

# Handbuch Data Science mit Python

Grundlegende Tools für die Arbeit mit Daten

# DAS INHALTS- VERZEICHNIS

» Hier geht's  
direkt  
zum Buch

---

# Inhalt

|                         |           |
|-------------------------|-----------|
| <b>Einleitung</b> ..... | <b>13</b> |
|-------------------------|-----------|

---

## Teil I: Mehr als normales Python: Jupyter

|  |           |
|--|-----------|
| <b>1 Der Einstieg in IPython und Jupyter</b> ..... | <b>21</b> |
| Die IPython-Shell starten .....                    | 21        |
| Das Jupyter Notebook starten .....                 | 22        |
| Hilfe und Dokumentation in IPython .....           | 22        |
| Tastaturkürzel in der IPython-Shell .....          | 27        |
| <b>2 Erweiterte interaktive Features</b> .....     | <b>31</b> |
| Magische Befehle in IPython .....                  | 31        |
| Verlauf der Ein- und Ausgabe .....                 | 33        |
| IPython und Shell-Befehle .....                    | 36        |
| <b>3 Debugging und Profiling</b> .....             | <b>41</b> |
| Fehler und Debugging .....                         | 41        |
| Profiling und Timing von Code .....                | 45        |
| Weitere IPython-Ressourcen .....                   | 50        |

---

## Teil II: Einführung in NumPy

|   |           |
|---|-----------|
| <b>4 Die Datentypen in Python</b> .....                     | <b>55</b> |
| Python-Integer sind mehr als nur ganzzahlige Werte .....    | 56        |
| Python-Listen sind mehr als nur einfache Listen .....       | 57        |
| Arrays feststehenden Typs in Python .....                   | 59        |
| Arrays anhand von Listen erzeugen .....                     | 59        |
| Neue Arrays erzeugen .....                                  | 60        |
| NumPys Standarddatentypen .....                             | 61        |
| <b>5 Grundlagen von NumPy-Arrays</b> .....                  | <b>63</b> |
| Attribute von NumPy-Arrays .....                            | 63        |
| Indizierung von Arrays: Zugriff auf einzelne Elemente ..... | 64        |
| Slicing: Teilmengen eines Arrays auswählen .....            | 65        |
| Arrays umformen .....                                       | 68        |
| Arrays verketteten und aufteilen .....                      | 69        |

|           |   |            |
|-----------|---|------------|
| <b>6</b>  | <b>Berechnungen mit NumPy-Arrays: universelle Funktionen</b> .....                  | <b>71</b>  |
|           | Langsame Schleifen .....  | 71         |
|           | Kurz vorgestellt: UFuncs .....  | 72         |
|           | NumPys UFuncs im Detail .....   | 73         |
|           | UFunc-Features für Fortgeschrittene .....   | 77         |
|           | UFuncs: mehr erfahren .....   | 79         |
| <b>7</b>  | <b>Aggregationen: Minimum, Maximum und alles dazwischen</b> .....                   | <b>81</b>  |
|           | Summieren der Werte eines Arrays .....  | 81         |
|           | Minimum und Maximum .....   | 82         |
|           | Beispiel: Durchschnittliche Größe der US-Präsidenten .....                          | 84         |
| <b>8</b>  | <b>Berechnungen mit Arrays: Broadcasting</b> .....                                  | <b>87</b>  |
|           | Kurz vorgestellt: Broadcasting .....  | 87         |
|           | Für das Broadcasting geltende Regeln .....  | 89         |
|           | Broadcasting in der Praxis .....  | 91         |
| <b>9</b>  | <b>Vergleiche, Maskierungen und boolesche Logik</b> .....                           | <b>95</b>  |
|           | Beispiel: Regentage zählen .....  | 95         |
|           | Vergleichsoperatoren als UFuncs .....   | 96         |
|           | Boolesche Arrays verwenden .....  | 98         |
|           | Boolesche Arrays als Maskierungen .....   | 100        |
|           | Verwendung der Schlüsselwörter »and« bzw. »or« und der<br>Operatoren & bzw.   ..... | 101        |
| <b>10</b> | <b>Fancy Indexing</b> .....   | <b>103</b> |
|           | Fancy Indexing im Detail .....  | 103        |
|           | Kombinierte Indizierung .....   | 105        |
|           | Beispiel: Auswahl zufälliger Punkte .....   | 105        |
|           | Werte per Fancy Indexing modifizieren .....   | 107        |
|           | Beispiel: Daten gruppieren .....  | 108        |
| <b>11</b> | <b>Arrays sortieren</b> .....   | <b>111</b> |
|           | Schnelle Sortierung in NumPy: np.sort und np.argsort .....                          | 112        |
|           | Nach Zeilen und Spalten sortieren .....   | 112        |
|           | Teilsortierungen: Partitionierung .....   | 113        |
|           | Beispiel: k nächste Nachbarn .....  | 113        |
| <b>12</b> | <b>Strukturierte Daten: NumPys strukturierte Arrays</b> .....                       | <b>117</b> |
|           | Strukturierte Arrays erzeugen .....   | 118        |
|           | Erweiterte zusammengesetzte Typen .....   | 119        |
|           | Record-Arrays: strukturierte Arrays mit Pfiff .....                                 | 120        |
|           | Weiter mit Pandas .....   | 120        |

