

# Blatt 7

Vincent Kümmerle und Elvis Gnaglo

3. Dezember 2025

## 1 Zotero

Machinelles Lernen wird verwendet, um Muster in Datensammlungen herauszufinden. [1]

The screenshot shows a Zotero library entry for a scientific article. The title is "Quantum machine learning". The entry details are as follows:

**Infos**

Eintragsart: Zeitschriftenartikel

Titel: Quantum machine learning

Autor: Biamonte, Jacob

Autor: Wittek, Peter

Autor: Pancotti, Nicola

Autor: Rebentrost, Patrick

Autor: Wiebe, Nathan

Autor: Lloyd, Seth

Publikation: Nature

Band: 549

Ausgabe: 7671

Seiten: 195-202

Datum: 2017-09

Reihe

Titel der Reihe

Reihe Text

Zeitschriften-Abkürzung

Sprache: en

DOI: 10.1038/nature23474

ISSN: 1476-4687

Kurztitel

URL: <https://www.nature.com/articles/nature23474>

Heruntergeladen am: 2.12.2025, 10:26:08

Archiv

Standort im Archiv

Bibliothekskatalog: [www.nature.com](http://www.nature.com)

Signatur

Abbildung 1: Zeitschriftenartikel über Quantum Machine Learning in der Zotero-Bibliothek.

Das Buch “Experimentalphysik 1” von Wolfgang Demtröder erklärt die Grundlagen der Mechanik und Wärmelehre. [2]

The screenshot shows a library catalog entry for a book. At the top, the title "Experimentalphysik 1: Mechanik und Wärme" is displayed. Below it, under the heading "Info", are the following details:

- Citation key: demtroderExperimentalphysik1Mechanik2008
- Item Type: Book
- Title: Experimentalphysik 1: Mechanik und Wärme
- Author: Demtröder, Wolfgang
- Series: (partially visible)

Below the "Info" section, there is a list of metadata fields with their corresponding values:

- Series Number: (empty)
- Volume: (empty)
- # of Volumes: (empty)
- Edition: (empty)
- Place: (empty)
- Publisher: Springer
- Date: 2008
- # of Pages: (empty)
- Language: (empty)
- ISBN: (empty)
- Short Title: (empty)
- URL: (empty)
- Accessed: (empty)
- Archive: (empty)
- Loc. in Archive: (empty)
- Library Catalog: (empty)
- Call Number: (empty)
- Rights: (empty)
- Extra: (empty)

At the bottom of the entry, the dates are listed as:

- Date Added: 12/2/2025, 10:28:02 AM
- Modified: 12/2/2025, 10:31:36 AM

Abbildung 2: Buch über die Mechanik und Wärmelehre in der Physik.

## 2 List Comprehension

```
1 import random
2 import math
3 import matplotlib.pyplot as plt
4
5 # Gegebene Funktion (angenommene Implementierung)
```

```

6 def generate_random_point_in_square(side_length):
7     """Generiert einen zufälligen Punkt (x, y) in einem Quadrat um
8     den Ursprung."""
9     # Bereich: -side_length/2 bis +side_length/2
10    half_side = side_length / 2
11    x = random.uniform(-half_side, half_side)
12    y = random.uniform(-half_side, half_side)
13    return (x, y)
14
15
16 # --- DEIN CODE HIER ---
17
18 def distance(p1, p2):
19     """Berechnet den Euklidischen Abstand zwischen zwei Punkten p1
20     und p2."""
21     return math.sqrt((p1[0] - p2[0])**2 + (p1[1] - p2[1])**2)
22
23
24 # Parameter
25 seitenlaenge = 2.0
26 kreis_radius = 1.0
27 kreis_mittelpunkt = (0, 0)
28anzahl_punkte = 1000
29
30# 1. Liste mit 1000 Punkten erzeugen (List Comprehension)
31points = [generate_random_point_in_square(seitenlaenge) for _ in
32          range(anzahl_punkte)]
33
34# 2. Punkte im Kreis filtern (List Comprehension)
35points_inside = [p for p in points if point_in_circle(p,
36            kreis_mittelpunkt, kreis_radius)]
37
38# 3. Verhältnis berechnen
39verhaeltnis = len(points_inside) / len(points)
40print(f"Anzahl Punkte gesamt: {len(points)}")
41print(f"Anzahl Punkte im Kreis: {len(points_inside)}")
42print(f"Verhältnis: {verhaeltnis}")
43
44# Begründung der Konvergenz:
45# Das Verhältnis konvergiert gegen Pi / 4.
46# Grund:
47# Fläche des Quadrats (A_q) = (2r)      = 4 r

```

```

46 # Fläche des Kreises (A_k) = pi * r
47 # Verhältnis A_k / A_q = (pi * r ) / (4 r ) = pi / 4
48 # Bei r=1 und Seitenlänge=2 ist das Verhältnis also pi/4 (~0.785).
49
50 # 4. Listen für Plot erstellen (List Comprehension)
51 # Alle Punkte (zur Darstellung des Quadrats)
52 x_all = [p[0] for p in points]
53 y_all = [p[1] for p in points]
54
55 # Nur Punkte im Kreis (zur farblichen Unterscheidung)
56 x_in = [p[0] for p in points_inside]
57 y_in = [p[1] for p in points_inside]
58
59 # Plotten
60 plt.figure(figsize=(6, 6))
61 plt.scatter(x_all, y_all, color='blue', s=5, label='Außerhalb')
62 plt.scatter(x_in, y_in, color='red', s=5, label='Innerhalb')
63 plt.title(f'Monte Carlo Simulation (Verhältnis = {verhaeltnis})')
64 plt.legend(loc='upper right')
65 plt.axis('equal') # Wichtig, damit der Kreis rund aussieht
66 plt.show()

```

### 3 Dateien lesen und Daten plotten

#### Literaturverzeichnis

- [1] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, S. Lloyd, *Nature* **2017**, 549, 195–202.
- [2] W. Demtröder, *Experimentalphysik 1: Mechanik Und Wärme*, Springer, **2008**.