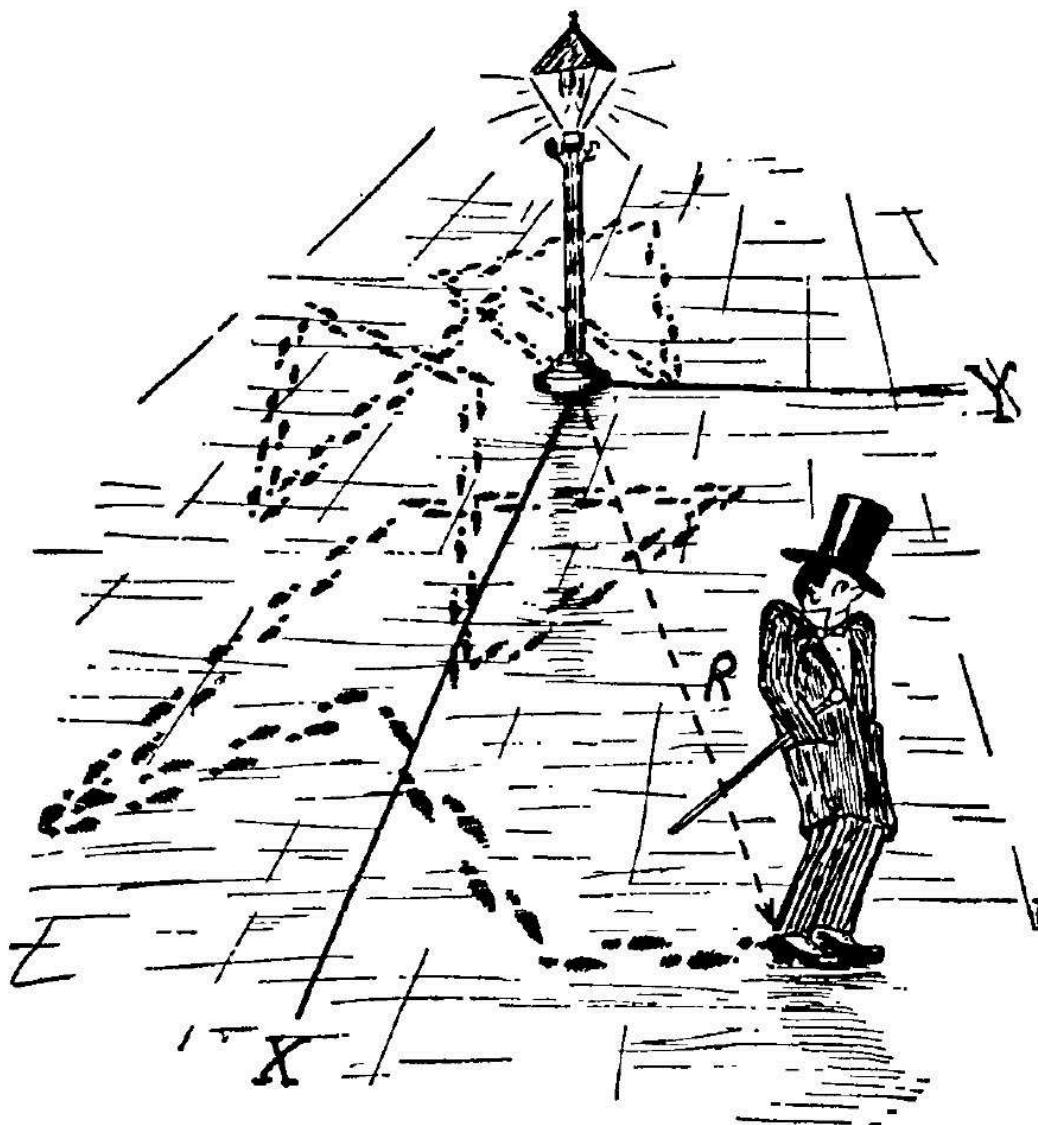


Mirko Birbaumer
Peter Büchel

Statistics for Data Science

Vorlesungsskript FS 19



Hochschule Luzern Informatik

Inhaltsverzeichnis

1. Einführung	1
1.1. Was ist Statistik?	1
1.2. Wozu kann ich Statistik brauchen?	6
2. Deskriptive Statistik	8
2.1. Deskriptive Statistik eindimensionaler Daten	8
2.1.1. Messungen der Schmelzwärme von Eis	8
2.1.2. Darstellung von Messwerten	10
2.1.3. Kennzahlen	11
2.1.4. Weitere Kennzahlen	16
2.1.5. Graphische Methoden	24
2.2. Deskriptive Statistik zweidimensionaler Daten	34
2.2.1. Graphische Darstellung: Streudiagramm	35
2.2.2. Einfache lineare Regression	37
2.2.3. Empirische Korrelation	47
2.3. Datenbereinigung	50
2.3.1. Entfernen von Datenpunkten	51
2.3.2. Data Imputation	56
3. Modelle für Messdaten	63
3.1. Stetige Zufallsvariablen und Wahrscheinlichkeitsverteilungen	63
3.1.1. Diskrete Wahrscheinlichkeitsverteilung	63
3.1.2. Stetige Verteilungen	66
3.1.3. Wahrscheinlichkeitsdichte	68
3.1.4. Kennzahlen von stetigen Verteilungen	75
3.2. Wichtige stetige Verteilungen	77
3.2.1. Uniforme Verteilung	77
3.2.2. Exponentialverteilung	81
3.2.3. Normalverteilung (Gauss-Verteilung)	85
3.3. Funktionen einer Zufallsvariable	91
3.3.1. Lineare Transformationen von Zufallsvariablen	92
3.3.2. Nichtlineare Transformationen von Zufallsvariablen	98
3.4. Funktionen von mehreren Zufallsvariablen	100
3.4.1. Unabhängigkeit und i.i.d. Annahme	101
3.4.2. Kennzahlen von S_n und \bar{X}_n	102
3.4.3. Verteilungen von S_n und \bar{X}_n	105

Inhaltsverzeichnis

3.4.4. Fehlerrechnung bei Messreihen	117
4. Statistik für Messdaten	123
4.1. Überprüfen der (Normal-) Verteilungsannahme	123
4.1.1. Q-Q-Plot	124
4.1.2. Normal-Plot	128
4.2. Parameterschätzung für stetige Wahrscheinlichkeitsverteilungen	131
4.2.1. Momentenmethode	131
4.2.2. Maximum-Likelihood-Methode	133
4.3. Statistische Tests bei normalverteilten Daten	142
4.3.1. Problemstellung	142
4.3.2. Hypothesentest	148
4.3.3. Der P -Wert	161
4.3.4. Der z -Test (σ_X bekannt)	164
4.3.5. Fehler 1. und 2. Art beim statistischen Test	169
4.3.6. Der t -Test (σ_X unbekannt)	172
4.4. Vertrauensintervall für μ	179
4.4.1. Bootstrapping	179
4.4.2. Vertrauensintervalle für Normalverteilungen	194
4.5. Statistische Tests bei nicht-normalverteilten Daten	204
4.5.1. Der Vorzeichentest	204
4.5.2. Der Wilcoxon-Test	210
4.6. Statistische Tests bei zwei Stichproben	213
4.6.1. Gepaarte Stichproben	213
4.6.2. Ungepaarte Stichproben	216
4.7. Statistische Signifikanz und fachliche Relevanz bei statistischen Tests .	219
5. Versuchsplanung	223
5.1. Einleitung	224
5.2. Grundelemente der Versuchsplanung	230
5.2.1. Grundprinzipien	235
5.3. Der Versuchsplan	236
5.4. Eine Checkliste zur Planung und Durchführung einer Studie	238
5.4.1. Problemstellung	239
5.4.2. Festlegen der Zielvariablen und Identifikation von erklärenden Variablen	239
5.4.3. Vorversuche	239
5.4.4. Planung	239
5.4.5. Stichprobenumfang	240
5.4.6. Daten	241
5.4.7. Daten-Bereinigung	241
5.4.8. Auswertung, Interpretation	241
5.4.9. Bericht	242
5.4.10. Beratung	242
5.4.11. Abschluss	243

Inhaltsverzeichnis

6. Varianz-Analyse	244
6.1. Vorbemerkungen	244
6.2. Mehrere Gruppen, einfache Varianzanalyse	245
6.2.1. Gruppenmittel-Modell	249
6.2.2. Anova-Test	257
6.3. Multiple Vergleiche, multiple Tests	265
6.3.1. Kontraste	266
6.3.2. Bonferroni-Regel	270
6.3.3. Bonferroni-Holm Regel	271
6.3.4. Paarweise Vergleiche	272
6.3.5. Konfirmatorische und explorative Analyse	274
6.4. Zweiweg-Varianzanalyse	278
6.4.1. Randomisiertes vollständiges Block-Design (RCBD)	278
6.4.2. Verbesserung der Genauigkeit mit RCBD im Vergleich zu CRD	291
6.4.3. Faktorielle Experimente mit zwei Faktoren	293
6.4.4. Vorgehensweise bei der Analyse von ANOVA-Tabellen	310
7. Einführung in die Zeitreihen	314
7.1. Einleitung	314
7.2. Beispiele	317
7.3. Zeitreihen mit pandas	320
7.4. Elementare Transformationen, Visualisierung und Zerlegung von Zeitreihen	327
7.4.1. Datentransformationen	328
7.4.2. Visualisierungen	333
7.4.3. Zerlegung von Zeitreihen	337
8. Mathematische Modelle für Zeitreihen	348
8.1. Vom Random Walk zum Thermischen Rauschen	348
8.1.1. Random Walk	348
8.2. Charakteristische Größen von stochastischen Signalen	355
8.3. Wahrscheinlichkeitsverteilungsfunktion von Stochastischen Prozessen	358
8.3.1. Ergodizität	362
8.4. Mathematische Konzepte für Zeitreihen	368
8.5. Serielle Korrelation	376
8.5.1. Mittelwertsfolge und Varianzfolge	376
8.5.2. Autokovarianz und Autokorrelation	378
8.6. Stationarität	396
8.6.1. Auf Stationarität testen	398
8.7. Brownsche Bewegung	403
A. R-Code	409
B. Aus der Normalverteilung hergeleitete Verteilungen	484
B.1. Dichtefunktion der Chi-Quadrat-Verteilung	484

Inhaltsverzeichnis

B.2. <i>t</i> -Verteilung	488
B.3. Chi-Quadrat Test	491
C. Ergänzungen und Herleitung von wichtigen Beziehungen in der Varianzanalyse	496
D. Signaltheorie und das Wiener-Khintchine Theorem	500
D.1. Herleitung der Lösung der Diffusionsgleichung aus Random Walk . . .	500
D.2. Korrelation von Signalen und Wiener-Khintchine-Theorem	502
D.3. Energie, Leistung, Korrelation	502
D.3.1. Ähnlichkeitsmaß für Signale	503
D.4. Korrelation und Faltung	505
D.4.1. Kreuzkorrelationstheorem	505
D.5. Wiener-Khintchine Theorem	507

Tabellenverzeichnis

2.1.	Messungen der latenten Schmelzwärme von Eis	8
2.2.	Weinkonsumation	35
2.3.	Buchpreis und Seitenzahl	38
2.4.	Größenvergleich von Vätern und Söhnen	44
2.5.	Verkehrstoten in aufeinanderfolgenden Jahren	45
4.1.	20 Messungen der Betondruckfestigkeit.	125
4.2.	Messungen der latenten Schmelzwärme von Eis	142
4.3.	Die wahre Verteilung F und die empirische Funktion F^* beim 8-seitigen Würfel.181	
4.4.	Messung der Schmelzwärme mit der Methode A.	205
4.5.	Vorzeichen der Schmelzwärme Messung in Bezug zum Median	205
4.6.	Daten und Ränge der Schmelzwärme-Messung	211
6.1.	Reissfestigkeit von Papier in Abhängigkeit der verwendeten Hartholzkonzentration	
6.2.	P -Werte für paarweise Vergleiche von je zwei Gruppen mit t -Test	246
6.3.	Varianzanalyse-Tabelle für eine einfache Varianzanalyse.	259
6.4.	Randomisiertes vollständiges Block-Design für das Röhrchen-Experiment, wobei die Zielw	
6.5.	Randomisiertes vollständiges Block-Design mit a Methoden und b Blöcken.281	
6.6.	Methoden- und Blockmittelwerte für den Datensatz Vaskuläre Röhrchen .285	
6.7.	Schätzungen von den Behandlungseffekten α_i	286
6.8.	Varianzanalyse-Tabelle für Zweiweg-Varianzanalyse ohne Wiederholungen (<i>Replikate</i>).288	
6.9.	Die Zielgrösse Haftungsfestigkeit des Datensatzes Grundierungsfarben ist aufgeführt,	
6.10.	Überlebenszeiten von Elritzen in Abhängigkeit der Cyanid-Konzentration bei zwei Wasser	
6.11.	Faktorielles Experiment mit zwei Faktoren mit jeweils zwei Stufen	297
6.12.	Faktorielles Experiment mit zwei Faktoren und Interaktion.	299
6.13.	Datenschema für faktorielles Experiment mit zwei Faktoren A (a Stufen) und B (b Stufen)	
6.14.	Varianzanalyse-Tabelle für die Zweiweg-Varianzanalyse mit Wechselwirkung.304	
7.1.	Passagierzahlen in Tausenden	314
B.1.	Number of decays within 7.5 seconds	494

Abbildungsverzeichnis

1.1. Konzept der Wahrscheinlichkeitsrechnung	4
2.1. Histogramm IQ-Test	25
2.2. Boxplot	29
2.3. Boxplot Schmelzwärme von Eis	31
2.4. Emp. kumulative Verteilungsfunktion von Schmelzwärme von Eis	32
2.5. Emp. kumulative Verteilungsfunktion von Schmelzwärme von Eis	34
2.6. Streudiagramm Mortalität und Weinkonsum	36
2.7. Streudiagramm Seitenzahl - Buchpreis	39
2.8. Gerade durch Streudiagramm	40
2.9. Residuen	40
2.10. Streudiagramm Körpergrößen Väter-Söhne	45
2.11. Verkehrstote	46
2.12. Regressionsgerade Weinkonsum-Sterblichkeit	47
2.13. 21 verschiedene Datensätze	50
3.1. Beispiel für eine kumulative Verteilungsfunktion	71
3.2. Dichte und Wahrscheinlichkeit einer Zufallsvariablen	74
3.3. Illustration Verteilungsfunktion	74
3.4. Dichtefunktion mit einer „unregelmässigen“ Form	75
3.5. Illustration des Quantils	76
3.6. Dichte und Verteilungsfunktion der uniformen Verteilung	78
3.7. Dichtefunktion von Uniform([1, 10]) ausgewertet an der Stelle $x = 5$	79
3.8. Darstellung der Wahrscheinlichkeit $P(1 \leq X \leq 5)$ als Fläche	79
3.9. Darstellung der Wahrscheinlichkeit $P(1.2 \leq X \leq 4.8)$ als Fläche	80
3.10. Dichte und Verteilungsfunktion der Exponentialverteilung	82
3.11. Wahrscheinlichkeit $P(0 \leq X \leq 4)$ als Fläche dargestellt	84
3.12. Dichte und Verteilungsfunktion der Normalverteilung	86
3.13. Wahrscheinlichkeit $P(X > 130)$	87
3.14. Quartile für 95 % der Fläche um 100	88
3.15. Wahrscheinlichkeit für IQ zwischen $\mu \pm \sigma$	89
3.16. Dichte der Normalverteilung	90
3.17. Histogramm mit 10 Ziehungen	107
3.18. 4 Histogramme mit 10 Ziehungen	108
3.19. Histogramm vom Durchschnitt von zwei Versuchen mit 10 Ziehungen	109
3.20. Histogramm vom Durchschnitt von drei Versuchen mit 10 Ziehungen	110

Abbildungsverzeichnis

3.21. Histogramm vom Durchschnitt von drei Versuchen mit 10 Ziehungen	111
3.22. Histogramme vom Mittelwert von drei Versuchen von jeweils 10 Ziehungen	112
3.23. 4 Histogramme vom Durchschnitt von 1000 Ziehungen	113
3.24. [4 Histogramme vom Durchschnitt von 1000 Ziehungen mit Dichtekurven	114
3.25. 100 Messungen der Fallzeit eines Körpers.	120
4.1. QQ-Plot vom Datensatz der Betondruckfestigkeit	128
4.2. Normal-Plot von $\mathcal{N}(0, 1)$ und Cauchy-Verteilung	129
4.3. Normal-Plots	130
4.4. Schlecht passende Parameterwerte der Verteilung mit kleinen Wahrscheinlichkeiten für die beobachteten Daten	131
4.5. Gut passender Parameterwert der Verteilung mit grossen Wahrscheinlichkeiten für die beobachteten Daten	132
4.6. Mittelwert, der sehr weit vom erwarteten Wert von $\mu = 80$ entfernt ist	146
4.7. Mittelwert, der weit vom erwarteten Wert von $\mu = 80$ entfernt ist	147
4.8. Verwerfungsbereich 1	151
4.9. Verwerfungsbereich 2	153
4.10. Verwerfungsbereich Körpergrösse	157
4.11. Verwerfungsbereich für das Beispiel, bei welchem die erwartete Körpergrösse von Frauen	161
4.12. P-Werte in Abhängigkeit von der Anzahl Messungen n in der Messreihe bei festem beobachteten Wert	163
4.13. P-Wert	163
4.14. Dichtefunktion des zweiseitigen Z-Tests zum Niveau α	166
4.15. Fehler 1. und 2. Art, Macht	172
4.16. Dichten der t -Verteilung	174
4.17. 100 Bootstraps-Vertrauensintervalle	194
4.18. Verwerfungsbereich bei Standardnormalverteilung	195
4.19. Verwerfungsbereich der Normalverteilung mit $\mu = 6$ und $\sigma = 2$	196
4.20. 100 Vertrauensintervalle der Normalverteilung mit $\mu = 6$ und $\sigma = 2$	198
4.21. Binomialverteilung für $n = 13$ und $\pi = 0.5$	206
4.22. Verschiedene Fälle (1 bis 5) von statistischer Signifikanz und fachlicher Relevanz. Die Verteilung ist in allen Fällen normalverteilt	207
6.1. Stripchart für den Beispieldatensatz Reissfestigkeit von Papier.	248
6.2. Boxplots für den Beispieldatensatz Reissfestigkeit von Papier	249
6.3. Stripchart für den Datensatz Meat in Abhängigkeit der vier Verpackungsmethoden.	255
6.4. Residuenplots für den Beispieldatensatz Meat in Abhängigkeit der vier Verpackungsmethoden	256
6.5. Schema eines ANOVA-Tests.	260
6.6. Simulierte Verteilung der Teststatistik zum Beispiel Reissfestigkeit unter der Nullhypothese	261
6.7. Schematische Darstellung (mit künstlichen Daten) für die Homogenitätsannahme: innerhalb der Gruppen ist die Varianz gleich	262
6.8. Beispieldatensatz : Vaskuläre Röhrchen . Der Prozentsatz von Flicks pro Produktionsmenge	263
6.9. Schematische Darstellung (mit künstlichen Daten) für die Homogenitäts-Annahme unter H ₀ : innerhalb der Gruppen ist die Varianz gleich	264
6.10. Graph der durchschnittlichen Haftungsfestigkeit für den Datensatz Grundierungsfarbe	265
6.11. "Überlebenszeiten von Elritzen in Abhängigkeit der Cyanid-Konzentration	296
6.12. Faktorielles Experiment mit zwei Faktoren und je zwei Stufen. Totaleffekt $e(B)$ ist gleich gleich groß	297
6.13. Faktorielles Experiment mit zwei Faktoren und Interaktion. Der Totaleffekt $e(B)$ hängt von den Faktoren ab	298
6.14. Haupteffekte und Zwei-Faktor-Wechselwirkungseffekte für ein Modell mit zwei Faktoren	299
6.15. Residuenplot für das Beispiel Elritzen, wobei die Residuen werden gegen die geschätzten Werte	300

Abbildungsverzeichnis

6.16. Residuenplot für das Beispiel Elritzen mit transformierter Zielgrösse $\tilde{y} = 1/y$.308	
6.17. Überlebenszeiten von Elritzen in Abhängigkeit der Cyanid-Konzentration mit transformie	
6.18. Residuen- und Normalplot für den Datensatz Grundierungsfarben . 310	
6.19. Interaktionsplot zwischen Faktor season und density für den Datensatz snails .312	
7.1. Plot der Passagierzahlen aus Tabelle 7.1	315
7.2. CO ₂ -Emissionen in der Schweiz.	318
7.3. Jährliche Temperaturenanomalien in Europa bezüglich des Durchschnittes zwischen 1910	
7.4. Städtische Luftqualitätsmessungen am 11. März 2004	320
7.5. Städtische Luftqualitätsmessungen am 11.-25. März 2004	321
7.6. Tagesabschlüsse des Tesla Aktienindex.	321
7.7. Log-Return des Tesla Aktienkurses	322
7.8. Vierteljährliche Bierproduktion in Australien	325
7.9. Bier- und Elektrizitätsverbrauch in Australien von 1959-1994	327
7.10. Box-Cox-Transformationen für verschiedene Werte für λ	329
7.11. Zeitverschiebung für $k = 4$ und $k = -5$	332
7.12. Temperatur über das gesamte Intervall	334
7.13. Temperatur über 20 Tage im März	335
7.14. Temperatur über 20 Tage im März gruppiert nach Stunden.	336
7.15. Lagged scatterplots für $k = 1$ (links) und $k = 10$ (rechts)	336
7.16. Schätzung des Trends	339
7.17. Der saisonale Effekt resultiert durch Subtraktion des Trends von der ursprünglichen Zeitreihe	
7.18. Gemittelte Saisonalität \hat{s}_i	341
7.19. Restterm (Residuen) \hat{r}	342
7.20. Zerlegung einer additiven Zeitreihe	343
7.21. Restterm für den Logarithmus der Daten	344
7.22. STL-Zerlegung für den (logarithmierten) AirPassenger -Datensatz. . 346	
8.1. Tagesabschlüsse des Tesla Aktienindex.	349
8.2. Weisses Rauschen.	350
8.3. Random Walks	350
8.4. Schritte auf einem eindimensionalen Gitter bei einem Random-Walk. . 351	
8.5. Thermische Bewegung von Elektronen, Rauschspannung	355
8.6. Zwei diskrete stochastische Signale $S_1(t)$ und $S_2(t)$	355
8.7. Relative Häufigkeiten von Signalwerten	356
8.8. Prozentualer Anteil von Signalwerten	357
8.9. Kontinuierlicher Zufallsprozess	358
8.10. Interpretation der Wahrscheinlichkeitsdichtefunktion.	360
8.11. Scharmittelwert	362
8.12. Gleich grosse Flächen	363
8.13. Zeitmittelwert	364
8.14. Maxwell-Boltzmann für das zweiatomige Stickstoffmolekül	367
8.15. Eine Zeitreihe als Beobachtung eines Random Walk.	370
8.16. Zeitreihe mit und ohne Drift	370

Abbildungsverzeichnis

8.17. Eine Realisierung eines Prozesses des weissen Rauschen.	372
8.18. Eine Realisierung eines moving average mit Fensterlänge 3.	373
8.19. Eine Realisierung eines autoregressiven Prozesses beruhend auf den zwei vorangehenden	374
8.20. Mittelwert über alle möglichen Zeitreihen zum Zeitpunkt t	378
8.21. Punkte, die fast auf einer Geraden liegen.	379
8.22. Punkte, die fast auf einer Geraden liegen.	380
8.23. Von den Koordinaten wurden die jeweiligen Mittelwerte subtrahiert. .	381
8.24. Punkte, die fast auf einer Geraden liegen.	382
8.25. Punkte, die fast auf einer Geraden liegen.	383
8.26. Messresultate der Wellenhöhe über eine Zeitspanne von 39.7 s.	387
8.27. Messresultate der Wellenhöhe über eine Zeitspanne von 6 s.	388
8.28. Paare von Wellenhöhen durch lag 1 (0.1 s) getrennt.	389
8.29. Streudiagramm von Wellenhöhen mit lag 2, 3, 5, 10.	389
8.30. Korrelogramm für die Wellenhöhen als Funktion vom lag k	390
8.31. Korrelogramm für den Moving Average Prozess	395
8.32. Korrelogramm für den Datensatz AirPassengers für die ersten 18 Monate.	400
8.33. Restterm der Zeitreihe AirPassengers , nach Entfernung von Trend und Saisonalität.	401
8.34. Korrelogramm der Zufallskomponente für AirPassengers	402
8.35. Brownsche Bewegung	404
8.36. Wahrscheinlichkeitsdichte und mittleres Verschiebungskquadrat	405
A.1. Box-Cox-transformations for different values of λ	474
A.2. Hourly air temperatur of 20 consecutive days in march 2014 in an Italian city	477
A.3. Grouped boxplot of air temperature data	478
A.4. Lagged scatterplot of the air temperature data	478
B.1. Chi-squared distribution	488
D.1. Leistungsdichtespektrum von weissem bandbegrenztem Rauschen. .	508
D.2. Rechtecksignal	509
D.3. Rücktransformation in den Zeitraum	511
D.4. Das T_0 -periodische Rechtecksignal	512
D.5. T_0 -periodische Fortsetzung	514

Kapitel 1.

Einführung

It is easy to lie with statistics.
It is hard to tell the truth
without statistics.

(Andrejs Dunkels)

1.1. Was ist Statistik?

Das Wort *Wahrscheinlichkeit* taucht in der Alltagssprache häufig auf. Hier einige Beispiele:

- Wir hören im Wetterbericht: „Die Wahrscheinlichkeit, dass es heute morgen regnet, liegt bei 60 Prozent“.
- Weiter hört man: „Die Wahrscheinlichkeit, dass ich hundert Jahre alt werde, ist klein“.
- In Basel möchte ein Seismologe bestimmen, wie gross die Wahrscheinlichkeit ist, dass Geothermie-Bohrungen ein Erdbeben von einer bestimmten Größenordnung auslösen.
- Ein Atomphysiker stellt sich andererseits die Frage: „Wie gross ist die Wahrscheinlichkeit, dass ein Geiger-Zähler in den nächsten 10 Sekunden 20 Zerfälle registriert?“.
- Ein Schweizer Politiker oder Nationalbanker interessiert sich momentan wohl für die Frage: „Wie gross ist die Wahrscheinlichkeit, dass der Wert vom Euro in diesem Jahr über 1.20 Franken liegt?“
- Oder: „Wie gross ist die Wahrscheinlichkeit, dass es einen Börsencrash gibt?“

Wahrscheinlichkeiten geben wir im Zusammenhang mit *Vermutungen* an. Warum stellen wir Vermutungen an? Wir stellen Vermutungen an, wenn wir eine Aussage oder Vorhersage machen möchten, aber dazu nur über *unvollständige Informationen* oder *unsichere Kenntnisse* verfügen oder weil wir eine *Entscheidung* fällen möchten:

- „Soll ich heute morgen einen Regenschirm mitnehmen?“
- „Soll ich mich bei einer Bank bewerben oder selbstversorgender Bio-Bauer werden?“

In den Naturwissenschaften möchten wir mit Hilfe unserer beschränkten oder unvollständigen Kenntnissen ein physikalisches System so allgemein wie möglich beschreiben. Die Beschreibung eines physikalischen Systems stellt aber letztlich nichts anderes als eine Vermutung (Modell) dar, denn wir können ein (realistisches) physikalisches System niemals bis ins letzte Detail beschreiben. Nun gibt es bessere und schlechtere Vermutungen, wie ein physikalisches System beschaffen ist. Die *Stochastik* hilft uns dabei, bessere Vermutungen anzustellen.

Beispiel 1.1.1

Betrachten wir das Beispiel einer Münze: Wir möchten vorhersagen, ob ein Münzwurf das Ergebnis „Kopf“ oder „Zahl“ ergibt. Wüssten wir die genaue Massenverteilung der Münze, die genaue Anfangsgeschwindigkeit und Anfangsposition der Münze und die Positionen und Geschwindigkeiten aller Luftmoleküle zu jedem Zeitpunkt während des Wurfs, könnten wir wohl mit Hilfe der Mechanik vorhersagen, ob der Münzwurf mit Kopf oder Zahl auf dem Boden landet.

Nun verfügen wir in der Praxis nie über *alle* diese Informationen. Aufgrund unserer Unkenntnis stellen wir die Vermutung an, dass die Massenverteilung der Münze dergestalt ist, dass wir diese als *fair* bezeichnen, d.h., die Anzahl Würfe mit „Kopf“ ist in *etwa gleich* der Anzahl Würfe mit „Zahl“, wenn wir die Münze sehr oft werfen. Je nach dem, wie stark sich die Anzahl Würfe mit „Kopf“ von der Anzahl Würfe mit „Zahl“ unterscheidet, können wir mit Hilfe der Stochastik aussagen, wie gut unsere Beobachtung mit der Vermutung zusammenpasst, dass die Münze fair ist und ob wir an unserer Vermutung (dass die Münze fair ist) festhalten sollten.

In der Statistik können wir zusätzlich auch Angaben darüber machen, wie plausibel ein Modell aufgrund von Beobachtungen ist (was auf den ersten Blick erstaunlich erscheint). Werfen wir eine Münze 10 000-mal und erhalten 7000-mal „Kopf“, können wir dann immer noch behaupten, dass die Münze fair ist? Theoretisch müssten wir 5000-mal „Kopf“ erhalten, aber es wäre ja möglich, dass zufälligerweise 7000-mal „Kopf“ geworfen wurde. Wann können wir den Zufall ausschliessen? Wenn wir 5050-mal „Kopf“ werfen, dann gehen wir vermutlich von einer fairen Münze aus. Aber bei 7000-mal „Kopf“? Wohl eher nicht. Aber wo liegt die Grenze? Bei welcher Anzahl „Kopf“ beginnen wir an der Fairness der Münze zu zweifeln? Solche Fragen können wir mit der Stochastik beantworten.

□

Beispiel 1.1.2

Auch in der kinetischen Gastheorie, wo wir es mit der Größenordnung von 10^{22} Gasmolekülen zu tun haben, können wir im besten Fall aussagen, wie wahrscheinlich es ist, dass ein Gasmolekül bei einer bestimmten Temperatur eine Geschwindigkeit in einem bestimmten Intervall hat. Denn es ist nicht realisierbar, jedem Molekül eine genaue Position und Geschwindigkeit zuzuordnen.

Dies hat nicht nur mit der Komplexität des Problems zu tun; wir wissen mittlerweile, dass die Quantenmechanik verbietet, die genaue Position und Geschwindigkeit eines Atoms gleichzeitig zu bestimmen (*Heisenbergsche Unschärferelation*).

Das Konzept von Wahrscheinlichkeiten ist essentiell, um das atomare Geschehen zu beschreiben.

□

Stochastik ist ein Teilgebiet der Mathematik und fasst als Oberbegriff die Gebiete Wahrscheinlichkeitsrechnung und Statistik zusammen.

In der *Wahrscheinlichkeitsrechnung* geht man von einem Modell aus (man beschreibt einen sogenannten datengenerierenden Prozess) und leitet daraus entsprechende Eigenschaften ab. Wie in Abbildung 1.1 dargestellt, kann man sich unter einem Modell symbolisch eine Urne vorstellen, aus der man Kugeln (Daten) zieht.

Um den Zusammenhang zwischen Wahrscheinlichkeit und Statistik besser zu verstehen, machen wir zwei Beispiele, ein theoretisches und ein praktisches.

Beispiel 1.1.3

Zum Beispiel können wir uns die Frage stellen: „Wie gross ist die Wahrscheinlichkeit, eine rote Kugel zu ziehen?“ Diese Frage können wir beantworten, wenn wir wissen, wie viele rote und blaue Kugeln in der Urne sind. Hat es drei rote und fünf blaue Kugeln in der Urne, so beträgt die Wahrscheinlichkeit, zufällig eine rote Kugel zu ziehen, $\frac{3}{8}$. Auf diese Situation bezieht sich Abbildung 1.1 links. Das *Modell* lautet in diesem Fall:

- Es hat drei rote und fünf blaue Kugeln in der Urne.
- Jede Kugel wird mit der gleichen Wahrscheinlichkeit gezogen.

Wir können dieses Modell nun überprüfen, indem wir viele Male eine Kugel ziehen und wieder zurücklegen. Wird die rote Kugel mit einer Wahrscheinlichkeit von etwa $\frac{3}{8}$ gezogen, so wird am Modell nicht gezwifelt.

Falls sich aber die Wahrscheinlichkeit zu etwa $\frac{7}{8}$ ergibt, so wird wohl etwas am Modell nicht stimmen. Beispielsweise könnten aus irgendeinem Grund die blauen Kugeln immer an den Boden der Urne rutschen (Abbildung 1.1 links), und somit

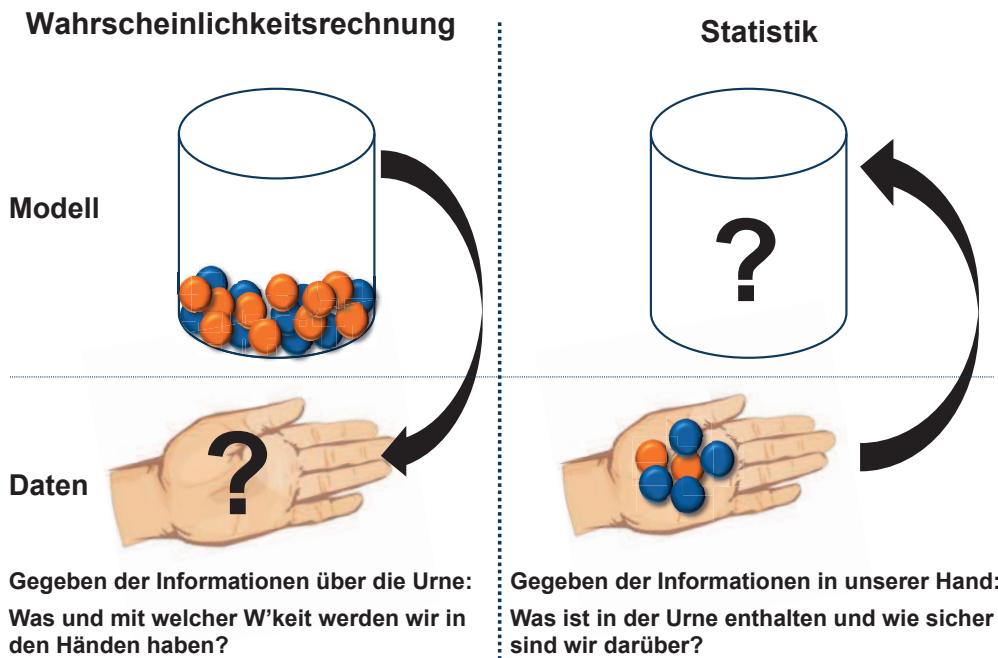


Abbildung 1.1.: Darstellung der Konzepte der Wahrscheinlichkeitsrechnung und der Statistik.
Das Modell wird hier durch eine Urne symbolisiert.

werden die roten Kugeln mit einer grösseren Wahrscheinlichkeit gezogen, obwohl es weniger rote Kugeln hat.

Ist in Abbildung 1.1 rechts die Anzahl der blauen und roten Kugeln in der Urne unbekannt, so können wir zum Beispiel 100-mal eine Kugel ziehen, die nach jeder Ziehung wieder zurückgelegt wird. Ziehen wir 40 rote Kugeln, so können wir *vermuten*, dass in der Urne 40 % der Kugeln rot sind. Diese Vermutung können wir überprüfen, indem wir weitere 100 Kugeln ziehen. Stimmt das Resultat mehr oder weniger mit dem ersten Versuch überein, so halten wir an unserer Vermutung fest. Ziehen wir aber 60 rote Kugeln, so müssen wir unsere Vermutung überprüfen.

Die Statistik hilft uns, quantitative Aussagen zu machen, wie gut eine Beobachtung mit einer solchen Vermutung (Modell) zusammenpasst.

□

In anwendungsorientierten Aufgaben und Fragestellungen entspricht der Ausgangspunkt praktisch immer der Abbildung 1.1 rechts. In der *Statistik* geht es darum, aus vorhandenen Daten Rückschlüsse auf das Modell zu machen, aufgrund dessen wir dann wieder Vorhersagen machen können. Wir denken also gerade „in die andere Richtung“.

Beispiel 1.1.4

Eine Gemeinde will für einen Bach einen Damm bauen, da dieser Bach oft über die Ufer tritt. Die Frage ist nun, wie hoch der Damm gebaut werden soll. Ein zu niedriger Damm ist zwar kostengünstig, dafür werden die Ausgaben für die Überschwemmungen sehr hoch. Auf der anderen Seite ist ein zu hoher Damm sehr teuer, auch wenn es keine Überschwemmungen mehr gibt. Wie können wir also vorgehen, um eine „gute“ Dammhöhe zu ermitteln? Eine „gute“ Dammhöhe zeichnet sich dadurch aus, dass der Damm genügend Sicherheit bietet, aber gleichzeitig auch noch finanziert wird.

Mit der Statistik (Abbildung 1.1 rechts) versuchen wir, ein Modell zu bilden. Wir analysieren Datenpunkte (z.B. jährliche Wasserstandsmessungen in den letzten 50 Jahren) und versuchen mit diesem beschränkten Wissen herauszufinden, was wohl ein gutes Modell für die „wahren“ jährlichen Höchststände des Wasserpegels ist.

Wie könnte nun ein solches Modell aussehen? Wir berechnen den Durchschnitt und die Standardabweichung der jährlichen Wasserhöchststände der letzten 50 Jahre. Unter der *Annahme*, dass sich die mittleren Höchststände und die Standardabweichung für die nächsten Jahre nicht ändern, *modellieren* wir die jährlichen Höchststände mit einer *Gauss-Verteilung*, wobei der Erwartungswert und die Standardabweichung der Gauss-Verteilung durch die Daten ermittelt werden (mehr zu diesen Begriffen später). Dieses Modell ist sehr einfach. Allerdings ist die Annahme, dass die durchschnittlichen Höchststände gleich bleiben, durch den Klimawandel infrage gestellt. Dies müssten wir in unserem Modell auch noch berücksichtigen. Ob eine Gauss-Verteilung wirklich passt, müsste ebenfalls überprüft werden.

Das Modell (Abbildung 1.1 links) sollte Vorhersagen über die Jahreshöchststände machen können. Diese Vorhersagen werden, wie oben erwähnt, in der Regel in Wahrscheinlichkeiten angegeben. Ein solches Wahrscheinlichkeitsmodell ist dann die Grundlage, aufgrund dessen wir die optimale Dammhöhe bestimmen werden. Die Unsicherheit in Bezug auf den jährlichen maximalen Wasserstand (z.B. in einer *kommenden* 30-Jahr Periode) quantifizieren wir nun mit diesem Wahrscheinlichkeitsmodell. Wir können zum Beispiel die Wahrscheinlichkeit berechnen, dass in einer 30-Jahr Periode der maximale Wasserstand gewisse Höhen überschreitet. Oder wir können die zu erwartenden Kosten in einer 30-Jahr Periode aufgrund von Überschwemmungen bei einer gegebenen Dammhöhe berechnen. Solche Quantifizierungen ermöglichen dann eine sinnvolle Kosten-Nutzen-Rechnung für den Bau eines Damms.

□

1.2. Wozu kann ich Statistik brauchen?

Die Statistik hat ihren Ursprung in der Mathematik, greift aber in viele Bereiche der modernen Wissenschaften über. Große Teile der Biologie, der Medizin, der Wirtschaft, der Ingenieurwissenschaften und der Umweltforschung wären heute ohne Statistik undenkbar.

Der Chefökonom von Google, Hal Varian, sagte vor einigen Jahren: Der sexieste Beruf des kommenden Jahrzehnts ist der des Statistikers. In der Vergangenheit bis heute stammen viele Anregungen und Problemstellungen für die Statistik aus den Bereichen Biologie und Pharmazie.

Beispiel 1.2.1

Eine häufige Frage bei Tier und Mensch lautet: wie bestätigt man die Wirksamkeit eines neuen Medikamentes? Dabei erhält eine zufällig ausgewählte Gruppe (Medikamentengruppe) von Patienten das neue Medikament in Form einer Tablette. Eine andere zufällig ausgewählte Gruppe von Patienten (Kontrollgruppe) erhält ein Placebo¹.

Die Medikamentengruppe hat gegenüber der Kontrollgruppe nach zwei Wochen eine deutliche Verbesserung der Symptome gezeigt. Das Medikament wirkt also. Oder vielleicht doch nicht? Kann es sein, dass das Medikament gar nicht wirkt und alle Personen der Medikamentengruppe unabhängig vom Medikament *zufällig* eine Verbesserung der Symptome hatte? Wie können wir „Zufall“ ausschließen? Mit Statistik lässt sich diese Frage beantworten.



Beispiel 1.2.2

In der Genetik stellt sich zum Beispiel folgendes Problem: Man beobachtet, dass in einer Gruppe von Krebspatienten gewisse Gene stärker aktiv sind als in einer Kontrollgruppe. Könnte es Zufall sein, dass alle Personen, bei denen diese Gene aktiver sind, zufällig in derselben Gruppe gelandet sind? Hat also das aktive Gen gar nichts mit der Krebserkrankung zu tun, obwohl es bei allen Krebspatienten vorkommt?

Wie können wir hier mathematisch Zufall ausschließen?



¹Tablette mit gleichem Aussehen und Geschmack wie das Medikament, aber ohne Wirkstoff.

Beispiel 1.2.3

Die Ingenieurswissenschaften liefern sehr interessante Aufgabenstellungen für die Statistik. Einige Beispiele, die vielen Leuten gar nicht bekannt sein dürften:

- Spracherkennungsprogramme (z.B. *Siri* von Apple)
- Programme, mit denen Roboter visuell ihre Umwelt erkennen können
- Programme für Analyse und Prognose von Geweben auf Krebs
- Schachprogramm *AlphaZero*

Alle diese Anwendungen funktionieren nur mit Statistik („Machine Learning“). Auch da geht es um den Umgang mit Unsicherheiten und Variabilität. Bei der Spracherkennung wird eine Silbe oder ein Wort von jeder Person leicht anders ausgesprochen. Ein gutes System muss trotz diesen Variationen ein Wort oder einen Text erkennen können.

Und ein Roboter wird kaum zweimal die genau gleiche Situation antreffen. Trotzdem muss er entscheiden können, ob eine angetroffene Situation einer gespeicherten Standardsituation *ähnlich* ist. Gewebe, die von einer bestimmten Krebsart befallen sind, sind *nie* identisch, sondern sind sich bloss *ähnlich*. Ähnliche Überlegungen gelten für das Schachspiel.

□

Diese Vorlesung soll Ihnen helfen, ein Fundament zu legen und die Grundbegriffe in der Statistik verstehen und anwenden zu können.

Kapitel 2.

Deskriptive Statistik

2.1. Deskriptive Statistik eindimensionaler Daten

Die deskriptive Statistik befasst sich mit der Darstellung von Datensätzen. Dabei werden diese Daten durch gewisse Zahlen charakterisiert - zum Beispiel den Mittelwert und graphisch dargestellt - zum Beispiel mit Hilfe eines Histogramms. Wir befassen uns zunächst mit *eindimensionalen* Daten, bei welchen *eine* Messgröße an einem Untersuchungsobjekt ermittelt wurde. Anhand des folgenden Beispiels werden wir einige wichtige Begriffe und Vorgehensweisen genauer kennenlernen.

2.1.1. Messungen der Schmelzwärme von Eis

Als Einführungsbeispiel betrachten wir zwei *Datensätze*, bei welchen zwei Methoden zur Bestimmung der latenten Schmelzwärme von Eis verglichen werden.

Beispiel 2.1.1

Wiederholte Messungen der freigesetzten Wärme beim Übergang von Eis bei $-0.7\text{ }^{\circ}\text{C}$ zu Wasser bei $0\text{ }^{\circ}\text{C}$ ergaben die Werte (in cal/g), die in Tabelle 2.1 aufgeführt sind.

Obwohl die Messungen mit der grösstmöglichen Sorgfalt durchgeführt und alle (kontrollierbaren) Störeinflüsse ausgeschaltet wurden, variieren die Messwerte innerhalb beider Methoden. Es stellen sich hier nun die folgenden Fragen:

Methode A	79.98	80.04	80.02	80.04	80.03	80.03	80.04	79.97	80.05
Methode A	80.03	80.02	80.00	80.02					
Methode B	80.02	79.94	79.98	79.97	79.97	80.03	79.95	79.97	

Table 2.1.: Messungen zur Bestimmung der latenten Schmelzwärme von Eis anhand von zwei Methoden.

Kapitel 2. Deskriptive Statistik

- Gibt es einen Unterschied zwischen der Methode A und der Methode B?
- Falls es einen Unterschied zwischen der Methode A und der Methode B gibt, wie können wir diesen Unterschied nachweisen?

Es fällt auf, dass bei beiden Methoden die Messwerte um den Wert 80 streuen. Bei Methode A liegen aber nur 2 Werte von 13 *unter* 80, während bei der Methode B nur 2 von 8 Werten *über* 80 liegen. Die Werte der Methode A sind also eher grösser als die der Methode B. Was heisst hier aber „eher“? Es ist also von Interesse, die Messreihen irgendwie so zusammenzufassen, dass wir die beiden Methoden miteinander vergleichen können.

□

Die *deskriptive Statistik* beschäftigt sich damit, auf welche Weisen (quantitative) Daten organisiert und zusammengefasst werden können. Dies hat zum Ziel, dass die Interpretation und darauffolgende statistische Analyse dieser Daten vereinfacht werden. Wir machen dies mit Hilfe von

- Zusammenfassungen von Daten, die die wichtigen Merkmale der Daten hervorheben sollen, wie eben zum Beispiel die mittlere Lage der Messwerte und die Streuung dieser Messwerte um die mittlere Lage.
- graphischen Darstellungen

Diese sogenannten *Kennzahlen* sollen die Daten numerisch zusammenfassen und grob charakterisieren.

Bei statistischen Analysen, wie wir sie im Laufe der Vorlesung kennenlernen werden, ist es ausserordentlich wichtig, nicht einfach blind ein Modell anzupassen oder ein statistisches Verfahren anzuwenden. Die Daten sollten immer mit Hilfe von geeigneten graphischen Darstellungsmethoden *und* den Kennzahlen dargestellt werden, da man nur auf diese Weise (teils unerwartete) Strukturen und Besonderheiten entdecken kann.

Im Folgenden werden die Daten mit x_1, \dots, x_n bezeichnet, wobei n der *Umfang* der Messreihe genannt wird. Im Fall der Messreihe der Methode A ist $n = 13$:

$$x_1 = 79.98, \quad x_2 = 80.04, \quad \dots, \quad x_{13} = 80.02$$

2.1.2. Darstellung von Messwerten

Bevor wir uns mit Kennzahlen und graphischen Darstellungen von Datensätzen auseinandersetzen, müssen wir Regeln für die Darstellung von Messwerten festlegen. Dazu benötigen wir die Begriffe *Nachkommastellen* und *signifikante Stellen*.

Als *Nachkommastellen* werden die in der dezimalen Darstellung einer Zahl verwendeten Ziffern rechts des Kommas bezeichnet. Im obigen Beispiel haben die Messpunkte

$$x_1 = 79.98, \quad x_2 = 80.04, \quad \dots, \quad x_{13} = 80.02$$

zwei Nachkommastellen.

Die *signifikanten Stellen* werden als die erste von Null verschiedene Stelle bis zur Rundungsstelle definiert. Die Rundungsstelle ist die letzte Stelle, die nach dem Runden noch angegeben werden kann. Im obigen Beispiel haben wir also *vier* signifikante Stellen.

Beispiel 2.1.2

Zahl	Anzahl Signifikante Stellen	Anzahl Nachkommastellen
98.76	4	2
0.009876	4	6
$987.6 \cdot 10^4$	4	1
$9.876 \cdot 10^6$	4	3

□

Bemerkungen:

- Ganze Zahlen haben keine Nachkommastellen.
- In manchen Fällen ist die Bestimmung der signifikanten Stellen unklar: Besitzt 20 eine, zwei oder sogar mehr signifikante Stellen? Je nach Zusammenhang ist eine Zahl exakt zu werten, wenn sie z. B. als natürliche Zahl verwendet wird; oder sie ist als gerundete Zahl zu werten, wenn sie als Zahlenwert zu einer physikalischen Grösse verwendet wird. Zu einer exakten Zahl stellt sich die Frage nach der Signifikanz nicht, da sie mit beliebig vielen Nachkoma-Nullen verlängert werden kann.
- Um zu einer mittels Messtechnik ermittelten Grösse beim Zahlenwert 20 eine Mehrdeutigkeit zu vermeiden, soll man die wissenschaftliche Schreibweise mit Zehnerpotenz-Faktor wählen. Im Fall von einer signifikanten Stelle also $2 \cdot 10^1$; im Fall von drei signifikanten Stellen $2.00 \cdot 10^1$.

Darstellung Rechenergebnis

Bei der Darstellung eines Rechenergebnis von Messwerten gelten folgende zwei Regeln:

1. Das Ergebnis einer *Addition/Subtraktion* bekommt genauso viele Nachkommastellen wie die Zahl mit den wenigsten Nachkommastellen.
2. Das Ergebnis einer *Multiplikation/Division* bekommt genauso viele signifikante Stellen wie die Zahl mit den wenigsten signifikanten Stellen.

Beispiel 2.1.3

Zahlen	Kleinste Anzahl Signifikante Stellen	Kleinste Anzahl Nachkommastellen	Ergebnis
$20.567 + 0.0007$		3	20.568
$12 + 1.234$		0	13
$12.00 + 1.234$		2	13.23
$12.000 + 1.234$		3	13.234
$1.234 \cdot 3.33$	3		4.11
$1.234 \cdot 0.0015$	2		0.0019

□

Bemerkungen:

- i. Eine Rundung sollte erst möglichst spät innerhalb des Rechengangs durchgeführt werden. Sonst können sich mehrere Rundungsabweichungen zu einer größeren Gesamtabweichung zusammensetzen. Um dies zu vermeiden, sollen in Zwischenrechnungen bekannte Größen mit mindestens einer Stelle mehr eingesetzt werden als im Endresultat angegeben werden kann.

2.1.3. Kennzahlen

Häufig ist es sinnvoll, Datensätze *numerisch* zusammenzufassen. Die Datensätze werden dabei auf eine oder mehrere Zahlen reduziert. Dazu verwenden wir meistens zwei *Kenngrößen*: Eine beschreibt die mittlere Lage der Messwerte und die andere die Variabilität oder Streuung dieser Messwerte. Mit Streuung meinen wir die „durchschnittliche“ Abweichung der Messwerte von der mittleren Lage.

Arithmetisches Mittel

Die bekannteste Grösse für die mittlere Lage ist der wohlbekannte Durchschnitt oder das

Arithmetische Mittel \bar{x}

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_n}{n} = \frac{1}{n} \sum_{i=1}^n x_i$$

Bemerkungen:

- i. Manchmal verwenden wir die Notation \bar{x}_n , wobei wir mit n den Umfang der Messreihe bezeichnen.

Beispiel 2.1.4 Messung der Schmelzwärme von Eis mit Methode A

Das arithmetische Mittel der $n = 13$ Messungen ist

$$\bar{x}_{13} = \frac{79.98 + 80.04 + \cdots + 80.03 + 80.02 + 80.00 + 80.02}{13} = 80.02$$

Wir summieren also alle Werte auf und dividieren die Summe durch die Anzahl der Werte.

Mit **Python** berechnen wir den Mittelwert wie folgt: ([zu R](#))

```
from pandas import Series, DataFrame
import pandas as pd

methodeA = Series([79.98, 80.04, 80.02, 80.04, 80.03, 80.03,
80.04, 79.97, 80.05, 80.03, 80.02, 80.00, 80.02])

methodeA.mean()

## 80.02076923076923
```

□

Empirische Varianz und Standardabweichung

Obwohl das arithmetische Mittel schon einiges über einen Datensatz aussagt, beschreibt er diesen aber nur unvollständig. Wir betrachten als Beispiel die folgenden beiden Datensätze von (fiktiven) Schulnoten:

2; 6; 3; 5 und 4; 4; 4

Beide haben denselben Mittelwert 4, aber die Verteilung der Daten um den Mittelwert ist sehr unterschiedlich. Im ersten Fall gibt es zwei gute und zwei schlechte Schüler, und im zweiten Fall sind alle Schüler gleich gut. Wir sagen, die Datensätze haben eine unterschiedliche Streuung um die Mittelwerte.

Wir wollen diese Streuung numerisch erfassen. Ein erster Ansatz besteht darin, dass man den Durchschnitt der *Unterschiede zum Mittelwert* nimmt. Im ersten Fall wäre dies

$$\frac{(2 - 4) + (6 - 4) + (3 - 4) + (5 - 4)}{4} = \frac{-2 + 2 - 1 + 1}{4} = 0$$

Im zweiten Fall ergibt der Durchschnitt der Unterschiede zum Mittelwert ebenfalls 0. Diese Methode trägt also nicht viel zur Beschreibung der Streuung bei, da die Unterschiede zum Mittelwert *negativ* werden können, und sich diese wie im obigen Fall aufheben können.

Der nächste Ansatz geht dahin, dass wir die Unterschiede zum Mittelwert durch die Absolutwerte der Unterschiede zum Mittelwert ersetzen. Im ersten Fall erhalten wir dann

$$\frac{|(2 - 4)| + |(6 - 4)| + |(3 - 4)| + |(5 - 4)|}{4} = \frac{2 + 2 + 1 + 1}{4} = 1.5$$

Die Noten weichen nun im Schnitt 1.5 Noten vom Mittelwert ab. Im zweiten Fall ist dieser Wert natürlich 0. Je grösser dieser Wert (der immer grösser gleich 0 ist), desto mehr unterscheiden sich die Daten bei gleichem Mittelwert voneinander. Dieser Wert für die Streuung heisst auch *mittlere absolute Abweichung*.

Da es sich mit Absolutwerten nicht einfach rechnen lässt (zum Beispiel Ableitungen), führen wir die *empirische Varianz* und *empirische Standardabweichung* als Mass für die Variabilität oder Streuung der Messwerte ein. Diese sind definiert durch

Empirische Varianz $\text{Var}(x)$ und Standardabweichung s_x

$$\text{Var}(x) = \frac{(x_1 - \bar{x}_n)^2 + (x_2 - \bar{x}_n)^2 + \cdots + (x_n - \bar{x}_n)^2}{n - 1} = \frac{1}{n - 1} \sum_{i=1}^n (x_i - \bar{x}_n)^2$$

$$s_x = \sqrt{\text{Var}(x)} = \sqrt{\frac{1}{n - 1} \sum_{i=1}^n (x_i - \bar{x}_n)^2}$$

Bemerkungen:

- Bei der Varianz quadrieren wir die Abweichungen $x_i - \bar{x}_n$, damit sich die Abweichungen vom Mittelwert nicht gegenseitig aufheben können. Der Nenner $n - 1$,

anstelle von n , ist mathematisch begründet¹.

- ii. Die Standardabweichung ist die Wurzel der Varianz. Da wir für die Berechnung der Varianz die Quadrate der Abstände zum Mittelwert verwendet haben, bekommen wir durch das Wurzelziehen wieder dieselbe Einheit wie bei den Daten selbst. Der Wert der empirischen Varianz hat keine physikalische Bedeutung. Wir wissen nur, je grösser der Wert, desto grösser die Streuung.

Beispiel 2.1.5 Messung der Schmelzwärme von Eis mit Methode A

Das arithmetische Mittel der $n = 13$ Messungen ist $\bar{x}_{13} = 80.02$ (siehe oben) und die empirische Varianz ergibt

$$\begin{aligned}\text{Var}(x) &= \frac{(79.98 - 80.02)^2 + (80.04 - 80.02)^2 + \dots + (80.00 - 80.02)^2 + (80.02 - 80.02)^2}{13 - 1} \\ &= 0.0005744\end{aligned}$$

Die empirische Standardabweichung ist dann

$$s_x = \sqrt{0.000574} = 0.02397$$

Somit ist die „mittlere“ Abweichung vom Mittelwert 0.02397 cal/g.

Für Methode B finden wir $\bar{x}_8 = 79.98$ und $s_x = 0.03137$ mit der analogen Interpretation.

□

Beispiel 2.1.6

Die empirische Varianz bzw. Standardabweichung (englisch: standard deviation) ist von Hand mühsam auszurechnen, deswegen benutzen wir **Python**. Für die Varianz verwenden wir den Befehl ([zu R](#))

```
methodeA.var()

## 0.0005743589743590099
```

und für die Standardabweichung (englisch: *standard deviation*)

```
methodeA.std()

## 0.023965787580611863
```

¹Eine genaue Begründung finden Sie im Anhang in Kapitel ??



2.1.4. Weitere Kennzahlen

Im Folgenden werden wir zwei alternative Kenngrößen studieren, und zwar den *Median* als Lagemass und die *Quartilsdifferenz* als Streuungsmass.

Median

Ein weiteres Lagemass für die mittlere Lage ist der *Median*. Es handelt sich dabei um den Wert, bei dem rund die Hälfte der Messwerte unterhalb von diesem Wert liegen. Ist beispielsweise bei einer Prüfung der Median 4.6, dann hat die Hälfte der Klasse eine Note unterhalb von 4.6. Umgekehrt liegen die Noten der anderen Hälfte *oberhalb* dieser Note.

Um den *Median* zu bestimmen, müssen wir die Daten zuerst der Grösse nach ordnen:

$$x_{(1)} \leq x_{(2)} \leq \cdots \leq x_{(n)}$$

Für die Daten der Methode A ergibt dies

79.97; 79.98; 80.00; 80.02; 80.02; 80.02; 80.03; 80.03; 80.03; 80.04; 80.04; 80.04; 80.05

In **Python** kann eine Messreihe wie folgt der Grösse nach geordnet werden ([zu R](#))

```
methodeA.sort_values

##    7      79.97
##    0      79.98
##   11     80.00
##    2      80.02
##   10     80.02
##   12     80.02
##    4      80.03
##    5      80.03
##    9      80.03
##    1      80.04
##    3      80.04
##    6      80.04
##    8      80.05
##  dtype: float64
```

Kapitel 2. Deskriptive Statistik

Der Median von diesen 13 Messungen ist dann der Wert der mittleren Beobachtung. Dies ist in diesem Fall der Wert der 7. Beobachtung:

79.97; 79.98; 80.00; 80.02; 80.02; 80.02; 80.03; 80.03; 80.03; 80.04; 80.04; 80.04; 80.05

Der Median des Datensatzes der Methode A lautet somit 80.03. 6 Beobachtungen sind kleiner oder gleich 80.03 und 6 Messwerte sind grösser oder gleich 80.03. In diesem Beispiel ist die Anzahl der Daten ungerade, und somit gibt es eine mittlere Beobachtung. Ist die Anzahl der Daten gerade, so gibt es zwei gleichwertige mittlere Beobachtungen. Als Median benützen wir in diesem Fall den Mittelwert der beiden mittleren Beobachtungen. Der Datensatz der Methode B hat 8 Beobachtungen. Wir ordnen den Datensatz der Grösse nach und definieren als Median den Durchschnitt von der 4. und 5. Beobachtung:

79.94; 79.95; 79.97; 79.97; 79.97; 79.97; 79.97; 80.02; 80.03

$$\frac{79.97 + 79.97}{2} = 79.97$$

Der Median der Daten der Methode B ist 79.97. Das heisst, die Hälfte der Messwerte ist kleiner oder gleich diesem Wert und die andere Hälfte ist grösser oder gleich diesem Wert. Die Werte der beiden mittleren Beobachtungen sind hier zufällig gleich, dies ist im Allgemeinen natürlich nicht der Fall.

Beispiel 2.1.7

Mit **Python** bestimmen wir den Median für Methode A wie folgt : ([zu R](#))

```
methodeA.median()  
  
## 80.03
```

Und für Methode B

```
methodeB = Series([80.02, 79.94, 79.98, 79.97, 79.97, 80.03,  
79.95, 79.97])  
  
methodeB.median()  
  
## 79.97
```

□

Wir haben nun zwei Lagemasse für die Mitte eines Datensatzes: das arithmetische Mittel und den Median. Welches sind die Vorzüge der jeweiligen Lagemasse? Eine Eigenschaft des Medians ist die *Robustheit*. Der Median wird weniger stark durch extreme Beobachtungen beeinflusst als das arithmetische Mittel.

Beispiel 2.1.8 Messung der Schmelzwärme von Eis mit Methode A

Bei der grössten Beobachtung ($x_9 = 80.05$) ist ein Tippfehler passiert und $x_9 = 800.5$ eingegeben worden. Das arithmetische Mittel ist dann

$$\bar{x}_{13} = 135.44$$

Der Median ist aber nach wie vor

$$x_{(7)} = 80.03$$

Das arithmetische Mittel wird also durch Veränderung einer Beobachtung sehr stark beeinflusst, während der Median hier gleich bleibt – er ist *robust*.

□

Quartile

Das *untere Quartil* ist derjenige Wert, bei welchem 25 % aller Beobachtungen kleiner oder gleich gross und 75 % grösser oder gleich gross wie dieser Wert sind. Dementsprechend ist das *obere Quartil* derjenige Wert, bei dem 75 % aller Beobachtungen kleiner oder gleich gross und 25 % grösser oder gleich gross sind wie dieser Wert.

Allerdings gibt es für die meisten Datensätze nicht *exakt* 25 % der Anzahl Beobachtungen. Was in solchen Fällen gemacht wird, ist leider nicht einheitlich geregelt.

Beispiel 2.1.9

Die Methode A hat $n = 13$ Messpunkte und 25 % dieser Anzahl ist 3.25. Nun gibt es mehrere Möglichkeiten, das untere Quartil zu definieren

- Wir *wählen* in diesem Fall den nächstgrösseren Wert $x_{(4)}$ als unteres Quartil. Dann sind etwas mehr als 25 % der Werte kleiner oder gleich diesem Wert:

79.97; 79.98; 80.00; **80.02**; 80.02; 80.02; 80.03; 80.03; 80.03; 80.04; 80.04; 80.04; 80.05

Das untere Quartil ist somit 80.02. Rund ein Viertel der Messwerte sind kleiner oder gleich gross wie dieser Wert.

- Wir *könnten* aber auch abrunden, da 3.25 näher bei 3 als bei vier liegt und erhalten dann als unteres Quartil 80.00.
- Wir *könnten* auch den Mittelwert zwischen dem 3. und 4. Wert nehmen und erhalten als unteres Quartil 80.01.
- Wir *könnten* auch linear extrapolieren. Wir nehmen dann 0.25 vom Unterschied zwischen dem 3. und 4. Wert und erhalten als unteres Quartil 80.005.

□

Es gibt noch weitere Überlegungen, wie wir das untere bzw. das obere Quartil definieren können. Je nach Software wird eine dieser Definitionen von Quartilen verwendet.

Merkregel: Quartile

Das *untere Quartil* ist derjenige Wert, bei welchem *etwa* 25 % aller Beobachtungen kleiner oder gleich gross und 75 % grösser oder gleich gross wie dieser Wert sind.

Dementsprechend ist das *obere Quartil* derjenige Wert, bei dem etwa 75 % aller Beobachtungen kleiner oder gleich gross und 25 % grösser oder gleich gross sind wie dieser Wert.

Das Wort „*etwa*“ bedeutet hier „so nahe wie möglich bei“.

Die Software **Python** kennt keine eigenen Befehle für die Quartile. Wir können allerdings den allgemeineren Befehl **quantile** benutzen (die Quantile werden wir gleich noch genauer behandeln).

Beispiel 2.1.10

Für das untere Quartil der Methode A lautet der Befehl ([zu R](#))

```
methodeA.quantile(q=.25)  
## 80.02
```

und damit sind 25 % der Werte kleiner oder gleich 80.02 und 75 % grösser oder gleich 80.04.

Für das obere Quartil gilt

```
methodeA.quantile(q=.75)  
## 80.04
```

und damit sind 75 % der Werte kleiner oder gleich 80.04 und 25 % grösser oder gleich 80.04.

Für die Methode *B* erhalten wir

```
methodeB.quantile(q=.25)  
## 79.965
```

□

Bemerkungen:

- i. Die verschiedenen Definitionen ergeben natürlich auch verschiedene Werte für die Quartile, wie wir in Beispiel 2.1.9 gesehen haben.
- ii. Die Werte für die Quartile sind für die verschiedenen Definitionen allerdings meist sehr ähnlich oder gleich und ändern an unserer Interpretations kaum etwas.
- iii. Wegen der Robustheit der Quartile verschieben sich die Werte der Quartile für die verschiedenen Definitionen um höchstens einen Wert. Sind die Datensätze gross, so spielt dies kaum eine Rolle.

Beispiel 2.1.11

Wir betrachten die verschiedenen Definitionen für die Quartile, die in **Python** implementiert sind. Diese werden mit der Option `interpolation=...` festgelegt. Für die Details der Definitionen der unterschiedlichen Methoden verweisen wir auf die Hilfsfunktionen in Ipython.

Für das untere Quartil der Methode *A* erhalten wir ([zu R](#))

```
methodeA.quantile(q=.25, interpolation="linear")  
methodeA.quantile(q=.25, interpolation="lower")  
methodeA.quantile(q=.25, interpolation="higher")  
methodeA.quantile(q=.25, interpolation="midpoint")  
methodeA.quantile(q=.25, interpolation="nearest")  
  
## 80.02  
## 80.02  
## 80.02  
## 80.02  
## 80.02
```

Kapitel 2. Deskriptive Statistik

In diesem Fall gibt es gar keine Unterschiede zwischen den verschiedenen Definitionen.

Für das untere Quartil der Methode *B* erhalten wir ([zu R](#))

```
methodeB.quantile(q=.25, interpolation="linear")
methodeB.quantile(q=.25, interpolation="lower")
methodeB.quantile(q=.25, interpolation="higher")
methodeB.quantile(q=.25, interpolation="midpoint")
methodeB.quantile(q=.25, interpolation="nearest")

## 79.965
## 79.95
## 79.97
## 79.96
## 79.97
```

Die Unterschiede sind hier sehr klein und ändern die Interpretation kaum. Es sind etwa 25 % der Werte kleiner oder gleich 79.97 (oder 79.95 oder 79.96). Diese Werte sind sehr nahe zusammen.

□

Quartilsdifferenz

Die Quartilsdifferenz

$$\text{oberes Quartil} - \text{unteres Quartil}$$

ist ein weiteres Streuungsmass für die Daten. Es misst die Länge des Intervalls, das etwa die Hälfte der mittleren Beobachtungen enthält. Je kleiner dieses Streuungsmass ist, desto näher liegt die Hälfte aller Werte beim Median und desto kleiner ist die Streuung. Dieses Streuungsmass ist robust.

Beispiel 2.1.12

So ist die Quartilsdifferenz bei der Methode *A*

$$80.04 - 80.02 = 0.02$$

([zu R](#))

```

q75, q25 = methodeA.quantile(q = [.75, .25])
iqr = q75 - q25
iqr

## 0.02

```

Rund die Hälfte aller Messwerte liegt also in einem Bereich der Länge 0.02.



Quantile

Mit den *Quantilen* können wir das Konzept der Quartile auf jede andere Prozentzahl verallgemeinern. So ist das 10 %-Quantil derjenige Wert, wo 10 % der Werte kleiner oder gleich und 90 % der Werte grösser oder gleich diesem Wert sind.

Definition: Quantile

Das *empirische α -Quantil* ist anschaulich gesprochen der Wert, bei dem $\alpha \times 100\%$ der Datenpunkte kleiner oder gleich und $(1 - \alpha) \times 100\%$ der Punkte grösser oder gleich sind.

Bemerkungen:

- Der empirische Median ist das empirische 50 %-Quantil; das empirische 25 %-Quantil ist das untere Quartil und das empirische 75 %-Quantil das obere Quartil.
- Die exakte Definition für das empirische α -Quantil lautet:

$$\frac{1}{2}(x_{(\alpha n)} + x_{(\alpha n+1)}) , \text{ falls } \alpha \cdot n \text{ eine natürliche Zahl ist,}$$

$$x_{(k)} \text{ wobei } k \text{ die Zahl } \alpha \cdot n \text{ aufgerundet ist , falls } \alpha \cdot n \notin \mathbb{N}.$$

Diese Definition wird in späteren Kapitel zum tieferen Verständnis noch benötigt - wie zum Beispiel im Falle der QQ-Plots.

Beispiel 2.1.13 Messung der Schmelzwärme von Eis mit Methode A

Wir bestimmen Median, unteres und oberes Quartil mit Hilfe der Definition oben.

Es sind $n = 13$ Messwerte, die wir zuerst der Grösse nach ordnen: der kleinste Wert ist $x_{(1)} = 79.97$, der drittgrösste Wert $x_{(3)} = 80.00$, der grösste Wert $x_{(13)} = 80.05$. Wir wollen das 25 %-Quantil, den Median und das 75 %-Quantil bestimmen. Im Fall vom 25 %-Quantil ist dann $\alpha = 0.25$, also

$$\alpha \cdot n = 0.25 \cdot 13 = 3.25$$

was keine natürliche Zahl ist; folglich runden wir 3.25 auf 4 auf und erhalten für das 25 %-Quantil $x_{(4)} = 80.02$. Im Fall vom Median ist $\alpha = 0.5$, also

$$\alpha \cdot n = 0.5 \cdot 13 = 6.5$$

was keine natürliche Zahl ist; folglich runden wir 6.5 auf 7 auf und erhalten für den Median $x_{(7)} = 80.03$. Im Fall vom 75 %-Quantil ist $\alpha = 0.75$, also

$$\alpha \cdot n = 0.75 \cdot 13 = 9.75$$

was keine natürliche Zahl ist; folglich runden wir 9.75 auf 10 auf und erhalten für das 75 %-Quantil den Beobachtungswert $x_{(10)} = 80.04$.

□

Beispiel 2.1.14

Das 10 %- und 70 %-Quantil der Methode A berechnen wir wie folgt: ([zu R](#))

```
methodeA.quantile(q=.1)
methodeA.quantile(q=.7)

## 79.984
## 80.034
```

Rund 10 % der Messwerte sind kleiner oder gleich 79.98 . Entsprechend sind rund 70 % der Messwerte kleiner oder gleich 80.03.

□

Beispiel 2.1.15

In einer Schulkasse mit 24 SchülerInnen gab es an einer Prüfung folgende Noten:

4.2, 2.3, 5.6, 4.5, 4.8, 3.9, 5.9, 2.4, 5.9, 6, 4, 3.7, 5, 5.2, 4.5, 3.6, 5, 6, 2.8, 3.3, 5.5, 4.2, 4.9, 5.1

Wir berechnen nun mit **Python** verschiedene Quantile: ([zu R](#))

```
noten = Series([4.2, 2.3, 5.6, 4.5, 4.8, 3.9, 5.9, 2.4, 5.9,
                6, 4, 3.7, 5, 5.2, 4.5, 3.6, 5, 6, 2.8, 3.3,
                5.5, 4.2, 4.9, 5.1])

noten.quantile(q = np.linspace(start=.2, stop=1, num=5))

## 0.2      3.66
## 0.4      4.26
## 0.6      4.98
## 0.8      5.54
## 1.0      6.00
## dtype: float64
```

Rund 20 % der SchülerInnen haben also eine 3.6 oder waren schlechter. Genau 20 % der SchülerInnen ist nicht möglich, da dies 4.8 SchülerInnen entsprechen würde. Das 60 %-Quantil besagt, dass 60 Prozent der SchülerInnen eine 4.9 haben oder schlechter waren. Folglich haben 40 % der SchülerInnen eine Note 4.9 oder besser.

Bemerkungen:

- i. Auch hier ändert sich die Aussage nicht gross, wenn wir sagen, dass rund 20 % der SchülerInnen 3.7 oder schlechter waren.



2.1.5. Graphische Methoden

Histogramm

Einen graphischen Überblick über die auftretenden Werte erhalten wir mit einem sogenannten *Histogramm*. Histogramme helfen uns bei der Frage, in welchem Wertebereich besonders viele Datenpunkte liegen. Ist die Datenmenge gross, so macht es keinen Sinn, alle Werte einzeln zu betrachten. Wir bilden sogenannte *Klassen*, die jeweils einen Ausschnitt des Beobachtungsbereiches darstellen. Um ein Histogramm

Kapitel 2. Deskriptive Statistik

zu zeichnen, bildet man Klassen (einfachheitshalber mit konstanter Breite) und zählt, wie viele Beobachtungen in jede Klasse fallen. Es gibt verschiedene Arten von Histogrammen; wir behandeln hier nur die gebräuchlichste.

Beispiel 2.1.16

In Abbildung 2.1 sehen wir ein Histogramm von dem Ergebnis eines IQ-Testes von 200 Personen.

- Die Breite der Klassen wurde mit 10 IQ-Punkten festgelegt und ist für jede Klasse gleich.
- Die Höhe der Balken gibt die Anzahl Personen an, die in diese Klasse fallen. Zum Beispiel fallen ca. 14 Personen in die Klasse zwischen 120 und 130 IQ-Punkten.

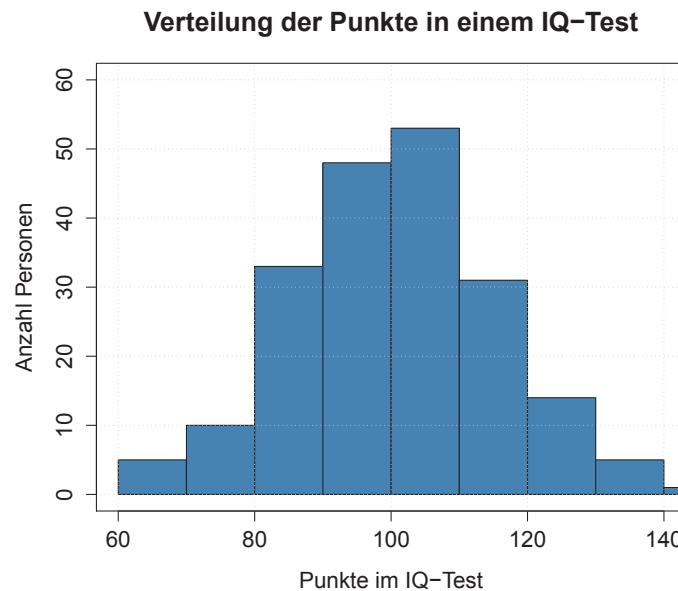


Abbildung 2.1.: Histogramm von dem IQ-Test Ergebnis von 200 Personen.



Schrittweise Konstruktion eines Histogramms:

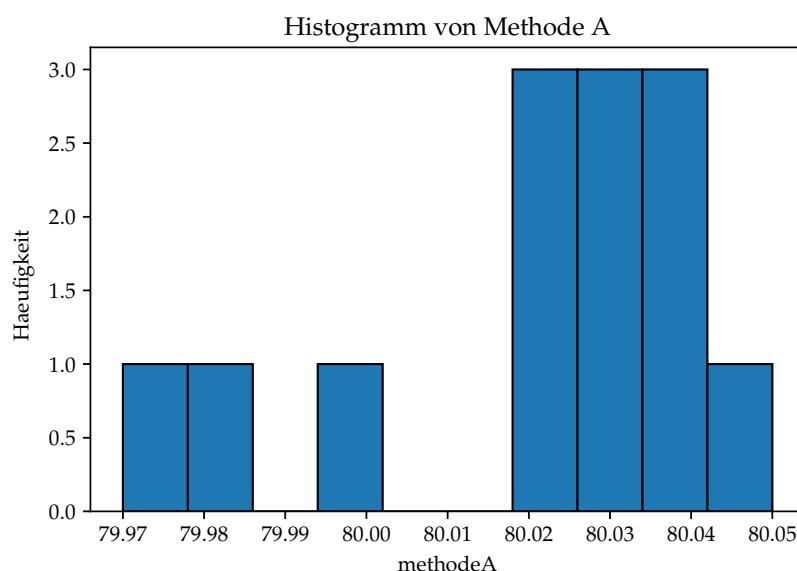
- Wir teilen die Datenmenge in Klassen ein. Für die Festlegung der Anzahl der Klassen bzw. Rechtecke existieren verschiedene Faustregeln: bei weniger als 50 Messungen ist die Klassenzahl 5 bis 7, bei mehr als 250 Messungen wählt man

10 bis 20 Klassen.² Im einfachsten Fall wird die gleiche Breite für alle Klassen gewählt, was aber nicht unbedingt der Fall sein muss.

- Dann zeichnen wir für jede Klasse einen Balken, dessen Höhe proportional zur Anzahl Beobachtungen in dieser Klasse ist.
- Dividieren wir die Anzahl Beobachtungen in einer Klasse durch die Gesamtzahl der Beobachtungen, so erhalten wir den prozentualen Anteil einer Klasse zur Gesamtbeobachtung.

Beispiel 2.1.17

Für die Methode A sieht das Histogramm wie folgt aus:



Es wurde mit folgendem Code erzeugt: ([zu R](#))

```
import pandas as pd
from pandas import DataFrame, Series
import matplotlib.pyplot as plt

methodeA = Series([79.98, 80.04, 80.02, 80.04, 80.03, 80.03,
80.04, 79.97, 80.05, 80.03, 80.02, 80.00, 80.02])

methodeB = Series([80.02, 79.94, 79.98, 79.97, 79.97, 80.03,
79.95, 79.97])

methodeA.plot(kind="hist", edgecolor="black")
```

²Gegebenenfalls kann man die Anzahl der Klassen k auch nach der Sturges-Regel berechnen: $k = 1 + \log_2 n = 1 + 3.3 \cdot \log_{10} n$, wobei n die Anzahl Messungen ist.

```
plt.title("Histogramm von Methode A")
plt.xlabel("methodeA")
plt.ylabel("Haeufigkeit")

plt.show()
```

Bemerkungen:

- i. **pandas** selbst kennt keine graphischen Darstellungsmöglichkeiten, greift aber auf die Bibliothek **matplotlib** zurück.
- ii. Hier wird die allgemeine Pandas-Methode **plot** für Graphiken verwendet. Wir wählen dann die Option **... (kind="hist", ...)** für das Histogramm.
- iii. **pandas** wählt standardmäßig 10 Balken. Dies kann mit **... (kind="hist", ..., bins=7)** beispielweise auf sieben Balken geändert werden.
- iv. Bedeutung der Anzahlen (Frequency): Da 10 Klassen gewählt wurden und die Werte im Bereich [79.97, 80.05] liegen, ist die Balkenbreite $(80.05 - 79.97)/10 = 0.008$.

In der 1. Klasse 79.97-79.978 sind die Beobachtungen mit dem 79.97 berücksichtigt; in der 2. Klasse 79.98; usw.

Die Frage stellt sich noch, was mit Werten geschieht, die genau auf einer Balkengrenze liegen. Diese Werte kann man im linken oder rechten Balken berücksichtigen, aber *nicht* in beiden. Je nach Wahl würde das Histogramm etwas anders aussehen. Wie das genau passiert (auch hier gibt es verschiedene Methoden) ist für uns irrelevant, da bei grossen Datensätzen solche Überlegungen kaum eine Rolle spielen.

- v. Mit dem **Python**-Befehl lassen sich auch die Anzahl und die Breiten der Klassen festlegen, Überschriften ändern, usw. (siehe Übungen).

□

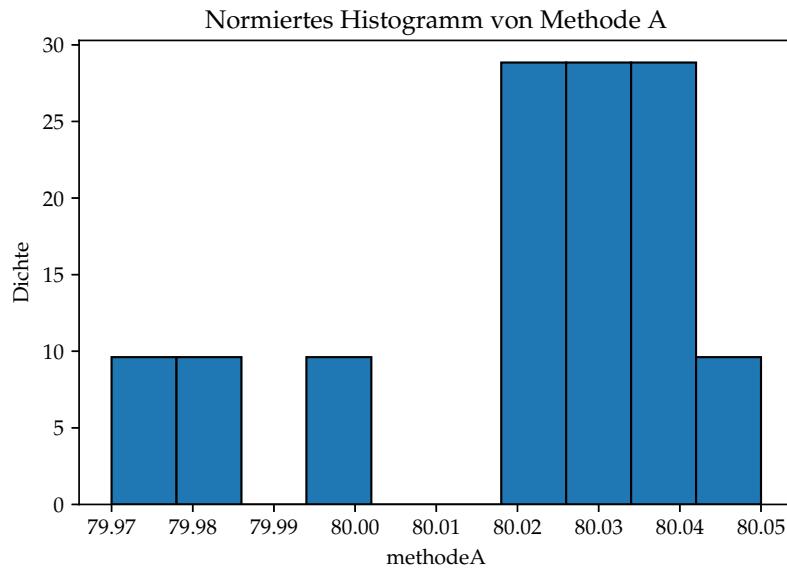
Beispiel 2.1.18

Im Histogramm oben entspricht die Höhe der Balken gerade der Anzahl der Beobachtungen in einer Klasse. Oft ist es besser und übersichtlicher, wenn wir die Balkenhöhe so wählen, dass die *Balkenfläche* dem prozentualen Anteil der jeweiligen Beobachtungen an der Gesamtanzahl Beobachtungen entspricht. Die *Gesamtfläche aller Balken* muss dann gleich eins sein.

Kapitel 2. Deskriptive Statistik

(zu R)

```
methodeA.plot(kind="hist", normed=True, edgecolor="black")  
  
plt.title("Normiertes Histogramm von Methode A")  
plt.xlabel("methodeA")  
plt.ylabel("Dichte")  
  
plt.show()
```



Auf der vertikalen Achse sind nun die Dichten angegeben. Wir können also herauslesen, dass sich $(80.018 - 80.026) \cdot 28.846 = 0.23$, also etwa 23 % der Daten zwischen 80.018 und 80.026 befinden.

Die Balkenhöhe ermittelt sich, indem man die Anzahl Beobachtungen in einem Balken mit $\frac{1}{n}$ multipliziert, wobei n die Gesamtanzahl Beobachtungen bezeichnet, und diese Zahl durch die Balkenbreite dividiert. Für unser Beispiel oben sind 3 Beobachtungen im Intervall [80.018, 80.026]. Die Balkenhöhe ist dann

$$\frac{\frac{1}{13} \cdot 3}{0.008} = 28.8462$$

□

Diese Darstellung hat den Vorteil, dass man Messungen mit unterschiedlichen Um-

f  gen besser miteinander vergleichen kann. W  rde man also mit Methode A nun eine Messung mit 30 Beobachtungen durchf  ren, liessen sich mit Dichten besser die Verteilungen von Messwerten auf die jeweiligen Klassen vergleichen.

Boxplot

Der *Boxplot* (siehe Abbildung 2.2) besteht aus

- einem Rechteck, dessen H  he vom empirischen 25 %- und vom 75 %-Quantil begrenzt wird,
- Linien, die von diesem Rechteck bis zum kleinsten- bzw. gr  sst „normalen“ Wert f  hren (per Definition ist ein „normaler“ Wert h  chstens 1.5 mal die Quartilsdifferenz von einem der beiden Quartile entfernt),
- einem horizontalen Strich f  r den Median,
- kleinen Kreisen, die Ausreisser markieren.

