

Inhaltsverzeichnis

	Einleitung	13
	Über den Autor	18
1	Mehr als normales Python: IPython	19
1.1	Shell oder Notebook?	19
1.1.1	Die IPython-Shell starten	20
1.1.2	Das Jupyter-Notebook starten	20
1.2	Hilfe und Dokumentation in IPython	21
1.2.1	Mit ? auf die Dokumentation zugreifen	22
1.2.2	Mit ?? auf den Quellcode zugreifen	23
1.2.3	Module mit der Tab-Vervollständigung erkunden	24
1.3	Tastaturkürzel in der IPython-Shell	26
1.3.1	Tastaturkürzel zum Navigieren	27
1.3.2	Tastaturkürzel bei der Texteingabe	27
1.3.3	Tastaturkürzel für den Befehlsverlauf	28
1.3.4	Sonstige Tastaturkürzel	29
1.4	Magische Befehle in IPython	29
1.4.1	Einfügen von Codeblöcken mit %paste und %cpaste	29
1.4.2	Externen Code ausführen mit %run	31
1.4.3	Messung der Ausführungszeit von Code mit %timeit	31
1.4.4	Hilfe für die magischen Funktionen anzeigen mit ?, %magic und %lsmagic	32
1.5	Verlauf der Ein- und Ausgabe	32
1.5.1	Die IPython-Objekte In und Out	33
1.5.2	Der Unterstrich als Abkürzung und vorhergehende Ausgaben	34
1.5.3	Ausgaben unterdrücken	34
1.5.4	Weitere ähnliche magische Befehle	35
1.6	IPython und Shell-Befehle	35
1.6.1	Kurz vorgestellt: die Shell	36
1.6.2	Shell-Befehle in IPython	37
1.6.3	Werte mit der Shell austauschen	37
1.7	Magische Befehle für die Shell	38
1.8	Fehler und Debugging	39
1.8.1	Exceptions handhaben: %xmode	39
1.8.2	Debugging: Wenn das Lesen von Tracebacks nicht ausreicht	41
1.9	Profiling und Timing von Code	44
1.9.1	Timing von Codeschnipseln: %timeit und %time	45
1.9.2	Profiling kompletter Skripte: %prun	46

1.9.3	Zeilenweises Profiling mit %lprun	47
1.9.4	Profiling des Speicherbedarfs: %memit und %mprun	48
1.10	Weitere IPython-Ressourcen	50
1.10.1	Quellen im Internet	50
1.10.2	Bücher	50
2	Einführung in NumPy	51
2.1	Die Datentypen in Python.	52
2.1.1	Python-Integers sind mehr als nur ganzzahlige Werte	53
2.1.2	Python-Listen sind mehr als nur einfache Listen.	54
2.1.3	Arrays feststehenden Typs in Python	56
2.1.4	Arrays anhand von Listen erzeugen	56
2.1.5	Neue Arrays erzeugen	57
2.1.6	NumPys Standarddatentypen	58
2.2	Grundlagen von NumPy-Arrays.	59
2.2.1	Attribute von NumPy-Arrays	60
2.2.2	Indizierung von Arrays: Zugriff auf einzelne Elemente	61
2.2.3	Slicing: Teilmengen eines Arrays auswählen.	62
2.2.4	Arrays umformen	65
2.2.5	Arrays verketten und aufteilen.	66
2.3	Berechnungen mit NumPy-Arrays: universelle Funktionen	68
2.3.1	Langsame Schleifen	68
2.3.2	Kurz vorgestellt: UFuncs	70
2.3.3	NumPys UFuncs im Detail	70
2.3.4	UFunc-Features für Fortgeschrittene	75
2.3.5	UFuncs: mehr erfahren	77
2.4	Aggregationen: Minimum, Maximum und alles dazwischen	77
2.4.1	Summieren der Werte eines Arrays	77
2.4.2	Minimum und Maximum	78
2.4.3	Beispiel: Durchschnittliche Größe der US-Präsidenten	80
2.5	Berechnungen mit Arrays: Broadcasting.	82
2.5.1	Kurz vorgestellt: Broadcasting	82
2.5.2	Für das Broadcasting geltende Regeln	84
2.5.3	Broadcasting in der Praxis	87
2.6	Vergleiche, Maskierungen und boolesche Logik	88
2.6.1	Beispiel: Regentage zählen.	89
2.6.2	Vergleichsoperatoren als UFuncs	90
2.6.3	Boolesche Arrays verwenden	91
2.6.4	Boolesche Arrays als Maskierungen	94
2.7	Fancy Indexing	97
2.7.1	Fancy Indexing im Detail.	97
2.7.2	Kombinierte Indizierung.	98
2.7.3	Beispiel: Auswahl zufälliger Punkte	99
2.7.4	Werte per Fancy Indexing modifizieren	101
2.7.5	Beispiel: Daten gruppieren	102

