

Blatt 8

Vincent Kümmerle und Elvis Gnaglo

9. Dezember 2025

1 Numpy-arrays

Gegeben sind die ersten 3 Zeilen des folgenden Code-Ausschnitts:

```
1 import numpy as np
2 np.random.seed(42)
3 data = np.random.rand(100, 2)

4
5 # Gibt die gesamte Anzahl aller Elemente aus
6 anzahl_elemente = data.size
7 print(f"1. Gesamtanzahl der Elemente: {anzahl_elemente}")

8
9 # Gibt das Element aus Zeile 0 und Spalte 1 aus
10 element_0_1 = data[0, 1]
11 print(f"2. Element (Zeile 0, Spalte 1): {element_0_1}")

12
13 # Gibt die Elemente aus der gesamten letzten Zeile aus
14 letzte_zeile = data[-1, :]
15 print(f"3. Die gesamte letzte Zeile: {letzte_zeile}")

16
17 # Gibt die Elemente aus der 10. Spalte aus. Fehler: es gibt nur 2
18 # Spalten.
19 # zehnte_spalte = data[:,9]
20 # print(f"4. Die Zehnte Spalte: {zehnte_spalte}")

21 # Definiert einen sub_array von Zeile 50 bis 59 und Spalte 0. Dabei
22 # ist zu beachten, dass der Start inklusiv ist, weswegen der Index
23 # 50 angegeben ist und das Ende exklusiv ist, weswegen der Index 60
24 # angegeben werden muss, damit der Wert 59 mitinbegriffen ist.
25 sub_array = data[50:60, 0]
26 print(f"5. Sub-Array (Zeile 50-59, Spalte 0):\n{sub_array}")

27
28 # Berechnet den Mittelwert der Zeilen 50 bis 99 in der Spalte 0.
29 mean_val = np.mean(data[50:100, 0])
```

```
27 print(f"6. Mittelwert (Zeile 50-99, Spalte 0): {mean_val}")
```

2 Plotten einer gedämpften harmonischen Schwingung mit matplotlib und argparse

2.1 Teil 1: Implementierung und Darstellung

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # Funktion zur Berechnung der gedämpften harmonischen Schwingung
5 def damped_oscillation(t, A, gamma, omega):
6     """
7         Berechnet die Auslenkung einer gedämpften harmonischen Schwingung
8         zur Zeit t.
9
10    x(t) = A * e^(-gamma * t) * cos(omega * t)
11    """
12
13 def main():
14     A = 1.0
15     gamma = 0.2
16     omega = 2 * np.pi
17     t = np.linspace(0, 10, 500)
18     x = damped_oscillation(t, A, gamma, omega)
19
20     plt.figure(figsize=(8,4))
21     plt.plot(t, x, label="x(t)")
22     plt.title("Gedämpfte harmonische Schwingung")
23     plt.xlabel("Zeit t [s]")
24     plt.ylabel("Auslenkung x(t)")
25     plt.legend()
26     plt.savefig("damped_oscillator.pdf")
27     plt.show()
28 if __name__ == "__main__":
29     main()
```

2.2 Teil 2: Erweiterung mit argparse (Kommandozeilen-Argumente)

3 Array-Slicing