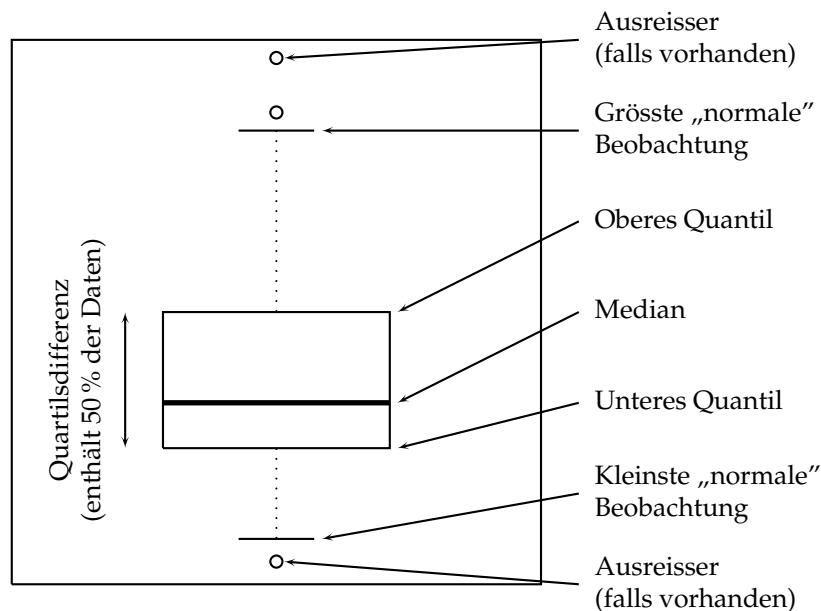
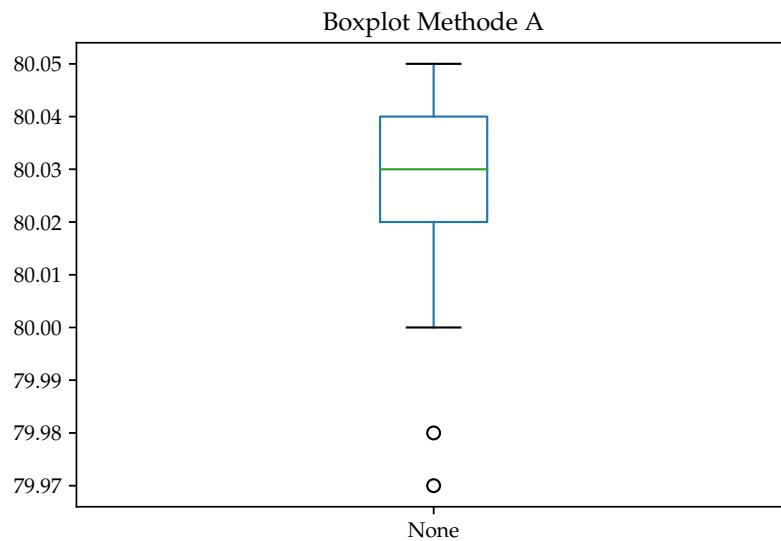


Abbildung 2.2.: Boxplot

Beispiel 2.1.19

Der Boxplot zur Methode A sieht wie folgt aus: [\(zu R\)](#)

```
methodeA.plot(kind="box", title="Boxplot Methode A")
```



Bemerkungen:

- i. Die Hälfte der Beobachtungen befindet sich zwischen dem oberen Quartil 80.04 und dem unteren Quartil 80.02, mit Quartilsdifferenz 0.02
- ii. Der Median liegt bei 80.03.
- iii. Der „normale“ Bereich der Werte liegt zwischen 80.00 und 80.05.
- iv. Wir haben zwei Ausreißer 79.97 und 79.98.
- v. Die Punkte i) und ii) hatten wir schon bei den Quantilen berechnet. Der Boxplot stellt somit unsere Berechnungen graphisch dar.

□

Der Boxplot ist vor allem dann geeignet, wenn man die Verteilungen der Daten in verschiedenen Gruppen (die im Allgemeinen verschiedenen Versuchsbedingungen entsprechen) vergleichen will.

Beispiel 2.1.20

Bei unserem Einführungsbeispiel der Schmelzwärme haben wir zwei Methoden zu deren Bestimmung verwendet. Somit können wir die Boxplots auch gegenüberstellen und die Methoden miteinander vergleichen (siehe Abbildung 2.3).

Kapitel 2. Deskriptive Statistik

(zu R)

```
methode = DataFrame({  
    "methodeA": methodeA,  
    "methodeB": methodeB  
})  
  
methode.plot(kind="box", title="Boxplot von Methode A und B")
```

Hier wurde aus den beiden **Series** ein **DataFrame methode** erzeugt.

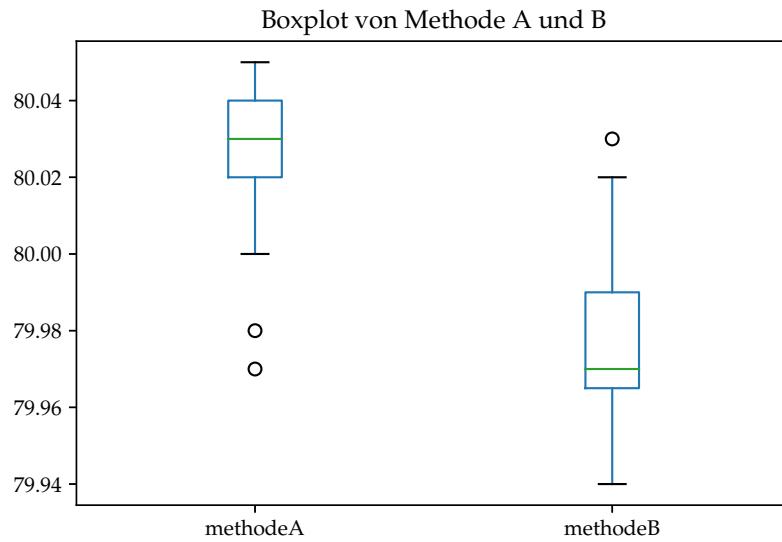


Abbildung 2.3.: Boxplots für die zwei Methoden zur Bestimmung der Schmelzwärme von Eis.

Bemerkungen:

- i. Methode A liefert die grösseren Werte als Methode B, da der Median von A grösser ist.
- ii. Die Daten von Methode A haben weniger Streuung als die Daten von Methode B, da das Rechteck weniger hoch ist (Quartilsdifferenz!).



Empirische kumulative Verteilungsfunktion

Eine weitere graphische Darstellung der Daten ist die *empirische kumulative Verteilungsfunktion*. Diese Darstellung hat den Vorteil gegenüber einem Histogramm, dass man den Median sehr leicht ablesen kann.

Die *empirische kumulative Verteilungsfunktion* $F_n(\cdot)$ ist eine Treppenfunktion, die wie folgt erzeugt wird: links von $x_{(1)}$ ist die Funktion gleich null und bei jedem $x_{(i)}$ wird ein Sprung der Höhe $1/n$ gemacht (falls ein Wert mehrmals vorkommt, ist der Sprung das entsprechende Vielfache von $1/n$). Im folgenden Beispiel wird dieses Vorgehen konkret durchgeführt.

Beispiel 2.1.21 Methode A der Schmelzwärme

In Abbildung 2.4 ist die empirische kumulative Verteilungsfunktion der Methode A aufgezeichnet.

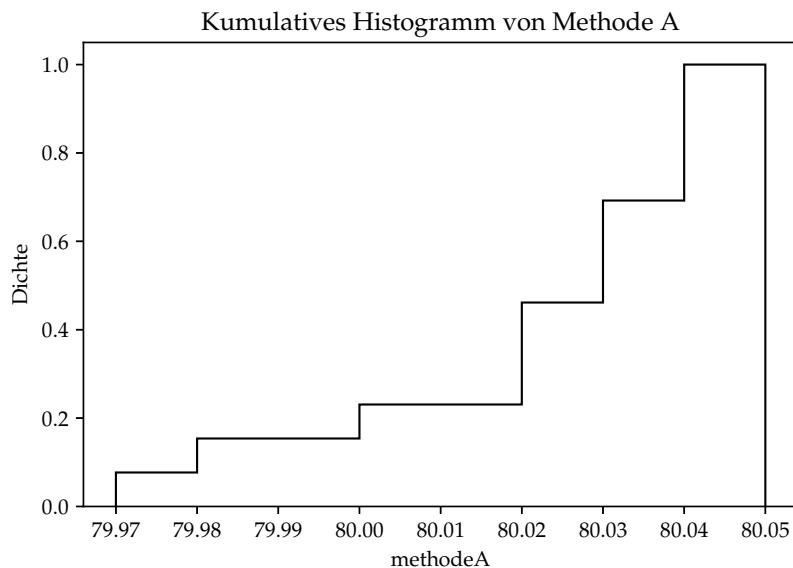


Abbildung 2.4.: Empirische kumulative Verteilungsfunktion der Messungen der Schmelzwärme von Eis mit Methode A.

Sie entsteht wie folgt:

- Links von 79.97 ist die Funktion 0, da es keinen kleineren Beobachtungswert hat.
- Bei 79.97 macht die Funktion einen Sprung auf $n = 1/13 \approx 0.077$.

- Die Funktion bleibt dann gleich bis 80.00, da es vorher keinen zusätzlichen Beobachtungswert gibt. Bei 80.00 macht die Funktion wieder einen Sprung um 0.077 nach oben, weil es dort einen Messwert hat.
- Bei 80.02 macht die Funktion einen Sprung um $3 \cdot 0.077$ nach oben, da es dort 3 Beobachtungswerte gibt.
- usw.
- Bei 80.05 macht die Funktion ihren letzten Sprung, der Funktionswert wird 1.

□

Was können wir aus der kumulativen Verteilungsfunktion herauslesen?

- Bei 0.5 auf der vertikalen Achse haben wir gerade die Hälfte aller Werte aufsummiert. Zeichnen wir von 0.5 eine horizontale Linie (siehe grüne Linie in Abbildung 2.5), wird die kumulative Verteilungsfunktion bei 80.03 geschnitten. Das entspricht gerade dem Median.
- Dort, wo die kumulative Funktion einen grossen Sprung macht, hat es auch viele Beobachtungswerte. Das heisst, die meisten Beobachtungswerte liegen hier zwischen 80.02 und 80.04. Die Werte entsprechen aber gerade dem unteren und oberen Quartil. Man vergleiche die Funktion mit dem zugehörigen Boxplot.

Beispiel 2.1.22

□

Allgemein

Die **empirische kumulative Verteilungsfunktion** ist definiert als

$$F_n(a) = \frac{1}{n} \text{Anzahl}\{i \mid x_i \leq a\}.$$

Beispiel 2.1.23

□

Mit **Python** lässt sich die kumulative Verteilungsfunktion in der Abbildung 2.5 folgendermassen aufzeichnen: ([zu R](#))

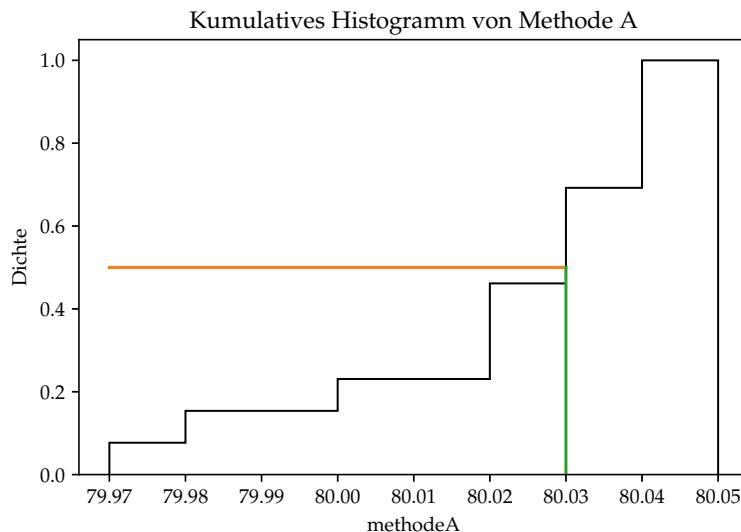


Abbildung 2.5.: Empirische kumulative Verteilungsfunktion der Messungen der Schmelzwärme von Eis mit Methode A.

```
methodeA.plot(kind="hist", cumulative=True, histtype="step",
normed=True, bins=8, edgecolor="black")
```

2.2. Deskriptive Statistik zweidimensionaler Daten

Bei zweidimensionalen Daten werden an einem Versuchsobjekt jeweils *zwei* verschiedene Größen gemessen. So wird beispielsweise an einer Gruppe von Menschen jeweils die Körpergrösse *und* das Körpergewicht gemessen.

Beispiel 2.2.1 Weinkonsum und Mortalität

Wir betrachten als Beispiel einen Datensatz (siehe Tabelle 2.2), der den durchschnittlichen Weinkonsum (in Liter pro Person und Jahr) und die Sterblichkeit (Mortalität) aufgrund von Herz- und Kreislauferkrankungen (Anzahl Todesfälle pro 1000 Personen zwischen 55 und 64 Jahren pro Jahr) in 18 industrialisierten Ländern umfasst³. Es stellt sich nun die Frage, ob diese Daten suggerieren, dass es einen Zusammenhang zwischen der Sterblichkeitsrate aufgrund von Herzkreislauferkrankung und Weinkonsum gibt.

Ein kurzer Blick auf die Tabelle zeigt, dass ein höherer Weinkonsum eher weniger Todesfälle wegen Herz- und Kreislauferkrankheiten zur Folge hat.

³A.S.St.Leger, A.L.Chochrane, and F.Moore, "Factors Associated with Cardiac Mortality in Developed Countries with Particular Reference to the Consumption of Wine." *Lancet*, 1979

Land	Weinkonsum	Mortalität Herzerkrankung
Norwegen	2.8	6.2
Schottland	3.2	9.0
Grossbritannien	3.2	7.1
Irland	3.4	6.8
Finnland	4.3	10.2
Kanada	4.9	7.8
Vereinigte Staaten	5.1	9.3
Niederlande	5.2	5.9
New Zealand	5.9	8.9
Dänemark	5.9	5.5
Schweden	6.6	7.1
Australien	8.3	9.1
Belgien	12.6	5.1
Deutschland	15.1	4.7
Österreich	25.1	4.7
Schweiz	33.1	3.1
Italien	75.9	3.2
Frankreich	75.9	2.1

Table 2.2.: Weinkonsumation (Liter pro Person pro Jahr) und Mortalität aufgrund von Herzkreislauferkrankung (Todesfälle pro 1000) in 18 Ländern.

□

2.2.1. Graphische Darstellung: Streudiagramm

Ein wichtiger Schritt in der Untersuchung zweidimensionaler Daten ist die graphische Darstellung. Dies geschieht meist über ein sogenanntes *Streudiagramm* (engl.: *Scatterplot*). Dabei werden jeweils zwei Messungen als Koordinaten von Punkten in einem Koordinatensystem interpretiert und dargestellt.

Beispiel 2.2.2

In unserem Beispiel stellt ein Land eine Versuchseinheit dar, und es wird die Grösse „Weinkonsum“ x_1, \dots, x_{18} und die Grösse „Mortalität“ y_1, \dots, y_{18} gemessen. Wenn wir die Daten in der Form $(x_1, y_1), \dots, (x_{18}, y_{18})$ schreiben, interessiert man sich in erster Linie für die Zusammenhänge und Abhängigkeiten zwischen den Variablen x und y . Die Abhängigkeit zwischen den beiden Messgrössen kann man aus dem

Kapitel 2. Deskriptive Statistik

Streudiagramm ersehen, welches die Daten als Punkte in der Ebene darstellt: Die i -te Beobachtung (i -tes Land) entspricht dem Punkt mit Koordinaten (x_i, y_i) . Abbildung 2.6 zeigt das Streudiagramm für die Messgrößen „Weinkonsum“ (x_1, x_2, \dots, x_{18}) und „Mortalität“ (y_1, y_2, \dots, y_{18}). Man sieht einen klaren monoton fallenden Zusammenhang: Länder mit hohem Weinkonsum haben also eine Tendenz zu einer tieferen Mortalitätsrate wegen Herz- und Kreislaufkrankheiten.

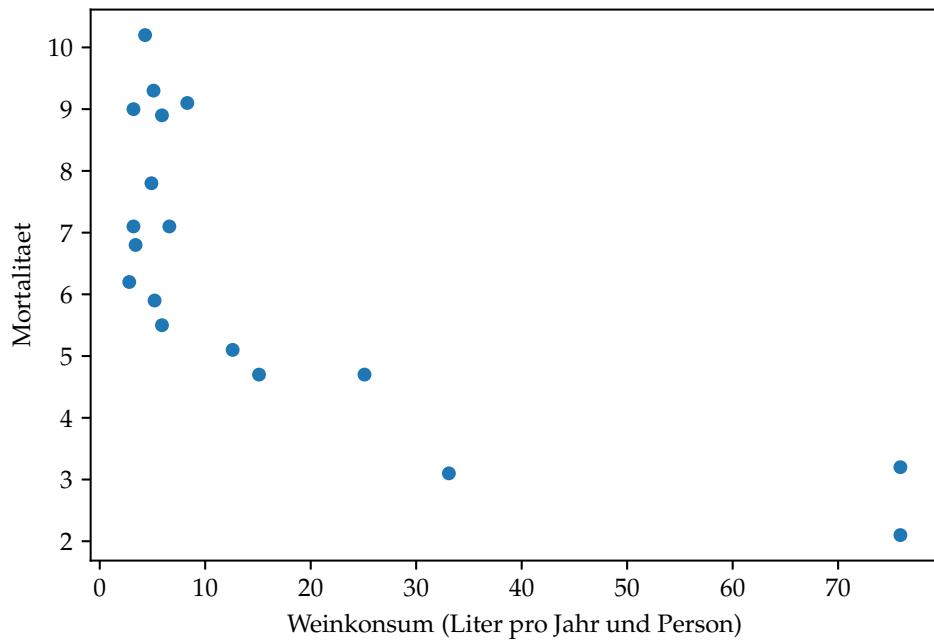


Abbildung 2.6.: Streudiagramm für die Mortalität und den Weinkonsum in 18 industrialisierten Ländern.

Das Streudiagramm in Abbildung 2.6 wurde mit **Python** erstellt. ([zu R](#))

```
import pandas as pd
from pandas import DataFrame, Series
import numpy as np

mort = DataFrame({
    "wine": ([2.8, 3.2, 3.2, 3.4, 4.3, 4.9, 5.1, 5.2, 5.9, 5.9,
              6.6, 8.3, 12.6, 15.1, 25.1, 33.1, 75.9, 75.9]),
    "mor": ([6.2, 9.0, 7.1, 6.8, 10.2, 7.8, 9.3, 5.9, 8.9, 5.5,
              7.1, 9.1, 5.1, 4.7, 4.7, 3.1, 3.2, 2.1])
})

mort.plot(kind="scatter", x="wine", y="mor")
```

```
plt.xlabel("Weinkonsum (Liter pro Jahr und Person)")  
plt.ylabel("Mortalitaet")  
  
plt.show()
```

Bemerkungen:

- i. Die Schlussfolgerung, dass hoher Weinkonsum gesund ist, ist voreilig und vermutlich *falsch*. Es scheint, dass höherer Weinkonsum zu weniger Toten wegen Herz- und Kreislaufkrankheiten führt. Der Einfluss des höheren Weinkonsums auf andere Körperorgane (z.B. Leber) oder auf die Anzahl Verkehrsunfälle, wird hier *nicht* untersucht.
- ii. Obwohl sich aufgrund des Streudiagramms ein Zusammenhang zwischen Weinkonsum und Mortalität *erahnen* lässt, muss nicht zwingend ein *kausaler* Zusammenhang zwischen den beiden Größen vorhanden sein.

□

2.2.2. Einfache lineare Regression

Wir haben im vorangehenden Beispiel eine negative (je mehr desto weniger) Abhängigkeit zwischen Mortalität und Weinkonsum festgestellt. Oft wird angenommen, dass diese Abhängigkeit sehr einfach ist, nämlich *linear*.

Beispiel 2.2.3 Zusammenhang Seitenzahl-Preis eines Buches

Wir erklären das Modell der einfachen linearen Regression zunächst mit einem fiktiven Beispiel. Je dicker ein Roman (Hardcover) ist, desto teurer ist er in der Regel. Es gibt also einen Zusammenhang zwischen Seitenzahl x und Buchpreis y . Wir gehen in einen Buchladen und suchen zehn Romane verschiedener Dicken aus. Wir nehmen dabei je ein Buch mit der Seitenzahl 50, 100, 150, ..., 450, 500. Von jedem Buch notieren wir die Seitenzahl und den Buchpreis. Mit diesen Daten erstellen wir Tabelle 2.3.

Aus der Tabelle ist tatsächlich ersichtlich, dass dickere Bücher tendenziell mehr kosten. Wenn wir einen formelmässigen Zusammenhang zwischen Buchpreis und Seitenzahl hätten, könnten wir Vorhersagen über den Preis für Bücher mit Seitenzahlen machen, die wir nicht beobachtet haben. Was würde dann voraussichtlich ein Buch mit 375 Seiten kosten? Oder wir könnten herausfinden, wie teuer ein Buch mit „null“

	Seitenzahl	Buchpreis (SFr)
Buch 1	50	6.4
Buch 2	100	9.5
Buch 3	150	15.6
Buch 4	200	15.1
Buch 5	250	17.8
Buch 6	300	23.4
Buch 7	350	23.4
Buch 8	400	22.5
Buch 9	450	26.1
Buch 10	500	29.1

Table 2.3.: Zusammenhang zwischen Buchpreis und Seitenzahl (fiktiv).

Seiten wäre. Das wären die Grundkosten des Verlags, die unabhängig von der Seitenzahl anfallen: Einband, administrativer Aufwand für jedes Buch, etc. Wie könnten wir diesen Zusammenhang mit einer Formel beschreiben? Das Streudiagramm in Abbildung 2.7 zeigt diesen Zusammenhang graphisch deutlicher auf.

Auf den ersten Blick scheint eine Gerade recht gut zu den Daten zu passen. Diese Gerade hätte die Form

$$y = a + bx$$

mit y dem Buchpreis und x der Seitenzahl sind. Der Parameter a beschreibt dann die Grundkosten des Verlags und der Parameter b entspricht den Kosten pro Seite.

□

Methode der kleinsten Quadrate

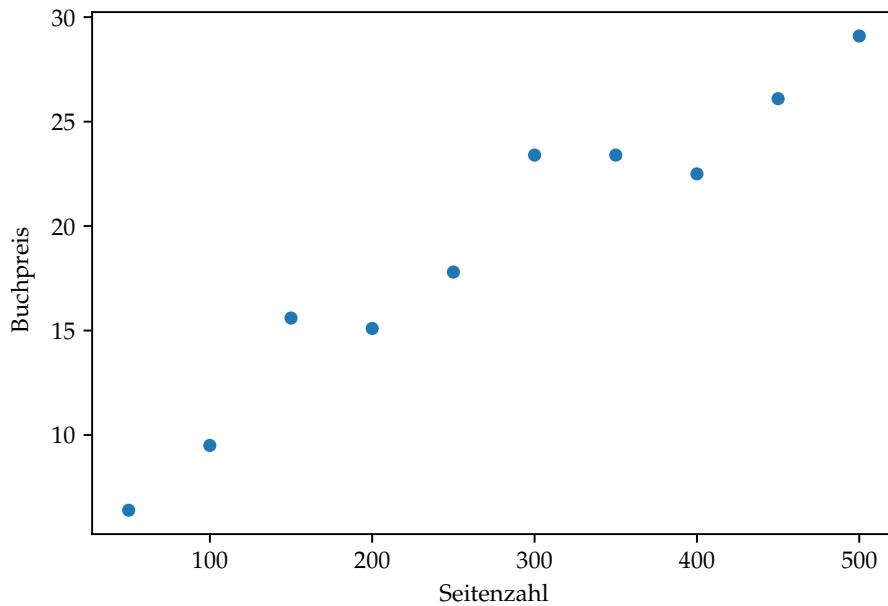
Versuchen wir mit einem Lineal eine Gerade durch *alle* Punkte in Abbildung 2.7 zu legen, so werden wir feststellen, dass das nicht möglich ist (siehe Abbildung 2.8). Die Punkte folgen also nur *ungefähr* einer Geraden.

Da stellt sich uns die Frage: „Wie können wir eine Gerade finden, die *möglichst gut* zu allen Punkten passt?“ Worauf sich bereits die nächste Frage stellt: Was heisst „*möglichst gut*“? Wollen wir ein Mass finden, um zu entscheiden, wie gut die Gerade passt, treten ähnliche Schwierigkeiten auf, wie bei der Bestimmung der Varianz.

Wir könnten die Gerade so wählen, dass wir die vertikalen Differenzen zwischen Beobachtung und Gerade (siehe Abbildung 2.9) zusammenzählen und davon ausgehen, dass eine kleine Summe der Abstände eine gute Anpassung bedeutet.

Kapitel 2. Deskriptive Statistik

(zu R)



```
book = DataFrame({
    "pages" : np.linspace(50,500,10),
    "price" : [6.4, 9.5, 15.6, 15.1, 17.8, 23.4,
               23.4, 22.5, 26.1, 29.1]
})

book.plot(kind="scatter", x="pages", y="price")

plt.xlabel("Seitenzahl")
plt.ylabel("Buchpreis")

plt.show()
```

Abbildung 2.7.: Streudiagramm Seitenzahl - Buchpreis

Wir bezeichnen die vertikale Differenz zwischen einem Beobachtungspunkt (x_i, y_i) und der Geraden (der Punkt auf der Geraden hat die Koordinaten $(x_i, a + bx_i)$) als Residuum:

$$r_i = y_i - (a + bx_i) = y_i - a - bx_i$$

Für unser Beispiel sind die Residuen r_6 und r_8 für diese Gerade in Abbildung 2.9 dargestellt. Das Residuum r_6 ist positiv, da der Punkt oberhalb der Geraden liegt. Entsprechend ist $r_8 < 0$.

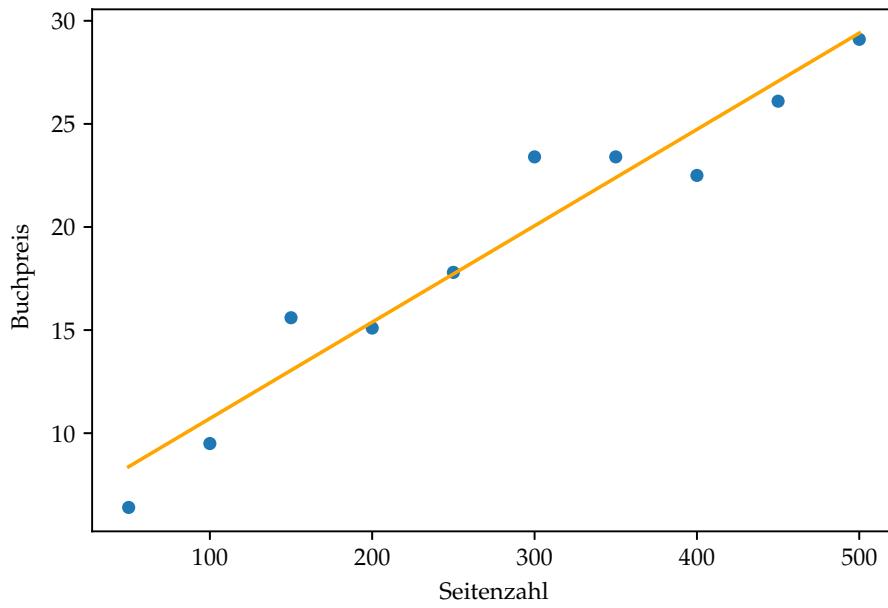


Abbildung 2.8.: Gerade durch das Streudiagramm

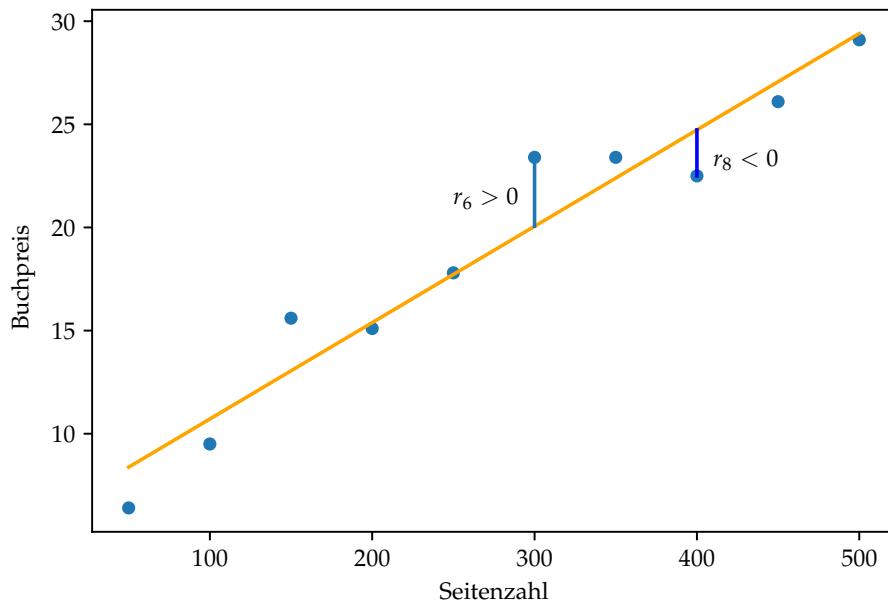


Abbildung 2.9.: Residuen

Wir möchten also die Gerade $y = a + bx$ so bestimmen, dass die Summe

$$r_1 + r_2 + \dots + r_n = \sum_i r_i$$

minimal wird. Diese Methode hat aber eine gravierende Schwäche: Wenn die Hälfte der Punkte weit über der Geraden, die andere Hälfte weit unter der Geraden liegen, so ist die Summe der Abweichungen (Residuen) etwa null. Dabei passt die Gerade gar nicht gut zu den Datenpunkten. Die positiven Abweichungen heben sich nur mit den negativen Abweichungen auf. Wir müssen also das Vorzeichen der Abweichungen eliminieren, bevor wir zusammenzählen. Eine Möglichkeit besteht darin, den Absolutbetrag der Abweichungen aufzusummen, also $\sum_i |r_i|$, und diese Summe zu minimieren. Da es sich aber mit Absolutbeträgen nicht besonders bequem rechnen lässt (zum Beispiel, wenn man Ausdrücke mit Absolutbeträgen ableiten möchte), halten wir nach einer anderen Möglichkeit Ausschau. Die besteht darin, die Quadrate der Abweichungen aufzusummen, also

$$r_1^2 + r_2^2 + \cdots + r_n^2 = \sum_i r_i^2$$

Die Parameter a und b sind so zu wählen, dass diese Summe minimal wird. Letztere Methode hat sich durchgesetzt, weil man mit ihr viel leichter rechnen kann, als mit den Absolutbeträgen. Eine Gerade passt (nach unserem Gütekriterium) also dann am besten zu Punkten, wenn die Summe der Quadrate der vertikalen Abweichungen minimal ist. Dieses Vorgehen ist unter dem Namen Methode der kleinsten Quadrate bekannt.

Beispiel 2.2.4

□

In unserem Fall erhalten wir mit **Python** die Werte $a = 6.04$ und $b = 0.047$. ([zu R](#))

```
b, a = np.polyfit(book["pages"], book["price"], deg=1)
print(a, b)

## 6.039999999999992 0.04672727272727273
```

Die Geradengleichung lautet

$$y = 6.04 + 0.04673x$$

Die Grundkosten des Verlags sind also rund 6 SFr. Pro Seite verlangt der Verlag rund 5 Rappen.

Bemerkungen:

- i. In der Stochastik wird die Geradengleichung in der Form

$$y = a + bx$$

geschrieben, anstatt eher

$$y = ax + b$$

wie sonst in der Mathematik.

- ii. Der Befehl

```
np.polyfit(book["pages"], book["price"], deg=1)
```

aus **numpy** passt ein Polynom vom Grad 1 (lineare Funktion) an die Daten an. Dies Ausgabe sind 2 Werte: der erste ist die Steigung der Geraden, der zweite der y -Achsenabschnitt.

- iii. Diese Gerade wird auch *Regressionsgerade* genannt.

Beispiel 2.2.5

Wie viel würde nach diesem Modell ein Buch von 375 Seiten kosten? Dazu setzen wir $x = 375$ in die Geradengleichung oben ein und erhalten

$$y = 6.04 + 0.04673 \cdot 375 \approx 23.60$$

Das Buch dürfte also etwa CHF. 23.60 kosten. Dieses Modell ist allerdings nur begrenzt gültig. Vor allem bei *Extrapolationen* muss man vorsichtig sein. Wir könnten schon ausrechnen, wie viel ein Buch mit einer Million Seiten kostet, aber dieser Betrag entspricht dann sicher nicht mehr der Realität.

Diese Gerade in Abbildung 2.8 auf Seite 40 wird in **python** wie folgt gezeichnet: ([zu R](#))

```
book.plot(kind="scatter", x="pages", y="price")
b, a = np.polyfit(book["pages"], book["price"], deg=1)

x = np.linspace(book["pages"].min(), book["pages"].max())

plt.plot(x, a+b*x, c="orange")

plt.xlabel("Seitenzahl")
plt.ylabel("Buchpreis")

plt.show()
```

Der Befehl

```
x = np.linspace(book["pages"].min(), book["pages"].max())
```

erzeugt einen Vektor x der Länge 50, der als 1. Wert den Minimalwert von `pages` im Dataframe `book` hat und als letzten Wert dessen Maximalwert.

□

Wie berechnet Python die Parameter a und b ?

Die Parameter a und b werden wie folgt bestimmt:

Die Parameter a und b sollen den folgenden Ausdruck minimieren (Methode der Kleinsten-Quadrat)

$$\sum_{i=1}^n (y_i - (a + bx_i))^2$$

Die Lösung dieses Optimierungsproblems ergibt

$$\hat{b} = \frac{\sum_{i=1}^n (y_i - \bar{y})(x_i - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

$$\hat{a} = \bar{y} - \hat{b} \bar{x}$$

wobei \bar{x} und \bar{y} die entsprechenden Durchschnitte sind. \hat{a} und \hat{b} sind die Schätzer von den Parametern a und b , also die Werte, für welche $\sum_{i=1}^n (y_i - (a + bx_i))^2$ am kleinsten wird.

Bemerkungen:

- i. Wie man auf die Berechnung von a und b kommt, leiten wir hier nicht her. Nur soviel zur Idee: da

$$\sum_i r_i^2 = \sum_{i=1}^n (y_i - (a + bx_i))^2$$

minimal werden muss, muss auch die Ableitung von $\sum_i r_i^2$ nach a und nach b gleich 0 sein. Wir erhalten also ein Gleichungssystem bestehend aus zwei Gleichungen und zwei Unbekannten:

$$\begin{aligned}\frac{\partial}{\partial a} \sum_{i=1}^n (y_i - (a + bx_i))^2 &= \sum_{i=1}^n -2(y_i - a - bx_i) \stackrel{!}{=} 0 \\ \frac{\partial}{\partial b} \sum_{i=1}^n (y_i - (a + bx_i))^2 &= \sum_{i=1}^n -2(y_i - a - bx_i) \cdot x_i \stackrel{!}{=} 0\end{aligned}$$

Die algebraischen Umformungen, die zu den Schätzern von a und b führen, sind dann etwas aufwändig und werden hier nicht aufgeführt.

- ii. Die Berechnungen von \hat{a} und \hat{b} werden wir immer mit **Python** machen.

Beispiel 2.2.6

Es ist zu vermuten, dass es einen Zusammenhang zwischen der Körpergrösse der Väter und der Körpergrösse der Söhne gibt. Der britische Statistiker Karl Pearson trug dazu um 1900 die Körpergrösse von 10 (in Wahrheit waren es 1078) zufällig ausgewählten Männern gegen die Grösse ihrer Väter auf. Dabei erhielt er die Daten von Tabelle 2.4.

Grösse des Vaters	152	157	163	165	168	170	173	178	183	188
Grösse des Sohnes	162	166	168	166	170	170	171	173	178	178

Table 2.4.: Grössenvergleich von Vätern und Söhnen.

Es *scheint* hier tatsächlich einen Zusammenhang zu geben: je grösser der Vater, desto grösser der Sohn. Wenn wir noch das Streudiagramm aufzeichnen (siehe Abbildung 2.10), sehen wir, dass ein (möglicher) linearer Zusammenhang besteht: Die Punktwolke „folgt“ der Geraden $y = 0.445x + 94.7$, wobei wir die Parameter mit der Methode der Kleinsten Quadrate aus den Daten berechnet haben.

Wir können also für die in der Tabelle 2.4 nicht vorkommende Grösse von 180 cm des Vaters den zu erwartenden Wert für die Grösse seines Sohnes berechnen.

$$y = 0.445 \cdot 180 + 94.7 \approx 175 \text{ cm}$$

Wir müssen bei dieser Formel allerdings aufpassen, dass wir sie nicht dort anwenden, wo wir sie gar nicht dürfen. So erhalten wir für $x = 0$ einen Wert von 94.7. Was bedeutet dies aber? Wenn der Vater 0 cm gross ist, so wäre der Sohn gemäss dieses Modells ungefähr 95 cm gross, und das macht keinen Sinn.

□

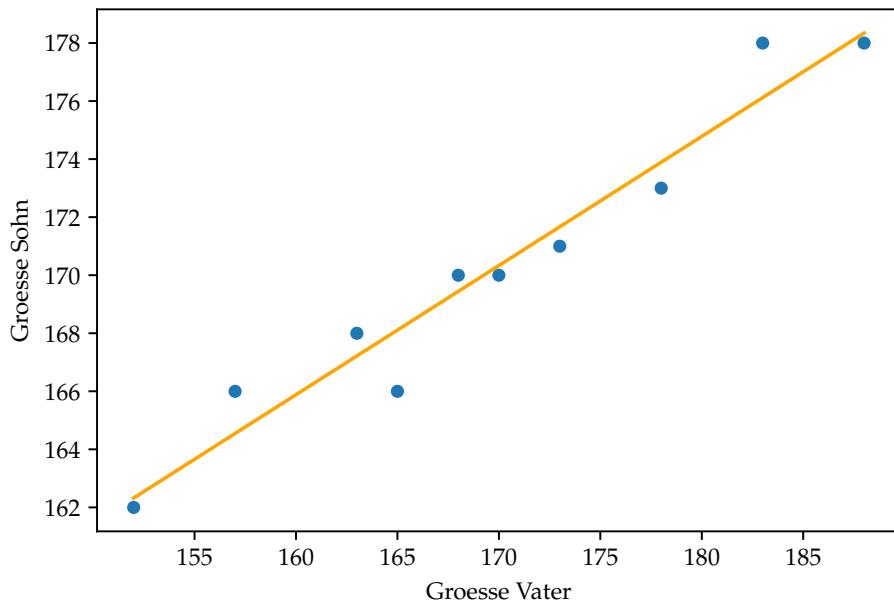


Abbildung 2.10.: Streudiagramm Körpergrößen Väter-Söhne.

Beispiel 2.2.7

Die folgende Tabelle stellt einen Zusammenhang zwischen den Zahlen der Verkehrstoten her, die es 1988 und 1989 in zwölf Bezirken in den USA geben hat (die Bezirke weisen in etwa dieselbe Bevölkerungszahl auf).

Bezirk	1	2	3	4	5	6	7	8	9	10	11	12
Verkehrstote 1988	121	96	85	113	102	118	90	84	107	112	95	101
Verkehrstote 1989	104	91	101	110	117	108	96	102	114	96	88	106

Table 2.5.: Verkehrstote in zwei aufeinanderfolgenden Jahren.

Aus der Tabelle ist kein offensichtlicher Zusammenhang ersichtlich. Betrachten wir das Streudiagramm in Abbildung 2.11, so sehen wir, dass kein Zusammenhang besteht. Dies war aber auch zu erwarten, wenn wir vernünftigerweise davon ausgehen können, dass es zwischen den Verkehrstoten der einzelnen Bezirke keinen Zusammenhang gibt. In Abbildung 2.11 ist die Regressionsgerade eingezeichnet. Diese können wir zwar berechnen und einzeichnen. Allerdings macht diese hier keinen Sinn, da es keinen linearen Zusammenhang zwischen den Messgrößen gibt.

Wir werden im nächsten Abschnitt eine Größe kennenlernen, mit der wir eine Aussage darüber machen können, wie stark der lineare Zusammenhang zwischen zwei

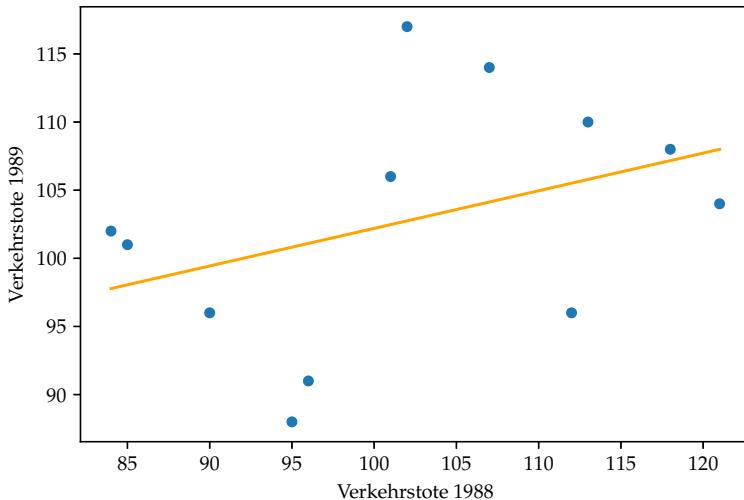


Abbildung 2.11.: Verkehrstote

Messgrößen ist.

□

Beispiel 2.2.8

Als weiteres Beispiel betrachten wir wieder die Erhebung, die den Zusammenhang zwischen Weinkonsum und der Sterblichkeit untersucht. Legen wir den Daten ein lineares Modell zu Grunde

$$y = a + bx$$

wobei x den jährlichen Weinkonsum pro Person und y die Mortalität pro 1000 Personen bezeichnet. Dann können wir aufgrund der Datenpunkte die Parameter a und b mit Hilfe der Methode der Kleinsten Quadrate schätzen und erhalten die Regressionsgerade

$$y = 7.68655 - 0.07608x$$

Betrachten wir allerdings das Streudiagramm mit der Regressionsgeraden (siehe Abbildung 2.12), so stellen wir fest, dass der Zusammenhang zwischen den Messgrößen nicht linear ist. Das Streudiagramm deutet eher auf eine Hyperbelfunktion hin.

Die Regressionsgerade sagt hier also wenig über den wahren Zusammenhang aus.

□

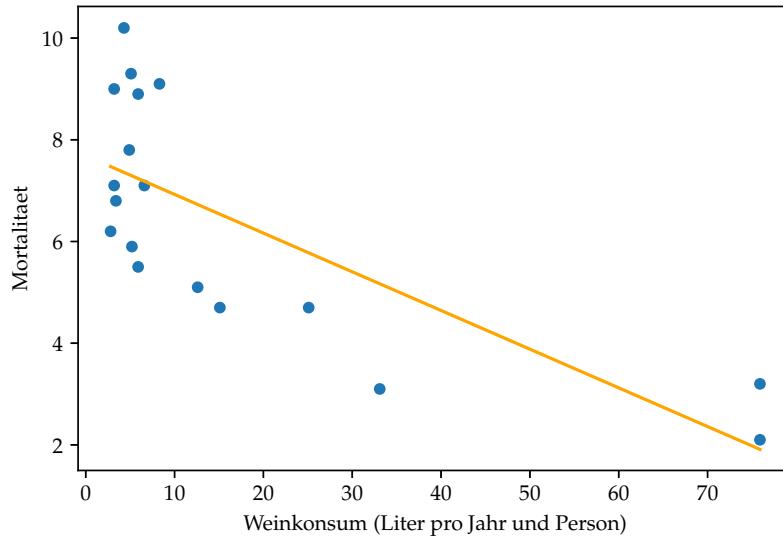


Abbildung 2.12.: Regressionsgerade Weinkonsum-Sterblichkeit

Die Regressionsgerade können wir (fast) immer bestimmen. In den letzten beiden Beispielen haben wir aber gesehen, dass die Regressionsgerade sehr wenig über die wirkliche Verteilung der Punkte im Streudiagramm aussagt. Dafür gibt es zwei Gründe:

- Die Punkte folgen scheinbar gar keiner Gesetzmässigkeit
- Die Punkte folgen einer nichtlinearen Gesetzmässigkeit

Wie können wir nun aber feststellen, ob ein linearer Zusammenhang der Daten besteht oder nicht? Eine Möglichkeit ist sicher, die Situation graphisch zu betrachten, wie wir das eben gemacht haben. Wir können aber auch einen Wert angeben, der den Zusammenhang numerisch beschreibt.

2.2.3. Empirische Korrelation

Für die quantitative Zusammenfassung der linearen Abhängigkeit von zwei Grössen ist die **empirische Korrelation r** als Kennzahl (oder auch mit $\hat{\rho}$ bezeichnet) am gebräuchlichsten.

Empirische Korrelation

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(\sum_{i=1}^n (x_i - \bar{x})^2) \cdot (\sum_{i=1}^n (y_i - \bar{y})^2)}}$$

Die empirische Korrelation ist eine dimensionslose Zahl zwischen -1 und $+1$ und misst Stärke und Richtung der *linearen Abhängigkeit* zwischen den Daten x und y . Die empirische Korrelation hat folgende Eigenschaften

1. Ist $r = +1$, dann liegen die Punkte auf einer steigenden Geraden ($y = a + bx$ mit $a \in \mathbb{R}$ und ein $b > 0$) und umgekehrt.
2. Ist $r = -1$, dann liegen die Punkte auf einer fallenden Geraden ($y = a + bx$ mit $a \in \mathbb{R}$ und ein $b < 0$) und umgekehrt.
3. Sind x und y unabhängig (d.h. es besteht kein Zusammenhang), so ist $r = 0$.

Die Umkehrung gilt im allgemeinen nicht: $r = 0$ heisst *nicht*, dass x und y unabhängig voneinander sind (siehe Abbildung 2.13 auf Seite 50)

Die ersten beiden Eigenschaften lassen sich einfach nachvollziehen: man setzt im Ausdruck für den Korrelationskoeffizienten $y_i = a + bx_i$ und $\bar{y} = a + b\bar{x}$ ein. Dann ergibt sich

$$\begin{aligned} r &= \frac{\sum_{i=1}^n (x_i - \bar{x})(bx_i + a - (b\bar{x} + a))}{\sqrt{(\sum_{i=1}^n (x_i - \bar{x})^2) \cdot (\sum_{i=1}^n (bx_i + a - (b\bar{x} + a))^2)}} \\ &= \frac{b \cdot \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})}{|b| \cdot \sqrt{(\sum_{i=1}^n (x_i - \bar{x})^2) \cdot (\sum_{i=1}^n (x_i - \bar{x})^2)}} \\ &= \text{sign}(b), \end{aligned}$$

wobei $\text{sign}(b)$ das Vorzeichen von b bezeichnet, also $\text{sign}(b) = 1$ falls $b > 0$ und $\text{sign}(b) = -1$ falls $b < 0$.

Man sollte jedoch nie r berechnen, ohne einen Blick auf das Streudiagramm zu werfen, da ganz verschiedene Strukturen den gleichen Wert von r ergeben können. Siehe dazu Abbildung 2.13 auf Seite 50.

Beispiel 2.2.9

Für unser Seitenzahl-Preis-Beispiel erhalten wir mit **Python** (zu **R**)

```
book.corr().iloc[0,1]
## 0.9681121878410434
```

Der Wert liegt also sehr nahe bei 1 und somit besteht ein enger linearer Zusammenhang. Dazu ist der Wert positiv, was einem „je mehr, desto mehr“, also einem positiven linearen Zusammenhang entspricht.

Bemerkungen:

- i. Der Befehl

```
book.corr()
```

ist allgemeiner und liefert die sogenannte Korrelationsmatrix.



Beispiel 2.2.10

Auch im Beispiel der Körpergrösse von Vater und Sohn erwarten wir einen hohen Korrelationskoeffizienten. Wir erhalten 0.973.



Beispiel 2.2.11

Bei den Verkehrsunfällen haben wir keinen Zusammenhang und erwarten einen Korrelationskoeffizienten nahe null. Er beträgt 0.386.



Beispiel 2.2.12

Auch beim Weinkonsum erwarten wir einen negativen Korrelationskoeffizienten, da mit steigendem Weinkonsum die Mortalität sinkt und der nahe bei null liegt. Er beträgt -0.746 . Ohne die Daten in einem Streudiagramm darzustellen, würde man aufgrund dieses Wertes fälschlicherweise auf einen starken negativen linearen Zusammenhang schliessen.



Beispiel 2.2.13

In Abbildung 2.13 sind 21 verschiedene Datensätze dargestellt, die je aus gleich vielen Beobachtungspaaren (x_i, y_i) mit den entsprechenden Punkten im Streudiagramm bestehen. Über jedem Datensatz steht jeweils die zugehörige empirische Korrelation.

Bei perfektem linearen Zusammenhang ist die empirische Korrelation +1 oder -1 (je nachdem ob die Steigung positiv oder negativ; siehe zweite Zeile in Abbildung 2.13). Je mehr die Punkte um den linearen Zusammenhang streuen, desto kleiner wird der Betrag der empirischen Korrelation (siehe erste Zeile).

Da die empirische Korrelation nur den *linearen* Zusammenhang misst, kann es einen (nichtlinearen) Zusammenhang zwischen den beiden Variablen x und y geben, auch wenn die empirische Korrelation null ist (siehe unterste Zeile in Abbildung 2.13).

□

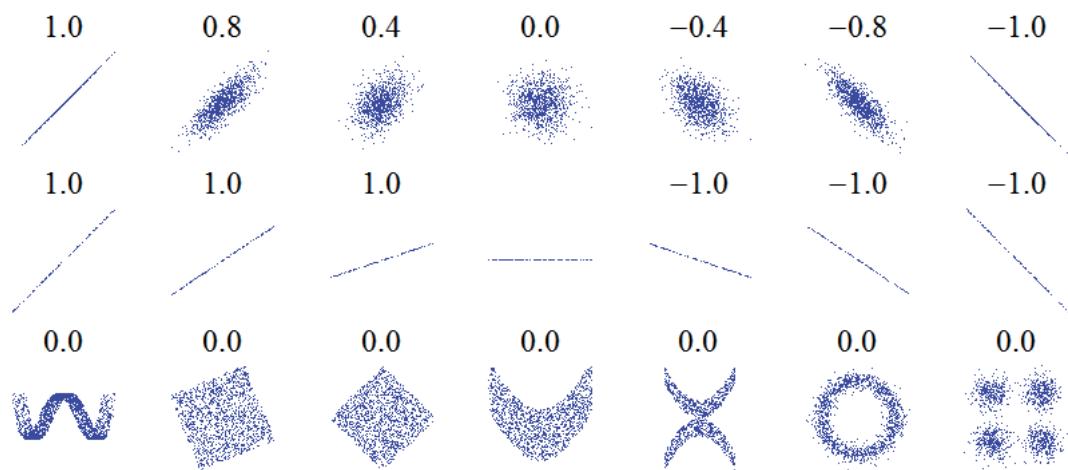


Abbildung 2.13.: 21 verschiedene Datensätze und deren empirische Korrelationskoeffizienten.

2.3. Datenbereinigung

Eines der häufigsten Probleme, mit welchem man in der Datenanalyse konfrontiert wird, ist der Umgang mit fehlenden Werten im Datensatz. Es gibt keine allgemeingültige Regel, wie man mit fehlenden Werten verfahren soll. Bevor man aber fehlende Datenpunkte entfernt oder für fehlende Datenpunkte einfach neue (interpolierte) Datenwerte einsetzt (engl. *data imputation*), sollte man verstehen, warum diese Datenwerte fehlen. Wir unterscheiden folgende Fälle:

1. *Missing Completely at Random (MCAR)*: In diesem Fall ist die Ursache für das Fehlen der Variable völlig unsystematisch. Betrachten wir als Beispiel eine Studie, bei welcher der Grund für die Fettleibigkeit bei K12-Kindern ermittelt wird. MCAR bedeutet in diesem Fall, dass die Eltern zum Beispiel vergessen haben, ihre Kinder in die Klinik zur Studie zu bringen.
2. *Missing at Random (MAR)*: In diesem Fall liegt dem Fehlen von Daten eine gewisse Systematik zugrunde. Bei Verwendung der obengenannten K12-Studie sind die fehlenden Daten in diesem Fall zum Beispiel auf den Umzug der Eltern in eine andere Stadt zurückzuführen, weshalb die Kinder die Studie aufgeben mussten - das Fehlen hat nichts mit der Studie zu tun.
3. *Missing Not at Random (MNAR)* : Ein möglicher nichtzufälliger Grund für das Fehlen von Datenwerten ist, dass der fehlende Wert vom hypothetischen Wert abhängt. Ein Beispiel für MNAR ist, dass die Eltern durch die Art der Studie beunruhigt sind und nicht wollen, dass ihre Kinder beispielsweise gemobbt werden, deswegen zogen sie ihre Kinder aus der Studie zurück. Die Schwierigkeit mit MNAR-Daten ist intrinsisch und hat hier mit dem Problem der Identifizierbarkeit der Studienteilnehmer zu tun. Oder so legen zum Beispiel Menschen mit hohen Gehältern in Umfragen ihr Einkommen nicht offen. Ein anderer nichtzufälliger Grund könnte darin liegen, dass fehlende Werte zum Beispiel vom Wert einer anderen Variablen abhängig sind. Zum Beispiel können wir davon ausgehen, dass Frauen ihr Alter generell nicht offen legen wollen. Hier wird der fehlende Wert in der Altersvariable durch die Geschlechtsvariable beeinflusst.

Im Fall von zufällig fehlenden Datenwerten kann es zulässig sein, die Daten entweder aus dem Datensatz zu entfernen oder mit (interpolierten) Daten zu ersetzen, während im Fall von nicht-zufälligem Fehlen von Datenwerten das Entfernen von Beobachtungen mit fehlenden Werten eine Verzerrung im Modell erzeugen kann. Wir müssen also sehr vorsichtig sein, bevor wir Beobachtungen entfernen oder neue Werte einsetzen. Beachten Sie, dass die Datenimputation nicht unbedingt zu besseren Resultaten führt.

2.3.1. Entfernen von Datenpunkten

Es gibt zwei grundlegende Arten, wie Daten entfernt werden können. Die erste ist bekannt als *listenweise Datenentfernung* (oder auch als vollständige Fallanalyse), die zweite Methode ist das *paarweise Entfernen von Daten*. Durch das listenweise Entfernen von Daten werden alle Beobachtungen mit fehlenden Werten entfernt. Beim paarweisen Entfernen von Daten werden die fehlenden Beobachtungen nur fallweise entfernt.

Listenweise Datenentfernung

Bei der listenweise Datenentfernung werden alle Daten für eine Beobachtung gelöscht, die einen oder mehrere fehlende Werte hat. Insbesondere wenn die fehlenden Daten auf eine kleine Anzahl von Beobachtungen beschränkt sind, können Sie diese Fälle einfach aus der Analyse ausschliessen. In den meisten Fällen ist es jedoch nachteilig, die listenweise Löschung zu verwenden. Dies liegt daran, dass die Annahmen von MCAR (Missing Completely Random) in der Regel selten gestützt werden können. Als Ergebnis erzeugen listenweise Löscherfahren voreingenommene Parameterschätzungen.

Beispiel 2.3.1

Wir erstellen einen Dummy-Datensatz mit einigen **NaN**'s unter Verwendung des Pandas-Dataframes ([zu R](#)) :

```
import pandas as pd
import numpy as np
data = {'Name': ['John', 'Paul', np.nan, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
        'Value': [55, 84, np.nan, 90, 63, 15, 46]}
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                 'Goals', 'Assists', 'Value'])
print(df)
df
```

	Name	Age	Sex	Goals	Assists	Value
## 0	John	21.0	M	5.0	7.0	55.0
## 1	Paul	23.0	NaN	10.0	4.0	84.0
## 2	NaN	NaN	NaN	NaN	NaN	NaN
## 3	Wale	19.0	M	19.0	9.0	90.0
## 4	Mary	25.0	F	5.0	7.0	63.0
## 5	Carli	NaN	F	0.0	6.0	15.0
## 6	Steve	15.0	M	7.0	4.0	46.0

Mit **df.dropna()** können alle **NaN**'s entfernt werden ([zu R](#)) :

```
import pandas as pd
import numpy as np
data = {'Name': ['John', 'Paul', np.nan, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
```

Kapitel 2. Deskriptive Statistik

```
'Value': [55, 84, np.nan, 90, 63, 15, 46] }
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                'Goals', 'Assists', 'Value'])
print(df.dropna())
df.dropna()

##      Name    Age  Sex  Goals  Assists  Value
## 0    John  21.0    M     5.0      7.0   55.0
## 3    Wale  19.0    M    19.0      9.0   90.0
## 4   Mary  25.0    F     5.0      7.0   63.0
## 6  Steve  15.0    M     7.0      4.0   46.0
```

Mit **df.dropna(how = 'all')** werden hingegen nur die Zeilen entfernt, bei denen alle Einträge **NaN**'s aufweisen ([zu R](#)) :

```
import pandas as pd
import numpy as np
data = {'Name': ['John', 'Paul', np.NaN, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
        'Value': [55, 84, np.nan, 90, 63, 15, 46]}
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                'Goals', 'Assists', 'Value'])
print(df.dropna(how = 'all'))
df.dropna(how = 'all')

##      Name    Age  Sex  Goals  Assists  Value
## 0    John  21.0    M     5.0      7.0   55.0
## 1    Paul  23.0  NaN    10.0      4.0   84.0
## 3    Wale  19.0    M    19.0      9.0   90.0
## 4   Mary  25.0    F     5.0      7.0   63.0
## 5   Carli  NaN    F     0.0      6.0   15.0
## 6  Steve  15.0    M     7.0      4.0   46.0
```

Wir können eine Spalte, die nur aus **NaN**'s besteht, mit Hilfe von **df.dropna(axis = 1)** entfernen ([zu R](#)) :

```
import pandas as pd
import numpy as np
data = {'Name': ['John', 'Paul', np.NaN, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
```

Kapitel 2. Deskriptive Statistik

```
'Value': [55, 84, np.nan, 90, 63, 15, 46] }
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                'Goals', 'Assists', 'Value'])
print(df.dropna(axis = 1, how = 'all'))
df.dropna(axis = 1, how = 'all')

##      Name    Age  Sex  Goals  Assists  Value
## 0    John  21.0    M    5.0     7.0   55.0
## 1    Paul  23.0   NaN   10.0     4.0   84.0
## 2     NaN   NaN   NaN     NaN     NaN   NaN
## 3    Wale  19.0    M   19.0     9.0   90.0
## 4   Mary  25.0    F    5.0     7.0   63.0
## 5   Carli   NaN    F    0.0     6.0   15.0
## 6   Steve  15.0    M    7.0     4.0   46.0
```

Wollen wir zum Beispiel eine Spalte bestehend aus **NaN's** generieren, so verwenden wir **df['New'] = np.nan** (zu R) :

```
import pandas as pd
import numpy as np
data = {'Name': ['John', 'Paul', np.NaN, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Assists': [7, 4, np.nan, 9, 7, 6, 4],
        'Value': [55, 84, np.nan, 90, 63, 15, 46]}
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                'Goals', 'Assists', 'Value'])
df['New'] = np.nan
print(df)

##      Name    Age  Sex  Goals  Assists  Value  New
## 0    John  21.0    M    5.0     7.0   55.0  NaN
## 1    Paul  23.0   NaN   10.0     4.0   84.0  NaN
## 2     NaN   NaN   NaN     NaN     NaN   NaN  NaN
## 3    Wale  19.0    M   19.0     9.0   90.0  NaN
## 4   Mary  25.0    F    5.0     7.0   63.0  NaN
## 5   Carli   NaN    F    0.0     6.0   15.0  NaN
## 6   Steve  15.0    M    7.0     4.0   46.0  NaN
```

Wollen wir einen Schwellenwert für die Mindestanzahl der nicht-fehlenden Beobachtungen definieren, so verwenden wir **df.dropna(thresh = x)**:

(zu R)

Kapitel 2. Deskriptive Statistik

```
import pandas as pd
import numpy as np
data = {'Name': ['John', 'Paul', np.nan, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
        'Assists': [7, 4, np.nan, 9, 7, 6, 4],
        df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                         'Goals', 'Assists', 'Value'])
df['New'] = np.nan
print(df.dropna(axis=1, thresh = 2))
df.dropna(axis=1, thresh = 2)

##      Name    Age  Sex  Goals  Assists  Value
## 0    John  21.0    M    5.0     7.0   55.0
## 1    Paul  23.0   NaN   10.0     4.0   84.0
## 2     NaN   NaN   NaN     NaN     NaN   NaN
## 3    Wale  19.0    M   19.0     9.0   90.0
## 4    Mary  25.0    F    5.0     7.0   63.0
## 5   Carli   NaN    F    0.0     6.0   15.0
## 6   Steve  15.0    M    7.0     4.0   46.0
```

In diesem Fall braucht es in jeder Spalte folglich mindestens zwei Werte, die verschieden von **NaN** sind, ansonsten wird die Spalte gelöscht.



Paarweises Entfernen von Datenpunkten

Wir werden hier nicht auf diese Methode eingehen.

Weglassen von Variablen

Falls mehr als 60 % der Beobachtungen fehlen, kann eine Variable auch weggelassen werden, sofern sie nicht wichtig ist. Allerdings ist Datenimputation im Vergleich zum Weglassen von Variablen in der Regel vorzuziehen.

2.3.2. Data Imputation

Imputation aufgrund vom Mittelwert, Median oder Modus

Fehlen bei einer Messgrösse für bestimmte Beobachtungen Werte, so werden diese durch den Mittelwert oder Median der vorhandenen Werte der entsprechenden Messgrösse ersetzt. Dies wird im Falle von numerischen Variablen verwendet. Im Falle von kategorialen Variablen (Variablen für unterschiedliche Gruppen) ersetzt man den Wert durch den häufigsten Wert (Modus). Diese Imputationstechnik funktioniert gut, wenn die Werte völlig zufällig fehlen.

Beispiel 2.3.2

Scikit-learn kommt mit einer imputierten Funktion in der Form von **sklearn.preprocessing** (zu R)

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import Imputer
data = {'Name': ['John', 'Paul', np.nan, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
        'Assists': [7, 4, np.nan, 9, 7, 6, 4],
        'Value': [55, 84, np.nan, 90, 63, 15, 46]}
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex',
                                 'Goals', 'Assists', 'Value'])
imputer = Imputer(missing_values = 'NaN', strategy = 'mean', axis = 0, verbose =
transformed_values = imputer.fit_transform(values)
df_new = pd.DataFrame(transformed_values, columns =['Age', 'Goals', 'Assists', 'Value'])
print(df_new)
df_new

## /usr/local/lib/python3.6/site-packages/sklearn/utils/deprecation.py:58: Depre
##     warnings.warn(msg, category=DeprecationWarning)
##         Age      Goals      Assists      Value
## 0   21.0    5.000000    7.000000   55.000000
## 1   23.0   10.000000   4.000000   84.000000
## 2   20.6    7.666667   6.166667   58.833333
## 3   19.0   19.000000   9.000000   90.000000
## 4   25.0    5.000000   7.000000   63.000000
## 5   20.6    0.000000   6.000000   15.000000
## 6   15.0    7.000000   4.000000   46.000000
```

Kapitel 2. Deskriptive Statistik

strategy bezieht sich auf die Imputationsstrategie, und der Standardwert ist der Mittelwert (**mean**) der Achse (0 für Spalten und 1 für Zeilen). Die anderen Strategien sind **median** und **most_frequent**.

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import Imputer
data = {'Name': ['John', 'Paul', np.nan, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
        'Assists': [7, 4, np.nan, 9, 7, 6, 4],
        'Value': [55, 84, np.nan, 90, 63, 15, 46]}
data = pd.DataFrame(data)
print(data)

## /usr/local/lib/python3.6/site-packages/sklearn/utils/deprecation.py:58: DeprecationWarning: 
##     warnings.warn(msg, category=DeprecationWarning)
##      Age      Goals      Assists      Value
## 0   21.0    5.000000    7.000000   55.000000
## 1   23.0   10.000000   4.000000   84.000000
## 2   20.6    7.666667   6.166667   58.833333
## 3   19.0   19.000000   9.000000   90.000000
## 4   25.0    5.000000   7.000000   63.000000
## 5   20.6    0.000000   6.000000   15.000000
## 6   15.0    7.000000   4.000000   46.000000
```

Eine andere API, die für diese Imputationsstrategie verwendet werden kann, ist

SimpleFill().fit_transform():

```
import pandas as pd
import numpy as np
from pandas import DataFrame
from sklearn.preprocessing import Imputer
from fancyimpute import SimpleFill
data = {'Name': ['John', 'Paul', np.nan, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M']}
```

Kapitel 2. Deskriptive Statistik

```
'Goals': [5,10,np.nan,19,5,0,7],  
'Assists': [7,4,np.nan,9,7,6,4],  
'Value': [55,84,np.nan,90,63,15,46]}  
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex', 'Goals', 'Assists', 'Value'])  
  
df_imputed = DataFrame(SimpleFill().fit_transform(values),  
columns =["Age", "Goals", "Assists", "Value"])  
  
print(df_imputed)  
df_imputed  
  
## Using TensorFlow backend.  
##      Age      Goals   Assists      Value  
## 0  21.0  5.000000  7.000000  55.000000  
## 1  23.0 10.000000  4.000000  84.000000  
## 2  20.6  7.666667  6.166667  58.833333  
## 3  19.0 19.000000  9.000000  90.000000  
## 4  25.0  5.000000  7.000000  63.000000  
## 5  20.6  0.000000  6.000000  15.000000  
## 6  15.0  7.000000  4.000000  46.000000
```

□

Datenimputation mit K-Nearest Neighbour

Mit **KNN (k=x) .fit_transform(data)** werden die fehlenden Werte aufgrund der nächstliegenden Werte (Nachbarn) ersetzt. Die Anzahl k der zu betrachtenden nächsten Nachbarn muss festgelegt werden. ([zu R](#))

Beispiel 2.3.3

```
import pandas as pd  
import numpy as np  
from pandas import DataFrame  
from sklearn.preprocessing import Imputer  
from fancyimpute import KNN  
data = {'Name': ['John', 'Paul', np.NaN, 'Wale', 'Mary', 'Carli', 'Steve'],  
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],  
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
```

Kapitel 2. Deskriptive Statistik

```
'Goals': [5,10,np.nan,19,5,0,7],  
'Assists': [7,4,np.nan,9,7,6,4],  
'Value': [55,84,np.nan,90,63,15,46]}  
df=pd.DataFrame(data, columns =['Name', 'Age', 'Sex', 'Goals', 'Assists', 'Value'])  
  
df_imputed = DataFrame(KNN(k=3).fit_transform(values),  
columns =["Age", "Goals", "Assists", "Value"] )  
  
df_imputed  
print(df_imputed)  
  
## Using TensorFlow backend.  
## Imputing row 1/7 with 0 missing, elapsed time: 0.000  
## [KNN] Warning: 4/28 still missing after imputation, replacing with 0  
##      Age  Goals  Assists  Value  
## 0  21.000000    5.0     7.0   55.0  
## 1  23.000000   10.0     4.0   84.0  
## 2  0.000000    0.0     0.0    0.0  
## 3  19.000000   19.0     9.0   90.0  
## 4  25.000000    5.0     7.0   63.0  
## 5  18.931218    0.0     6.0   15.0  
## 6  15.000000    7.0     4.0   46.0
```

□

Datenimputation mit MICE

Mit `fancyimpute.MICE().complete(data)` werden fehlende Datenwerte durch Mehrfachimputation (*Multiple Imputation by Chained Equations*) ersetzt. MICE ist ein Prozess, bei dem die fehlenden Werte mehrfach ersetzt werden, um „vollständige“ Datensätze zu erstellen. Multiple Imputation hat viele Vorteile gegenüber herkömmlichen Methoden. Multiple Imputation durch verkettete Gleichungen (MICE) ist eine Imputationsmethode, die mit der Annahme arbeitet, dass die fehlenden Daten zufällig (MAR) fehlen. Bedenken Sie, dass für MAR die Art der fehlenden Daten mit den beobachteten Daten, aber nicht mit den fehlenden Daten zusammenhängt. Der MICE-Algorithmus arbeitet mit mehreren Regressionsmodellen, und jeder fehlende Wert wird abhängig von den beobachteten (nicht fehlenden) Werten bedingt modelliert. (zu R)

Beispiel 2.3.4

```
import pandas as pd
import numpy as np
from pandas import DataFrame
from sklearn.preprocessing import Imputer
from fancyimpute import IterativeImputer
data = {'Name': ['John', 'Paul', np.NaN, 'Wale', 'Mary', 'Carli', 'Steve'],
        'Age': [21, 23, np.nan, 19, 25, np.nan, 15],
        'Sex': ['M', np.nan, np.nan, 'M', 'F', 'F', 'M'],
        'Goals': [5, 10, np.nan, 19, 5, 0, 7],
        'Assists': [7, 4, np.nan, 9, 7, 6, 4],
        'Value': [55, 84, np.nan, 90, 63, 15, 46]}
df=pd.DataFrame(data, columns=['Name', 'Age', 'Sex', 'Goals', 'Assists', 'Value'])

df_imputed = DataFrame(IterativeImputer().fit_transform(values),
                       columns=["Age", "Goals", "Assists", "Value"])

df_imputed
print(df_imputed)

## Using TensorFlow backend.

##          Age      Goals    Assists      Value
## 0  21.000000  5.000000  7.000000  55.000000
## 1  23.000000 10.000000  4.000000  84.000000
## 2 19.541591  7.666667  6.166667  58.833333
## 3 19.000000 19.000000  9.000000  90.000000
## 4 25.000000  5.000000  7.000000  63.000000
## 5 14.249546  0.000000  6.000000  15.000000
## 6 15.000000  7.000000  4.000000  46.000000
```



Konzeptionelle Lernziele

Sie sind fähig, ...

- die wichtigsten Methoden der deskriptiven Statistik zu erklären und zu interpretieren.
- folgende Größen auszurechnen: arithmetisches Mittel, Standardabweichung, Varianz, Quantil, Median und Korrelationskoeffizient.
- die Grundidee der einfachen linearen Regression zu erklären, insbesondere das Regressionsmodell zu definieren, die Koeffizienten zu interpretieren und zu erklären, wie diese geschätzt werden.
- die Konstruktion von Graphiken zu erklären und diese zu interpretieren, insbesondere: Histogramm, Boxplot, empirische kumulativen Verteilungsfunktion und Streudiagramm.
- unterschiedliche Szenarien zu erklären, warum Datenpunkte in einem Datensatz fehlen.

Computer-Basierte Lernziele

Sie sollten fähig sein, ...

- Daten mit Hilfe von **pandas** als **Series** oder **DataFrame** einzulesen.
- den empirischen Mittelwert und die empirische Standardabweichung mit den **pandas**-Methoden **var()**, resp. **std()** zu berechnen.
- den empirischen Median und die Quantile mit den **pandas**-Methoden **median()** und **quantile** zu berechnen und daraus die Quartildifferenz zu bestimmen.
- ein Histogramm mit Hilfe der **pandas**-Methode **plot(kind="hist")** zu erstellen.
- ein Boxplot für einen Datensatz mit Hilfe der **pandas**-Methode **plot(kind="box")** zu erstellen.
- ein Streudiagramm für einen zweidimensionalen Datensatz mit Hilfe der **pandas**-Methode **plot(kind="scatter")** zu erstellen.
- die Koeffizienten eines einfachen linearen Regressionsmodells mit Hilfe der **numpy**-Methode **np.polyfit()** zu berechnen.
- den Korrelationskoeffizienten für zwei Größen mit Hilfe der **pandas**-Methode **corr()** zu berechnen.
- Zeilen oder Spalten in einer Datenmatrix mit Hilfe von **dropna()** zu entfernen.

Kapitel 2. Deskriptive Statistik

- fehlende Datenpunkte mit Hilfe von `SimpleFill().fit_transform()`,
`KNN(k=3).fit_transform()` und `IterativeImputer().fit_transform()` zu ersetzen (imputieren).

Kapitel 3.

Modelle für Messdaten

Everybody believes in the exponential law of errors [i.e., the Normal distribution]: the experimenters, because they think it can be proved by mathematics; and the mathematicians, because they believe it has been established by observation.

(E. T. Whittaker and G. Robinson)

3.1. Stetige Zufallsvariablen und Wahrscheinlichkeitsverteilungen

In vielen Anwendungen hat man es nicht mit Zähl-, sondern mit *Messdaten* (Messungen) zu tun. Diese können grundsätzlich jeden Wert in einem bestimmten Bereich annehmen, wobei die Genauigkeitsangabe des Messwertes durch die Messgenauigkeit vorgegeben wird. Wir wollen zuerst an einem Beispiel mit diskreten Messdaten den wichtigen Begriff der Zufallsvariable nochmals in Erinnerung rufen.

3.1.1. Diskrete Wahrscheinlichkeitsverteilung

Eine Zufallsvariable X ordnet jedem Zufallsexperiment *genau* eine Zahl zu. Wir können somit X auch als *Funktion* auffassen.

Beispiel 3.1.1

Die Zufallsvariable X ordnet einer zufällig ausgewählten, in der Schweiz lebenden Person, die Körpergrösse in cm zu. Die Körpergrösse wird also auf Zentimeter gerundet. Die *Definitionsmenge* dieser Zufallsvariable X ist dann die Menge der in der Schweiz lebenden Personen.

Die Zufallsvariable X kann nur folgende Werte annehmen (*Wertemenge*)

$$W_X = \{20, 21, 22, \dots, 250\}$$

Der Wertebereich wurde absichtlich zu gross gewählt, damit auch sicher alle vorkommenden Werte dabei sind. Die Wertemenge besteht also nur aus endlich vielen ganzen Zahlen. Eine solche Menge heisst *diskret*. Wichtig ist hier, dass wir *keinen* Wert zwischen zwei Werten der Wertemenge auswählen können, also z.B. 175.25 cm. Die Menge ist „löchrig“. Dies kann man (sehr vereinfacht) als Definition von „diskret“ auffassen.

Wir wählen nun zufällig (deshalb Zufallsvariable) eine Person aus, die *Tabea* heisst. Wir nehmen an, dass jeder Name nur genau einmal vorkommen kann, was natürlich nicht der Fall ist. Aber wir hätten auch die AHV-Nummer wählen können, die eindeutig ist. Tabea hat eine Körpergrösse von 166 cm (auf cm gerundet). Mit der Zufallsvariable X (Funktion) können wir dies wie folgt formulieren

$$X(\text{Tabea}) = 166$$

Nun wählen wir zufällig eine weitere Person aus, die den Namen *Tadeo* und eine Körpergrösse von 176 cm hat. Wir schreiben dann

$$X(\text{Tadeo}) = 176$$

Dies können wir mit jeder in der Schweiz lebenden Person machen.

Mit dem Ausdruck

$$X = 174$$

beschreiben wir das *Ereignis*, eine Person ausgesucht zu haben, die eine gerundete Körpergrösse von 174 cm hat. Wir sprechen hier auch von einer *Realisierung*

$$x = 174$$

von X .

Beachten Sie Gross- und Kleinschreibung: $x = 174$ ist eine *Zahl* und $X = 174$ ist eine *Menge* (der Personen mit gerundeter Körpergrösse von 174 cm).

Diesem Ereignis können wir nun eine Wahrscheinlichkeit

$$P(X = 174)$$

Kapitel 3. Modelle für Messdaten

zuordnen. Diese berechnet sich hier (siehe auch Bemerkung unten), indem wir die Anzahl Personen mit gerundeter Körpergrösse von 174 cm durch die Anzahl der in der Schweiz lebenden Personen dividieren. Auf diese Weise können wir *alle* Wahrscheinlichkeiten

$$P(X = x)$$

berechnen, wobei x jeden Wert in der Wertemenge annehmen kann. Insbesondere ist

$$P(X = 250) = 0$$

da es keine Person mit so einer Körpergrösse gibt (oder zumindest ist es sehr unwahrscheinlich). Aus diesem Grund spielt es auch keine Rolle, wenn wir die Wertemenge zu gross wählen.

Wir können weitere Wahrscheinlichkeiten bestimmen. So ist

$$P(X \leq 170)$$

die Wahrscheinlichkeit, dass eine zufällig ausgewählte Person eine gerundete Körpergrösse von 170 cm *oder weniger* hat. Beachten Sie, dass dies *nicht* der Wahrscheinlichkeit

$$P(X < 170)$$

entspricht. Dies ist die Wahrscheinlichkeit, dass eine zufällig ausgewählte Person eine gerundete Körpergrösse *kleiner als* 170 cm hat. Die Körpergrösse 170 cm gehört hier *nicht* dazu.

Berechnen wir die Wahrscheinlichkeiten für alle x in der Wertemenge, so erhalten wir eine *Wahrscheinlichkeitsverteilung*. Es gilt insbesondere

$$P(X = 20) + P(X = 21) + \dots + P(X = 249) + P(X = 250) = 1$$

Jede Person muss eine Körpergrösse haben.

□

Bemerkungen:

- i. Wichtig ist die Unterscheidung zwischen X und x . Die Variable X ist eine Funktion, x ist ein konkreter Wert (Realisierung) von X .
- ii. Berechnen wir konkret die Wahrscheinlichkeit aus dem Beispiel vorher, dass eine zufällig ausgewählte Person die Körpergrösse 165 cm hat (Anzahl Personen mit dieser Körpergrösse dividiert durch die Anzahl aller Personen), so ändert sich diese leicht von Tag zu Tag, da Personen sterben und andere geboren werden. Falls eine Zufallsvariable eine zeitliche Abhängigkeit aufweist, also $X(t)$, so haben wir es mit einem *stochastischen Prozess* zu tun.

Ändert sich die Wahrscheinlichkeitsverteilung dieser Zufallsvariablen zeitlich, so spricht man von einem *nicht-stationären* Prozess. In diesem Kapitel gehen wir aber davon aus, dass sich die Wahrscheinlichkeitsverteilung zeitlich nicht ändert, d.h., die Wahrscheinlichkeit $P(X = 165)$ ist heute dieselbe wie vor einem Jahr.

In diesem Kapitel beschäftigen wir uns mit Messgrößen, die man beliebig genau messen kann und die *nicht* gerundet werden.

3.1.2. Stetige Verteilungen

Der Wertebereich W_X einer Zufallsvariablen X ist die Menge aller Werte, die X annehmen kann. Eine Zufallsvariable X heisst *stetig*, wenn deren Wertebereich W_X kontinuierlich ist. Beispiele von kontinuierlichen Wertebereichen sind

$$W_X = \mathbb{R}, \mathbb{R}^+ \quad \text{oder} \quad [0, 1]$$

Konvention bei der Klammernotation:

Bei einer runden Klammer liegt der Wert ausserhalb des Intervalls, bei einer eckigen Klammer liegt der Wert innerhalb des Intervalls. Das Intervall $(a, b]$ beschreibt also alle Punkte x mit $x > a$ und $x \leq b$.

Die Wahrscheinlichkeitsverteilung einer diskreten Zufallsvariablen (z.B. Binomialverteilung oder Beispiel 3.1.1) kann beschrieben werden, indem man die Punktswahrscheinlichkeiten $P(X = x)$ für alle möglichen Werte x im Wertebereich angibt.

Für eine stetige Zufallsvariable X gilt jedoch für alle $x \in W_X$:

$$P(X = x) = 0$$

Dies impliziert, dass wir die Wahrscheinlichkeitsverteilung von X *nicht* mittels der Angaben von Punktswahrscheinlichkeiten beschreiben können.

Beispiel 3.1.2

Diesen Sachverhalt kann man intuitiv einfach an einem Beispiel verstehen. Wir betrachten die Zahlen zwischen 0 und 10 (exklusive 10).

Angenommen, wir haben eine Zufallsvariable X_0 , die jeden Wert aus der diskreten Menge

$$W_0 = \{0, 1, 2, \dots, 8, 9\}$$

mit gleicher Wahrscheinlichkeit annimmt. Die Wahrscheinlichkeit, dass die Zufallsvariable X_0 einen konkreten Wert x (z. B. $x = 3$) aus dem Bereich W_0 annimmt, ist also

$$P(X_0 = x) = \frac{1}{10}$$

Kapitel 3. Modelle für Messdaten

weil W_0 aus zehn Elementen besteht. Jetzt vergrössern wir die diskrete Menge, indem wir jede Zahl auf eine Nachkommastelle genau angeben:

$$W_1 = \{0.0, 0.1, 0.2, \dots, 9.8, 9.9\}$$

Die Zufallsvariable X_1 nimmt jeden Wert aus W_1 mit gleicher Wahrscheinlichkeit an, also ist

$$P(X_1 = x) = \frac{1}{100}$$

weil W_1 aus hundert Elementen besteht. Zum Beispiel haben wir also

$$P(X_1 = 3.2) = \frac{1}{100}$$

Wenn man noch eine Nachkommastelle hinzufügt, erhält man eine Menge aus tausend Elementen und die Wahrscheinlichkeit, ein bestimmtes Element zufällig zu ziehen (z. B. die Zahl 3.21), beträgt nur noch $1/1000$. Wenn wir diese Entwicklung fortsetzen, kommen wir zu folgender Regel: Wenn jede der Zahlen zwischen 0 und kleiner als 10 mit i Nachkommastellen mit gleicher Wahrscheinlichkeit gezogen wird, beträgt die Wahrscheinlichkeit, eine bestimmte Zahl aus dieser Menge zu ziehen,

$$\frac{1}{10^{i+1}}$$

Je mehr Nachkommastellen wir berücksichtigen, umso grösser wird der Nenner und umso kleiner wird dann die Wahrscheinlichkeit eine bestimmte Zahl zu ziehen. Wenn wir unendlich viele Nachkommastellen zulassen, wird aus der diskreten Menge die kontinuierliche Menge

$$W_\infty = [0, 10)$$

Entsprechend beträgt die Wahrscheinlichkeit, ein bestimmtes Element mit unendlich vielen Nachkommastellen aus dieser Menge zu ziehen,

$$P(X_\infty = x) = \frac{1}{\infty} = 0$$

□

Beispiel 3.1.3

Wir messen wieder die Körpergrösse von Personen, nur gehen wir von nun an davon aus, dass wir die Körpergrösse beliebig genau messen können. Die Zufallsvariable X ordnet wieder jeder Person die zugehörige Körpergrösse zu.

Die Wahrscheinlichkeit, eine Körpergrösse von genau 182.254 680 895 434... cm zu messen, ist gleich 0:

$$P(X = 182.254 680 895 434\dots) = 0$$

wobei X die Körpergrösse misst. Was für eine Wahrscheinlichkeit können wir aber im Zusammenhang von Körpergrössen angeben?

Nun wir könnten die Wahrscheinlichkeit angeben, dass ein Messwert in einem bestimmten Bereich liegt, wie z.B. zwischen 174 und 175 cm:

$$P(174 < X \leq 175)$$

Diese Wahrscheinlichkeit ist dann *nicht* mehr 0. Um diese Wahrscheinlichkeit zu berechnen, können wir allerdings nicht einfach die Punktswahrscheinlichkeiten aufaddieren, da diese 0 ergäbe.

□

Wir brauchen also ein neues Konzept, und zwar die sogenannte *Wahrscheinlichkeitsdichte*.

3.1.3. Wahrscheinlichkeitsdichte

Haben wir einen Datensatz mit experimentellen Messdaten, so werden wir feststellen, dass die relative Häufigkeit von Messpunkten in bestimmten Intervallen grösser ist als in anderen.

Die *Wahrscheinlichkeitsverteilung einer stetigen Zufallsvariablen X* kann also beschrieben werden, indem man die Wahrscheinlichkeiten für alle Intervalle $(a, b]$ mit $a < b$ angibt:

$$P(X \in (a, b]) = P(a < X \leq b)$$

Dazu genügt es, die *kumulative Verteilungsfunktion*

$$F(x) = P(X \leq x)$$

anzugeben, denn es gilt

$$P(a < X \leq b) = F(b) - F(a)$$

Bemerkungen:

- i. Weil für stetige Zufallsvariablen

$$P(X = a) = P(X = b) = 0$$

gilt, spielt es keine Rolle, ob wir $<$ oder \leq schreiben. Also gilt beispielsweise:

$$P(a < X \leq b) = P(a \leq X \leq b)$$

Bei diskreten Zufallsvariablen ist diese Unterscheidung allerdings wichtig.

Beispiel 3.1.4

Wir messen wieder die Körpergrösse mit der zugehörigen Zufallsvariablen X . Dann können wir die Wahrscheinlichkeit angeben, dass eine Person zwischen 165 cm und 175 cm gross ist:

$$P(165 < X \leq 175)$$

Diese Wahrscheinlichkeit können aber auch wie folgt schreiben:

$$P(165 < X \leq 175) = P(X \leq 175) - P(X \leq 165)$$

also die Wahrscheinlichkeit, dass eine zufällig ausgewählte Person höchstens 175 cm gross ist minus die Wahrscheinlichkeit, dass eine zufällig ausgewählte Person höchstens 165 cm gross ist. Da bleiben nur noch die Personen zwischen 165 cm und 175 cm übrig.