2.8	Arrays sortieren	104
2.8.1	Schnelle Sortierung in NumPy: np.sort und np.argsort	105
2.8.2	Teilsortierungen: Partitionierung	107
2.8.3	Beispiel: k nächste Nachbarn	107
2.9	Strukturierte Daten: NumPys strukturierte Arrays	112
2.9.1	Strukturierte Arrays erzeugen	113
2.9.2	Erweiterte zusammengesetzte Typen	114
2.9.3	Record-Arrays: strukturierte Arrays mit Pfiff	115
2.9.4	Weiter mit Pandas	115
3	Datenbearbeitung mit Pandas	117
3.1	Pandas installieren und verwenden	117
3.2	Kurz vorgestellt: Pandas-Objekte	118
3.2.1	Das Pandas-Series-Objekt	118
3.2.2	Das Pandas-DataFrame-Objekt	122
3.2.3	Das Pandas-Index-Objekt	126
3.3	Daten indizieren und auswählen	127
3.3.1	Series-Daten auswählen	127
3.3.2	DataFrame-Daten auswählen	131
3.4	Mit Pandas-Daten arbeiten	135
3.4.1	UFuncs: Indexerhaltung	136
3.4.2	UFuncs: Indexanpassung	137
3.4.3	UFuncs: Operationen mit DataFrame und Series	139
3.5	Handhabung fehlender Daten	140
3.5.1	Überlegungen zu fehlenden Daten	141
3.5.2	Fehlende Daten in Pandas	141
3.5.3	Mit null-Werten arbeiten	145
3.6	Hierarchische Indizierung	149
3.6.1	Mehrfach indizierte Series	149
3.6.2	Methoden zum Erzeugen eines MultiIndex	153
3.6.3	Indizierung und Slicing eines MultiIndex	156
3.6.4	Multi-Indizes umordnen	159
3.6.5	Datenaggregationen mit Multi-Indizes	162
3.7	Datenmengen kombinieren: concat und append	164
3.7.1	Verkettung von NumPy-Arrays	165
3.7.2	Einfache Verkettungen mit pd.concat	165
3.8	Datenmengen kombinieren: Merge und Join	169
3.8.1	Relationale Algebra	170
3.8.2	Join-Kategorien	170
3.8.3	Angabe der zu verknüpfenden Spalten	173
3.8.4	Mengenarithmetik bei Joins	176
3.8.5	Konflikte bei Spaltennamen: das Schlüsselwort suffixes	177
3.8.6	Beispiel: Daten von US-Bundesstaaten	178
3.9	Aggregation und Gruppierung	183
3.9.1	Planetendaten	183

3.9.2	Einfache Aggregationen in Pandas	184
3.9.3	GroupBy: Aufteilen, Anwenden und Kombinieren	186
3.10	Pivot-Tabellen	195
3.10.1	Gründe für Pivot-Tabellen	195
3.10.2	Pivot-Tabellen von Hand erstellen	196
3.10.3	Die Syntax von Pivot-Tabellen	197
3.10.4	Beispiel: Geburtenraten	199
3.11	Vektorisierte String-Operationen	204
3.11.1	Kurz vorgestellt: String-Operationen in Pandas	204
3.11.2	Liste der Pandas-Stringmethoden	206
3.11.3	Beispiel: Rezeptdatenbank	211
3.12	Zeitreihen verwenden	215
3.12.1	Kalenderdaten und Zeiten in Python	215
3.12.2	Zeitreihen in Pandas: Indizierung durch Zeitangaben	219
3.12.3	Datenstrukturen für Zeitreihen in Pandas	220
3.12.4	Häufigkeiten und Abstände	222
3.12.5	Resampling, zeitliches Verschieben und geglättete Statistik	224
3.12.6	Mehr erfahren	229
3.12.7	Beispiel: Visualisierung von Fahrradzählungen in Seattle	229
3.13	Leistungsstarkes Pandas: eval() und query()	236
3.13.1	Der Zweck von query() und eval(): zusammengesetzte Ausdrücke	236
3.13.2	Effiziente Operationen mit pandas.eval()	237
3.13.3	DataFrame.eval() für spaltenweise Operationen	239
3.13.4	Die DataFrame.query()-Methode	241
3.13.5	Performance: Verwendung von eval() und query()	242
3.14	Weitere Ressourcen	242
4	Visualisierung mit Matplotlib	245
4.1	Allgemeine Tipps zu Matplotlib	246
4.1.1	Matplotlib importieren	246
4.1.2	Stil einstellen	246
4.1.3	show() oder kein show()? – Anzeige von Diagrammen	246
4.1.4	Grafiken als Datei speichern	248
4.2	Zwei Seiten derselben Medaille	250
4.3	Einfache Liniendiagramme	251
4.3.1	Anpassen des Diagramms: Linienfarben und -stile	254
4.3.2	Anpassen des Diagramms: Begrenzungen	256
4.3.3	Diagramme beschriften	258
4.4	Einfache Streudiagramme	260
4.4.1	Streudiagramme mit plt.plot() erstellen	260
4.4.2	Streudiagramme mit plt.scatter() erstellen	263
4.4.3	plot kontra scatter: eine Anmerkung zur Effizienz	265
4.5	Visualisierung von Fehlern	265
4.5.1	Einfache Fehlerbalken	265
4.5.2	Stetige Fehler	267

