieren

Individuelle Hinweise ⓘ

Was sollte ChatGPT über dich 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

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

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.

GGF. BESSER: GPTS VON ANDEREN

+ Erstellen ST um eigene weiterzugeben

GPTs

Entdecke und erstelle individuelle ChatGPT-Versionen, die Hinweise, Zusatzwissen und Kombinationen aus Fähigkeiten vereinen.

Q In GPTs suchen

Highlights Schreiben Produktivität Recherche und Analyse Bildung Lifestyle Programmierung

Featured

Curated top picks from this week

 **Code Tutor**
Let's code together! I'm Khanmigo Lite, by Khan Academy. I won't write the code for you, but I'll hel...
Von khanacademy.org

 **Whimsical Diagrams**
Explains and visualizes concepts with flowcharts, mindmaps and sequence diagrams.
Von whimsical.com

 **Resume**
By combining the expertise of top resume writers with advanced AI, we assist in diagnosing and...
Von jobright.ai

 **Universal Primer**
The fastest way to learn anything hard.
Von Siqi Chen



Code Tutor

Von khanacademy.org

Let's code together! I'm Khanmigo Lite, by Khan Academy. I won't write the code for you, but I'll help you work things out. Can you tell me the challenge you're working on?

Help me with my homework assignment

How are you different than regular Khanmigo?

How can I improve my code's efficiency?

Help me understand this programming...

Senden einer Nachricht an Code Tutor



ChatGPT kann Fehler machen. Überprüfen Sie wichtige Informationen.

TIPPS ZUR NUTZUNG VON CHATBOTS

- **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**

NACHTEILE VON CHATBOTS

- **Ständiger Wechsel zwischen zwei Anwendungen**
- **Mühsames Copy&Paste der benötigten Informationen sowie des erhaltenen Codes**
- **Mühsames einfügen einzelner Zeilen in den vorhandenen Code**
- **Unklarheit darüber, was geändert wurde, bei Generierung komplett neuer Code-Abschnitte**

GITHUB COPILOT

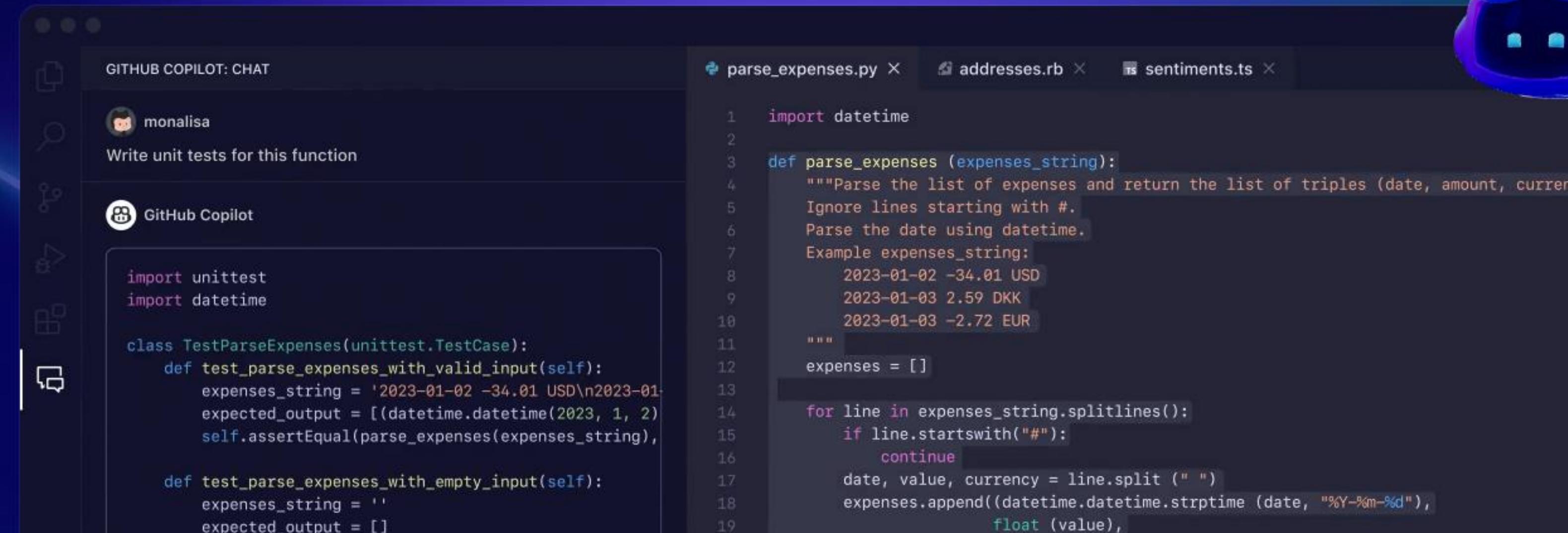


GitHub Copilot

The world's most widely adopted AI developer tool

[Start a free trial >](#)