Wir wollen anhand dieses Beispiels noch einige Eigenschaften der kumulativen Verteilungsfunktion betrachten. Die kumulative Verteilungsfunktion

$$F(x) = P(X \leq x)$$

beschreibt in diesem Fall die Wahrscheinlichkeit, dass eine zufällig ausgewählte Person kleiner oder gleich x Zentimeter ist. Dann ist

$$F(175)$$

die Wahrscheinlichkeit, dass eine zufällig ausgewählte Person nicht grösser als 175 cm ist.

Die Wahrscheinlichkeit oben lässt sich dann schreiben als

$$P(165 < X \leq 175) = F(175) - F(165)$$

Hier ist auch offensichtlich, dass

$$F(165) \leq F(175)$$

Analog gilt für alle Körpergrössen $a < b$

$$F(a) \leq F(b)$$

Die kumulative Verteilungsfunktion ist also monoton steigend.

Die Wahrscheinlichkeit

$$F(-\infty) = P(X < -\infty)$$

dass eine zufällig auswählte Person eine Körpergrösse kleiner als $-\infty$ hat, ist natürlich 0 (*niemand* kann kleiner als $-\infty$ sein).

Entsprechend ist die Wahrscheinlichkeit

$$F(\infty) = P(X < \infty)$$

dass eine zufällig ausgewählte Person eine Körpergrösse kleiner als ∞ hat, ist 1 (*alle* Personen haben eine Körpergrösse kleiner als ∞). □

Allgemein

Die kumulative Verteilungsfunktion hat folgende wichtige Eigenschaften (siehe Abbildung 3.1):

1. Da es sich bei

$$F(x) = P(X \leq x)$$

um eine Wahrscheinlichkeit handelt, gilt

$$0 \leq F(x) \leq 1$$

2. Die Wahrscheinlichkeit

$$P(X < -\infty)$$

dass ein Messwert kleiner als $-\infty$ ist, ist offensichtlich 0. Und damit ist auch

$$F(-\infty) = 0$$

3. Die Wahrscheinlichkeit

$$P(X \leq \infty)$$

dass ein Messwert kleiner als ∞ ist, ist offensichtlich 1:

$$F(\infty) = 1$$

4. Die Funktion von $F(x)$ ist monoton wachsend. Es gilt also für $a < b$:

$$F(a) \leq F(b)$$

Die Ableitung $F'(x)$ von $F(x)$ ist also immer grösser gleich 0.

Kapitel 3. Modelle für Messdaten

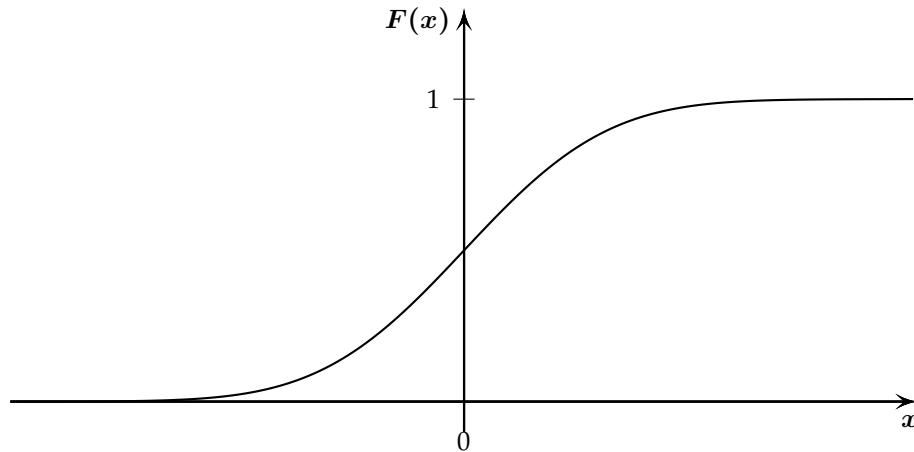


Abbildung 3.1.: Beispiel für eine kumulative Verteilungsfunktion

Zusammenfassend heisst dies, dass die Wahrscheinlichkeitsverteilung einer stetigen Zufallsvariablen X durch die kumulative Verteilungsfunktion beschrieben werden kann.

Für *stetige* Zufallsvariablen können wir einen zur Punktwahrscheinlichkeit $P(X = x)$ analogen Begriff mit Hilfe der Ableitung der kumulativen Verteilungsfunktion gewinnen. Was nun folgt, erinnert an die Bildung eines bestimmten Integrales. Das Schlussresultat ist dann auch ein bestimmtes Integral.

Wir wollen die Wahrscheinlichkeit

$$P(a < X \leq b)$$

als Summe von Wahrscheinlichkeiten schreiben.

Das Intervall $(a, b]$ unterteilen wir dazu in n -Teilintervalle mit Teilpunkten

$$a = x_0, x_1, x_2, \dots, b = x_n$$

Die einzelnen Teilintervalle haben die gleiche Länge Δx :

$$\Delta x = \frac{b - a}{n}$$

Dann können wir die Wahrscheinlichkeit $P(a < X \leq b)$ in der Form

$$\begin{aligned} P(a < X \leq b) &= P(x_0 < X \leq x_1) + P(x_1 < X \leq x_2) + \dots + P(x_{n-1} < X \leq x_n) \\ &= P(x_0 < X \leq x_0 + \Delta x) + \dots + P(x_{n-1} < X \leq x_{n-1} + \Delta x) \\ &= \sum_{i=0}^{n-1} P(x_i < X \leq x_i + \Delta x) \end{aligned}$$

Kapitel 3. Modelle für Messdaten

schreiben. Für die Summanden in der letzten Summe verwenden wir nun die kumulative Verteilungsfunktion F :

$$P(a < X \leq b) = \sum_{i=0}^{n-1} P(x_i < X \leq x_i + \Delta x) = \sum_{i=0}^{n-1} (F(x_i + \Delta x) - F(x_i))$$

Wir multiplizieren die Summanden nun mit $\frac{\Delta x}{\Delta x}$:

$$P(a < X \leq b) = \sum_{i=0}^{n-1} (F(x_i + \Delta x) - F(x_i)) = \sum_{i=0}^{n-1} \frac{F(x_i + \Delta x) - F(x_i)}{\Delta x} \Delta x$$

Aus der Definition der Ableitung folgt für kleine Δx :

$$F'(x_i) \approx \frac{F(x_i + \Delta x) - F(x_i)}{\Delta x}$$

Setzen wir dies in unsere Summe ein, so erhalten wir

$$P(a < X \leq b) \approx \sum_{i=0}^{n-1} F'(x_i) \Delta x$$

Diese Summe ist aber eine Riemann-Summe, die für $\Delta x \rightarrow 0$ gegen ein bestimmtes Integral strebt. Es gilt also

$$P(a < X \leq b) = \int_a^b F'(x) \, dx$$

Diese Funktion $F'(x)$ nennen wir

Wahrscheinlichkeitsdichte

Die *Wahrscheinlichkeitsdichte* f ist definiert als Ableitung der kumulativen Verteilungsfunktion:

$$f(x) = F'(x)$$

Damit erhalten wir folgende Interpretation: die Wahrscheinlichkeit, dass die Zufallsvariable X einen Wert in $(x, x + \Delta x]$ annimmt, lautet

$$P(x < X \leq x + \Delta x) \approx f(x) \Delta x$$

falls Δx klein ist. Die Begründung dafür ist:

$$\frac{P(x < X \leq x + \Delta x)}{\Delta x} = \frac{F(x + \Delta x) - F(x)}{\Delta x} \approx f(x)$$

Kapitel 3. Modelle für Messdaten

wobei die letzte Approximation aus der Definition einer Ableitung folgt. In differenzierbarer Schreibweise können wir obige Beziehung ausdrücken als

$$P(x < X \leq x + dx) = f(x) dx$$

Aus der Dichte kann man die kumulative Verteilungsfunktion zurückgewinnen:

$$F(x) = \int_{-\infty}^x f(s) ds$$

weil F eine Stammfunktion von f ist und $F(-\infty) = 0$.

Eigenschaften Wahrscheinlichkeitsdichte

Für eine Wahrscheinlichkeitsdichte $f(x)$ gelten folgende Eigenschaften (siehe Abbildung 3.2):

1. Es gilt

$$f(x) \geq 0$$

für alle x , da $F(x)$ monoton wachsend ist und damit deren Ableitung grösser gleich 0 sein muss.

2. Es gilt

$$P(a < X \leq b) = F(b) - F(a) = \int_a^b f(x) dx$$

Dies entspricht der Fläche zwischen a und b unter $f(x)$ (siehe Abbildung 3.2).

3. Es gilt

$$\int_{-\infty}^{\infty} f(x) dx = 1$$

Dies ist die Wahrscheinlichkeit, dass *irgendein* Wert gemessen wird.

Bemerkungen:

- i. Wir haben noch keine konkreten kumulativen Verteilungsfunktionen und Dichtefunktionen kennengelernt. Dies werden wir bald nachholen.

Wichtig ist der Zusammenhang zwischen Wahrscheinlichkeit und Flächen:

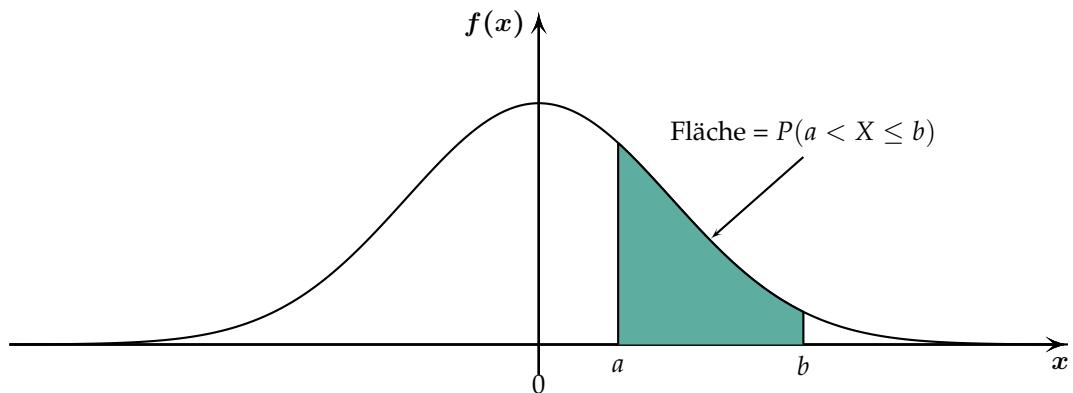


Abbildung 3.2.: Illustration einer Dichte einer Zufallsvariablen und der Wahrscheinlichkeit, in das Intervall $(a, b]$ zu fallen (grüne Fläche).

Merkregel

Für stetige Wahrscheinlichkeitsverteilungen entsprechen Wahrscheinlichkeiten Flächen unter der Dichtefunktion.

Die kumulative Verteilungsfunktion können wir uns auch wie in Abbildung 3.3 vorstellen. Der Wert α der Funktion $F(x)$ an der Stelle von beispielsweise 0.5 (Abbildung 3.3 links) entspricht gerade der Fläche unter der Dichtekurve von $-\infty$ bis 0.5 (Abbildung 3.3 rechts).

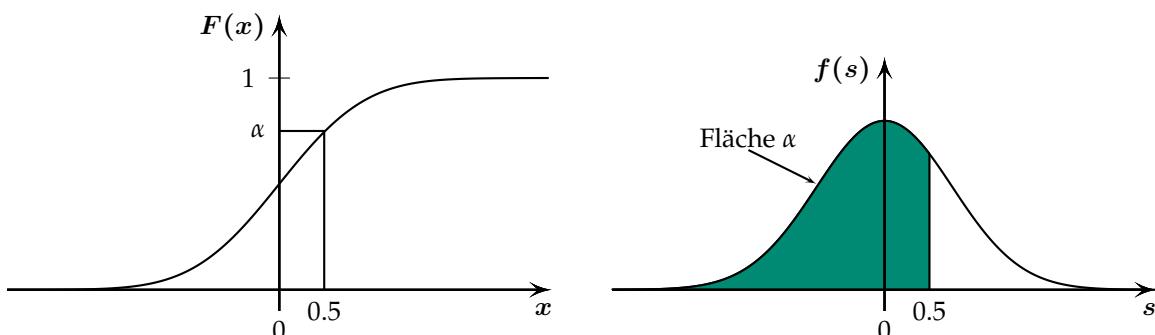


Abbildung 3.3.: Illustration der kumulativen Verteilungsfunktion $F(x)$ für $x = 0.5$

Die Dichtefunktion in Abbildung 3.2 sieht sehr „regelmässig“ aus. Dies muss nicht sein. Solange die Funktionswerte nicht negativ sind und die Fläche zwischen der Kurve und der x -Achse 1 ist, gibt es (fast) keine Einschränkungen in Bezug auf die Form einer Dichtefunktion, siehe Abbildung 3.4.

Allerdings werden wir sehen, dass die Dichtefunktion in Abbildung 3.2 eine zentrale Rolle spielen wird.

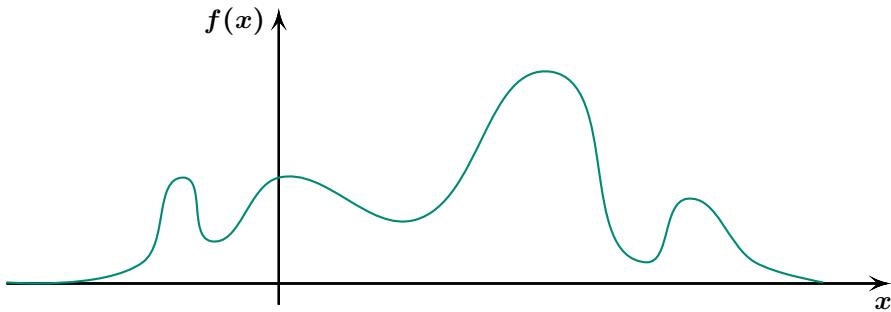


Abbildung 3.4.: Dichtefunktion mit einer „unregelmässigen“ Form.

3.1.4. Kennzahlen von stetigen Verteilungen

Der *Erwartungswert* $E(X)$ und die *Standardabweichung* σ_X einer stetigen Zufallsvariablen X haben dieselbe Bedeutung wie im diskreten Fall: Der Erwartungswert beschreibt die mittlere Lage der Verteilung und die Standardabweichung deren Streuung. Wir erinnern uns, dass der Erwartungswert einer diskreten Zufallsvariablen definiert ist als

$$E(X) = \sum_i x_i \cdot P(X = x_i)$$

Die Formeln für den *Erwartungswert* und die *Varianz* einer stetigen Zufallsvariablen ergeben sich, indem wir beim diskreten Fall $P(X = x)$ durch $f(x) dx$ und die Summe durch ein Integral ersetzen:

Erwartungswert und Varianz

Erwartungswert und *Varianz* sind wie folgt definiert:

$$E(X) = \mu_X = \int_{-\infty}^{\infty} xf(x) dx$$

$$\text{Var}(X) = \sigma_X^2 = E((X - E(X))^2) = \int_{-\infty}^{\infty} (x - E(X))^2 f(x) dx$$

Durch einige Umformungen der Definition der Varianz erhalten wir folgende wichtige Beziehung, die für die Berechnung der Varianz oft einfacher ist als die Berechnung durch die Definition.

$$\text{Var}(X) = E(X^2) - E(X)^2$$

Auch die Quantile sind analog definiert, wie im diskreten Fall:

Quantile

Die Quantile $q(\alpha)$ für $0 < \alpha < 1$ einer Zufallsvariablen X , bzw. deren Verteilung, sind wie folgt definiert:

$$P(X \leq q(\alpha)) = \alpha$$

Das heisst:

$$F(q(\alpha)) = \alpha \Leftrightarrow q(\alpha) = F^{-1}(\alpha)$$

Dies bedeutet nichts anderes, dass die Quantile die Umkehrung der kumulativen Verteilungsfunktion sind.

Der Input der kumulativen Verteilungsfunktion ist eine Zahl (Messwert) und der Funktionswert ist eine Wahrscheinlichkeit (siehe Abbildung 3.3). Der Input der Quantile ist eine Wahrscheinlichkeit und der Funktionswert ist eine Zahl (Messwert). Siehe auch Abbildung 3.5. Dies kann auch so interpretiert werden, dass $q(\alpha)$ der Punkt ist, bei welchem die Fläche von $-\infty$ bis $q(\alpha)$ unter der Dichte f gleich α ist.

Das 50 %-Quantil heisst der *Median*.

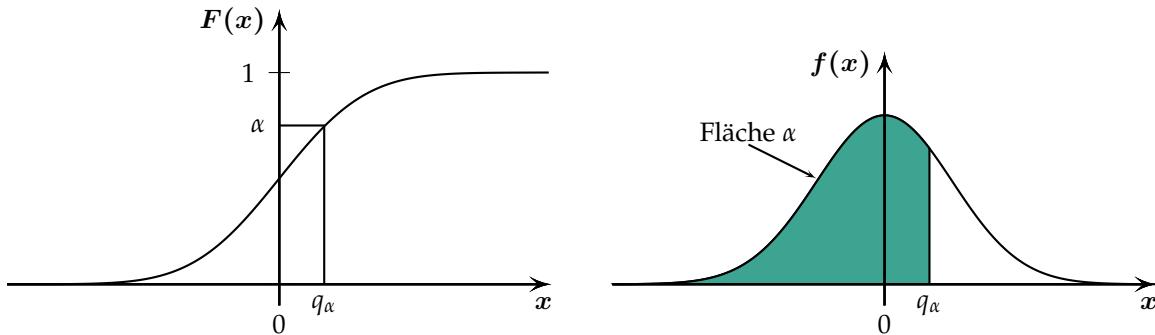


Abbildung 3.5.: Illustration des Quantils q_α anhand der Verteilungsfunktion $F(x)$ (links) und der Dichte $f(x)$ (rechts) für $\alpha = 0.75$.

Beispiel 3.1.5

Wir betrachten wiederum die Verteilung von Körpergrößen. Ist beispielsweise für $\alpha = 0.75$ das zugehörige Quantil gegeben durch

$$q(\alpha) = 182.5$$

so bedeutet dies, dass 75 % der gemessenen Personen kleiner oder gleich 182.5 cm sind.

□

3.2. Wichtige stetige Verteilungen

Im diskreten Fall sind die Binomialverteilung und die Poisson-Verteilung die wichtigsten diskreten Verteilungen. In diesem Kapitel werden wir die wichtigsten stetigen Verteilungen kennenlernen.

Wir haben im Unterkapitel 3.1 gesehen, dass wir die Wahrscheinlichkeitsverteilung einer stetigen Zufallsvariablen mit der kumulativen Verteilungsfunktion F oder der Dichte f charakterisieren können. Wir wollen nun die Funktionen F und f in konkreten Fällen kennenlernen.

3.2.1. Uniforme Verteilung

Die uniforme Verteilung tritt als Formalisierung der völligen „Ignoranz“ auf.

Uniforme Wahrscheinlichkeitsverteilung

Eine Zufallsvariable X mit Wertebereich $W_X = [a, b]$ heisst Uniform($[a, b]$) verteilt, falls

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{falls } a \leq x \leq b \\ 0 & \text{sonst} \end{cases}$$

Die Dichte ist also konstant auf dem Intervall $[a, b]$ (siehe Abbildung 3.6 links). Das heisst, dass die gleiche Wahrscheinlichkeit vorliegt auf dem ganzen Wertebereich $W_X = [a, b]$, deshalb der Name *uniform* (gleichförmig).

Der Funktionswert $\frac{1}{b-a}$ im Intervall $[a, b]$ folgt aus der Tatsache, dass die Fläche unter der gesamten Dichtekurve 1 sein muss, d.h. die Fläche des Rechtecks muss 1 sein (siehe Abbildung 3.6 links). Die Breite des Intervall ist $b - a$ und dann muss die Höhe $\frac{1}{b-a}$ sein.

Die zugehörige kumulative Verteilungsfunktion (siehe Abbildung 3.6 rechts) erhalten wir durch Integration der Dichtefunktion. Wir erhalten dann

$$F(x) = \begin{cases} 0 & \text{falls } x < a \\ \frac{x-a}{b-a} & \text{falls } a \leq x \leq b \\ 1 & \text{falls } x > b \end{cases}$$

Für $X \sim \text{Uniform}([a, b])$ sind die Kennzahlen wie folgt:

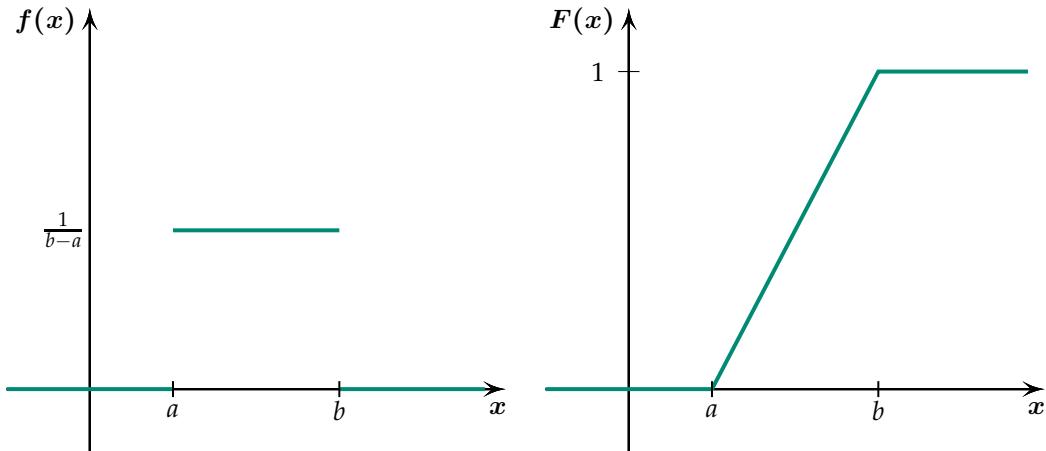


Abbildung 3.6.: Dichte (links) und Verteilungsfunktion (rechts) der uniformen Verteilung.

$$E(X) = \frac{a+b}{2}, \quad \text{Var}(X) = \frac{(b-a)^2}{12}, \quad \sigma_X = \frac{b-a}{\sqrt{12}}$$

Begründung (der Vollständigkeit halber):

$$\begin{aligned} E(X) &= \int_{-\infty}^{\infty} xf(x) dx = \int_a^b \frac{x}{b-a} dx = \frac{b^2 - a^2}{2(b-a)} = \frac{a+b}{2} \\ \text{Var}(X) &= \int_{-\infty}^{\infty} (x - E(X))^2 f(x) dx = \int_a^b \left(x - \frac{a+b}{2}\right)^2 \frac{1}{b-a} dx = \frac{(b-a)^2}{12} \\ \sigma_X &= \frac{b-a}{\sqrt{12}} \end{aligned}$$

Beispiel 3.2.1

Mit **Python** lässt sich der Wert der Wahrscheinlichkeitsdichtefunktion `Uniform([1, 10])` an der Stelle $x = 5$ folgendermassen berechnen (siehe Abbildung 3.7) ([zu R](#))

```
from scipy.stats import uniform, expon, norm

uniform.pdf(x=5, loc=1, scale=9)

## 0.1111111111111111
```

Dieser Wert ist *keine* Wahrscheinlichkeit, sondern nur der Wert der Wahrscheinlichkeitsdichtefunktion.

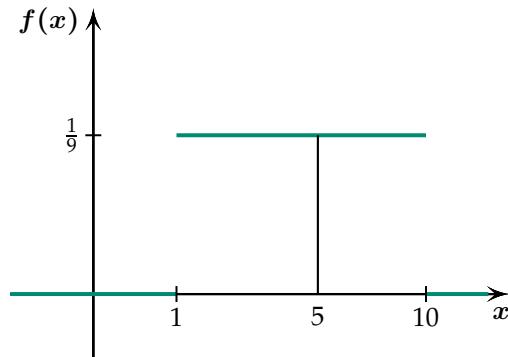


Abbildung 3.7.: Dichtefunktion von $\text{Uniform}([1, 10])$ ausgewertet an der Stelle $x = 5$

Bemerkungen:

- i. Der Anhang `pdf(...)` zum Befehl `uniform` ist die Abkürzung für *probability density function*. Dies heisst auf deutsch *Wahrscheinlichkeitsdichtefunktion*.

ii. **Vorsicht:**

Im Gegensatz zu den Funktionen `binom.pmf()` oder `poisson.pmf()` berechnet die Funktion `uniform.pdf()` *nicht* die zugehörige Wahrscheinlichkeit, sondern die Wahrscheinlichkeitsdichte. Deswegen werden die beiden Funktionen auch verschieden bezeichnet, nämlich `.pmf()` und `.pdf()`.

- iii. Die Eingabe des Intervales $[a, b]$ geschieht nicht mit Anfangs- und Endpunkt, sondern mit Anfangspunkt `loc=a` und der Länge des Intervalle `scale=b-a`

Falls $X \sim \text{Uniform}([1, 10])$, dann entspricht die Wahrscheinlichkeit $P(1 \leq X \leq 5)$ in diesem Fall gerade der Wahrscheinlichkeit $P(X \leq 5)$, da $P(X \leq 1)$ gleich 0 ist. Diese Wahrscheinlichkeit können wir als Fläche darstellen (siehe Abbildung 3.8).

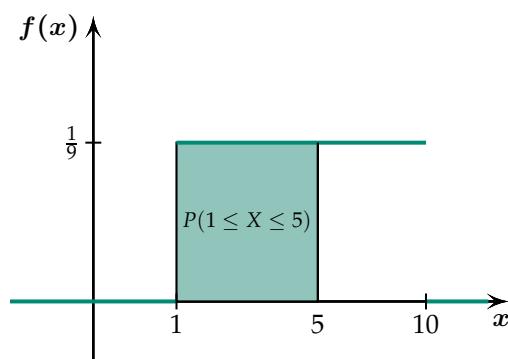


Abbildung 3.8.: Darstellung der Wahrscheinlichkeit $P(1 \leq X \leq 5)$ als Fläche

Mit `Python` berechnen wir diese Wahrscheinlichkeit folgendermassen: ([zu R](#))

Kapitel 3. Modelle für Messdaten

```
uniform.cdf(x=5, loc=1, scale=9)

## 0.4444444444444444
```

Der Anhang **.cdf()** berechnet die kumulative Verteilungsfunktion an der Stelle **x=...**. Der Anhang steht für *cumulative distribution function*.

Die Wahrscheinlichkeit $P(1.2 \leq X \leq 4.8)$ ist in Abbildung 3.9 dargestellt.

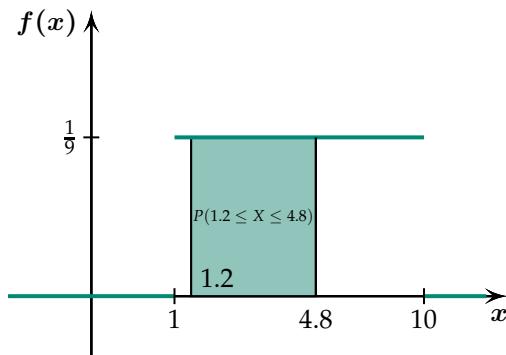


Abbildung 3.9.: Darstellung der Wahrscheinlichkeit $P(1.2 \leq X \leq 4.8)$ als Fläche

Mit **Python**: ([zu R](#))

```
uniform.cdf(x=4.8, loc=1, scale=9) - uniform.cdf(x=1.2, loc=1, scale=9)

## 0.4
```

Von grosser Bedeutung ist die Generierung von uniform verteilten Zufallsvariablen. Mit **Python** lassen sich uniform verteilte Zufallsvariablen folgendermassen erzeugen: ([zu R](#))

```
uniform.rvs(loc=1, scale=9, size=5)

## [6.11701418 5.14810356 8.30841588 7.4116105 2.43940551]
```

Der Anhang **.rvs()** (*random variates*) berechnet **size=...** Zufallszahlen.

□

3.2.2. Exponentialverteilung

Die Exponentialverteilung ist das einfachste Modell für *Wartezeiten auf Ausfälle*, also für die *Lebensdauer*.

Beispiel 3.2.2

Lebenszeit von elektronischen Geräten, wenn Alterungserscheinungen nicht betrachtet werden müssen.

□

Beispiel 3.2.3

Wie lange müssen wir warten, bis der nächste Zerfall eines Alphastrahlers stattfindet? In diesem Fall fassen wir die Zerfallszeit als eine Lebensdauer auf und zwar als Lebensdauer eines Isotops.

□

Die Poissonverteilung beschreibt die *Anzahl Beobachtungen* in einem festen Zeintervall. Mit der Exponentialverteilung ermitteln wir die Wahrscheinlichkeit für eine *Lebensdauer*.

Notation

Die natürliche Exponentialfunktion e^x schreiben wir oft in der Form:

$$\exp(x) := e^x$$

Exponentialverteilung

Eine Zufallsvariable X mit Wertebereich $W_X = \mathbb{R}^+ = [0, \infty)$ heisst *exponentiellverteilt* mit Parameter $\lambda \in \mathbb{R}^+$ falls

$$f(x) = \begin{cases} \lambda \cdot \exp(-\lambda x), & \text{falls } x \geq 0 \\ 0 & \text{sonst} \end{cases}$$

Wir schreiben

$$X \sim \text{Exp}(\lambda)$$

Kapitel 3. Modelle für Messdaten

Die zugehörige kumulative Verteilungsfunktion ist gegeben durch

$$F(x) = \begin{cases} 1 - e^{-\lambda x} & \text{falls } x \geq 0 \\ 0 & \text{falls } x < 0 \end{cases}$$

Die Dichte und kumulative Verteilungsfunktion sind für verschiedene λ in Abbildung 3.10 zu sehen.

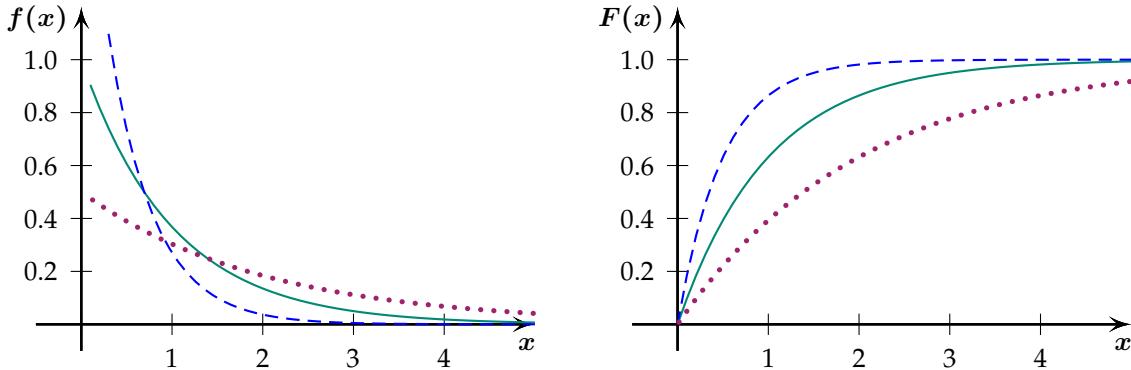


Abbildung 3.10.: Dichte und Verteilungsfunktion der Exponentialverteilung für $\lambda = 1$ (grün), $\lambda = 2$ (blau gestrichelt) und $\lambda = 1/2$ (violett gepunktet).

Für $X \sim \text{Exp}(\lambda)$ sind die Kennzahlen wie folgt:

$$\mathbb{E}(X) = \frac{1}{\lambda}, \quad \text{Var}(X) = \frac{1}{\lambda^2}, \quad \sigma_X = \frac{1}{\sqrt{\lambda}}$$

Begründung (zur Vollständigkeit):

$$\begin{aligned} \mathbb{E}(X) &= \int_{-\infty}^{\infty} x f(x) \, dx = \int_0^{\infty} x \cdot \lambda \cdot e^{-\lambda x} \, dx = \frac{1}{\lambda} \\ \text{Var}(X) &= \int_{-\infty}^{\infty} (x - \mathbb{E}(X))^2 f(x) \, dx = \int_0^{\infty} \left(x - \frac{1}{\lambda}\right)^2 \cdot \lambda \cdot e^{-\lambda x} \, dx = \frac{1}{\lambda^2} \\ \sigma_X &= \sqrt{\text{Var}(X)} = \sqrt{\frac{1}{\lambda^2}} = \frac{1}{\sqrt{\lambda}} \end{aligned}$$

Beispiel 3.2.4

Wie lange dauert es, bis ein bestimmtes radioaktives Isotop mit einer Wahrscheinlichkeit von $1/2$ zerfallen ist?

Kapitel 3. Modelle für Messdaten

Wir bezeichnen mit T die Zerfallszeit. Diese kann auch als Lebensdauer aufgefasst werden. Als Modell für diese zufällige Dauer ist die Exponentialverteilung geeignet, also

$$T \sim \text{Exp}(\lambda)$$

Der Parameter λ hängt vom jeweiligen Isotop ab und muss in der Regel aus Experimenten geschätzt werden. Für welchen Zeitpunkt wird die Wahrscheinlichkeit, dass das Isotop bis dahin zerfällt, resp. „überlebt“, gleich $1/2$?

Die Antwort gibt der Median

$$F(t_{1/2}) = 1 - \exp(-\lambda t_{1/2}) = \frac{1}{2} \Rightarrow \exp(-\lambda t_{1/2}) = \frac{1}{2}$$

Diese Gleichung lösen wir nach $t_{1/2}$ auf, indem wir beide Seiten der Gleichung logarithmieren:

$$-\lambda t_{1/2} = \ln\left(\frac{1}{2}\right) \Rightarrow t_{1/2} = \frac{\ln(2)}{\lambda} = \frac{0.693}{\lambda}$$

In einem radioaktiven Sample gibt es allerdings sehr viele aktive Isotope. Wenn die Wahrscheinlichkeit, dass ein einzelnes Isotop bis zum Zeitpunkt $\frac{0.693}{\lambda}$ zerfällt, $1/2$ beträgt, dann ist die relative Häufigkeit der überlebenden Isotope im Sample zum Zeitpunkt $\frac{0.693}{\lambda}$ ebenfalls $1/2$. Die relative Häufigkeit der zerfallenen aktiven Isotope beträgt demnach ebenfalls $1/2$. Man nennt deshalb

$$t_{1/2} = \frac{\ln(2)}{\lambda}$$

die *Halbwertszeit*. Nach dieser Zeit sind im Mittel die Hälfte aller aktiven Isotope zerfallen.

□

Beispiel 3.2.5

Angenommen $X \sim \text{Exp}(3)$, dann lässt sich die Wahrscheinlichkeit $P(0 \leq X \leq 4)$ wieder graphische darstellen (siehe Abbildung 3.11)

Diese Wahrscheinlichkeit berechnen wir mit **Python** wie folgt: ([zu R](#))

```
from scipy.stats import uniform, expon, norm

expon.cdf(x=4, scale=1/3)

## 0.9999938557876467
```

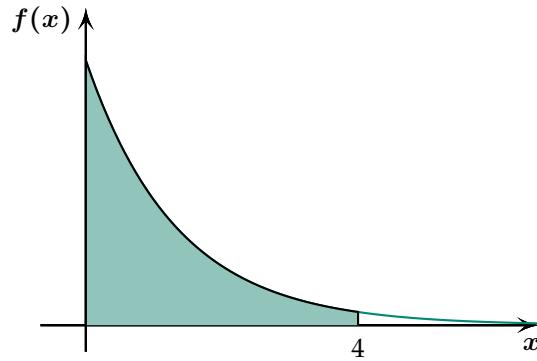


Abbildung 3.11.: Wahrscheinlichkeit $P(0 \leq X \leq 4)$ als Fläche dargestellt

Bemerkungen:

- i. Für `scale` muss $\frac{1}{\lambda}$ eingegeben werden.
- ii. Die Anhänge `.pdf` (Dichtefunktion), `.cdf` (kumulative Verteilungsfunktion), `.rvs` (Zufallszahlen) und `.ppf` (Quantile) haben die gleiche Bedeutung wie bei der uniformen Verteilung.

□

Es besteht folgender Zusammenhang zwischen der Exponential- und der Poisson-Verteilung:

Zusammenhang zwischen Poisson-Verteilung und Exponentialverteilung

Wenn die Zeiten zwischen den Ausfällen eines Systems exponential(λ)-verteilt sind, dann ist die Anzahl Ausfälle in einem Intervall der Länge t Poisson(λt)-verteilt.

Beispiel 3.2.6

Angenommen, zum Zeitpunkt $t_0 = 0$ ereignet sich in einem radioaktiven Sample ein radioaktiver Zerfall. Wie gross ist die Wahrscheinlichkeit, dass erst nach dem Zeitpunkt t erneut ein Zerfall eintreten kann?

Die Wahrscheinlichkeit, dass sich erst nach der Zeit t wieder ein Zerfall ereignet, ist

$$P(T > t) = P(\text{kein Zerfall in } [0, t])$$

Kapitel 3. Modelle für Messdaten

Die Anzahl Zerfälle im Zeitintervall $[0, t]$ folgt einer Poisson-Verteilung mit Parameter λt . Folglich ist

$$P(T > t) = P(\text{kein Zerfall in } [0, t]) = \frac{(\lambda t)^0 e^{-\lambda t}}{0!} = e^{-\lambda t}$$

Also folgt die Lebenszeit T eines Atoms einer Exponentialverteilung mit Parameter λ . Die kumulative Verteilungsfunktion ist gegeben durch

$$F(t) = P(T \leq t) = 1 - P(T > t) = 1 - e^{-\lambda t} \quad \text{für } t \geq 0$$

Dies ist gerade die kumulative Verteilungsfunktion der Exponentialverteilung.

□

3.2.3. Normalverteilung (Gauss-Verteilung)

Die *Normalverteilung* (manchmal auch *Gauss-Verteilung* genannt) ist die häufigste Verteilung für Messwerte. Sie tritt in vielen Anwendungen auf und ist die wichtigste Wahrscheinlichkeitsverteilung in der Statistik. Sie hat neben praktischer auch grosse theoretische Bedeutung.

Normalverteilung

Eine Zufallsvariable X mit Wertebereich $W_X = \mathbb{R}$ heisst *normalverteilt* mit Parametern $\mu \in \mathbb{R}$ und $\sigma^2 \in \mathbb{R}^+$ falls

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Wir notieren die Verteilung für die Zufallsvariable X folgendermassen

$$X \sim \mathcal{N}(\mu, \sigma^2)$$

Die zugehörige kumulative Verteilungsfunktion

$$F(x) = \int_{-\infty}^x f(y) dy$$

ist nicht explizit mit Standardfunktionen wie x^2 , $\exp(x)$, $\log(x)$ etc. darstellbar. Diese Integrale werden (von Computersoftware) numerisch berechnet.

Für $X \sim \mathcal{N}(\mu, \sigma^2)$ sind die Kennzahlen wie folgt:

$$\mathrm{E}(X)x = \mu, \quad \mathrm{Var}(X) = \sigma^2, \quad \sigma_X = \sigma$$

Begründung (die Auswertung der Integrale ist allerdings mühsam):

$$\begin{aligned}\mathrm{E}(X) &= \int_{-\infty}^{\infty} xf(x) \, dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \, dx = \mu \\ \mathrm{Var}(X) &= \int_{-\infty}^{\infty} (x - \mathrm{E}(X))^2 f(x) \, dx = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} (x - \mu)^2 \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \, dx = \sigma^2 \\ \sigma_X &= \sigma\end{aligned}$$

Das heisst, dass die Parameter μ und σ^2 eine natürliche Interpretation als Erwartungswert und Varianz der Verteilung haben. Drei Normalverteilungen mit verschiedenen Werten von μ und σ sind in Abbildung 3.12 dargestellt.

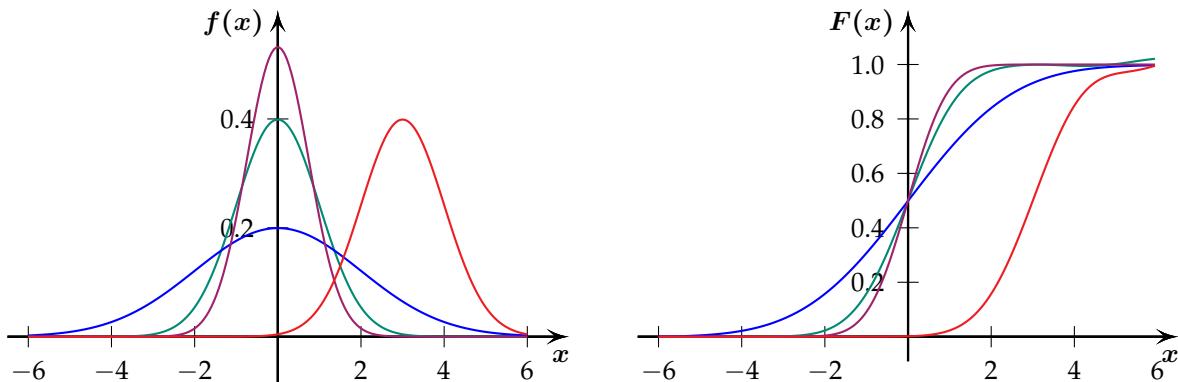


Abbildung 3.12.: Dichten (*links*) und kumulative Verteilungsfunktionen (*rechts*) der Normalverteilungen für $\mu = 0, \sigma = 1$ (grün), $\mu = 0, \sigma = 2$ (blau), $\mu = 0, \sigma = 0.75$ (violett) und $\mu = 3, \sigma = 1$ (rot).

Die Dichte der Normalverteilung ist symmetrisch um den Erwartungswert μ . Je grösser σ , desto flacher oder breiter wird die Dichte. Für kleine σ gibt es einen „schmalen und hohen“ Gipfel. Mit μ verschieben wir einfach die Dichte nach links bzw. rechts.

Beispiel 3.2.7

Der Intelligenzquotient (IQ) wird in der Regel mit Intelligenztests ermittelt. Die Ergebnisse von einem IQ Test folgen in etwa einer Normalverteilung mit Mittelwert 100 und Standardabweichung 15.

- a) Im Allgemeinen gilt eine Person als hochbegabt, wenn ihr IQ zwei und mehr Standardabweichungen vom Mittelwert nach oben entfernt ist. Wir suchen die Wahrscheinlichkeit, dass jemand hochbegabt ist, also einen IQ von mehr als 130 hat. Dies ist die Wahrscheinlichkeit $P(X > 130)$, wobei

$$X \sim \mathcal{N}(100, 15^2)$$

Diese Wahrscheinlichkeit können wir wieder als Fläche darstellen (siehe Abbildung 3.13).

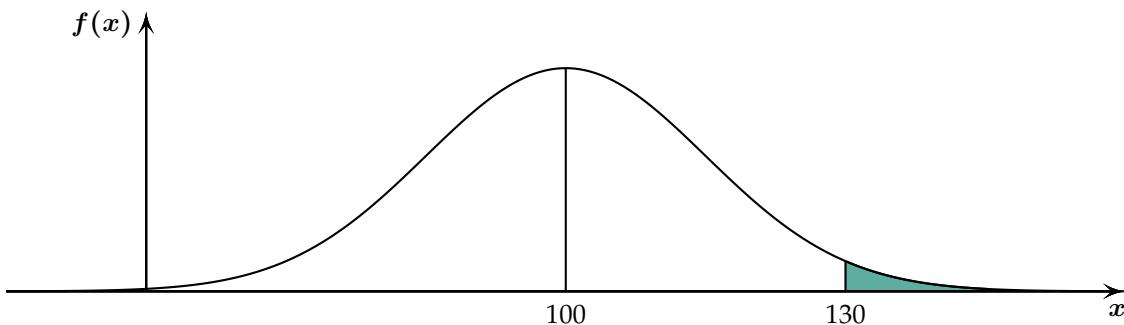


Abbildung 3.13.: Wahrscheinlichkeit $P(X > 130)$

Mit **Python** suchen wir also die Wahrscheinlichkeit ([zu R](#))

$$1 - P(X \leq 130)$$

```
from scipy.stats import uniform, expon, norm
1 - norm.cdf(x=130, loc=100, scale=15)
## 0.02275013194817921
```

Also rund 2 % der Bevölkerung ist hochbegabt.

Bemerkungen:

- i. **loc** steht hier für den Erwartungswert μ und **scale** für die Standardabweichung σ .
 - ii. Hier wird $P(X = 130)$ auch subtrahiert, aber da $P(X = 130) = 0$ ist, spielt dies bei stetigen im Gegensatz zu diskreten Verteilungen keine Rolle.
- b) Wir können fragen, welches Intervall enthält 95 % der IQ's um den Mittelwert $\mu = 100$. Auch hier stellen wir diese Wahrscheinlichkeit als Fläche dar (siehe Abbildung 3.14).

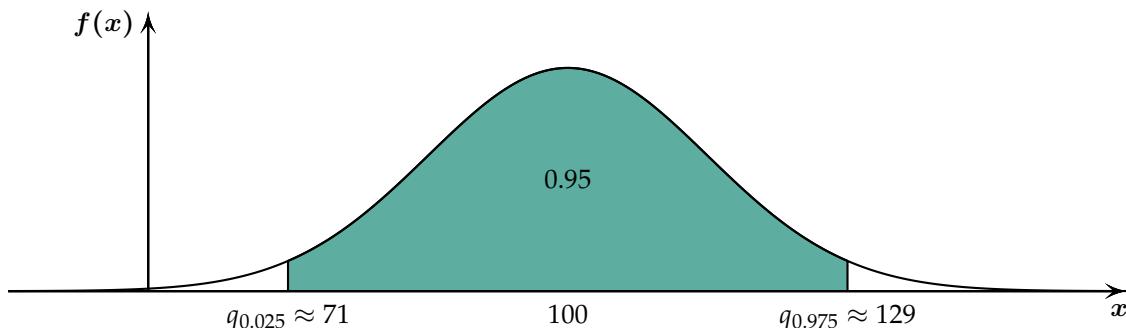


Abbildung 3.14.: Quartile für 95 % der Fläche um 100

Die Fläche in der Mitte der Abbildung 3.14 nimmt dann 95 % der Gesamtfläche ein. Die kleinen Flächen links und rechts sind dann jeweils 0.025.

Die Wahrscheinlichkeiten sind also gegeben und wir suchen die zugehörigen Werte. Dies entspricht nichts anderem als der Bestimmung der Quartile $q_{0.025}$ und $q_{0.975}$.

Mit **Python** bestimmen wir die Quantile wie folgt: [\(zu R\)](#)

```
norm.ppf(q=0.025, loc=100, scale=15)

norm.ppf(q=0.975, loc=100, scale=15)

## 70.60054023189917
## 129.39945976810083
```

Oder kürzer: [\(zu R\)](#)

```
norm.ppf(q=[0.025,0.975], loc=100, scale=15)

## [ 70.60054023 129.39945977]
```

Also haben 95 % der Menschen einen IQ zwischen ungefähr 70 und 130. Dies entspricht aber gerade einem Abstand von etwa 2 Standardabweichungen vom Mittelwert $\mu = 100$.

Bemerkungen:

- Der Anhang **ppf(...)** zum Befehl **norm** ist die Abkürzung für *probability point function*. Diese Funktion ist die Umkehrung der **cdf()**-Funktion. Sie bestimmt aus einem Wert eine Wahrscheinlichkeit.
- Wir können uns auch fragen, wie viel Prozent der Bevölkerung innerhalb einer Standardabweichung vom Mittelwert liegen. Wir suchen also die Wahrscheinlichkeit

$$P(85 \leq X \leq 115)$$

Auch diese Wahrscheinlichkeit stellen wieder als Fläche dar (siehe Abbildung 3.15).

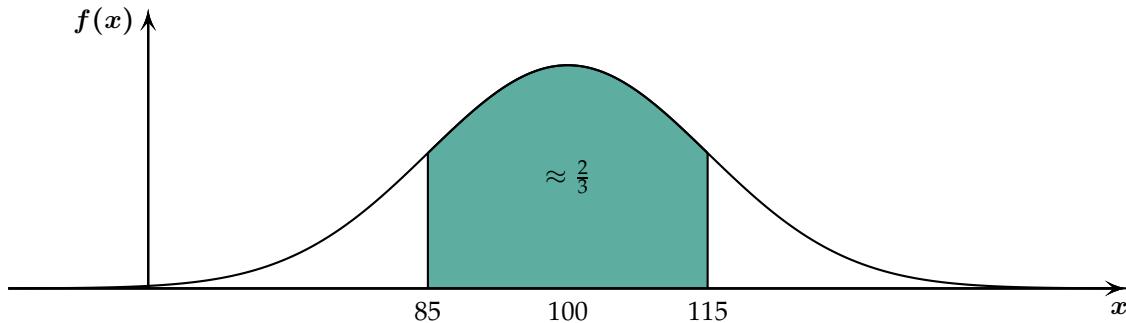


Abbildung 3.15.: Wahrscheinlichkeit für IQ zwischen $\mu \pm \sigma$

Mit **Python** berechnen wir diese Wahrscheinlichkeit wie folgt

```
norm.cdf(x=115, loc=100, scale=15) - norm.cdf(x=85, loc=100, scale=15)
## 0.6826894921370859
```

D.h., etwa $2/3$ der Bevölkerung haben einen IQ zwischen 85 und 115.

□

Die letzten beiden Resultate aus dem Beispiel oben gelten für alle Normalverteilungen $\mathcal{N}(\mu, \sigma^2)$. Die Wahrscheinlichkeit, dass eine Beobachtung höchstens eine Standardabweichung vom Erwartungswert abweicht, ist etwa $2/3$:

$$P(\mu - \sigma \leq X \leq \mu + \sigma) \approx \frac{2}{3}$$

Wir können auch noch die Wahrscheinlichkeit berechnen, dass eine Beobachtung höchstens zwei Standardabweichungen vom Erwartungswert abweicht:

$$P(\mu - 2\sigma \leq X \leq \mu + 2\sigma) \approx 0.95$$

Da diese Wahrscheinlichkeiten durch Integrale berechnet werden, können wir sie als Flächen interpretieren. Die Fläche der Normalverteilung über dem Intervall $[\mu - \sigma, \mu + \sigma]$ ist ca. $2/3$. Die Fläche über dem Intervall $[\mu - 2\sigma, \mu + 2\sigma]$ ist ca. 0.95, siehe auch Abbildung 3.16.

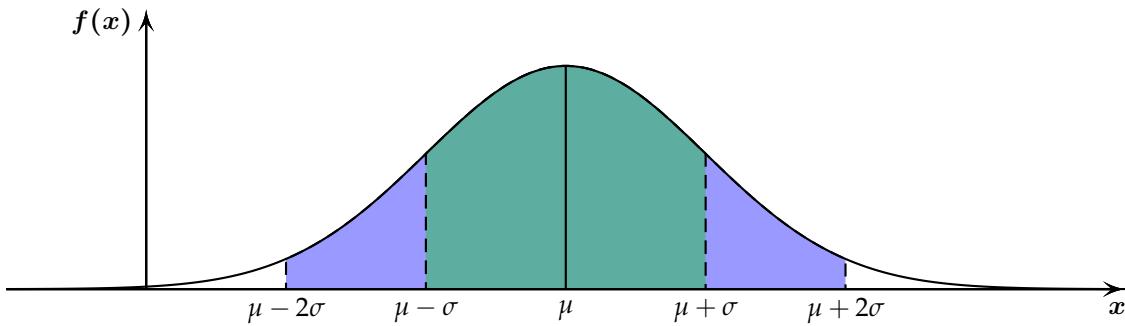


Abbildung 3.16.: Dichte der Normalverteilung. Ca. 68 % der Fläche befindet sich im Intervall $[\mu - \sigma, \mu + \sigma]$, ca. 95 % der Fläche im Intervall $[\mu - 2\sigma, \mu + 2\sigma]$.

Die Standardnormalverteilung

Die Normalverteilung $\mathcal{N}(0, 1)$ trägt einen speziellen Namen.

Standardnormalverteilung

Die Normalverteilung mit $\mu = 0$ und $\sigma^2 = 1$ heisst *Standardnormalverteilung*. Deren Dichte und kumulative Verteilungsfunktion werden wie folgt bezeichnet:

$$\varphi(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

$$\Phi(x) = \int_{-\infty}^x \varphi(y) \, dy$$

Wie die Werte von Φ mit **Python** ermittelt werden können, betrachten wir im nächsten Beispiel.

Beispiel 3.2.8

Sei $Z \sim \mathcal{N}(0, 1)$.

- Was ist $P(Z \leq 1.13)$? ([zu R](#))

```
from scipy.stats import uniform, expon, norm

norm.cdf(x=1.13)
## 0.8707618877599821
```

- Für welchen Wert von z ist $\Phi(z) = 0.7910$? ([zu R](#))

Anders gefragt: Was ist $\Phi^{-1}(0.7910)$?

```
norm.ppf(q=0.7910)
## 0.8098959147358981
```

Dies ist gerade das 0.7910-Quantil.

- Was ist $P(Z \leq -0.2)$?

Weil die Standardnormalverteilung um null herum symmetrisch ist, ist die Fläche links von -0.2 wegen der Symmetrie genau so gross wie die Fläche rechts von 0.2 . D.h.:

$$P(Z \leq -0.2) = P(Z \geq 0.2) = 1 - P(Z \leq 0.2)$$

Dies überprüfen wir leicht mit [Python](#) ([zu R](#))

```
norm.cdf(x=-0.2)
1 - norm.cdf(x=0.2)
## 0.42074029056089696
## 0.420740290560897
```

□

Wir werden sehen, dass eine Normalverteilung $\mathcal{N}(\mu, \sigma^2)$ immer in eine Standard-Normalverteilung transformiert werden kann. Deshalb genügen an sich die Werte von Φ , um Wahrscheinlichkeiten und Quantile einer allgemeinen $\mathcal{N}(\mu, \sigma^2)$ -Verteilung zu berechnen.

3.3. Funktionen einer Zufallsvariable

Wenn $g : \mathbb{R} \rightarrow \mathbb{R}$ eine Funktion von \mathbb{R} nach \mathbb{R} und X eine Zufallsvariable ist, dann ist die Zusammensetzung

$$Y = g(X)$$

eine *neue* Zufallsvariable. Die Zusammensetzung bedeutet einfach, dass zu jeder Realisierung x von X die Realisierung $y = g(x)$ von Y gehört. Solche Transformationen treten häufig auf. Wir beginnen mit dem wichtigsten Spezialfall:

3.3.1. Lineare Transformationen von Zufallsvariablen

Wir betrachten hier zuerst den Fall einer *linearen Transformation*

$$y = g(x) = a + bx \quad (a, b \in \mathbb{R})$$

Beispiel 3.3.1

Wir haben in einer Messreihe Temperaturen in Grad Celsius gemessen und wollen diese nun in Grad Fahrenheit umformen. Die Temperatur T_c in Grad Celsius kann man folgendermassen in die Temperatur T_F in Grad Fahrenheit umrechnen:

$$T_F = \frac{9}{5} \cdot T_C + 32$$

Wir kennen die Standardabweichung des Messfehlers auf dieser Skala: $\sigma_C = 1/3$ Grad Celsius.

Wie lautet die Standardabweichung des Messfehlers aber nun in Grad Fahrenheit? Wir könnten natürlich alle Temperaturen mit der Formel oben in Grad Fahrenheit umrechnen und dann die Standardabweichung erneut bestimmen. Dies ist aber nicht nötig, wie wir gleich sehen werden.

□

Eigenschaften von linearen Transformationen einer Zufallsvariablen

Für

$$Y = a + bX$$

gelten dann folgende Beziehungen:

(i) Erwartungswert:

$$\mathbb{E}(Y) = \mathbb{E}(a + bX) = a + b\mathbb{E}(X)$$

(ii) Varianz, Standardabweichung:

$$\text{Var}(Y) = \text{Var}(a + bX) = b^2 \text{Var}(X), \quad \sigma_Y = |b|\sigma_X$$

(iii) Quantile: α – Quantil von Y

$$q_Y(\alpha) = a + bq_X(\alpha)$$

(iv) Standardisierung:

$$f_Y(y) = \frac{1}{b} f_X\left(\frac{y-a}{b}\right)$$

Die Richtigkeit der ersten Gleichung können wir uns leicht überlegen. Für den Spezialfall

$$Y = a + X$$

werden die Beobachtungen X alle um a verschoben und damit verschiebt sich auch der Erwartungswert um a . Für den anderen Spezialfall

$$Y = bX$$

werden alle Beobachtungen X mit b multipliziert. Somit wird auch der Erwartungswert von X mit b multipliziert.

Der Vollständigkeit halber folgen noch die Herleitungen der 4 Regeln.

Die erste Gleichung folgt aus

$$\begin{aligned} E(Y) &= E(a + bX) \\ &= \int_{-\infty}^{\infty} (a + bx) f_X(x) dx \\ &= a \int_{-\infty}^{\infty} f_X(x) dx + b \int_{-\infty}^{\infty} x f_X(x) dx \\ &= a \cdot 1 + b \cdot E(X) \\ &= a + b E(X) \end{aligned}$$

Die zweite Gleichung folgt aus der Definition der Varianz und aus der ersten Gleichung:

$$\begin{aligned} \text{Var}(Y) &= \text{Var}(a + bX) \\ &= E((a + bX - E(a + bX))^2) \\ &= E((a + bX - (a + bE(X)))^2) = E(b^2(X - E(X))^2) \\ &= b^2 E((X - E(X))^2) = b^2 \text{Var}(X) \end{aligned}$$

Durch Ziehen der Wurzel erhält man für die Standardabweichung $\sigma_Y = |b| \cdot \sigma_X$.

Die dritte Gleichung folgt aus

$$\begin{aligned} F_Y(q_Y(\alpha)) &= P(Y \leq q_Y(\alpha)) \\ &= P(a + bX \leq q_Y(\alpha)) \\ &= P\left(X \leq \frac{q_Y(\alpha) - a}{b}\right) \\ &= F_X\left(\frac{q_Y(\alpha) - a}{b}\right) \end{aligned}$$

Aus

$$F_Y(q_Y(\alpha)) = \alpha = F_X\left(\frac{q_Y(\alpha) - a}{b}\right)$$

folgt $\frac{q_Y(\alpha) - a}{b} = q_X(\alpha)$ und somit die dritte Gleichung. Die vierte Beziehung erhält man aus

$$\begin{aligned} f_Y(y) &= \frac{d}{dy} F_Y(y) \\ &= \frac{d}{dy} F_X\left(\frac{y - a}{b}\right) \\ &= \frac{1}{b} f_X\left(\frac{y - a}{b}\right) \end{aligned}$$

Wir haben dabei folgende Gleichheit benutzt

$$\begin{aligned} F_Y(y) &= P(Y \leq y) \\ &= P(aX + b \leq y) \\ &= P\left(X \leq \frac{y - b}{a}\right) \\ &= F_X\left(\frac{y - b}{a}\right) \end{aligned}$$

Beispiel 3.3.2

Wir haben eine Temperatur in Grad Celsius gemessen und kennen die Standardabweichung des Messfehlers auf dieser Skala: $\sigma_C = 1/3$ Grad Celsius. Für einen Bericht, der im englischsprachigen Raum gelesen werden soll, wollen wir die Temperatur aber nicht in Grad Celsius, sondern in Grad Fahrenheit angeben.

Wie gross ist die Standardabweichung σ_F des Messfehlers, wenn wir die Temperatur in Grad Fahrenheit angeben?

Wie wir schon gesehen haben, können wir die Temperatur T_c in Grad Celsius in die Temperatur T_F in Grad Fahrenheit umrechnen:

$$T_F = \frac{9}{5} \cdot T_C + 32$$

Daher ist die Standardabweichung in Grad Fahrenheit (nach Regel (ii) für lineare Transformationen $b = \frac{9}{5}$):

$$\sigma_F = \frac{9}{5} \sigma_C = \frac{9}{5} \cdot \frac{1}{3} = \frac{3}{5}$$

□

Beispiel 3.3.3 Fahrenheit für Fortgeschrittene

Wenn die Messungen von T_C einer Normalverteilung mit Mittelwert μ_C und Standardabweichung σ_C folgen, also $T_C \sim \mathcal{N}(\mu_C, \sigma_C^2)$, dann ist

$$T_F = a + bT_C \sim \mathcal{N}(a + b\mu_C, b^2\sigma_C^2)$$

mit $a = 32$ und $b = \frac{9}{5}$.

Denn nach der Regel (iv) der Regeln für lineare Transformationen gilt

$$\begin{aligned} f_{T_F}(T_F) &= \frac{1}{\sqrt{2\pi}\sigma_C b} \exp\left(-\frac{\left(\frac{T_F-a}{b} - \mu_C\right)^2}{2\sigma_C^2}\right) \\ &= \frac{1}{\sqrt{2\pi}\sigma_C b} \exp\left(-\frac{(T_F - (a + b\mu_C))^2}{2\sigma_C^2 b^2}\right) \end{aligned}$$

Eine linear transformierte Normalverteilung ist also wiederum eine Normalverteilung. Diese Eigenschaft, dass man mit linearen Transformationen innerhalb der Verteilungsklasse bleibt, ist eine spezielle Eigenschaft der Normalverteilung und im allgemeinen nicht richtig.

□

Standardisieren einer Zufallsvariablen

Wir können X immer linear transformieren, so dass die transformierte Zufallsvariable Erwartungswert gleich 0 und Varianz gleich 1 hat. Dies geschieht wie folgt: wir betrachten die lineare Transformation

$$g(x) = \frac{x - E(X)}{\sigma_X} = -\frac{E(X)}{\sigma_X} + \frac{1}{\sigma_X}x = a + bx$$

mit

$$a = -\frac{E(X)}{\sigma_X} \quad \text{und} \quad b = \frac{1}{\sigma_X}$$

Damit bilden wir die transformierte Zufallsvariable

$$Z = g(X) = \frac{X - E(X)}{\sigma_X}$$

Mit Hilfe der Regeln für lineare Transformationen gilt dann für den Erwartungswert

$$E(Z) = a + bE(X) = -\frac{E(X)}{\sigma_X} + \frac{1}{\sigma_X}E(X) = 0$$

und für die Varianz

$$\text{Var}(Z) = b^2 \text{Var}(X) = \left(\frac{1}{\sigma_X}\right)^2 \text{Var}(X) = \frac{1}{\sigma_X^2} \cdot \sigma_X^2 = 1$$

Standardisieren einer normalverteilten Zufallsvariablen

Falls $X \sim \mathcal{N}(\mu, \sigma^2)$, so ist die standardisierte Zufallsvariable wieder normalverteilt, hat nun aber Erwartungswert null und Varianz eins. Man erhält also die Standardnormalverteilung:

$$Z = \frac{X - \mu}{\sigma} \sim \mathcal{N}(0, 1)$$

Im Wesentlichen verschiebt das $-\mu$ das Maximum der Kurve der Normalverteilung in den Ursprung. Die Division durch σ bewirkt eine Stauchung (oder Streckung) der Kurve in x -Richtung, so dass die Standardabweichung gleich 1 wird.

Damit lassen sich Wahrscheinlichkeiten für beliebige Normalverteilungen mit Hilfe der Standardnormalverteilung berechnen. Es reicht daher, wenn man nur die Werte der Standardnormalverteilung mit **Python** berechnen kann.

Beispiel 3.3.4

Sei $X \sim \mathcal{N}(\mu, \sigma^2)$ mit $\mu = 2$ und $\sigma^2 = 4$. Berechnen Sie $P(X \leq 5)$.

$$\begin{aligned} P(X \leq 5) &= P\left(\frac{X - \mu}{\sigma} \leq \frac{5 - \mu}{\sigma}\right) \\ &= P\left(Z \leq \frac{5 - 2}{2}\right) \\ &= P(Z \leq 1.5) \\ &= \Phi(1.5) \\ &= 0.93 \end{aligned}$$

□

Beispiel 3.3.5

Sei $X \sim \mathcal{N}(\mu, \sigma^2)$ mit $\mu = 2$ und $\sigma^2 = 4$. Berechnen Sie $P(X > 5)$.

$$P(X > 5) = 1 - P(X \leq 5) = 1 - \Phi(1.5) = 1 - 0.93 = 0.07$$

□

Beispiel 3.3.6

Sei $X \sim \mathcal{N}(\mu, \sigma^2)$ mit $\mu = 2$ und $\sigma^2 = 4$. Wie gross ist das 90 % Quantil γ von X ?

$$\begin{aligned} P(X \leq \gamma) &= 0.9 \\ \Rightarrow P\left(Z \leq \frac{\gamma - \mu}{\sigma}\right) &= 0.9 \\ \Rightarrow \Phi\left(\frac{\gamma - \mu}{\sigma}\right) &= 0.9 \end{aligned}$$

Mit Hilfe einer Statistik-Software finden wir, dass

$$\frac{\gamma - \mu}{\sigma} = \Phi^{-1}(0.9) = 1.28$$

Auflösen nach γ ergibt:

$$\gamma = \mu + 1.28\sigma = 2 + 1.28 \cdot 2 = 4.56$$

□

Beispiel 3.3.7

Sei $X \sim \mathcal{N}(\mu, \sigma^2)$ mit $\mu = 2$ und $\sigma^2 = 4$. Berechnen Sie $P(|X| \leq 2)$.

$$\begin{aligned} P(|X| \leq 2) &= P(-2 \leq X \leq 2) \\ &= P(X \leq 2) - P(X \leq -2) \\ &= P\left(Z \leq \frac{2-2}{2}\right) - P\left(Z \leq \frac{-2-2}{2}\right) \\ &= P(Z \leq 0) - P(Z \leq -2) \\ &= \Phi(0) - \Phi(-2) \\ &= \Phi(0) - (1 - \Phi(2)) \\ &= 0.5 - (1 - 0.97) \\ &= 0.5 - 0.03 \\ &= 0.47 \end{aligned}$$

□

Bemerkungen:

- i. Wir hätten alle diese Aufgaben auch ohne Standardisierung mit dem **Python**-Befehl

```
norm.pdf(..., loc=2, scale=2)
```

```
norm.ppf(..., loc=2, scale=2)
```

lösen können.

Da stellt sich natürlich die Frage, *warum* man Zufallsvariablen standardisiert. Der Nutzen der Standardisierung hat historische Gründe. Vor dem Computerzeitalter musste man die Werte für Wahrscheinlichkeiten und Quantile für Normalverteilungen aus Tabellen ablesen, da man die Dichtefunktion der Normalverteilung nicht integrieren konnte (sie besitzt keine elementare Stammfunktion). Das heisst, man musste diese Integrale numerisch berechnen.

Das Problem ist nun aber, dass man für *jede* Verteilung (verschiedene μ und σ) solche Tabellen hätte erstellen müssen. Das geht aber nicht, da es unendlich viele solcher Tabellen gebraucht hätte. Durch Standardisieren musste allerdings nur eine solche Tabelle erstellt werden, nämlich die der Standardnormalverteilung.

- ii. Standardisieren hat heute noch theoretische Bedeutung, da es viele Herleitungen vereinfacht, aber es verliert dauernd an praktischer Bedeutung.
- iii. Wir werden die Standardisierung von Zufallsvariablen allerdings später noch bei der sogenannten *t*-Verteilung benötigen.

3.3.2. Nichtlineare Transformationen von Zufallsvariablen

Wir beschränken uns einfacheitshalber zuerst auf den Fall einer *quadratischen Transformation*

$$y = g(x) = x^2$$

Beispiel 3.3.8

Es bezeichne $X \sim \text{Uniform}([a, b])$ eine uniformverteilte Zufallsvariable. Wir definieren die Zufallsvariable Y durch die quadratische Transformation der Zufallsvariablen X

$$Y = X^2$$

Wie lautet also der Erwartungswert $E[Y]$?

Kapitel 3. Modelle für Messdaten

Wir könnten nun versucht sein, zuerst die Dichte von Y , also $f_Y(Y)$ zu bestimmen und dann aufgrund von $f_Y(y)$ den Erwartungswert $E[Y]$ zu berechnen. Es stellt sich allerdings heraus¹, dass die Berechnung des Erwartungswertes von Y viel einfacher geht:

$$E(Y) = \int_{-\infty}^{\infty} y f_Y(y) dy = \int_{-\infty}^{\infty} x^2 f_X(x) dx = \int_a^b x^2 \frac{1}{b-a} dx = \frac{a^2 + ab + b^2}{3}$$

□

Beispiel 3.3.9 Boltzmann-Verteilung - für Fortgeschrittene

Wir bezeichnen mit V den Geschwindigkeitsbetrag eines Gasmoleküls. Gemäss der kinetischen Gastheorie ist der Betrag der Geschwindigkeit eines Gasmoleküls zufällig und folgt der Wahrscheinlichkeitsverteilung mit der Wahrscheinlichkeitsdichte

$$f_V(v) = 4\pi \left(\frac{m}{2\pi k_B T} \right)^{3/2} v^2 \exp \left(-\frac{mv^2}{2k_B T} \right)$$

wobei k_B die Boltzmann-Konstante und T die Temperatur in Kelvin, m die Masse des Gasmoleküls bezeichnet. Aufgrund dieser Wahrscheinlichkeitsdichte können wir den Erwartungswert des Geschwindigkeitsbetrags $E(V)$ eines Gasmoleküls in einem Gas bestimmen. Wir interessieren uns aber für die mittlere kinetische Energie $Y = \frac{1}{2}mV^2$.

Wir könnten nun wiederum zuerst die Dichte von Y , also $f_Y(Y)$ zu bestimmen versuchen und dann aufgrund von $f_Y(y)$ den Erwartungswert der kinetischen Energie $E(Y)$ berechnen. Die Berechnung des Erwartungswertes der kinetischen Energie gestaltet sich aber analog zum vorhergehenden Beispiel direkter (die Bestimmung der Stammfunktion ist allerdings nicht so einfach):

$$E(Y) = \int_0^{\infty} \frac{1}{2} m v^2 f_V(v) dv = 4\pi \left(\frac{m}{2\pi k_B T} \right)^{3/2} \int_0^{\infty} v^2 \exp \left(-\frac{mv^2}{2k_B T} \right) dv = \frac{3}{2} k_B T$$

□

¹Dies kann in einer nicht besonders aufwendigen Rechnung bewiesen werden, siehe Anhang.

Allgemeine Transformationen von Zufallsvariablen

Wir betrachten den Fall einer *allgemeinen Transformation* der Zufallsvariablen X

$$Y = g(X)$$

Die kumulative Verteilungsfunktion und die Dichte von Y sind durch die Verteilungsfunktion und die Dichte von X bestimmt. Für den Erwartungswert gilt stets die folgende Formel

$$E(Y) = E(g(X)) = \int_{-\infty}^{\infty} g(x)f_X(x) dx$$

3.4. Funktionen von mehreren Zufallsvariablen

Im letzten Unterkapitel haben wir untersucht, wie die Funktion *einer* Zufallsvariable verteilt ist. In den meisten Anwendungen hat man es aber nicht mit einer, sondern mit *mehreren* Zufallsvariablen zu tun. Üblicherweise misst man die *gleiche* Grösse mehrmals; man hat zum Beispiel mehrere Individuen, oder man wiederholt die Messungen. In diesem Abschnitt untersuchen wir, wie eine Funktion mehrerer Zufallsvariablen verteilt ist.

Die Messungen x_1, x_2, \dots, x_n fassen wir als Realisierungen der Zufallsvariablen

$$X_1, \dots, X_n$$

auf. Diese Auffassung, dass X_i die i -te Wiederholung von unserem Zufallsexperiment ist, ist oft bequemer als die Interpretation, dass die Messungen n unabhängige Realisierungen einer Zufallsvariablen X sind.

Beispiel 3.4.1

Wir machen 20 Messungen der Wasserverschmutzung in einem See. Wir haben also Messungen

$$x_1, x_2, \dots, x_{20}$$

die Realisierungen der Zufallsvariablen

$$X_1, X_2, \dots, X_{20}$$

darstellen. Wir gehen davon aus, dass diese 20 Zufallsvariablen Wahrscheinlichkeitsverteilungen haben, die gleich sind, da die Wasserproben alle aus demselben See stammen und mit der identischen Methode gemessen werden.

Kapitel 3. Modelle für Messdaten

Uns interessiert nun der *Durchschnitt* dieser Messungen und die Verteilung der zugehörigen Zufallsvariablen. Dazu benötigen wir eine Theorie für Funktionen von mehreren Zufallsvariablen.

□

Funktionen der Messwerte x_1, x_2, \dots, x_n haben die Form:

$$y = g(x_1, \dots, x_n)$$

Diese Funktion hat als Input n unabhängige Variablen und eine reelle Zahl als Output. Wenn x_1, x_2, \dots, x_n Realisierungen der Zufallsvariablen X_1, \dots, X_n sind, dann ist y eine Realisierung der Zufallsvariablen

$$Y = g(X_1, \dots, X_n)$$

Wir betrachten hier vor allem die Spezialfälle *Summe*

$$g(X_1, \dots, X_n) = S_n = X_1 + \dots + X_n = \sum_{i=1}^n X_i$$

und *arithmetisches Mittel*

$$g(X_1, \dots, X_n) = \bar{X}_n = \frac{1}{n}(X_1 + X_2 + \dots + X_n) = \frac{1}{n} \sum_{i=1}^n X_i = \frac{1}{n} S_n$$

Das arithmetische Mittel der Daten \bar{x}_n ist also eine Realisierung der Zufallsvariablen \bar{X}_n .

Wir sind an der Verteilung der Zufallsvariablen \bar{X}_n interessiert: Die Kenntnis dieser Verteilung wird uns erlauben, Statistik aufgrund von arithmetischen Mitteln von Daten zu machen.

3.4.1. Unabhängigkeit und i.i.d. Annahme

Oft treffen wir die Annahme, dass die Zufallsvariablen X_1, \dots, X_n *unabhängig* voneinander sind. Anschaulich heisst das, es gibt keine gemeinsamen Faktoren, die den Ausgang der verschiedenen Messungen beeinflussen und keine „carry over“ Phänomene von einer Messung zur nächsten. Das heisst nichts anderes, dass eine Messung keinen Einfluss hat auf das Resultat der nachfolgenden Messungen.

Wenn die Zufallsvariablen X_1, \dots, X_n unabhängig sind und alle *dieselbe* Verteilung haben, dann schreiben wir das kurz als

$$X_1, \dots, X_n \text{ i.i.d.}$$

Die Abkürzung i.i.d. steht für:

independent, identically distributed

Wir werden meistens mit dieser i.i.d. Annahme arbeiten, da wir oft von n Durchführungen des gleichen Experimentes ausgehen. Welche Verteilung die X_i 's haben, lassen wir offen. Oft handelt es sich um eine Normalverteilung, dies muss aber *nicht* so sein. Die Unabhängigkeit spielt eine Rolle bei den *Regeln für Erwartungswerte und Varianzen von Summen*. Die Beziehung

$$E(X_1 + X_2) = E(X_1) + E(X_2)$$

gilt immer,

$$\text{Var}(X_1 + X_2) = \text{Var}(X_1) + \text{Var}(X_2)$$

jedoch nur, wenn X_1 und X_2 unabhängig sind.

3.4.2. Kennzahlen von S_n und \bar{X}_n

Beispiel 3.4.2 Relative Häufigkeit von fairen Münzwürfen

Wirft man eine faire Münze bloss ein paar wenige Male, so ist das Verhältnis von Würfen mit Kopf zur Anzahl Würfe mit Zahl selten genau $1/2$. Es ist aber eine vernünftige Annahme, dass dieses Verhältnis für sehr viele Würfe mit einer fairen Münze von Würfen *etwa* $1/2$ ergibt.

Der südafrikanische Mathematiker John Kerich hat diese Vermutung als Kriegsgefangener während des zweiten Weltkrieges getestet. Er hat 10'000 mal eine Münze geworfen und beobachtete dabei 5067 mal Kopf. Die aufeinanderfolgenden Münzwürfe sind modelliert als unabhängige Zufallsvariablen X_i . Die Zufallsvariable nimmt entweder den Wert 0 oder 1 an, je nachdem, ob der i -te Wurf Zahl oder Kopf ergibt. Die relative Häufigkeit von Kopf in n Versuchen entspricht dann gerade dem Durchschnitt

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$$

In John Kerichs Zufallsexperiment ergab sich für \bar{x}_{10000} also der Wert 0.5067

$$\bar{x}_{10000} = 0.5067$$

Hätte er die Münze bloss $n = 10$ geworfen, wäre ein Wert von $\bar{x}_{10} = 0.7$ oder ein Wert von $\bar{x}_{10} = 0.3$ nicht sehr überraschend gewesen, selbst wenn die Münze fair ist. Allerdings würde man die Fairness der Münze stark anzweifeln, wenn von 10'000 Würfen bloss 3000 mal Kopf geworfen würde.

□

Kapitel 3. Modelle für Messdaten

Wir nehmen in diesem Abschnitt an, dass

$$X_1, \dots, X_n \stackrel{\text{i.i.d.}}{\sim} \text{kumulative Verteilungsfkt. } F$$

Wegen dem zweiten „i“ in i.i.d. hat jedes X_i dieselbe Verteilung und dieselben Kennzahlen:

$$E(X_i) = \mu \quad \text{und} \quad \text{Var}(X_i) = \sigma_X^2$$

Die Kennzahlen von S_n und \bar{X}_n folgen dann aus den allgemeinen Regeln für Erwartungswert und Varianz von Summen:

Kennzahlen von $S_n = X_1 + X_2 + \dots + X_n$

$$E(S_n) = n\mu, \quad \text{Var}(S_n) = n \text{Var}(X), \quad \sigma(S_n) = \sqrt{n}\sigma_X$$

Kennzahlen von $\bar{X}_n = \frac{X_1+X_2+\dots+X_n}{n}$

$$E(\bar{X}_n) = \mu, \quad \text{Var}(\bar{X}_n) = \frac{\sigma_X^2}{n}, \quad \sigma(\bar{X}_n) = \frac{\sigma_X}{\sqrt{n}}$$

Die Standardabweichung von \bar{X}_n heisst auch der *Standardfehler* des arithmetischen Mittels.

Begründung: Für $E(S_n)$ gilt:

$$E(S_n) = E(X_1 + X_2 + \dots + X_n) = E(X_1) + X_2 + \dots + E(X_n) = \mu + \mu + \dots + \mu = n\mu$$

Die anderen Beziehungen werden auf ähnliche Art begründet.

Die Standardabweichung der Summe wächst also mit wachsendem n , aber langsamer als die Anzahl Beobachtungen n .

Der Erwartungswert von \bar{X}_n ist also gleich demjenigen einer einzelnen Zufallsvariablen X_i , die *Streuung nimmt jedoch ab mit wachsendem n* .

Gesetz der grossen Zahlen

Für $n \rightarrow \infty$ geht die Streuung gegen null. Es gilt das *Gesetz der grossen Zahlen*: Falls X_1, \dots, X_n i.i.d., dann

$$\bar{X}_n \longrightarrow \mu \quad \text{für } n \rightarrow \infty$$

Standardfehler

Die Standardabweichung des arithmetischen Mittels (*Standardfehler*) ist jedoch nicht proportional zu $1/n$, sondern nimmt nur ab mit dem Faktor $1/\sqrt{n}$

$$\sigma_{\bar{X}_n} = \frac{1}{\sqrt{n}} \sigma_X$$

Um den *Standardfehler* zu halbieren, braucht man also *viermal* so viele Beobachtungen. Dies nennt man auch das \sqrt{n} -Gesetz.

Beispiel 3.4.3

Im Falle einer fairen Münze können wir X_i auffassen als

$$X_i \sim \text{Bernoulli}(\pi = 0.5)$$

mit Erwartungswert

$$E(X_i) = \pi = \frac{1}{2}$$

mit Varianz

$$\text{Var}(X_i) = \pi(1 - \pi) = \frac{1}{4},$$

und Standardabweichung

$$\sigma_{X_i} = \sigma_X = \sqrt{\pi(1 - \pi)} = \frac{1}{2}$$

Im Falle von John Kerichs Experiment ergibt sich für die Kennzahl S_n mit $n = 10000$

$$\begin{aligned} E(S_{n=10000}) &= n E(X_i) = 5000 \\ \text{Var}(S_{n=10000}) &= n \text{Var}(X_i) = 10'000 \cdot \frac{1}{4} = 2500 \\ \sigma(S_n) &= 50 \end{aligned}$$

Die Kennzahlen von der relativen Häufigkeit berechnen sich zu:

$$\begin{aligned} E(\bar{X}_{n=10000}) &= E(X_i) = 0.5 \\ \text{Var}(\bar{X}_{n=10000}) &= \frac{1}{n} \text{Var}(\bar{X}_i) = \frac{1}{10'000} \cdot \frac{1}{4} = 0.000025 \\ \sigma(\bar{X}_n) &= \frac{\sigma_X}{\sqrt{n}} = \frac{0.5}{100} = 0.005 \end{aligned}$$

Die von John Kerich beobachtete relative Häufigkeit von Kopf ist 0.5067 und ist somit etwas ausserhalb von einer Standardabweichung vom erwarteten Mittelwert oder vom Standardfehler von 0.005.

Würde John Kerich sein Experiment mit $n = 10'000$ sehr viele Male wiederholen, so würde er für die Standardabweichung des Mittelwertes also etwa 0.005 erwarten.

□

3.4.3. Verteilungen von S_n und \bar{X}_n

Die Verteilungen von S_n und \bar{X}_n sind im allgemeinen schwierig anzugeben. Es gibt folgende Ausnahmen, unter der Annahme, dass die X_1, \dots, X_n i.i.d. sind:

1. Wenn $X_i \in \{0, 1\}$, dann ist

$$S_n \sim \text{Bin}(n, \pi) \quad \text{mit} \quad \pi = P(X_i = 1)$$

2. Wenn $X_i \sim \text{Pois}(\lambda)$, dann ist

$$S_n \sim \text{Pois}(n\lambda)$$

3. Wenn $X_i \sim \mathcal{N}(\mu, \sigma^2)$, dann ist

$$S_n \sim \mathcal{N}(n\mu, n\sigma^2) \quad \text{und} \quad \bar{X}_n \sim \mathcal{N}\left(\mu, \frac{\sigma_X^2}{n}\right)$$

Falls die einzelnen X_i 's nicht normalverteilt sind, so gilt erstaunlicherweise die unter 3. aufgeführte Verteilungsformel immer noch approximativ. Dies liefert der berühmte Zentrale Grenzwertsatz².

Zentraler Grenzwertsatz

Falls X_1, \dots, X_n i.i.d. mit irgendeiner Verteilung mit Erwartungswert μ und Varianz σ^2 , dann gilt

$$S_n \approx \mathcal{N}(n\mu, n\sigma_X^2)$$

$$\bar{X}_n \approx \mathcal{N}(\mu, \sigma_X^2/n)$$

wobei die Approximation im allgemeinen besser wird mit grösserem n . Überdies ist auch die Approximation besser, je näher die Verteilung von X_i bei der Normalverteilung $\mathcal{N}(\mu, \sigma_X^2)$ ist.

Selbst wenn wir die Verteilung der X_i nicht kennen, so haben wir eine Ahnung über die approximative Verteilung von S_n und X_n . Der Zentrale Grenzwertsatz (ZGWS) ist mitunter ein Grund für die Wichtigkeit der Normalverteilung.

²Einen Beweis für diesen sehr bedeutenden Satz finden Sie in Kapitel ?? des Anhangs.

Beispiel 3.4.4

Wir wollen hier den Zentralen Grenzwertsatz noch an einem Beispiel veranschaulichen. Dabei untersuchen wir das Verhalten von \bar{X}_n . Wir haben eine Ergebnismenge

$$\Omega = \{0, 10, 11\}$$

aus der wir eine Zahl ziehen. Die Zufallsvariable X gibt den Wert der gezogenen Zahl an. Zudem gilt

$$P(X = 0) = P(X = 10) = P(X = 11) = \frac{1}{3}$$

Damit gilt für den Erwartungswert von X : ([zu R](#))

$$E(X) = \frac{1}{3} \cdot 0 + \frac{1}{3} \cdot 10 + \frac{1}{3} \cdot 11 = 7$$

```
import numpy as np
werte = np.array([0,10,11])
ew = np.sum(werte*1/3)
ew

## 7.0
```

und für die Varianz gilt dann ([zu R](#))

$$\text{Var}(X) = \frac{1}{3} \cdot (0 - 7)^2 + \frac{1}{3} \cdot (10 - 7)^2 + \frac{1}{3} \cdot (11 - 7)^2 = 24.6667$$

```
var_X = np.sum((werte-ew)**2*1/3)
var_X

## 24.666666666666664
```

mit der Standardabweichung ([zu R](#))

$$\sigma_X = \sqrt{\text{Var}(X)} = 4.9666$$

```
sd_X = np.sqrt(var_X)

## 4.96655480858378
```

Kapitel 3. Modelle für Messdaten

Von nun an soll ein Versuch aus 10 Ziehungen bestehen. An sich müssten wir vielmehr Ziehungen machen, was wir später auch tun werden, aber bei 10 Ziehungen „sieht“ man bereits, was passiert.