4.6	Dichtediagramme und Konturdiagramme	268
4.6.1	Visualisierung einer dreidimensionalen Funktion.	268
4.7	Histogramme, Binnings und Dichte	272
4.7.1	Zweidimensionale Histogramme und Binnings	274
4.8	Anpassen der Legende.	277
4.8.1	Legendenelemente festlegen	279
4.8.2	Legenden mit Punktgrößen	280
4.8.3	Mehrere Legenden	282
4.9	Anpassen von Farbskalen	283
4.9.1	Farbskala anpassen	284
4.9.2	Beispiel: Handgeschriebene Ziffern	288
4.10	Untergeordnete Diagramme.	290
4.10.1	plt.axes: Untergeordnete Diagramme von Hand erstellen.	290
4.10.2	plt.subplot: Untergeordnete Diagramme in einem Raster anordnen	292
4.10.3	plt.subplots: Das gesamte Raster gleichzeitig ändern	293
4.10.4	plt.GridSpec: Kompliziertere Anordnungen	294
4.11	Text und Beschriftungen	296
4.11.1	Beispiel: Auswirkungen von Feiertagen auf die Geburtenzahlen in den USA	296
4.11.2	Transformationen und Textposition	299
4.11.3	Pfeile und Beschriftungen	300
4.12	Achsenmarkierungen anpassen	303
4.12.1	Vorrangige und nachrangige Achsenmarkierungen	304
4.12.2	Markierungen oder Beschriftungen verbergen.	305
4.12.3	Anzahl der Achsenmarkierungen verringern oder erhöhen	306
4.12.4	Formatierung der Achsenmarkierungen.	307
4.12.5	Zusammenfassung der Formatter- und Locator-Klassen.	310
4.13	Matplotlib anpassen: Konfigurationen und Stylesheets	311
4.13.1	Diagramme von Hand anpassen	311
4.13.2	Voreinstellungen ändern: rcParams	312
4.13.3	Stylesheets	314
4.14	Dreidimensionale Diagramme in Matplotlib.	318
4.14.1	Dreidimensionale Punkte und Linien	319
4.14.2	Dreidimensionale Konturdiagramme	320
4.14.3	Drahtgitter- und Oberflächendiagramme	322
4.14.4	Triangulation von Oberflächen	323
4.15	Basemap: geografische Daten verwenden	326
4.15.1	Kartenprojektionen	328
4.15.2	Zeichnen eines Kartenhintergrunds	332
4.15.3	Daten auf einer Karte anzeigen	334
4.15.4	Beispiel: Kalifornische Städte	335
4.15.5	Beispiel: Oberflächentemperaturen.	337
4.16	Visualisierung mit Seaborn.	339
4.16.1	Seaborn kontra Matplotlib	339
4.16.2	Seaborn-Diagramme	341