[See plans & pricing >](#)



GITHUB COPILOT: CHAT

monalisa

Write unit tests for this function

GitHub Copilot

```
import unittest
import datetime

class TestParseExpenses(unittest.TestCase):
    def test_parse_expenses_with_valid_input(self):
        expenses_string = '2023-01-02 -34.01 USD\n2023-01-03 2.59 DKK
        expected_output = [(datetime.datetime(2023, 1, 2),
        self.assertEqual(parse_expenses(expenses_string),
        expected_output)

    def test_parse_expenses_with_empty_input(self):
        expenses_string = ''
        expected_output = []
        self.assertEqual(parse_expenses(expenses_string),
        expected_output)
```

parse_expenses.py x addresses.rb x sentiments.ts x

```
import datetime

def parse_expenses (expenses_string):
    """Parse the list of expenses and return the list of triples (date, amount, currency
    Ignore lines starting with #.
    Parse the date using datetime.
    Example expenses_string:
        2023-01-02 -34.01 USD
        2023-01-03 2.59 DKK
        2023-01-03 -2.72 EUR
    """
    expenses = []

    for line in expenses_string.splitlines():
        if line.startswith("#"):
            continue
        date, value, currency = line.split (" ")
        expenses.append((datetime.datetime.strptime (date, "%Y-%m-%d"),
                        float (value),
```