Die Behauptung des Zentralen Grenzwertsatzes ist nun, dass sich der Durchschnitt \bar{X}_n immer mehr

$$\mathcal{N}\left(7, \frac{24.6667}{n}\right)$$

annähert. Dies wollen wir durch Simulationen untersuchen.

Wir beginnen mit einem Versuch (10 Ziehungen). ([zu R](#))

```
sim = np.random.choice(werte, size=10, replace = True)
sim

## [ 0  0  0 11 11  0 11 10 11 11]
```

Diese Daten stellen wir noch als Histogramm dar (siehe Abbildung 3.17).

```
# range: macht die Bins von 0 bis 12 mit Abstand 1
# edgecolor: zeichnet die Balkenumrandungen ein mit
# facecolor kann noch die Farbe der Balken verändert
# werden

plt.hist(sim, bins = range(0, 13, 1), edgecolor = "black")
```

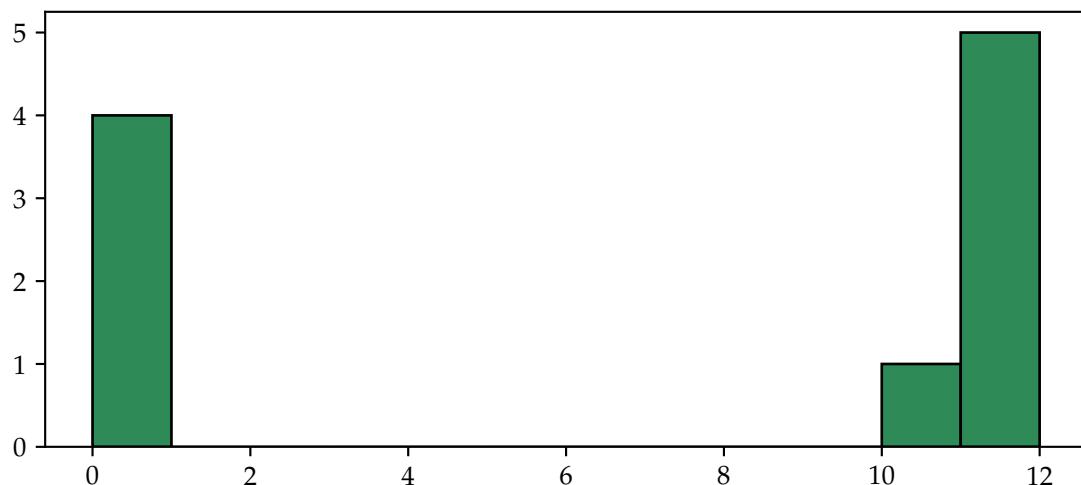


Abbildung 3.17.: Histogramm mit 10 Ziehungen

Kapitel 3. Modelle für Messdaten

Bei jedem Versuch erhalten wir ein anderes Histogramm. In Abbildung 3.18 sind 4 aufgeführt. (zu R)

```
# subplot: Macht 4 subplots mit 2 Reihen und 2 Spalten
# i: der i-te subplot

for i in range(1,5):
    plt.subplot(2,2,i)
    sim = np.random.choice(werte, size=10, replace = True)
    plt.hist(sim, bins=range(0,13,1), edgecolor="black")
```

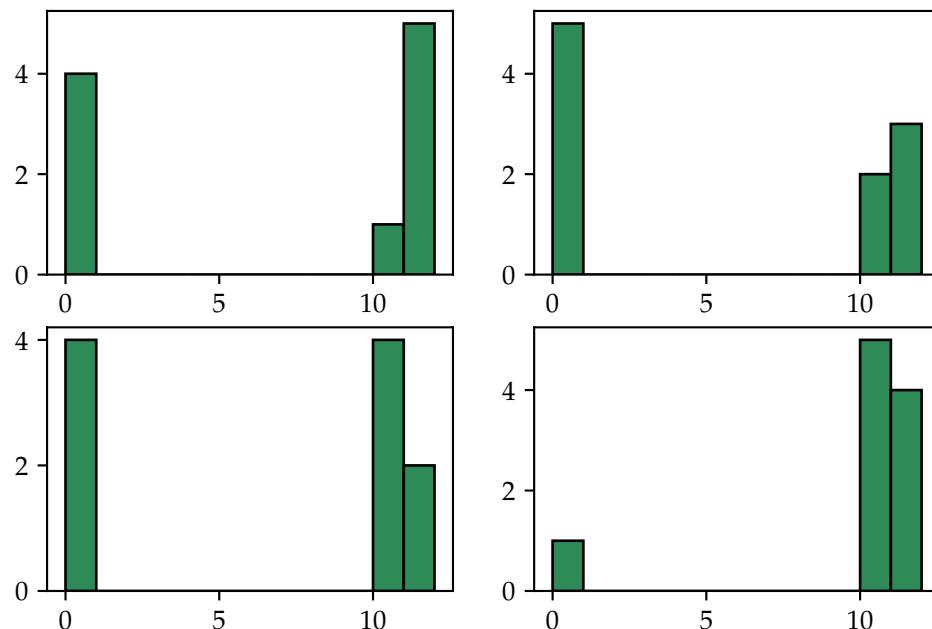


Abbildung 3.18.: 4 Histogramme mit 10 Ziehungen

Offensichtlich haben wir es hier mit keiner Normalverteilung zu tun. In diesem Versuch kamen nur die Zahlen 0, 10, 11 vor.

Nun können wir zwei solche Versuche (je 10 Ziehungen) hintereinander ausführen und den *Durchschnitt* aus den beiden Versuchen berechnen. (zu R)

```
sim_1 = np.random.choice(werte, size=10, replace = True)
sim_1

sim_2 = np.random.choice(werte, size=10, replace = True)
sim_2
```

Kapitel 3. Modelle für Messdaten

```
sim_mean_2 = (sim_1+sim_2)/2
sim_mean_2

## [11  0 11  0 11 11  0 10  0 11]
## [11  0 11 11 11 11 10 10 11 10]
## [11.   0.  11.   5.5 11.  11.   5.  10.   5.5 10.5]

# linspace: Spaltenbreiten 0.5 ueber das Intervall

plt.hist(sim_mean_2, bins = np.linspace(0, 11.5, 24), edgecolor = "black")
```

Diese Werte sind im Histogramm in Abbildung 3.19 dargestellt.

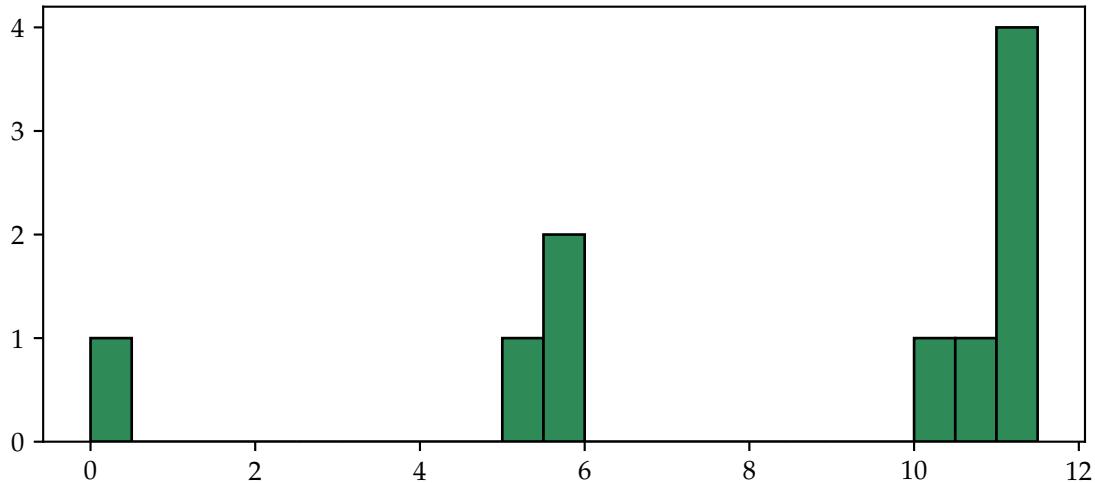


Abbildung 3.19.: Histogramm vom Durchschnitt von zwei Versuchen mit 10 Ziehungen

Neben den Zahlen 0, 10, 11 können nun auch noch die Zahlen 5, 5.5 und 10.5 vorkommen. Es zeigt sich auch in diesem Fall, dass das Histogramm mit jedem Versuch anders ausschaut (siehe Abbildung 3.20). (zu R)

Obwohl jeder Versuch anders ausschaut, zeichnen sich doch bestimmte Tendenzen ab. Die 0 ist zum Beispiel weniger oft vertreten, da eine doppelte 0 nur mit Wahrscheinlichkeit $\frac{1}{9}$ vorkommt.

Wir können dasselbe für 3 Versuche wiederholen und den Durchschnitt nehmen (siehe Abbildung 3.21). (zu R)

Kapitel 3. Modelle für Messdaten

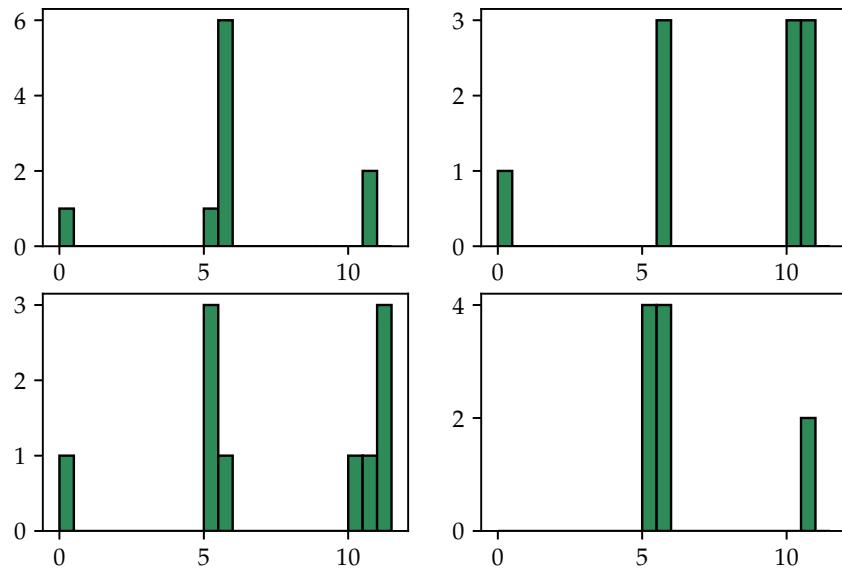


Abbildung 3.20.: Histogramm vom Durchschnitt von drei Versuchen mit 10 Ziehungen

```

sim_1 = np.random.choice(werte, size=10, replace = True)
sim_1

sim_2 = np.random.choice(werte, size=10, replace = True)
sim_2
sim_3 = np.random.choice(werte, size=10, replace = True)
sim_3

sim_mean_3 = (sim_1+sim_2+sim_3)/3
sim_mean_3

## [1] 11  0 11  0 11 11  0 10  0 11
## [1] 11  0 11 11 11 10 10 11 10
## [1]  0 10 10 10 11  0  0  0 10 10
## [1] 7.333 3.333 10.667 7.     11.     7.333 3.333 6.667 7.     10.333

```

```
# linspace: Spaltenbreiten 1/3 ueber das Intervall
```

```
plt.hist(sim_mean_3, bins = np.linspace(0, 11 + 1/3, 35),
         edgecolor = "black")
```

Wir führen wieder mehrere Versuche durch (siehe Abbildung 3.22). (zu R)

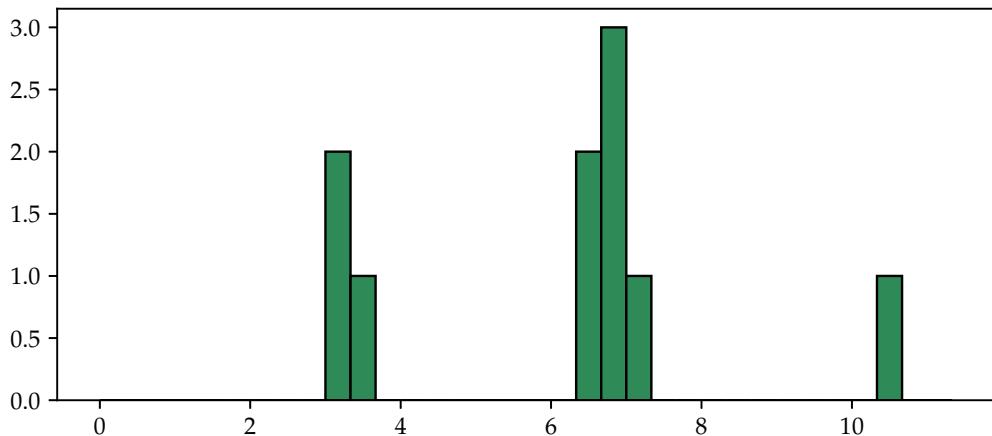


Abbildung 3.21.: Histogramm vom Durchschnitt von drei Versuchen mit 10 Ziehungen

```
for i in range(1,5):
    plt.subplot(2,2,i)
    sim_1 = np.random.choice(werte, size=10, replace = True)
    sim_2 = np.random.choice(werte, size=10, replace = True)
    sim_3 = np.random.choice(werte, size=10, replace = True)
    sim_mean_3 = (sim_1+sim_2+sim_3)/3
    plt.hist(sim_mean_3,bins=np.linspace(0,11+1/3,35),edgecolor="black")
```

Es zeigt sich ebenfalls eine Tendenz gegen den Erwartungswert 7.

Wir stellen also fest, dass es beim *Durchschnitt* immer mehr Werte gibt, die sich um den Erwartungswert 7 häufen. Warum ist dies so? Die Zahl 0 im Durchschnitt kommt praktisch nicht mehr vor, da die Wahrscheinlichkeit, dass 3 mal an derselben Stelle eine 0 vorkommt, nur noch $1/27$ ist. Dasselbe ist für die Zahl 11 der Fall.

Für viele Versuche ist dieses Vorgehen ungeeignet und nicht sehr elegant (obwohl man „sieht“, was passiert).

Wir wollen nun 16, 64, 256 und 1024 solche Versuche durchführen, aber mit jeweils 1000 Ziehungen. Von denen nehmen wir jeweils den Durchschnitt wie in den Beispielen vorher. Das Resultat ist in der Abbildung 3.23 dargestellt. ([zu R](#))

```
# Code nur fuer n=16

# 1. Subplot mit 2 Reihen und zwei Spalten
plt.subplot(2,2,1)

n = 16
```

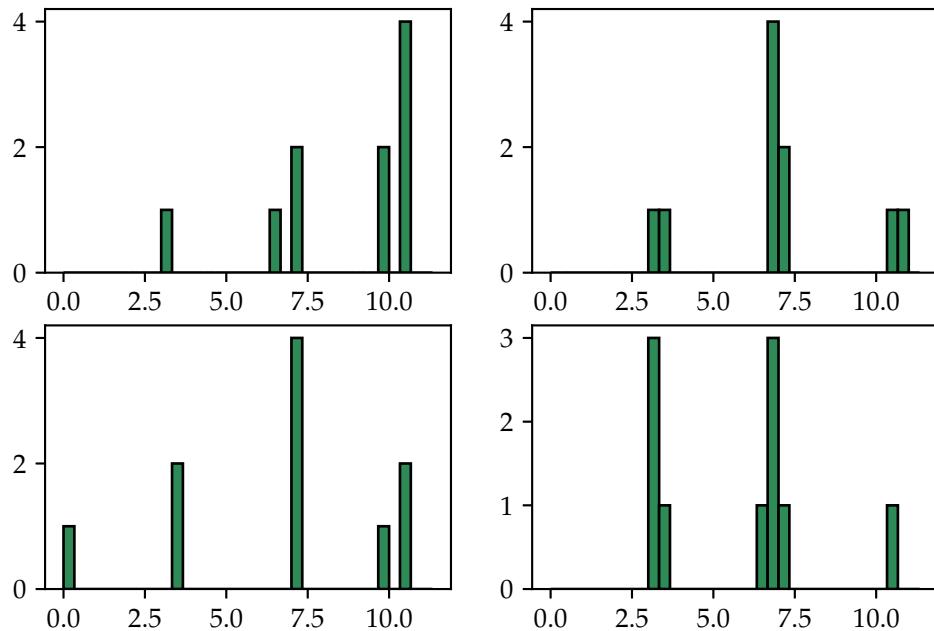


Abbildung 3.22.: Histogramme vom Mittelwert von drei Versuchen von jeweils 10 Ziehungen

```
# es werden 1000*n zufällige Zahlen aus werte gezogen
sim = np.random.choice(werte, 1000*n, replace = True)

# reshape: Vektor wird in eine n x 1000-Matrix umgewandelt (1000 Spalten)
sim = np.reshape(sim, (n,1000))

# Mittelwert aller Spalten (axis=0)
sim_mean = np.mean(sim, axis=0)

# Festlegung des x-Bereiches (damit die Skizzen miteinander vergleichbar sind)
plt.xlim(2,12)

plt.title("n=16")
plt.hist(sim_mean, edgecolor="black", bins="auto")
```

Was (bei genauerem Hinschauen) auffällt, ist :

- die Werte häufen sich um den Erwartungswert 7
- die Standardabweichung des Mittelwertes (also der Standardfehler) wird kleiner, und zwar halbiert sie sich etwa beim Vervierfachen der Anzahl Versuche
- die Histogramme scheinen einer Normalverteilung zu folgen

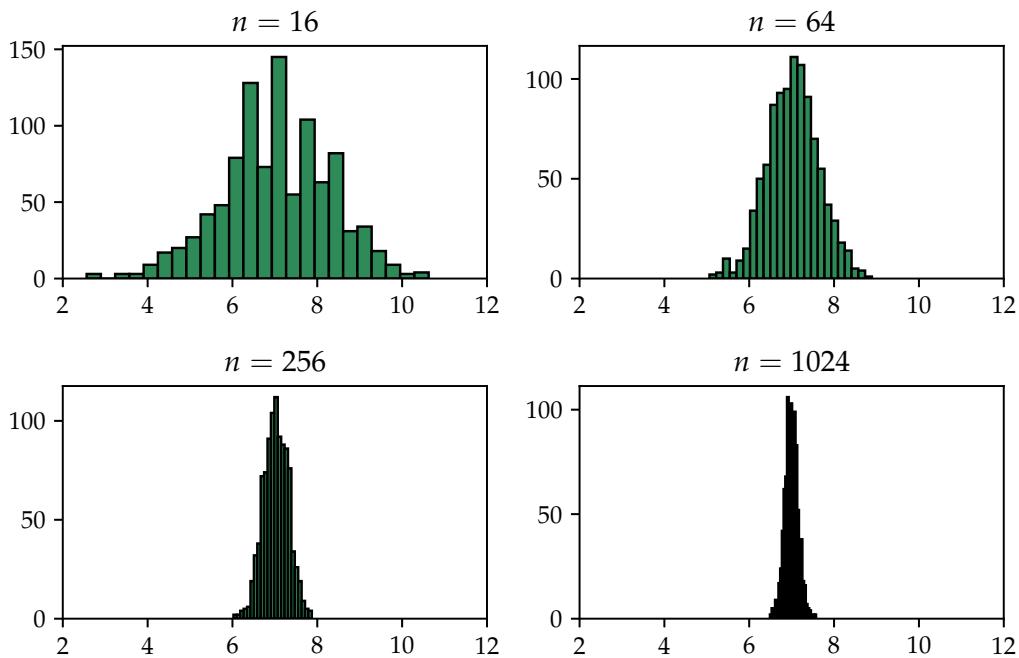


Abbildung 3.23.: 4 Histogramme vom Durchschnitt von 16, 64, 256, 1024 Versuchen mit 1000 Ziehungen

Den dritten Punkt wollen wir noch genauer untersuchen, indem wir die jeweiligen Dichtekurven für

$$\mathcal{N} \left(7, \frac{24.6667}{n} \right)$$

einzeichnen (siehe Abbildung 3.24). (zu R)

```
from scipy.stats import norm

plt.subplot(2, 2, 1)

n = 16

sim = np.reshape(np.random.choice(werte, 1000*n, replace = True), (n, 1000))

sim_mean = np.mean(sim, axis=0)

plt.xlim(2,12)
plt.title("n=16")

# Festlegung der x-Werte: 500 Werte zwischen 2 und 12
x = np.linspace(2,12,500)
```

Kapitel 3. Modelle für Messdaten

```
# Berechnung der zugehörigen Funktionswerte
y = norm.pdf(x, loc=7, scale=sd_X/np.sqrt(n))

plt.plot(x,y,color="orange")

#Normierung, dass die Fläche des Histogrammes 1 ist
plt.hist(sim_mean, edgecolor="black", bins="auto", normed=True)
```

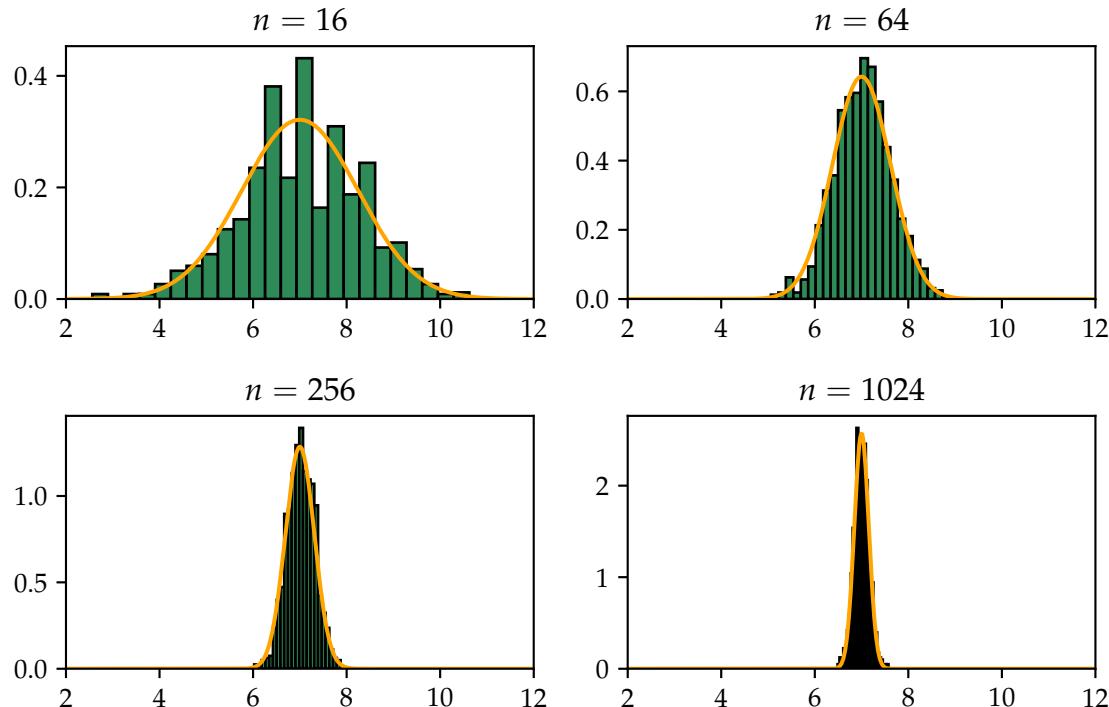


Abbildung 3.24.: 4 Histogramme vom Durchschnitt von 16, 64, 256, 1024 Versuchen mit 1000 Ziehungen mit Dichtekurven

Es fällt auf, dass die Dichtekurven für grössere n immer besser zu den Histogrammen passen.

Es sei nochmals erwähnt, dass wir mit einer Verteilung begonnen haben, die *nichts* mit einer Normalverteilung zu tun hat. Aber die Verteilung der *Mittelwerte* \bar{X}_n (oder auch die Summen) nähert sich mit wachsendem n einer Normalverteilung an.

□

Wir haben schon bei der Binomialverteilung gesehen, dass diese für grosse n „glockenförmig“ aussieht. Dasselbe gilt für die Poissonverteilung für grösser werdendes λ .

Kapitel 3. Modelle für Messdaten

Man kann daher die Normalverteilung verwenden, um die Binomialverteilung mit grossem n zu approximieren (denn die Binomialverteilung ist eine i.i.d. Summe von Bernoulliverteilungen). Man spricht dann von der sogenannten *Normalapproximation* der Binomialverteilung.

Wenn

$$X \sim \text{Bin}(n, \pi)$$

dann haben wir

$$\mathbb{E}(X) = n\pi \quad \text{und} \quad \text{Var}(X) = n\pi(1 - \pi)$$

Für grosse n können wir also X gemäss dem ZGWS approximativ als Normalverteilung mit Erwartungswert $\mathbb{E}(X) = n\pi$ und Varianz $\sigma^2 = n\pi(1 - \pi)$ behandeln. D.h. es gilt dann

$$X \sim \text{Bin}(n, \pi) \approx \mathcal{N}(n\pi, n\pi(1 - \pi))$$

Beispiel 3.4.5

Wie gross ist die Wahrscheinlichkeit, dass bei 10 000 Würfen mit einer fairen Münze maximal 5100 mal Kopf erscheint?

Die Anzahl Würfe, bei denen Kopf erscheint, ist $\text{Bin}(10\,000, 0.5)$ -verteilt. Diese Verteilung approximieren wir mit einer Normalverteilung, d.h.,

$$X \sim \mathcal{N}(\mu, \sigma^2)$$

mit

$$\mu = n \cdot \pi = 10\,000 \cdot 0.5 = 5000 \quad \text{und} \quad \sigma^2 = n \cdot \pi(1 - \pi) = 10000 \cdot 0.5 \cdot (1 - 0.5) = 2500$$

Und wir erhalten

$$X \sim \text{Bin}(10\,000, 0.5) \approx \mathcal{N}(5000, 2500)$$

Von Interesse ist

$$P(X \leq 5100)$$

Mit **Python** erfolgt die Berechnung folgendermassen ([zu R](#))

```
from scipy.stats import uniform, expon, norm
import math

norm.cdf(x=5100, loc=5000, scale = math.sqrt(2500))

## 0.9772498680518208
```

Vergleichen wir das vorhergehende Resultat mit der Verteilung

$$X \sim \text{Bin}(10\,000, 0.5)$$

dann erhalten wir (zu R)

$$P(X \leq 5100) \approx 0.98$$

```
from scipy.stats import uniform, expon, norm, binom
import math

binom.cdf(k=5100, n=10000, p=0.5)

## 0.9777871004771368
```

Die Übereinstimmung ist also sehr gut, bis auf die 3. Stelle nach dem Komma.

□

Beispiel 3.4.6

Wir ziehen $n = 10$ Zufallszahlen X_i . Die zehn Zufallsvariablen sind unabhängig und es gilt für jedes i

$$X_i \sim \text{Uniform}([0, 1])$$

Wie gross ist die Wahrscheinlichkeit, dass die Summe der Zufallszahlen

$$S_{10} = \sum_{i=1}^{10} X_i$$

grösser als sechs ist?

D. h., wir suchen

$$P(S_{10} > 6)$$

Aus Abschnitt 3.2.1 wissen wir, wie man Erwartungswert und Varianz von jedem X_i berechnet:

$$\mathbb{E}(X_i) = \frac{1+0}{2} = 0.5 \quad \text{und} \quad \text{Var}(X_i) = \frac{(1-0)^2}{12} = \frac{1}{12}$$

Aus dem Zentralen Grenzwertsatz folgt:

$$S_n \approx \mathcal{N}(n \mathbb{E}(X_i), n \text{Var}(X_i)) = \mathcal{N}\left(5, \frac{10}{12}\right) = \mathcal{N}(5, 0.83)$$

Damit kommen wir zu folgender Lösung:

$$P(S_n > 6) = 1 - P(S_n \leq 6) = 1 - 0.86 = 0.14$$

□

Für eine exakte Formulierung des Zentralen Grenzwertsatzes betrachtet man die standardisierte Zufallsvariable

$$Z_n = \frac{\sqrt{n}(\bar{X}_n - \mu)}{\sigma_X}$$

Diese ist ungefähr $\mathcal{N}(0, 1)$ verteilt, was bedeutet, dass für alle x gilt

$$\lim_{n \rightarrow \infty} P(Z_n \leq x) = \Phi(x)$$

Der Zentrale Grenzwertsatz gilt auch für diskrete Zufallsvariablen X_i .

3.4.4. Fehlerrechnung bei Messreihen

Messungen physikalischer Größen sind grundsätzlich fehlerbehaftet, d. h., man erhält Messwerte, die vom wahren Wert mehr oder weniger abweichen. Je nach Ursache der Abweichung unterscheidet man zwischen *systematischen* und *zufälligen* Fehlern. Systematische Fehler röhren von der Unvollkommenheit der Messgeräte her, wie zum Beispiel Funktionsfehler und Eichfehler, oder von der Unvollkommenheit der Messverfahren.

Beispiel 3.4.7 Systematische Fehler

Beispiele von systematischen Messfehlern:

1. Bei Kurzschluss der Eingänge zeigt ein Voltmeter nicht mehr 0 V an (Nullpunktsfehler).
2. Durch den Innenwiderstand eines Voltmeters sind gemessene Strom- oder Spannungswerte stets zu klein.
3. Bei einem Pendelversuch wird durch die Luft- und Lagerreibung die Schwingung gedämpft, wodurch die Frequenz der Schwingung verringert wird.

□

Systematische Fehler sollten nach Möglichkeit vermieden oder klein gehalten werden. Sie sind jedoch nicht Gegenstand einer Fehlerrechnung.

Zufällige Fehler besitzen beiderlei Vorzeichen (im Gegensatz zu systematischen Fehlern) und entstehen vor allem durch die Naturgesetze selber wie zum Beispiel aufgrund der statistischen Natur von Kernzerfällen, durch Ungeschicklichkeit beim Messen oder durch statistisch schwankende äußere und innere Einflüsse. Beispiele für solche schwankenden Einflüsse sind Schwankungen der Umgebungsparameter wie Druck, Temperatur, Luftfeuchtigkeit oder Schwankungen der Ausgangsparameter.

Kapitel 3. Modelle für Messdaten

Wird die Messung im Rahmen einer Messreihe mehrfach durchgeführt, so streuen die Messwerte um den arithmetischen Mittelwert

$$\bar{x}_n = \frac{1}{n} \sum_{i=1}^n x_i$$

Je grösser die Messreihe ist (also je grösser n), um so näher liegt der arithmetische Mittelwert am wahren Wert und umso kleiner wird der Fehler des Mittelwertes. Der Fehler des Mittelwertes ist durch den *Standardfehler* gegeben

$$s_{\bar{x}_n} = \frac{s_x}{\sqrt{n}}$$

wobei die *empirische Standardabweichung* der Messreihe

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

unabhängig von der Anzahl Datenpunkte ist.

Bemerkungen:

- i. Hier wird s für die Standardabweichung verwendet anstatt σ . Der Unterschied liegt darin, dass s die *empirische Standardabweichung* ist, die aus den Daten mit der Formel oben berechnet wurde. Die theoretische Standardabweichung, die aus einer Wahrscheinlichkeitsverteilung folgt, wird mit σ bezeichnet.

Absoluter und relativer Fehler

Der Standardfehler der Messreihe entspricht dem *absoluten Fehler*, den man im Zusammenhang mit dem arithmetischen Mittelwert folgendermassen angibt:

$$\bar{x}_n \pm s_{\bar{x}_n} = \bar{x}_n \pm \frac{s_x}{\sqrt{n}}$$

Der *relative Fehler* wird folgendermassen angegeben:

$$\bar{x}_n \pm \frac{s_{\bar{x}_n}}{\bar{x}_n}$$

Bemerkungen:

- i. *Darstellung von Messwerten:* Mittelwert und Fehler sollen mit gleich vielen Dezimalstellen geschrieben werden, wobei der Mittelwert genauso viele signifikante Stellen hat wie das ungenaueste Messergebnis in der Messreihe.

ii. *Signifikante Stellen*: Die folgenden Ausdrücke haben zwei signifikante Stellen:

$$0.0012, \quad 0.012, \quad 0.12, \quad 1.2, \quad 12$$

iii. Sowohl zum arithmetischen Mittelwert als auch zum Fehler des Mittelwerts gehörenden Einheiten. Die Einheit ist für beide dieselbe wie für die Messgrösse.

Beispiel 3.4.8 Umlaufzeit von Plattentellern

Umlaufzeit eines Plattentellers mit *absolutem Fehler* des Mittelwertes:

$$T = (1.817 \pm 0.012) \text{ s}$$

Umlaufzeit eines Plattentellers mit *relativem Fehler* des Mittelwertes:

$$T = 1.817 \text{ s} \pm \frac{0.012 \text{ s}}{1.817 \text{ s}} = 1.817 \text{ s} \pm 0.66 \%$$

□

Beispiel 3.4.9 Fallzeiten

Wir haben 100 mal die Fallzeit eines Körpers gemessen. Diese Messungen sind in der Abbildung 3.25 graphisch dargestellt. Der arithmetische Mittelwert der 100 Messungen ergibt

$$\bar{t}_{100} = 4.2 \text{ s}$$

die empirische Standardabweichung

$$s_t = 0.85 \text{ s}$$

Der Fehler, der uns allerdings interessiert, ist der statistische Fehler des Mittelwertes, also der Standardfehler

$$s_{\bar{t}_n} = \frac{s_t}{\sqrt{n}}$$

Der Fehler des Mittelwerts oder Standardfehler ergibt für diese Messreihe 0.085 s. Wir schreiben also für das Ergebnis unserer Messung mit absolutem Fehler

$$T = (4.2 \pm 0.1) \text{ s}$$

und mit relativem Fehler

$$T = 4.2 \text{ s} \pm 2 \%$$

Man beachte, dass im Falle des absoluten Fehlers eine Nachkommastelle angegeben wurde und im Falle des relativen Fehlers zwei signifikante Stellen.

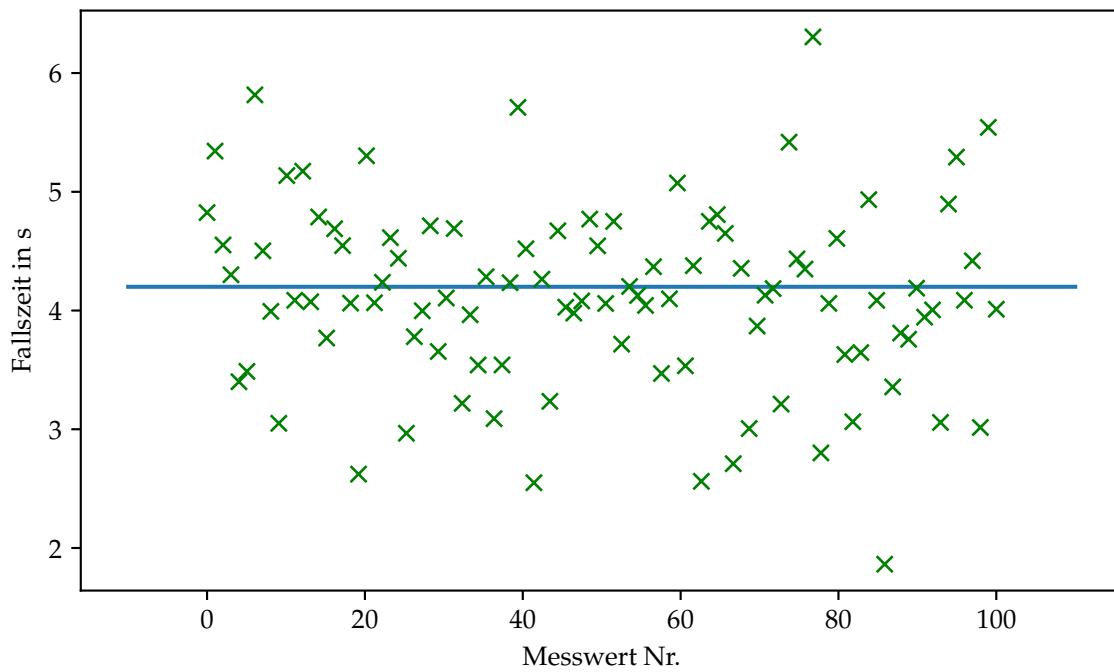


Abbildung 3.25.: 100 Messungen der Fallzeit eines Körpers.

Würde man nur die ersten 10 Zeitmessungen zur Mittelwertbildung heranziehen, erhielt man nach korrekter Rundung folgendes Ergebnis für den arithmetischen Mittelwert: 4.1 s. Die empirische Standardabweichung ergibt allerdings in diesem Fall 0.61 s. Den Fehler des Mittelwerts oder Standardfehler ermitteln wir zu 0.19 s und das Endergebnis schreiben wir im Falle von 10 Datenpunkten mit absolutem Fehler

$$T = (4.1 \pm 0.2) \text{ s}$$

und mit relativem Fehler

$$T = 4.1 \text{ s} \pm 4.7\%$$

Es ist ersichtlich, dass der Fehler des Mittelwertes in diesem Fall grösser ist und somit das Messergebnis insgesamt schlechter zu bewerten ist.

□

Konzeptionelle Lernziele

Sie sollten fähig sein, . . .

- das Konzept der Wahrscheinlichkeitsdichtefunktion und der kumulativen Verteilungsfunktion zu erklären.
- die uniforme Verteilung, die Exponential-Verteilung und die Normalverteilung zu definieren.
- die Normalverteilung zu standardisieren.
- den Erwartungswert, die Varianz und Quantilen für linear transformierte Zufallsvariablen zu berechnen.
- den Erwartungswert und die Varianz von Summen von unabhängigen Zufallsvariablen zu berechnen.
- das Gesetz der grossen Zahlen, das \sqrt{n} -Gesetz und den Zentralen Grenzwertsatz zu erklären.
- den Zentralen Grenzwertsatz auf Summen von Zufallsvariablen anzuwenden.
- den Unterschied zwischen der Standardabweichung einer einzelnen Beobachtung X_i und dem Standardfehler einer Messreihe zu erklären.
- den Standardfehler einer Messreihe zu berechnen.

Computer-Basierte Lernziele

Sie sollten fähig sein . . .

- die Werte der Wahrscheinlichkeitsdichtefunktion von uniform, exponential und normal verteilten Zufallsvariablen zu berechnen, und zwar mit Hilfe von `uniform.pdf()`, `expon.pdf()` und `norm.pdf()` bei gegebenen Parameterwerten.
- die Werte der kumulativen Verteilungsfunktion von uniform, exponential und normal verteilten Zufallsvariablen zu berechnen, und zwar mit Hilfe von `uniform.cdf()`, `expon.cdf()` und `norm.cdf()` bei gegebenen Parameterwerten.
- uniform, exponential und normal verteilte Zufallszahlen zu erzeugen, und zwar mit Hilfe von `uniform.rvs()`, `expon.rvs()` und `norm.rvs()` bei gegebenen Parameterwerten.
- Wahrscheinlichkeiten von uniform, exponential und normal verteilten Zufallsvariablen zu berechnen.

Übersicht Verteilungen

Diskrete Verteilungen				
Verteilung	kum. Verteilungsfunktion	Wahrscheinlichkeitsfunktion	E(X)	Var(X)
Binomialverteilung $\text{Bin}(n, \pi)$	$F(x) = \sum_{i=0}^x \binom{n}{i} \pi^i (1-\pi)^{n-i}$	$P(X = x) = \binom{n}{i} \pi^i (1-p)^{n-i}$	$n\pi$	$n\pi(1-\pi)$
Poisson-Verteilung $\text{Pois}(\lambda)$	$F(x) = \sum_{i=0}^x e^{-\lambda} \frac{\lambda^x}{x!}$	$P(X = x) = e^{-\lambda} \frac{\lambda^x}{x!}$	λ	λ

Stetige Verteilungen				
Verteilung	kum. Verteilungsfunktion	Wahrscheinlichkeitsdichte		
Uniforme Verteilung $\text{Unif}(a, b)$	$F(x) = \begin{cases} 0 & \text{falls } x < a \\ \frac{x-a}{b-a} & \text{falls } a \leq x \leq b \\ 1 & \text{falls } x > b \end{cases}$	$f(x) = \begin{cases} \frac{1}{b-a} & \text{falls } a \leq x \leq b \\ 0 & \text{sonst} \end{cases}$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$
Exponentialverteilung $\text{Exp}(\lambda)$	$F(x) = \begin{cases} 1 - e^{-\lambda x} & \text{falls } x \geq 0 \\ 0 & \text{sonst} \end{cases}$	$f(x) = \begin{cases} \lambda \cdot e^{-\lambda x} & \text{falls } x \geq 0 \\ 0 & \text{sonst} \end{cases}$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$
Normalverteilung $\mathcal{N}(\mu, \sigma^2)$	$F(x) = \int_{-\infty}^x f(y) dy$	$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$	μ	σ^2

Kapitel 4.

Statistik für Messdaten

We must be careful not to
confuse data with the
abstractions we use to analyze
them.

(William James)

4.1. Überprüfen der (Normal-) Verteilungsannahme

In Kapitel 3 sind wir davon ausgegangen, dass wir die Wahrscheinlichkeitsverteilung einer Zufallsvariablen, wie zum Beispiel $\mathcal{N}(3, 2)$, kennen. Aufgrund von dieser Wahrscheinlichkeitsverteilung haben wir dann Kennzahlen und diverse Wahrscheinlichkeiten berechnet. In der Praxis allerdings müssen wir uns oft basierend auf (wenigen) Daten für eine Verteilungsfamilie, wie zum Beispiel die Normalverteilung, entscheiden, mit der wir ein Zufallereignis modellieren wollen.

Nehmen wir also an, dass wir einen Datensatz x_1, \dots, x_n mit n Beobachtungen haben. Die Wahl einer Verteilungsfamilie kann einerseits durch Erfahrung („was sich bisher bewährt hat“) oder aber auch durch physikalische Argumente geschehen. Ob eine Verteilungsfamilie zu einem konkreten Datensatz passt, kann man qualitativ gut mit graphischen Methoden überprüfen. So könnten wir beispielsweise schauen, wie gut das (normierte) Histogramm der Daten zur Dichte unserer Modellverteilung passt, z.B. zu einer bestimmten Normalverteilung. Es zeigt sich aber, dass man Abweichungen besser durch den Vergleich der Quantilen erkennen kann.

Q-Q-Plots sind graphische Darstellungen, um zu überprüfen, wie gut eine Verteilungsfamilie zu einem Datensatz passt. Daneben gibt es auch quantitative Methoden beruhend auf Teststatistiken, um festzustellen, wie gut eine (vermutete) Verteilungsfamilie zu einem Datensatz passt.

4.1.1. Q-Q-Plot

Die Idee des *Q-Q-Plot* (Quantil-Quantil Plot) besteht darin, die empirischen Quantile gegen die theoretischen Quantile der vermuteten Modell-Verteilung zu plotten.

In der folgenden Box wird das allgemeine Vorgehen beschrieben.

QQ-Plot

- (i) Berechne für

$$\alpha_k = \frac{k - 0.5}{n} \quad \text{mit } k = 1, \dots, n$$

also für

$$\alpha_1 = \frac{0.5}{n}, \dots, \alpha_n = \frac{n - 0.5}{n}$$

die theoretischen Quantile der Modell-Verteilung

$$q(\alpha_k) = F^{-1}(\alpha_k)$$

wobei n die Anzahl Datenpunkte bezeichnet.

- (ii) Bestimme die empirischen α_k -Quantile, welche den geordneten Beobachtungen

$$x_{(1)} < x_{(2)} < \dots < x_{(n)}$$

entsprechen. ($x_{(\alpha_k \cdot n + 0.5)} = x_{(k)}$)

- (iii) Zeichne die theoretischen Quantile $q(\alpha_k)$ auf der x -Achse gegen die empirischen Quantile $x_{(k)}$ auf der y -Achse auf.

Wenn die Beobachtungen gemäss der Modell-Verteilung erzeugt wurden, sollten diese Punkte ungefähr auf der Winkelhalbierenden $y = x$ liegen.

Bemerkungen:

- Wir erinnern daran, dass wir mit $x_{(1)}$ die kleinste Beobachtung im Datensatz bezeichnen, mit $x_{(n)}$ die grösste.
- Warum berechnen wir die theoretischen Quantile zu den Werten $\alpha_k = \frac{k-0.5}{n}$?

Nun hatten wir das empirische α -Quantil als die Beobachtung $x_{(\alpha \cdot n + 0.5)}$ definiert, wobei wir den Wert $\alpha \cdot n + 0.5$ runden (α und n sind gegeben).

Betrachten wir nun aber die Beobachtung $x_{(k)}$, also die k -grösste Beobachtung, dann entspricht k dem gerundeten Wert von $(\alpha \cdot n + 0.5)$. Also ist $x_{(k)}$ das α_k -Quantil mit $\alpha_k = \frac{k-0.5}{n}$.

Beispiel 4.1.1 Betondruckfestigkeit

Es wurden Messungen der Betondruckfestigkeit an $n = 20$ verschiedenen Proben durchgeführt. Wir wollen schauen, wie gut die Daten x_k mit einer Normalverteilung beschrieben werden können.

Die Werte sind in der Tabelle 4.1 der Grösse nach aufgeführt (siehe 2. Spalte). Für jeden Messwert k wird nun $\alpha_k = \frac{k-0.5}{n}$ berechnet. Der erste Wert entspricht also dem empirischen 2.5%-Quantil, die $k = 11$ -te Messung entspricht in etwa dem Median und die $k = 16$ -te Messung entspricht dem 75 %-Quantil.

k	$x_{(k)}$	$\alpha_k = (k - 0.5)/n$	q_{α_k} für $\mathcal{N}(32.7, 4.15^2)$	$\Phi^{-1}(\alpha_k)$
1	24.4	0.0250	24.5	-1.96
2	27.6	0.075	26.7	-1.44
3	27.8	0.125	27.9	-1.15
4	27.9	0.175	28.8	-0.935
5	28.5	0.225	29.5	-0.755
6	30.1	0.275	30.2	-0.600
7	30.3	0.325	30.8	-0.453
8	31.7	0.375	31.3	-0.319
9	32.2	0.425	31.9	-0.189
10	32.8	0.475	32.4	-0.0627
11	33.3	0.525	32.9	0.0627
12	33.5	0.575	33.4	0.189
13	34.1	0.625	34.0	0.319
14	34.6	0.675	34.5	0.454
15	35.8	0.725	35.1	0.598
16	35.9	0.775	36.0	0.755
17	36.8	0.825	36.5	0.935
18	37.1	0.875	37.4	1.15
19	39.2	0.925	38.6	1.44
20	39.7	0.975	40.8	1.96

Table 4.1.: 20 Messungen der Betondruckfestigkeit.

Nun vermuten wir, dass diese Messungen normalverteilt sind. Wir schätzen¹ also den Parameter μ der Normalverteilung durch den empirischen Mittelwert und den Parameter σ durch die empirische Standardabweichung und finden

$$\hat{\mu} = 32.7 \quad \text{und} \quad \hat{\sigma} = 4.15$$

¹Dieses plausible Vorgehen für die Parameterschätzung werden wir im nächsten Kapitel 4.2 ausführlich begründen.

Kapitel 4. Statistik für Messdaten

Wir berechnen als nächstes für jedes α_k das entsprechende α_k -Quantil, also $\Phi^{-1}(\alpha_k)$, der Normalverteilung

$$\mathcal{N}(32.7, 4.15^2)$$

Sollte unser Datensatz tatsächlich normalverteilt sein, dann erwarten wir, dass die empirischen Quantilen unseres Datensatzes und die Quantilen der Normalverteilung in etwa gleich gross sind, also auf der Winkelhalbierenden $y = x$ liegen.

Mit **Python** erstellt man einen Q-Q-Plot folgendermassen ([zu R](#))

```
import matplotlib.pyplot as plt
import numpy as np
from pandas import Series
from scipy.stats import norm, probplot

x = Series([24.4, 27.6, 27.8, 27.9, 28.5, 30.1, 30.3, 31.7, 32.2, 32.8, 33.3, 33.5, 34.1, 34.5, 35.2, 35.8, 36.5, 37.2, 37.8, 38.5, 39.2, 40.0])

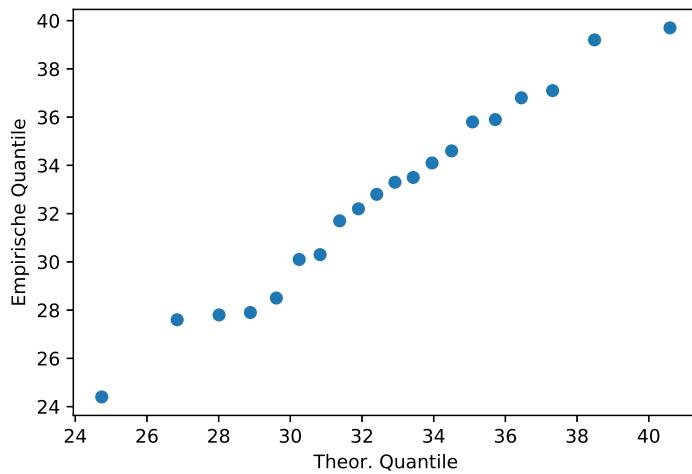
alphak = (np.arange(1, x.size + 1) - 0.5) / x.size

quantile_theor = norm.ppf(q=alphak, loc=x.mean(), scale=x.std())

quantile_empir = np.sort(x)

plt.xlabel("Theor. Quantile")
plt.ylabel("Empirische Quantile")
plt.plot(quantile_theor, quantile_empir, "o")

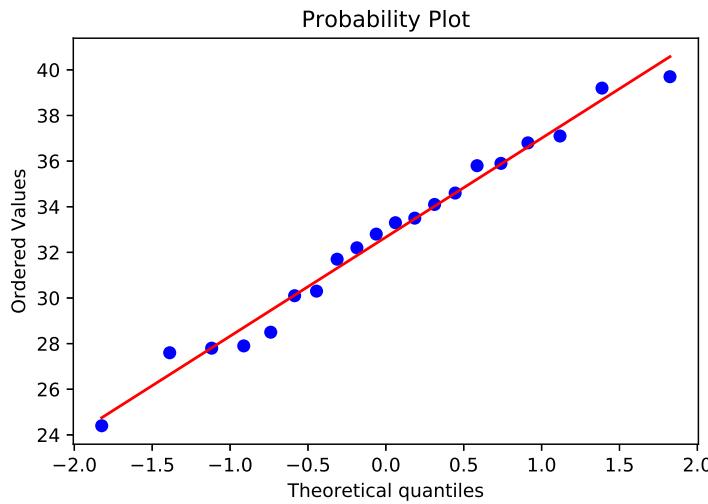
plt.show()
```



Oder viel einfacher mit dem Befehl **probplot** aus **scipy.stats** ([zu R](#))

Kapitel 4. Statistik für Messdaten

```
x = np.Series([24.4, 27.6, 27.8, 27.9, 28.5, 30.1, 30.3, 31.7, 32.2, 32.8, 33.3])
probplot(x, plot=plt)
```



Im Unterschied zum Plot vorher, ist die x -Achse auf die Standardnormalverteilung normiert. Zudem ist noch eine rote Linie eingezeichnet. Je mehr Punkte auf dieser Linie liegen, desto eher schliessen wir, dass die Messpunkte der Normalverteilung

$$\mathcal{N}(32.7, 4.15^2)$$

folgen. Bei realen Daten sind natürlich kleiner Abweichungen die Regel.

□

Bemerkungen:

- i. Die wahre theoretische Verteilung $\mathcal{N}(32.7, 4.15^2)$ wird hier jeweils normiert. Darum die Werte zwischen -2 und 2 auf der horizontalen Achse.
- ii. Hier wurde **Series** anstatt **np.array** verwendet. Der Grund dafür liegt darin, dass **np.array** die Standardabweichung nicht definitionsgemäß berechnet (Division durch n anstatt durch $n - 1$).

Für **np.array** kann man die definitionsgemäße Berechnung der Standardabweichung mit der Option **ddof=1**, also **x.std(ddof=1)**, erreichen. Bei **pandas** ist **ddof** standardmäßig (default) 1, bei **numpy** ist der Default-Wert 0.

Beispiel 4.1.2

Die Idee des QQ-Plots ist nochmals in Abbildung 4.1 dargestellt. Es werden dabei verschiedene Quantile des Modells und des Datensatzes betrachtet. Stimmen diese mehr oder weniger überein, so können wir annehmen, das unser Datensatz der vermuteten Modellverteilung folgt.

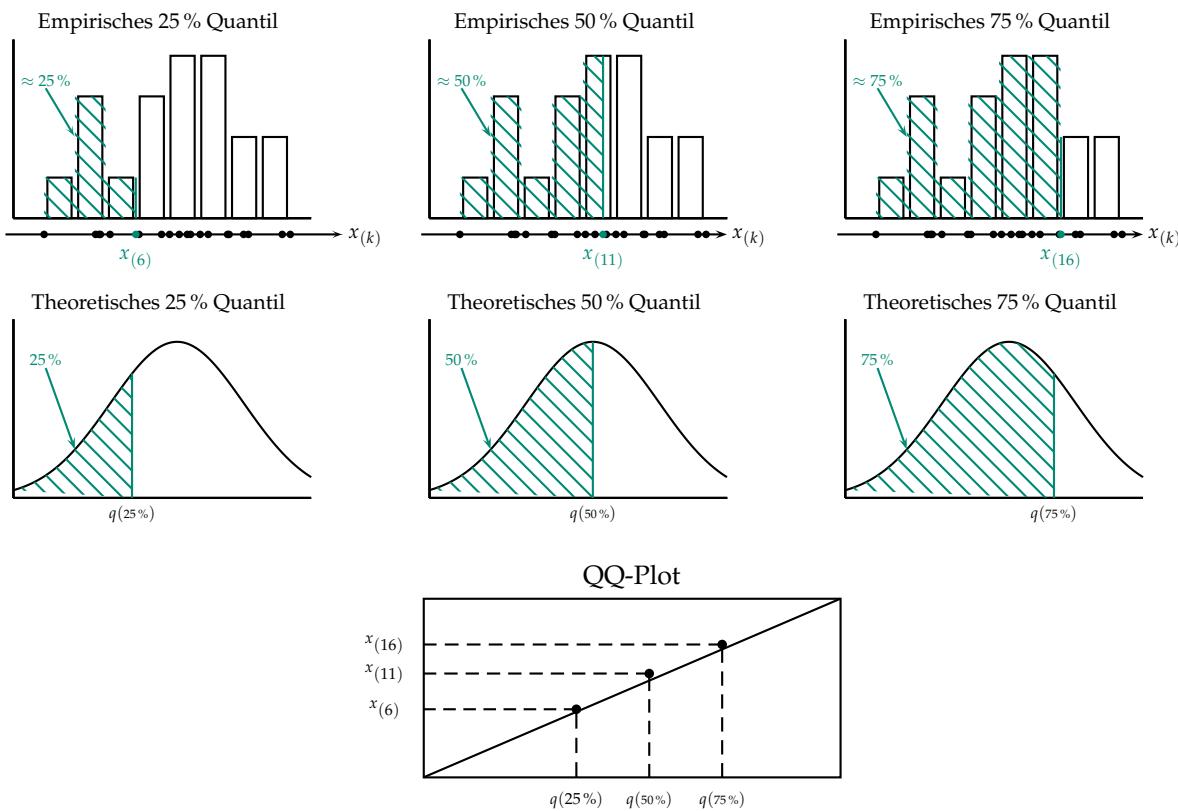


Abbildung 4.1.: Empirisches und theoretisches Quantil mit QQ-Plot vom Datensatz der Betondruckfestigkeit.

□

4.1.2. Normal-Plot

Meist will man nicht eine spezifische Verteilung, sondern eine ganze Klasse von Verteilungen prüfen. Zum Beispiel möchte man überprüfen, ob ein Datensatz einer Normalverteilung mit beliebigem μ und σ folgt.

Kapitel 4. Statistik für Messdaten

Ein Spezialfall von einem Q-Q-Plot ist der *Normal-Plot*: ein Q-Q Plot, bei dem die Modell-Verteilung gleich der Standardnormalverteilung $\mathcal{N}(0, 1)$ ist, heißt Normal-Plot. Falls die Daten Realisierungen von $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ sind und wir die Zufallsvariable X auffassen als $X = \mu_X + \sigma_X \cdot Y$ mit $Y \sim \mathcal{N}(0, 1)$, so gilt für die Quantile von X :

$$q_X(\alpha) = \underbrace{\mu_X}_{y(x)} + \underbrace{\sigma_X \cdot q_Y(\alpha)}_{b \cdot x}$$

wobei

$$q_Y(\alpha) = \Phi^{-1}(\alpha)$$

die α -Quantile der Standardnormalverteilung (Modellverteilung) ist (siehe Regel (iii) der linearen Transformationen in Kapitel 3.3).

Wenn man also einen Normal-Plot macht, so sollten die Punkte im Normal-Plot ungefähr auf der Geraden mit Achsenabschnitt μ und Steigung σ liegen.

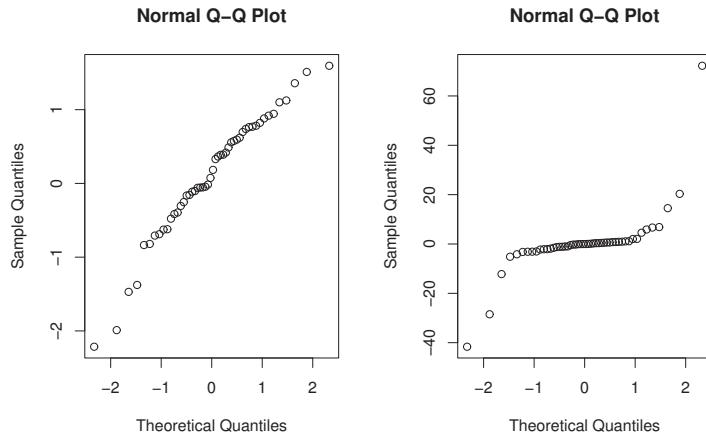


Abbildung 4.2.: Links: Normal-Plot für 50 Realisierungen von $\mathcal{N}(0, 1)$. Rechts: Normal-Plot für 50 Realisierungen von einer Verteilung, die sehr langschwänzig ist.

Abbildung 4.2 zeigt zwei Normal-Plots: einmal ist die datengenerierende Verteilung eine Normalverteilung. Die Punkte liegen in etwa auf einer Geraden. Das andere Mal sind die Daten von einer sehr langschwänzigen Verteilung erzeugt worden. Hier liegen die Punkte gar nicht auf einer Geraden, sondern an der rechten Extremität der Verteilung weit oberhalb der Geraden und auf der linken Extremität weit unterhalb der Geraden.

Natürlich liegen die Punkte in einem Normal-Plot nicht perfekt auf einer Geraden. Damit Sie ein Gefühl dafür bekommen, was eine akzeptable Abweichung von der Geraden ist, sind in Abbildung 4.3 die Normal-Plots von neun Datensätzen (je 50

Kapitel 4. Statistik für Messdaten

Beobachtungen) gezeigt, die alle von einer Standardnormalverteilung simuliert wurden. Abweichungen in dem Mass, wie sie dort zu sehen sind, kann man also erwarten, wenn die Daten tatsächlich von einer Normalverteilung stammen.

Abweichungen von einer Geraden in diesem Ausmass sind also zu erwarten, wenn die Daten wirklich normal verteilt sind. Falls die Abweichung von einer Geraden deutlich grösser ist, sind die Daten wohl nicht normalverteilt (siehe Abbildung 4.1 rechts).

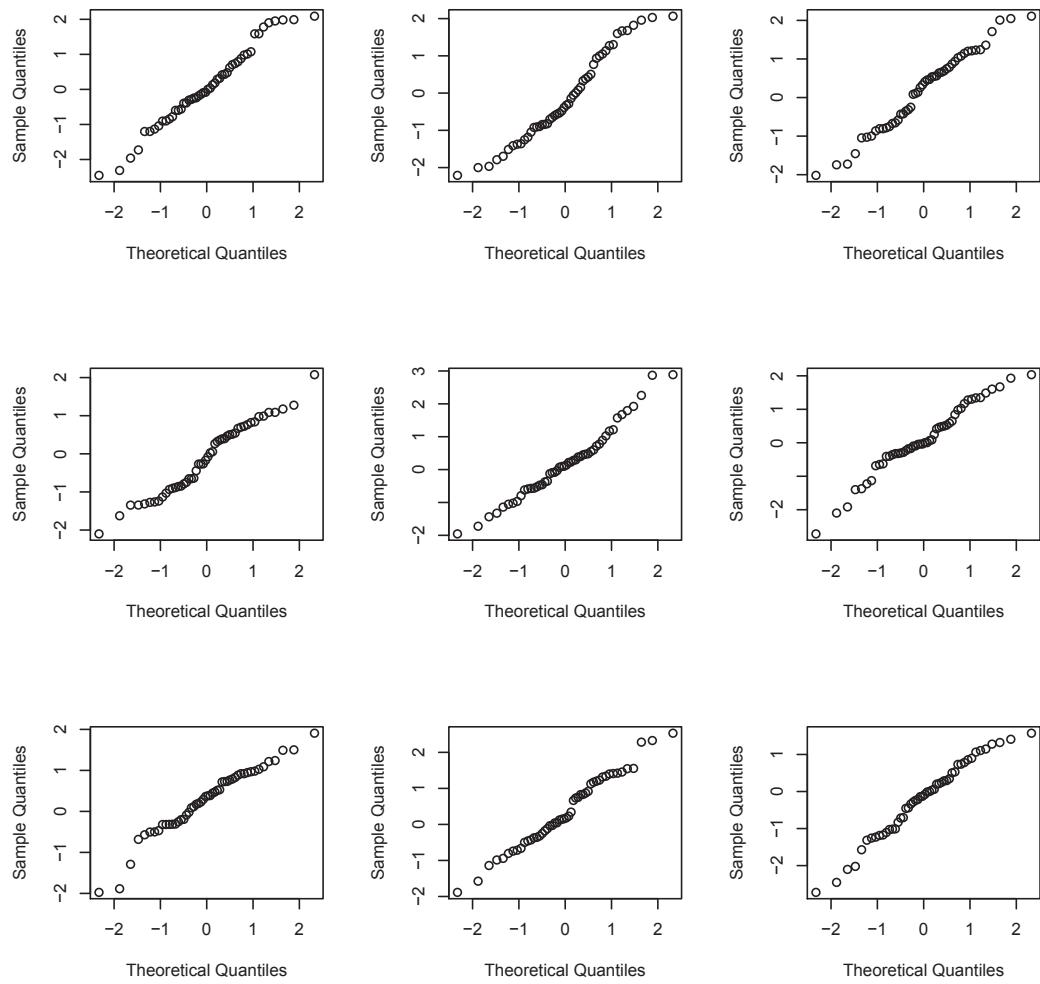


Abbildung 4.3.: Normal-Plots von neun Datensätzen (je 50 Beobachtungen), die alle von einer Standard-Normalverteilung simuliert wurden.

4.2. Parameterschätzung für stetige Wahrscheinlichkeitsverteilungen

Im Allgemeinen haben wir wie in Beispiel 4.1.1 der Betonfestigkeit einen Datensatz gegeben, von dem wir die Verteilung *nicht* kennen. Wir haben in diesem Beispiel eine Normalverteilung der Daten *vermutet*, was sich dann im QQ-Plot bestätigt hat. Allerdings kennen wir nach wie vor die *wahren* Parameter μ und σ nicht. Das Ziel ist es nun, die unbekannten Parameter aus den Daten annähernd zu bestimmen. Wir sprechen dann von der *Schätzung* der Parameter.

Es gibt dazu mehrere Methoden, und wir werden die beiden wichtigsten, die *Momentenmethode* und die *Maximum-Likelihood-Methode*, im Folgenden besprechen.

4.2.1. Momentenmethode

Um die Parameter einer Verteilung zu schätzen, können wir die *Momentenmethode* verwenden. Bei der Momentenmethode gehen wir wie folgt vor:

- Wir fassen unsere *Daten*

$$x_1, \quad x_2, \quad \dots, \quad x_n$$

als Realisierungen von Zufallsvariablen

$$X_1, \quad X_2, \quad \dots, \quad X_n$$

auf mit bekannter Verteilung (z.B. Normal- oder Exponentialverteilung).

Der oder die Parameter sind aber unbekannt (z.B. λ bei der Exponentialverteilung).

- Wir berechnen den (theoretischen) Erwartungswert $E(X)$, der abhängig vom unbekannten Parameter ist und lösen die Gleichung nach dem unbekannten Parameter auf, welchen wir schätzen wollen.

Bei der Exponentialverteilung gilt

$$E(X) = \frac{1}{\lambda}$$

und diese Gleichung lösen wir nach λ auf:

$$\lambda = \frac{1}{E(X)}$$

- Wir ersetzen den (theoretischen) Erwartungswert durch dessen (empirisches) Gegenstück, den empirischen Mittelwert \bar{x} . Wir erhalten so eine Schätzung für den unbekannten Parameter.

Für die Exponentialverteilung ersetzen wir $E(X)$ durch \bar{x} und λ durch $\hat{\lambda}$ (der geschätzte Wert)

$$\hat{\lambda} = \frac{1}{\bar{x}}$$

Beispiel 4.2.1 Betondruckfestigkeit

Bei der Messung der Betondruckfestigkeit aus Beispiel 4.1.1 hatten wir die Vermutung, dass die Daten normalverteilt sind. Mit einem Q-Q-Plot konnten wir das auch bestätigen. Somit folgt die Zufallsvariable X für die Betondruckfestigkeit einer Normalverteilung, also

$$X \sim \mathcal{N}(\mu, \sigma^2)$$

mit den (wahren, aber unbekannten) Parametern μ und σ .

Wir möchten nun mit der Momentenmethode einen Schätzer für μ finden. In diesem Fall gilt, dass

$$E(X) = \mu$$

(siehe Abschnitt 3.2.3) und wir müssen den Erwartungswert bloss noch durch den empirischen Mittelwert ersetzen:

$$\hat{\mu} = \widehat{E(X)} = \bar{x}_n$$

Also

$$\hat{\mu} = \bar{x}_n = \frac{x_1 + x_2 + \dots + x_{20}}{20} = \frac{653.3}{20} = 32.7$$

Für die Schätzung der Standardabweichung σ mit Hilfe der Momentenmethode benötigen wir folgende Beziehungen:

$$\begin{aligned}\sigma^2 &= E(X^2) - E(X)^2 \\ &= E(X^2) - \mu^2\end{aligned}$$

Schätzen wir $E(X)$ durch \bar{x}_n und $E(X^2)$ durch $\frac{1}{n} \sum_{i=1}^n x_i^2$, so ergibt sich folgendes Gleichungssystem

$$\begin{aligned}\hat{\mu} &= \bar{x}_n \\ \hat{\mu}^2 + \hat{\sigma}^2 &= \frac{1}{n} \sum_{i=1}^n x_i^2\end{aligned}$$

Die zweite Gleichung können wir nach $\hat{\sigma}^2$ auflösen, und durch Umformen erhalten erhalten wir:

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^n (x_i - \bar{x}_n)^2}{n}$$

Der Momentenmethode-Schätzer für die Varianz ist *nicht erwartungstreu*, d. h. der Faktor bei der Varianz müsste $\frac{1}{n-1}$ sein, ist aber in diesem Falle $\frac{1}{n}$.

Somit ergibt sich für unseren Datensatz folgende Schätzung für die Standardabweichung:

$$\begin{aligned}\hat{\sigma} &= \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{\mu})^2} \\ &= \sqrt{\frac{1}{20} \sum_{i=1}^{20} (x_i - 32.7)^2} \\ &= 4.04\end{aligned}$$

□

Beispiel 4.2.2 Exponentialverteilung

Wir messen die Lebenserwartung eines neuen elektronischen Moduls, welches in der Motorsteuerung eines Autos verwendet werden soll. Der Test ergab die folgenden Messungen (in Monaten):

$$x_1 = 11.96, x_2 = 5.03, x_3 = 67.40, x_4 = 16.07, x_5 = 31.50, x_6 = 7.73, x_7 = 11.10, x_8 = 22.38$$

Die Lebenserwartung ist in der Regel exponentialverteilt mit Parameter λ . Der Momentenschätzer von λ lautet:

$$E(X) = \frac{1}{\lambda} \quad \Rightarrow \quad \hat{\lambda} = \frac{1}{\bar{x}} = \frac{1}{21.6} = 0.0462$$

□

4.2.2. Maximum-Likelihood-Methode

Eine weitere, vielleicht die wichtigste, Schätzmethode für Parameter ist die *Maximum-Likelihood-Methode* (MLE). Sie beruht auf der Bestimmung eines Maximums mit Hilfe der Differentialrechnung.

Wir gehen dabei von n Beobachtungen

$$X_1 = x_1, \quad X_2 = x_2, \quad \dots, \quad X_n = x_n$$

aus. Zusätzlich nehmen wir an, dass diese Beobachtungen i.i.d. sind.

Maximum-Likelihood-Methode für diskrete Verteilungen

Am einfachsten lässt sich die Maximum-Likelihood Methode an einem Beispiel erklären.

Beispiel 4.2.3

Die Anzahl „Kopf“ bei n Münzwürfen ist folgendermassen verteilt:

$$X \sim \text{Bin}(n, \pi)$$

In unserem Beispiel sei $n = 100$, und die Zufallsvariable X hat beispielsweise den Wert 58 angenommen.

Die Aufgabe besteht nun darin, einen Wert für π zu finden, der möglichst gut zu unserer Beobachtung $x = 58$ passt. Welches Kriterium könnte man verwenden, um zu zeigen, dass ein Wert π_1 besser zu der Beobachtung $x = 58$ passt als π_2 ?

Eine Möglichkeit ist die folgende: Wir berechnen die Wahrscheinlichkeit, genau 58 mal Kopf bei 100 Münzwürfen zu erzielen. Einmal verwenden wir dabei z.B. $\pi_1 = 0.5$ und das andere mal $\pi_2 = 0.6$ und berechnen die zugehörigen Wahrscheinlichkeiten

$$P_{0.5}(X = 58) = 0.0223 \quad \text{und} \quad P_{0.6}(X = 58) = 0.074$$

Anschliessend wählen wir das π aus, das zu der grösseren Wahrscheinlichkeit für 58 mal Kopf führt. Dies wäre hier die Schätzung

$$\hat{\pi} = 0.6$$

□

Um den *wahrscheinlichsten* Wert für π zu erhalten, müssten wir natürlich nicht nur zwei Werte von π vergleichen, sondern alle, die denkbar sind. Mit anderen Worten: Wir müssen π so wählen, dass der Ausdruck

$$P[X = 58] = \binom{100}{58} \pi^{58} (1 - \pi)^{42}$$

maximal wird.

Dieses Problem lässt sich an sich leicht Methoden aus dem Grundlagenmodul Mathematik lösen

- Wir müssen diesen Ausdruck nach π ableiten
- Diese Ableitung nach π gleich null setzen

- Diese Gleichung nach π auflösen

$$\frac{d}{d\pi} \left(\binom{n}{x} \pi^x (1-\pi)^{n-x} \right) = 0 \quad \Rightarrow \quad \pi = \dots$$

Oft ist der Ausdruck, den man ableiten muss, aber recht kompliziert (wie auch in unserem Beispiel). In vielen Fällen kann man dann einen Trick anwenden: Jedes Extremum der Funktion $f(x)$ ist auch ein Extremum der Funktion $\log(f(x))$ und umgekehrt. D.h., anstatt $P[X = x]$ bzgl. π zu maximieren, können wir auch $\log(P[X = x])$ bzgl. π maximieren, falls das leichter geht. Das Ergebnis ist für beide Methoden völlig identisch.

In unserem Fall ist die zweite Variante tatsächlich etwas einfacher. Trotzdem ist folgende Rechnung etwas aufwendig und wird hier nur vollständigkeitshalber aufgeführt:

$$\begin{aligned} \log(P[X = x]) &= \log \left[\binom{n}{x} \pi^x (1-\pi)^{n-x} \right] \\ &= \log \left[\binom{n}{x} \right] + \log(\pi^x) + \log((1-\pi)^{n-x}) \\ &= \log \left[\binom{n}{x} \right] + x \cdot \log(\pi) + (n-x) \cdot \log(1-\pi) \end{aligned}$$

Durch Ableiten nach π und Nullsetzen erhalten wir:

$$\begin{aligned} \frac{d}{d\pi} \left[\log \left(\binom{n}{x} \right) + x \cdot \log(\pi) + (n-x) \cdot \log(1-\pi) \right] &= 0 \\ \frac{x}{\pi} + (n-x) \cdot \frac{1}{1-\pi} \cdot (-1) &= 0 \end{aligned}$$

Beachten Sie, dass $\log(\binom{n}{x})$ nicht von π abhängt und deshalb beim Ableiten verschwindet. Außerdem gilt: $\frac{d}{dx} \log(x) = \frac{1}{x}$.

Wenn wir diese Gleichung nach π auflösen, erhalten wir:

$$\begin{aligned} \frac{x}{\pi} - (n-x) \cdot \frac{1}{1-\pi} &= 0 \\ \frac{x}{\pi} &= \frac{n-x}{1-\pi} \end{aligned}$$

$$x - \pi x = \pi n - \pi x$$

$$\pi = \frac{x}{n}$$

Kapitel 4. Statistik für Messdaten

Man erhält folglich als Resultat für die Schätzung von π

$$\hat{\pi} = \frac{x}{n}$$

Mit den Zahlen in unserem Beispiel erhalten wir

$$\hat{\pi} = \frac{58}{100} = 0.58$$

Im Falle der Binomialverteilung ist das Ergebnis der Maximum-Likelihood-Methode identisch mit dem Ergebnis der Momentenmethode. Im Allgemeinen gilt dies aber nicht.

Mit obiger Methode wählen wir also das π , mit dem die Beobachtung am wahrscheinlichsten ist. Daher nennt man diese Methode *Maximum-Likelihood-Methode*. Sie ist die mit Abstand gebräuchlichste Methode, um Parameter zu schätzen und oft der Momentenmethode überlegen.

Wir sprechen bei der zu maximierenden Funktion auch von der *Likelihood-Funktion* $L(\pi)$.

$$L(\pi) = \binom{n}{x} \pi^x (1 - \pi)^{n-x}$$

In der Regel erfolgt die Schätzung aufgrund von mehreren Beobachtungen.

Beispiel 4.2.4

Wir werfen mit einer Münze und machen dabei zwei Versuche. Im 1. Versuch werfen wir 50-mal und erreichen 30-mal K; im 2. Versuch sind es 115-mal K auf 160 Würfe. Wir haben also zwei Zufallsvariablen X_1 und X_2 . Es gilt

$$X_1 \sim \text{Bin}(50, \pi) \quad \text{und} \quad X_2 \sim \text{Bin}(160, \pi)$$

Wichtig hier ist, dass der Parameter π bei beiden Versuchen identisch ist. Diesen unbekannten Parameter π wollen wir nun mit der Maximum-Likelihood-Methode schätzen.

Aus den Angaben suchen wir die Wahrscheinlichkeit

$$P[(X_1 = 30) \cap (X_2 = 115)]$$

Da wir die beiden Versuche als unabhängig betrachten, gilt

$$\begin{aligned} P[(X_1 = 30) \cap (X_2 = 115)] &= P[X_1 = 30] \cdot P[X_2 = 115] \\ &= \binom{50}{30} \pi^{30} (1 - \pi)^{20} \cdot \binom{160}{115} \pi^{115} (1 - \pi)^{45} \end{aligned}$$

Kapitel 4. Statistik für Messdaten

Diese Funktion ist unsere Likelihood-Funktion $L(\pi)$:

$$L(\pi) = \binom{50}{30} \pi^{30} (1-\pi)^{20} \cdot \binom{160}{115} \pi^{115} (1-\pi)^{45}$$

Diese Funktion müssen wir nach π ableiten, gleich 0 setzen und nach π auflösen. Dies führen wir hier nicht aus, das Resultat lautet

$$\hat{\pi} = \frac{145}{210}$$

Im allgemeinen Fall für $X_1 = x_1$ und $X_2 = x_2$ mit n_1 bzw. n_2 Versuchen, erhalten wir mit der Maximum-Likelihood-Methode

$$\hat{\pi} = \frac{x_1 + x_2}{n_1 + n_2}$$

Dies ist dasselbe Resultat wie bei der Momentenmethode, was für andere Verteilungen aber nicht der Fall sein muss.

□

Im Fall von einer diskreten Wahrscheinlichkeitsverteilung lässt sich also die (Punkt-) Wahrscheinlichkeit, dass sich diese n Beobachtungen (Ereignisse) ereignet haben, schreiben als (Multiplikationsregel für unabhängige Ereignisse)

$$\begin{aligned} P[(X_1 = x_1) \cap \dots \cap (X_n = x_n)] &= P[X_1 = x_1] \cdot P[X_2 = x_2] \cdot \dots \cdot P[X_n = x_n] \\ &= \prod_{i=1}^n P[X_i = x_i] \end{aligned}$$

Bemerkungen:

- i. Das Zeichen \prod ist eine abgekürzte Schreibweise für Produkte, wie \sum für Summen steht.

Die Wahrscheinlichkeit, dass die n unabhängigen Zufallsvariablen x_1, x_2, \dots, x_n beobachtet werden, hängt vom Parameter ϑ ab, welchen wir schätzen möchten. Die dazugehörige *Likelihood-Funktion* $L(\vartheta)$ lautet also:

$$\begin{aligned} L(\vartheta) &= P[X_1 = x_1 | \vartheta] \cdot P[X_2 = x_2 | \vartheta] \cdot \dots \cdot P[X_n = x_n | \vartheta] \\ &= \prod_{i=1}^n P[X_i = x_i | \vartheta] \end{aligned}$$

wobei $P[X_i = x_i | \vartheta]$ die Punktwahrscheinlichkeit bezeichnet, dass der Wert x_i beobachtet wurde, gegeben den Parameterwert ϑ . Die Idee der Maximum-Likelihood-Methode ist nun, den Parameter ϑ so zu schätzen, dass die beobachteten Ereignisse am wahrscheinlichsten sind. Wir suchen also das Maximum für die Maximum-Likelihood-Funktion $L(\vartheta)$.

Maximum-Likelihood-Methode für stetige Verteilungen

Im Folgenden betrachten wir nun kontinuierliche Wahrscheinlichkeitsverteilungen mit der Wahrscheinlichkeitsdichtefunktion $f(x; \vartheta)$.

Unter der Annahme von $f(x; \vartheta)$, einschliesslich eines Wertes von ϑ und der Unabhängigkeit unserer Beobachtungen, ist die Wahrscheinlichkeit, dass die erste Beobachtung x_1 im Intervall $[x_1, x_1 + dx_1]$ liegt,

$$f(x_1; \vartheta) dx_1$$

die Wahrscheinlichkeit, dass die zweite Beobachtung x_2 im Intervall $[x_2, x_2 + dx_2]$ liegt, ist

$$f(x_2; \vartheta) dx_2$$

etc...

Somit ist die Wahrscheinlichkeit, dass jede Beobachtung x_i im Intervall $[x_i, x_i + dx_i]$ liegt, gegeben durch

$$\prod_{i=1}^n f(x_i; \vartheta) dx_i$$

Falls die vermutete Wahrscheinlichkeitsdichtefunktion $f(x_i; \vartheta)$ und der Parameterwert von ϑ korrekt sind, erwartet man eine hohe Wahrscheinlichkeit für die Daten, die tatsächlich beobachtet wurden.

Wir wollen dies graphisch veranschaulichen. In Abbildung 4.4 sind drei normalverteilte Messungen eingezeichnet, die in der Nähe von 1 liegen, also weg von 0 liegen. Wir wählen als Parameter ϑ den Erwartungswert μ .

Die Normalverteilungskurve mit $\mu = 0$ „passt“ hier schlecht zu den Funktionswerten bzw. den zugehörigen Wahrscheinlichkeiten.

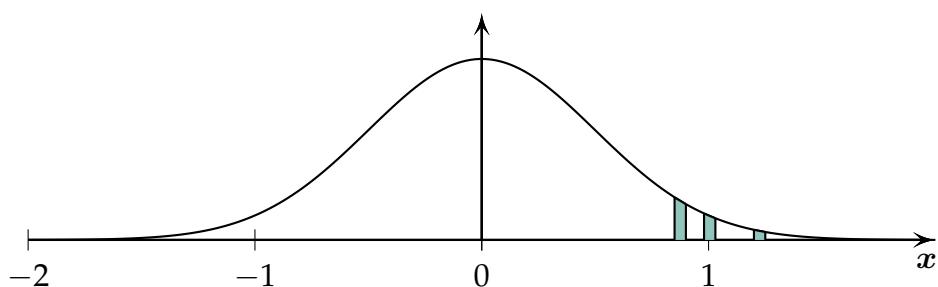


Abbildung 4.4.: Schlecht passende Parameterwerte der Verteilung mit kleinen Wahrscheinlichkeiten für die beobachteten Werte.

Eine Veränderung von μ resultiert in einer Translation der Kurve in x -Richtung. Durch diese Veränderung von μ können wir erreichen, dass die Funktionswerte besser zur

Kurve passen (siehe Abbildung 4.5). Ziel ist es, jenes μ zu finden, damit die Funktionswerte am besten zur Kurve passen und zwar im dem Sinne, dass

$$\prod_{i=1}^n f(x_i; \vartheta) dx_i$$

so gross wie möglich, also maximal wird. Im graphischen Beispiel wird das wahre μ wohl in der Nähe von 1 liegen. Aber wo?

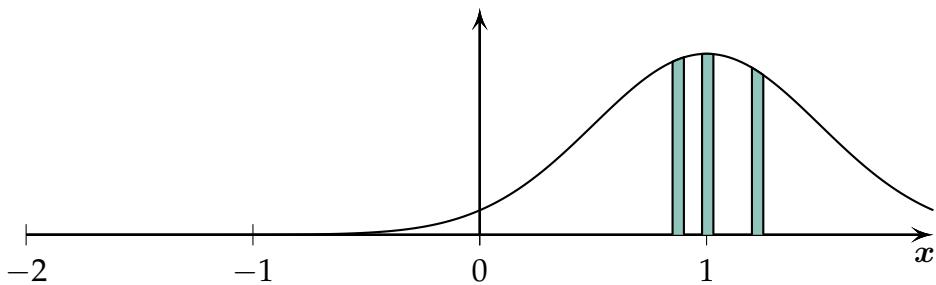


Abbildung 4.5.: Gut passender Parameterwert der Verteilung mit grossen Wahrscheinlichkeiten für die beobachteten Werte.

Allgemein

Falls der Parameterwert ϑ weit weg vom wahren Parameterwert liegt, wird die obige Funktion einen kleinen Wert ergeben. Da die (infinitesimalen) Intervallbreiten dx_i nicht vom Parameterwert ϑ abhängen, gelten diese Überlegungen ebenfalls für die folgende Funktion

$$L(\vartheta) = \prod_{i=1}^n f(x_i; \vartheta)$$

die die *Likelihood-Funktion* für kontinuierliche Wahrscheinlichkeitsverteilungen darstellt. Um den Parameterwert von ϑ zu schätzen, werden wir wiederum $L(\vartheta)$ maximieren.

Beispiel 4.2.5 Normalverteilung; σ^2 bekannt

Sei X normalverteilt mit unbekanntem μ und bekanntem σ^2 . Die Likelihood-Funktion für den Datensatz x_1, x_2, \dots, x_n ist gegeben durch:

$$L(\mu) = \prod_{i=1}^n \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

Es ist einfacher, die Log-Likelihood-Funktion zu maximieren:

$$\ln(L(\mu)) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2$$

Die Ableitung ergibt:

$$\frac{d \ln(L(\mu))}{d\mu} = \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)$$

Setzen wir diesen Ausdruck gleich 0 und lösen nach μ auf, ergibt dies:

$$\hat{\mu} = \frac{\sum_{i=1}^n x_i}{n} = \bar{x}_n$$

Somit ist der empirische Mittelwert der Likelihood-Schätzer für μ gegeben durch den empirischen Mittelwert. Dieser Schätzer ist identisch mit dem Momentenschätzer.