---

## Teil III: Datenbearbeitung mit Pandas

|  |            |
|--|------------|
| <b>13 Kurz vorgestellt: Pandas-Objekte</b> .....             | <b>125</b> |
| Das Pandas-Series-Objekt .....                               | 125        |
| Das Pandas-DataFrame-Objekt .....                            | 128        |
| Das Pandas-Index-Objekt .....                                | 131        |
| <b>14 Daten indizieren und auswählen</b> .....               | <b>133</b> |
| Series-Daten auswählen .....                                 | 133        |
| DataFrame-Daten auswählen .....                              | 136        |
| <b>15 Mit Pandas-Daten arbeiten</b> .....                    | <b>141</b> |
| UFuncs: Indexerhaltung .....                                 | 141        |
| UFuncs: Indexanpassung .....                                 | 142        |
| UFuncs: Operationen mit DataFrame und Series .....           | 144        |
| <b>16 Handhabung fehlender Daten</b> .....                   | <b>147</b> |
| Kompromisse beim Umgang mit fehlenden Daten .....            | 147        |
| Fehlende Daten in Pandas .....                               | 148        |
| Pandas nullfähige Datentypen .....                           | 151        |
| Mit Nullwerten arbeiten .....                                | 152        |
| <b>17 Hierarchische Indizierung</b> .....                    | <b>157</b> |
| Mehrfach indizierte Series .....                             | 157        |
| Methoden zum Erzeugen eines MultiIndex .....                 | 161        |
| Indizierung und Slicing eines MultiIndex .....               | 163        |
| Multi-Indizes umordnen .....                                 | 166        |
| <b>18 Datenmengen kombinieren: concat und append</b> .....   | <b>171</b> |
| Verkettung von NumPy-Arrays .....                            | 172        |
| Einfache Verkettungen mit pd.concat .....                    | 172        |
| <b>19 Datenmengen kombinieren: merge und join</b> .....      | <b>177</b> |
| Relationale Algebra .....                                    | 177        |
| Join-Kategorien .....  | 178        |
| Angabe der zu verknüpfenden Spalten .....                    | 180        |
| Mengenarithmetik bei Joins .....                             | 183        |
| Konflikte bei Spaltennamen: das Schlüsselwort suffixes ..... | 184        |
| Beispiel: Daten von US-Bundesstaaten .....                   | 185        |
| <b>20 Aggregation und Gruppierung</b> .....                  | <b>191</b> |
| Planetendaten .....  | 191        |
| Einfache Aggregationen in Pandas .....                       | 192        |
| GroupBy: Aufteilen, Anwenden und Kombinieren .....           | 194        |