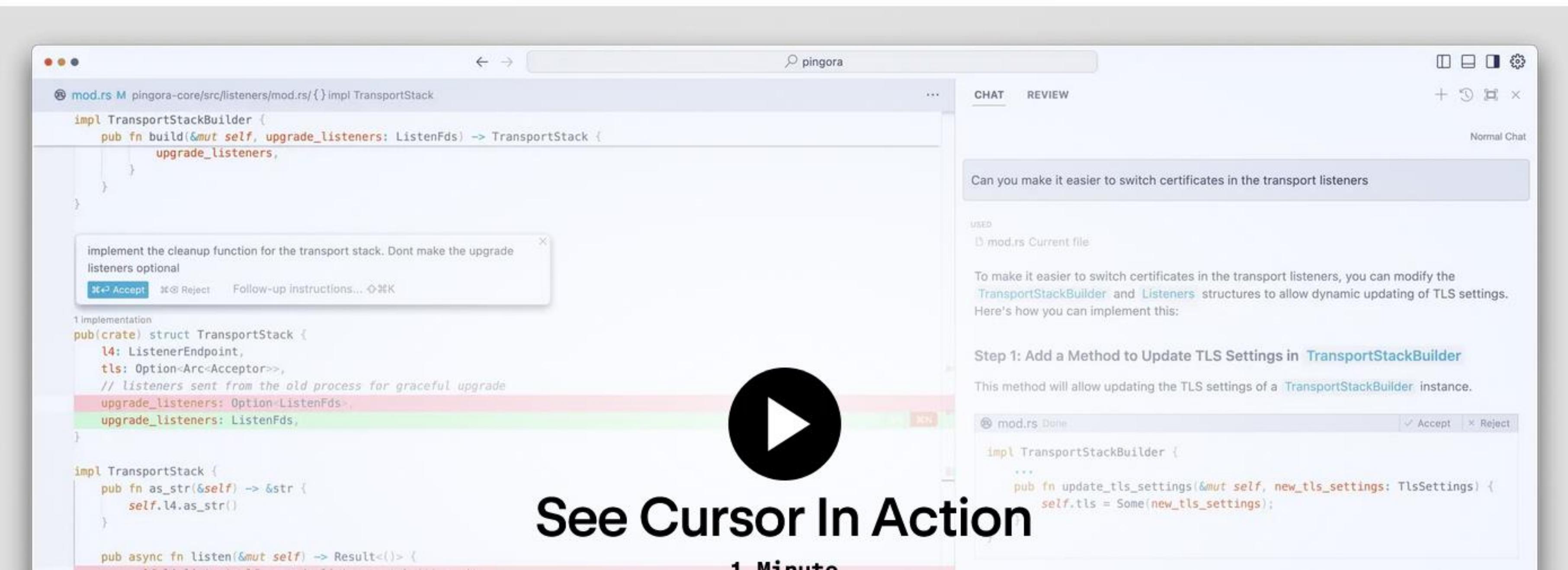
CURSOR

The AI Code Editor

Built to make you extraordinarily productive,
Cursor is the best way to code with AI.

 Download for Free

 Watch Demo
1 Minute



HINWEISE ZUR NUTZUNG INTEGRIERTER KI-UNTERSTÜTZUNG

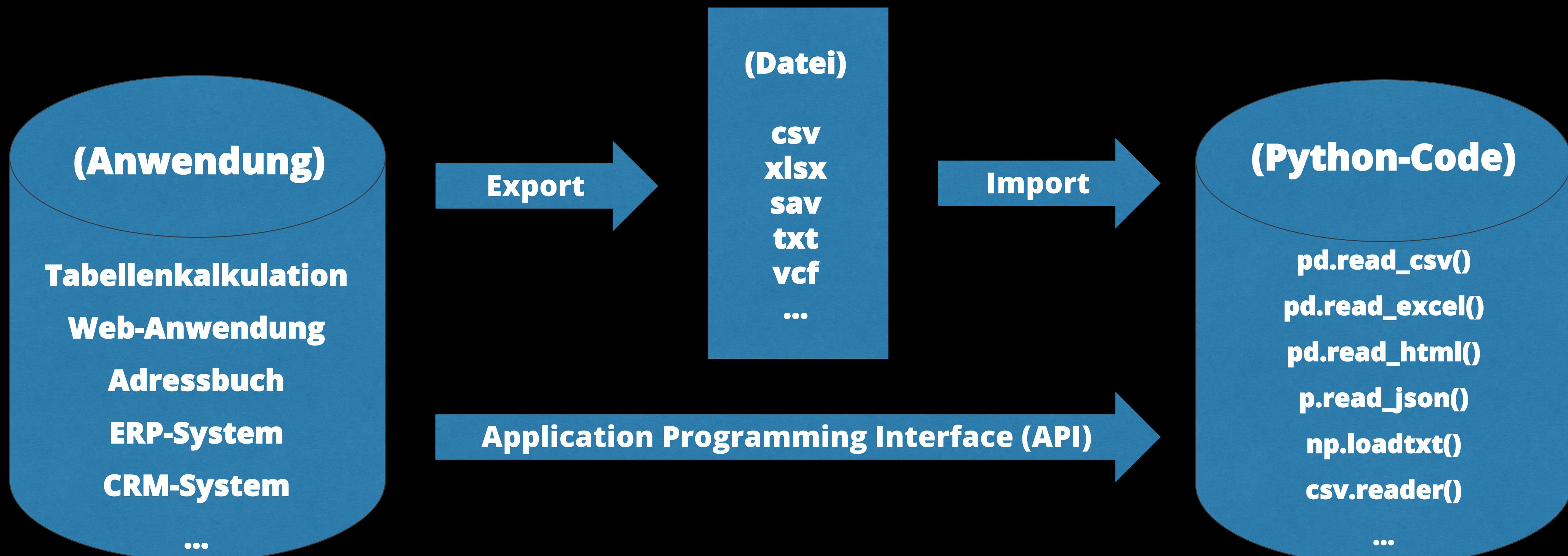
- **Schrittweises Vorgehen**
Für eine größere Aufgabe nutze den Chat, um diese zunächst in Teilschritte zu zerlegen und klar definierte Teilschritte zu implementieren und zu testen.
- **Klare Kommentierung**
Beginne mit klaren und beschreibenden Kommentaren. Die AI-Completion reagiert gut auf Kommentare, die detailliert beschreiben, was du machen möchtest.
- **Starte ggf. mit dem Erstellen eines Tests**
Das Schreiben von Tests kann helfen, den richtigen Implementierungscode zu erstellen, da er versucht, Code zu generieren, der die Tests besteht.
- **Vergib aussagekräftige Namen**
Gib Funktionen und Variablen aussagekräftige Namen. Diese werden genutzt, um den Kontext zu verstehen und bessere Vorschläge zu machen.
- **Weniger Informationen helfen manchmal mehr**
Nutze z.B., @codebase eher im Ausnahmefall.