□

Beispiel 4.2.6 Normalverteilt; σ^2 unbekannt

Sei X normalverteilt mit Erwartungswert μ und Varianz σ^2 , wobei der Erwartungswert und die Varianz unbekannt sind. Die Likelihood-Funktion für den Datensatz x_1, x_2, \dots, x_n ist gegeben durch:

$$L(\mu, \sigma^2) = \prod_{i=1}^n \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}}$$

Die Log-Likelihood-Funktion ist dann:

$$\ln(L(\mu, \sigma^2)) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (x_i - \mu)^2$$

Wir suchen nun die Schätzer für μ und σ^2 , welche die Log-Likelihood-Funktion maximieren. Daher leiten wir die Log-Likelihood-Funktion partiell nach beiden Variablen ab und setzen die Ableitungen gleich 0:

$$\begin{aligned} \frac{\partial \ln(L(\mu, \sigma^2))}{\partial \mu} &= \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu) \stackrel{!}{=} 0 \\ \frac{\partial \ln(L(\mu, \sigma^2))}{\partial (\sigma^2)} &= -\frac{n}{\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 \stackrel{!}{=} 0 \end{aligned}$$

Die Lösung dieses Gleichungssystems sind unsere Maximum-Likelihood-Schätzer

$$\hat{\mu} = \bar{x}, \quad \hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2$$

Die Maximum-Likelihood-Schätzung von σ^2 ist identisch mit der Schätzung der Momentenmethode. Die Maximum-Likelihood-Schätzung von σ^2 ist folglich nicht erwartungstreu.

□

Beispiel 4.2.7 Exponentialverteilung

Sei X exponentialverteilt mit Parameter λ . Die Likelihood-Funktion für den Datensatz x_1, x_2, \dots, x_n ist gegeben durch:

$$L(\lambda) = \prod_{i=1}^n \lambda e^{-\lambda x_i}$$

Die Log-Likelihood-Funktion ist

$$\ln(L(\lambda)) = n \ln(\lambda) - \lambda \sum_{i=1}^n x_i$$

Wir leiten die Log-Likelihood-Funktion nach λ ab und setzen die Ableitung gleich 0

$$\frac{d \ln(L(\lambda))}{d \lambda} = \frac{n}{\lambda} - \sum_{i=1}^n x_i \stackrel{!}{=} 0$$

Der Maximum-Likelihood-Schätzer $\hat{\lambda}$ ist die Lösung der obigen Gleichung:

$$\hat{\lambda} = \frac{n}{\sum_{i=1}^n x_i} = \frac{1}{\bar{x}}$$

□

Bemerkungen:

- i. Sowohl bei der Exponential- wie bei der Normalverteilung stimmen Momenten- und Maximum-Likelihood-Schätzer überein. Dies muss für andere Verteilungen allerdings nicht gelten.

4.3. Statistische Tests und Vertrauensintervall für eine Stichprobe bei normalverteilten Daten

4.3.1. Problemstellung

Wir veranschaulichen die Thematik dieses Unterkapitels an folgendem Beispiel.

Beispiel 4.3.1

Wir betrachten zwei *Datensätze*, bei welchen zwei Methoden zur Bestimmung der latenten Schmelzwärme von Eis verglichen werden. Wiederholte Messungen der freigesetzten Wärme beim Übergang von Eis bei $-0.7\text{ }^{\circ}\text{C}$ zu Wasser bei $0\text{ }^{\circ}\text{C}$ ergaben die Werte (in cal/g), die in Tabelle 4.2 aufgeführt sind.

Methode A	79.98	80.04	80.02	80.04	80.03	80.03	80.04	79.97	80.05
Methode A	80.03	80.02	80.00	80.02					
Methode B	80.02	79.94	79.98	79.97	79.97	80.03	79.95	79.97	

Table 4.2.: Messungen zur Bestimmung der latenten Schmelzwärme von Eis anhand von zwei Methoden.

Wir können diese Messungen als Realisierungen von unabhängigen, identisch verteilten Zufallsvariablen X_i betrachten. So ist der zweite Messwert $x_2 = 80.04$ der Methode A eine Realisierung der Zufallsvariable X_2 .

□

Allgemeiner fassen wir nun die Messdaten x_1, \dots, x_n als Realisierungen von

$$X_1, \dots, X_n \text{ i.i.d.} \sim \mathcal{N}(\mu, \sigma_X^2)$$

Zwei Kennzahlen der Zufallsvariablen X_i sind:

$$\mathbb{E}(X_i) = \mu \quad \text{und} \quad \text{Var}(X_i) = \sigma_X^2$$

Typischerweise sind diese (und andere) Kennzahlen unbekannt, und man möchte Rückschlüsse darüber aus den Daten wie in Beispiel 4.3.1 machen.

Die (Punkt-) Schätzungen für den Erwartungswert und die Varianz sind:

$$\begin{aligned}\widehat{\mu} &= \frac{1}{n} \sum_{i=1}^n X_i \\ \widehat{\sigma}_X^2 &= \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2\end{aligned}$$

Beachten Sie, dass die Schätzer hier als Funktionen der Zufallsvariablen X_1, \dots, X_n geschrieben sind: insbesondere sind $\hat{\mu}$ und $\hat{\sigma}_X^2$ selbst wieder Zufallsvariablen - die Verteilungseigenschaften von $\hat{\mu}$ wurden im Kapitel 3.4 diskutiert.

In den folgenden Beispielen werden wir die Schätzungen für das Beispiel der Methode A zur Messung der Schmelzwärme aus dem Beispiel 4.3.1 durchführen und auf die Problem- und Fragestellungen hinweisen.

Beispiel 4.3.2

Für unser Beispiel der Methode A zur Messung der Schmelzwärme lauten die Schätzungen für den Mittelwert μ und die Varianz σ_X^2

$$\hat{\mu} = 80.02 \quad \text{und} \quad \hat{\sigma}_X^2 = 0.024^2$$

Diese Werte berechnen wir mit **Python**: ([zu R](#))

```
from pandas import Series, DataFrame
import pandas as pd
import numpy as np

methodeA = Series([79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04,
                  79.97, 80.05, 80.03, 80.02, 80.00, 80.02])

methodeA.mean()
methodeA.std()

## 80.02076923076923
## 0.023965787580611863
```

Es stellt sich nun das Problem, dass für *andere* Messreihen diese Schätzwerte natürlich andere Werte ergeben. Die Streuung der Schätzer wollen wir nun im folgenden Beispiel zur Messung der Schmelzwärme untersuchen:



Beispiel 4.3.3

Wir wollen neue Messreihen simulieren, die „ähnlich“ aussehen wie die Werte in Methode A. Dazu machen wir die *Annahme*, dass die Messwerte in Methode A normalverteilt sind mit den wahren Parametern $\mu = 80$ und $\sigma_X^2 = 0.02^2$.

Dann generieren wir mit **norm.rvs()** aus **scipy.stats** Zufallszahlen, die dieser Verteilung folgen. Wir beschränken uns hier der Übersichtlichkeit halber auf Messreihen der Länge 6. Zudem runden wir die Resultate auf zwei Nachkommastellen (**np.round(..., 2)**). ([zu R](#))

Kapitel 4. Statistik für Messdaten

```
from scipy.stats import norm
np.random.seed(1)

methodeA_sim1 = Series(np.round(norm.rvs(size=6, loc=80, scale=0.02), 2))

methodeA_sim1

methodeA_sim1.mean()
methodeA_sim1.std()

## 0      80.03
## 1      79.99
## 2      79.99
## 3      79.98
## 4      80.02
## 5      79.95
## dtype: float64
## 79.99333333333333
## 0.028751811537128993
```

Wir sehen, dass die geschätzten Werte $\hat{\mu}$ und $\hat{\sigma}^2$ jeweils (leicht) anders sind, als im Beispiel 4.3.2 vorher.

Führen wir dies fünfmal durch, so sehen die Resultate wie folgt aus: ([zu R](#))

```
np.random.seed(17)

for i in range(5):
    methodeA_sim1 = Series(np.round(norm.rvs(size=6, loc=80, scale=0.02), 2))
    print("Mittelwert:", np.round(methodeA_sim1.mean(), 3))
    print("Standardabw.:", np.round(methodeA_sim1.std(), 3))
    print()

## Mittelwert: 80.01
## Standardabw.: 0.027
##
## Mittelwert: 80.007
## Standardabw.: 0.02
##
## Mittelwert: 79.992
## Standardabw.: 0.028
##
## Mittelwert: 79.995
## Standardabw.: 0.016
##
## Mittelwert: 79.992
## Standardabw.: 0.013
```

Die Mittelwerte sind hier alle nahe bei 80, was auch zu erwarten war. Wir haben hier keine Zweifel, dass der wahre Mittelwert nicht $\mu = 80$ sein könnte. Diese Abweichungen sind durchaus zu erwarten.

□

Bemerkungen:

- i. Mit `np.random.seed(...)` wird erreicht, dass immer dieselben Zufallszahlen erzeugt werden. Dies hat den Vorteil, dass sich die Zahlen mit jeder Erstellung dieses Skriptes nicht ändern.

Beispiel 4.3.4

Im Beispiel 4.3.3 vorher liegen die geschätzten Mittelwerte alle sehr nahe bei $\mu = 80$. Allerdings sind auch folgende Fälle möglich: ([zu R](#))

```
np.random.seed(463137)

methodeA_sim2 = Series(np.round(norm.rvs(size=6, loc=80, scale=0.02), 2))

methodeA_sim2

methodeA_sim2.mean()
methodeA_sim2.std()

## 0      80.07
## 1      80.06
## 2      80.03
## 3      80.03
## 4      80.02
## 5      80.03
## dtype: float64
## 80.04
## 0.0199999999999862
```

Der Mittelwert dieser Messreihe folgt gemäss Abschnitt 3.4.3 folgender Verteilung:

$$\bar{X}_6 \sim \mathcal{N} \left(80, \frac{0.02^2}{6} \right) = \mathcal{N} \left(80, 0.0082^2 \right)$$

Der Mittelwert der oben simulierten Messreihe, nämlich 80.04, ist fast 5 Standardfehler grösser als 80, was eben möglich ist, aber nicht sehr wahrscheinlich ist (siehe Abbildung 4.6). Schliesslich würden wir erwarten, dass der Mittelwert in der Nähe von $\mu = 80$ liegt, sofern der wahre Mittelwert tatsächlich $\mu = 80$ ist.

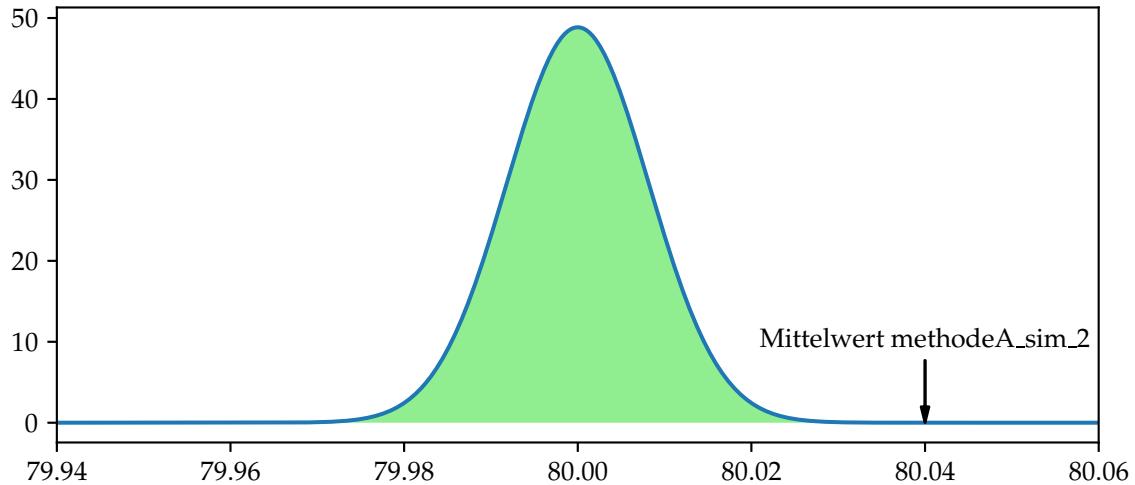


Abbildung 4.6.: Mittelwert, der sehr weit vom erwarteten Wert von $\mu = 80$ entfernt ist

Wie wahrscheinlich (oder unwahrscheinlich) ist nun aber ein solcher Mittelwert?

□

Beispiel 4.3.5

Ein weiteres Beispiel: ([zu R](#))

```
np.random.seed(647)
methodeA_sim3 = Series(np.round(norm.rvs(size=6, loc=80, scale= 0.02), 2))

methodeA_sim3

methodeA_sim3.mean()
methodeA_sim3.std()

## 0      79.98
## 1      79.99
## 2      80.00
## 3      79.93
## 4      80.00
## 5      79.98
## dtype: float64
## 79.98
## 0.02607680962080759
```

Hier liegt der Mittelwert etwa 3 Standardabweichungen unterhalb von 80 (siehe Abbildung 4.7). Dies ist zwar immer noch weit weg vom erwarteten Mittelwert 80, aber nicht mehr so deutlich wie in Beispiel 4.3.4. Schliesslich erwarten wir ja, dass der Durchschnitt der Messreihe in der Nähe vom wahren $\mu = 80$ liegt. Liegt der Durchschnitt weit weg von diesem wahren Mittelwert, so beginnen zu zweifeln, ob der wahre Mittelwert tatsächlich 80 ist.

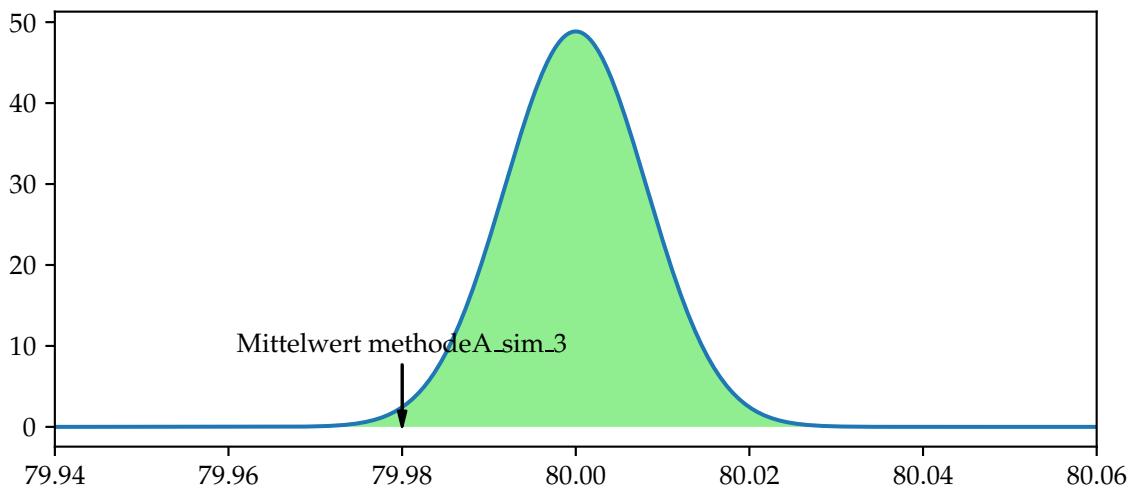


Abbildung 4.7.: Mittelwert, der weit vom erwarteten Wert von $\mu = 80$ entfernt ist

□

Die beiden vorangehenden Beispiele führen uns zu folgenden Fragestellungen:

- Sind die Messreihen in den Beispielen 4.6 und 4.7 mit der Annahme $\mu = 80$ noch kompatibel oder müssen wir an dieser Annahme zweifeln? Das heisst: Liegt der Mittelwert der Messreihe in der „Nähe“ des wahren Mittelwertes $\mu = 80$ oder liegt er so „weit“ entfernt, dass wir an der Angabe des wahren $\mu = 80$ zweifeln müssen?

Hier stellt sich natürlich die Frage, was „nahe“ heisst. Dies werden wir im nächsten Abschnitt beantworten.

- Können wir ein Intervall für $\hat{\mu}$ angeben, indem sich der wahre Wert von $\mu = 80$ mit einer gewissen Wahrscheinlichkeit befindet?

Wir wollen nochmals darauf hinweisen, dass der wahre Mittelwert grundsätzlich *nicht* bekannt ist.

Allgemein:

Wann immer wir eine Messreihe neu erheben, werden wir für die betreffende Messreihe unterschiedliche Werte (Realisierungen) von $\hat{\mu}$ und $\hat{\sigma}_X^2$ ermitteln. Es ist also vernünftig, für den Schätzer $\hat{\mu}$ ein Intervall anzugeben, in welches der wahre Wert von μ mit einer bestimmten Wahrscheinlichkeit fällt. Es handelt sich also darum, für einen geschätzten Parameter ein sogenanntes *Vertrauensintervall* anzugeben.

Weiter kann man sich fragen, ob eine Realisierung von μ kompatibel ist mit einem vermuteten μ_0 . Die entsprechende Fragestellung wird durch einen *statistischen Test* beantwortet.

Wir beginnen mit dem *statistischen Test* oder *Hypothesentest*.

4.3.2. Hypothesentest

Der Hypothesentest stellt ein wichtiges statistisches Mittel dar, um zu entscheiden, ob eine Messreihe zum Wert einer vermuteten Grösse „passt“, wobei wir den wahren Wert dieser Grösse *nicht* kennen.

Beispiel 4.3.6

Eine Brauerei bestellt eine neue Abfüllmaschine für 500 ml Büchsen. Die Abfüllmaschine füllt *nie genau* 500 ml ab, sondern nur *ungefähr* 500 ml.

Die Brauerei ist daran interessiert, dass die Abfüllmaschine möglichst genau abfüllt. Füllt die Maschine zuviel ab, so ist dies schlecht für die Brauerei, da sie zuviel Bier für denselben Preis verkauft. Füllt sie zuwenig ab, sind die Kunden und der Konsumentenschutz unzufrieden, da diese für den entsprechenden Preis zu wenig Bier bekommen.

Die Herstellerfirma behauptet, dass die Maschine die Büchsen normalverteilt mit $\mu = 500$ ml und $\sigma = 1$ ml abfüllt. Die Brauerei macht dazu 100 Stichproben. Der Mittelwert dieser Stichproben ist 499.57 ml. Dies ist zwar weniger als 500 ml, aber liegt dies noch „im Rahmen“ der Angaben $\mu = 500$ ml und $\sigma = 1$ ml des Herstellers der Abfüllanlage? Wie können wir dies überprüfen?

□

Allgemein

Sie stellen eine Maschine her und müssen sich auf die Angaben der Spezifikationen der Hersteller für die Bestandteile verlassen können. Wie können Sie feststellen, ob die Bestandteile die Spezifikationen auch erfüllen?

Beispiel 4.3.7

Eine Anfrage beim Bundesamt für Sport ergibt, dass die durchschnittliche Körpergrösse der erwachsenen Frauen in der Schweiz bei 180 cm mit einer Standardabweichung von 10 cm liegt.

Diese Angabe für den Mittelwert ist gefühlsmässig wohl falsch, da viel zu hoch. Wie können wir dies aber mathematisch überprüfen und begründen, ohne uns auf unser Gefühl zu verlassen zu müssen?

□

Ziel dieses Abschnittes ist es, ein standardisiertes, reproduzierbares Verfahren einzuführen, mit dem wir entscheiden können, ob der Mittelwert einer Messreihe zu einem bestimmten „wahren“ Mittelwert μ passt oder nicht.

Wir wollen die wichtigsten Begriffe für den Hypothesentest mit Hilfe der Beispiele zur Bestimmung der Schmelzwärme 4.3.1 und 4.3.5 vertiefen.

Beispiel 4.3.8

Wie in Beispiel 4.3.2 gehen wir davon aus, dass die Daten normalverteilt sind mit $\mu = 80.00$ und $\sigma = 0.02$. Wie können wir überprüfen, ob der vermutete wahre Mittelwert $\mu = 80$ mit der beobachteten Messreihe kompatibel ist?

Die Grundidee ist, mit einer beobachteten Messreihe zu überprüfen, ob *unter der Annahme* $\mu = 80$, der Mittelwert dieser Messreihe wahrscheinlich ist oder nicht. Wir wählen dazu eine Messreihe der Länge 6 aus und gehen von folgendem Modell aus:

Modell

Die 6 Messwerte sind Realisierungen der Zufallsvariablen X_1, X_2, \dots, X_6 , wobei X_i eine kontinuierliche Messgrösse ist. Es soll gelten:

$$X_1, \dots, X_6 \text{ i.i.d. } \sim \mathcal{N}(80, 0.02^2)$$

Als nächstes formalisieren wir die *Annahme* oder *Vermutung* $\mu = 80$:

Nullhypothese

$$H_0 : \mu = \mu_0 = 80$$

Alternativhypothese

$$H_A : \mu \neq \mu_0 = 80 \text{ oder } „<“ \text{ oder } „>“$$

Als Messreihe wählen wir [methodeA_sim3](#) aus Beispiel 4.3.5:

```
## 0      79.98
## 1      79.99
## 2      80.00
## 3      79.93
## 4      80.00
## 5      79.98
## dtype: float64
## Mittelwert: 79.98
```

Der (geschätzte) Mittelwert ist $\hat{\mu} = 79.98$. Als nächstes müssen wir konkretisieren, was es bedeutet, dass dieser Mittelwert unter der Annahme $\mu = 80$ (un)wahrscheinlich ist. Die Wahrscheinlichkeit

$$P(\bar{X}_6 = 79.98)$$

bringt uns hier nicht weiter, da diese null ist. Da $\hat{\mu} < 80$ ist, können wir aber folgende Wahrscheinlichkeit betrachten:

$$P(\bar{X}_6 \leq 79.98)$$

Unter der Annahme $\mu = 80$ und $\sigma = 0.02$ ist \bar{X}_6 wie folgt verteilt

$$\bar{X}_6 \sim \mathcal{N}\left(80, \frac{0.02^2}{6}\right)$$

Die Grösse \bar{X}_6 ermöglicht es uns, zu *testen*, ob der beobachtete Wert von \bar{X}_6 , also der Mittelwert unserer Messreihe, mit der Nullhypothese $\mu_0 = 80$ kompatibel ist.

Teststatistik

Verteilung der Teststatistik T unter der Nullhypothese H_0 :

$$T = \bar{X}_6 \sim \mathcal{N}\left(80, \frac{0.02^2}{6}\right)$$

Damit erhalten wir für die Wahrscheinlichkeit ([zu R](#))

$$P(\bar{X}_6 \leq 79.98) = 0.007$$

```
norm.cdf(x=79.98, loc=80, scale=0.02/np.sqrt(6))

## 0.007152939217724509
```

Diese Wahrscheinlichkeit beträgt 0.7 % und ist eher klein. Ist sie nun *zu klein*?

Nun treffen wir eine *Abmachung*: Ist die Wahrscheinlichkeit $P(\bar{X}_6 \leq 79.98)$ kleiner als 2.5 %, dann erachten wir sie als *zu klein* (siehe Abbildung 4.8).

Gemäss dieser Abmachung ist

$$P(\bar{X}_6 \leq 79.98) < 0.025$$

und wir betrachten den geschätzten Mittelwert $\hat{\mu} = 79.98$ als *zu unwahrscheinlich*, als dass dieser zum Wert $\mu = 80$ passen könnte (siehe auch Abbildung 4.9). Wir gehen also davon aus, dass der angegebene Mittelwert von $\mu = 80$ nicht plausibel ist.

Für das bessere Verständnis wollen wir diesen Sachverhalt noch graphisch darstellen. In Abbildung 4.8 teilen wir die Normalverteilungskurve in drei Teile auf:

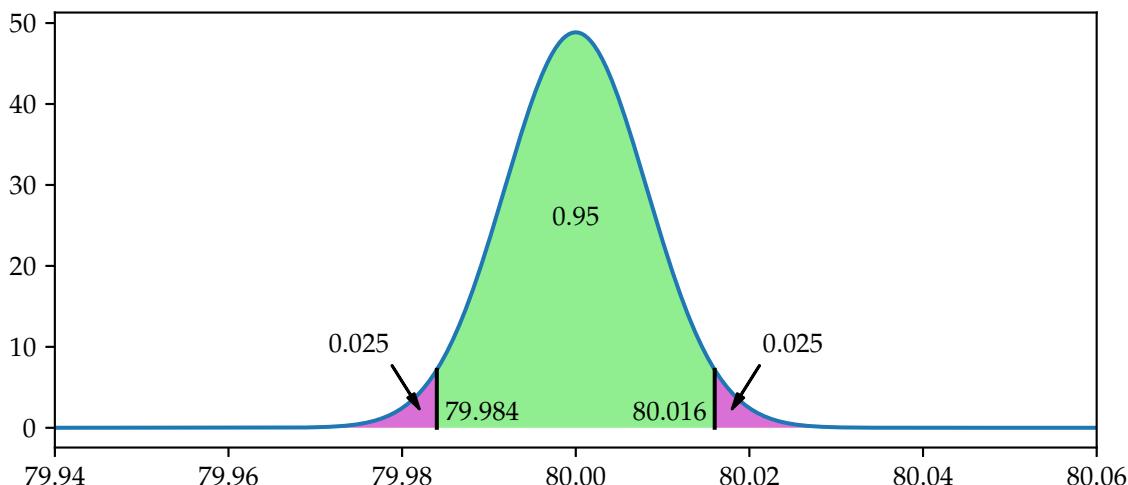


Abbildung 4.8.: Verwerfungsbereich 1

- Der symmetrische Teil um den Mittelwert $\mu = 80$ soll 0.95, also 95 % betragen.
- Die beiden Teile links und rechts müssen zusammen 0.05 ergeben. Also ergibt sich für jeden Teil 0.025.
- Die Grenzen entsprechen den 0.025- und 0.975-Quantilen. ([zu R](#))

```
norm.ppf(q=0.025, loc=80, scale=0.02/np.sqrt(6))
norm.ppf(q=0.975, loc=80, scale=0.02/np.sqrt(6))

## 79.98399696107882
## 80.01600303892118
```

- Die Fläche 0.05 des gesamten roten Bereiches heisst *Signifikanzniveau*.

Signifikanzniveau α

Das Signifikanzniveau α , gibt an, wie hoch das Risiko ist, das man bereit ist einzugehen, eine falsche Entscheidung zu treffen.

Wird kein Wert für das Signifikanzniveau angegeben, so wird für α ein Wert von 0.05 bzw. 0.01 verwendet.

Wir verwenden hier

$$\alpha = 0.05$$

Liegt der gemessene Mittelwert im roten Bereich in Abbildung 4.8, so zweifeln wir an der Nullhypothese

$$H_0 : \mu = 80$$

Wir sagen, wir *verwerfen* die Nullhypothese $\mu = 80$. Wir nennen diesen Bereich, wo die Nullhypothese verworfen wird, deshalb

Verwerfungsbereich

$$K = (-\infty, 79.984] \cup [80.016, \infty)$$

Wir gehen also davon aus, dass ein Mittelwert einer Messreihe im Verwerfungsbereich so unwahrscheinlich ist, dass wir an der Richtigkeit von $\mu = 80$ zweifeln und annehmen müssen, dass das wahre μ nicht 80 ist.

Nun können wir mit unserer Messreihe überprüfen, ob deren Mittelwert im Verwerfungsbereich liegt oder nicht und machen den sogenannten

Testentscheid

In unserem Beispiel hatten wir (siehe Abbildung 4.9)

$$\bar{X}_6 = 79.98 \in K$$

Dieser Wert liegt im Verwerfungsbereich. Also gehen wir nicht vom wahren $\mu = 80$ aus, da der Mittelwert der Messreihe nicht zu diesem Parameter passt.

Das heisst, dieser Wert ist zu unwahrscheinlich, als dass $\mu = 80$ plausibel ist.

Wir verwerfen also die Nullhypothese und nehmen die Alternativhypothese an, d.h.

$$\mu \neq 80$$

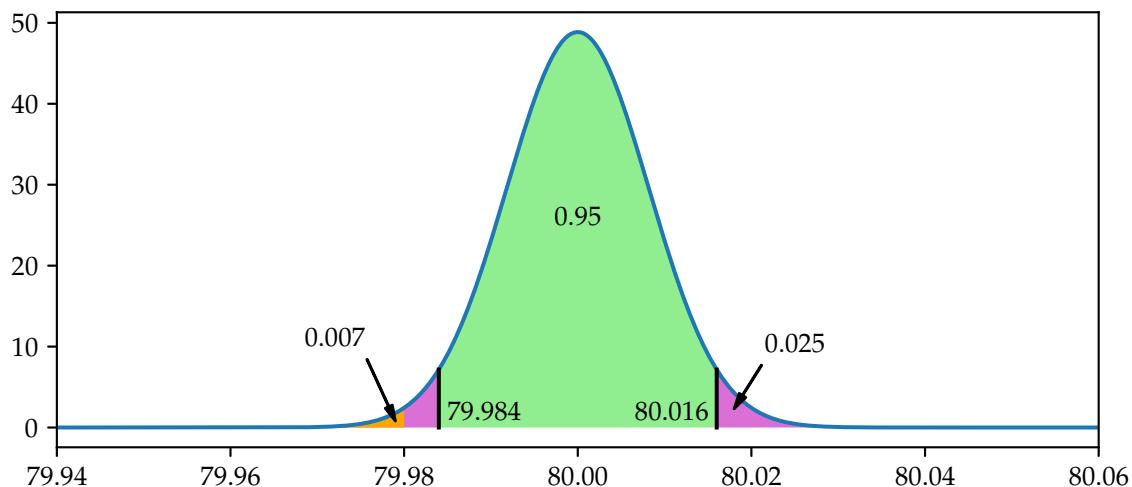


Abbildung 4.9.: Verwerfungsbereich 2

□

Bemerkungen:

- i. Warum haben wir hier den Verwerfungsbereich nach oben und nach unten aufgeteilt, wenn wir schon wissen, dass der gemessene Mittelwert kleiner als $\mu = 80$ ist? Nun, das wussten wir *vor* der Messung nicht. Der gemessene Mittelwert hätte also durchaus auch grösser als $\mu = 80$ sein können (siehe Beispiel 4.3.4). Wir sprechen in diesem Fall von einem *zweiseitigen Test*.
- ii. Es gibt auch *einseitige Tests* (siehe Beispiel 4.3.11).
- iii. Wir haben hier eine *Abmachung* getroffen, das der gesamte Verwerfungsbereich 5 % betragen soll. Diese Abmachung hat sich als praktisch erwiesen, aber wir hätten auch 1 % wählen können, was auch ab und zu gemacht wird. Der Wert des Signifikanzniveaus hängt von der jeweiligen wissenschaftlichen Disziplin ab. In der Psychologie und den Geisteswissenschaften wird α in der Regel relativ hoch gewählt, in der Teilchenphysik sehr klein.

iv. Im Beispiel 4.3.5 folgt die simulierte Messreihe allerdings tatsächlich der Normalverteilung $\mathcal{N}(80, 0.02^2)$ und doch haben wir den Parameter $\mu = 80$ verworfen. Dies bedeutet, wir haben im Rahmen unseres Testentscheids einen *Fehler* gemacht. Auf die Problematik von Fehlentscheiden gehen wir im Abschnitt 4.3.5 ein.

Beispiel 4.3.9

Betrachten wir eine *unterschiedliche* Messreihe, z.B. diejenige in Beispiel 4.3.4, so ist Modell, Nullhypothese, Alternativhypothese, Teststatistik, Signifikanzniveau und Verwerfungsbereich gleich wie im Beispiel 4.3.8 zuvor. Wir brauchen also bloss noch den Testentscheid durchzuführen.

```
## 0     80.07
## 1     80.06
## 2     80.03
## 3     80.03
## 4     80.02
## 5     80.03
## dtype: float64
## Mittelwert: 80.04
```

Der geschätzte Mittelwert fällt in den Verwerfungsbereich und somit wird auch hier die Nullhypothese verworfen.

Es gilt für die Wahrscheinlichkeit ([zu R](#))

$$P(\bar{X}_6 > 80.04)$$

```
1 - norm.cdf(x=80.04, loc=80, scale=0.02/np.sqrt(6))

## 4.816785043049165e-07
```

Sie ist bei weitem kleiner als 0.025 und damit so unwahrscheinlich, dass wir auch auf diese Weise $\mu = 80$ als nicht richtig annehmen (müssen). Wir *verwerfen* die Nullhypothese.

Im Gegensatz zum Verwerfungsbereich, wo nur die Entscheidung gefällt wird, ob der geschätzte Mittelwert im Verwerfungsbereich liegt oder nicht, macht der Wert von $P(\bar{X}_6 > 80.04)$ noch eine Aussage über die „Sicherheit“ unseres Testentscheids. In diesem Fall ist $5 \cdot 10^{-7}$ sehr viel kleiner als 0.025, und damit können wir mit grosser Sicherheit davon ausgehen, dass $\mu = 80$ nicht gilt. Wir kommen beim *P*-Wert auf diesen Sachverhalt zurück.

Aber auch hier sei nochmals erwähnt, dass diese Messreihe von der tatsächlichen Verteilung $\mathcal{N}(80.00, 0.02^2)$ stammt. Allerdings ist sie so unwahrscheinlich, dass wir an der Annahme (Hypothese) $\mu = 80$ zweifeln müssen.

□

Beispiel 4.3.10

Wir wollen nun testen, ob die Angabe der Herstellerfirma in Beispiel 4.3.6 mit der tatsächlich beobachteten Messreihe kompatibel ist.

Die Herstellerfirma behauptet, dass die Maschine die Büchsen normalverteilt mit $\mu = 500 \text{ ml}$ und $\sigma = 1 \text{ ml}$ abfüllt. Die Brauerei erhebt 100 Stichproben. Der Mittelwert dieser Stichproben ergibt sich zu 499.57 ml. Wir nehmen an, die Messungen sind normalverteilt mit bekanntem $\sigma = 1$.

Modell

X_i : Inhalt der i -ten Büchse

$$X_1, \dots, X_{100} \text{ i.i.d. } \sim \mathcal{N}(\mu, 1^2)$$

Nullhypothese

$$H_0 : \mu_0 = 500$$

Alternativhypothese

$$H_A : \mu \neq \mu_0 = 500$$

Teststatistik mit Signifikanzniveau $\alpha = 0.05$

$$\bar{X}_{100} \sim \mathcal{N}\left(500, \frac{1^2}{100}\right)$$

Verwerfungsbereich

Die Grenze des Verwerfungsbereichs ermitteln wir durch (zu R)

```
norm.ppf(q=[0.025, 0.975], loc=500, scale=1/np.sqrt(100))

## [499.8040036 500.1959964]
```

Also

$$K = (-\infty, 499.804) \cup (500.196, \infty)$$

Testentscheid

Es gilt

$$499.57 \in K$$

Somit wird die Nullhypothese verworfen. Wir können der Angabe des Herstellers der Abfüllanlage *nicht* vertrauen.

□

Beispiel 4.3.11

Wir kommen auf das Beispiel 4.3.7 zurück. Das Bundesamt für Sport behauptet, dass die durchschnittliche Körpergrösse der erwachsenen Frauen in der Schweiz bei 180 cm mit einer Standardabweichung von 10 cm liegt. Unsere Vermutung ist, dass dieser Wert zu gross ist. Hier macht ein zweiseitiger Test wenig Sinn, da wir „wissen“, dass dieser Mittelwert zu gross ist. Das heisst, der wahre Wert liegt wohl eher tiefer.

Die Überlegung ist an sich dieselbe wie in Beispiel 4.3.8, wobei wir aber den Verwerfungsbereich nicht auf beide Seiten auslegen, sondern nur nach unten, da wir erwarten, dass der wahre Mittelwert tiefer als $\mu = 180$ ist (siehe Abbildung 4.10). Wir machen einen *einseitigen* Test.

Wir wählen zufällig 8 erwachsene Frauen aus, deren durchschnittliche Körpergrösse 171.54 cm beträgt. Es wird angenommen, dass die Körpergrösse normalverteilt ist mit $\mathcal{N}(\mu, 10^2)$. Wir nehmen weiter an, dass die Standardabweichung dieselbe ist, wie vom Bundesamt für Sport angegeben.

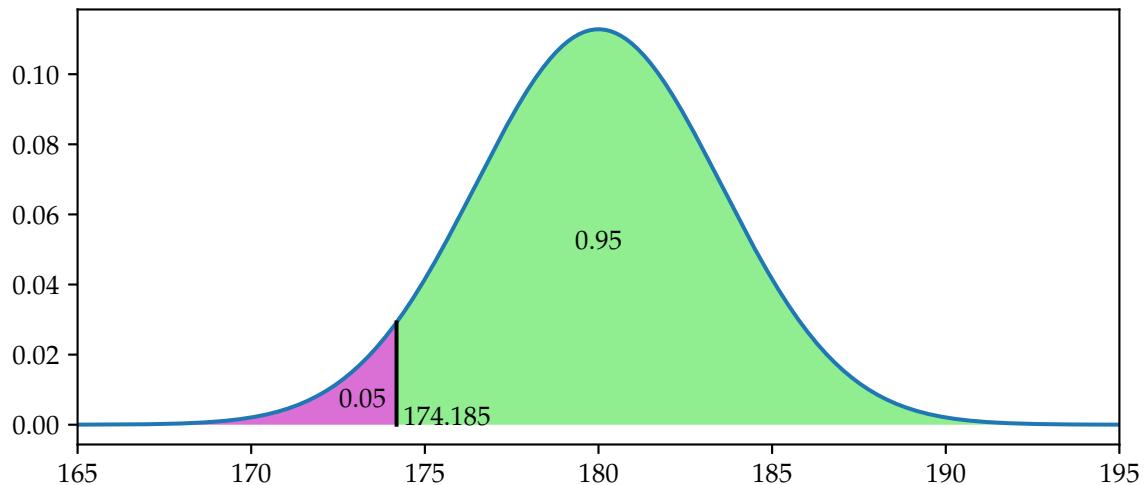


Abbildung 4.10.: Verwerfungsbereich Körpergrösse

Modell:

X_i : Körpergrösse der i -ten Frau. Es gilt

$$X_1, \dots, X_8 \text{ i.i.d. } \sim \mathcal{N}(\mu, 10^2)$$

Um die Nullhypothese zu überprüfen, gehen wir von der Annahme aus, dass der wahre Mittelwert wirklich 180 cm ist.

Nullhypothese

$$H_0 : \mu_0 = 180$$

Alternativhypothese

Da wir die (ziemlich starke) Vermutung hegen, dass der Mittelwert μ kleiner als 180 ist, lautet die Alternativhypothese:

$$H_A : \mu < \mu_0 = 180$$

Teststatistik

Die Teststatistik, mit der wir die Nullhypothese testen, ist in diesem Beispiel gegen durch \bar{X}_8 .

Falls die Nullhypothese zutrifft, d.h., falls das Bundesamt für Sport eine korrekte Aussage gemacht hat, gilt

$$\bar{X}_8 \sim \mathcal{N}\left(, \frac{10^2}{8}\right)$$

Signifikanzniveau und Verwerfungsbereich

Nun führen wir eine Untersuchung unter $n = 8$ Personen durch. Wir ermitteln also in einer Messreihe den Wert \bar{x}_8 und testen, ob die Wahrscheinlichkeit

$$P(\bar{X}_8 < \bar{x}_8)$$

kleiner als das Signifikanzniveau $\alpha = 0.05$ ist oder nicht.

Der Verwerfungsbereich zeigt hier also einseitig nach unten. In Abbildung 4.10) ist der Verwerfungsbereich für $n = 8$ pink eingezeichnet.

Die Grenze des Verwerfungsbereichs (siehe Abbildung 4.10) ermitteln wir durch

```
norm.ppf(q=0.05, loc=180, scale=10/np.sqrt(8))
## 174.18456423161663
```

Somit ist der Verwerfungsbereich

$$K = (-\infty, 174.185)$$

Dieser Verwerfungsbereich ist natürlich viel zu gross, da wohl kaum Körpergrössen von erwachsenen Frauen unter 50 cm zu erwarten sind. Wir arbeiten hier mit einem *Modell*, das eben nur in einem bestimmten Bereich Sinn macht.

Testentscheid

Der beobachtete Mittelwert 171.54 cm fällt also in den Verwerfungsbereich, und somit verwerfen wir die Nullhypothese $\mu_0 = 180$.

Dieser Mittelwert der zufällig ausgewählten acht Frauen erscheint immer noch relativ hoch, aber er reicht schon, damit wir an der Annahme $\mu = 180$ zweifeln müssen.

Der Wert für $P(\bar{X}_6 < 171.54)$ ist (siehe Abbildung 4.11) (zu R) .

$$P(\bar{X}_6 < 171.54) = 0.008$$

```
norm.cdf(x=171.54, loc=180, scale=10/np.sqrt(8))

## 0.008359052027838012
```

Dieser Wert heisst P -Wert und stellt ein Mass für die Sicherheit dar, mit der wir den Testentscheid treffen. Wird die Nullhypothese verworfen, so deutet ein sehr kleiner P -Wert darauf hin, dass unser Testentscheid sicherer ist, als wenn er in der Nähe des Signifikanzniveaus (hier $\alpha = 0.05$) liegen würde.

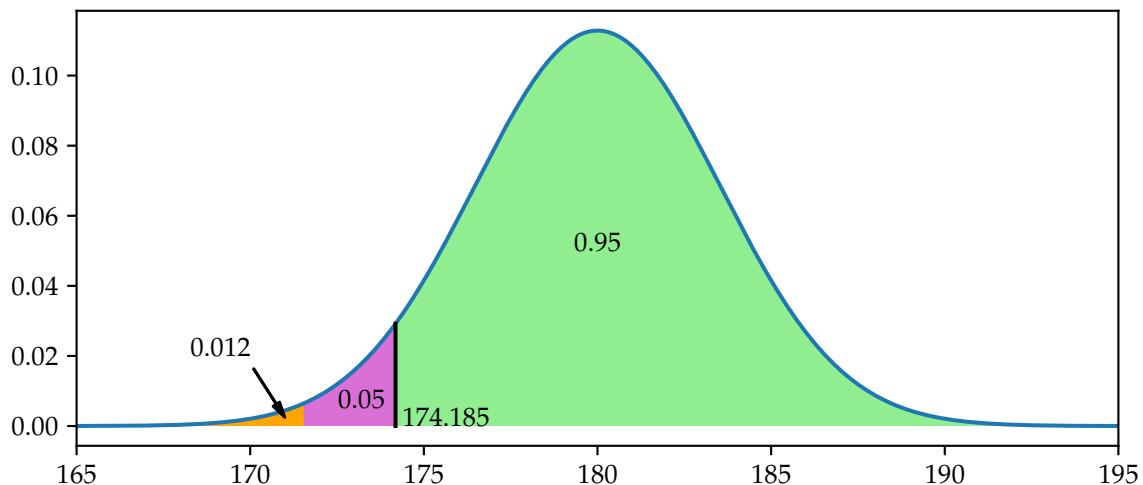


Abbildung 4.11.: Verwerfungsbereich für das Beispiel, bei welchem die erwartete Körpergrösse von Frauen 180 cm betragen soll.

□

Wir haben in Beispiel 4.3.8 die Grösse $\mu = 80$ verworfen, da die Messreihe einen zu tiefen Mittelwert lieferte. In Beispiel 4.3.12 haben wir die Grösse $\mu = 80$ verworfen, weil der beobachtete Mittelwert (viel) zu gross ist. In beiden Fällen waren die Mittelwerte so unwahrscheinlich, so dass wir am hypothetischen Mittelwert $\mu = 80$ zweifeln müssen.

Wie sieht es nun aber aus, wenn wir eine neue Messreihe bilden, die aus *beiden* Messreihen `sim_2` und `sim_3` besteht? Diese Messreihe hat dann die Länge 12. Oder anders gefragt: Welchen Einfluss hat die Anzahl der Messungen auf den Verwerfungsbereich?

Beispiel 4.3.12

Wir wollen dies am Beispiel ?? untersuchen und gehen wie folgt vor: Wir haben Messreihen verschiedener Längen n , die alle den (beobachteten) Mittelwert $\bar{x}_6 = 79.78$ haben. Dann bestimmen wir für alle Messreihen den Wert

$$P(\bar{X}_n \leq 79.98)$$

mit

$$\bar{X}_6 \sim \mathcal{N}\left(80, \frac{0.02^2}{n}\right)$$

Ist dieser Wert grösser als 0.025, dann wird die Nullhypothese nicht verworfen, ansonsten schon.

Für $n = 2$ erhalten wir folgenden Wert für

$$P(\bar{X}_2 \leq 79.98) = 0.079 > 0.025$$

Die Nullhypothese wird also nicht verworfen. Bei 2 Messwerten erachten wir die Abweichung vom wahren Mittelwert als zufällig möglich. ([zu R](#))

```
norm.cdf(x=79.98, loc=80, scale=0.02/np.sqrt(2))

## 0.07864960352518385
```

Für $n = 4$ erhalten wir

$$P(\bar{X}_4 \leq 79.98) = 0.022 < 0.025$$

Hier wird die Nullhypothese (knapp) verworfen. ([zu R](#))

```
norm.cdf(x=79.98, loc=80, scale=0.02/np.sqrt(4))

## 0.022750131948200674
```

Für $n = 6$ erhalten wir

$$P(\bar{X}_6 \leq 79.98) = 0.007 < 0.025$$

Die Nullhypothese wird klarer verworfen als für $n = 4$. ([zu R](#))

```
norm.cdf(x=79.98, loc=80, scale=0.02/np.sqrt(6))

## 0.007152939217724509
```

Und schliesslich noch für $n = 8$:

$$P(\bar{X}_6 \leq 79.98) = 0.002 < 0.025$$

Die Nullhypothese wird noch klarer verworfen, als bei $n = 6$. ([zu R](#))

```
norm.cdf(x=79.98, loc=80, scale=0.02/np.sqrt(8))
## 0.0023388674905277422
```

Mit zunehmendem n wird der Wert

$$P(\bar{X}_n \leq 79.98)$$

immer kleiner. Dies liegt daran, dass der Standardfehler mit grösser werdendem n kleiner wird, und damit werden die Normalverteilungskurven schmäler (siehe Abbildung 4.12). Je mehr Messungen wir also haben, desto gewichtiger ist eine Abweichung vom wahren Mittelwert.

□

4.3.3. Der P-Wert

Der *P*-Wert ist ein Wert zwischen 0 und 1, der angibt, wie gut die *Nullhypothese* und *Daten* zusammenpassen (0: passt gar nicht; 1: passt sehr gut). Etwas präziser formuliert, definieren wir den *P*-Wert als die Wahrscheinlichkeit, unter Gültigkeit der Nullhypothese das erhaltene Ergebnis oder ein *extremeres* zu erhalten (siehe Abbildung 4.13).

Mit dem *P*-Wert wird also angedeutet, wie extrem das Ergebnis ist: Je kleiner der *P*-Wert, desto mehr spricht das Ergebnis gegen die Nullhypothese. Werte kleiner als eine im voraus festgesetzte Grenze, wie 5 %, 1 % oder 0.1 % sind Anlass, die Nullhypothese abzulehnen.

P-Wert

Der *P*-Wert ist die Wahrscheinlichkeit, unter der Nullhypothese ein mindestens so extremes Ereignis (in Richtung der Alternative) zu beobachten wie das aktuell beobachtete.

Wir können den Testentscheid auch mit Hilfe des *P*-Wertes durchführen.

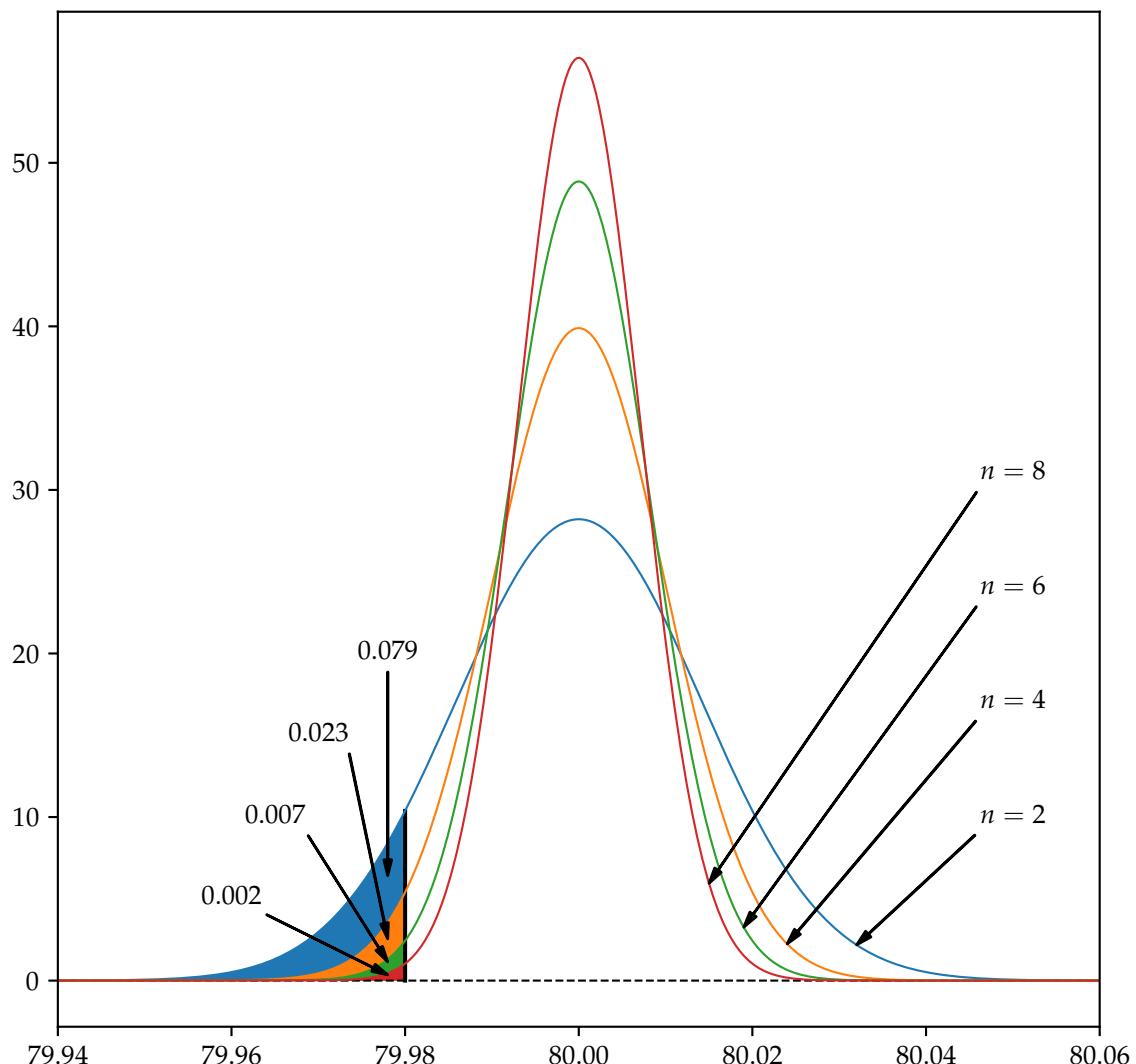


Abbildung 4.12.: P -Werte in Abhängigkeit von der Anzahl Messungen n in der Messreihe bei festem beobachteten Mittelwert.

P -Wert und Statistischer Test

Man kann anhand des P -Wertes direkt den Testentscheid ablesen: Wenn der P -Wert kleiner als das Signifikanzniveau α ist, so verwirft man H_0 , ansonsten nicht. Verglichen mit dem blossen Testentscheid enthält der P -Wert aber mehr Information, da man direkt sieht, „wie stark“ die Nullhypothese verworfen wird. Bei einem vorgegebenen Signifikanzniveau α (z.B. $\alpha = 0.05$) gilt aufgrund der Definition des P -Werts für einen einseitigen Test:

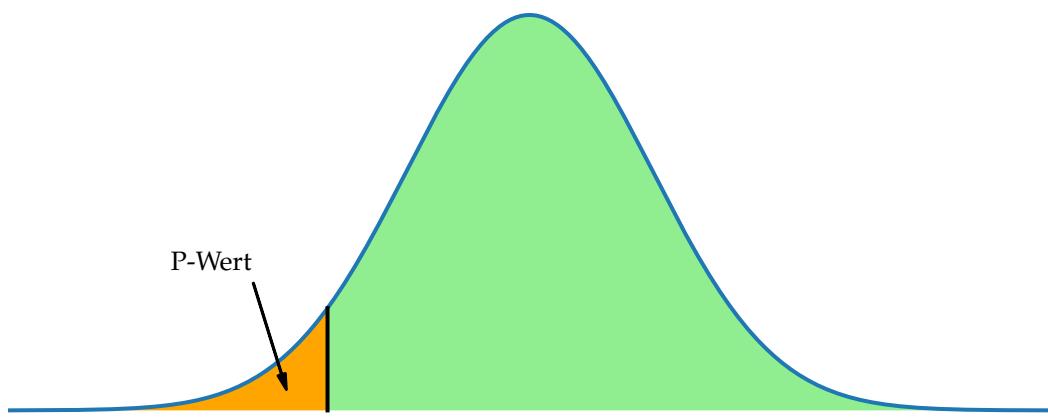


Abbildung 4.13.: P -Wert

1. Verwerfe H_0 falls P -Wert $\leq \alpha$
2. Belasse H_0 falls P -Wert $> \alpha$

Viele Computer-Pakete liefern den Testentscheid nur indirekt, indem der P -Wert angegeben wird. Man kann sich den P -Wert auch als „vollstandardisierte“ Teststatistik vorstellen.

Zusätzlich zu dieser Entscheidungsregel quantifiziert der P -Wert, *wie signifikant* eine Alternative ist (d.h. wie gross die Evidenz ist für das Verwerfen von H_0). Manchmal werden sprachliche Formeln oder Symbole anstelle der P -Werte angegeben:

- P -Wert ≈ 0.05 : schwach signifikant, “.”
- P -Wert ≈ 0.01 : signifikant, “*”
- P -Wert ≈ 0.001 : stark signifikant, “**”
- P -Wert $\leq 10^{-4}$: "ausserst signifikant, “***”

Bemerkungen:

- i. *Achtung:* Der P -Wert ist nicht die Wahrscheinlichkeit, dass die Nullhypothese stimmt. Darüber können wir hier gar keine Aussagen machen, da die Parameter fix und nicht zufällig sind.

ii. Streng genommen handelt es sich beim P-Wert um *keine* Wahrscheinlichkeit. Denn die Wahrscheinlichkeit, aufgefasst als relative Auftretenshäufigkeit eines Ereignisses, macht hier keinen Sinn. Der P-Wert macht keine Aussage über die Auftretenshäufigkeit der Beobachtung oder eines extremeres Ereignisses, sondern bloss wie verträglich dieses Ereignis mit der Nullhypothese ist.

Strikte betrachtet ist der P-Wert einfach eine transformierte Zufallsvariable. Eine alternative und unmissverständlichere Definition des P-Wertes lautet:

Der P-Wert ist das kleinste Signifikanzniveau, bei dem die Nullhypothese H_0 (gerade noch) verworfen wird.

Diese Definition vermeidet es, den P-Wert als Wahrscheinlichkeit zu interpretieren. Nichtsdestotrotz werden wir im Folgenden im Zusammenhang mit dem P-Wert von einer Wahrscheinlichkeit sprechen.

Wir haben den P-Wert für einseitige Tests definiert. Wie sieht nun aber der P-Wert für zweiseitige Tests aus?

Beispiel 4.3.13

In Beispiel 4.3.8 haben wir die Wahrscheinlichkeit

$$P(\bar{X}_6 \leq 79.98) = 0.007$$

berechnet. Da aber in diesem Beispiel die Alternativhypothese zweiseitig ist, wird diese Wahrscheinlichkeit verdoppelt. Somit lautet der

$$\text{P-Wert} = 2 \cdot P(\bar{X}_6 \leq 79.98) = 0.014$$

Dieser P-Wert wird dann mit dem Signifikanzniveau $\alpha = 0.05$ verglichen.

□

4.3.4. Der z -Test (σ_X bekannt)

Wir wollen nun den Hypothesentest aus den Beispielen vorher schematisch festhalten.

Wir nehmen an, dass die Daten x_1, \dots, x_n Realisierungen von

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu, \sigma_X^2)$$

sind. Überdies machen wir die Annahme, dass σ_X^2 bekannt ist. Der z -Test für den Parameter μ erfolgt dann wie folgt.

Kapitel 4. Statistik für Messdaten

1. *Modell:* X_i ist eine kontinuierliche Messgrösse:

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu, \sigma_X^2), \quad \sigma_X \text{ bekannt}$$

2. *Nullhypothese:*

$$H_0 : \mu = \mu_0$$

Alternative:

$$H_A : \mu \neq \mu_0 \quad (\text{oder } < \text{ oder } >)$$

3. *Teststatistik: Verteilung der Teststatistik unter H_0 :*

$$T : \bar{X}_n \sim \mathcal{N}\left(\mu_0, \frac{\sigma_X^2}{n}\right)$$

4. *Signifikanzniveau:*

$$\alpha$$

5. *Verwerfungsbereich für die Teststatistik:*

$$K = (-\infty, x_{\frac{\alpha}{2}}] \cup [x_{1-\frac{\alpha}{2}}, \infty) \text{ bei } H_A : \mu \neq \mu_0,$$

$$K = (-\infty, x_\alpha] \text{ bei } H_A : \mu < \mu_0,$$

$$K = [x_{1-\alpha}, \infty) \text{ bei } H_A : \mu > \mu_0$$

wobei

$$x_{\alpha/2}$$

das $\alpha/2$ -Quantil bezeichnet.

6. *Testentscheid:*

Überprüfe, ob der beobachtete Wert der Teststatistik im Verwerfungsbereich liegt.

Und hier noch die standardisierte Variante:

1. *Modell:* X_i ist eine kontinuierliche Messgrösse;

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu, \sigma_X^2), \quad \sigma_X \text{ bekannt}$$

2. *Nullhypothese:*

$$H_0 : \mu = \mu_0$$

Alternative:

$$H_A : \mu \neq \mu_0 \quad (\text{oder } < \text{ oder } >)$$

3. Teststatistik:

$$Z = \frac{(\bar{X}_n - \mu_0)}{\sigma_{\bar{X}_n}} = \frac{(\bar{X}_n - \mu_0)}{\sigma_X / \sqrt{n}} = \frac{\text{beobachtet} - \text{erwartet}}{\text{Standardfehler}}$$

Verteilung der Teststatistik unter H_0 :

$$Z \sim \mathcal{N}(0, 1)$$

4. Signifikanzniveau:

$$\alpha$$

5. Verwerfungsbereich für die Teststatistik:

$$K = (-\infty, z_{\frac{\alpha}{2}}] \cup [z_{1-\frac{\alpha}{2}}, \infty) \text{ bei } H_A : \mu \neq \mu_0,$$

$$K = (-\infty, z_{\alpha}] \text{ bei } H_A : \mu < \mu_0,$$

$$K = [z_{1-\alpha}, \infty) \text{ bei } H_A : \mu > \mu_0,$$

wobei

$$z_{\alpha/2} = \Phi^{-1}(\alpha/2)$$

6. Testentscheid:

Überprüfe, ob der beobachtete Wert der Teststatistik im Verwerfungsbereich liegt.

Bemerkungen:

- i. Bevor Software zur Verfügung stand, musste man standardisieren.
- ii. Wegen der Variable z wird der Test z -Test genannt.

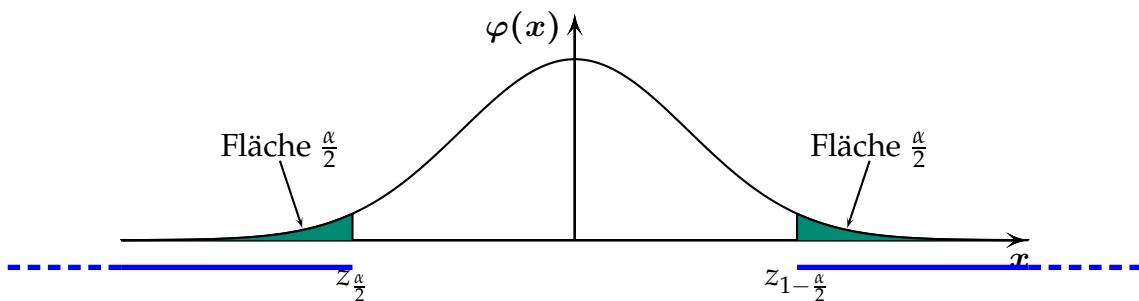


Abbildung 4.14.: Dichtefunktion der Teststatistik Z mit Verwerfungsbereich (blau) des zweiseitigen Z -Tests zum Niveau α . Beachte $z_{\alpha/2} = -z_{1-\alpha/2}$, wobei $z_{\alpha/2} = \Phi^{-1}(\alpha/2)$ ist.

Der z -Test basiert auf *mehreren* Beobachtungen. Denn die realisierte Teststatistik z fasst die Beobachtungen in Form des arithmetischen Mittelwertes zusammen.

Beispiel 4.3.14

Bei Methode A (vgl. Beispiel 4.3.1) scheint die Schmelzwärme grösser als 80.00 zu sein. Angenommen, wir wissen aus vorhergehenden Studien, dass die Standardabweichung unseres Messinstruments $\sigma_X = 0.01$ ist. Ist es plausibel, dass die wahre Schmelzwärme 80.00 cal/g ist? Wir führen dazu einen z-Test durch:

1. *Modell:* X_i ist eine kontinuierliche Messgrösse;

$$X_1, \dots, X_n \text{ i.i.d.} \sim \mathcal{N}(\mu, \sigma_X^2), \quad \sigma_X = 0.01 \text{ bekannt}, n = 13$$

2. *Nullhypothese:*

$$H_0 : \mu = \mu_0 = 80.00$$

Alternative:

$$H_A : \mu \neq \mu_0$$

3. *Teststatistik:*

Der Mittelwert der Messungen

$$T : \bar{X}_n$$

Verteilung der Teststatistik unter H_0 :

$$T \sim \mathcal{N}\left(\mu_0, \frac{\sigma_X^2}{n}\right) = \mathcal{N}\left(80, \frac{0.01^2}{13}\right)$$

4. *Signifikanzniveau:*

$$\alpha = 0.05$$

5. *Verwerfungsbereich für die Teststatistik:*

$$K = (-\infty, c_u] \cup [c_l, \infty) \quad \text{bei } H_A : \mu \neq \mu_0,$$

Mit **Python** erhalten wir mit $\alpha = 0.05$: ([zu R](#))

```
norm.ppf(q=[0.025, 0.975], loc=80, scale=0.01/np.sqrt(13))
```

```
## [79.99456404 80.00543596]
```

oder einfacher

```
norm.interval(alpha=0.95, loc=80, scale=0.01/np.sqrt(13))
```

```
## (79.9945640379659, 80.0054359620341)
```

Damit erhalten wir den Verwerfungsbereich der Teststatistik:

$$K = (-\infty, 80.00] \cup [80.01, \infty)$$

6. *Testentscheid:*

Aus den $n = 13$ Daten errechnen wir

$$\bar{x}_n = 80.02$$

Der beobachtete Wert liegt im Verwerfungsbereich der Teststatistik. Daher wird die Nullhypothese auf dem 5 % Signifikanzniveau verworfen.

Und noch die standardisierte Variante:

1. *Modell:* X_i ist eine kontinuierliche Messgrösse;

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu, \sigma_X^2), \quad \sigma_X = 0.01 \text{ bekannt}, \quad n = 13$$

2. *Nullhypothese:*

$$H_0 : \mu = \mu_0 = 80.00$$

Alternative:

$$H_A : \mu \neq \mu_0$$

3. *Teststatistik:*

$$Z = \frac{\sqrt{n}(\bar{X}_n - \mu_0)}{\sigma_X}$$

Verteilung der Teststatistik unter H_0 :

$$Z \sim \mathcal{N}(0, 1)$$

4. *Signifikanzniveau:*

$$\alpha = 0.05$$

5. *Verwerfungsbereich für die Teststatistik:*

$$K = (-\infty, z_{\frac{\alpha}{2}}] \cup [z_{1-\frac{\alpha}{2}}, \infty) \quad \text{bei } H_A : \mu \neq \mu_0,$$

Mit dem Computer erhalten wir mit $\alpha = 0.05$:

$$z_{\frac{\alpha}{2}} = \Phi^{-1}\left(\frac{\alpha}{2}\right) = \Phi^{-1}(0.025) = -1.96$$

Damit ergibt sich für den Verwerfungsbereich der Teststatistik

$$K = (-\infty, -1.96] \cup [1.96, \infty)$$

6. Testentscheid:

Aus den $n = 13$ Daten errechnen wir

$$\bar{x}_n = 80.02$$

Damit ergibt sich als Wert für die Teststatistik

$$z = \frac{\sqrt{13}(80.02 - 80.00)}{0.01} = 7.211$$

Der beobachtete Wert liegt im Verwerfungsbereich der Teststatistik. Daher wird die Nullhypothese auf dem 5 % Signifikanzniveau verworfen.

Bemerkungen:

- i. Hier ist der Umweg über die Standardisierung wegen der technischen Hilfsmittel (Computersoftware) an sich unnötig.

□

4.3.5. Fehler 1. und 2. Art beim statistischen Test

Verwerfen wir eine Nullhypothese, so heisst das, dass das beobachtete Ereignis unter dieser Hypothese so unwahrscheinlich ist, dass wir Zufall ausschliessen können. Allerdings ist es immer noch möglich, dass die Nullhypothese *richtig* ist und wir einfach ein sehr unwahrscheinliches Ereignis beobachtet haben. Wir haben dann mit dem Verwerfen der Nullhypothese einen *Fehler* gemacht. In diesem Fall sprechen wir von einem *Fehler 1. Art*. Auf der anderen Seite können wir die Nullhypothese beibehalten, obwohl die Alternativhypothese richtig ist. Dies nennen wir einen *Fehler 2. Art*. Bei einem statistischen Test treten also 2 Arten von Fehlern auf:

- *Fehler 1. Art*: Fälschliches Verwerfen von H_0 , obwohl H_0 richtig ist.
- *Fehler 2. Art*: Fälschliches Beibehalten von H_0 , obwohl die Alternative zutrifft.

Wir werden uns zuerst mit dem Fehler 1. Art auseinandersetzen. Formal lautet die Definition

Fehler 1. Art

Der *Fehler 1. Art* ist definiert als

$$P(\text{Fehler 1. Art}) = P_{H_0}(T \in K) \leq \alpha$$

Nach Konstruktion des Tests ist die Wahrscheinlichkeit für einen Fehler 1. Art

höchstens gleich α .

Betrachten wir nun im Folgenden den Fehler 2. Art, der gegeben ist durch das fälschliche Beibehalten von H_0 , obwohl die Alternative zutrifft. Formal lautet die Definition

Fehler 2. Art

Der *Fehler 2. Art* ist definiert als

$$P(\text{Fehler 2. Art}) = P_{H_A}(T \in \bar{K})$$

Welche Fehlerart ist wichtiger?

Dem Fehler 1. Art wird traditionell mehr Gewicht gegeben als dem Fehler 2. Art: Wissenschaftler arbeiten genau und haben Angst, einen Humbug zu publizieren, der sich dann als falsch herausstellt. Denn wenn Wissenschaftler einen Effekt (signifikante Abweichung von Nullhypothese) beobachten, möchten sie sicher sein, dass es sich nicht bloss um Zufall handelt. Der Fehler 1. Art soll vermieden werden. Dabei nimmt man in Kauf, dass man manchmal einen wichtigen Effekt verpasst. Der Fehler 2. Art ist also zweitrangig.

Der Fehler 1. Art wird direkt kontrolliert mittels der Konstruktion eines Tests, indem α möglichst klein gehalten wird. Über die Wahrscheinlichkeit eines Fehlers 2. Art haben wir hingegen keine solche Kontrolle. Die beiden Fehlerarten konkurrieren sich gegenseitig:

$P(\text{Fehler 2. Art})$ wird grösser falls α kleiner gewählt wird

Die Wahl von α steuert also einen Kompromiss zwischen Fehler 1. und 2. Art. Weil man aber primär einen Fehler 1. Art vermeiden will, wählt man α klein, z.B. $\alpha = 0.05$. Je kleiner α , desto kleiner der Verwerfungsbereich. Die vertikale Linie wandert also nach rechts und somit wird der Fehler 2. Art umso grösser.

Die Macht eines statistischen Tests

Statt der Wahrscheinlichkeit eines Fehlers 2. Art gibt man oft die sogenannte *Macht* eines Tests an. Die Macht gibt die Wahrscheinlichkeit an, H_A zu entdecken, falls H_A richtig ist.

Macht

Die *Macht* ist definiert als

$$\begin{aligned}\text{Macht} &= 1 - P(\text{Fehler 2. Art}) \\ &= P(\text{Verwerfen von } H_0 \text{ falls } H_A \text{ stimmt}) \\ &= P_{H_A}(T \in K)\end{aligned}$$

Bemerkungen:

- i. Auf englisch (und manchmal auch auf deutsch) wird ein Fehler 1. Art *False Positive* und ein Fehler 2. Art *False Negative* genannt. Die Macht wird als *power* übersetzt.

Beispiel 4.3.15

Wir betrachten die Körpergrösse von Frauen, die normalverteilt ist mit $\mu = 168$. Wir wollen die Hypothese, dass die Körpergrösse von Männern grösser ist als die von Frauen. Die Nullhypothese in Bezug auf die Körpergrösse von Männern lautet also $H_0 : \mu_0 = 168$ und die Alternativhypothese ist

$$H_A : \mu > 168$$

In Abbildung 4.15 oben ist der Verwerfungsbereich eingezeichnet, der bei 172 cm liegt. Fällt nun der Mittelwert einer Messreihe der Körpergrösse von Männern über 172 cm, so wird die Nullhypothese verworfen, das heisst Männer sind in der Tat grösser.

Allerdings ist es möglich, dass der wahre Mittelwert tatsächlich bei $\mu = 168$ liegt und wir einfach zufällig grosse Männer ausgewählt haben, obwohl sie durchschnittlich kleiner sind. In diesem Fall haben wir einen Fehler gemacht, den Fehler 1. Art. Wir verwerfen die Nullhypothese, obwohl diese stimmt.

Liegt der wahre Mittelwert für die Körpergrösse von Männern aber tatsächlich bei $\mu = 180$, dann wurde für jeden Wert im Verwerfungsbereich die Nullhypothese richtigerweise verworfen (genau das, was wir wollen). Dies nennt man die *Macht* eines statistischen Tests und ist in Abbildung 4.15 in der Mitte eingezeichnet.

Der Fehler 2. Art ist in Abbildung 4.15 unten dargestellt.. Hier wird die Nullhypothese nicht verworfen, obwohl man es müsste.

□

Kapitel 4. Statistik für Messdaten

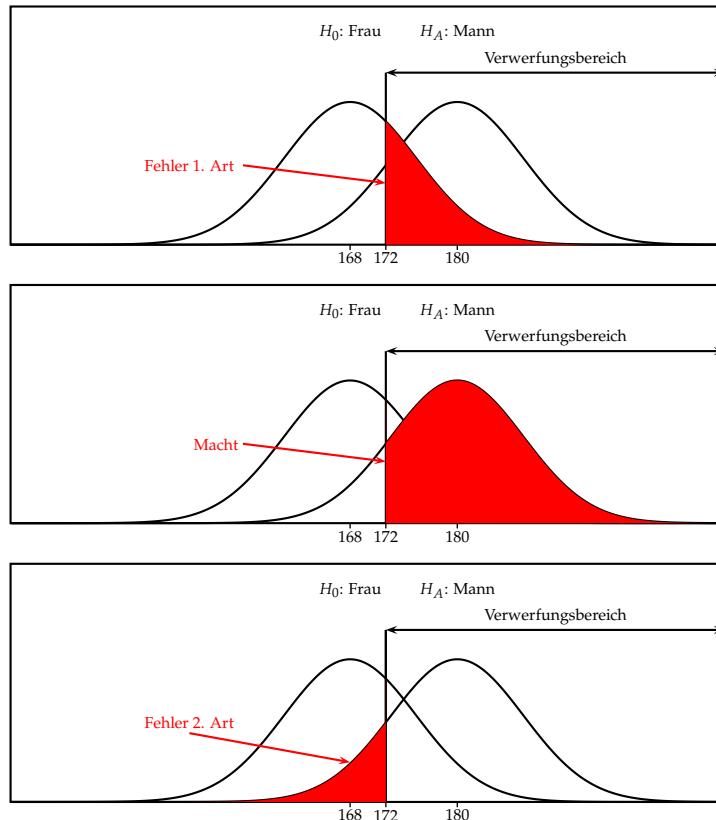


Abbildung 4.15.: Fehler 1. und 2. Art, Macht

4.3.6. Der t -Test (σ_X unbekannt)

Wie vorhin nehmen wir an, dass die Daten Realisierungen von

$$X_1, \dots, X_n \text{ i.i.d.} \sim \mathcal{N}(\mu, \sigma_X^2)$$

sind. In der Praxis ist die Annahme, dass σ_X bekannt ist, oftmals unrealistisch. In solchen Fällen kann die Teststatistik z nicht berechnet werden, weil sie auf σ_X basiert.

Allerdings können wir stattdessen die Schätzung

$$\hat{\sigma}_X^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2$$

benutzen. Dies führt aber zu einer zusätzlichen Unsicherheit, was zur Folge hat, dass sich die Verteilung der Teststatistik ändert.

t -Verteilung

Die Teststatistik beim t -Test ist gegeben durch

$$T = \frac{\bar{X}_n - \mu_0}{\hat{\sigma}_X / \sqrt{n}} = \frac{\text{beobachtet - erwartet}}{\text{geschätzter Standardfehler}}$$

Deren Verteilung unter der Nullhypothese

$$H_0 : \mu = \mu_0$$

ist eine sogenannte t -Verteilung mit $n - 1$ Freiheitsgraden, die wir mit

$$t_{n-1}$$

bezeichnen.

Die t_n -Verteilung² ist eine symmetrische Verteilung um 0, welche langschwänziger ist als die Standardnormalverteilung $\mathcal{N}(0, 1)$ (siehe Abbildung 4.16). Für

$$Y \sim t_n$$

gilt:

$$\begin{aligned} E(Y) &= 0 \\ \text{Var}(Y) &= \frac{n}{n-2} \end{aligned}$$

Für grosse n ist t_n ähnlich zu $\mathcal{N}(0, 1)$. Insbesondere strebt die t_n -Verteilung für $n \rightarrow \infty$ gegen die Standardnormalverteilung $\mathcal{N}(0, 1)$.

Beispiel 4.3.16

Der folgende Datensatz besteht aus normalverteilten Datenpunkten x_1, \dots, x_{20}

5.9	3.4	6.6	6.3	4.2	2.0	6.0	4.8	4.2	2.1
8.7	4.4	5.1	2.7	8.5	5.8	4.9	5.3	5.5	7.9

Wir vermuten, dass unsere Daten x_1, x_2, \dots, x_{20} Realisierungen von

$$X_i \sim \mathcal{N}(5, \sigma_X)$$

sind, wobei wir σ_X nicht kennen. Wir müssen σ_X also aus den Daten schätzen. ([zu R](#))

²Eine genaue Herleitung der t -Verteilung wird im Kapitel B.2 des Anhangs gegeben.

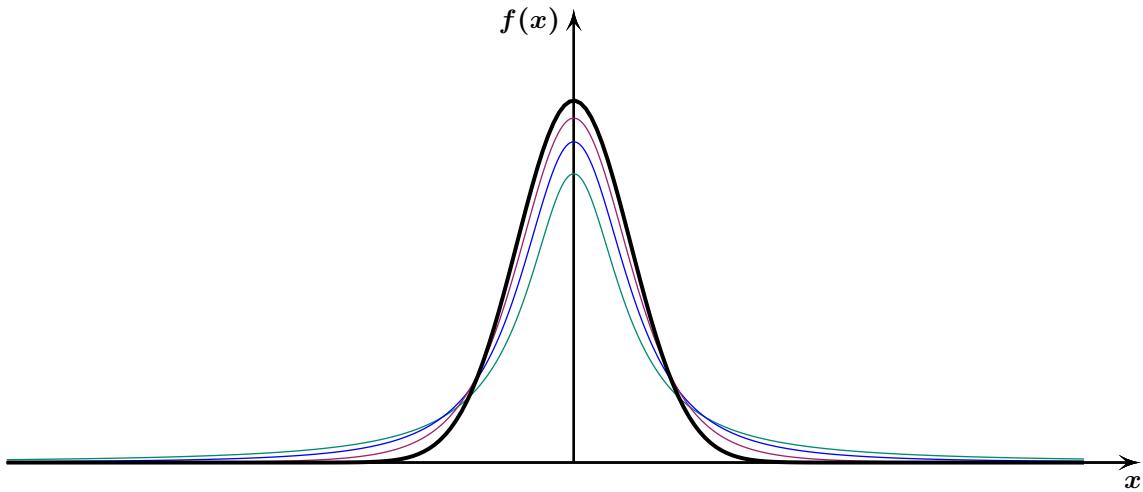


Abbildung 4.16.: Dichten der t -Verteilung mit 1 (grün), 2 (blau) und 5 (violett) Freiheitsgraden. Die schwarze Kurve ist die Dichte der Standardnormalverteilung.

```
from scipy.stats import norm, t
import numpy as np
from pandas import Series

x = Series([5.9, 3.4, 6.6, 6.3, 4.2, 2.0, 6.0, 4.8, 4.2, 2.1, 8.7, 4.4, 5.1, 2.7])

x.std()

## 1.8838021462764578
```

Die Nullhypothese lautet in diesem Fall $\mu_0 = 5$. Wie gross ist die Wahrscheinlichkeit, dass der Mittelwert eines Datensatzes mit Verteilung

$$X_i \sim \mathcal{N}(5, \hat{\sigma}_X)$$

kleiner ist als der Mittelwert unseres Datensatzes, wobei $\hat{\sigma}_X$ die aus unseren Daten geschätzte Standardabweichung ist? Wir möchten also die Wahrscheinlichkeit

$$P(\bar{X}_{20} \leq \bar{x}_{20})$$

bestimmen, wobei

$$\bar{X}_{20} \sim \mathcal{N}\left(5, \frac{\hat{\sigma}_X^2}{20}\right)$$

ist. \bar{x}_{20} bezeichnet den arithmetischen Mittelwert unseres Datensatzes und $\hat{\sigma}_X$ ist die geschätzte empirische Standardabweichung des Datensatzes. (zu R)

Kapitel 4. Statistik für Messdaten

```
t.cdf(x=x.mean(), df=x.size-1, loc=5, scale=x.std()/np.sqrt(x.size))

## 0.6921780567888249
```

Bemerkungen:

- i. Bei **Python** muss der standardisierte T -Wert *nicht* berechnet werden.
- ii. Bei **R** muss der T -Wert standardisiert werden (siehe folgende Ausführung).

Um diese Wahrscheinlichkeit zu berechnen, standardisieren wir den Mittelwert, indem wir von \bar{X}_n den erwarteten Mittelwert $\mu_0 = 5$ subtrahieren, und dann durch den geschätzten Standardfehler $\hat{\sigma}_X / \sqrt{n}$ dividieren. Wir berechnen die Wahrscheinlichkeit

$$P(\bar{X}_{20} \leq \bar{x}_{20})$$

folglich mittels

$$\begin{aligned} P\left(\frac{\bar{X}_n - 5}{\hat{\sigma}_X / \sqrt{n}} \leq \frac{\bar{x}_{20} - 5}{\hat{\sigma}_X / \sqrt{n}}\right) &= P\left(T \leq \frac{5.2 - 5}{1.9 / \sqrt{20}}\right) \\ &= P(T \leq 0.51) \\ &= 0.69 \end{aligned}$$

Dabei haben wir benutzt, dass T einer t_{20-1} -Verteilung folgt. ([zu R](#))

```
x = Series([5.9, 3.4, 6.6, 6.3, 4.2, 2.0, 6.0, 4.8, 4.2, 2.1, 8.7, 4.4, 5.1, 2.7,
mean_x = x.mean()

std_x = x.std()

t_x = (mean_x-5) / (std_x/np.sqrt(x.size))

t.cdf(x=t_x, df=x.size-1)

## 0.6921780567888249
```

□

Den Verwerfungsbereich beim Test erhalten wir, indem wir einen Bereich der Wahrscheinlichkeit α bei der t_{n-1} -Verteilung abschneiden (je nach Alternative auf einer

Kapitel 4. Statistik für Messdaten

Seite, oder je die Hälfte auf beiden Seiten). Dazu brauchen wir die Quantile $t_{n,\alpha}$, welche mittels Computer berechnet werden können.

Zusammenfassend erfolgt der t -Test gemäss den folgenden sechs Schritten:

1. *Modell:* X_i ist eine kontinuierliche Messgrösse;

$$X_1, \dots, X_n \text{ i.i.d. } \mathcal{N}(\mu, \sigma_X^2), \quad \sigma_X \text{ wird durch } \hat{\sigma}_X \text{ geschätzt}$$

2. *Nullhypothese:*

$$H_0 : \mu = \mu_0$$

Alternative:

$$H_A : \mu \neq \mu_0 \quad (\text{oder } < \text{ oder } >)$$

3. *Teststatistik:*

$$T = \frac{\sqrt{n}(\bar{X}_n - \mu_0)}{\hat{\sigma}_X} = \frac{\text{beobachtet - erwartet}}{\text{geschätzter Standardfehler}}$$

Verteilung der Teststatistik unter H_0 :

$$T \sim t_{n-1}$$

4. *Signifikanzniveau:*

$$\alpha$$

5. *Verwerfungsbereich für die Teststatistik:*

$$K = (-\infty, t_{n-1; \frac{\alpha}{2}}] \cup [t_{n-1; 1-\frac{\alpha}{2}}, \infty) \quad \text{bei } H_A: \mu \neq \mu_0,$$

$$K = (-\infty, t_{n-1; \alpha}] \quad \text{bei } H_A: \mu < \mu_0,$$

$$K = [t_{n-1; 1-\alpha}, \infty) \quad \text{bei } H_A: \mu > \mu_0.$$

6. *Testentscheid:*

Überprüfe, ob der beobachtete Wert der Teststatistik im Verwerfungsbereich liegt.

Da das Quantil der t -Verteilung grösser ist als das Quantil der Normalverteilung, erhält man einen etwas kleineren Verwerfungsbereich als beim z -Test. Für grosse n ist der Unterschied allerdings minim (da $t_{n-1} \approx \mathcal{N}(0, 1)$ falls n gross).

Beispiel 4.3.17

Wir berechnen nun nochmals das Beispiel 4.3.14. Diesmal schätzen wir allerdings die Standardabweichung σ_X aus den Daten. Wir führen also einen t -Test auf dem 5 % Signifikanzniveau durch:

1. *Modell:* X_i ist eine kontinuierliche Messgröße;

$$X_1, \dots, X_n \text{ i.i.d. } \mathcal{N}(\mu, \sigma_X^2), \quad \sigma_X \text{ wird geschätzt, } \hat{\sigma}_X = 0.024$$

2. *Nullhypothese:*

$$H_0 : \mu = \mu_0 = 80.00$$

Alternative:

$$H_A : \mu \neq \mu_0$$

3. *Teststatistik:*

$$T = \frac{\sqrt{n}(\bar{X}_n - \mu_0)}{\hat{\sigma}_X}$$

Verteilung der Teststatistik unter H_0 :

$$T \sim t_{n-1}$$

4. *Signifikanzniveau:*

$$\alpha = 0.05$$

5. *Verwerfungsbereich für die Teststatistik:*

$$K = (-\infty, t_{n-1; \frac{\alpha}{2}}] \cup [t_{n-1; 1-\frac{\alpha}{2}}, \infty) \quad \text{bei } H_A: \mu \neq \mu_0,$$

Den Wert

$$t_{n-1; 1-\frac{\alpha}{2}} = t_{12; 0.975} = 2.179$$

ermitteln wir mit Hilfe von **Python**, wobei $\alpha = 0.05$ und $n = 13$, wie folgt ([zu R](#))

```
from scipy.stats import norm, t
t.ppf(q=0.975, df=12)
## 2.1788128296634177
```

Der Verwerfungsbereich der Teststatistik ist also:

$$K = (-\infty, -2.179] \cup [2.179, \infty)$$

6. Testentscheid:

Aus den $n = 13$ Daten haben wir

$$\bar{x} = 80.02 \quad \text{und} \quad \hat{\sigma}_X = 0.024$$

errechnet. Damit ergibt sich für die Teststatistik der Wert

$$t = \frac{\sqrt{n}(\bar{x}_n - \mu_0)}{\hat{\sigma}_X} = \frac{\sqrt{13}(80.02 - 80.00)}{0.024} = 3.00$$

Der beobachtete Wert der Teststatistik liegt im Verwerfungsbereich. Daher wird die Nullhypothese auf dem 5 % Niveau verworfen.