|           |   |            |
|-----------|---|------------|
| <b>21</b> | <b>Pivot-Tabellen</b> .....                                       | <b>203</b> |
|           | Gründe für Pivot-Tabellen .....                                   | 203        |
|           | Pivot-Tabellen von Hand erstellen .....                           | 204        |
|           | Die Syntax von Pivot-Tabellen .....                               | 204        |
|           | Beispiel: Geburtenraten .....                                     | 207        |
| <b>22</b> | <b>Vektorisierte String-Operationen</b> .....                     | <b>213</b> |
|           | Kurz vorgestellt: String-Operationen in Pandas .....              | 213        |
|           | Liste der Pandas-String-Methoden .....                            | 214        |
|           | Beispiel: Rezeptdatenbank .....                                   | 218        |
| <b>23</b> | <b>Zeitreihen verwenden</b> .....                                 | <b>223</b> |
|           | Kalenderdaten und Zeiten in Python .....                          | 223        |
|           | Zeitreihen in Pandas: Indizierung durch Zeitangaben .....         | 227        |
|           | Datenstrukturen für Zeitreihen in Pandas .....                    | 227        |
|           | Gleichförmige Sequenzen: pd.date_range .....                      | 228        |
|           | Häufigkeiten und Abstände .....                                   | 229        |
|           | Resampling, zeitliches Verschieben und geglättete Statistik ..... | 231        |
|           | Beispiel: Visualisierung von Fahrradzahlungen in Seattle .....    | 236        |
| <b>24</b> | <b>Leistungsstarkes Pandas: eval und query</b> .....              | <b>243</b> |
|           | Der Zweck von query und eval: zusammengesetzte Ausdrücke .....    | 243        |
|           | Effiziente Operationen mit pandas.eval .....                      | 244        |
|           | DataFrame.eval für spaltenweise Operationen .....                 | 246        |
|           | Die DataFrame.query-Methode .....                                 | 248        |
|           | Performance: Wann eval und query verwendet werden sollten .....   | 248        |
|           | Weitere Ressourcen .....  | 249        |

---

## Teil IV: Visualisierung mit Matplotlib

|           |   |            |
|-----------|---|------------|
| <b>25</b> | <b>Allgemeine Tipps zu Matplotlib</b> .....           | <b>253</b> |
|           | Matplotlib importieren .....                          | 253        |
|           | Stil einstellen .....                                 | 253        |
|           | show oder kein show? – Anzeige von Diagrammen .....   | 253        |
| <b>26</b> | <b>Einfache Liniendiagramme</b> .....                 | <b>261</b> |
|           | Anpassen des Diagramms: Linienfarben und -stile ..... | 264        |
|           | Anpassen des Diagramms: Begrenzungen .....            | 266        |
|           | Diagramme beschriften .....                           | 268        |
|           | Stolpersteine in Matplotlib .....                     | 270        |

|  |            |
|--|------------|
| <b>27 Einfache Streudiagramme</b> .....  | <b>271</b> |
| Streudiagramme mit plt.plot erstellen .....                                      | 271        |
| Streudiagramme mit plt.scatter erstellen .....                                   | 273        |
| plot kontra scatter: eine Anmerkung zur Effizienz .....                          | 276        |
| Visualisierung von Messunsicherheiten .....                                      | 276        |
| <b>28 Dichtediagramme und Konturdiagramme</b> .....                              | <b>281</b> |
| Visualisierung einer dreidimensionalen Funktion .....                            | 281        |
| Histogramme, Binnings und Dichte .....   | 285        |
| Zweidimensionale Histogramme und Binnings .....                                  | 287        |
| <b>29 Anpassen der Legende</b> .....   | <b>291</b> |
| Legendenelemente festlegen .....   | 293        |
| Legenden mit Punktgrößen .....   | 295        |
| Mehrere Legenden .....   | 296        |
| <b>30 Anpassen von Farbskalen</b> .....  | <b>299</b> |
| Farbskala anpassen .....   | 300        |
| Beispiel: Handgeschriebene Ziffern .....   | 304        |
| <b>31 Untergeordnete Diagramme</b> .....   | <b>307</b> |
| plt.axes: untergeordnete Diagramme von Hand erstellen .....                      | 307        |
| plt.subplot: untergeordnete Diagramme in einem Raster anordnen .....             | 309        |
| plt.subplots: das gesamte Raster gleichzeitig ändern .....                       | 310        |
| plt.GridSpec: kompliziertere Anordnungen .....                                   | 312        |
| <b>32 Text und Beschriftungen</b> .....  | <b>315</b> |
| Beispiel: Auswirkungen von Feiertagen auf die Geburtenzahlen<br>in den USA ..... | 315        |
| Transformationen und Textposition .....  | 317        |
| Pfeile und Beschriftungen .....  | 319        |
| <b>33 Achsenmarkierungen anpassen</b> .....                                      | <b>323</b> |
| Vorrangige und nachrangige Achsenmarkierungen .....                              | 323        |
| Markierungen oder Beschriftungen verbergen .....                                 | 325        |
| Anzahl der Achsenmarkierungen verringern oder erhöhen .....                      | 326        |
| Formatierung der Achsenmarkierungen .....  | 328        |
| Zusammenfassung der Formatter- und Locator-Klassen .....                         | 330        |
| <b>34 Matplotlib anpassen: Konfigurationen und Stylesheets</b> .....             | <b>333</b> |
| Diagramme von Hand anpassen .....  | 333        |
| Voreinstellungen ändern: rcParams .....  | 335        |