TIPPS ZUR SUCHE MIT GOOGLE

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

EINLESEN VON DATEN

IMPORT VON DATEN



ENCODING

Methode zur Darstellung von Texten/Zeichen in Computersystemen: Zuordnung von Bitfolgen zu Zeichen

Relevant bei:

- **Text-/CSV-Dateien (.txt, .csv)**
- **XML-Dateien**
- **HTML-Dateien**
- **JSON-Dateien (wenn als Text gespeichert)**

Häufige Encoding-Formate:

- **UTF-8 (Unicode)** - Standard für Webanwendungen, unterstützt alle Sprachen
- **ASCII** - Basis-Zeichensatz (128 Zeichen, nur englische Zeichen)
- **Latin-1 (ISO-8859-1)** - Erweiterter ASCII für westeuropäische Sprachen
- **Windows-1252** - Microsoft-Version von Latin-1

Typische Probleme:

- **Umlaute (ä,ö,ü) und Sonderzeichen (€,§,°)**
- **"Mojibake"** - falsch dargestellte Zeichen (z.B. "Ã¼" statt "ü")
- **Unterschiedliche Standards in verschiedenen Betriebssystemen**

Best Practices:

- **UTF-8 als Standard verwenden**
- **Encoding beim Datei-Import explizit angeben (encoding='utf-8')**
- **Encoding der Quelldatei vor dem Import prüfen (BOM (Byte Order Mark) beachten)**

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 CHATBOTS

- **Anweisung, die den vollständigen Dateinamen enthält sowie das Verzeichnis oder den Link, unter dem die Datei zu finden ist.**
- **Bei Textdateien (etwa 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

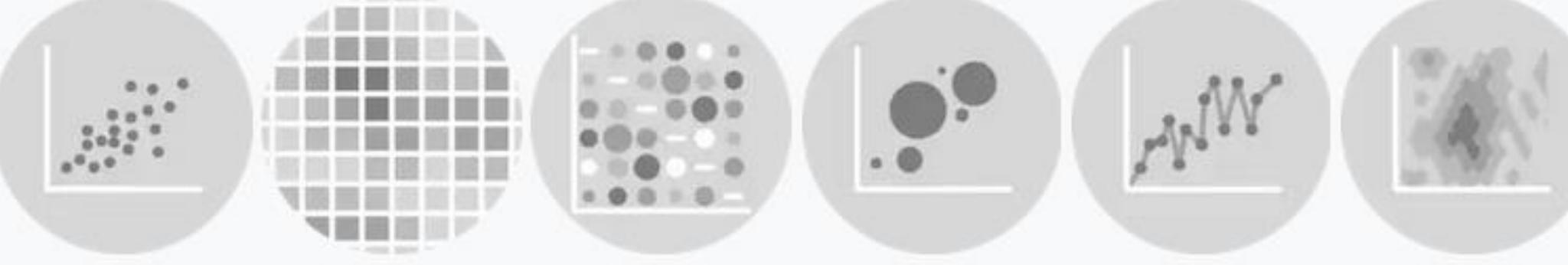
← Python Graph Gallery CHART TYPES TOOLS ▾ ALL BEST RELATED ▾ ABOUT [Subscribe](#)