Bemerkungen:

- i. Wir können die Grenzen des Verwerfungsbereichs mit **Python** (aber *nicht* mit **R**) auch direkt berechnen: ([zu R](#))

```
t.interval(alpha=0.95, df=12, loc=80, scale=0.024/np.sqrt(13))

## (79.98549694515017, 80.01450305484983)
```

$$K = (-\infty, 79.99] \cup [80.01, \infty)$$

Der Wert von 80.02 liegt im Verwerfungsbereich und somit wird die Nullhypothese verworfen.

- ii. Wir können den Testentscheid des obigen Beispiels mit **Python** auch direkt bestimmen, ohne Umweg über den Verwerfungsbereich: ([zu R](#))

```
import scipy.stats as st
x = Series([79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04, 79.97, 80.05, 80.01, 80.03, 80.02, 80.04])

st.ttest_1samp(a=x, popmean=80)

## Ttest_1sampResult(statistic=3.1246428367325474, pvalue=0.0087787788183911)
```

Der P -Wert mit einer zweiseitigen Alternative ist 0.008 779 und ist somit auf dem 5 % Signifikanzniveau signifikant.

- iii. Der P -Wert bei 2-seitiger Alternative $H_A: \mu \neq \mu_0$ kann wie folgt berechnet werden (der beobachtete Wert der Teststatistik ist $t = \frac{\sqrt{n}|\bar{x}_n - \mu_0|}{\hat{\sigma}_X}$):

$$\begin{aligned} P\text{-Wert} &= P(|T| > |t|) \\ &= P(T < -|t|) + P(T > |t|) \\ &= 2 \cdot P(T > |t|) \end{aligned}$$

- iv. Der beobachtete Wert der Teststatistik ist 3.12 und folgt unter der Nullhypothese einer t -Verteilung mit **df = 12** Freiheitsgraden.

- v. Der beobachtete Mittelwert der Daten ist 80.02. Ein 95 %-Vertrauensintervall für den wahren Mittelwert der Messungen ist [80.006, 80.035] (siehe folgender Abschnitt).

□

4.4. Vertrauensintervall für μ

Bei der Punktschätzung für den Mittelwert μ einer Messreihe erhalten wir einen einzigen Zahlwert. Wir wissen allerdings nicht, wie nahe dieser geschätzte Mittelwert beim wahren, aber meist unbekannten Mittelwert der Verteilung der Messreihe liegt.

Mit dem Vertrauensintervall geben wir ein Intervall an, in welches der wahre Mittelwert mit einer bestimmten vorgegebenen Wahrscheinlichkeit fällt. Wir werden im Folgenden zwei Methoden kennenlernen, um Vertrauensintervalle zu bestimmen: einerseits das Bootstrapping-Verfahren, welches keine Annahme über die Messdaten macht, andererseits eine formale Methode, bei der wir von einer Normalverteilung der Messdaten ausgehen.

4.4.1. Bootstrapping

Das Bootstrapping-Verfahren ist eine wichtige statistische Methode, die oft bei der Schätzung von Parametern von Wahrscheinlichkeitsverteilungen zur Anwendung kommt. Wir werden diese Methode zur Bildung eines Vertrauensintervales kennenlernen.

Wir wollen das Vertrauensintervall zuerst konstruieren und dann die entsprechende Interpretation liefern.

Idee des Bootstrapping

Die Implementierung des Bootstrapping ist relativ einfach ist, wobei diese Methode den Einsatz von Computern erfordert. Die Grundidee ist, dass aus einer Messreihe durch Resampling (Stichproben aus dieser Messreihe) Informationen über die Messreihe selbst gewonnen werden.

Beispiel 4.4.1

Wir haben die Messreihe **methode_B**, bei der die Schmelzwärme aufgezeichnet wurde:

$$80.02, \quad 79.94, \quad 79.98, \quad 79.97, \quad 79.97, \quad 80.03, \quad 79.95, \quad 79.97$$

Diese Messreihe folgt einer unbekannten Verteilung und hat einen unbekannten Erwartungswert μ . Wir nennen die unbekannte Verteilung F , wobei wir den Mittelwert \bar{x} der Messreihe als Punktschätzung von μ betrachten können. Aber wie gut ist diese Schätzung?

Indem wir die unbekannte Verteilung F durch das (wiederholte) Erzeugen von Datensätzen simulieren und jeweils die Mittelwerte dieser simulierten Datensätze bestimmen, können wir die Verteilung der Mittelwerte und damit ein Vertrauensintervall für den wahren Mittelwert konstruieren. Dies ist die Idee von Bootstrapping.

□

Empirische Verteilung von Daten

Die empirische Verteilung der Daten entspricht der Verteilung, die wir in den Daten tatsächlich beobachten. Wir werden dies am folgenden Beispiel illustrieren.

Beispiel 4.4.2

Wir würfeln einen 8-seitigen fairen Würfel 10 mal und erhalten folgende Werte in aufsteigender Reihenfolge:

$$1, \quad 1, \quad 2, \quad 3, \quad 3, \quad 3, \quad 3, \quad 4, \quad 7, \quad 7$$

Nun schreiben wir diese Zahlen auf je einen Papierstreifen, legen diese in einen Hut und beginnen diese Streifen zufällig zu ziehen, wobei wir jeden Streifen nach dem Ziehen wieder in den Hut zurücklegen.

Dann ist die Wahrscheinlichkeit, eine 3 zu ziehen, $\frac{4}{10}$ und die Wahrscheinlichkeit, eine 4 zu ziehen, $\frac{1}{10}$. Die Wahrscheinlichkeitsverteilung für die Zufallsvariable X , den Wert einer Zahl zu ziehen, lautet dann wie folgt:

x		1		2		3		4		7
$P(X = x)$		$\frac{2}{10}$		$\frac{1}{10}$		$\frac{4}{10}$		$\frac{1}{10}$		$\frac{2}{10}$

□

Haben die Daten eine (unbekannte) Verteilung F , so bezeichnen wir die zugehörige empirische Verteilung mit F^* . Ist die Datenmenge gross, so sollte nach dem Gesetz der grossen Zahlen F^* eine gute Annäherung von F sein.

Beispiel 4.4.3

Für das Würfelbeispiel oben sind in der Tabelle 4.3 die wahre und die empirische Verteilung aufgeführt.

x	1	2	3	4	5	6	7	8
wahres $P(X = x)$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$
empirisches $P(X = x)$	$\frac{2}{10}$	$\frac{1}{10}$	$\frac{4}{10}$	0	0	$\frac{1}{10}$	$\frac{2}{10}$	0

Table 4.3.: Die wahre Verteilung F und die empirische Funktion F^* beim 8-seitigen Würfel.

□

Resampling

Der wesentliche Schritt beim Bootstrapping-Verfahren ist das Resampling. Dabei wird aus den Daten selbst wieder eine Stichprobe gezogen (mit Zurücklegen).

Beispiel 4.4.4

Beim Würfelbeispiel hatten wir die 10 Datenpunkte

$$1, \quad 1, \quad 2, \quad 3, \quad 3, \quad 3, \quad 3, \quad 4, \quad 7, \quad 7$$

Wir betrachten dies als Stichprobe (sample), die aus einer zugrundeliegenden Verteilung stammt. Beim Resampling werden nun die Papierstreifen in einen Hut gelegt, zufällig gezogen und wieder zurückgelegt. Machen wir dies fünfmal, so erhalten wir eine Stichprobe der Stichprobe (Resampling) und erhalten zum Beispiel die Werte

$$3, \quad 2, \quad 3, \quad 3, \quad 1$$

Da wir die mit den Zahlen beschrifteten Streifen wieder zurücklegen, können Zahlen mehrfach erscheinen.

□

Bezeichnen wir die ursprünglichen Datenpunkte mit

$$x_1, x_2, \dots, x_n$$

so bezeichnen wir die durch Resampling erzeugten Datenpunkte der Länge m mit einem Stern

$$x_1^*, x_2^*, \dots, x_m^*$$

Entsprechend ist \bar{x} der Mittelwert der ursprünglichen Stichprobe oder Messreihe und \bar{x}^* der Mittelwert der Daten, die wir durch Resampling erhalten haben.

Die empirische Bootstrap-Stichprobe

Nehmen wir an, wir haben n Datenpunkte

$$x_1, x_2, \dots, x_n$$

die einer (unbekannten) Verteilung F folgen. Eine empirische Bootstrap-Stichprobe ist eine durch Resampling erzeugte Stichprobe derselben Länge n . Da der Standardfehler von der Länge der Messreihe abhängt, wird eine durch Resampling gewählte Stichprobe mit derselben Länge gewählt. Somit können wir σ durch σ^* approximieren (schätzen). Dies werden wir benutzen, um das Vertrauensintervall zu konstruieren.

Beispiel 4.4.5

Aus Beispiel 4.4.1 haben wir die Messreihe beruhend auf **methode_B**

$$80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97$$

die klein genug ist, damit wir jeden Schritt explizit durchführen können.

Eine Bootstrap-Stichprobe (enlg. *bootstrap sample*) ist eine Stichprobe mit derselben Länge wie die ursprüngliche Messreihe, wobei die Datenpunkte aus der ursprünglichen Messreihe durch Ziehen mit Zurücklegen erzeugt wurden.

Wir stellen uns nun vor, dass wir Kugeln mit den obigen Zahlen beschriften. Die Zahl 79.97 kommt dann dreimal vor, die Zahl 80.02 aber nur einmal. Dann legen wir die Kugeln in eine Schüssel und ziehen blind eine Kugel aus der Schüssel. Wir notieren die Zahl und legen die Kugel wieder zurück in die Schüssel. Dies wiederholen wir achtmal, da die ursprüngliche Messreihe die Länge 8 hat.

In der Praxis machen wir dieses Resampling natürlich mit einer Computer-Software, zum Beispiel mit **Python**. ([zu R](#))

Kapitel 4. Statistik für Messdaten

```
import numpy as np
np.random.seed(1)
methode_B = np.array([80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95,
79.97])

# Arithmetisches Mittel der Messreihe methode_B
print('Arithmetisches Mittel von Messreihe Methode B:', methode_B.mean())

# Länge n der Messreihe methode_B
n = methode_B.size

# Anzahl Bootstrap samples
nboot = 1

# Bootstrap Sample wird aus Messreihe durch zufälliges
# Ziehen mit Zurücklegen generiert
bootstrap_sample = np.random.choice(methode_B, n*nboot, replace=True)

bootstrap_sample
print('Bootstrap Sample : ', bootstrap_sample)
# Arithmetisches Mittel des Bootstrap Sample
print('Arithmetisches Mittel von Bootstrap Sample:', bootstrap_sample.mean())


import numpy as np
methode_B = np.array([80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95,
79.97])
n = methode_B.size
nboot = 1
bootstrap_sample = np.random.choice(methode_B, n*nboot, replace=True)
bootstrap_sample
print('Bootstrap Sample : ', bootstrap_sample)

## Bootstrap Sample : [80.03 79.97 79.97 80.02 79.97 79.94 79.97 80.03]
```

Hier kommt z.B. der Messwert 80.02 nicht vor, dafür die 80.03 zweimal. Der Mittelwert dieser Bootstrap-Stichprobe ist 79.99. ([zu R](#))

```
## 79.9875
```

Die Idee des Bootstrapping-Verfahrens ist nun, dass dieses Resampling sehr oft durchgeführt wird, um die Streuung von

$$\mu \approx \bar{x}$$

Kapitel 4. Statistik für Messdaten

abzuschätzen, wobei 79.99 der Durchschnitt der ursprünglichen Messreihe war.



Beispiel 4.4.6

Wir illustrieren das Bootstrapping-Verfahren mit Hilfe der Messreihe **methode_B**

80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97

die klein genug ist, so dass wir jeden Schritt explizit ausführen können. Wir wollen im Folgenden ein 95 %-Bootstrap-Vertrauensintervall für das wahre μ angeben.

Die Schätzung von μ für unsere ursprüngliche Testreihe ist

$$\mu \approx \bar{x} = 79.99$$

Mit **Python** erzeugen wir nun 20 Bootstrap-Stichproben, wobei alle die Länge 8 haben. Jede der 20 Spalten im folgenden Array entspricht einer Bootstrap-Stichprobe.

79.97	80.02	80.02	79.94	79.97	79.97	80.03	79.97	79.95	79.94	79.98	79.97	79.95	80.03	79.98	79.97	79.97	79.98	79.97	79.98	79.97	
80.03	79.95	79.98	79.97	79.97	79.97	79.94	79.94	79.97	79.95	80.02	79.95	80.03	79.94	79.94	79.97	79.97	79.97	80.03	79.95	79.94	79.94
79.94	80.02	79.97	79.94	80.02	80.02	80.03	79.97	79.97	79.98	79.94	80.02	79.95	79.97	79.97	79.95	80.03	79.94	79.94	79.95	79.97	79.97
79.97	80.02	79.94	79.94	79.97	79.97	79.98	79.97	80.02	80.03	79.97	79.94	79.98	79.97	80.02	79.97	79.94	79.97	79.98	79.97	79.97	79.97
79.98	79.94	80.02	79.95	79.94	79.97	79.97	79.95	79.97	80.03	79.97	79.97	80.03	79.94	79.97	79.95	80.02	80.02	79.98	79.98	79.98	79.98
79.97	79.97	79.97	79.97	79.94	79.97	79.97	79.97	79.98	79.95	79.95	80.02	80.02	79.97	79.97	79.94	80.03	79.97	80.02	79.97	79.97	79.97
80.02	80.03	80.03	79.95	79.97	79.97	80.03	79.95	79.98	79.97	80.03	79.97	79.97	80.03	79.97	79.97	79.97	79.97	79.97	79.97	79.95	79.95
79.97	79.97	79.97	80.03	79.97	79.94	80.02	79.97	79.98	80.02	80.03	79.98	79.97	79.97	79.94	79.97	79.97	80.02	79.98	79.98	79.97	79.97

Wir berechnen nun die Mittelwerte in allen Spalten und ordnen diese der Reihe nach:
(zu R)

```
import numpy as np
np.random.seed(1)
methode_B = np.array([80.02, 79.94, 79.98, 79.97, 79.97, 80.03,
                     79.95, 79.97])
n = methode_B.size
nboot = 20
bootstrap_sample = np.random.choice(methode_B, n*nboot, replace=True)
bootstrap_sample_array = np.reshape(bootstrap_sample, (n, nboot))
bootstrap_sample_array

xbarstar = bootstrap_sample_array.mean(axis=0)

np.sort(xbarstar)
print(np.sort(xbarstar))

## [79.96375 79.965    79.96625 79.96875 79.97125 79.97375 79.97625 79.9775
## 79.97875 79.98     79.98125 79.98125 79.9825 79.9825 79.98375 79.985
## 79.985    79.9875 79.99     79.99375]
```

Beim 95 %-Bootstrap-Vertrauensintervall wählen wir die „mittleren“ 95 % dieser Daten. Diese werden durch die 2.5 %- und 97.5 %-Quantile begrenzt. ([zu R](#))

```
xbarstar = bootstrap_sample_array.mean(axis=0)

d = np.percentile(xbarstar, q=[2.5, 97.5])
print('Vertrauensintervall: ', d)

## Vertrauensintervall: [79.96434375 79.99196875]
```

Das 95 %-Bootstrap-Vertrauensintervall lautet dann

$$[79.96, 79.99]$$

□

Bootstrapping-Verfahren

Seien x_1, \dots, x_n Realisierungen von unabhängigen und identisch verteilten Zufallsvariablen X_1, \dots, X_n . Wir nehmen im Folgenden an, dass

$$\hat{\mu} = \hat{\mu}(x_1, \dots, x_n) = \bar{x}_n$$

der Schätzer des Erwartungswertes μ ist.

1. Wir wählen eine (grosse) Anzahl $B \in \mathbb{N}$.
2. Für $b = 1, \dots, B$
 - ziehen wir n Stichproben $\{x_1^*, \dots, x_n^*\}$ aus $\{x_1, \dots, x_n\}$ mit Zurücklegen.
 - Wir berechnen den Schätzer $\hat{\mu}_b^* = \hat{\mu}(x_1^*, \dots, x_n^*) = \bar{x}_n^*$
3. Die empirische Verteilungsfunktion \hat{F}^* von $(\hat{\mu}_1^*, \dots, \hat{\mu}_B^*)$ nähert die Verteilung der wahren Verteilung von $\hat{\mu}$ an. Somit kann das $1 - \alpha$ -Bootstrap-Vertrauensintervall durch

$$[\mu_{\alpha/2}^*, \mu_{1-\alpha/2}^*]$$

angegeben werden, wobei $\mu_{\alpha/2}^*$ und $\mu_{1-\alpha/2}^*$ das empirische $\alpha/2$ -Quantil, resp. das empirische $1 - \alpha/2$ -Quantil der Verteilungsfunktion \hat{F}^* von $(\hat{\mu}_1^*, \dots, \hat{\mu}_B^*)$ bezeichnet.

Bemerkungen:

- i. Es ist wichtig festzuhalten, dass eine Bootstrap-Stichprobe x_1^*, \dots, x_n^* Duplikate aus der ursprünglichen Messreihe enthalten kann und dass andere Datenpunkte wiederum gar nicht austauschen können.
- ii. Der Vorteil des Bootstrapping-Verfahrens besteht darin, dass keine Verteilung angenommen werden muss. Das Verfahren ist für beliebige Verteilungen anwendbar.
- iii. Oft wird auch ein um den Mittelwert \bar{x}_n (fast) symmetrisches Vertrauensintervall angegeben. Dazu werden die Abweichungen der empirischen Mittelwerte der Bootstrap-Stichproben vom Mittelwert \bar{x} der ursprünglichen Messreihe ermittelt:

$$\delta^* = \bar{x}^* - \bar{x}$$

Das (symmetrische) $1 - \alpha\%$ -Vertrauensintervall ist dann

$$[\bar{x} + \delta_{\alpha/2}^*, \bar{x} + \delta_{1-\alpha/2}^*]$$

wobei $\delta_{\alpha/2}^*$ und $\delta_{1-\alpha/2}^*$ das empirische $\alpha/2$ -Quantil, resp. das empirische $1 - \alpha/2$ -Quantil der Abweichungen δ^* bezeichnet.

Beispiel 4.4.7

Kehren wir zur Messreihe der mit **methode_B** ermittelten Datenpunkte zurück:

80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97

Nun berechnen wir $\delta^* = \bar{x}^* - \bar{x}$ für jede Bootstrap-Stichprobe und ordnen diese der Grösse nach

```
tempdata = np.random.choice(methode_B, n*nboot, replace=True)
bootstrap_sample_array = np.reshape(tempdata, (n, nboot))

xbarstar = bootstrap_sample_array.mean(axis=0)
delta = np.sort(xbarstar - xbar)

print('Abweichungen: ', delta)

## Abweichungen: [-1.5000000e-02 -1.3750000e-02 -1.2500000e-02 -1.0000000e-02
## -7.5000000e-03 -5.0000000e-03 -2.5000000e-03 -1.2500000e-03
## 1.42108547e-14 1.2500000e-03 2.5000000e-03 2.5000000e-03
## 3.7500000e-03 3.7500000e-03 5.0000000e-03 6.2500000e-03
## 6.2500000e-03 8.7500000e-03 1.1250000e-02 1.5000000e-02]
```

Kapitel 4. Statistik für Messdaten

Wir finden folgende Abweichungen:

$-1.50 \cdot 10^{-2} - 1.38 \cdot 10^{-2} - 1.25 \cdot 10^{-2} - 1.00 \cdot 10^{-2} - 7.50 \cdot 10^{-3} - 5.00 \cdot 10^{-3} - 2.50 \cdot 10^{-3} - 1.25 \cdot 10^{-3} 1.42 \cdot 10^{-14} 1.25 \cdot 10^{-3} 2.$

Bezeichnen wir mit $\delta_{0.025}^*$ die 2.5 %-Quantile und mit $\delta_{0.975}^*$ die 97.5 %-Quantile, so ergibt sich mit **Python** -2.23 resp. 2.54 .

```
tempdata = np.random.choice(methode_B, n*nboot, replace=True)
bootstrap_sample_array = np.reshape(tempdata, (n, nboot))

xbarstar = bootstrap_sample_array.mean(axis=0)
delta = np.sort(xbarstar - xbar)
d = np.percentile(delta, q=[2.5, 97.5])

print('0.025 und 0.975 Quantilen der Abweichungen: ', d)

## 0.025 und 0.975 Quantilen der Abweichungen: [-0.01440625  0.01321875]
```

Somit erhalten wir für unser *symmetrisches* 95 %-Bootstrap-Vertrauensintervall

$$[\bar{x} + \delta_{0.05}^*, \bar{x} + \delta_{0.975}^*] = [79.97 - 0.0144, 79.97 + 0.0132] = [79.96, 79.98]$$

□

Beispiel 4.4.8

Im Beispiel 4.4.6 haben wir der Übersichtlichkeit halber nur 20 Bootstrap-Stichproben erzeugt. Aber mit **Python** können wir auch 10 000 Bootstrap-Stichproben erzeugen, womit wir wesentlich genauere Abschätzungen für das 95 %-Bootstrap-Vertrauensintervall erhalten. ([zu R](#))

```
import numpy as np
np.random.seed(1)
methode_B = np.array([80.02, 79.94, 79.98, 79.97, 79.97, 80.03,
                     79.95, 79.97])
n = methode_B.size
nboot = 10000
bootstrap_sample = np.random.choice(methode_B, n*nboot, replace=True)
bootstrap_sample_array = np.reshape(bootstrap_sample, (n, nboot))
bootstrap_sample_array

xbarstar = bootstrap_sample_array.mean(axis=0)

d = np.percentile(xbarstar, q=[2.5, 97.5])
print('Vertrauensintervall: ', d)
```

Kapitel 4. Statistik für Messdaten

```
## Vertrauensintervall: [79.96 79.99875]
```

Somit ergibt dies das Vertrauensintervall:

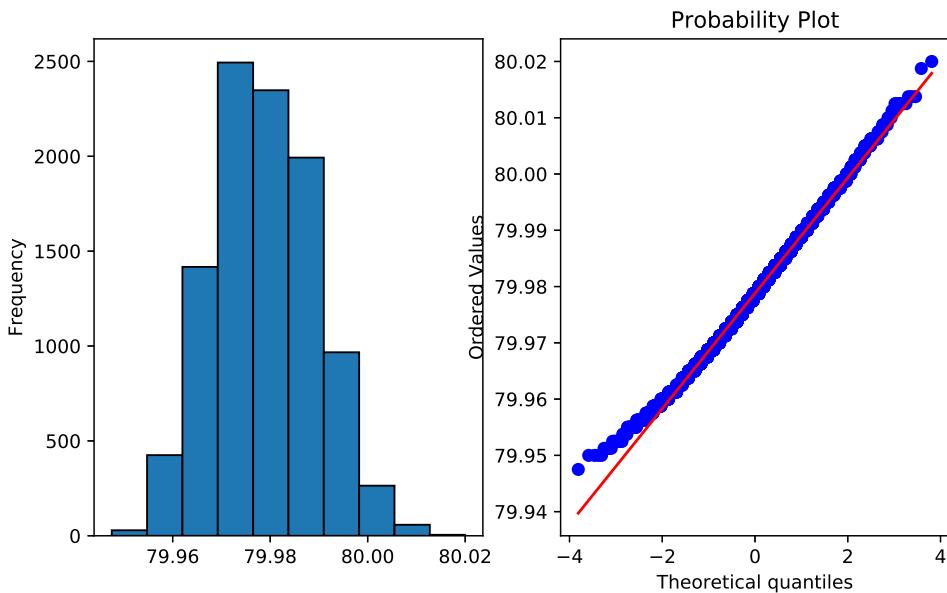
$$[79.96, 80.00]$$

Nun können wir uns auch noch die Frage stellen, wie die Mittelwerte der Bootstrap-Stichproben μ^* verteilt sind.

```
import numpy as np
from pandas import Series
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import norm, probplot

x = np.array([80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97])
n = x.size
np.random.seed(1)
xbar = x.mean()

nboot = 10000
tmpdata = np.random.choice(x, n*nboot, replace=True)
bootstrapsample = np.reshape(tmpdata, (n, nboot))
xbarstar = Series(bootstrapsample.mean(axis=0))
plt.subplot(121)
xbarstar.plot(kind="hist", edgecolor="black")
plt.subplot(122)
probplot(xbarstar, plot=plt)
```



Daraus schliessen wir, dass die Bootstrap-Schätzer annähernd normalverteilt sind.

□

Als nächstes wollen wir den Standardfehler von $\hat{\mu}$ mit Hilfe der Bootstrap-Methode schätzen:

$$\text{se}_B(\hat{\mu}) = \sqrt{\frac{1}{B-1} \sum_{b=1}^B (\hat{\mu}_b^* - \bar{\mu}^*)^2}, \quad \text{mit}$$

$$\bar{\mu}^* = \frac{1}{B} \sum_{b=1}^B \hat{\mu}_b^*.$$

Wie wir gesehen haben, sind die Bootstrap-Schätzer normalverteilt. Deshalb kann ein 95 %-Bootstrap-Vertrauensintervall auch wie folgt angegeben werden:

$$\bar{x} \pm 2 \cdot \text{se}_B$$

Beispiel 4.4.9

```

import numpy as np
from pandas import Series
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import norm, probplot

n = x.size
np.random.seed(1)
xbar = x.mean()

nboot = 10000
tmpdata = np.random.choice(x, n*nboot, replace=True)
bootstrapsample = np.reshape(tmpdata, (n, nboot))
xbarstar = Series(bootstrapsample.mean(axis=0))
d = xbar - 2*xbarstar.std() , xbar + 2*xbarstar.std()

print('Vertrauensintervall: ', d)

## Vertrauensintervall: (79.95819874674378, 79.9993012532562)

```

Somit ergibt dies das Vertrauensintervall:

$$[79.96, 80.00]$$

□

Wie lässt sich nun dieses Vertrauenintervall interpretieren? Dazu machen wir ein weiteres Beispiel.

Beispiel 4.4.10

Wir simulieren nun Daten, deren wahres μ wir kennen. Dazu wählen wir 100 Zufallszahlen, die der Verteilung $\mathcal{N}(40, 5^2)$ folgen. Das wahre μ ist also 40. Wir können uns nun fragen, ob dieses μ nun im entsprechenden 95 %-Vertrauensintervall liegt oder nicht. ([zu R](#))

```
## Vertrauensintervall: [38.90855602 40.02272279]
```

```

import numpy as np
x = np.random.normal(loc=40, scale=5, size=100)

n = x.size

```

Kapitel 4. Statistik für Messdaten

```
np.random.seed(4)
xbar = x.mean()

# Anzahl Bootstrap Stichproben
nboot = 20

# Erzeuge Bootstrap Stichproben
bootstrap_samples = np.random.choice(x, n*nboot, replace=True)

bootstrap_sample_array = np.reshape(bootstrap_samples, (n, nboot))

# Arithmetisches Mittel für jede Bootstrap Stichprobe
xbarstar = bootstrap_sample_array.mean(axis=0)

# 2.5% und 97.5% Quantilen der Mittelwerte
# der Bootstrap Samples
ci = np.percentile(xbarstar, q=[2.5, 97.5])

# Vertrauensintervalle
print("Vertrauensintervall: ", ci)
```

Dies ergibt das Vertrauensintervall [39.6, 41.39]. Das wahre μ liegt somit in diesem Intervall. Ist dies aber immer der Fall? Wir generieren nun 100 Testreihen, wobei jede Testreihe 100 normalverteilte Zufallszahlen mit Mittelwert 40 enthält. Wir bestimmen für jede Testreihe das Vertrauensintervall und schauen, ob das wahre μ darin liegt. (zu R)

```
import numpy as np
# Wir erzeugen 100'000 normalverteilte Zufallszahlen
# mit Mittelwert 40 und Standardabweichung 5
x = np.random.normal(loc=40, scale=5, size=100000)

# Wir ordnen diese Zahlen in einem Array an, der aus 1'000 Zeilen
# und 100 Spalten besteht
measurement_array = np.reshape(x, (1000, 100))
print(measurement_array.shape)
print(measurement_array[1].size)

# Anzahl Bootstrap Samples
nboot = 1000

# Länge von jeder Bootstrap Stichprobe
n = 100
```

Kapitel 4. Statistik für Messdaten

```
# k zählt Anzahl Vertrauensintervalle, die das
# wahre mu=40 nicht enthalten
k=0
# Wir iterieren über alle 100 Testreihen und bestimmen für jede
# Testreihe ein Vertrauensintervall (mittels bootstrap)
for i in range(0,100):
    x = measurement_array[i]
    # Arithmetisches Mittel pro Zeile im Array wird berechnet
    xbar = x.mean()
    # für die Zeile x wird nun ein Vertrauensintervall
    # mittels Bootstrap konstruiert
    bootstrap_samples = np.random.choice(x, n*nboot, replace=True)
    bootstrap_sample_array = np.reshape(bootstrap_samples, (n, nboot))
    xbarstar = bootstrap_sample_array.mean(axis=0)
    d = np.percentile(xbarstar, q=[2.5, 97.5])
    # Falls 40 im Vertrauensintervall für Zeile i NICHT enthalten ist
    # wird k um 1 erhöht
    if d[0]<= 40 <= d[1]:
        k=k+1

print(k)
```

```
## (100, 100)
## 100
## 96
```

In 96 Fällen liegt das wahre μ im Vertrauensintervall der entsprechenden Messreihe. Das heisst, in 96 % der Fälle ist das wahre μ im 95 % Vertrauensintervall.

Wir können dies auch noch graphisch darstellen. In Abbildung 4.17 sehen wir 100 Simulationen von 95 % Bootstrap-Vertrauensintervallen. Zudem ist das wahre Mittel $\mu = 40$ eingezeichnet.

Wir sehen, dass vier Vertrauensintervalle (schwarz eingezeichnet) die horizontale Linie 40 nicht schneiden. Diese Vertrauensintervalle enthalten somit das wahre Mittel *nicht*. Somit ist das wahre Mittel in 96 % aller 95 %-Vertrauenintervalle enthalten. (zu R)

```
import numpy as np
import matplotlib.pyplot as plt

# Wir generieren 10'000 normalverteilte Zufallszahlen
# mit Mittelwert 40 und Standardabweichung 5
x = np.random.normal(loc=40, scale=100, size=10000)
```

```
# Wir ordnen die Zufallszahlen in einem array mit 100 Spalten
# und 100 Zeilen an
measurement_array = np.reshape(x, (100,100))
print(measurement_array.shape)

# Anzahl Bootstrap Samples
nboot = 10000
n = 100

# Wir iterieren über die 100 Testreihen
for i in range(0,100):
    # wir lesen die i-te Zeile aus dem measurement_array heraus
    y = measurement_array[i]
    # Mittelwert von i-ten Testreihe
    xbar = y.mean()
    # Bestimmung des Vertrauensintervalls der i-ten Testreihe
    tmpdata = np.random.choice(y, n*nboot, replace=True)
    bootstrapsample = np.reshape(tmpdata, (n, nboot))
    xbarstar = bootstrapsample.mean(axis=0)
    d = np.percentile(xbarstar, q=[2.5, 97.5])
    plt.plot([i,i],[xbar-d[1],xbar+d[1]])
    if (d[0]<= 40 <= d[1]) == False:
        plt.plot([i,i],[d[0], d[1]],c="black", linewidth=3)

plt.plot([-5,105],[40,40])

plt.show()
```



Im Beispiel liegt der wahre Wert also in etwa 95 % der Fälle in einem der 95 %-Bootstrap-Vertrauensintervalle. Dies können wir verallgemeinern. Lassen wir die Anzahl Testreihen sehr gross werden, so liegt nach dem Gesetz der grossen Zahlen das wahre Mittel in 95 % aller 95 % Bootstrap-Vertrauensintervalle.

Vertrauensintervalle

Im $(1 - \alpha) \cdot 100\%$ -Vertrauensintervall liegt der wahre Wert μ mit einer Wahrscheinlichkeit von $\alpha \cdot 100\%$.

Ist das Vertrauensintervall sehr breit, so haben wir wenig Gewissheit, wo der wahre Wert von μ liegt. Auf der anderen Seite gibt uns ein schmales Vertrauensintervall eine gute Schätzung für den wahren Wert von μ .

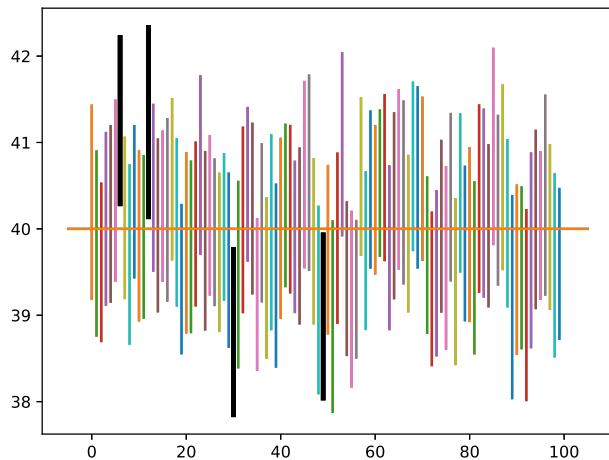


Abbildung 4.17.: 100 Bootstraps-Vertrauensintervalle

4.4.2. Vertrauensintervalle für Normalverteilungen

Wir rufen uns nochmal den Verwerfungsbereich einer normalverteilten Zufallsvariable X mit bekanntem σ_X in Erinnerung. Wir gehen von normalverteilten Zufallsvariablen aus. Wir beschränken uns vorläufig auf das Signifikanzniveau 5 % und eine zweiseitige Alternativhypothese, d.h. in diesem Falle auf einen zweiseitigen Verwerfungsbereich.

Beispiel 4.4.11

Wir bestimmen im Folgenden den zweiseitigen Verwerfungsbereich einer standardnormalverteilten Zufallsvariable Z .

In Abbildung 4.18 sehen wir, dass der Verwerfungsbereich durch die Quantile $z_{0.025}$ und $z_{0.975}$ begrenzt wird. Diese Quantile haben die Werte -1.96 und 1.96 . ([zu R](#))

```
from scipy.stats import norm
norm.ppf(q=[0.025, 0.975])
## [-1.95996398  1.95996398]
```

Die Wahrscheinlichkeit beträgt also 95 %, dass ein Messwert von Z zwischen -1.96 und 1.96 fällt. Dieses Intervall beschreiben wir wie folgt

$$-1.96 \leq Z \leq 1.96$$

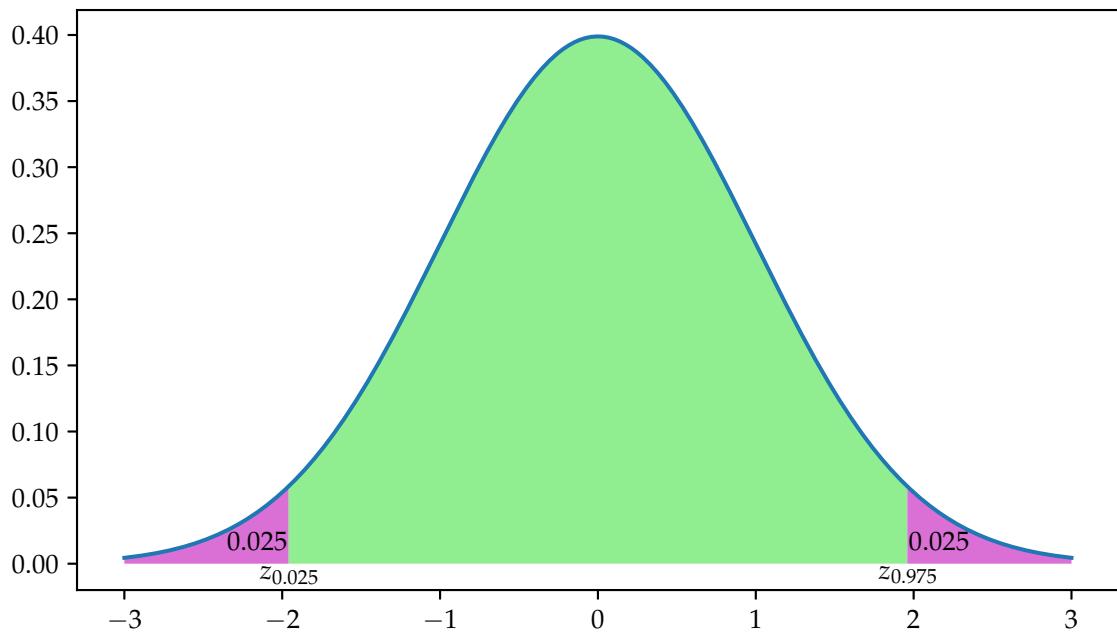


Abbildung 4.18.: Verwerfungsbereich bei Standardnormalverteilung

□

Wir haben im Abschnitt 3.3.1 gesehen, dass jede normalverteilte Zufallsvariable standardisiert werden kann.

Beispiel 4.4.12

Ist beispielsweise

$$X \sim \mathcal{N}(6, 2^2)$$

so lautet die standardisierte Zufallsvariable

$$Z = \frac{X - 6}{2}$$

Setzen wir dies in Gleichung

$$-1.96 \leq Z \leq 1.96$$

vom vorangehenden Beispiel 4.4.11 ein, so erhalten wir:

$$-1.96 \leq \frac{X - 6}{2} \leq 1.96$$

Kapitel 4. Statistik für Messdaten

Wir lösen nun diese Ungleichung zwischen den Ungleichheitszeichen nach X auf. Dazu multiplizieren wir die Ungleichung mit 2 und addieren 6. Wir erhalten dann

$$6 - 1.96 \cdot 2 \leq X \leq 6 + 1.96 \cdot 2$$

oder

$$2.08 \leq X \leq 9.92$$

Die beiden Zahlen geben gerade die Grenzen des Verwerfungsbereiches an (siehe Abbildung 4.19). (zu R)

```
from scipy.stats import norm

norm.ppf(q=[0.025, 0.975], loc=6, scale=2)

## [2.08007203 9.91992797]
```

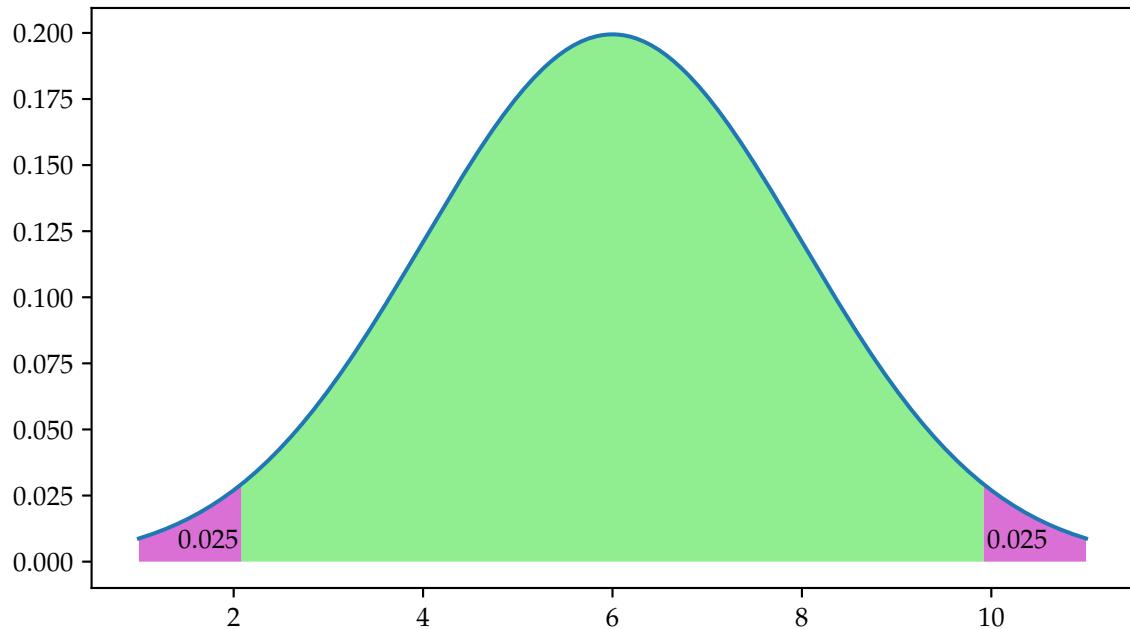


Abbildung 4.19.: Verwerfungsbereich der Normalverteilung mit $\mu = 6$ und $\sigma = 2$

Wir können nun den Bereich

$$2.08 \leq X \leq 9.92$$

auch als das Intervall auffassen, in welchem 95 % aller Messwerte liegen.

□

Beispiel 4.4.12 wollen wir nun verallgemeinern. Ist

$$X \sim \mathcal{N}(\mu, \sigma^2)$$

so lautet die standardisierte Zufallsvariable:

$$Z = \frac{X - \mu}{\sigma}$$

Auf dem 5 %-Signifikanzniveau gilt dann analog zu Beispiel 4.4.11

$$-1.96 \leq \frac{X - \mu}{\sigma} \leq 1.96$$

Wir lösen wieder zwischen den Ungleichheitszeichen nach X auf. Wir multiplizieren mit σ , addieren μ und erhalten

$$\mu - 1.96 \cdot \sigma \leq X \leq \mu + 1.96 \cdot \sigma$$

Das heisst, ein normalverteilter Messwert X liegt mit Wahrscheinlichkeit 95 % im Intervall

$$[\mu - 1.96 \cdot \sigma, \mu + 1.96 \cdot \sigma]$$

Dieses Intervall macht eine Aussage über die Wahrscheinlichkeit, ein welchen Bereich die Messwerte fallen, wenn μ bekannt ist.

Nun kommt die entscheidende Überlegung für das Vertrauensintervall: Wir können die Ungleichung

$$-1.96 \leq \frac{X - \mu}{\sigma} \leq 1.96$$

auch nach μ auflösen. Wir erhalten dann:

$$X - 1.96 \cdot \sigma \leq \mu \leq X + 1.96 \cdot \sigma$$

Diese Ungleichung macht nun eine Aussage über dass wahre μ , wenn ein Messwert für X gegeben ist. Das heisst, das wahre μ liegt mit einer Wahrscheinlichkeit von 95 % im Intervall

$$[X - 1.96 \cdot \sigma, X + 1.96 \cdot \sigma]$$

Wie ist dies nun zu verstehen?

Beispiel 4.4.13

Wir betrachten $X \sim \mathcal{N}(6, 2^2)$ und simulieren einen Wert. (zu R)

```
from scipy.stats import norm
import numpy as np

norm.rvs(size=1, loc=6, scale=2)
print(norm.rvs(size=1, loc=6, scale=2) )

## [4.77648717]
```

Dann bilden wir das Intervall

$$I = [4.78 - 1.96 \cdot 2, 4.78 + 1.96 \cdot 2]$$

und erhalten

$$I = [0.86, 8.7]$$

Der wahre Wert, in diesem Beispiel 6, liegt nun tatsächlich in diesem Intervall.

Ist dies aber immer der Fall?

Wir simulieren nun 100 Messwerte und bestimmen für alle das Intervall oben. Diese Intervalle sind in Abbildung 4.20 vertikal eingezeichnet. Die horizontale blaue Linie entspricht dem wahren Mittelwert 6.

Die schwarz eingezeichneten Intervalle enthalten den wahren Wert $\mu = 6$ nicht. Dies sind hier genau 5 Intervalle. Demnach enthalten 95 % der Intervalle den wahren Wert $\mu = 6$. (zu R)

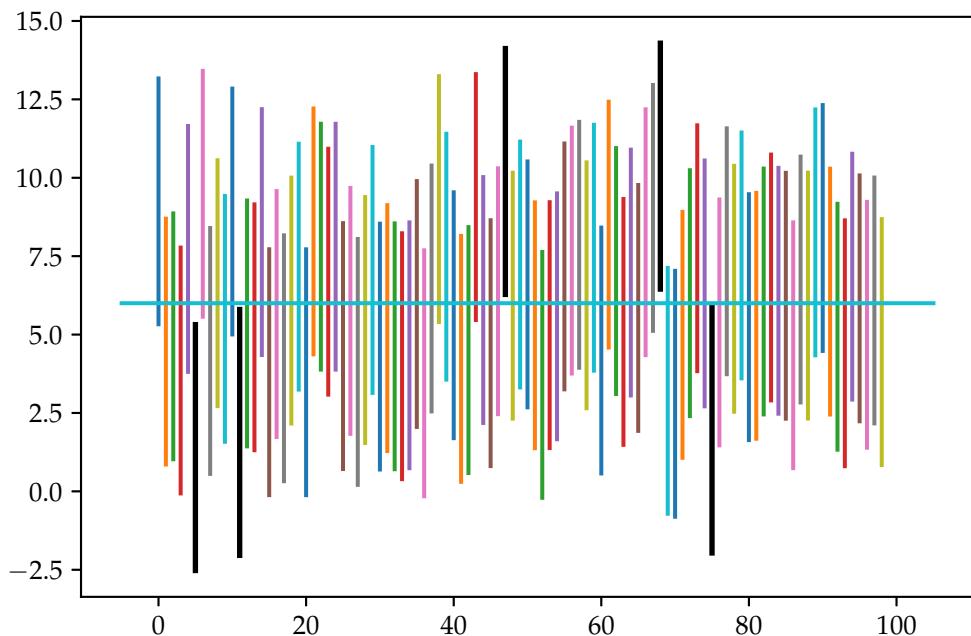


Abbildung 4.20.: 100 Vertrauensintervalle der Normalverteilung mit $\mu = 6$ und $\sigma = 2$

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

x = norm.rvs(loc=6, scale=2, size=100)
```

```

yu = x - 1.96*2
yo = x + 1.96*2

for i in range(99):
    plt.plot([i, i], [yu[i], yo[i]])
    if (yu[i] <= 6 <= yo[i]) == False:
        plt.plot([i, i], [yu[i], yo[i]], c="black", linewidth=2)

plt.plot([-5, 105], [6, 6])

plt.show()

```

Somit folgt für die Interpretation eines Vertrauensintervalls wie

$$I = [0.86, 8.7]$$

aus dem vorangehenden Beispiel : Mit einer Wahrscheinlichkeit von 95 % liegt das wahre (aber unbekannte) μ in diesem Intervall.

□

Wir sind in diesem Abschnitt bis jetzt immer von einem Signifikanzniveau von 5 % ausgegangen. Dies ist zwar oft der Fall, muss aber nicht notwendigerweise so sein.

Für ein allgemeines Signifikanzniveau α ersetzen wir in der Herleitung oben einfach -1.96 durch $z_{\frac{\alpha}{2}}$ und 1.96 durch $z_{1-\frac{\alpha}{2}}$. Das allgemeine Vertrauensintervall lautet dann

$$I = [x - z_{1-\frac{\alpha}{2}} \cdot \sigma, x + z_{\frac{\alpha}{2}} \cdot \sigma]$$

Beispiel 4.4.14

Für $\alpha = 0.01$ gilt dann ([zu R](#))

$$I = [x - 2.58 \cdot \sigma, x + 2.58 \cdot \sigma]$$

```

from scipy.stats import norm

norm.ppf(q=[0.005, 0.995])

## [-2.5758293  2.5758293]

```

□

Vertrauensintervall allgemein

Das sogenannte *Vertrauensintervall* oder *Konfidenzintervall* bei Messdaten besteht aus denjenigen Werten μ , bei denen der entsprechende Test die Nullhypothese *nicht* verwirft. Das sind also alle Parameterwerte des Zufallsmodells, bei denen die Daten recht wahrscheinlich oder plausibel sind.

Das Vertrauensintervall enthält dann das wahre, aber meist unbekannte μ mit einer gegebenen Wahrscheinlichkeit. Zum Beispiel ist das 95 %-Vertrauensintervall für μ dasjenige Intervall, das μ mit einer Wahrscheinlichkeit von 0.95 enthält. Das heißt, wenn wir sehr viele Testreihen machen und jeweils das Vertrauensintervall bestimmen, so wird μ in 95 % dieser Intervalle enthalten sein.

Ist das Signifikanzniveau α , so nennen wir das Intervall $(1 - \alpha) \cdot 100\%$ -Vertrauensintervall.

Bemerkungen:

- i. Liegt der Mittelwert einer Messreihe im *Verwerfungsbereich*, so wird die Nullhypothese *verworfen*.
- ii. Liegt der Mittelwert einer Messreihe im *Vertrauensintervall*, so wird die Nullhypothese *nicht verworfen*.
- iii. Wir haben hier nur das Vertrauensintervall für μ betrachtet. Man kann aber das Vertrauensintervall für jeden Parameter bestimmen.

Vertrauensintervalle für μ einer Messreihe

Wir gehen nun wieder von *Messreihen* aus und nehmen wiederum an, dass die Daten Realisierungen von

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu, \sigma_X^2)$$

sind.

Wir müssen wieder unterscheiden, ob σ_X bekannt oder unbekannt ist.

Vertrauensintervalle, falls σ_X bekannt

Diesen Fall haben wir in der Einleitung betrachtet. Der Mittelwert \bar{X}_n folgt der Verteilung

$$\bar{X}_n \sim \mathcal{N} \left(\mu, \sigma_{\bar{X}_n}^2 \right) = \mathcal{N} \left(\mu, \frac{\sigma_X^2}{n} \right)$$

Ersetzen wir in der Formel der Einleitung

$$[x - z_{1-\frac{\alpha}{2}} \cdot \sigma, x + z_{\frac{\alpha}{2}} \cdot \sigma]$$

den Messwert x durch den Mittelwert \bar{x}_n der Messreihe und σ durch $\sigma_{\bar{X}_n} = \frac{\sigma_X}{\sqrt{n}}$, so erhalten wir

$$\left[\bar{x}_n - z_{1-\frac{\alpha}{2}} \cdot \frac{\sigma_X}{\sqrt{n}}, \bar{x}_n + z_{\frac{\alpha}{2}} \cdot \frac{\sigma_X}{\sqrt{n}} \right]$$

Beispiel 4.4.15 Methode A zur Bestimmung der Schmelzwärme

Beim Schmelzwärmedatensatz im Beispiel 4.3.1 sei $\mu = 80$ und $\sigma_X = 0.02$. Die Standardabweichung wird hier also als bekannt angenommen. Für den Mittelwert haben wir 80.02 erhalten.

Es gilt

$$z_{0.975} = 1.96$$

Dann ist das zweiseitige Konfidenzintervall für die mit Methode A gemessene Schmelzwärme

$$I = 80.02 \pm 1.96 \cdot 0.02 / \sqrt{13} = [80.007, 80.033]$$

Insbesondere liegt 80.00 nicht im Intervall I . Der Wert $\mu = 80.00$ ist folglich nicht mit den Daten kompatibel, was wir bereits mit Hilfe des z -Tests ermittelt hatten.

Die Berechnung von Hand ist ein bisschen mühsam, aber mit **Python** ist es einfach. Wir berechnen das Vertrauensintervall wie folgt: ([zu R](#))

```
from scipy.stats import norm, t
import numpy as np

norm.interval(alpha=0.95, loc=80.02, scale=0.024/np.sqrt(13))

## (80.00695369111817, 80.03304630888182)
```

□

Allgemein gilt für bekanntes σ_X :

Zweiseitiges und Einseitiges Vertrauensintervall

Dies führt dann auf die folgenden *zweiseitigen Vertrauensintervalle* (die dazugehörigen Tests sind zweiseitig mit Alternative $H_A: \mu \neq \mu_0$) zum Niveau $1 - \alpha$:

$$\left[\bar{x}_n - z_{1-\alpha/2} \cdot \frac{\sigma_X}{\sqrt{n}}, \bar{x}_n + z_{1-\alpha/2} \cdot \frac{\sigma_X}{\sqrt{n}} \right]$$

Analog kann man auch *einseitige Vertrauensintervalle* konstruieren. Sie enthalten alle Parameter, bei denen ein einseitiger Test nicht verwerfen würde. Beim z-Test sehen die einseitigen $(1 - \alpha)$ -Vertrauensintervalle so aus:

$$\begin{aligned} \text{Falls } H_A: \mu < \mu_0: & \left(-\infty, \bar{x}_n + z_{1-\alpha} \cdot \frac{\sigma_X}{\sqrt{n}} \right] \\ \text{Falls } H_A: \mu > \mu_0: & \left[\bar{x}_n - z_{1-\alpha} \cdot \frac{\sigma_X}{\sqrt{n}}, \infty \right) \end{aligned}$$

Vertrauensintervalle, falls σ_X unbekannt

Ist σ_X unbekannt, so verwenden wir die t -Verteilung und die geschätzte Standardabweichung $\hat{\sigma}_X$. Wir ersetzen also in der grünen Box oben $z_{1-\alpha/2}$ bzw. z_α durch $t_{n-1, 1-\frac{\alpha}{2}}$ bzw. $t_{n-1, \alpha}$ und σ_X durch $\hat{\sigma}_X$.

Bei einem zweiseitigen t -Test hat der Verwerfungsbereich die Form

$$K = (-\infty, -t_{n-1, 1-\frac{\alpha}{2}}] \cup [t_{n-1, 1-\frac{\alpha}{2}}, \infty)$$

Der t -Test verwirft H_0 nicht, wenn der Wert der Teststatistik nicht im Verwerfungsbereich der Teststatistik ist. Wenn H_0 nicht verworfen wird, muss also gelten:

$$-t_{n-1, 1-\frac{\alpha}{2}} \leq \frac{\sqrt{n}(\bar{x}_n - \mu_0)}{\hat{\sigma}_X} \quad \text{und} \quad t_{n-1, 1-\frac{\alpha}{2}} \geq \frac{\sqrt{n}(\bar{x}_n - \mu_0)}{\hat{\sigma}_X}$$

Um das zweiseitige Vertrauensintervall von μ zu finden, müssen wir alle Werte von μ_0 finden, die obige Gleichungen erfüllen. Am einfachsten geht das, wenn wir beide Gleichungen nach μ_0 auflösen:

$$\mu_0 \leq \bar{x}_n + \frac{\hat{\sigma}_X \cdot t_{n-1, 1-\frac{\alpha}{2}}}{\sqrt{n}} \quad \text{und} \quad \mu_0 \geq \bar{x}_n - \frac{\hat{\sigma}_X \cdot t_{n-1, 1-\frac{\alpha}{2}}}{\sqrt{n}}$$

Zweiseitiges und Vertrauensintervall

Dies führt dann auf die folgenden *zweiseitigen Vertrauensintervalle* (die dazuge-

hörigen Tests sind zweiseitig mit Alternative $H_A: \mu \neq \mu_0$) zum Niveau $1 - \alpha$:

$$\left[\bar{x}_n - t_{n-1, 1-\alpha/2} \cdot \frac{\hat{\sigma}_X}{\sqrt{n}}, \bar{x}_n + t_{n-1, 1-\alpha/2} \cdot \frac{\hat{\sigma}_X}{\sqrt{n}} \right]$$

Analog kann man auch *einseitige Vertrauensintervalle* konstruieren. Sie enthalten alle Parameter, bei denen ein einseitiger Test nicht verwerfen würde. Beim t -Test sehen die einseitigen $(1 - \alpha)$ -Vertrauensintervalle wie folgt aus:

$$\begin{aligned} \text{Falls } H_A: \mu < \mu_0: & \left(-\infty, \bar{x}_n + t_{n-1, 1-\alpha} \cdot \frac{\hat{\sigma}_X}{\sqrt{n}} \right] \\ \text{Falls } H_A: \mu > \mu_0: & \left[\bar{x}_n - t_{n-1, 1-\alpha} \cdot \frac{\hat{\sigma}_X}{\sqrt{n}}, \infty \right) \end{aligned}$$

Beispiel 4.4.16 Methode A zur Bestimmung der Schmelzwärme

Wir haben

$$n - 1 = 13 - 1 = 12$$

Freiheitsgrade und [\(zu R\)](#)

$$t_{12, 0.975} = 2.18$$

```
from scipy.stats import t

t.ppf(q=0.975, df=12)

## 2.1788128296634177
```

Die mittlere mit Methode A gemessene Schmelzwärme ist

$$\bar{x}_n = 80.02$$

und die Standardabweichung lautet

$$\hat{\sigma}_X = 0.024$$

Dann ist das zweiseitige Konfidenzintervall für die mit Methode A gemessene Schmelzwärme also gegeben durch

$$I = 80.02 \pm 2.18 \cdot 0.024 / \sqrt{13} = [80.01, 80.03]$$

Insbesondere liegt 80.00 nicht im Intervall I . Der Wert $\mu = 80.00$ ist folglich nicht mit den Daten kompatibel, was wir bereits mit Hilfe des t -Tests ermittelt hatten.

Mit [Python](#)-Befehl berechnen wir das Vertrauensintervall wie folgt: [\(zu R\)](#)

```
import scipy.stats as st
from scipy.stats import norm, t
import numpy as np

t.interval(alpha=0.95, df=12, loc=80.02, scale=0.024/np.sqrt(13))

## (80.00549694515017, 80.03450305484982)
```

□

4.5. Statistische Tests bei nicht-normalverteilten Daten

Der z - und t -Test sind optimal, falls die Daten Realisierungen von normalverteilten Zufallsvariablen sind, also

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu, \sigma_X^2)$$

Optimalität bedeutet hier, dass der Test die grösste Macht hat.

Wir betrachten in diesem Kapitel die allgemeinere Situation, in der die Daten Realisierungen sind von

$$X_1, \dots, X_n \text{ i.i.d.}$$

wobei X_i einer *beliebigen* Verteilung folgen kann. Wir bezeichnen mit μ einen Lageparameter der Verteilung (z.B. μ = Median der Verteilung von X_i). Die Nullhypothese ist von der Form

$$H_0 : \mu = \mu_0$$

4.5.1. Der Vorzeichentest

Der Vorzeichentest testet Hypothesen über den Median der Verteilung von X_i , den wir hier mit μ bezeichnen. Im Falle einer symmetrischen Verteilung ist

$$\mu = E(X_i)$$

Beim Vorzeichentest betrachten wir die Anzahl Beobachtungen, die grösser oder kleiner als μ sind.

Methode A	79.98	80.04	80.02	80.04	80.03	80.03	80.04	79.97	80.05
Methode A	80.03	80.02	80.00	80.02					

Table 4.4.: Messung der Schmelzwärme mit der Methode A.

Beispiel 4.5.1

Wir betrachten nochmals die Daten zur Messung der Schmelzwärme mit der Methode A (siehe Tabelle 4.4).

Wir berechnen nun die *Unterschiede* zum vermuteten Median $\mu = 80.00$ (siehe Tabelle 4.5).

k	x_k	$x_k - \mu_0$	Vorzeichen
1	79.98	-0.02	-
2	80.04	0.04	+
3	80.02	0.02	+
4	80.04	0.04	+
5	80.03	0.03	+
6	80.03	0.03	+
7	80.04	0.04	+
8	79.97	-0.03	-
9	80.05	0.05	+
10	80.03	0.03	+
11	80.02	0.02	+
12	80.00	0.00	+
13	80.02	0.02	+

Table 4.5.: Vorzeichen der Schmelzwärme Messung mit der Methode A in Bezug zum Median.

Bei der Testdurchführung untersuchen wir die Anzahl Erfolge „+“ gegenüber der Anzahl Misserfolgen „-“. Ist $\mu = 80.00$ tatsächlich der Median, so ist die Wahrscheinlichkeit, dass ein Messwert ein positives Vorzeichen hat, $\pi = 0.5$. Somit haben wir es hier mit einem sogenannten *Binomialtest* mit $n = 13$ und $\pi_0 = 0.5$ zu tun. Der Binomialtest ist ein Hypothesentest für die diskrete Binomialverteilung (siehe Abbildung 4.21). Das Vorgehen ist genau gleich wie beim Hypothesentest der Normalverteilung oder t -Verteilung.

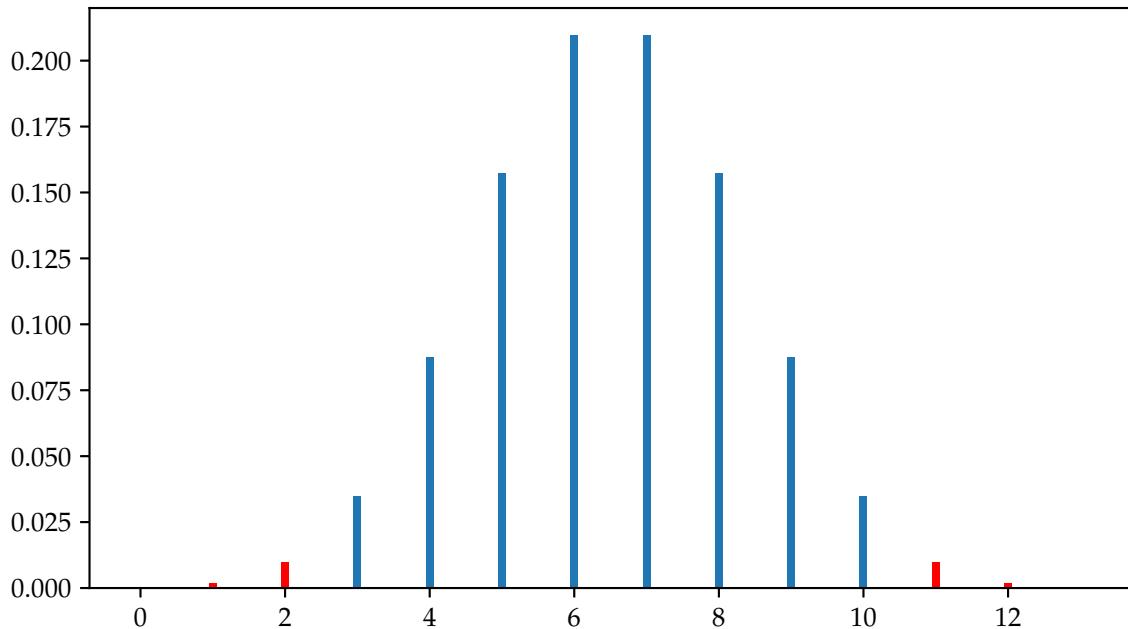


Abbildung 4.21.: Binomialverteilung für $n = 13$ und $\pi = 0.5$

Die Werte der Verteilung in Abbildung 4.21 aufaddiert ergeben 1. Für den Verwerfungsbereich betrachten wir die Balken auf der linken und rechten Seite der Verteilung, die auf Signifikanzniveau von 0.05 aufaddiert jeweils weniger als 0.025 ergeben. Dies sind die roten Balken in Abbildung 4.21.

Der Verwerfungsbereich ist dann

$$K = \{0, 1, 2\} \cup \{11, 12, 13\}$$

Die Grenzen des Verwerfungsbereich wurden hier auch mit Hilfe der Methode `.ppf(...)` zur Berechnung der Quantilen bestimmt: ([zu R](#))

```
from scipy.stats import binom
binom.ppf(q=[0.025, 0.975], n=13, p=0.5)
## [ 3. 10.]
```

Bei diskreten Verteilungen muss man immer noch überprüfen, ob die entsprechenden aufsummierten Werte gerade noch über- oder unterhalb der 0.025-Grenze liegen. ([zu R](#))

Kapitel 4. Statistik für Messdaten

```
from scipy.stats import binom

binom.cdf(k=3, n=13, p=0.5)
1 - binom.cdf(k=10, n=13, p=0.5)

## 0.046142578125000014
## 0.01123046875
```

Für $k = 3$ liegt der Wert der Summe überhalb 0.025, folglich müssen wir anstatt 3 den Wert 2 als untere Grenze des Verwerfungsbereichs wählen. Für $k = 10$ ist der Wert grösser als 0.975, und somit startet der Verwerfungsbereich bei $k = 11$.

Nun ist die Anzahl der positiven Vorzeichen $k = 11$. Dieser Wert liegt im Verwerfungsbereich, weshalb wir die Nullhypothese verwerfen können. Wir wollen noch den entsprechenden P -Wert berechnen: ([zu R](#))

```
from scipy.stats import binom

1 - binom.cdf(k=10, n=13, p=0.5)

## 0.01123046875
```

Da wir hier einen zweiseitigen Test betrachten, müssen wir diesen Wert mit 2 multiplizieren. Also lautet der P -Wert

$$P\text{-Wert} = 0.0225$$

Dieser Wert ist kleiner als das Signifikanzniveau, und somit wird die Nullhypothese verworfen. Wir können den P -Wert mit [Python](#) auch direkt berechnen: ([zu R](#))

```
import scipy.stats as st

st.binom_test(x=11, n=13, p=0.5, alternative="two-sided")

## 0.02246093750000003
```

Die Nullhypothese lautet

$$H_0 : \mu = \mu_0 = 80.00$$

Die Teststatistik V ist die Anzahl positiver Vorzeichen von $X_i - \mu_0$. Die realisierte Teststatistik ist dann $v = 11$, und der P -Wert des Vorzeichentests bei zweiseitiger Alternative

$$H_A : \mu \neq \mu_0 = 80.00$$

ist 0.02246 (beim t -Test war der P -Wert 0.0088). Wir können die Nullhypothese also sowohl beim Vorzeichentest wie auch beim t -Test *verwerfen*. Wir könnten mit **Python** auch einen einseitigen Test durchführen, indem wir für eine einseitig nach oben gerichtete Alternativhypothese `alternative='greater'`) und für eine einseitig nach unten gerichtete Alternativhypothese `alternative='less'`) in `st.binom_test()` setzen.

□

Wenn μ der Median der Verteilung von X ist, dann ist die Wahrscheinlichkeit, dass eine Realisierung von X grösser als μ ist, genauso gross wie die Wahrscheinlichkeit, dass eine Realisierung von X kleiner als μ ist, also

$$P(X > \mu) = 0.5$$

Zweiseitiger Vorzeichentest

Der (zweiseitige) Vorzeichen-Test ist folgendermassen aufgebaut:

1. *Modell*:

$$X_1, \dots, X_n \text{ i.i.d.}$$

wobei X_i eine beliebige Verteilung hat.

2. *Nullhypothese*:

$$H_0 : \mu = \mu_0 \quad (\mu \text{ ist der Median})$$

Alternative:

$$H_A : \mu \neq \mu_0 \quad (\text{oder einseitige Variante})$$

3. *Teststatistik*:

$$V : \text{Anzahl } X_i \text{'s mit } (X_i \geq \mu_0)$$

Verteilung der Teststatistik unter H_0 :

$$V \sim \text{Bin}(n, \pi_0) \quad \text{mit} \quad \pi_0 = 0.5$$

4. *Signifikanzniveau*:

$$\alpha$$

5. *Verwerfungsbereich für die Teststatistik*:

$$K = [0, c_u] \cup [c_o, n] \text{ falls } H_A : \mu \neq \mu_0$$

Die Grenzen c_u und c_o müssen mit der Binomialverteilung oder der Normalapproximation berechnet werden.

6. *Testentscheid:*

Entscheide, ob der beobachtete Wert der Teststatistik im Verwerfungsbereich der Teststatistik liegt.

Bemerkungen:

- i. Der Vorzeichentest ist also nichts anderes als ein Binomialtest. Wenn wir $\mu_0 = 0$ wählen, entspricht die Teststatistik gerade der Anzahl „+“ im Datensatz, daher der Name „Vorzeichentest“. Wenn Sie den Binomialtest verstanden haben, müssen Sie für den Vorzeichentest also gar nichts Neues lernen.
- ii. Beim einseitigen Vorzeichentest mit Alternativhypothese

$$\mu > \mu_0$$

ist die Teststatistik die Anzahl positiver Vorzeichen von

$$X_i - \mu_0$$

Dementsprechend zeigt der Verwerfungsbereich nach oben. Alternativ könnten wir aber auch die negativen Vorzeichen von

$$X_i - \mu_0$$

als Teststatistik definieren und die Alternativhypothese anpassen zu:

$$\mu < \mu_0$$

und damit auch den nach unten zeigenden Verwerfungsbereich.

- iii. Beim einseitigen Vorzeichentest mit Alternativhypothese

$$\mu < \mu_0$$

ist die Teststatistik die Anzahl positiver Vorzeichen von

$$X_i - \mu_0$$

Dementsprechend zeigt der Verwerfungsbereich nach unten. Alternativ könnten wir aber auch die negativen Vorzeichen von

$$X_i - \mu_0$$

als Teststatistik definieren und die Alternativhypothese anpassen zu:

$$\mu > \mu_0$$

und damit auch den jetzt nach oben zeigenden Verwerfungsbereich.

Der Vorzeichentest ist immer angebracht, falls die Daten Realisierungen von

$$X_1, \dots, X_n \text{ i.i.d.}$$

sind. Die Wahrscheinlichkeit für einen Fehler 1. Art wird kontrolliert durch das Signifikanzniveau α bei beliebiger Verteilung der X_i 's.

Vom Standpunkt der Macht gibt es keine eindeutige Antwort, ob der Vorzeichen- oder der t -Test besser ist. Wenn die Verteilung der X_i langschwängig ist, kann der Vorzeichentest grössere Macht haben. Weil der Vorzeichentest die Information, um wie viel die X_i von dem Wert μ_0 abweichen (siehe die Definition der Teststatistik V oben), nicht ausnützt, kann die Macht aber auch wesentlich schlechter sein als beim t -Test.

4.5.2. Der Wilcoxon-Test

Der Wilcoxon-Test ist ein Kompromiss, der weniger voraussetzt als der t -Test, die Information der Daten aber besser ausnützt als der Vorzeichentest. Der Wilcoxon-Test setzt bloss voraus, dass die Verteilung unter der Nullhypothese *symmetrisch* bezüglich μ_0 ist.

Als erstes bildet man die Ränge der Differenzen bezüglich des Absolutwertes:

$$\text{Rang}(|X_i - \mu_0|) = k$$

wobei $|X_i - \mu_0|$ den k -ten kleinsten Wert hat unter

$$|X_1 - \mu_0|, \dots, |X_n - \mu_0|$$

Wenn einzelne Werte zusammenfallen, teilt man die Ränge auf durch Mittelung. Falls für ein X_i

$$|X_i - \mu_0| = 0$$

ist, dann lässt man diesen Datenpunkt weg. Ferner sei V_i der Indikator dafür, ob

$$X_i - \mu_0$$

positiv ist, d.h.

$$V_i = 1$$

falls

$$X_i > \mu_0$$

ist und $V_i = 0$ sonst. Dann verwirft man die Nullhypothese, falls

$$W = \sum_{i=1}^n \text{Rang}(|X_i - \mu_0|) V_i$$

entweder zu gross oder zu klein ist, je nach Spezifikation der Alternative. Wir gehen an dieser Stelle nicht darauf ein, wie die Verteilung von W beschaffen ist. Wir werden den Testentscheid direkt am P -Wert des Computer-Outputs ablesen.

Man kann zeigen, dass dieser Test das Niveau exakt einhält: Die Wahrscheinlichkeit für einen Fehler 1. Art ist gleich α , wenn die X_i i.i.d. sind und eine um μ_0 symmetrische Dichte haben.

Beim t -Test wird das Niveau auch ungefähr eingehalten bei vielen nicht-normalen Verteilungen (wegen dem zentralen Grenzwertsatz), aber die Wahrscheinlichkeit eines Fehlers 2. Art ist unter Umständen beim t -Test viel grösser als beim Wilcoxon-Test.

Beispiel 4.5.2

Für die Schmelzwärme der Methode A sind die Ränge und die V_i 's in Tabelle 4.6 dargestellt.

k	x_k	$ x_k - \mu_0 $	Rang($ x_k - \mu_0 $)	V_k
1	79.98	0.02	2.5	0
2	80.04	0.04	10	1
3	80.02	0.02	2.5	1
4	80.04	0.04	10	1
5	80.03	0.03	6.5	1
6	80.03	0.03	6.5	1
7	80.04	0.04	10	1
8	79.97	0.03	6.5	0
9	80.05	0.05	12	1
10	80.03	0.03	6.5	1
11	80.02	0.02	2.5	1
12	80.00	0.00		1
13	80.02	0.02	2.5	1

Table 4.6.: Daten und entsprechende Ränge im Beispiel der Schmelzwärme Messung mit der Methode A .

Betrachten wir den Wert $x_1 = 79.98$. Der kleinste Wert ist

$$|x_1 - \mu_0| = |79.98 - 80.00| = 0.02$$

da der Fall

$$|x_{12} - \mu_0| = |80.00 - 80.00| = 0$$

weggelassen wird.

Folglich ist der Rang von $x_1 = 79.98$ gleich 1. Nun gibt es drei weitere Werte mit einem Absolutbetrag der Differenz von 0.2. Somit teilen wir die Summe der Ränge 1, 2, 3 und 4 durch vier, was den Wert 2.5 ergibt.

Es gibt 12 Differenzen $x_i - \mu_0$ und die gesamte Rangsumme ist

$$1 + 2 + 3 + \dots + 12 = \frac{13 \cdot 12}{2} = 78$$

Die Idee ist nun, dass die Nullhypothese nicht verworfen wird, wenn die Rangsumme W nicht allzu sehr von der Mitte der Rangsumme 39 abweicht.

Die Rangsumme lautet in unserem Beispiel

$$W = 10 + 2.5 + 10 + 6.5 + 6.5 + 10 + 0 + 12 + 6.5 + 2.5 + 0 + 2.5 = 69$$

Die Nullhypothese lautet

$$H_0: \mu = \mu_0 = 80.00$$

Der Wilcoxon-Test ergibt dann bei zweiseitiger Alternative

$$H_A : \mu \neq \mu_0 = 80.00$$

einen *P*-Wert von 0.0195.

Mit **Python** erfolgt der Wilcoxon-Test folgendermassen: ([zu R](#))

Bemerkungen:

- i. Python bewertet $x_i - \mu_0 < 0$ mit 1. Darum steht für den Statistikwert auch 9 anstatt 69.

Da die Verteilung beim Wilcoxon-Test als symmetrisch angenommen wird, ist das Resultat von 9 negativen Werten äquivalent zu den 69 positiven Werten.

1

Wilcoxon-Test

Der Wilcoxon-Test ist in den allermeisten Fällen dem t -Test oder Vorzeichentest vorzuziehen: er hat in vielen Situationen oftmals wesentlich grössere Macht und selbst in den ungünstigsten Fällen ist er nie viel schlechter.

Wenn man trotzdem den t -Test verwendet, dann sollte man die Daten auch grafisch ansehen, damit wenigstens grobe Abweichungen von der Normalverteilung entdeckt werden. Insbesondere sollte der Normal-Plot (siehe Abschnitt 4.1) angeschaut werden.

4.6. Statistische Tests bei zwei Stichproben

Wir besprechen in diesem Kapitel statistische Methoden, um einen Vergleich zwischen zwei Gruppen, Versuchsbedingungen oder Behandlungen hinsichtlich der Lage der Verteilung anzustellen.

4.6.1. Gepaarte Stichproben

Struktur der Daten

Wann immer möglich sollte man eine Versuchseinheit beiden Versuchsbedingungen unterwerfen. Es liegt eine *gepaarte Stichprobe* vor, wenn

- beide Versuchsbedingungen an derselben Versuchseinheit eingesetzt werden
- oder wenn jeder Versuchseinheit aus der einen Gruppe genau eine Versuchseinheit aus der anderen Gruppe zugeordnet werden kann.

Die Daten sind dann von der folgenden Struktur:

$$\begin{aligned}x_1, \dots, x_n &\text{ unter Versuchsbedingung 1} \\y_1, \dots, y_n &\text{ unter Versuchsbedingung 2}\end{aligned}$$

Notwendigerweise ist dann die Stichprobengrösse n für beide Versuchsbedingungen dieselbe. Zudem sind x_i und y_i abhängig, weil die Werte von der gleichen Versuchseinheit kommen.

Beispiel 4.6.1

Wir testen den Muskelzuwachs aufgrund von Krafttraining. Dazu messen wir die Kraft von 10 Testpersonen zu Beginn des Trainings. Anschliessend durchlaufen alle Testpersonen ein 6-wöchiges Trainingsprogramm. Dann wird die Kraft erneut gemessen.

Für jede Testperson gibt es also zwei Messungen: Vorher und nachher, wobei die Zuordnung eindeutig ist. Somit handelt es sich um gepaarte Stichproben.



Beispiel 4.6.2

Die Wirksamkeit von Augentropfen zur Reduktion des Augeninnendrucks soll untersucht werden. Wir haben 12 Patienten. Bei jedem Patienten wählen wir zufällig ein Auge aus. In das eine Auge kommen die Augentropfen mit dem Wirkstoff. In das andere Auge kommen Tropfen ohne Wirkstoff (Placebo).

Für jede Testperson haben wir also zwei Messungen: Eine für das rechte, die andere für das linke Auge, wobei die Zuordnung eindeutig ist. Somit handelt es sich um gepaarte Stichproben.



Beispiel 4.6.3

Wir haben eine Gruppe von 15 eineiigen Zwillingen, die sich für eine Studie für ein Haarwuchsmittel gemeldet haben. Bei jedem Zwillingspaar wird eine Person zufällig ausgewählt und erhält das Medikament. Die andere Person des Zwillingspaars erhält ein Placebo. Nach drei Wochen misst man den Haarwuchs.

Zu jeder Person aus der Gruppe mit Haarwuchsmittel kann man eindeutig eine Person aus der Gruppe ohne Haarwuchsmittel zuordnen. Somit handelt es sich um gepaarte Stichproben.



Statistischer Test für gepaarte Stichproben

Bei der Analyse von gepaarten Vergleichen arbeitet man mit den Differenzen innerhalb der Paare,

$$d_i = x_i - y_i \quad (i = 1, \dots, n),$$

welche wir als Realisierungen von i.i.d. Zufallsvariablen

$$D_1, \dots, D_n$$

auffassen. Kein Unterschied zwischen den beiden Versuchsbedingungen heisst dann einfach

$$E[D_i] = 0$$

(oder auch $\text{Median}(D_i) = 0$, je nach Test). Statistische Tests dafür sind in Unterkapitel 4.3 beschrieben: Falls die Daten normalverteilt sind, eignet sich ein *t*-Test.

Beispiel 4.6.4

Einen *t*-Test für gepaarte Stichproben wird in **Python** mit dem Befehl **ttest_rel** durchgeführt: ([zu R](#))

```
import scipy.stats as st
from scipy.stats import norm, t, binom
import numpy as np
from pandas import Series

vorher = Series([25, 25, 27, 44, 30, 67, 53, 53, 52, 60, 28])
nachher = Series([27, 29, 37, 56, 46, 82, 57, 80, 61, 59, 43])
st.ttest_rel(nachher, vorher)
print(st.ttest_rel(nachher, vorher))

## Ttest_relResult(statistic=4.271608818429545, pvalue=0.0016328499219996722)
```

Für das Vertrauenintervall erhalten wir ([zu R](#))

```
vorher = Series([25, 25, 27, 44, 30, 67, 53, 53, 52, 60, 28])
nachher = Series([27, 29, 37, 56, 46, 82, 57, 80, 61, 59, 43])
dif = nachher - vorher

t.interval(alpha=.95, df=dif.size-1, loc=dif.mean(), scale=dif.std()/np.sqrt(dif.size))

## (4.91430993515407, 15.631144610300478)
```

□

Bemerkungen:

- i. Man hätte auch direkt die Differenzen d_i von den gepaarten Stichproben berechnen und einen t -Test für eine Stichprobe durchführen können.
- ii. Der Unterschied der Gruppenmittelwerte hat bei einer zweiseitigen Alternative einen P -Wert von 0.0016 und ist somit auf dem 5 % Signifikanzniveau signifikant.
- iii. Die Teststatistik folgt unter der Nullhypothese einer t -Verteilung mit $\text{df} = 10$ Freiheitsgraden. Der Wert der Teststatistik ist 4.27
- iv. Der Unterschied *nachher-vorher* erscheint im Funktionsaufruf, indem sich das erste Argument auf das „nachher“ und das zweite Argument auf das „vorher“ bezieht.
- v. Das 95 %-Vertrauensintervall für diese Differenz ist gegeben durch

$$[4.91, 15.63]$$

Alternativ kommt ein Vorzeichentest oder ein Wilcoxon-Test in Frage. Dabei ist zu beachten, dass die vorausgesetzte Symmetrie für die Verteilung von D_i beim Wilcoxon-Test immer gilt unter der Nullhypothese, dass X_i und Y_i dieselbe Verteilung haben.

4.6.2. Ungepaarte Stichproben

Oft ist es nicht möglich, jeder Behandlungseinheit aus der einen Gruppe eine Behandlungseinheit aus der zweiten Gruppe eindeutig zuzuordnen. In diesem Fall ist eine Paarung nicht möglich, und man spricht von *ungepaarten* Stichproben. Hier muss die Zuordnung zur Behandlungsgruppe durch das Los erfolgen, um systematische Fehler zu vermeiden. (vgl. Abschnitt ?? unten).

Struktur der Daten

Bei ungepaarten Stichproben hat man Daten x_1, \dots, x_n und y_1, \dots, y_m (siehe Abschnitt 4.6.2), welche wir als Realisierungen der folgenden Zufallsvariablen auffassen:

$$X_1, \dots, X_n \text{ i.i.d.}$$

$$Y_1, \dots, Y_m \text{ i.i.d.}$$

wobei auch alle X_i 's von allen Y_j 's unabhängig sind. Bei einer solchen zufälligen Zuordnung von Versuchseinheiten zu einer von zwei verschiedenen Versuchsbedingungen spricht man von einer ungepaarten Stichprobe. Im Allgemeinen ist in einer ungepaarten Stichprobe $m \neq n$, aber nicht notwendigerweise. Entscheidend ist, dass x_i und y_i zu verschiedenen Versuchseinheiten gehören und als unabhängig angenommen werden können.

Beispiel 4.6.5

Datensatz zu latenter Schmelzwärme von Eis im Kapitel 2.2. Wir haben die Schmelzwärme mit zwei verschiedenen Methoden hintereinander gemessen. Jede Messung ist entweder mit Methode A oder mit Methode B, aber nicht mit beiden gleichzeitig gemacht worden. Es gibt also keinen eindeutigen Zusammenhang zwischen den Messungen der Methode A und den Messungen der Methode B. Daher sind die beiden Stichproben ungepaart. Des weiteren bemerken wir, dass die Messreihen unterschiedlich viele Messwerte beinhalten.

□

Beispiel 4.6.6

Zufällige Zuordnung von 100 Testpatienten zu einer Gruppe der Grösse 50 mit Medikamentenbehandlung und zu einer anderen Gruppe der Grösse 50 mit Placebo-Behandlung. Es gibt keine eindeutige Zuordnung von einem Patienten aus der Medikamentengruppe zu einem Patienten in der Placebo-Gruppe. Daher handelt es sich um ungepaarte Stichproben, obwohl beide Gruppen gleich gross sind.

□

Zwei-Stichproben *t*-Test für ungepaarte Stichproben

Nehmen wir an, wir haben einen ungepaarten Datensatz mit

$$X_1, \dots, X_n \text{ i.i.d. } \sim \mathcal{N}(\mu_X, \sigma_X^2)$$

$$Y_1, \dots, Y_m \text{ i.i.d. } \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$$

Beispiel 4.6.7

Als Beispiel betrachten wir den Datensatz mit den Wärmekapazitäten und analysieren mit **Python**, ob es einen signifikanten Unterschied zwischen den beiden Messmethoden A und B gibt. ([zu R](#))

```
import scipy.stats as st
from scipy.stats import norm, t, binom
import numpy as np
from pandas import Series
import scipy.stats as st
```

Kapitel 4. Statistik für Messdaten

```
x = Series([79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04, 79.97, 80.05, 80.03,
y = Series([80.02, 79.94, 79.98, 79.97, 80.03, 79.95, 79.97])

st.ttest_ind(x, y, equal_var=False)
print(st.ttest_ind(x,y,equal_var=False))