|           |   |            |
|-----------|---|------------|
| <b>35</b> | <b>Dreidimensionale Diagramme in Matplotlib</b> ..... | <b>341</b> |
|           | Dreidimensionale Punkte und Linien .....              | 342        |
|           | Dreidimensionale Konturdiagramme .....                | 343        |
|           | Drahtgitter- und Oberflächendiagramme .....           | 344        |
|           | Triangulation von Oberflächen .....                   | 346        |
|           | Beispiel: Visualisierung eines Möbiusbands .....      | 348        |
| <b>36</b> | <b>Visualisierung mit Seaborn</b> .....               | <b>351</b> |
|           | Seaborn-Diagramme .....                               | 352        |
|           | Kategoriale Diagramme .....                           | 356        |
|           | Beispiel: Ergebnisse eines Marathonlaufs .....        | 359        |
|           | Weiterführende Ressourcen .....                       | 365        |
|           | Weitere Grafikbibliotheken für Python .....           | 366        |

---

## Teil V: Machine Learning

|           |  |            |
|-----------|--|------------|
| <b>37</b> | <b>Was ist Machine Learning?</b> .....                       | <b>369</b> |
|           | Kategorien des Machine Learning .....                        | 369        |
|           | Qualitative Beispiele für Machine-Learning-Anwendungen ..... | 370        |
|           | Zusammenfassung .....  | 379        |
| <b>38</b> | <b>Kurz vorgestellt: Scikit-Learn</b> .....                  | <b>381</b> |
|           | Datenrepräsentierung in Scikit-Learn .....                   | 381        |
|           | Die Estimator-API .....                                      | 384        |
|           | Anwendung: Handgeschriebene Ziffern untersuchen .....        | 392        |
|           | Zusammenfassung .....  | 397        |
| <b>39</b> | <b>Hyperparameter und Modellvalidierung</b> .....            | <b>399</b> |
|           | Überlegungen zum Thema Modellvalidierung .....               | 399        |
|           | Auswahl des besten Modells .....                             | 403        |
|           | Lernkurven .....   | 410        |
|           | Validierung in der Praxis: Rastersuche .....                 | 414        |
|           | Zusammenfassung .....  | 415        |
| <b>40</b> | <b>Feature Engineering</b> .....                             | <b>417</b> |
|           | Kategoriale Features .....                                   | 417        |
|           | Texte als Features .....                                     | 418        |
|           | Bilder als Features .....                                    | 420        |
|           | Abgeleitete Features .....                                   | 420        |
|           | Vervollständigung fehlender Daten .....                      | 422        |
|           | Feature-Pipelines .....                                      | 423        |