Distribution



Violin Density Histogram Boxplot Ridgeline Beeswarm

Correlation



Scatterplot Heatmap Correlogram Bubble Connected Scatter 2D Density

Ranking



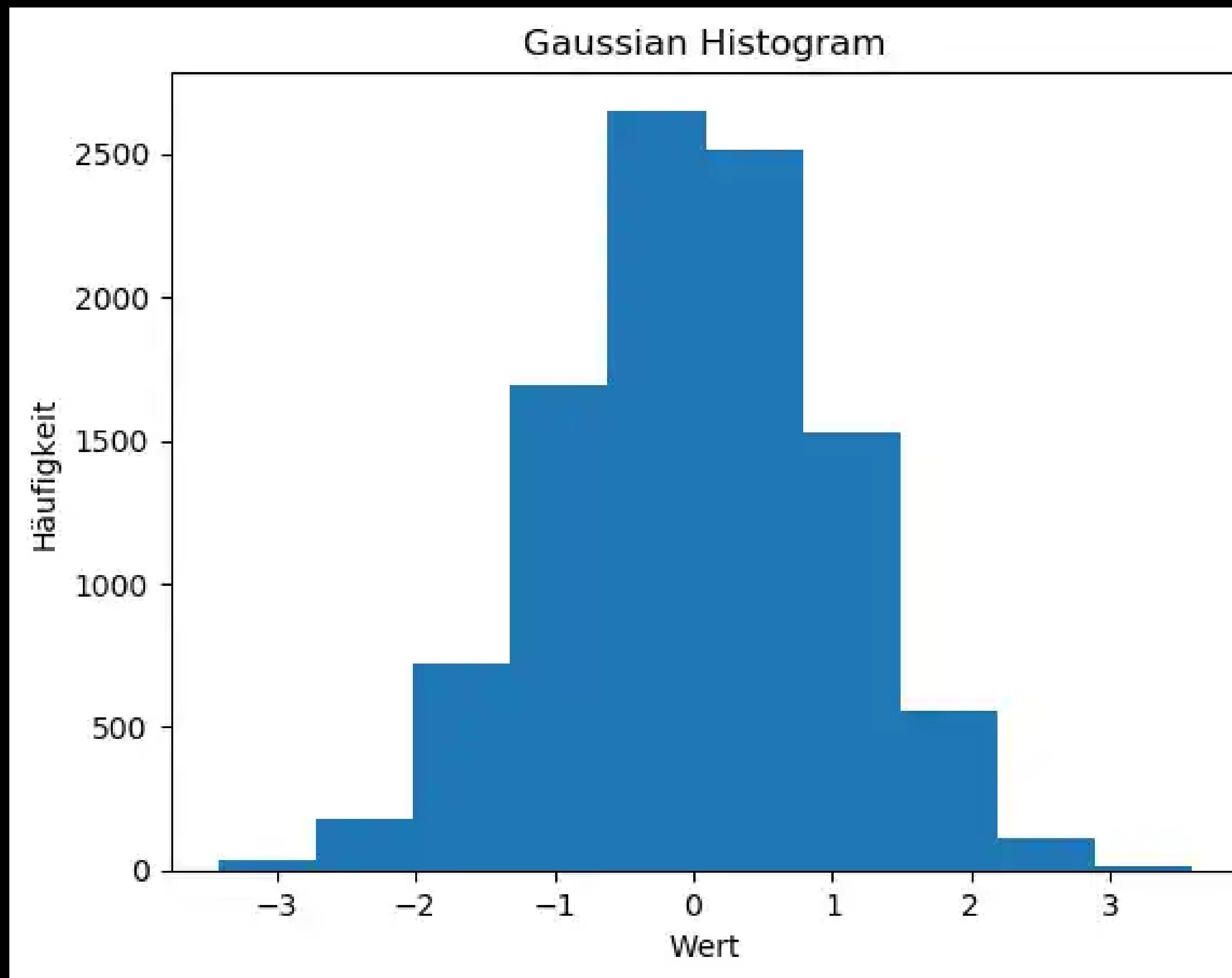
<https://python-graph-gallery.com/>

SKALENTYPEN

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

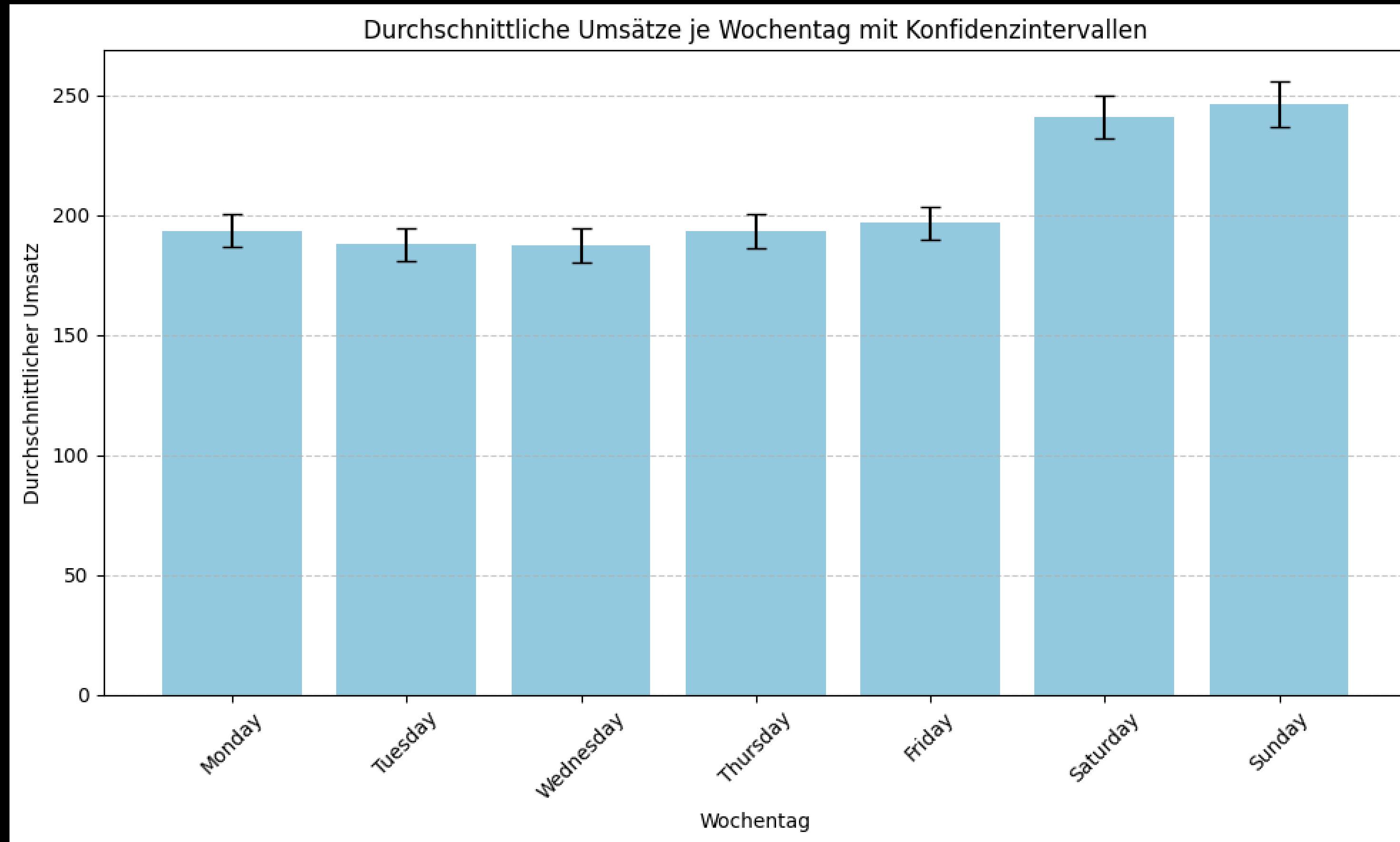
GÄNGIGE DIAGRAMMTYPEN

HISTOGRAMM



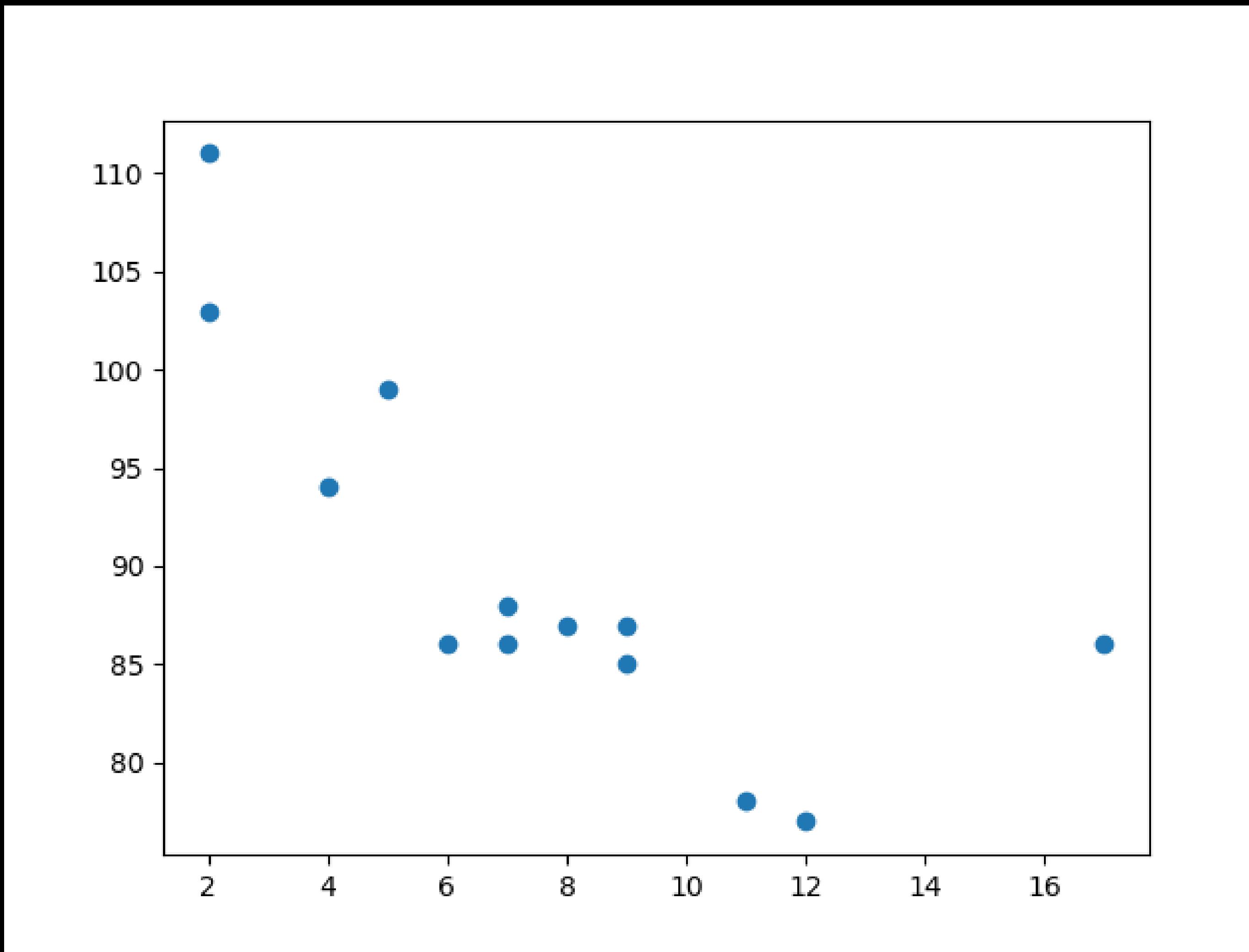
Darstellung der
Verteilung einer mind.
ordinalskalierten
Variable

BALKENDIAGRAMM



Darstellung aller Variabtentypen möglich; häufig zur Darstellung des Zusammenhangs mit einer mind. intervallskalierten Variable genutzt.