## Ttest_indResult(statistic=2.839932638516127, pvalue=0.01866020947068376)
```

In der Zeile `statistic = ...` steht zunächst der beobachtete Wert der Teststatistik: $t = 2.8399$. Unter der Nullhypothese folgt die Teststatistik einer t -Verteilung mit $df = 19$ Freiheitsgraden. Das ergibt bei einer zweiseitigen Alternative einen P -Wert von 0.01866 . Der Unterschied ist also auf dem 5 % Signifikanzniveau signifikant, weil der P -Wert kleiner als 5 % ist. **Python** gibt im Falle von `ttest` den zweiseitigen P -Wert heraus. Der Computer berechnet auch das 95 %-Vertrauensintervall des Unterschieds in den Gruppenmittelwerten: Mit 95 % Wahrscheinlichkeit ist der Gruppenmittelwert von `x` um eine Zahl im Bereich $[0.0085, 0.0730]$ grösser als der Gruppenmittelwert von `y`. Die Null ist nicht enthalten, also ist der Unterschied der Mittelwerte signifikant.

□

Zwei-Stichproben Wilcoxon-Test (Mann-Whitney-Test)

Die Voraussetzungen für den Zwei-Stichproben Wilcoxon-Test, manchmal auch Mann-Whitney Test genannt, bezüglich

$$\begin{aligned} X_1, \dots, X_n &\text{ i.i.d.} \\ Y_1, \dots, Y_m &\text{ i.i.d.} \end{aligned}$$

sind wie folgt:

$$\begin{aligned} X_1, \dots, X_n \text{ i.i.d.} &\sim F_X \\ Y_1, \dots, Y_m \text{ i.i.d.} &\sim F_Y \end{aligned}$$

wobei F_X eine beliebige stetige Verteilungsfunktion mit

$$F_Y(x) = F_X(x - \delta)$$

Dies bedeutet, dass die Verteilung von Y_j die um δ verschobene Verteilung von X_i ist, denn:

$$P(Y_j \leq x + \delta) = F_Y(x + \delta) = F_X(x + \delta - \delta) = F_X(x) = P(X_i \leq x)$$

Die Berechnung des P -Werts eines Zwei-Stichproben Wilcoxon-Tests kann mittels Computer erfolgen. Aus den gleichen Gründen wie im Fall einer Stichprobe (siehe Kapitel 4.5) ist der Wilcoxon-Test im Allgemeinen dem t -Test vorzuziehen.

Beispiel 4.6.8

Wir berechnen noch Beispiel 4.6.7 mit dem Wilcoxon-Test für ungepaarte Stichproben. In **Python** wird Wilcoxon-Test für ungepaarte Stichproben mit **mannwhitneyu(x, y)** durchgeführt: (zu R)

```
import scipy.stats as st
from scipy.stats import norm, t, binom
import numpy as np
from pandas import Series

x = Series([79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04, 79.97, 80.05, 80.03])

y = Series([80.02, 79.94, 79.98, 79.97, 80.03, 79.95, 79.97])

st.mannwhitneyu(x, y, alternative='two-sided')

## MannwhitneyuResult(statistic=76.5, pvalue=0.014537644944738774)
```

Der zweiseitige P -Wert ist also durch 0.01454 gegeben, wobei dieser etwas kleiner ist als im Falle des t -Tests.

Bemerkungen:

- i. Mit **alternative="two-sided"** liefert uns die **Python**-Ausbabe den zweiseitigen P -Wert. Für den einseitigen P -Wert müssen wir im Falle einer einseitig nach oben gerichteten Alternativhypothese **alternative='greater'**, im Falle einer einseitig nach unten gerichteten Alternativhypothese **alternative='less'** verwenden.
- ii. Falls wir einen Wilcoxon-Test für zwei gepaarte Stichproben durchführen möchten, so verwenden wir den **Python**-Befehl **st.wilcoxon(x=..., y=..., correction=True)** wobei die Ausgabe des P -Wertes zweiseitig ist.



4.7. Statistische Signifikanz und fachliche Relevanz bei statistischen Tests

Der Begriff der statistischen Signifikanz wird oft missbraucht, um gleichzeitig auch die entsprechende fachliche Relevanz zu untermauern. Diese beiden Begriffe müssen

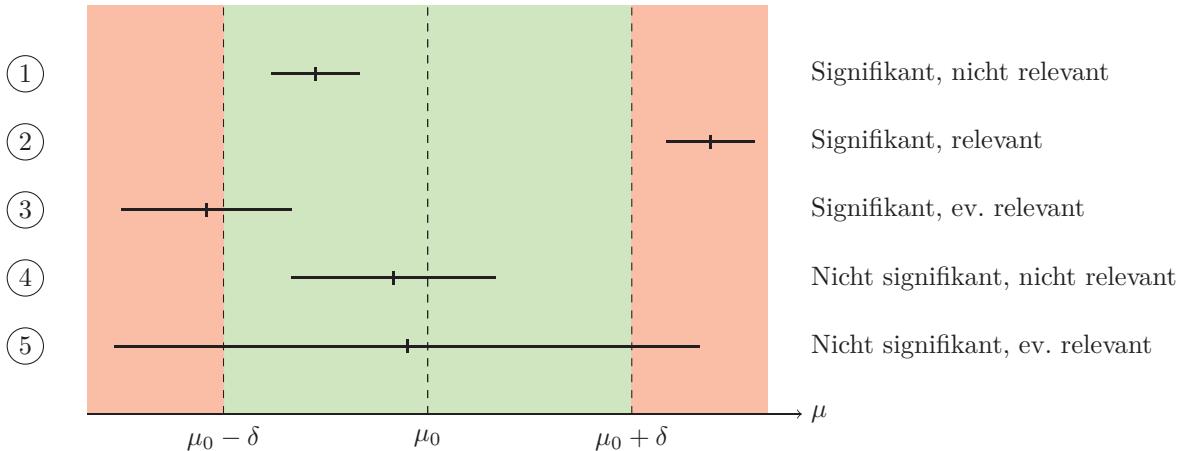


Abbildung 4.22.: Verschiedene Fälle (1 bis 5) von statistischer Signifikanz und fachlicher Relevanz. Die Vertrauensintervalle für μ sind durch Striche dargestellt. Der „irrelevante Bereich“ geht von $\mu_0 - \delta$ bis zu $\mu_0 + \delta$ (grün), wobei das δ durch entsprechendes Fachwissen definiert wurde.

aber nicht unbedingt miteinander einhergehen. Wenn man genügend viele Beobachtungen sammelt, dann wird man jede Nullhypothese verwerfen können (denn diese stimmt in der Praxis nie exakt). Bedeutet dies nun, dass die vorgestellten Konzepte in der Praxis alle nutzlos sind? Die Antwort ist natürlich nein, aber man muss sie richtig verwenden.

Hierzu müssen wir das Beste aus beiden Bereichen miteinander kombinieren: Entsprechendes Fachwissen und der statistische Output. Wir müssen zuerst basierend auf Fachwissen definieren, was ein relevanter Effekt oder Unterschied ist (die Statistik kann uns hier nicht helfen). Wenn wir dies gemacht haben, können wir die Statistik ins Spiel bringen.

Betrachten wir als Beispiel die Produktion von Schrauben: Nehmen wir an, dass Abweichungen bis 0.5 mm von der Solllänge 100 mm keine Rolle spielen, also nicht relevant sind. Wir haben also einen „irrelevanten Bereich“, der von 99.5 mm bis 100.5 mm geht. Ausserhalb sprechen wir vom *Relevanzbereich*. Die Idee besteht nun darin, zu schauen, wie das Vertrauensintervall für μ liegt. Angenommen wir haben in einer Stichprobe das Vertrauensintervall für das wahre μ zu [99.73, 99.99] bestimmt, was vollständig im irrelevanten Bereich liegt. Wir würden die Abweichung als statistisch signifikant, aber als nichtrelevant taxieren. Andere mögliche Fälle und deren Interpretation sind in Abbildung 4.22 dargestellt.

Konzeptionelle Lernziele

Sie sollten fähig sein, . . .

- das Konzept von einem QQ-Plot und von einem Normal-Plot zu erklären.
- zu entscheiden, ob ein Datensatz einer Normalverteilung folgt.
- das Konzept der Maximum Likelihood Methode und Momentenmethode zur Parameterschätzung zu erklären.
- die Maximum Likelihood Methode für die Parameterschätzung einer beliebigen Verteilungsfamilie anzuwenden.
- einen z - und t -Test für eine gegebene Messreihe sowohl von Hand wie mittels Computersoftware (bei einseitiger und zweiseitiger Alterntivhypothese) durchzuführen.
- den Unterschied zwischen gepaarten und ungepaarten Stichproben zu erklären.
- einen geeigneten Hypothesentest für gepaarte und ungepaarte Stichproben durchzuführen.
- das Konzept des p -Wertes zu erklären und den p -Wert für einen Datensatz und bei formulierter Hypothese zu berechnen.
- das Konzept von Bootstrapping im Zusammenhang mit der Parameterschätzung zu erklären.
- das Konzept des Vertrauensintervalls zu erklären, und zwar sowohl mit Hilfe der auf dem Hypothesentest beruhenden Definition und mit Hilfe von Bootstrapping.
- den p -Wert und das Vertrauensintervall für den Testentscheid zu verwenden.

Computer-Basierte Lernziele

Sie sollten fähig sein . . .

- einen QQ-Plot und einen Normal-Plot für eine Messreihe mit Hilfe von `probplot()` zu erzeugen.
- für gegebene Messreihe und Hypothese einen t -Test mit Hilfe von `st.ttest_1samp()` durchzuführen.
- das Vertrauensintervall mit Hilfe von `t.interval()` zu berechnen.
- für gegebene Messreihe und Hypothese einen Vorzeichen-Test mit Hilfe von `st.binom_test()` und einen Wilcoxon-Test mit Hilfe von `st.wilcoxon()` druchzuführen.

Kapitel 4. Statistik für Messdaten

- für gegebene gepaarte oder ungepaarte Stichproben und Hypothesen einen *t*-Test und Mann-Whitney-U Hypothesentest mit Hilfe von `st.ttest_ind()`, resp. `st.mannwhitneyu()` durchzuführen.

Kapitel 5.

Versuchsplanung

One day, when I was a junior medical student, a very important Boston surgeon visited the school and delivered a great treatise on a large number of patients who had undergone successful operations for vascular reconstruction. At the end of the lecture, a young student at the back of the room timidly asked „Do you have any controls?“ Well, the great surgeon drew himself up to his full height, hit the desk, and said, „Do you mean did I not operate one half of the patients?“ The hall grew very quiet then. The voice at the back of the room hesitantly replied, „Yes, that's what I had in mind.“ Then the visitor's fist really came down as he thundered, „Of course not. That would have doomed half of them to their death.“ God, it was quiet then, and one could scarcely hear the small voice ask, „Which half?“

(E.E. Peacock)

5.1. Einleitung

Bei wissenschaftlichen Studien werden empirische Daten entweder durch ein *Experiment* oder durch eine *Erhebung* erfasst. Nur in seltenen Fällen besteht eine freie Wahl zwischen diesen beiden Formen der Datenerfassung. Eine Unterscheidung ist aber

wichtig, da die beiden Formen zu unterschiedlichen Interpretationsmöglichkeiten der Resultate führen.

Bei *Experimenten* werden Subjekte oder Objekte im Rahmen einer *kontrolliert geschaffenen* Situation beobachtet. Die zu analysierenden Prädiktoren (Variablen) werden darin systematisch variiert und die übrigen möglichst ausgeschaltet. Mittels Messungen, Befragungen oder Beobachtungen wird das Geschehen festgehalten. Die daraus resultierenden Daten bilden die Basis für die Beantwortung der eigentlichen Fragen, die zum Experiment führten.

Beispiel 5.1.1 Polio Impfung

Polio hat in der ersten Hälfte des 20. Jahrhunderts Hunderttausende von Todesfällen verursacht und zwar vor allem von Kindern. Um 1950 gab es bereits mehrere Impfungen, die entwickelt wurden. Die vielversprechendste stammte von Jonas Salk. Die Labortests ergaben gute Resultate. Im Jahre 1954 hatten sich die Gesundheitsbehörden bereit erklärt, die Wirksamkeit der von Salk entwickelten Polio-Impfung an Menschen zu testen.

In diesem Medikamentenversuch wurde einer bestimmten Anzahl Personen die Impfung verabreicht, während eine Gruppe von Personen unbehandelt blieben. Um die Wirksamkeit der Polio-Impfung zu testen, wurde festgestellt, wieviele Personen in den beiden Gruppen an Polio erkrankten. Damit das Medikament wirksam ist, müssten in der Gruppe von behandelten Personen viel weniger Personen an Polio erkrankt sein, als in der unbehandelten Gruppe.



Bei einer *Erhebung* werden Subjekte oder Objekte im Rahmen einer existierenden (unkontrollierten) Situation beobachtet. Die zu analysierenden Prädiktoren (Variablen) sind jedoch darin nicht direkt einstellbar.

Bei den Erhebungen kann es sich um Beobachtungsstudien von natürlichen Abläufen handeln wie z.B. bei der Erfassung der Luftqualität in Fitnessräumen, Tiefgaragen, Tunnels, etc. oder um Stichproben-Erhebungen wie z.B. in Meinungsumfragen und anderen soziologischen Untersuchungen, aber auch in der Qualitätssicherung. In Erhebungen begnügt man sich mit der systematischen Erfassung von potentiellen Prädiktoren, damit ihre Effekte allenfalls später aus der Zielvariablen (Resultat des Versuches) herausgerechnet werden können.

Beispiel 5.1.2

In einer Studie möchte man den Fleischkonsum pro Haushalt pro Jahr erfassen. Als mögliche Prädiktoren werden in der Studie folgende Kategorien berücksichtigt:

1. Region

Kapitel 5. Versuchsplanung

2. Alter, Beruf, Bildungsstatus des führenden Elternteils, Haushaltsgrösse, Einkommen, Anzahl und Alter der Kinder
3. Soziale Umgebung, Gesundheitszustand der Familie

Diese Prädiktoren können bloss *beobachtet, aber nicht aktiv beeinflusst werden.*



Wir unterscheiden bei Erhebungen zwischen *Querschnitts-Studien*, *prospektiven Kohortenstudien* und *retrospektiven Fall-Kontroll-Studien*. Diese Studienarten beziehen die zeitliche Dimension eines Zustandes mit ein:

- Querschnittsstudien stellen eine Zeitaufnahme (Snapshot) einer Population zu einem bestimmten Zeitpunkt dar.
- Prospektive Fall-Kontroll-Studien versuchen, die Frage „Was wird passieren, wenn...?“ zu beantworten.
- Retrospektive Fall-Kontroll-Studien sind geeignet, um die Frage „Warum hat es sich auf diese Art und Weise entwickelt?“ zu beantworten.

Beispiel 5.1.3

Die Studie zum Fleischkonsum pro Haushalt und Jahr entspricht einer Querschnitts-Studie. Es wird z.B. der *momentane* Fleischkonsum untersucht. Oder der Fleischkonsum im Jahre 1925.



Beispiel 5.1.4

Das Risiko von Rauchern, an Lungenkrebs zu erkranken, und zwar im Vergleich zu Nicht-Rauchern, wird in *prospektiven Kohortenstudien* untersucht. Wir beginnen heute mit der Untersuchung und beobachten die (eventuelle) Diagnose von Lungenkrebs von Rauchern und Nicht-Rauchern in der Zukunft.



Beispiel 5.1.5

Der Vergleich zwischen gesunden und kranken Menschen in Bezug auf ihre Lebensart entspricht einer *retrospektiven Fall-Kontroll-Studie*. Wir untersuchen, wie sich die Menschen in der Vergangenheit verhalten haben und vergleichen dies mit dem Gesundheitszustand heute.

□

In vielen wissenschaftlichen Untersuchungen geht es um die Ergründung von *kausalen* Zusammenhängen (Ursache-Wirkungs-Beziehungen). Aus zwei Gründen sind mit Experimenten kausale Zusammenhänge im Allgemeinen viel einfacher nachzuweisen als mit Erhebungen:

1. Bei einem Experiment sind mindestens ein Teil der interessierenden Prädiktoren kontrolliert eingestellt. Bei Erhebungen ist man auf die Bedingungen angewiesen, die zur Zeit der Beobachtungsaufnahme herrschen.

Beispiel:

Der Einfluss von Temperatur, Luftfeuchtigkeit und CO₂-Gehalt auf die Luftqualität (Konzentration verschiedener Gase etc.) kann dann optimal bestimmt werden, wenn die entsprechenden Prädiktoren in einem möglichst weiten Bereich variieren. Das lässt sich in einem Versuch gezielt erreichen.

Bei einer Erhebung kann es vorkommen, dass eine Größe praktisch konstant bleibt oder in einem so kleinen Bereich variiert, dass ihr Einfluss auf die Zielgröße (die Luftqualität) unerheblich ist.

2. In einer Erhebung haben wir keine Kontrolle darüber, welcher Mechanismus die einzelnen „Versuchssubjekte“ den jeweiligen „Behandlungsgruppen“ zugeordnet hat. Im Falle von uns unbekannten Prädiktoren könnte es vorkommen, dass diese sowohl die Zielgröße als auch die Prädiktoren beeinflussen - solche verborgenen Größen werden *confounders* genannt. In diesem Fall würde eine Beziehung zwischen (beobachteten) Prädiktoren und Zielgröße festgestellt, obwohl es in Tat und Wahrheit keine solche zugrundeliegende Ursache-Wirkungs-Beziehung gibt.

Beispiel:

Es wurde eine relativ starke Korrelation zwischen Schokoladenkonsum und Nobelpreisdichte in einem Land festgestellt. Es ist unplausibel, dass es einen kausalen Zusammenhang zwischen Schokoladenkonsum und Nobelpreisdichte gibt. Diese Korrelation dürfte wohl eher auf einen in dieser Studie nicht berücksichtigten Prädiktor, zurückzuführen sein, der sowohl die Nobelpreisdichte als auch den Schokoladenkonsum beeinflusst.

Kapitel 5. Versuchsplanung

Zum Nachweis eines kausalen Zusammenhangs zwischen der Zielgrösse und den Prädiktoren sind also Experimente vorzuziehen. Es gibt aber Fragestellungen, bei denen man nur über Erhebungen zu einer Antwort kommen kann, weil Experimente z.B. aus ethischen, finanziellen oder technischen Gründen nicht durchführbar sind.

Beispiel 5.1.6

Der berühmte Statistiker R.A. Fisher, der in der Tabakindustrie tätig war, wehrte sich dagegen, dass Rauchen als Ursache für Lungenkrebs verantwortlich gemacht wurde. Sein Einwand ging dahin, dass unter Umständen ein versteckter Prädiktor (wie z.B. der Genotyp) dafür verantwortlich ist, dass jemand raucht und auch an Lungenkrebs erkrankt.

Eine Methode, diese Behauptung zu widerlegen, bestünde darin, eigentliche Nicht-Raucher in einer langwährenden Studie rauchen zu lassen, um festzustellen, ob es in dieser Gruppe eine wesentlich kleinere Rate an Lungenkrebskrankungen gibt als in der bestehenden Raucher-Gruppe. Ein solcher Versuch wäre wohl aber aufgrund ethischer Bedenken nicht durchführbar.

□

Wie können wir uns also gegen das Problem des confounding schützen? Das einzige Mittel ist die *Randomisierung*. Die zufällige Zuordnung von Versuchseinheiten zu den unterschiedlichen Behandlungen mittelt den Effekt von potentiellen Confoundern aus.

„Randomization generally costs little in time and trouble, but it can save us from disaster.“ (Oehlert)

Statistische Versuchsplanung

Statistische Versuchsplanung (eng. statistical design of experiment - DoE) bezeichnet den Prozess der Planung des Experiments, so dass die Daten mit statistischen Methoden zielführend ausgewertet werden können. Dahinter steckt die Einsicht, dass die statistische Auswertung die *einzig objektive Auswertungsmethode* ist, sobald die zu analysierenden Daten experimentellen Fehlern wie Messfehlern unterworfen sind. Die Qualität der Resultate wie auch deren Auswertung hängt allerdings wesentlich vom gewählten Versuchsplan (Durchführung des Versuches) ab. In einem wissenschaftlichen Umfeld ist es zwingend, die Planung und die Analyse insgesamt so auszulegen, dass die Schlussfolgerungen aus dem Experiment *stichhaltig und vorurteilslos* sind.

Kapitel 5. Versuchsplanung

Dieser Teil des Moduls ist der statistischen Seite der Planung von Experimenten und der Analyse der daraus erhaltenen Daten gewidmet. Wir werden uns hauptsächlich auf Experimente konzentrieren, denen biologische, chemische oder physikalische Prozesse und Systeme zu Grunde liegen. Solche Prozesse können z.B. Teil von Produkten sein. Oder es werden Systeme aus der Kommunikationstechnologie untersucht.

Um die betrachteten Prozesse und Systeme besser zu verstehen oder sie verbessern zu können, werden potentiell einflussreiche Prädiktoren (Eingangsgrößen) systematisch so verändert, dass die Gründe für die Veränderungen in den Ausgangsgrößen identifiziert und verstanden werden können.

Folgende Aspekte sollen hier etwas genauer betrachtet werden:

- *Vergleich von Behandlungen*

Eines der häufigsten Ziele bei Experimenten ist es, mehrere Behandlungsarten miteinander zu vergleichen und die beste auszuwählen. Zum Beispiel sollen sechs Weizensorten verglichen werden bezüglich ihres Ertrages und ihrer Resistenz gegen versalzene Böden. Falls es zwischen den Weizensorten Unterschiede gibt, stellt sich die Frage, inwiefern sie sich unterscheiden und welche die beste ist. Weitere Beispiele für den Vergleich können unterschiedliche Maschinentypen, verschiedene Lieferanten oder Herstellungsverfahren, etc. sein.

- *Variablen-Screening*

Falls es viele potentiell einflussreiche Größen in einem System oder Prozess gibt, jedoch nur wenige wirklich wichtig sind, will man mit einem Screening-Experiment diese wichtigen Größen identifizieren. Damit die Kosten niedrig gehalten werden können, sollte mit einer möglichst geringen Zahl von Experimenten die Aufgabe erledigt werden können.

- *Bestimmen von optimalen Einstellungen*

Sind die wichtigsten Prädiktoren auf den Prozess oder auf das System bestimmt, so will man häufig jene Einstellung suchen, die zu einem optimalen Prozessverhalten (z.B. Ertrag) führt. Dazu werden mit einfachen Approximationen die Zusammenhänge zwischen Ertrag (Zielgröße) und den Prädiktoren beschrieben und damit die Optimierung durchgeführt.

- *Systemrobustheit*

Neben der Optimierung ist es auch wichtig, dass das System oder der Prozess möglichst unempfindlich gegen unkontrollierbare Störungen ist. Solche Störungen schlagen sich vor allem in der Variabilität der Systemantwort (Zielgröße) nieder. Man nimmt hier vielleicht einen Qualitätsverlust in der Produktion in Kauf, dafür ist die Qualität immer gleich gut. Die Produktionsqualität hängt dann nicht von ändernder Temperatur, Luftfeuchtigkeit, etc. ab.

Bevor wir uns einigen Aspekten dieser weitläufigen Themen widmen, folgt zuerst eine Diskussion einiger allgemeiner Grundprinzipien der Versuchsplanung.

5.2. Grundelemente der Versuchsplanung

Versuchsplanung baut auf wenigen Grundelementen auf. Diese befassen sich mit spezifischen Anforderungen, die die Resultate des Versuchs erfüllen sollten. So wird gefordert, dass

- sowohl der Bias (d.h. der systematische Fehler) wie auch die Streuung der Messfehler möglichst klein sein soll,
- die Effekte von im Versuch unberücksichtigten Faktoren die Resultate nicht verfälschen sollen,
- sich die Schlussfolgerungen aus dem Experiment möglichst verallgemeinern lassen.

Bei den Grundelementen, die wir anhand der zwei folgenden Beispiele einführen und motivieren möchten, handelt es sich um die Unterscheidung zwischen *primären* und *sekundären* Variablen, um *Blockbildung*, um *Randomisierung* (d.h. die zufällige Abarbeitung der einzelnen Versuchsbedingungen (eng. Runs), um *Wiederholungen* von Messungen bei gleichen Versuchsbedingungen und um *balancierte* Versuchspläne.

Diese Grundelemente möchten wir anhand der zwei folgenden Beispiele einführen und motivieren.

Beispiel 5.2.1 Polio Impfung

Kommen wir zurück zum Impfstofffeldversuch von Salk. In einem naiven Ansatz wären wir versucht, eine sehr grosse Anzahl von Kindern mit dem von Salk entwickelten Impfstoff gegen Polio zu impfen. Wir werden dann die Auftretenshäufigkeit von Polio für dieses Jahr registrieren und diese Zahl mit der Auftretenshäufigkeit vom Vorjahr vergleichen. Dieser Ansatz wäre allerdings grundlegend falsch, denn Polio ist eine sich epidemisch verbreitende Krankheit, d.h., die Auftretenshäufigkeit von Polio kann sich von einem Jahr zum nächsten substantiell ändern. Was für einen Effekt des Impfstoffes wir auch immer feststellen werden, wir sind *nicht* in der Lage zu unterscheiden, ob es sich um den Effekt des jeweiligen Jahres oder um den Effekt des Impfstoffes handelt. In diesem Fall nennen wir die beiden Effekte *vermischt* (eng. confounded).

Die einzige Möglichkeit, die beiden Effekte voneinander unterscheiden zu können, besteht darin, eine Gruppe von Kindern dieses Jahr ungeimpft zu lassen und diese als *Kontrollgruppe* zu benutzen. Diese Vorgehensweise erlaubt es uns, die Wirksamkeit

des Impfstoffes zu messen, indem wir die Auftretenshäufigkeiten in der geimpften und in der Kontrollgruppe miteinander vergleichen.

Damit die Studie durchgeführt werden kann, ist die Einwilligung der Eltern notwendig. Nun könnten wir wiederum versucht sein, die mit dem Impfstoff behandelte Gruppe aus Kindern zusammenzusetzen, deren Eltern in die Studie eingewilligt haben. Die Kontrollgruppe bestünde dann aus den Kindern, deren Eltern nicht in die Studie eingewilligt haben. Es liese sich aber vermuten, dass Eltern mit höherem Einkommen eher in die Studie einwilligen als Eltern mit tiefem Einkommen. Hinzu kommt, dass Kinder von besserverdienenden Eltern aus Hygienegründen weniger resistent gegen Polio sein könnten. In diesem Fall wäre die Studie gegen den Impfstoff voreingenommen - der Hintergrund der Familie ist mit dem Effekt des Impfstoffes vermischt (*confounded*). Wir benötigen also eine Kontroll- und eine Behandlungsgruppe, die aus derselben Population stammt. Jedes Kind, dessen Eltern in die Studie eingewilligt haben, wird mit einer 50 % Wahrscheinlichkeit der Kontroll- oder Behandlungsgruppe zugeordnet. Diese Vorgehensweise nennt man *Randomisierung*.

Kindern in der Kontrollgruppe wird ein Placebo verabreicht, wobei ihnen nicht mitgeteilt wird, ob sie in der Kontroll- oder Behandlungsgruppe sind. Wir möchten dadurch sicherstellen, dass der Effekt durch die Wirksamkeit des Impfstoffes zu stande kam und nicht durch die Idee, behandelt zu werden. Des Weiteren sind auch die behandelnden Ärzte nicht informiert, ob einem Kind ein Placebo oder der Impfstoff verabreicht wird. Dies nennt man eine *Doppel-Blind-Studie*. Zusammenfassend sprechen wir von einer *randomisierten kontrollierten Doppelblind-Studie*.

□

Beispiel 5.2.2 Festigkeit von Beton

In einer Studie wollte man die Festigkeit von Beton in Abhängigkeit von unterschiedlichen Austrocknungsmethoden untersuchen. Deshalb werden Betonproben aus einem Produktionsdurchgang (Batch) genommen. Innerhalb dieses Batches sollten dieselben (homogenen) Bedingungen in Bezug auf Herstellung des Betons herrschen. Diese Betonproben werden zu kleinen Testzyllern geformt und dann in einer Klimakammer bis zu 28 Tagen ausgetrocknet. In der Klimakammer kann die Temperatur und die Feuchtigkeit kontrolliert werden. Anschließend wird die Festigkeit gemessen. Der Beton wird dann aufgrund dieser Festigkeitsmessungen klassifiziert (AST C 31 Standard Test Method for Making and Curing Concrete Test Specimens in the Field).

Wir wollen nun drei verschiedene Austrocknungsmethoden vergleichen. Es hat sich allerdings gezeigt, dass die Festigkeit von Beton von einem Batch zum anderen variieren kann. Deshalb werden aus zehn verschiedenen Batches je drei Proben genommen, zu Zylindern geformt und je ein Zylinder gemäß einer der drei Methoden

ausgetrocknet. Die Frage, ob sich die drei Methoden unterscheiden, scheint sich durch einen einfachen Versuchsplan beantworten zu lassen. Nun sind aber, wie oben dargelegt, Unterschiede in der Festigkeit zwischen Batches möglich und folglich können sich Festigkeitsmessungen innerhalb eines Batches systematisch von anderen Batches unterscheiden. Also ist es besser, den Einfluss der Variablen „Batch“ zu „kontrollieren“. In der Versuchsplanung stellt die Variable „Batch“ die *sekundäre Variable* dar, während die für die Beantwortung der eigentlichen Frage relevante Grösse die Austrocknungsmethode ist und als *primäre Variable* bezeichnet wird.

Durch die Erfassung der jeweiligen Batches verwenden wir in diesem Versuchsaufbau die sogenannte *Blockbildung*.



Primäre und sekundäre Variablen

Allgemein hängt die Zielgrösse von vielen Prädiktoren ab, von denen eine oder einige direkt mit einer Hauptfrage der geplanten Studie zusammenhängen und andere, die zwar nicht direkt mit der Hauptfrage der Studie zusammenhängen, die aber mit vernünftigem Aufwand ebenfalls feststellbar oder sogar kontrollierbar sind. Wir wollen die ersten *primäre* und die letzten *sekundäre Variablen*¹ nennen.

Weitere Variablen entziehen sich unserem Zugriff. Ihre Einflüsse auf die Zielgrösse bildet im Modell einen Teil des zufälligen Fehlers. Der „zufällige Fehler“ darf als Gesamtheit der Einflüsse aller nicht erfassten Prädiktoren aufgefasst werden.

In der Varianzanalyse (siehe nächstes Kapitel) werden oft die Ausdrücke *Prüf-Faktor* und *Stör-Faktor* in ähnlichem Sinne wie hier primäre und sekundäre Variable verwendet. Die primären und sekundären Variablen sind bei Experimenten meistens (in einem vernünftigen Bereich) beliebig wählbar:

- Man kann Temperatur und Feuchtigkeit in der Klimakammer oder in einem chemischen Reaktor einstellen. Interessiert man sich für den Zusammenhang zwischen Temperatur (Prädiktor) und Ertrag der chemischen Reaktion (Zielgrösse), so ist die Temperatur die primäre Variable und die Feuchtigkeit die sekundäre Variable.
- Man kann auf einem Feld eine bestimmte Weinsorte pflanzen und eine bestimmte Düngersorte und -menge ausbringen. Interessiert man sich für den Zusammenhang zwischen Düngersorte (Prädiktor) und Ertrag (Zielgrösse), so stellen Weinsorte und Düngermenge die sekundären Variablen dar.

¹ „Variablen“ können hier und im Folgenden als Prädiktoren im Sinne der Regression oder Faktoren der Varianzanalyse sein (siehe nächstes Kapitel).

Wenn mehrere primäre Variablen zu untersuchen sind, dann sollen sie möglichst „unabhängig“ voneinander sein. Genauer bedeutet das für kontinuierliche X -Variablen das Folgende: Obwohl die X-Variablen nicht als Zufallsvariablen im Modell erscheinen, kann man empirische Korrelationen zwischen ihnen ausrechnen. Man wählt, soweit möglich, die Versuchsbedingungen so, dass diese Korrelationen idealerweise null werden.

Handelt es sich bei den primären Variablen um Faktoren², so führt diese Vorgehensweise zu sogenannt *ausgewogenen Plänen* (eng. *balanced designs*): Im einfachsten Fall sollen für jede Kombination von Stufen (Levels) der einzelnen Faktoren die gleiche Anzahl Beobachtungen gemacht werden.

Wozu sollen sekundäre Variablen erfasst werden? Die sekundären Variablen sollten aus zwei Gründen erfasst werden: Sowohl die Genauigkeit als auch die Interpretierbarkeit der Resultate verbessern sich. Je mehr wir kontrollieren können umso (meist) besser.

- In einem Modell, das nur die primären Variablen enthält, ist die Streuung (die Varianz σ^2) der zufälligen Fehler grösser, da sie die Einflüsse der sekundären Variablen zusätzlich umfasst. Die meisten Testgrössen vergleichen eine „mittlere Quadratsumme“ mit einer Schätzung $\hat{\sigma}^2$ dieser Streuung, und die Länge der Vertrauensintervalle enthalten $\hat{\sigma}$ als Faktor. Es ist also plausibel, dass Tests mächtiger und Vertrauensintervalle kürzer werden, wenn sekundäre Variablen mit spürbarem Einfluss ins Modell aufgenommen werden und dadurch $\hat{\sigma}$ abnimmt. Bezieht man allerdings Variablen ein, die keinen Einfluss auf die Zielgröße haben, dann führt dies in geringem Masse zum gegenteiligen Effekt: Man verliert Freiheitsgrade und damit Genauigkeit bei der Schätzung von $\hat{\sigma}^2$.
- Der zweite Grund, sekundäre Variablen zu berücksichtigen, ist noch bedeutsamer: Wenn Variablen mit wesentlichem Einfluss auf die Zielgröße unberücksichtigt bleiben, ist die Interpretation der Resultate für die primären Variablen in Frage gestellt.

Blockbildung

Aus der vorangehenden Diskussion, wie die Streuung der Fehler reduziert werden kann, kamen wir zum Schluss, dass sekundäre Variablen wenn immer möglich im Modell berücksichtigt werden sollten. Eine spezielle Form der Erfassung von sekundären Variablen ist die *Blockbildung*. Ein Block zeichnet sich dadurch aus, dass innerhalb eines Blockes *homogene* Bedingungen herrschen.

²Ein *Faktor* ist eine Variable, die nur endlich viele Werte annehmen kann. Z.B. 1, 2, 3 oder „schlecht“, „mittel“, „gut“

Beispiel 5.2.3

Im Beispiel der Festigkeit von Beton wurden Batches als „Blöcke“ betrachtet. Der Blockeffekt, den wir in unserem Modell mitberücksichtigen (siehe Beispiel 5.2.2) „erklärt“ die Unterschiede in den Batches - wie auch immer diese Unterschiede entstehen mögen. Er steht also für viele mögliche „sekundäre“ und eventuell nicht einmal erfassbare Variablen.



In vielen Anwendungen gibt es Untersuchungseinheiten, die zur Blockbildung geeignet sind: Herkunft von Rohmaterialien, Altersgruppen von Patienten, Schulklassen, Ställe, Äcker, Würfe von Versuchstieren, Produktions-Los, usw. .

Wenn eine Variable konstant gehalten, beispielsweise die Temperatur auf 25 °C geregelt wird, dann stellt sich jedoch die Frage, ob die Schlüsse, die sich aus der Studie ergeben, für andere Bedingungen auch gelten. Einen Ausweg aus diesem Dilemma bietet wieder die Blockbildung. Möglichst kleine Streuung im Block führt zu möglichst präzisen Aussagen, möglichst grosse Unterschiede zwischen Blöcken lassen die Verallgemeinerung der Aussagen zu.

Die allgemeine Regel besagt: „Block what you can; randomize what you cannot.“

Replikate

Es ist nützlich, mehrere Messungen oder Beobachtungen für die gleichen Versuchsbedingungen durchzuführen. Man nennt solche *Wiederholungen* auch *Replikate*. Sie führen, wie jede Erhöhung der Anzahl Beobachtungen, zu einer grösseren Genauigkeit der Aussagen. Zudem ermöglichen sie eine genauere Prüfung des Modells. Man kann aus den Replikaten eine Schätzung der Streuung der zufälligen Fehler gewinnen. Sie dient in der Regression zur Beurteilung der Vollständigkeit des Modells. In der Varianzanalyse (siehe nächstes Kapitel) erlaubt sie es, das Vorhandensein von Wechselwirkungen zu testen (mehr dazu später).

Wenn man es sich leisten kann, die Beobachtungszahl zu erhöhen, werden allerdings andere Gesichtspunkte oft dazu führen, dass Beobachtungen für weitere Versuchsbedingungen statt für Wiederholungen geplant werden. Die Versuchung ist gross, noch einen weiteren Faktor in die Studie einzubeziehen. Viele Studien scheitern an einer zu umfangreichen Fragestellung bei zu kleiner Beobachtungszahl. Mangels Genauigkeit kann dann gar keine Einzelfrage schlüssig beantwortet werden.

Wenn von Wiederholungen die Rede ist, muss sogleich eine Warnung angefügt werden: Wenn die Versuchsbedingungen einmal eingestellt werden und dann die Zielgrösse mehrmals gemessen wird, muss man davon ausgehen, dass die entsprechenden Zufallsfehler nicht unabhängig sind, unter anderem wegen dem oben besprochenen Einfluss der zeitlichen Reihenfolge. Solche *Schein-Wiederholungen* für Tests und Vertrauensintervalle wie echte Wiederholungen zu behandeln, ist unzulässig.

Schein-Replikate sind allerdings besser als nichts. Zur Analyse wird man ihren Mittelwert berechnen und dadurch für jede Versuchsbedingung einen Wert der Zielgrösse erhalten, der genauer ist als die Einzelwerte.

Garantiert „echte“, unabhängige Replikate erhält man durch zufällige Auswahl der Reihenfolge wie oben; irgendwo in dieser Abfolge erscheint jede Versuchsbedingung zum ersten, zum zweiten Mal und allenfalls noch häufiger.

Um interpretierbare Tests zu erhalten, muss man sich gut überlegen, gegen welche Zufallsstreuung sich ein zu prüfender Effekt abheben muss, was also „der richtige Fehlerterm“ ist.

Beispiel 5.2.4

Als Beispiel sei ein Versuch erwähnt, in dem verschiedene Sorten von Spinat auf Unterschiede im Geschmack getestet werden sollten - eine Frage der Sensorik. Um die Variabilität richtig zu erfassen, wurde jede Sorte Spinat an mehreren Orten gekauft. Dann wurde jede Sorte in einem Topf gekocht und schliesslich von mehreren Prüfern probiert. Die Varianzanalyse (mit den Faktoren Sorte und Prüfer) ergab signifikante Unterschiede. Aber was für Unterschiede?

Da jede Sorte nur in einem Topf gekocht wurde, hat man nur gezeigt, dass sich die Streuung zwischen Töpfen von der Reststreuung abhebt. Diese Streuung zwischen Töpfen kann von den Spinatsorten, aber auch von der Streuung zwischen Einkäufen der gleichen Sorte, von ungleichen Kochbedingungen und Topfeigenschaften abhängen. Man kann also mit diesen Versuchen einen Unterschied zwischen Sorten gar nicht schlüssig nachweisen. Allgemein führen Mischproben dazu, dass eine oder mehrere Streuungs-Komponenten für die Analyse verloren geht.

□

5.2.1. Grundprinzipien

Zusammenfassend seien nochmals die Grundprinzipien aufgezählt, die bei der Versuchsplanung allgemein beachtet werden sollen:

- Die *Haupt-Fragestellung* legt die Zielgröße sowie die primären Variablen und deren relevanten Wertebereich fest.
- Es sollen möglichst viele weitere Prädiktoren erfasst werden, die einen Einfluss auf die Zielgröße haben könnten.

Diese sekundären Variablen sollen, wenn möglich, konstant gehalten oder zur *Blockbildung* verwendet werden. Ist dies nicht möglich, so sollen sie trotzdem im Modell berücksichtigt werden.

- Alle Prädiktoren sollen möglichst unkorreliert sein und die Faktoren ausgewogen (d.h. für jede Stufe eines Faktors sollten gleich viele Messungen durchgeführt werden).
- Die Zuordnung von Versuchsbedingungen zu Untersuchungseinheiten, insbesondere die zeitliche Reihenfolge der Einzelversuche, soll durch Zufallszahlen bestimmt werden (*randomisierte Zuordnung*) - soweit die vorhergehende Versuchsplanung sie nicht schon festlegt.
- *Replikate* sind wichtig, denn wenn sie unabhängig gewonnen werden, ermöglichen sie eine zusätzliche Überprüfung des Modells.

5.3. Der Versuchsplan

Der *Versuchsplan* oder das *design of experiment* besteht aus einer Liste von Versuchsbedingungen, die festlegen, wie jeder Faktor (siehe Beispiel 5.3.1) auf welchen Stufen zu variieren ist.

Vollständig randomisierter Versuchsplan

Der einfachste Plan ist der *vollständig randomisierte Versuchsplan*, der auf eine Faktor-Variable (= primärer Faktor) mit mehreren Stufen angewendet wird. Pro Stufe werden eine oder mehrere Messungen gemacht, aber immer gleich viele.

Der Plan wird in zufälliger Reihenfolge abgearbeitet, damit sich keine systematischen Effekte einschleichen können.

Beispiel 5.3.1 Verpackungsmethoden von gelagertem Fleisch

In einer Studie soll der Effekt der Verpackungsart auf das Bakterienwachstum von gelagertem Fleisch untersucht werden. Es wurden vier Verpackungsarten (Faktoren „Behandlungsarten“) untersucht:

- Kommerzielle Plastikverpackung (mit Umgebungsluft)

- Vakuumverpackung
- 1 % CO₂, 40 % O₂, 59 % N
- 100 % CO₂

Die Versuchseinheiten bestehen aus 12 Rindssteaks von rund 75 g. Wir interessieren uns also für die Wirksamkeit einer Verpackungsart, das Bakterienwachstum zu unterdrücken. Die gemessene Zielgröße ist gegeben durch den Logarithmus³ der Anzahl Bakterien pro Quadratzentimeter, wobei die Anzahl Bakterien neun Tage nach der Verpackung und Lagerung bei 4 °C gemessen wurde.

In der Studie wurden drei Rindssteaks zufällig einer Verpackungsmethode zugeordnet. Da die Versuchseinheiten zufällig den Versuchsbedingungen zugeordnet wurden und für jede Stufe der Faktorvariablen „Verpackungsmethode“ gleich viele Messungen erhoben wurden, handelt es sich um einen *vollständig randomisierten Versuchsplan*.

□

Block-Design

Hat neben dem primären Faktor auch ein sekundärer Faktor (Störfaktor) einen Einfluss auf die Zielgröße, so wendet man ein *Block-Design* an. Kommt darin jede Stufe des ersten Faktors (z.B. Behandlung) in jedem Block mindestens einmal zur Anwendung, spricht man von einem *Versuchsplan mit vollständigen Blöcken*. Die Zuteilung der Behandlungen erfolgt in jedem Block separat.

Beispiel 5.3.2 Festigkeit von Beton

Im Beispiel der Festigkeit von Beton in Abhängigkeit von drei unterschiedlichen Austrocknungsmethoden wurden die Batches als Blöcke in der Analyse berücksichtigt. Damit kann einem allfälligen Effekt der Batches Rechnung getragen werden. Aus jedem Batch wurden drei Testzylinder entnommen und jeweils eine Austrocknungsmethode darauf angewendet.

□

³Da Bakterienwachstum exponentiell ist, wird oft der Logarithmus des Wachstums genommen, das dann linear ist. Ein Vorteil ist, dass dann die Größenordnung der Bakterienpopulation gemessen wird.

Vollständiger faktorieller Versuchsplan

Wenn k Faktoren zu untersuchen sind, die auch mehr als zwei Stufen haben können, bildet der *vollständige faktorielle Versuchsplan*, der alle Kombinationen der Versuchsbedingungen enthält, eine naheliegende Lösung. Schon bei drei Faktoren mit je 5 Stufen benötigt man jedoch 125 Beobachtungen, und das ist oft mehr, als man sich leisten kann.

2^k -faktorieller Versuchsplan

In Screening Experimenten werden viele potentielle Faktoren auf ihren Einfluss auf die Zielvariable abgesucht. Damit der Aufwand möglichst gering gehalten werden kann, werden jeweils nur zwei Stufen (z.B. „hoch“ gegen „tief“) pro Faktor untersucht. Den entsprechenden Versuchsplan, in welchem alle möglichen Kombinationen aufgeführt sind, nennt man *2^k -faktorieller Versuchsplan* (oder kurz ein 2^k -Plan).

Fraktioneller 2^k -Versuchsplan

Um Ressourcen noch stärker einzusparen, wird oft nur ein ausgewogener Teil des vollständigen 2^k -Plans realisiert, was zu einem *fraktionellen 2^k -faktoriellen Versuchsplan* führt.

Fraktioneller Versuchsplan

Unter der Annahme, dass es keine (oder zu vernachlässigende) Wechselwirkungen gibt, kann man auch mit weniger Kombinationen noch Haupteffekte der Faktoren schätzen und prüfen. Solche Versuchspläne werden *unvollständig* genannt.

Auch sie sollten ausgewogen (d.h. gleich viele Messungen pro Stufe) sein. Diese Forderung führt zu sogenannten *fraktionellen Plänen* (eng. fractional designs).

5.4. Eine Checkliste zur Planung und Durchführung einer Studie

In den meisten wissenschaftlichen Studien, die mit empirischen Daten zu tun haben, treten ähnliche Arbeitsabläufe und oft auch ähnliche Schwierigkeiten auf. Die folgende Aufzählung enthält viel Selbstverständliches. Die Erfahrung zeigt, dass es sich dennoch lohnt, die Punkte bewusst durchzugehen.

5.4.1. Problemstellung

Das Wesentliche an jeder Studie ist natürlich die wissenschaftliche Fragestellung. Das Ideal einer möglichst bedeutsamen, wichtigen Frage steht oft der Realität gegenüber, dass eine solche nicht direkt und umfassend mit einer empirischen Studie untersucht werden kann. Kompromisse sind nötig.

Es sollte aber nicht passieren, dass Versuche und Beobachtungen durchgeführt werden, ohne dass *präzise Fragen und Hypothesen* vorliegen. Das notwendige Studium der Fachliteratur lohnt sich. Es erspart viel ineffiziente praktische Arbeit.

5.4.2. Festlegen der Zielvariablen und Identifikation von erklärenden Variablen

Aufgrund der Fragestellung wird mehr oder weniger klar festgelegt, welches die Zielgröße ist.

Schwieriger kann es sein, für *stetige* Variablen geeignete Faktorstufen zu wählen. Einseitig sollten sie so festgelegt werden, dass sie zu sichtbaren Effekten in der Zielvariablen führen sollen, sofern die Variable wirklich einflussreich ist. Andererseits dürfen die Stufen nicht so weit auseinander liegen, dass ein potentielles Optimum dazwischen liegt.

5.4.3. Vorversuche

Eigene Erfahrung mit dem Beobachtungsgegenstand und mit Mess- und Beobachtungsstrategien sind wichtig. Auch kann hier praktisch nachgeprüft werden, ob die Zielgröße geeignet gewählt ist, auch bezüglich der Messbarkeit.

5.4.4. Planung

Die Fragestellung muss im Zusammenhang mit der Art der beobachtbaren Daten so präzisiert werden, dass mögliche statistische Modelle (z. B. Varianzanalyse- und Regressionsmodelle) formuliert werden können. Wie müssten die Daten aussehen, damit die zu untersuchende Hypothese als widerlegt oder nicht widerlegt („bestätigt“) gelten kann? Es lohnt sich, diese Frage konkret durchzuspielen.

Jetzt kommen die Gesichtspunkte der vorhergehenden Abschnitte zum Zug. Im Zusammenhang mit dem Aufwand wird die Anzahl der Versuche oder Beobachtungen und ein Versuchsplan festgelegt. Als ein Ergebnis der Planung sollte die Auswertung der Daten in Bezug auf die hauptsächlichen Fragen klar sein. Natürlich muss auch die eigentliche Datengewinnung gut geplant werden.

5.4.5. Stichprobenumfang

Bei der Planung eines Versuchs stellt sich zwangsläufig die Frage nach dem Stichprobenumfang. Damit ist auch die Frage nach der *Macht* des statistischen Tests verknüpft. Die Macht eines statistischen Tests ist definiert als die Wahrscheinlichkeit, die Nullhypothese zu verwerfen, falls die Alternativhypothese zutrifft (meist das, was wir wollen). In der Regel versuchen wir im Rahmen eines Experimentes ja gerade die Nullhypothese zu verwerfen, denn wir sind der Überzeugung, dass die Alternativhypothese zutrifft. Die Macht sagt uns dann, wie gross die Wahrscheinlichkeit ist, ein signifikantes Resultat zu erhalten, falls die Alternativhypothese zutrifft.

Die Berechnung der Macht ist einem Gedankenexperiment ähnlich: Wir benötigen keine Daten, aber eine genaue Angabe der Parameter unter der Alternativhypothese, von deren Richtigkeit wir überzeugt sind. Falls die Macht des Tests klein ist, ist die Wahrscheinlichkeit gross, dass wir ein nicht-signifikantes Resultat erhalten, obwohl die Alternativhypothese richtig wäre. In diesem Fall verschwenden wir Zeit und Geld mit unserem Experiment, da im Vorhinein klar ist, dass das Resultat mit grosser Wahrscheinlichkeit nicht signifikant ist. Falls die Macht zum Beispiel 0.2 beträgt, werden wir bloss in einem von fünf Fällen ein signifikantes Resultat erhalten. Idealerweise sollte die Macht grösser als 80 % sein.

Die Macht eines statistischen Tests hängt von folgenden Größen ab:

1. Versuchsplan (ausgewogen, unausgewogen, etc.)
2. Signifikanzniveau α (typischerweise 5 %)
3. Parameterwerte unter der Alternativhypothese (unter anderem Fehlervarianz σ^2)
4. Stichprobenumfang n

Es gilt die Faustregel, dass die Macht desto grösser ist, je mehr Beobachtungen (Stichprobenumfang) wir haben. Typischerweise wählen wir n so, dass die Macht eine „angebrachte“ Größenordnung erreicht. Ist n kleiner, so entspricht dies nicht unseren Anforderungen; ist n grösser, so verschwenden wir unsere Ressourcen.

In einfacheren Fällen können wir für die Macht eine Formel angeben, z.B. im Falle eines t -Tests für zwei Stichproben. In diesen Fällen existieren auch **Python**-Funktionen. In komplexeren Fällen berechnen wir die Macht mit Hilfe von *Simulationen* (siehe Beispiel 6.3.6).

Der Stichprobenumfang wird in der Praxis unglücklicherweise oft durch die finanziellen Ressourcen bestimmt. Die Machtanalyse gibt dann Auskunft darüber, ob es sinnvoll ist, die Studie durchzuführen oder ob es wohl eher eine Geldverschwendug ist.

Der schwierigste Teil besteht darin, die Parameter unter der Alternativhypothese festzulegen. Wenn diese aus der Literatur oder früheren Versuchen vorhanden ist, kann der nötige Stichprobenumfang ausgerechnet werden. Vorversuche sind meistens nur dazu geeignet, eine Größenordnung zufälliger Streuungen abzuschätzen.

5.4.6. Daten

Wie die Versuche oder Beobachtungs-Kampagnen durchgeführt werden, ist natürlich von Fall zu Fall verschieden. Mit Vorversuchen und Instruktionen soll sichergestellt werden, dass die Qualität der Daten von Anfang an hoch ist.

Die Mess- oder Beobachtungsmethoden sollten nur im Notfall und nur in genau dokumentierter Weise im Laufe der Hauptphase geändert werden.

Es ist wichtig, dass ein *Journal* geführt wird, in dem der Ablauf der Daten-Gewinnung dokumentiert und Besonderheiten und Unvorhergesehenes notiert werden, damit später unerwartete Abweichungen in den Daten eventuell erklärt werden können.

Gleichzeitig soll mit der Datenerfassung, -kontrolle und -speicherung auf dem Computer begonnen werden. Es ist auch sehr nützlich, wenn spätestens in dieser Phase Zeit eingeplant wird, um die statistischen Methoden und das Programmsystem genauer kennenzulernen.

5.4.7. Daten-Bereinigung

Die Daten auf Plausibilität hin zu prüfen und mit graphischen Darstellungen zu veranschaulichen, erspart viel Mühe bei der Auswertung. Diese vertiefte Datenkontrolle kann parallel zur Datenerfassung begonnen, aber nicht abgeschlossen werden. Schwierig wird es, wenn sich dabei noch Unzulänglichkeiten in der Versuchsanlage und der Datenerfassung zeigen: Man muss sorgfältig abwägen, ob die Verbesserung der Qualität wertvoller ist als die Homogenität der Daten und die genaue Einhaltung des Versuchsplanes.

5.4.8. Auswertung, Interpretation

Die Methodik, mit der die Daten im Hinblick auf die Haupt-Fragestellung ausgewertet werden, sollte zwar bereits klar sein. Dennoch beanspruchen die sorgfältige Überprüfung von Voraussetzungen und notwendige Modellanpassungen einige Zeit. Schliesslich tauchen oft im Laufe der Analysen neue Fragen und Hypothesen auf, die im Sinne der explorativen Analyse genauer untersucht werden können.

Die Interpretation der Ergebnisse im fachlichen Zusammenhang kann eindeutig sein - besonders, wenn die Studie sorgfältig geplant wurde (siehe oben). Es können aber auch mehrere Interpretationen und mehrere Modelle mit den Daten verträglich sein. Die Schlüsse müssen klar formuliert und, soweit möglich, deutlich mit den einzelnen Ergebnissen der Datenanalyse begründet werden, denn Aussenstehende wissen weniger über die spezielle Fragestellung und wollen weniger gedankliche Arbeit hineinstecken.

5.4.9. Bericht

Damit sind wir beim Berichte-Schreiben angelangt, das viele als den mühsamsten Teil einer Studie empfinden. Es ist aber klar, dass die Nützlichkeit der Forschung durch die Lesbarkeit der Berichte und die Verständlichkeit von Vorträgen begrenzt ist.

Es gibt nützliche Anleitungen zur Abfassung von Berichten. Zwei Punkte seien hier erwähnt:

- Für *wen* ist der Bericht hauptsächlich gedacht? Der Hauptteil sollte für das *vorgesehene* Publikum ohne Mühe verständlich sein.

Oft müssen zwecks Dokumentation zusätzliche Details festgehalten werden, die in Anhänge oder in einem speziellen Bericht versorgt werden können.

- Die Leserinnen und Leser sind sich an gewisse Formen gewöhnt. In vielen wissenschaftlichen Zeitschriften gibt es beispielsweise in jedem Artikel die Abschnitte „Einleitung“, „Material und Methoden“, „Ergebnisse“ und „Diskussion“.

Leider führt das gerade dann zu Schwierigkeiten, wenn unübliche statistische Methoden verwendet werden. Sie müssen gemäss diesen Schemen unter „Material und Methoden“ beschrieben werden, getrennt von den „Ergebnissen“, und das erschwert die Verständlichkeit.

Abgesehen von solchen Nachteilen erleichtert eine einheitliche Form die Kommunikation im Allgemeinen.

5.4.10. Beratung

Sie wissen einiges über Regression und die einfachsten Fälle der Varianzanalyse (siehe nächstes Kapitel), wenn Sie diesen Kurs verarbeitet haben. Vielfach lohnt es sich dennoch, einen statistischen *Beratungsdienst* beizuziehen, besonders bei der Planung und während der Auswertung und Interpretation.

Bei Studien mit anspruchsvoller statistischer Methodik sollte jemand mit guter statistischer Ausbildung dauernd mitarbeiten und den Bericht mitverfassen.

5.4.11. Abschluss

Es lohnt sich, die gemachten Erfahrungen mit allen Beteiligten zu sammeln und zu besprechen.

Kapitel 6.

Varianz-Analyse

6.1. Vorbemerkungen

In der Regressions- und Varianzanalyse werden Situationen behandelt, in denen wir uns für eine einzige Zielgröße Y interessieren, die direkt mess- oder beobachtbar ist. In der Regel wird diese Zielgröße durch verschiedene Größen beeinflusst. In der Regressionsanalyse haben wir die Zielvariable Y als Funktion (Regressionfunktion) von Prädiktoren aufgefasst, die mit „zufälligen Abweichungen“ (Fehlertermen) überlagert ist. Dieser Ansatz erlaubt es uns, den Einfluss der Prädiktoren unter Berücksichtigung der anderen Einflüsse herauszuarbeiten. Häufig werden zur Auswertung von Experimenten Modelle eingesetzt, in denen die Zielvariable Y als Funktion von nominalen Prädiktoren, sogenannten *Faktoren*, ausgedrückt wird.

Die Varianzanalyse kann als Spezialfall der Regressionsrechnung angesehen werden, in welchem die Prädiktoren als Faktorvariablen auftreten. Es lohnt sich aber, diesen Spezialfall gesondert zubetrachten, weil damit die Eigenheit, dass *nur* Faktorvariablen als Prädiktoren zum Einsatz kommen, speziell ausgenutzt werden kann. Als Oberbegriff über Varianzanalyse und Regression wird von *linearen Modellen* gesprochen.

Ursprünglich wurde die Varianzanalyse als ein statistisches Verfahren verstanden, das die Herkunft der Variabilität in den Daten analysiert. Betrachtet man einen einzigen Faktor (eine Variable) mit mehreren Stufen, so kann jede Stufe dieses Faktors als eine Gruppe aufgefasst werden. Die Varianzanalyse führt dann zu einer Beurteilung der Stärke der Variabilität der Zielgröße innerhalb der Gruppen, die unterschiedliche Behandlungen erfahren haben. Wir werden uns allerdings mehr auf die regressionsartige Modellierung von Daten aus Experimenten fokussieren. Insbesondere interessiert uns, inwiefern sich die unterschiedlichen Gruppen in der Zielgröße unterscheiden.

In den folgenden Abschnitten wird ein erster Einblick in die Varianzanalyse und die darin verwendeten Modelle gewährt.

6.2. Mehrere Gruppen, einfache Varianzanalyse

Der Vergleich von zwei Gruppen miteinander ist eine grundlegende Technik in der Statistik und wird in jedem Einführungskurs behandelt. Aber oft werden in einem Experiment mehr als zwei Gruppen untersucht, wie das im folgenden Beispiel der Reissfestigkeit von Papier der Fall ist.

Beispiel 6.2.1 Reissfestigkeit von Papier

Ein Papierhersteller, der Einkaufs-Papiertragtaschen herstellt, interessiert sich für die Verbesserung der Reissfestigkeit seines Produkts. Man vermutet, welches die Reissfestigkeit von der Hartholz-Konzentration im Papierbrei abhängt. Üblicherweise liegen diese Konzentrationen in einem Bereich von 5 % bis 20 %.

Die Produktionsingenieure beschlossen, die Reissfestigkeit bei vier Hartholzkonzentrationsstufen mit einem vollständig randomisierten Versuchsplan zu untersuchen: bei 5 %, 10 %, 15 % und 20 %. Für jede Konzentrationsstufe werden sechs Versuchssproben in einer Pilotanlage erstellt. Die resultierenden 24 Papierproben werden in zufälliger Reihenfolge im Labor auf ihre Reissfestigkeit getestet. Die gemessenen Reissfestigkeiten (in psi) sind in Tabelle 6.1 festgehalten (Quelle: Montgomery and Runger).

Hartholz-Konzentration [%]	Versuchsprobe					
	1	2	3	4	5	6
5	7	8	15	11	9	10
10	12	17	13	18	19	15
15	14	18	19	17	16	18
20	19	25	22	23	18	20

Table 6.1.: Reissfestigkeit von Papier in Abhängigkeit der verwendeten Hartholzkonzentration im Papierbrei.

Bei der Durchführung von Messungen ist darauf zu achten, dass das jeweilige Messumfeld so homogen wie möglich gehalten wird (möglichst gleiche Versuchsbedingungen), wie im letzten Kapitel bereits ausgeführt wurde. Falls sich wichtige Größen ändern können, müssen sie miterfasst werden. Weil man nie sicher ist, ob das gelingt, werden die Messungen üblicherweise in zufälliger Reihenfolge durchgeführt. So wurden auch in diesem Beispiel die Laborproben zufällig aus den 24 Proben gewählt, ohne Rücksicht auf die Hartholzkonzentration oder Fertigstellung der Probe.

□

Kapitel 6. Varianz-Analyse

Die Frage nach den Einflüssen der unterschiedlichen Behandlungen kann man zunächst untersuchen, indem man jede Gruppe durch einen *Zwei-Stichproben-Test* (d.h. z. B. durch den Rangsummen-Test von Wilcoxon, den *t*-Test von Student oder den Vorzeichen-Test) mit jeder anderen vergleicht. Die Resultate für einen bestimmten Test kann man in einer symmetrischen Matrix von *P*-Werten zusammenfassen.

Beispiel 6.2.2

In Tabelle 6.2 sind die P-Werte für den Zwei-Stichproben *t*-Test für die Reissfestigkeit von Papier aufgeführt.

Hartholz-Konzentration [%]	5 %	10 %	15 %	20 %
5 %	—			
10 %	0.0010	—		
15 %	0.00076	0.38	—	
20 %	0.00	0.0013	0.010	—

Table 6.2.: *P*-Werte für paarweise Vergleiche von je zwei Gruppen mit *t*-Test

Vergleichen wir z.B. die Werte für 10 % und 20 %, so erhalten wir einen *P*-Wert von 0.006. ([zu R](#))

```
from pandas import DataFrame
import pandas as pd
import numpy as np
import seaborn as sns
import scipy.stats as st

rf = DataFrame({
    "HC": np.repeat(["5%", "10%", "15%", "20%"], [6, 6, 6, 6]),
    "Strength": [7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18]
})

rf
print(rf)

per5 = rf.loc[rf["HC"]=="5%", "Strength"]
per10 = rf.loc[rf["HC"]=="10%", "Strength"]
per15 = rf.loc[rf["HC"]=="15%", "Strength"]
per20 = rf.loc[rf["HC"]=="20%", "Strength"]

st.ttest_ind(per10, per20)
print(st.ttest_ind(per10, per20))
```

Kapitel 6. Varianz-Analyse

```
##      HC Strength
## 0    5%      7
## 1    5%      8
## 2    5%     15
## 3    5%     11
## 4    5%      9
## 5    5%     10
## 6   10%     12
## 7   10%     17
## 8   10%     13
## 9   10%     18
## 10  10%     19
## 11  10%     15
## 12  15%     14
## 13  15%     18
## 14  15%     19
## 15  15%     17
## 16  15%     16
## 17  15%     18
## 18  20%     19
## 19  20%     25
## 20  20%     22
## 21  20%     23
## 22  20%     18
## 23  20%     20
## Ttest_indResult(statistic=-3.4979930040209894, pvalue=0.00574574017074254)
```

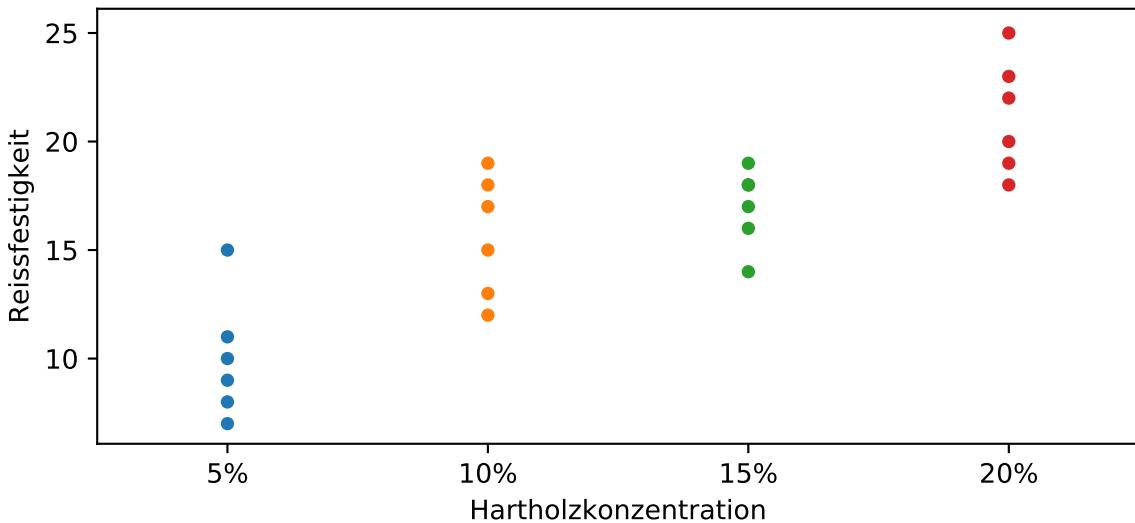
Hier zeigt sich, dass beispielsweise der Unterschied zwischen 10 % und 20 % Hartholz-Konzentration mit einem P -Wert von 0.006 signifikant ist, aber dass der Unterschied zwischen 10 % und 15 % Hartholz-Konzentration mit einem P -Wert von 0.35 *nicht* signifikant ist.

Diese Unterschiede können graphisch mit Hilfe von *Stripcharts* (siehe Abbildung 6.1) dargestellt und überprüft werden. ([zu R](#))

Eine weitere Darstellungsart sind Boxplots (siehe Abbildung 6.2). ([zu R](#))

Die beiden Arten der Auswertung führen in diesem Fall zum gleichen Ergebnis, nämlich dass sich die Reissfestigkeit bei Hartholz-Konzentrationen von 10 % und 15 % nicht signifikant unterscheidet. Im Allgemeinen müssen sich die beiden Verfahren aber nicht einig sein.





```

sns.stripplot(x="HC", y="Strength", data=rf)

plt.xlabel("Hartholzkonzentration")
plt.ylabel("Reissfestigkeit")

plt.show()

```

Abbildung 6.1.: Stripchart für den Beispieldatensatz **Reissfestigkeit** von Papier.

Gegen eine solche Vielzahl von Paar-Vergleichen müssen allerdings von der Grundsiede des statistischen Hypothesentests her schwerwiegende Bedenken angemeldet werden. Wenn man beispielsweise 7 Gruppen miteinander vergleicht, so ergeben sich

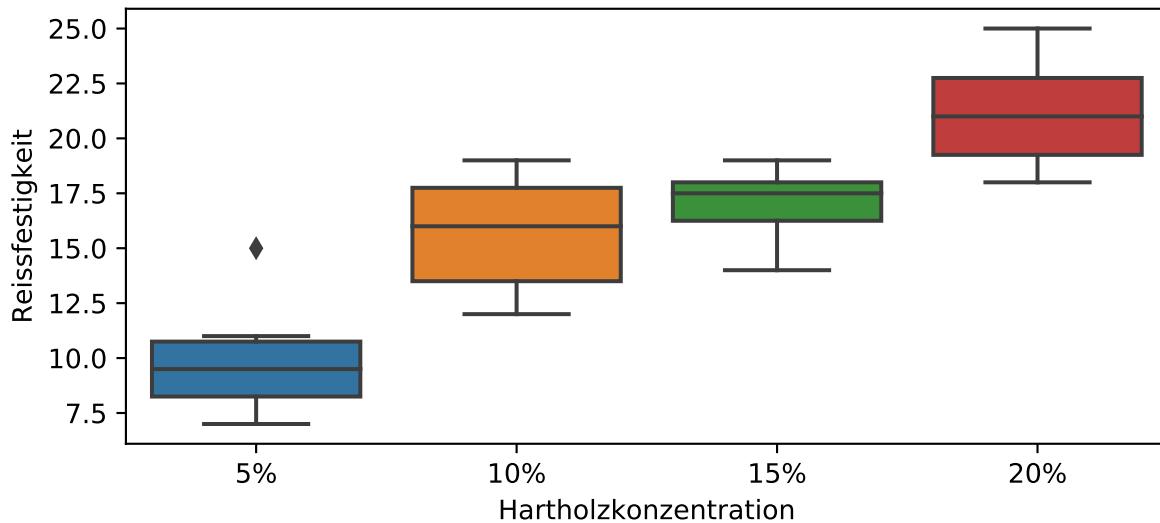
$$\frac{7 \cdot 6}{2} = 21$$

Paarvergleich-Tests.

Wir wollen eine einfache theoretische Überlegung machen, wie die Resultate dieser Tests aussehen können: Aufgrund der Irrtums-Wahrscheinlichkeit von 5 % ist es anschaulich klar, dass ab und zu unter 21 Tests eine „Fehlentscheidung 1. Art“, nämlich dass die Nullhypothese fälschlicherweise verworfen wird, auftritt. Bei 21 Tests ist die erwartete Anzahl Fehlentscheide 1. Art gegeben durch

$$21 \cdot 0.05 \sim 1$$

Wir müssen also damit rechnen, dass im Mittel 1 Hypothesentest fälschlicherweise verworfen wird.



```

sns.boxplot(x="HC", y="Strength", data=rf)

plt.xlabel("Hartholzkonzentration")
plt.ylabel("Reissfestigkeit")

plt.show()

```

Abbildung 6.2.: Boxplots für den Beispieldatensatz **Reissfestigkeit** von Papier

Die Nullhypothese, „alle Gruppen gehorchen dem gleichen Modell“, wird also viel zu oft verworfen, wenn die Regel lautet: „Die Nullhypothese wird verworfen, wenn der extremste Unterschied auf dem Niveau $\alpha = 5\%$ signifikant ist“. Wie kann man das vermeiden? Eine konsequente Antwort heisst: Wir dürfen nur *eine* Frage stellen, die wir mit einem Test beantworten. Die sinnvolle Frage lautet: „Gibt es überhaupt Unterschiede zwischen den Gruppen?“. Oder anders gesagt: „Unterscheidet sich wenigstens eine der Gruppen von einer andern?“ Die entsprechende Nullhypothese haben wir gerade vorher formuliert: „Alle Gruppen folgen dem gleichen Modell.“

6.2.1. Gruppenmittel-Modell

Die Situation, die im Beispiel 6.2.1 zum Datensatz **Reissfestigkeit** von Papier beschrieben ist, lässt sich durch ein lineares Modell (oder als Verallgemeinerung des Zwei-Stichproben-Modells) festhalten. Wir wollen g Gruppen vergleichen, wobei in jeder Gruppe gerade m Beobachtungen gemacht werden.

Ziel ist es, ein Modell zu entwickeln, dass die Reissfestigkeit in Abhängigkeit der Hartholzkonzentrationsstufen beschreibt.

Beispiel 6.2.3

Im Datensatz **Reissfestigkeit** wurden 4 unterschiedliche Hartholzkonzentrationen verwendet, also ist $g = 4$ und für jede Hartholzkonzentration wurden $m = 6$ Messungen für die Reissfestigkeit durchgeführt.

□

Allgemeines Modell

Das einfachste Modell in dieser Situation ist, dass die einzelnen Beobachtungen innerhalb einer Gruppe um einen gemeinsamen Wert streuen:

$$Y_{ij} = \mu_i + \varepsilon_{ij} \quad i = 1, 2, \dots, g; \quad j = 1, 2, \dots, m$$

wobei Y_{ij} die j -te Beobachtung in der i -ten Gruppe ist.

Die Größe μ_i ist der „Mittelwert“ der i -ten Gruppe. Wir gehen wieder davon aus, dass die Fehlerterme ε_{ij} unabhängig identisch normalverteilt sind. Alle Gruppen teilen also dieselbe Standardabweichung des Fehlerterms in diesem Modell.

Aus der Sichtweise der linearen Regression ist Y_{ij} die Zielgröße (die wir vorhersagen möchten), die Behandlungsart μ_i ist eine Faktorvariable. Folglich ist dies nichts anderes als ein Regressionsmodell mit einer Faktorvariable als Prädiktor.

Eine äquivalente Modellformulierung lautet:

$$Y_{ij} = \mu + \tau_i + \varepsilon_{ij} \quad i = 1, 2, \dots, g; \quad j = 1, 2, \dots, m$$

mit dem Fehler

$$\varepsilon \sim \mathcal{N}(0, \sigma^2)$$

Den Parameter μ haben also alle Beobachtungen gemeinsam. Die Parameter τ_i ($i = 1, \dots, g$) hingegen sind die behandlungsspezifischen Abweichungen von diesem globalen Mittelwert (im Beispiel 6.2.1 also spezifisch für jede Hartholz-Konzentration). Man nennt diese Parameter auch *Behandlungseffekte* (eng. *treatment effects*).

Leider sind die Parameter in diesem Modell nicht mehr eindeutig identifizierbar, denn wir haben $g + 1$ Parameter $(\mu, \tau_1, \dots, \tau_g)$ für g unterschiedliche Gruppenmittelwerte. Wir benötigen also eine Nebenbedingung, wobei es deren mehrere gibt. Wir könnten zum Beispiel

$$\mu = \mu_1$$

und folglich

$$\tau_1 = 0, \quad \tau_2 = \mu_2 - \mu, \quad \tau_3 = \mu_3 - \mu$$

etc. setzen. Gruppe 1 bildet in diesem Fall also die Referenz, oder die sogenannte *Baseline*. Somit können nur $g - 1$ der Behandlungseffekte τ_i frei variieren. Die Behandlungseffekte τ_i haben somit $g - 1$ Freiheitsgrade.

Wir nennen diese Nebenbedingungen oder Parametrisierung der Faktoren *Kontraste*. Wir kommen später nochmal auf Kontraste zurück.

Parameterschätzung

Wie schätzen wir nun die Parameter

$$\mu, \quad \tau_1, \quad \dots, \quad \tau_g$$

so dass das Modell möglichst gut zu den Daten passt? Wir wählen nun eine Parametrisierung, bei der μ dem „globalen Mittelwert“ oder „grand mean“ entspricht. In diesem Fall sind die Behandlungseffekte τ_i eindeutig bestimmt.

Als Kriterium werden wir wiederum die Summe der quadrierten Residuen minimieren:

$$\sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \hat{\mu} - \hat{\tau}_i)^2$$

Es kann gezeigt werden, dass

$$\hat{\mu}_i = \frac{1}{m} \sum_{j=1}^m Y_{ij}$$

ist. Folglich kann μ_i durch den empirischen Gruppenmittelwert der i -ten Gruppe geschätzt werden. Der globale Mittelwert μ kann durch

$$\hat{\mu} = \frac{1}{n} \sum_{i,j} Y_{ij}$$

geschätzt werden, wobei

$$n = g \cdot m$$

ist.

Da die geschätzten Gruppenmittelwerte durch

$$\hat{\mu}_i = \hat{\mu} + \hat{\tau}_i$$

gegeben sind, ergeben sich die Schätzungen der Behandlungseffekte durch

$$\hat{\tau}_i = \hat{\mu}_i - \hat{\mu}$$

Kapitel 6. Varianz-Analyse

Wir bemerken allgemein, dass die Gruppenmittelwerte $\hat{\mu}_i$ für alle Parametrisierungen identisch sind, während die Behandlungseffekte $\hat{\tau}_i$ von der Parametrisierung abhängig sind.

Um die Streuung der Parameter τ_i zu schätzen, müssen wir die Streuung σ der Fehlerterme ε_{ij} schätzen. Wir werden uns damit im nächsten Unterkapitel 6.2.2 beschäftigen.

Haben wir die Streuung σ des Fehlerterms ε_{ij} geschätzt, so ist der Standardfehler von μ gegeben durch

$$\frac{\sigma}{\sqrt{m \cdot g}} \equiv \frac{\sigma}{\sqrt{n}}$$

wobei n die Gesamtzahl Messungen bezeichnet, und der Standardfehler von μ_i ist gegeben durch σ/\sqrt{m} . Somit lautet der Standardfehler von $\tau_i = \mu_i - \mu$

$$\sigma \sqrt{\frac{1}{m} - \frac{1}{n}}$$

Folglich ist das 95 %-Vertrauensintervall für τ_i gegeben durch

$$\hat{\tau}_i \pm t_{0.975; n-g} \cdot \hat{\sigma} \sqrt{\frac{1}{m} - \frac{1}{n}}$$

wobei $t_{0.975; n-g}$ das 97.5 % Quantil der t-Verteilung mit $n - g$ Freiheitsgraden bezeichnet und die Schätzung von σ gerade $n - g$ Freiheitsgrade hat, wie wir gleich sehen werden.

Beispiel 6.2.4 Reissfestigkeit von Papier

Nun wollen wir die Koeffizienten des Gruppenmittel-Modells 6.2.1 für den Datensatz **Reissfestigkeit** bestimmen. Mit der **Python**-Funktion **ols** wird das Gruppenmittel-Modell an die Daten angepasst. ([zu R](#))

```
from pandas import DataFrame
import pandas as pd
import numpy as np
import seaborn as sns
import scipy.stats as st

from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm
from statsmodels.graphics.factorplots import interaction_plot
from statsmodels.stats.outliers_influence import summary_table
import matplotlib.pyplot as plt
from scipy import stats

rf = DataFrame({
```

Kapitel 6. Varianz-Analyse

```
"HC": np.repeat(["5%","10%","15%","20%"], [6, 6, 6, 6]),
"Strength": [7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19, 25
})

fit = ols("Strength~HC", data=rf).fit()
fit.summary()

##                                     OLS Regression Results
## =====
## Dep. Variable:                 Strength    R-squared:           0.746
## Model:                          OLS         Adj. R-squared:      0.708
## Method:                         Least Squares   F-statistic:        19.61
## Date:             Wed, 20 Feb 2019   Prob (F-statistic): 3.59e-06
## Time:              00:11:14          Log-Likelihood:     -54.344
## No. Observations:                  24          AIC:                116.7
## Df Residuals:                      20          BIC:                121.4
## Df Model:                           3
## Covariance Type:            nonrobust
## =====
##            coef    std err       t      P>|t|      [ 0.025      0.975]
## -----
## Intercept      15.6667     1.041     15.042      0.000     13.494     17.839
## HC[T.15%]       1.3333     1.473      0.905      0.376     -1.739      4.406
## HC[T.20%]        5.5000     1.473      3.734      0.001      2.428      8.572
## HC[T.5%]       -5.6667     1.473     -3.847      0.001     -8.739     -2.594
## =====
## Omnibus:                   0.929  Durbin-Watson:           2.181
## Prob(Omnibus):               0.628  Jarque-Bera (JB):      0.861
## Skew:                      0.248  Prob(JB):                0.650
## Kurtosis:                   2.215  Cond. No.                 4.79
## =====
## 
## Warnings:
## [1] Standard Errors assume that the covariance matrix of the errors is correctly spe
```

oder kürzer

```
fit.params

## Intercept      15.666667
## HC[T.15%]       1.333333
## HC[T.20%]        5.500000
## HC[T.5%]       -5.666667
## dtype: float64
```

Beachten Sie, dass im Output der Parameter für HC10 % fehlt. Der ist aber, da er zuerst auftritt, gleich 0.

Kapitel 6. Varianz-Analyse

Mit dem **Python**-Befehl **ols** ist klar ersichtlich, dass der Parameter μ geschätzt ist durch $\hat{\mu} = 15.66$ und dass die Parametrisierung $\mu = \mu_{10\%}$ lautet. Die geschätzten Gruppenmittelwerte lauten somit:

$$\begin{aligned}\hat{\mu}_{5\%} &= 15.7 - 5.7 = 10 \\ \hat{\mu}_{10\%} &= 15.7 + 0 = 15.7 \\ \hat{\mu}_{15\%} &= 15.7 + 1.3 = 17 \\ \hat{\mu}_{20\%} &= 15.7 + 5.5 = 21.2\end{aligned}$$

Die 95 %-Vertrauensintervalle für die Gruppenmittelwerte μ_i erhalten wir mit **Python** wie folgt. ([zu R](#))

```
fit_pred.conf_int()

## [[ 7.8274691 12.1725309 ]
## [ 7.8274691 12.1725309 ]
## [ 7.8274691 12.1725309 ]
## [ 7.8274691 12.1725309 ]
## [ 7.8274691 12.1725309 ]
## [ 7.8274691 12.1725309 ]
## [13.49413576 17.83919757]
## [13.49413576 17.83919757]
## [13.49413576 17.83919757]
## [13.49413576 17.83919757]
## [13.49413576 17.83919757]
## [13.49413576 17.83919757]
## [13.49413576 17.83919757]
## [14.8274691 19.1725309 ]
## [14.8274691 19.1725309 ]
## [14.8274691 19.1725309 ]
## [14.8274691 19.1725309 ]
## [14.8274691 19.1725309 ]
## [18.99413576 23.33919757]
## [18.99413576 23.33919757]
## [18.99413576 23.33919757]
## [18.99413576 23.33919757]
## [18.99413576 23.33919757]]
```

Somit ist zum Beispiel das 95 %-Vertrauensintervall für $\mu_{5\%}$ gegeben durch

$$[7.8, 12.2]$$

□

Beispiel 6.2.5

Zuerst wollen wir den Datensatz **Meat** graphisch darstellen, siehe dazu Abbildung 6.3. (zu R)

```
meat = DataFrame({
    "Treatment": np.repeat(["Kommerziell", "Vakuum", "Gemischt", "CO2"], [3, 3, 3, 3]),
    "meat_id": [7.66, 6.98, 7.80, 5.26, 5.44, 5.80, 7.41, 7.33, 7.04, 3.51, 2.91, 3.6]
})

sns.stripplot(x="Treatment", y="meat_id", data=meat)
plt.xlabel("Verpackungsmethode")
plt.ylabel("Logarithmus Bakterienzahl")

plt.show()
```

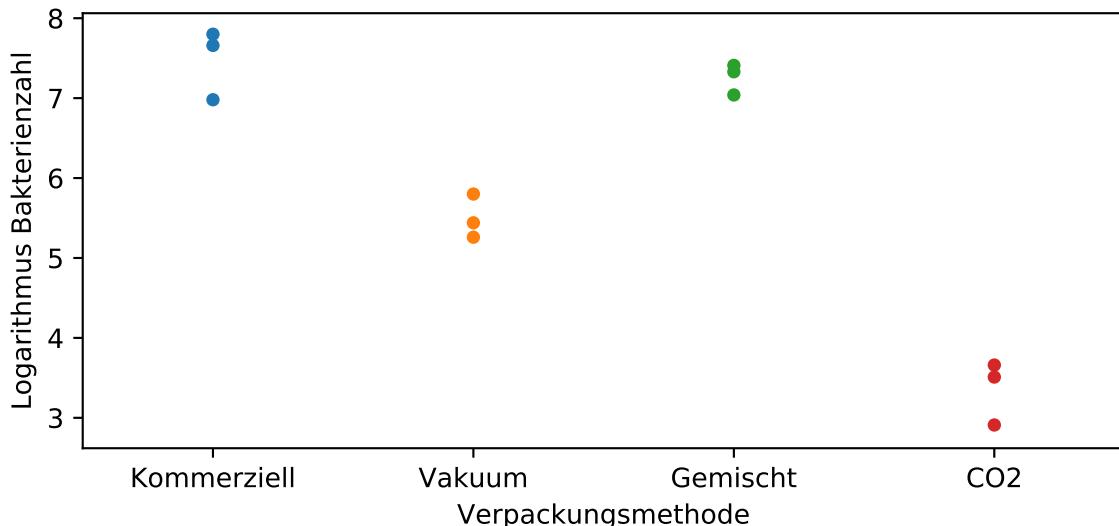


Abbildung 6.3.: Stripchart für den Datensatz **Meat** in Abhängigkeit der vier Verpackungsmethoden.

Nun wollen wir die Koeffizienten des Gruppenmittel-Modells 6.2.1 für den Datensatz **Meat** bestimmen. (zu R)

```
from pandas import DataFrame
import pandas as pd
import numpy as np
import seaborn as sns
import scipy.stats as st
from statsmodels.formula.api import ols
```

Kapitel 6. Varianz-Analyse

```
from statsmodels.stats.anova import anova_lm
from statsmodels.graphics.factorplots import interaction_plot
from statsmodels.stats.outliers_influence import summary_table
import matplotlib.pyplot as plt
from scipy import stats

meat = DataFrame({
    "Treatment": np.repeat(["Kommerziell", "Vakuum", "Gemischt", "CO2"], [3, 3, 3, 3]),
    "steak_id": [7.66, 6.98, 7.80, 5.26, 5.44, 5.80, 7.41, 7.33, 7.04, 3.51, 2.91, 3.3]
})

fit = ols("steak_id~Treatment", data=meat).fit()

fit.params

## Intercept           3.36
## Treatment[T.Gemischt] 3.90
## Treatment[T.Kommerziell] 4.12
## Treatment[T.Vakuum] 2.14
## dtype: float64
```

Somit lauten die geschätzten Gruppenmittelwerte

$$\begin{aligned}\hat{\mu}_{\text{CO}_2} &= 3.36 - 0 = 3.36 \\ \hat{\mu}_{\text{Kommerziell}} &= 3.36 + 4.12 = 7.48 \\ \hat{\mu}_{\text{Gemischt}} &= 3.36 + 3.90 = 7.26 \\ \hat{\mu}_{\text{Vakuum}} &= 3.36 + 2.14 = 5.50\end{aligned}$$

Die 95 %-Vertrauensintervalle für die Gruppenmittelwerte μ_i erhalten wir mit **Python** wie folgt: ([zu R](#))