4.17	Weitere Ressourcen	357
4.17.1	Matplotlib	357
4.17.2	Weitere Grafikbibliotheken für Python	357
5	Machine Learning	359
5.1	Was ist Machine Learning?	360
5.1.1	Kategorien des Machine Learnings	360
5.1.2	Qualitative Beispiele für Machine-Learning-Anwendungen	361
5.1.3	Zusammenfassung	369
5.2	Kurz vorgestellt: Scikit-Learn	369
5.2.1	Datenrepräsentierung in Scikit-Learn	370
5.2.2	Scikit-Learns Schätzer-API	372
5.2.3	Anwendung: Handgeschriebene Ziffern untersuchen	380
5.2.4	Zusammenfassung	385
5.3	Hyperparameter und Modellvalidierung	385
5.3.1	Überlegungen zum Thema Modellvalidierung	385
5.3.2	Auswahl des besten Modells	389
5.3.3	Lernkurven	396
5.3.4	Validierung in der Praxis: Rastersuche	399
5.3.5	Zusammenfassung	401
5.4	Merkmalerstellung	401
5.4.1	Kategoriale Merkmale	402
5.4.2	Texte als Merkmale	403
5.4.3	Bilder als Merkmale	404
5.4.4	Abgeleitete Merkmale	405
5.4.5	Vervollständigung fehlender Daten	407
5.4.6	Pipelines mit Merkmalen	408
5.5	Ausführlich: Naive Bayes-Klassifikation	409
5.5.1	Bayes-Klassifikation	409
5.5.2	Gauß'sche naive Bayes-Klassifikation	410
5.5.3	Multinomiale naive Bayes-Klassifikation	413
5.5.4	Einsatzgebiete für naive Bayes-Klassifikation	416
5.6	Ausführlich: Lineare Regression	417
5.6.1	Einfache lineare Regression	417
5.6.2	Regression der Basisfunktion	419
5.6.3	Regularisierung	423
5.6.4	Beispiel: Vorhersage des Fahrradverkehrs	427
5.7	Ausführlich: Support Vector Machines	432
5.7.1	Gründe für Support Vector Machines	433
5.7.2	Support Vector Machines: Maximierung des Randbereichs	434
5.7.3	Beispiel: Gesichtserkennung	443
5.7.4	Zusammenfassung Support Vector Machines	447
5.8	Ausführlich: Entscheidungsbäume und Random Forests	448
5.8.1	Gründe für Random Forests	448
5.8.2	Schätzerensembles: Random Forests	454
5.8.3	Random-Forest-Regression	455

5.8.4	Beispiel: Random Forest zur Klassifikation handgeschriebener Ziffern	457
5.8.5	Zusammenfassung Random Forests	459
5.9	Ausführlich: Hauptkomponentenanalyse	460
5.9.1	Kurz vorgestellt: Hauptkomponentenanalyse	460
5.9.2	Hauptkomponentenanalyse als Rauschfilter	467
5.9.3	Beispiel: Eigengesichter	469
5.9.4	Zusammenfassung Hauptkomponentenanalyse	472
5.10	Ausführlich: Manifold Learning	473
5.10.1	Manifold Learning: »HELLO«	473
5.10.2	Multidimensionale Skalierung (MDS)	475
5.10.3	MDS als Manifold Learning	477
5.10.4	Nichtlineare Einbettungen: Wenn MDS nicht funktioniert	479
5.10.5	Nichtlineare Mannigfaltigkeiten: lokal lineare Einbettung	480
5.10.6	Überlegungen zum Thema Manifold-Methoden	482
5.10.7	Beispiel: Isomap und Gesichter	483
5.10.8	Beispiel: Visualisierung der Strukturen in Zifferndaten	487
5.11	Ausführlich: k-Means-Clustering	490
5.11.1	Kurz vorgestellt: der k-Means-Algorithmus	490
5.11.2	k-Means-Algorithmus: Expectation-Maximization	492
5.11.3	Beispiele	497
5.12	Ausführlich: Gauß'sche Mixture-Modelle	503
5.12.1	Gründe für GMM: Schwächen von k-Means	503
5.12.2	EM-Verallgemeinerung: Gauß'sche Mixture-Modelle	507
5.12.3	GMM als Dichteschätzung	511
5.12.4	Beispiel: GMM zum Erzeugen neuer Daten verwenden	515
5.13	Ausführlich: Kerndichteschätzung	518
5.13.1	Gründe für Kerndichteschätzung: Histogramme	518
5.13.2	Kerndichteschätzung in der Praxis	522
5.13.3	Beispiel: Kerndichteschätzung auf Kugeloberflächen	524
5.13.4	Beispiel: Nicht ganz so naive Bayes-Klassifikation	527
5.14	Anwendung: Eine Gesichtserkennungs-Pipeline	532
5.14.1	HOG-Merkmale	533
5.14.2	HOG in Aktion: eine einfache Gesichtserkennung	534
5.14.3	Vorbehalte und Verbesserungen	539
5.15	Weitere Machine-Learning-Ressourcen	541
5.15.1	Machine Learning in Python	541
5.15.2	Machine Learning im Allgemeinen	541
	Stichwortverzeichnis	543