SCATTERPLOTT



**Darstellung der
Beziehung von zwei
mind. ordinalskalierten
Variablen;
aussagekräftiger für
intervallskalierte
Variablen**

PROJEKTDATENSATZ

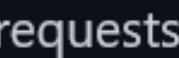
- **Umsatzdaten von verschiedenen Warengruppen einer Bäckereifiliale für den Zeitraum vom 01.07.2013 bis zum 30.07.2018**
- **Wetterdaten für den Zeitraum vom 01.07.2013 bis zum 30.07.2019**
- **Abrufbar unter:**
https://raw.githubusercontent.com/opencampus-sh/einfuehrung-in-data-science-und-ml/main/umsatzdaten_gekuerzt.csv

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

 + 

 [opencampus-sh/einfuehrung-in-data-science-und-ml](#) Edit Pins Watch 8 Fork 9 Star 12

Public

 Code  Issues  Pull requests  Actions  Projects  Wiki  Security  Insights  Settings

 main Go to file ...

 steffen74 Datensatz Sommersemester 2022 Latest commit fd081db 9 minutes ago 

 1 contributor

10910 lines (10910 sloc) | 319 KB Raw Blame   

 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)

 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 / wetter.csv](#) Go to file ...

 steffen74 Project Data Latest commit c61a127 on 20 Apr 2021 History

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