|   |            |
|---|------------|
| <b>41 Ausführlich: Naive Bayes-Klassifikation</b> .....                       | <b>425</b> |
| Bayes-Klassifikation .....  | 425        |
| Gaußsche naive Bayes-Klassifikation .....                                     | 426        |
| Multinomiale naive Bayes-Klassifikation .....                                 | 429        |
| Einsatzgebiete für naive Bayes-Klassifikation .....                           | 432        |
| <b>42 Ausführlich: Lineare Regression</b> .....                               | <b>435</b> |
| Einfache lineare Regression .....   | 435        |
| Regression der Basisfunktion .....  | 437        |
| Regularisierung .....   | 441        |
| Beispiel: Vorhersage des Fahrradverkehrs .....                                | 445        |
| <b>43 Ausführlich: Support Vector Machines</b> .....                          | <b>451</b> |
| Gründe für Support Vector Machines .....                                      | 451        |
| Support Vector Machines: Maximierung des Randbereichs .....                   | 453        |
| Beispiel: Gesichtserkennung .....   | 461        |
| Zusammenfassung .....   | 465        |
| <b>44 Ausführlich: Entscheidungsbäume und Random Forests</b> .....            | <b>467</b> |
| Gründe für Random Forests: Entscheidungsbäume .....                           | 467        |
| Estimator-Ensembles: Random Forests .....                                     | 471        |
| Random-Forest-Regression .....  | 473        |
| Beispiel: Random Forest zur Klassifikation handgeschriebener<br>Ziffern ..... | 475        |
| Zusammenfassung .....   | 477        |
| <b>45 Ausführlich: Hauptkomponentenanalyse</b> .....                          | <b>479</b> |
| Hauptkomponentenanalyse: ein Überblick .....                                  | 479        |
| Hauptkomponentenanalyse als Rauschfilter .....                                | 487        |
| Beispiel: Eigengesichter .....  | 489        |
| Zusammenfassung .....   | 492        |
| <b>46 Ausführlich: Manifold Learning</b> .....                                | <b>493</b> |
| Manifold Learning: »HELLO« .....  | 494        |
| Multidimensionale Skalierung .....  | 495        |
| Nichtlineare Mannigfaltigkeiten: lokal lineare Einbettung .....               | 500        |
| Überlegungen zum Thema Manifold-Methoden .....                                | 502        |
| Beispiel: Isomap und Gesichter .....  | 503        |
| Beispiel: Visualisierung der Strukturen in Zifferndaten .....                 | 507        |
| <b>47 Ausführlich: k-Means-Clustering</b> .....                               | <b>511</b> |
| Kurz vorgestellt: der k-Means-Algorithmus .....                               | 511        |
| Expectation-Maximization .....  | 513        |
| Beispiele .....   | 518        |



|  |            |
|--|------------|
| <b>48 Ausführlich: Gaußsche Mixture-Modelle</b> .....      | <b>525</b> |
| Gründe für GMM: Schwächen von k-Means .....                | 525        |
| EM-Verallgemeinerung: gaußsche Mixture-Modelle .....       | 528        |
| Wahl des Kovarianztyps .....                               | 532        |
| GMM als Dichteschätzung .....                              | 532        |
| Beispiel: GMM zum Erzeugen neuer Daten verwenden .....     | 536        |
| <b>49 Ausführlich: Kerndichteschätzung</b> .....           | <b>539</b> |
| Gründe für Kerndichteschätzung: Histogramme .....          | 539        |
| Kerndichteschätzung in der Praxis .....                    | 543        |
| Auswahl der Bandbreite durch Kreuzvalidierung .....        | 544        |
| Beispiel: Nicht ganz so naive Bayes-Klassifikation .....   | 545        |
| <b>50 Anwendung: Eine Gesichtserkennungspipeline</b> ..... | <b>551</b> |
| HOG-Features .....   | 552        |
| HOG in Aktion: eine einfache Gesichtserkennung .....       | 553        |
| Vorbehalte und Verbesserungen .....                        | 557        |
| Weitere Machine-Learning-Ressourcen .....                  | 559        |
| <b>Index</b> .....   | <b>561</b> |