```
fit_pred = fit.get_prediction()

fit_pred.conf_int()

## [[7.02684427 7.93315573]
## [7.02684427 7.93315573]
## [7.02684427 7.93315573]
## [5.04684427 5.95315573]
## [5.04684427 5.95315573]
## [5.04684427 5.95315573]
## [6.80684427 7.71315573]
## [6.80684427 7.71315573]]
```

```
## [6.80684427 7.71315573]
## [2.90684427 3.81315573]
## [2.90684427 3.81315573]
## [2.90684427 3.81315573] ]
```

Somit lautet zum Beispiel das 95 %-Vertrauensintervall für $\mu_{\text{Kommerziell}}$

$$[7.03, 7.93]$$

□

Beachten Sie, dass unter der Alternativhypothese die Beobachtungen Y_{ij} , im Unterschied zur Zufalls-Stichprobe, nicht alle die gleiche Verteilung haben. Man spricht deshalb von einer „strukturierten Stichprobe“. (Das ist bereits im Zwei-Stichproben-Problem der Fall, wenn der Unterschied zwischen den Stichproben nicht als null angenommen wird.)

Residuenanalyse

Die Modellannahmen des Gruppenmittelwertemodells besagen, dass die Abweichungen der Beobachtungen um die Gruppenmittelwerte im Mittel nicht nur null ist, sondern dass die Standardabweichung der Abweichungen in allen Gruppen identisch ist. Um die Modellannahmen im Gruppenmittelmodell zu überprüfen, zeichnen wir die Residuendiagramme auf.

Beispiel 6.2.6 Verpackungsmethoden von gelagertem Fleisch

Diese Residuenplots liefern keine Hinweise darauf, dass die Modellannahmen verletzt sind.

□

6.2.2. Anova-Test

Als nächstes wollen wir die Frage beantworten, ob es überhaupt Unterschiede zwischen den Gruppen gibt. Die entsprechende Nullhypothese ist

$$H_0 : \mu_1 = \mu_2 = \dots = \mu_g$$

Kapitel 6. Varianz-Analyse

```
meat <- data.frame(steak.id = c(1, 6, 7, 12, 5, 3, 10, 9,
 2, 8, 4, 11), treatment = rep(c("Kommerziell", "Vakuum",
 "Gemischt", "CO2"), each = 3), y = c(7.66, 6.98, 7.8,
 5.26, 5.44, 5.8, 7.41, 7.33, 7.04, 3.51, 2.91, 3.66))
fit <- aov(y ~ treatment, data = meat)
par(mfrow = c(2, 2))
plot(fit, which = c(1:3, 5))
```

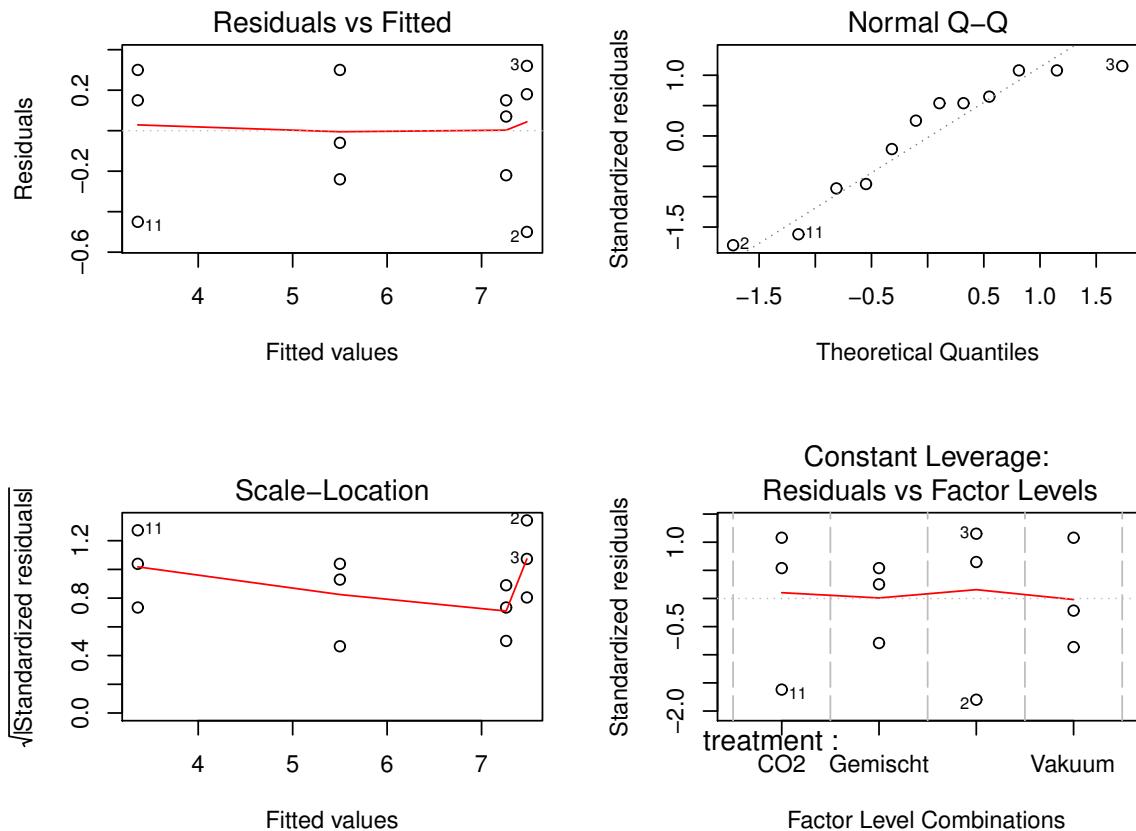


Abbildung 6.4.: Residuenplots für den Beispieldatensatz `Meat` in Abhängigkeit der vier Verpackungsmethoden.

Die Alternativhypothese lautet dann, dass sich mindestens zwei Gruppen unterscheiden, also $\mu_i \neq \mu_j$ mit mindestens einem Paar $i \neq j$.

Zum Beispiel würde die Nullhypothese zu Gunsten der Alternativhypothese verworfen, wenn folgender Fall zutreffen würde:

$$\mu_3 \neq \mu_5$$

Gesucht ist also eine Teststatistik, die extreme Werte annimmt, wenn sich die Grup-

Kapitel 6. Varianz-Analyse

pen in ihrer Lage unterscheiden. Es ist naheliegend, die Gruppenmittelwerte

$$\bar{Y}_{i\bullet} = \frac{1}{m} \sum_{j=1}^m Y_{ij}$$

zu betrachten, wobei wir mit \bullet die Summe über alle Werte vom entsprechenden Index bezeichnen.

Wenn sich die Gruppenmittelwerte stark unterscheiden, ist die Nullhypothese wohl falsch. Was „stark“ heisst, hängt aber auch von der Streuung der Beobachtungen *innerhalb* der Gruppen ab, die durch

$$\frac{1}{n-g} \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2$$

gemessen wird. Mit ihr wird nun die Streuung zwischen den Gruppenmittelwerten verglichen (vgl. auch Abbildung 6.5). Man verwendet die Streuung zwischen Gruppen

$$F = \frac{\text{Streuung zwischen Gruppen}}{\text{Streuung innerhalb Gruppe}} = \frac{\frac{1}{g-1} \sum_{i=1}^g m \cdot (\bar{Y}_{i\bullet} - \bar{Y}_{..})^2}{\frac{1}{n-g} \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2}$$

als Teststatistik, wobei

$$\bar{Y}_{..} = \frac{1}{n} \sum_{i=1}^g \sum_{j=1}^m Y_{ij}$$

das Grand Mean, g die Anzahl der Gruppen und $n = g \cdot m$ die Gesamtzahl Beobachtungen bezeichnen.

Es ist üblich, die Terme, die zur Teststatistik F führen, in einer Art Rechenschema zusammenzustellen, welches *Varianzanalyse-Tabelle* genannt wird (Tabelle 6.3). Auch wenn der Computer die Rechnungen übernimmt, bleibt dieses Schema wichtig, da die einzelnen Teile eine tiefere Bedeutung haben und da es sich auf kompliziertere Modelle verallgemeinern lässt.

Quelle	Quadratsumme	Freiheitsgrade	Mittleres Quadrat
Behandlung (Gruppen)	$SS_G = m \cdot \sum_i^g (\bar{Y}_{i\bullet} - \bar{Y}_{..})^2$	$DF_G = g - 1$	$MS_G = SS_G / DF_G$
Fehler (Residuen)	$SS_E = \sum_i^g \sum_j^m (Y_{ij} - \bar{Y}_{i\bullet})^2$	$DF_E = n - g$	$MS_E = SS_E / DF_E$
Total	$SS_T = \sum_i^g \sum_j^m (Y_{ij} - \bar{Y}_{..})^2$	$DF_T = n - 1$	

Table 6.3.: Varianzanalyse-Tabelle für eine einfache Varianzanalyse.

Die Zeilen des Schemas enthalten in der ersten Kolonne eine „Quadratsumme“. Man kann leicht zeigen, dass sich die beiden Terme SS_G und SS_E in Tabelle 6.3, die ebenfalls in der Teststatistik vorkommen, zu einer „totalen Quadratsumme“ SS_T ergänzen,

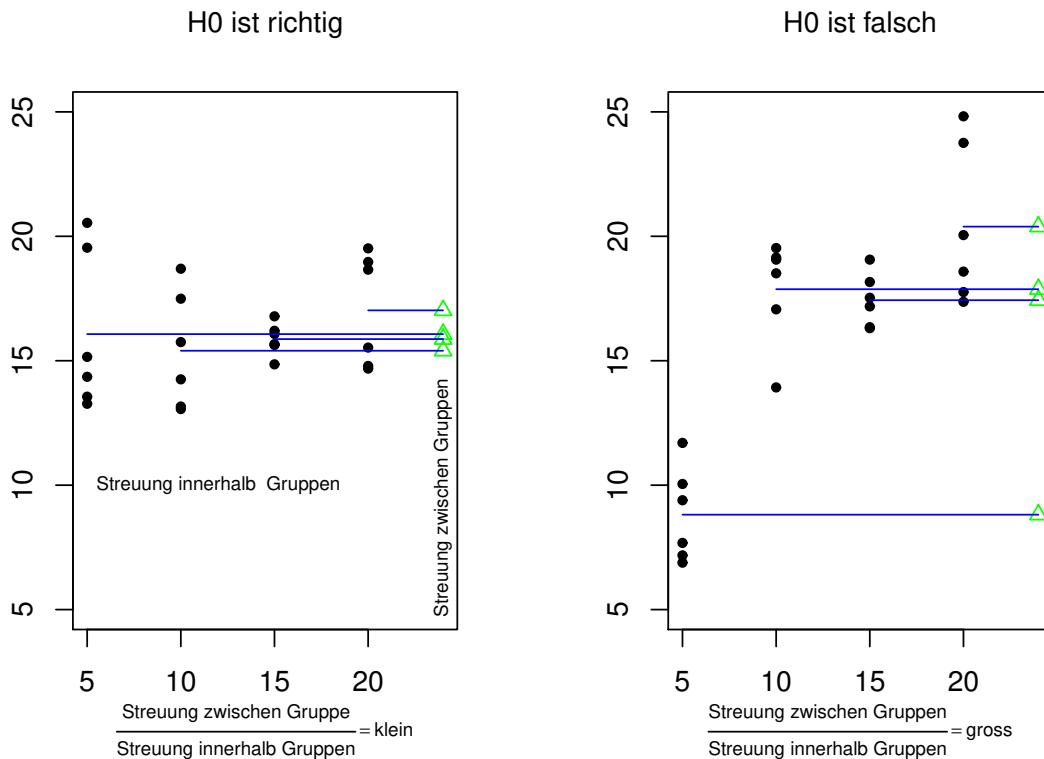


Abbildung 6.5.: Schema eines ANOVA-Tests.

die die quadrierten Abweichungen vom Mittelwert über alle Beobachtungen zusammenzählt. Da die Quadratsummen proportional zu Varianzen sind, spricht man von einer Zerlegung der totalen „Varianz“ in zwei Teile, von denen der eine (SS_G) die „Varianz“ zwischen den Gruppen, die andere (SS_E) die Varianz innerhalb der Gruppen misst. Der Name Varianzanalyse röhrt von diesen Zusammenhängen.

Varianzen entstehen allerdings erst, wenn man die Quadratsummen durch die Zahlen dividiert, die in der Kolonne „Freiheitsgrade“ aufgeführt sind. Man kann zeigen, dass unter der Nullhypothese $Y_{ij} \sim \mathcal{N}(\mu, \sigma^2)$ die mittleren Quadrate MS_G und MS_E erwartungstreue Schätzungen für σ^2 sind.

Die oben erwähnte Teststatistik F kann geschrieben werden als $F = MS_G/MS_E$. Sie vergleicht also die beiden Schätzungen. Die Verteilung von F hängt nur von der Anzahl der Freiheitsgrade DF_G und DF_E ab und wird als F-Verteilung mit DF_G und DF_E Freiheitsgraden bezeichnet. Es handelt sich im Prinzip um denselben Hypothesentest wie in ??

Bemerkungen:

- i. Damit die Behandlungseffekte τ_i eindeutig identifizierbar sind, haben wir diesen eine Nebenbedingung auferlegt:

$$\mu = \mu_1, \quad \tau_1 = 0, \quad \tau_2 = \mu_2 - \mu_1$$

etc. Alternativ könnten wir die Nebenbedingung

$$\tau_g = -(\tau_1 + \dots + \tau_{g-1})$$

fordern. Wichtig ist, dass nur $g - 1$ Elemente der Behandlungseffekte frei variieren können.

Deswegen haben die Behandlungseffekte $g - 1$ Freiheitsgrade (engl. *degrees of freedom*).

- ii. Wir bemerken, dass

$$\begin{aligned} \text{MS}_E &= \frac{1}{n-g} \text{SS}_E \\ &= \frac{1}{n-g} \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2 \\ &= \frac{1}{n-g} \sum_{i=1}^g (m-1) s_i^2 \\ &= \frac{1}{g} \sum_{i=1}^g s_i^2 \\ &= S_{\text{Pool}}^2 \end{aligned}$$

also dass das mittlere Quadrat MS_E dem Quadrat der gepoolten Standardabweichung entspricht, wobei s_i die empirische Standardabweichung der i -ten Gruppe bezeichnet. Wie im Anhang in Kapitel C dargelegt wird, ist MS_E ein *erwartungstreuer Schätzer* der Varianz des Fehlerterms ε_{ij} . Es gilt also $E[\text{MS}_E] = \sigma^2$. Die Schätzung des Fehlerterms hat $n - g$ Freiheitsgrade, oder

$$n - g = \sum_{i=1}^g (m-1)$$

- iii. Man kann zeigen, dass

$$E[\text{MS}_G] = \sigma^2 + \frac{\sum_{i=1}^g m \tau_i^2}{g-1}$$

Somit ist MS_G unter der Nullhypothese $\tau_1 = \tau_2 = \dots = \tau_g = 0$ ein Schätzer von σ^2 und deshalb gilt unter der Nullhypothese

$$F = \frac{\text{MS}_G}{\text{MS}_E} \approx 1$$

Kapitel 6. Varianz-Analyse

Wie im Anhang im Kapitel C gezeigt wird, folgt die Teststatistik F unter der Nullhypothese einer F-Verteilung.

- iv. Der Name *Varianzanalyse* folgt aus der Zerlegung

$$(Y_{ij} - \bar{Y}_{..}) = (Y_{ij} - \bar{Y}_{i\cdot}) + (\bar{Y}_{i\cdot} - \bar{Y}_{..})$$

Dies kann dann, wie im Anhang im Kapitel C gezeigt wird, geschrieben werden als

$$\underbrace{\sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{..})^2}_{SS_T} = \underbrace{\sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\cdot})^2}_{SS_E} + \underbrace{\sum_{i=1}^g \sum_{j=1}^m (\bar{Y}_{i\cdot} - \bar{Y}_{..})^2}_{SS_G}$$

Auf der rechten Seite des Gleichheitszeichens haben wir $(n - g) + (g - 1) = n - 1$ Freiheitsgrade, somit haben wir auf der linken Seite des Gleichheitszeichens $n - 1$ Freiheitsgrade.

Beispiel 6.2.7

Die Verteilung der Teststatistik für das Beispiel **Reissfestigkeit** von Papier kann auch simuliert werden. Dazu wurde ein lineares Regressionsmodell angepasst. Unter der Nullhypothese müssten alle Beobachtungen normalverteilt sein mit Mittelwert gleich 15.958 (Mittelwert der Reissfestigkeit) und Standardabweichung 2.55 (unter **mean_sq** in der Anova-Tabelle in der Zeile **Residual**, wobei die Wurzel aus diesem Wert zu ziehen ist).

In der Simulation werden pro Simulationsschritt 24 Zufallszahlen gemäss dieser Normalverteilung gezogen und anschliessend damit die Teststatistik berechnet, wobei die simulierten Zufallszahlen zufällig auf die 4 Hartholz-Konzentrationen aufgeteilt werden. (zu R)

```

n = 24
g = 4
m = 6

Fstat = np.empty(1000)
for i in np.arange(1000):
    rf_sim = st.norm.rvs(size=n, loc=15.958, scale=2.55)
    rf_mat = np.reshape(rf_sim, (-1, 4))
    grand_mean = np.mean(rf_sim)
    group_mean = np.mean(rf_mat, axis=0)
    MSG = m*np.sum((group_mean-grand_mean)**2) / (g-1)
    MSE = np.sum(np.sum((rf_mat-group_mean)**2, axis=0)) / (n-g)
    Fstat[i] = MSG/MSE

```

Kapitel 6. Varianz-Analyse

```
sns.distplot(Fstat,kde=False, norm_hist=True, hist_kws=dict(edgecolor="black",  
x = np.linspace(0.01,4,num=500)  
y = st.f.pdf(x=x1, dfn=3, dfd=20)  
  
plt.plot(x,y)  
  
plt.xlabel("Fstat")  
  
plt.show()
```

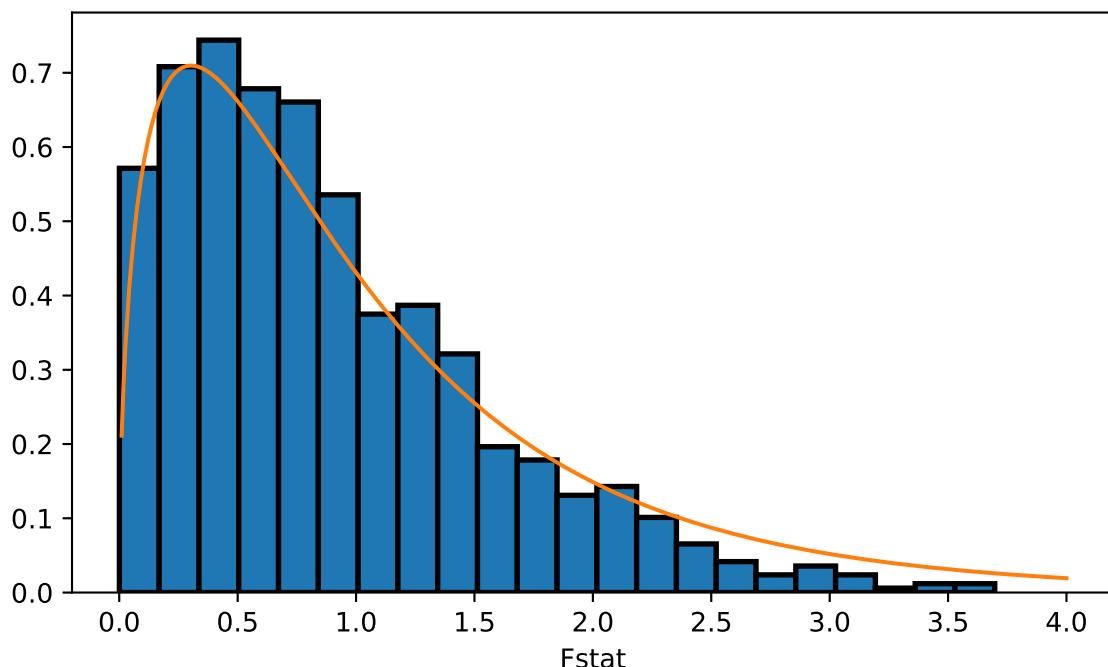


Abbildung 6.6.: Simulierte Verteilung der Teststatistik zum Beispiel **Reissfestigkeit** unter der Nullhypothese. Überlagert ist das Histogramm mit der aus der Theorie bekannten Verteilung der Teststatistik; nämlich der F -Verteilung mit 3 und 20 Freiheitsgraden.

Das Resultat von 1000 Simulationsschritten ist in Abbildung 6.6 in einem Histogramm dargestellt. Zusätzlich wurde dem Histogramm noch die Dichtefunktion der F -Verteilung mit 3 und 20 Freiheitsgraden überlagert.

□

Wie in der Regression werden die Teststatistik-Werte mit Hilfe der aus der Theorie

Kapitel 6. Varianz-Analyse

bekannten Verteilung der Teststatistik unter der Null-Hypothese in P -Werte umgerechnet. In der P -Wert-Skala sind die Verwerfungsbereiche (unplausible Werte) einfach zu merken: Bei P -Werten kleiner als das Niveau wird die Null-Hypothese verworfen, sonst beibehalten.

Beispiel 6.2.8 Reissfestigkeit von Papier

Betrachten wir die Varianzanalyse-Tabelle, wie sie vom Statistik-Programm **Python** ausgegeben wird. ([zu R](#))

```
rf = DataFrame({  
    "HC": np.repeat(["5%", "10%", "15%", "20%"], [6, 6, 6, 6]),  
    "Strength": [7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19, 25]  
})  
  
fit = ols("Strength~HC", data=rf).fit()  
  
anova_lm(fit)  
  
##                df      sum_sq     mean_sq         F    PR(>F)  
## HC            3.0    382.791667   127.597222  19.605207  0.000004  
## Residual     20.0    130.166667     6.508333       NaN        NaN
```

In der ersten Spalte **df** sind die Freiheitsgrade (degrees of freedom), in der zweiten Spalte **sum_sq** die Quadratsummen (Sum of Squares), dann in der dritten Spalte **mean_sq** die mittlere Quadratsumme (Mean Squared) gefolgt von der Teststatistik **F** und zuletzt der P -Wert (**Pr (>F)**).

Der Wert der Teststatistik und der entsprechende P -Wert werden auf der Zeile der Behandlung (entspricht hier der Zeile **HC**) aufgeführt. Der P -Wert von $4 \cdot 10^{-6}$ besagt, dass ein Effekt von unterschiedlichen Hartholz-Konzentrationen signifikant auf dem 5% Niveau nachgewiesen werden kann. Die Gruppenmittelwerte unterscheiden sich also signifikant.

Das ist eigentlich keine Überraschung, wenn man die Boxplots aus Abbildung 6.2 studiert.



Beispiel 6.2.9 Verpackungsmethoden von gelagertem Fleisch

Betrachten wir die Varianzanalyse-Tabelle für den Datensatz **Meat**. ([zu R](#))

```
meat = DataFrame({
  "Treatment": np.repeat(["Kommerziell", "Vakuum", "Gemischt", "CO2"], [3, 3, 3, 3]),
  "steak_id": [7.66, 6.98, 7.80, 5.26, 5.44, 5.80, 7.41, 7.33, 7.04, 3.51, 2.91, 3.3]
})

fit = ols("steak_id~Treatment", data=meat).fit()

anova_lm(fit)

##             df    sum_sq   mean_sq        F    PR(>F)
## Treatment  3.0  32.8728  10.95760  94.584376  0.000001
## Residual   8.0   0.9268   0.11585      NaN       NaN
```

Der P -Wert von $1 \cdot 10^{-6}$ besagt, dass ein Effekt von unterschiedlichen Verpackungsmethoden signifikant auf dem 5 % Niveau nachgewiesen werden kann. Die Gruppenmittelwerte unterscheiden sich also signifikant. Diese Feststellung deckt sich mit der Beobachtung in Abbildung 6.3.

□

6.3. Multiple Vergleiche, multiple Tests

Der besprochene F -Test der einfachen Varianzanalyse beantwortet die Frage, ob zwischen g Gruppen irgendwelche Unterschiede nachgewiesen werden können, d.h., ob die Nullhypothese, dass alle Beobachtungen der gleichen Verteilung folgen, mit den Daten verträglich sei.

Wird die Nullhypothese beibehalten, so erübrigen sich streng genommen weitere Analysen. Im Prinzip läuft dann jede Interpretation von Unterschieden, die durch Paarvergleiche gewonnen werden, Gefahr, etwas rein Zufälliges als systematisch zu bezeichnen. Diese Vorgehensweise entspricht aber einem überaus konservativen Test. Fortgeschrittene Methoden berücksichtigen diese Gefahr.

Wenn sich aber ein signifikantes Testresultat ergeben hat, stellt sich sofort die Frage, ob sich jede Gruppe von jeder anderen unterscheidet, oder ob nur einige der $g(g - 1)/2$ möglichen Unterschiede wirklich nachweisbar sind. Damit sind wir wieder beim ursprünglichen Problem (siehe Beispiele 6.2.2) der vielen Paar-Vergleiche angelangt. Man spricht in der Varianzanalyse vom Problem der multiplen Vergleiche oder multiplen Kontraste.

Es gibt verschiedene Methoden, die dieses Problem lösen, wie das Verfahren von Bonferroni, Bonferroni-Holm, Tukey's Method oder Dunnett's Method.

Im Folgenden wollen wir die Methode von Bonferroni und Bonferroni-Holm besprechen. Zuerst aber verfeinern wir die Formulierung der Nullhypothese beim *F*-Test mit Hilfe von Kontrasten.

6.3.1. Kontraste

Der *F*-Test bezieht sich auf eine Nullhypothese, die sehr unspezifisch ist. Wir würden natürlich gerne eine präzisere Antwort auf die Frage, *inwiefern* sich die Gruppen unterscheiden, erhalten. Ein spezifischere Frage in Bezug auf die Unterschiede zwischen Gruppen kann mit Hilfe von sogenannten *Kontrasten* beantwortet werden.

Beispiel 6.3.1 Kontrast: einfaches Beispiel

Wir möchten Gruppe 2 mit Gruppe 1 vergleichen - weitere Gruppen lassen wir ausser Acht. Wir formulieren die Nullhypothese wie folgt:

$$H_0 : \mu_1 = \mu_2 \quad \text{versus} \quad H_A : \mu_1 \neq \mu_2$$

Äquivalent können wir schreiben

$$H_0 : \mu_1 - \mu_2 = 0 \quad \text{versus} \quad H_A : \mu_1 - \mu_2 \neq 0$$

Der entsprechende *Kontrast* würde in diesem Fall $c = (1, -1, 0, 0, \dots, 0)$ lauten. Ein *Kontrast* $c \in \mathbb{R}^g$ ist ein Vektor, der die Nullhypothese kodiert, und zwar im Sinne von

$$H_0 : \sum_{i=1}^g c_i \cdot \mu_i = 0$$

Im Kontrast wird also die interessierende Forschungsfrage kodiert.

□

Formell ist ein *Kontrast* nichts anderes als ein Vektor

$$c = (c_1, c_2, \dots, c_g) \in \mathbb{R}^g$$

mit der Nebenbedingung $\sum_{i=1}^g c_i = 0$. Die Kontrast-Koeffizienten c_i addieren sich also zu null auf. Diese Nebenbedingung stellt sicher, dass Kontraste sich auf die Unterschiede zwischen Gruppen beziehen, und nicht auf die Gesamtgrösse von unserer Antwort.

Mathematisch ist c orthogonal zum Vektor $(1, 1, \dots, 1)$ resp. $(1/g, 1/g, \dots, 1/g)$, der sich auf den Gesamtmittelwert bezieht.

Mit anderen Worten: Kontraste beziehen sich nicht auf den Gesamtmittelwert.

Beispiel 6.3.2 Verpackungsmethoden von gelagertem Fleisch

Wie wir im Beispieldatensatz **Meat** gesehen hatten, werden folgende Verpackungsmethoden von gelagertem Fleisch in Bezug auf das Bakterienwachstum untersucht:

- Kommerzielle Plastikverpackung (mit Umgebungsluft)
- Vakuumverpackung
- 1 % CO₂, 40 % O₂, 59 % N
- 100 % CO₂

Wenn wir die kommerzielle Plastikverpackung und die Vakuumverpackung als die geläufigen (bisherigen) Verpackungsmethoden bezeichnen und gemischte und CO₂ Methode als die neuen Verpackungsmethoden, dann könnten wir daran interessiert sein, die bisherigen mit den neuen Verpackungsmethoden zu vergleichen. Der entsprechende Kontrast lautet:

$$c = \left(-\frac{1}{2}, -\frac{1}{2}, \frac{1}{2}, \frac{1}{2} \right)$$

Möchten wir die neuen Verpackungsmethoden mit der Vakuum-Verpackungsmethode vergleichen, so lautet der dazugehörige Kontrast

$$c = \left(0, -1, \frac{1}{2}, \frac{1}{2} \right)$$

Der Vergleich von der CO₂-Verpackungsmethode mit der gemischten erfolgt mit folgendem Kontrast

$$c = \left(0, 0, -1, 1 \right)$$

Wären wir am Vergleich zwischen der gemischten Verpackungsmethode und der kommerziellen Verpackungsmethode interessiert, so würden wir folgenden Kontrast benutzen:

$$c = \left(-1, 0, 1, 0 \right)$$

Mit **R** gehen wir wie folgt vor.

```
meat <- data.frame(steak.id = c(1, 6, 7, 12, 5, 3, 10, 9,
                                2, 8, 4, 11), treatment = rep(c("Commercial", "Vacuum",
                                "Mixed", "CO2"), each = 3), y = c(7.66, 6.98, 7.8, 5.26,
                                5.44, 5.8, 7.41, 7.33, 7.04, 3.51, 2.91, 3.66))

## Ueberpruefe die Reihenfolge der Kontraste (wichtig!)
levels(meat$treatment)

## [1] "CO2"          "Commercial"    "Mixed"        "Vacuum"
```

Kapitel 6. Varianz-Analyse

```
##### Kontraste ##

## Definiere individuelle Kontraste ##
contrasts(meat$treatment) <- c(1, -1, 1, -1) ## neu vs. alt

fit <- aov(y ~ treatment, data = meat)
summary(fit, split = list(treatment = list(new.vs.old = 1)))

##                                Df Sum Sq Mean Sq F value
## treatment                  3   32.87  10.958   94.58
##   treatment: new.vs.old    1     4.18    4.177   36.06
## Residuals                   8     0.93    0.116
##                                Pr(>F)
## treatment                  1.38e-06 ***
##   treatment: new.vs.old 0.000322 ***
## Residuals
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Wir können mit **R** auch mehrere Kontraste gleichzeitig testen.

```
## Benutze R-package multcomp #####
library(multcomp)

## Loading required package: mvtnorm
## Loading required package: survival
## Loading required package: TH.data
## Loading required package: MASS

##
## Attaching package: 'TH.data'
## The following object is masked from 'package:MASS':
## 
##      geyser

## Jeder Kontrast ist durch eine Zeile der Matrix M definiert
M <- rbind(c(1, -1, 1, -1), ## neu vs. alt
            c(1, 0, -1, 0)) ## CO2 vs. gemischt
fit.mc <- glht(fit, linfct = mcp(treatment = M))
summary(fit.mc, test = adjusted('none')) ## individual tests
```

Kapitel 6. Varianz-Analyse

```
##  
##      Simultaneous Tests for General Linear Hypotheses  
##  
##      Multiple Comparisons of Means: User-defined Contrasts  
##  
##  
## Fit: aov(formula = y ~ treatment, data = meat)  
##  
## Linear Hypotheses:  
##             Estimate Std. Error t value Pr(>|t|)  
## 1 == 0    -2.3600     0.3930  -6.005 0.000322 ***  
## 2 == 0    -3.9000     0.2779 -14.033 6.45e-07 ***  
## ---  
## Signif. codes:  
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
## (Adjusted p values reported -- none method)  
  
## Konstruiere individuelle Vertrauensintervalle  
confint(fit.mc, calpha = univariate_calpha())  
  
##  
##      Simultaneous Confidence Intervals  
##  
##      Multiple Comparisons of Means: User-defined Contrasts  
##  
##  
## Fit: aov(formula = y ~ treatment, data = meat)  
##  
## Quantile = 2.306  
## 95% confidence level  
##  
##  
## Linear Hypotheses:  
##             Estimate lwr      upr  
## 1 == 0    -2.3600   -3.2663 -1.4537  
## 2 == 0    -3.9000   -4.5409 -3.2591
```



6.3.2. Bonferroni-Regel

Wenn insgesamt m Tests durchgeführt werden sollen, so senkt man das Niveau α für jeden einzelnen Test auf $\alpha/m = 5\%/m$. Wenn ein Test auf diesem Niveau signifikant ist, kann die entsprechende Nullhypothese verworfen werden. Eine Begründung für diese Regel ist leicht zu geben: Wenn alle Nullhypotesen gelten, ist die Wahrscheinlichkeit eines Fehlschlusses 1. Art - nämlich der Schlussfolgerung, mindestens eine Nullhypothese sei falsch - gleich

$$\begin{aligned} P(\text{mind. ein Test signifikant}) &= P\left(\bigcup_{l=1}^m \{\text{l-ter Test ist signifikant}\}\right) \\ &\leq \sum_{l=1}^m P(\text{l-ter Test ist signifikant}) \\ &= \sum_{l=1}^m \alpha/m \\ &= \alpha = 5\% \end{aligned}$$

Beachten Sie, dass hier keine Unabhängigkeit der einzelnen Tests vorausgesetzt wurde; diese ist in den oben geschilderten Fällen nicht erfüllt.

Äquivalent zu diesem Vorgehen können einfach alle p-Werte mit m multipliziert werden und als Signifikanzniveau wird das ursprüngliche α beibehalten.

Beispiel 6.3.3

Mit **R** wird die Korrektur-Methode der p-Werte mit Hilfe von **p.adjusted** oder **adjusted** definiert.

```
## Benutze R-package multcomp #####
library(multcomp)

## Jeder Kontrast ist durch eine Zeile der Matrix M definiert
M <- rbind(c(1, -1, 1, -1), ## neu vs. alt
            c(1, 0, -1, 0)) ## CO2 vs. gemischt
fit.mc <- glht(fit, linfct = mcp(treatment = M))
# p-Werte werden gemaess Bonferroni angepasst
summary(fit.mc, test = adjusted('bonferroni')) ## individual tests

## 
##   Simultaneous Tests for General Linear Hypotheses
##
##   Multiple Comparisons of Means: User-defined Contrasts
##
```

```
## Fit: aov(formula = y ~ treatment, data = meat)
##
## Linear Hypotheses:
##             Estimate Std. Error t value Pr(>|t|)
## 1 == 0    -2.3600     0.3930  -6.005 0.000643 ***
## 2 == 0    -3.9000     0.2779 -14.033 1.29e-06 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- bonferroni method)
```

□

Falls m sehr gross wird, kann die Bonferroni-Regel zu einer sehr konservativen Regel werden, und zwar im Sinne, dass es sehr schwierig ist, die Nullhypothese zu verwerten. In diesem Fall wird die Macht des Tests sehr klein.

6.3.3. Bonferroni-Holm Regel

Die Regel nach Bonferroni-Holm ist weniger konservativ als die Regel von Bonferroni. Wir gehen dabei wie folgt vor : Wir sortieren alle p-Werte in aufsteigender Reihenfolge: $p_{(1)}, p_{(2)}, \dots, p_{(m)}$. Wir verwerfen die Nullhypothese, falls

$$p_{(j)} \leq \frac{\alpha}{(m - j + 1)}$$

für $j = 1, 2, \dots, m$. In diesem Fall erfährt gerade der erste p-Wert die traditionelle Korrektur gemäss Bonferroni. Dieses Vorgehen wird als Step-down-Prozedur bezeichnet: man bricht das Verfahren ab, sobald der erste nicht-signifikante p-Wert erreicht wird.

Beispiel 6.3.4

```
## Benutze R-package multcomp #####
library(multcomp)

## Jeder Kontrast ist durch eine Zeile der Matrix M definiert
M <- rbind(c(1, -1, 1, -1), ## neu vs. alt
            c(1, 0, -1, 0)) ## CO2 vs. gemischt
fit.mc <- glht(fit, linfct = mcp(treatment = M))
# p-Werte werden gemäss Bonferroni-Holm angepasst
## Individuelle Tests
summary(fit.mc, test = adjusted('BH'))
```

```

## Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: User-defined Contrasts
##
##
## Fit: aov(formula = y ~ treatment, data = meat)
##
## Linear Hypotheses:
##             Estimate Std. Error t value Pr(>|t|)
## 1 == 0    -2.3600    0.3930  -6.005 0.000322 ***
## 2 == 0    -3.9000    0.2779 -14.033 1.29e-06 ***
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- BH method)

```

□

6.3.4. Paarweise Vergleiche

Ein paarweiser Vergleich ist nichts anderes als der Vergleich zwischen zwei spezifischen Behandlungen - z.B. Vakuum versus CO₂. Dies ist ein multiples Testproblem, da es gesamthaft

$$g \frac{g-1}{2}$$

mögliche Vergleiche gibt. Die einfachste Lösung, die p-Werte für dieses multiple Testproblem anzupassen, bestünde in der Anwendung der Regel von Bonferroni oder Bonferroni-Holm. Eine wesentlich höhere Macht ergibt das Verfahren gemäss *Tukey Honest Significant Difference*.

Wir gehen hier nicht auf die theoretische Grundlage dieses Verfahrens ein, sondern illustrieren bloss die Anwendung mit **R**.

Beispiel 6.3.5

```

fit <- aov(y ~ treatment, data = meat)
TukeyHSD(fit)

## Tukey multiple comparisons of means
## 95% family-wise confidence level

```

Kapitel 6. Varianz-Analyse

```
##  
## Fit: aov(formula = y ~ treatment, data = meat)  
##  
## $treatment  
##          diff      lwr      upr      p adj  
## Commercial-CO2  4.12  3.230038  5.009962 0.0000020  
## Mixed-CO2       3.90  3.010038  4.789962 0.0000031  
## Vacuum-CO2      2.14  1.250038  3.029962 0.0002639  
## Mixed-Commercial -0.22 -1.109962  0.669962 0.8563618  
## Vacuum-Commercial -1.98 -2.869962 -1.090038 0.0004549  
## Vacuum-Mixed     -1.76 -2.649962 -0.870038 0.0010160
```

```
## Benuetzung des Pakets multcomp  
library(multcomp)  
confint(glht(fit, linfct = mcp(treatment = "Tukey")))  
  
##  
##   Simultaneous Confidence Intervals  
##  
## Multiple Comparisons of Means: Tukey Contrasts  
##  
##  
## Fit: aov(formula = y ~ treatment, data = meat)  
##  
## Quantile = 3.2002  
## 95% family-wise confidence level  
##  
##  
## Linear Hypotheses:  
##                         Estimate lwr      upr  
## Commercial - CO2 == 0    4.1200  3.2306  5.0094  
## Mixed - CO2 == 0         3.9000  3.0106  4.7894  
## Vacuum - CO2 == 0       2.1400  1.2506  3.0294  
## Mixed - Commercial == 0 -0.2200 -1.1094  0.6694  
## Vacuum - Commercial == 0 -1.9800 -2.8694 -1.0906  
## Vacuum - Mixed == 0     -1.7600 -2.6494 -0.8706
```

Die Vertrauenintervalle können auch graphisch dargestellt werden, siehe Abbildung ??



Falls multiple Vergleiche mit einer Kontrollbehandlung durchgeführt werden, bietet sich die Methode von *Dunnett* an.

6.3.5. Konfirmatorische und explorative Analyse

Das Problem des multiplen Vergleichs wird vermieden, indem man sich auf eine Frage beschränkt, die getestet werden soll, oder auf einige wenige, und eine der Regeln für multiples Testen oder für Kontraste beachtet. Die Fragen sollen vor Durchführung des Versuchs oder der Beobachtungen festgelegt sein. Wird dies nicht eingehalten, so stehen die „statistisch gesicherten“ Aussagen auf unsicherem Grund, und man darf streng genommen nicht von einer statistischen Überprüfung von Hypothesen oder beispielsweise einem statistisch gesicherten Effekt einer Behandlung sprechen. Beim Nachweis der Wirksamkeit neuer Medikamente müssen die Regeln von Gesetzes wegen eingehalten werden. Eine solche Auswertung von Ergebnissen einer Studie heisst *konfirmatorische Analyse*.

Andererseits wäre es unsinnig, die Daten, die man oft unter grossen Mühen gewinnt, nicht nach allen möglichen Gesichtspunkten hin anzusehen und auch nach Überraschungen abzuklopfen. Man spricht von einer *explorativen Analyse*. Graphische Darstellungen eignen sich dafür besonders gut. Die Versuchung ist gross, einen Unterschied zwischen irgendwelchen Gruppen, auf die man bei informellen Analysen stösst, mit einem geeigneten Test als „statistisch gesichert“ nachzuweisen. Damit treibt man aber das Problem des multiplen Testens zum Extrem: Man weiss nicht einmal, auf wie viele mögliche „Effekte“ man ebenfalls reagiert hätte, wenn sie sich in den Daten gezeigt hätten, oder anders gesagt, von wie vielen allfälligen Tests der durchgeföhrte das möglicherweise extremste Ergebnis zeigt. Formelle Tests sind deshalb in der explorativen Analyse von untergeordneter Bedeutung.

Zur genaueren Beurteilung eines interessanten angedeuteten Phänomens, das aber möglicherweise „rein zufällig“ in den Daten erscheint, ist es dennoch sinnvoll, P-Werte zu berechnen, sofern man sich bewusst bleibt (und ins Manuscript schreibt), dass eine Interpretation als Test-Ergebnis nicht erlaubt ist. Ein formell signifikantes Ergebnis - manchmal auch ein nicht signifikantes - beinhaltet eine Hypothese, die gegebenenfalls in einer weiteren Studie nach den oben erwähnten Regeln der Kunst überprüft werden kann.

Machtaanalyse beim statistischen Test

Bevor wir zur Zweiweg-Varianzanalyse schreiten, möchten wir uns unoch die Frage stellen, wie wir die *Macht* eines statistischen Tests berechnen können.

Beispiel 6.3.6 Berechnung der Macht anhand von Simulationen

Angenommen wir haben $g = 5$ unterschiedliche Behandlungen, die wir mit A, B, C, D und E bezeichnen. Vorangehende Experimente legen nahe, dass die Fehlervarianz

Kapitel 6. Varianz-Analyse

durch $\sigma^2 = 7.5$ gegeben ist. Wir gehen davon aus, dass sich mindestens zwei Gruppen durch 6 Einheiten unterscheiden. Wir benützen das Modell

$$Y_{ij} = \mu + \alpha_i + \varepsilon_{ij}$$

mit der Nebenbedingung $\sum_i \alpha_i = 0$.

Dies war nun *ein* Datensatz. Nun betrachten wir eine ganze Reihe solcher Datensätze, indem wir 1000 solcher Datensätze generieren. Für jeden Datensatz überprüfen wir, ob die globale Nullhypothese verworfen wird oder nicht.

Kapitel 6. Varianz-Analyse

```
## Machtanalyse mit Hilfe von Simulation

## Stichprobenumfang pro Gruppe
n <- 7 ## Stichprobenumfaenge: 4, 5, 6, 7

## Modellannahme der Parameter unter der
## Alternativhypothese
means <- c(7, 13, 10, 10, 10)
sigma <- sqrt(7.5)

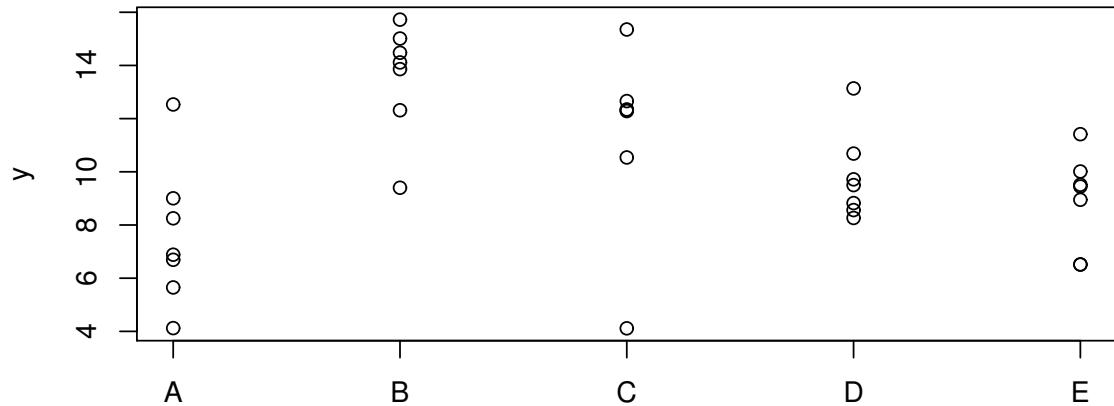
## Wir generieren ein Data-Frame
group <- factor(rep(LETTERS[1:5], each = n))
group

## [1] A A A A A A B B B B B C C C C C C D D D D
## [26] D D D E E E E E
## Levels: A B C D E

## Testlauf : Einzelne Datensaetze #####
y <- rnorm(n * 5, mean = rep(means, each = n), sd = sigma)

data <- data.frame(y = y, group = group)

## Visualisierung
stripchart(y ~ group, data = data, vertical = TRUE, pch = 1)
```



Kapitel 6. Varianz-Analyse

```
## Anpassung eines Einweg ANOVA model
fit <- aov(y ~ group, data = data)
summary(fit)

##           Df  Sum Sq Mean Sq F value    Pr(>F)
## group        4   148.8   37.21   6.172 0.000952 ***
## Residuals   30   180.8    6.03
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Extrahiere p-Wert aus dem globalen Test
summary(fit)[[1]][1, "Pr(>F)"]

## [1] 0.0009522544

## Wir speichern Testentscheidung in einem Vektor der
## Laenge 1000

results <- numeric(1000)

for (i in 1:1000) {
  ## Simuliere neue Antwort
  y <- rnorm(n * 5, mean = rep(means, each = n), sd = sigma)

  data <- data.frame(y = y, group = group)
  fit <- aov(y ~ group, data = data)
  results[i] <- summary(fit)[[1]][1, "Pr(>F)"] < 0.05
}

mean(results)

## [1] 0.882

## = Anteil von Faellen, in denen H_0 verworfen wird
## Resultate n = 4: 0.54 n = 5: 0.69 n = 6: 0.80 n = 7:
## 0.89
```

Wollen wir eine Macht von mindestens 0.8, so muss der Stichprobenumfang $n = 6$ oder grösser sein.

□

6.4. Zweiweg-Varianzanalyse

6.4.1. Randomisiertes vollständiges Block-Design (RCBD)

Die einfache Varianzanalyse verallgemeinert den Vergleich von zwei unabhängigen Stichproben auf mehrere Gruppen. Die entsprechende Verallgemeinerung der gepaarten auf mehrere verbundene Stichproben zeigen wir am folgenden Beispiel.

Beispiel 6.4.1 Vaskuläre Röhrchen

Ein Medizinprodukte-Hersteller produziert vaskuläre Röhrchen (künstliche Venen). Die Röhrchen werden auf einer Strangpresse durch Lochen und Pressen der Bolzen aus Polytetrafluorethylen (PTFE) hergestellt. Einige der Röhrchen aus einem Produktionslauf enthalten kleine, harte Vorsprünge auf deren äusseren Oberflächen. Diese Defekte heißen „Flicks“. Einheiten mit diesem Defekt werden zurückgewiesen.

Der Produktentwickler, der für diese vaskulären Röhrchen verantwortlich ist, vermutet, dass der Extrusionsdruck bei der Herstellung das Auftauchen dieser Vorsprünge beeinflusst. Aus diesem Grund führt er ein Experiment durch, um seine Hypothese zu testen.

Da der Kunststoff allerdings von einem externen Produzenten in Batches (Stapeln) an den Medizinprodukte-Hersteller geliefert wird, muss der Ingenieur in seiner Beurteilung also auch die Möglichkeit berücksichtigen, dass es zwischen den einzelnen Batches signifikante Unterschiede gibt. Auch wenn das Material konsistent bezüglich Parametern wie Molekulargewicht, Partikelgrösse usw. sein sollte, so gibt es Variationen bei der Herstellung des Kunststoffes und natürliche Variationen des Materials. Das heisst, die Qualität der einzelnen Batches kann unterschiedlich sein.

Druck (PSI)	Batch (Block)					
	1	2	3	4	5	6
8500	90.3	89.2	98.2	93.9	87.4	97.9
8700	92.5	89.5	90.6	94.7	87.0	95.8
8900	85.5	90.8	89.6	86.2	88.0	93.4
9100	82.5	89.5	85.6	87.4	78.9	90.7

Table 6.4.: Randomisiertes vollständiges Block-Design für das Röhrchen-Experiment, wobei die Zielvariable (Tabellenwerte) den Prozentsatz der Röhrchen in der Produktionsreihe misst, die keine Flicks enthalten. Der Einfluss des Drucks wird auf vier Stufen untersucht, der Einfluss der Herkunft der Batches auf 6 Stufen.

Der Produktentwickler möchte den Effekt des Extrusionsdrucks auf die Flicks untersuchen (auf vier Stufen), und zwar unter Berücksichtigung der Herkunft der Batches

Kapitel 6. Varianz-Analyse

(auf 6 Stufen). Darum entscheidet er sich für ein randomisiertes vollständiges Blockdesign (RCBD), wobei die Batches als Blöcke behandelt werden. Was ein RCBD ist, soll anhand der Tabelle 6.4 illustriert werden; die genaue Definition folgt später.

Die Zielvariable misst den Prozentsatz der Röhrchen in der Produktionsreihe, die keine Flicks enthalten. Hierbei betrachten wir die Werte Y_{ij} . Dies ist die Messung aus dem j -ten Batch und der i -ten Methode. So ist $y_{23} = 90.6$ der Prozentsatz der Röhrchen ohne Flicks, der für die Probe aus dem 3. Batch bei einem Druck von 8700 PSI gemessen wurde.

Da es natürliche Unterschiede zwischen den Batches gibt, setzt sich die gemessene Variation in Bezug auf Prozentsatz von Flicks aus Variation der Produktionsmethode (Druck) *und* aus der Variation der Batches (Herkunft) zusammen. Uns interessiert in erster Linie die Variation, die aufgrund der Produktionsmethode und nicht aufgrund der Herkunft der Batches entsteht. Um den Einfluss der Batches auf die Zielvariable „wegzusubtrahieren“, werden die Spalten als *Blöcke* betrachtet.

Die *Homogenitätsannahme* besagt dann, dass innerhalb eines Blockes *keine Variabilität* vorhanden ist und die Unterschiede der Messungen innerhalb eines Blockes ausschliesslich auf die entsprechenden Methoden zurückzuführen sind. Die unterschiedlichen Methoden in den jeweiligen Blöcken haben also alle denselben Effekt. Schematisch sieht das wie in Abbildung 6.7 aus - es handelt sich bei dieser Darstellung nicht um die tatsächlichen Werte aus Tabelle 6.4. Diese Abbildung wird später noch genauer erklärt.

In diesen Blöcken kommen *alle* Messmethoden vor und darum sprechen wir hier von einem *vollständigen Block-Design*.

Die Werte in der Tabelle oben wurden *nicht* der Reihe nach gemessen. Also nicht zuerst der Druck von 8500 für alle 6 Batches, dann für 8700 usw. Die Versuche wurden der Reihe von $(8500, 1), (8500, 2), \dots, (9100, 6)$ nach durchnummierter. Das ergibt 24 Versuche, die dann randomisiert werden: (zu R)

```
import numpy as np
np.random.choice(np.arange(1, 25), 24, replace=False)

## [13  9  3 22  4  6 24 12 11 23  2 16  7 20  8 17 21 14  1 18 10  5 19 15]
```

Damit wird zuerst $(8900, 1)$ (entspricht der Nummer 13), dann $(8700, 3)$ (entspricht der Nummer 9), danach $(8500, 3)$ (entspricht der Nummer 3) usw. gemessen. Wir sprechen in diesem Fall von einem *randomisierten Design*.

Dies ermöglicht, dass zeitliche Änderungen des Messgerätes (z.B. Messgerät „warmgelaufen“, Abnutzung, usw.) auf alle Batches und alle Methoden gleichmässig verteilt werden.

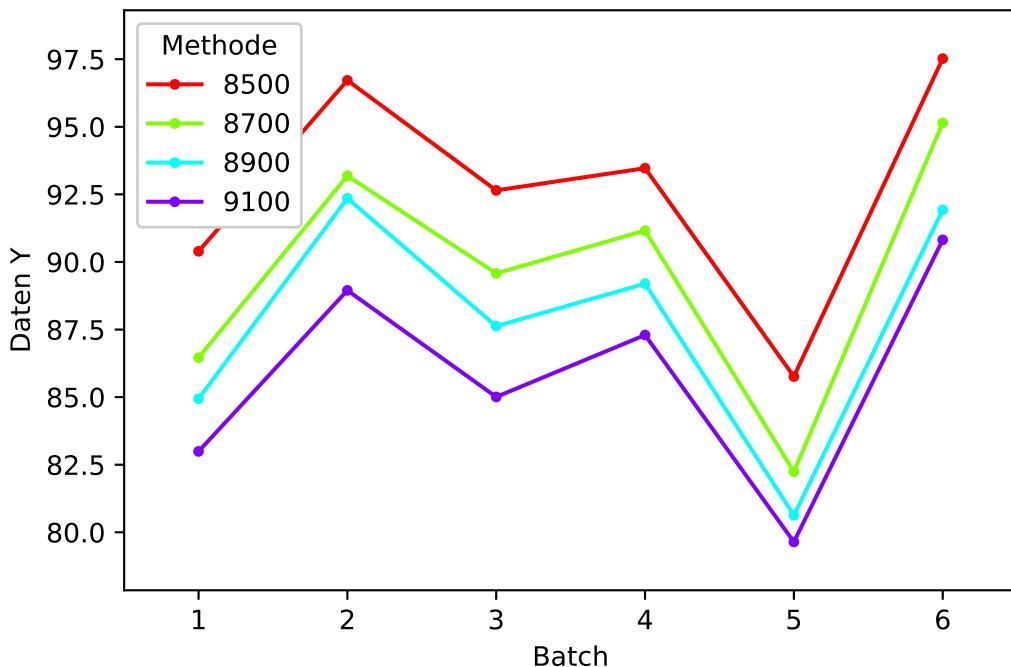


Abbildung 6.7.: Schematische Darstellung (mit künstlichen Daten) für die Homogenitätsannahme: innerhalb eines Blockes haben die Methoden alle denselben Effekt (Additivität der Effekte).

Deswegen sprechen wir im Falle der Versuchsanordnung, wie in Tabelle 6.4 dargestellt, von einem *randomisierten vollständigen Block-Design*.

□

Allgemein: RCBD

Im Allgemeinen haben wir a Methoden und b Blöcke (siehe Abbildung 6.5). Es gibt eine Beobachtung zu jeder Methode in jedem Block. Die Ordnung der Messungen wird zufällig festgelegt.

Wie können wir nun die zufälligen Phänomene in einem Modell beschreiben? Dazu betrachten wir wieder das Röhrchenbeispiel.

Beispiel 6.4.2

In Abbildung 6.8 sind Messwerte aus dem Datensatz **Vaskuläre Röhrchen** aus Tabelle 6.4 graphisch dargestellt. (zu R)

Kapitel 6. Varianz-Analyse

	Block 1	Block 2	...	Block b
Methode 1	y_{11}	y_{12}	...	y_{1b}
Methode 2	y_{21}	y_{22}	...	y_{2b}
:	:	:	:	:
Methode a	y_{a1}	y_{a2}	...	y_{ab}

Table 6.5.: Randomisiertes vollständiges Block-Design mit a Methoden und b Blöcken.

```

from pandas import DataFrame
import pandas as pd
import numpy as np
import seaborn as sns
import scipy.stats as st

from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm
from statsmodels.graphics.factorplots import interaction_plot
import matplotlib.pyplot as plt

Daten = DataFrame({
    "Batch": np.tile(["1", "2", "3", "4", "5", "6"], 4),
    "Methode": np.repeat(["8500", "8700", "8900", "9100"], 6),
    "Y": np.array([90.3, 89.2, 98.2, 93.9, 87.4, 97.9, 92.5, 89.5, 90.6, 94.7, 87,
    90.8, 89.6, 86.2, 88, 93.4, 82.5, 89.5, 85.6, 87.4, 78.9, 90.7])
})

plt.figure(figsize=(7,3))
interaction_plot(x=Daten["Batch"], trace=Daten["Methode"], response=Daten["Y"])

plt.ylabel("Daten Y")

plt.show()

```

Aus dieser Abbildung können wir folgende Beobachtungen herauslesen:

- Bei einem Druck von 9100 (PSI) ist die Prozentzahl von Röhrchen ohne Flicks im Allgemeinen tiefer als bei den anderen Druckwerten. In diesem Produktionslauf wurden bei diesem Druck relativ viele Röhrchen *mit* Flicks produziert.

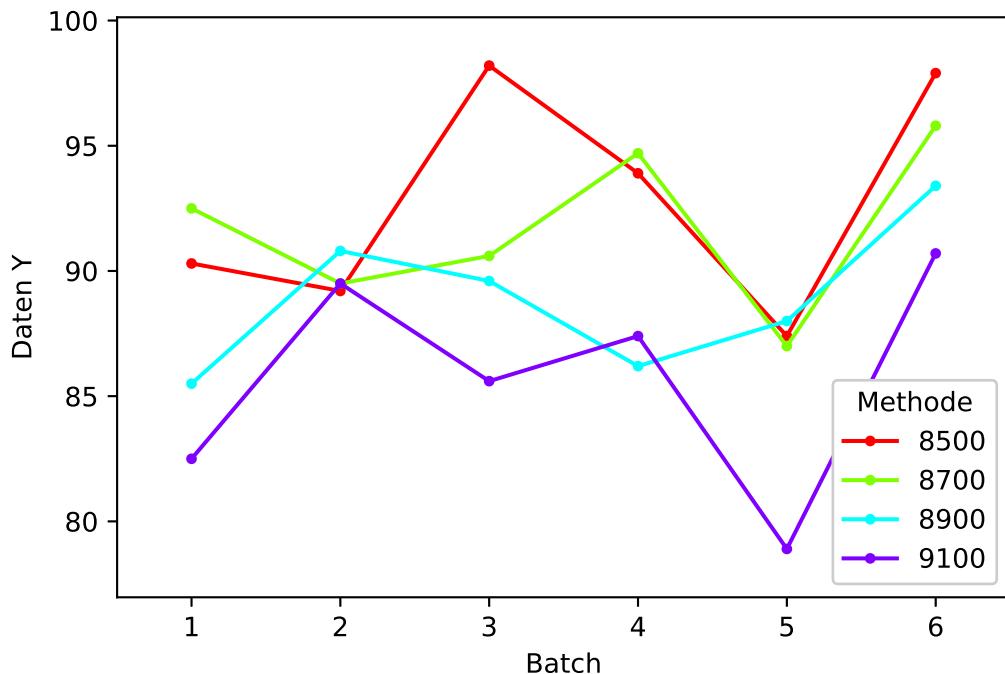


Abbildung 6.8.: Beispieldatensatz : **Vaskuläre Röhrchen**. Der Prozentsatz von Flicks pro Produktionsmethode (Druck) und pro Batch (Herkunft des Kunststoffes) gemäss Tabelle 6.4 ist graphisch dargestellt.

- Bei einem Druck von 8500 (PSI) ist die Prozentzahl von Röhrchen ohne Flicks im Allgemeinen grösser als bei den anderen Drücken. In diesem Produktionslauf wurden bei diesem Druck relativ viele Röhrchen *ohne* Flicks produziert.
- Batch 5 produzierte bis auf den Druck 8900 die tiefsten Werte.

Wir haben also Unterschiede zwischen den Methoden festgestellt. Die Frage stellt sich nun, sind diese Unterschiede *signifikant*?

Entwicklung eines allgemeinen Modells

Die oben angestellten Beobachtungen führen zu einer Idee, wie wir ein Modell entwickeln können, das uns erlaubt, den vermuteten Effekt statistisch zu testen.

Wenn starke Effekte des Batches vorhanden sind, werden sich alle Messwerte nach unten oder oben verschieben. Da die Messungen mit Fehlern behaftet sind, streuen die Beobachtungen entsprechend und verrauschen die tatsächlichen Effekte. Wir gehen wiederum von der Homogenitätsannahme aus, nämlich dass die Methoden in allen Batches denselben Effekt haben. Allerdings addieren wir nun noch ein Rauschen hinzu. In Abbildung 6.9 ist diese Idee graphisch dargestellt.

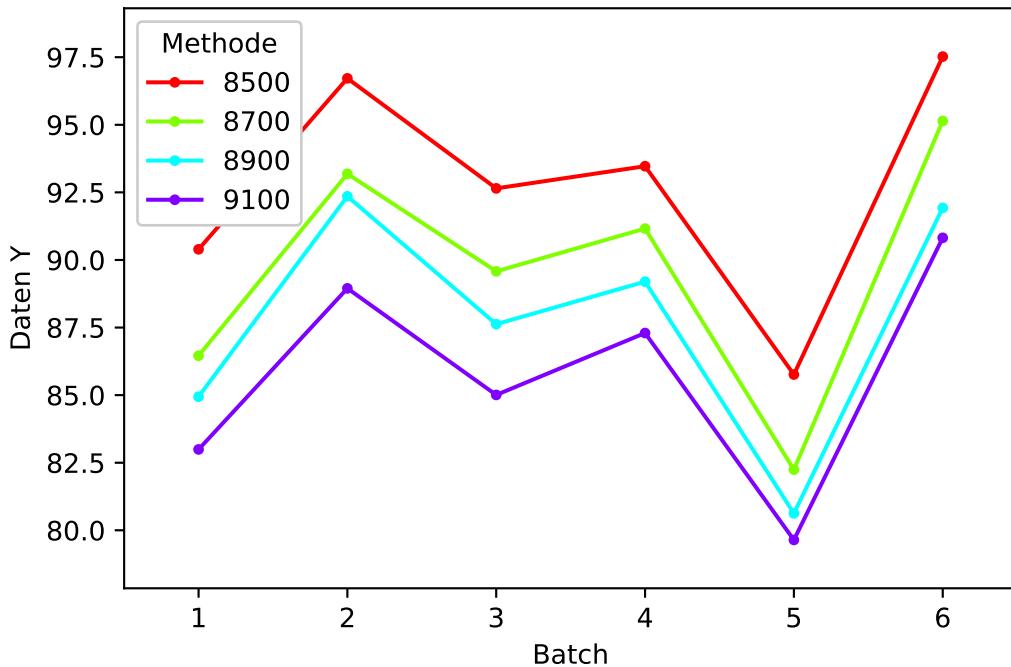


Abbildung 6.9.: Schematische Darstellung (mit künstlichen Daten) für die Homogenitäts-Annahme unter Berücksichtigung von Rauschen: innerhalb eines Blockes haben die Methoden alle denselben Effekt (Additivität der Effekte), allerdings führt das Rauschen zu nicht perfekt parallel verlaufenden Linien im Gegensatz zur Abbildung 6.7.

Die Linien sind nun nicht mehr parallel wie in Abbildung 6.7, sondern nur noch „ungefähr“ parallel. Hier sind die Messfehler bei den einzelnen Methoden mitberücksichtigt worden.

Ein mögliches Modell, das Gesetzmäßigkeiten von zufälligen Phänomenen trennen soll, fasst die einzelne Beobachtung Y_{ij} als eine zufällige Abweichung ε_{ij} von einem „Idealwert“ μ_{ij} auf:

$$Y_{ij} = \mu_{ij} + \varepsilon_{ij} \quad \text{mit } i = 1, \dots, 4; \quad j = 1, \dots, 6$$

wobei μ_{ij} durch die Methode i und den Batch j bestimmt wird.

Alternativ können wir die Messpunkte auch als Zusammensetzung aus einem „globalen Mittelwert“ μ , aus einem Behandlungseffekt α_i des Extrusionsdruckes i und aus einem Effekt β_j des Batches j auffassen, und zwar sollen sich diese Größen addieren:

$$\mu_{ij} = \mu + \alpha_i + \beta_j$$

Eine schematische Darstellung der Additivität der Effekte ist für den Prozentsatz von Flicks auf den Röhrchen in Abbildung 6.7 dargestellt.

Kapitel 6. Varianz-Analyse

Damit die Parameter μ , α_i und β_j eindeutig festgelegt werden können, benötigen wir noch Nebenbedingungen:

$$\sum_{i=1}^4 \alpha_i = 0, \quad \sum_{j=1}^6 \beta_j = 0$$

Diese Nebenbedingung bedeuten, dass sich Effekte der Methoden, wie auch die Effekte der Batches insgesamt aufheben.

□

Allgemeines Modell

Anstelle von Methode sprechen wir allgemeiner von *Faktor A* und anstatt von Batches von *Faktor B*. Wir fassen die einzelne Beobachtung Y_{ij} als eine zufällige Abweichung ε_{ij} von einem „Idealwert“ μ_{ij} auf:

$$Y_{ij} = \mu_{ij} + \varepsilon_{ij} \quad \text{mit } i = 1, 2, \dots, a; \quad j = 1, 2, \dots, b$$

Die Grösse μ_{ij} wird durch die i -te Stufe von Faktor A und durch die j -Stufe von Faktor B bestimmt. Genauer soll sich μ_{ij} aus einem „globalen Mittelwert“ μ , aus einem Effekt α_i der i -ten Stufe von Faktor A und aus einem Effekt β_j der j -ten Stufe von Faktor B zusammensetzen, und zwar sollen sich diese Grössen wieder aufaddieren:

$$\mu_{ij} = \mu + \alpha_i + \beta_j$$

Um die Effekte eindeutig festzulegen, auferlegen wir diesen folgende Nebenbedingungen:

$$\sum_{i=1}^a \alpha_i = 0, \quad \sum_{j=1}^b \beta_j = 0$$

Hypothesentest

In einem Experiment mit einem randomisierten vollständigen Block-Design möchten wir die Gleichheit der Methodenmittel testen. Der Hypothesentest erfolgt dann mit der Nullhypothese

$$H_0 : \quad \mu_1 = \mu_2 = \dots = \mu_a$$

gegen die Alternativhypothese

$$H_A : \quad \mu_i \neq \mu_j \quad \text{für mindestens ein Paar } i \neq j$$

Kapitel 6. Varianz-Analyse

Für das i -te Methodenmittel gilt

$$\mu_i = \frac{1}{b} \sum_{j=1}^b (\mu + \alpha_i + \beta_j) = \mu + \alpha_i$$

da $\sum_j \beta_j = 0$.

Somit können wir die Hypothesen wie folgt formulieren:

$$H_0 : \quad \alpha_1 = \alpha_2 = \dots = \alpha_a = 0$$

mit der Alternativhypothese

$$H_A : \quad \alpha_i \neq 0 \quad \text{für mindestens ein } i$$

Die Varianzanalyse aus dem Abschnitt 6.2 kann einfach auf RCBD erweitert werden. Sei $\bar{Y}_{i\bullet}$ das Mittel der i -ten Methode, $\bar{Y}_{\cdot j}$ das Mittel des j -ten Blockes, $\bar{Y}_{\cdot\cdot}$ das Mittel aller Beobachtungen (Grand Mean) und $N = ab$ die Gesamtzahl der Beobachtungen.

Mathematisch lauten diese Mittelwerte

$$\begin{aligned}\bar{Y}_{i\bullet} &= \frac{1}{b} \sum_{j=1}^b Y_{ij} \quad i = 1, \dots, a \\ \bar{Y}_{\cdot j} &= \frac{1}{a} \sum_{i=1}^a Y_{ij} \quad j = 1, \dots, b \\ \bar{Y}_{\cdot\cdot} &= \frac{1}{ab} \sum_{j=1}^b \sum_{i=1}^a Y_{ij}\end{aligned}$$

Beispiel 6.4.3

In Tabelle 6.6 sind noch die entsprechenden Mittelwerte aus Tabelle 6.4 aufgeführt.

Druck	Batch (Block)						$\bar{y}_{\cdot\cdot} = 89.8$
	1	2	3	4	5	6	
8500	90.3	89.2	98.2	93.9	87.4	97.9	$\bar{y}_{1\bullet} = 92.8$
8700	92.5	89.5	90.6	94.7	87.0	95.8	$\bar{y}_{2\bullet} = 91.7$
8900	85.5	90.8	89.6	86.2	88.0	93.4	$\bar{y}_{3\bullet} = 88.9$
9100	82.5	89.5	85.6	87.4	78.9	90.7	$\bar{y}_{4\bullet} = 85.8$
	$\bar{y}_{\cdot 1} = 87.7$	$\bar{y}_{\cdot 2} = 89.8$	$\bar{y}_{\cdot 3} = 91.0$	$\bar{y}_{\cdot 4} = 90.6$	$\bar{y}_{\cdot 5} = 85.3$	$\bar{y}_{\cdot 6} = 94.5$	

Table 6.6.: Methoden- und Blockmittelwerte für den Datensatz **Vaskuläre Röhrchen**.

□

Parameterschätzung

Als nächstes möchten wir die Parameter μ, α_i, β_j schätzen. Es liegt nahe μ durch $\bar{Y}_{..}$, $\mu + \alpha_i$ durch $\bar{Y}_{i\bullet}$ und $\mu + \beta_j$ durch $\bar{Y}_{\bullet j}$ zu schätzen, also

$$\hat{\mu} = \bar{Y}_{..}, \quad \hat{\alpha}_i = \bar{Y}_{i\bullet} - \bar{Y}_{..}, \quad \hat{\beta}_j = \bar{Y}_{\bullet j} - \bar{Y}_{..}$$

Diese Schätzer lassen sich auch aus dem Kleinsten-Quadrat Kriterium herleiten.

Diese Schätzungen bestimmen für jede Kombination von Methoden i und Batches j einen geschätzten oder „vorhergesagten“ Wert (engl. *fitted value* oder *fit*) der Zielgröße

$$\hat{Y}_{ij} = \hat{\mu} + \hat{\alpha}_i + \hat{\beta}_j$$

Die Abweichung des tatsächlich beobachteten Wertes y_{ij} von diesem Modellwert

$$r_{ij} = y_{ij} - \hat{Y}_{ij}$$

wird wiederum als *Residuum* bezeichnet.

Beispiel 6.4.4

In Tabelle 6.7 sind die Schätzungen für die Behandlungseffekte α_i aus Tabelle 6.6 aufgeführt.

Druck		$\hat{\alpha}_i = \bar{Y}_{i\bullet} - \bar{Y}_{..}$
8500	$\bar{y}_{1\bullet} = 92.8$	3.0
8700	$\bar{y}_{2\bullet} = 91.7$	1.9
8900	$\bar{y}_{3\bullet} = 88.9$	-0.9
9100	$\bar{y}_{4\bullet} = 85.8$	-4.0
		$\bar{y}_{..} = 89.8$

Table 6.7.: Schätzungen von den Behandlungseffekten α_i .

Mit **Python** ermitteln wir die Werte der Behandlungseffekte wie folgt: ([zu R](#))

```
from patsy.contrasts import Sum
fit = ols("Y ~ C(Methode, Sum)+C(Batch, Sum)", data=Daten).fit()

fit.params

## Intercept                  89.795833
## C(Methode, Sum) [S.8500]   3.020833
```

Kapitel 6. Varianz-Analyse

```
## C(Methode, Sum) [S.8700]      1.887500
## C(Methode, Sum) [S.8900]      -0.879167
## C(Batch, Sum) [S.1]           -2.095833
## C(Batch, Sum) [S.2]           -0.045833
## C(Batch, Sum) [S.3]           1.204167
## C(Batch, Sum) [S.4]           0.754167
## C(Batch, Sum) [S.5]           -4.470833
## dtype: float64
```

Mit `C(..., Sum)` aus dem `Patsy`-Modul können wir sicherstellen, dass die Nebenbedingungen

$$\sum_{i=1} \alpha_i = 0 \quad \text{und} \quad \sum_{i=j} \beta_j = 0$$

umgesetzt werden.

Wir lesen aus der `Python`-Ausgabe folglich, dass der globale Mittelwert $\mu = 89.79583$ ist, der Effekt der ersten Behandlung $\alpha_1 = 3.020833$ und der Effekt der zweiten Behandlung $\alpha_2 = 1.8875000$ etc. lauten.

Bemerkungen:

- i. Die letzten Werte für Methode 9100 und Batch 6 werden jeweils *nicht* ausgegeben.
Die lassen sich aber einfach berechnen, da die Summe der jeweiligen Werte 0 sein muss.

□

Anova-Test

Wir haben die Frage gestellt, ob sich die Methoden in ihren Effekten auf den Prozentsatz von Flickern unterscheiden.

Als Grundlage für statistische Tests kann man eine Varianzanalyse-Tabelle (Tabelle 6.14) aufstellen, die eine Erweiterung der früheren Tabelle um eine Zeile für den Faktor B darstellt.

Die Quadratsummen sind definiert durch

$$SS_A = \sum_{i=1}^a \sum_{j=1}^b \hat{\alpha}_i^2 = b \cdot \sum_{i=1}^a \hat{\alpha}_i^2$$

Kapitel 6. Varianz-Analyse

Quelle	Quadratsumme	Freiheitsgrade	Mittleres Quadrat	Teststatistik
Faktor A	SS_A	$DF_A = a - 1$	MS_A	MS_A/MS_E
Faktor B	SS_B	$DF_B = b - 1$	MS_B	MS_B/MS_E
Fehler	SS_E	$DF_E = N - a - b + 1$	MS_E	
Total	SS_T	$DF_T = N - 1$		

Table 6.8.: Varianzanalyse-Tabelle für Zweiweg-Varianzanalyse ohne Wiederholungen (*Replikate*).

und

$$SS_B = \sum_{i=1}^a \sum_{j=1}^b \hat{\beta}_j^2 = a \cdot \sum_{j=1}^b \hat{\beta}_j^2$$

und

$$SS_E = \sum_{i=1}^a \sum_{j=1}^b r_{ij}^2$$

SS_A ist proportional zur Streuung der Gruppenmittelwerte für die unterschiedlichen Methoden, denn

$$\hat{\alpha}_i = \bar{Y}_{i\bullet} - \bar{Y}_{..}$$

stellt ja gerade die Abweichung des Gruppenmittelwerts der i -ten Methode $\bar{Y}_{i\bullet}$ vom Grand Mean $\bar{Y}_{..}$ dar. Damit man dann tatsächlich die Varianz erhält, muss SS_A durch die Anzahl Freiheitsgrade

$$DF_A = a - 1$$

geteilt werden. Dies ergibt dann die Varianz

$$MS_A = \frac{SS_A}{DF_A}$$

Analog ist SS_B proportional zur Streuung der Gruppenmittelwerte für die unterschiedlichen Batches, denn $\hat{\beta}_j = \bar{Y}_{\bullet j} - \bar{Y}_{..}$ stellt ja gerade die Abweichung des Gruppenmittelwerts des j -ten Batches $\bar{Y}_{\bullet j}$ vom Grand Mean $\bar{Y}_{..}$ dar. Damit man dann wiederum die tatsächliche Varianz erhält, muss SS_B durch die Anzahl Freiheitsgrade $DF_B = b - 1$ geteilt werden. Dies ergibt dann die Varianz $MS_B = SS_B/DF_B$.

SS_E ist proportional zu den quadrierten Residuen. Es kann gezeigt werden, dass

$$MS_E = \frac{SS_E}{(a-1)(b-1)}$$

ein erwartungstreuer Schätzer von der Varianz σ^2 des Fehlerterms ε_{ij} ist, also

$$E[MS_E] = \sigma^2$$

Kapitel 6. Varianz-Analyse

Um die Nullhypothese

$$H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_a = 0$$

nämlich dass die Effekte des Faktors A alle null sind, zu testen, bietet sich das Verhältnis MS_A/MS_E als eine natürliche Testgrösse an. Denn diese Teststatistik misst, wie gross die Streuung zwischen den Gruppenmittelwerten für die unterschiedlichen Behandlungseffekte im Vergleich zur Streuung innerhalb der Gruppen, resp. zur Streuung des Fehlerterms ist.

$$F = \frac{MS_A}{MS_E} = \frac{SS_A/DF_A}{SS_E/DF_E}$$

wobei die Anzahl Freiheitsgrade der Residuen

$$DF_E = (a - 1)(b - 1) = N - a - b + 1$$

und

$$DF_A = a - 1$$

ist. Falls die Fehler ε_{ij} normalverteilt sind und alle die gleiche Varianz σ^2 haben, also

$$\varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

dann folgt F unter der Nullhypothese wieder einer F-Verteilung mit DF_A und DF_E Freiheitsgraden, und zwar unabhängig davon, ob Effekte des Faktors B vorhanden sind oder nicht.

Analog kann man testen, ob der Faktor B Effekte zeigt.

Bemerkungen:

- i. Damit die Behandlungseffekte α_i und die Effekte der Batches β_j eindeutig identifizierbar sind, haben wir diesen eine Nebenbedingung auferlegt:

$$\sum_{i=1}^a \alpha_i = 0, \quad \sum_{j=1}^b \beta_j = 0$$

Wichtig ist, dass nur $a - 1$ Elemente der Behandlungseffekte frei variieren können. Deswegen haben die Behandlungseffekte $a - 1$ Freiheitsgrade. Analog dazu haben die Effekte der Batches $b - 1$ Freiheitsgrade.

- ii. Der Name *Varianzanalyse* folgt aus der Zerlegung

$$(Y_{ij} - \bar{Y}_{..}) = (\bar{Y}_{i\bullet} - \bar{Y}_{..}) + (\bar{Y}_{\bullet j} - \bar{Y}_{..}) + (Y_{ij} - \bar{Y}_{i\bullet} - \bar{Y}_{\bullet j} + \bar{Y}_{..})$$

Kapitel 6. Varianz-Analyse

Dies kann dann geschrieben werden als

$$\underbrace{\sum_{i=1}^a \sum_{j=1}^b (Y_{ij} - \bar{Y}_{..})^2}_{SS_T} = \underbrace{\sum_{i=1}^a \sum_{j=1}^b (\bar{Y}_{i\cdot} - \bar{Y}_{..})^2}_{SS_A} + \underbrace{\sum_{i=1}^a \sum_{j=1}^b (\bar{Y}_{\cdot j} - \bar{Y}_{..})^2}_{SS_B}$$

$$+ \underbrace{\sum_{i=1}^a \sum_{j=1}^b (Y_{ij} - \bar{Y}_{i\cdot} - \bar{Y}_{\cdot j} + \bar{Y}_{..})^2}_{SS_E}$$

Auf der linken Seite des Gleichheitszeichen haben wir

$$a \cdot b - 1 = N - 1$$

Freiheitsgrade. Somit muss SS_E also

$$(a - 1)(b - 1) = N - a - b + 1$$

Freiheitsgrade haben.

iii. Man kann zeigen, dass

$$E[MS_A] = \sigma^2 + \frac{b \sum_{i=1}^a \alpha_i^2}{a - 1}$$

$$E[MS_B] = \sigma^2 + \frac{a \sum_{i=1}^b \beta_i^2}{b - 1}$$

$$E[MS_E] = \sigma^2$$

Somit sind unter der Nullhypothese MS_A und MS_B Schätzer von σ^2 und deshalb gilt unter der Nullhypothese

$$F = \frac{MS_A}{MS_E} \approx 1 \quad \text{resp.} \quad F = \frac{MS_B}{MS_E} \approx 1$$

Beispiel 6.4.5

Aus folgendem **Python**-Output lesen wir ab, dass für den interessierenden Faktor Methode ein *P*-Wert von 0.002 resultiert. ([zu R](#))

```
fit = ols("Y ~ C(Methode, Sum)+C(Batch, Sum)", data=Daten).fit()

anova_lm(fit)

##                df      sum_sq     mean_sq        F    PR(>F)
## C(Methode, Sum) 3.0   178.171250  59.390417  8.107077  0.001916
## C(Batch, Sum)   5.0   192.252083  38.450417  5.248666  0.005532
## Residual       15.0   109.886250   7.325750      NaN      NaN
```

Der Behandlungseffekt gilt aufgrund des P-Wertes von 0.00192 auf dem 5 % als gesichert. Es gibt also einen signifikanten Unterschied unter den Methoden.

Offensichtlich gilt auch der Effekt des Block-Faktors Batch als gesichert. Die Qualität der Röhrchen hängt folglich wesentlich von den einzelnen Batches ab. Dies bedeutet wiederum, dass die Qualität unter den Batches sehr unterschiedlich ist.

□

6.4.2. Verbesserung der Genauigkeit mit RCBD im Vergleich zu CRD

Nun möchten wir die Frage beantworten, wie sich der RCBD Versuchsplan auf die Varianz des Fehlerterms im Vergleich zu einem CRD Versuchsplan auswirkt. Dazu betrachten wir zuerst folgendes Beispiel.

Beispiel 6.4.6 Augentropfen

Wir möchten die Wirkung von neuen Augentropfen testen. Dazu wählen wir 10 Testpersonen, die auf dem einen Auge mit den herkömmlichen Augentropfen und auf dem anderen Auge gleichzeitig mit den neuen Augentropfen behandelt werden. Die Zielgröße sei die Rötung der Augen, die auf einer bestimmten quantitativen Skala gemessen wird. Für jeden Patienten wird dann die Differenz zwischen der Rötung beim Auge, das mit den herkömmlichen Augentropfen behandelt wurde, und der Rötung beim Auge, das mit den neuen Augentropfen behandelt wurde, ermittelt. Da wir einen Block als eine Menge von homogenen Versuchseinheiten bezeichnet haben, dürfen wir hier einen Patienten als einen Block auffassen. Mit einem *t*-Test kann nun festgestellt werden, ob diese Differenz eine signifikante Abweichung von null aufweist.

Ein alternativer Versuchsplan bestünde darin, 10 Testpersonen mit den herkömmlichen Augentropfen zu behandeln und 10 andere Testpersonen mit den neuen Augentropfen auf einem Auge zu behandeln. In diesem ungepaarten Test würde dann festgestellt, ob es einen signifikanten Unterschied zwischen den Mittelwerten der beiden Gruppen gäbe.

In welchem der beiden Versuchspläne wäre die Streuung wohl grösser? Natürlich wäre die Streuung im zweiten Versuchsplan viel grösser, da in diesem noch die Streuung zwischen den Testpersonen enthalten ist. Das Bilden von Blöcken verringert die Streuung.

Nun stellt sich die Frage, wie gross diese Reduktion der Streuung durch Blockbildung ist.

□

Wir wollen mit $\frac{\sigma_{\text{CRD}}^2}{n}$ die Varianz der Gruppenmittelwerte (jede Behandlung entspricht einer Gruppe) in einem vollständig randomisierten Versuchsplan bezeichnen, wobei n durch die Anzahl Messungen pro Behandlung gegeben ist.

Im Beispiel mit den Augentropfen ist dies also die Varianz des Mittelwertes der 10 „herkömmlich“ behandelten Testpersonen und des Mittelwertes der 10 „neu“ behandelten Testpersonen. Wir schätzen σ_{CRD}^2 durch eine angebrachte Gewichtung von MS_E und MS_{Block}

$$\hat{\sigma}_{\text{CRD}}^2 = w \cdot \text{MS}_{\text{Block}} + (1 - w) \cdot \text{MS}_E$$

wobei w ein Gewichtungsfaktor ist (siehe [?] für eine detaillierte Diskussion).

Mit $\frac{\sigma_{\text{RCB}}^2}{r}$ bezeichnen wir die Varianz der Gruppenmittelwerte in einem vollständig randomisierten Blockdesign, wobei r die Anzahl Blöcke bezeichnet. Die Anzahl Messungen pro Behandlung ist gerade durch die Anzahl Blöcke gegeben, da in einem üblichen vollständig randomisierten Blockdesign jede Behandlung genau einmal pro Block beobachtet wird. Im Beispiel mit den Augentropfen haben wir 10 Blöcke, nämlich die 10 Testpersonen.

Wir schätzen σ_{RCB}^2 in der Regel durch MS_E . Wenn wir dieselbe Genauigkeit, wofür die Streuung ein Mass ist, in einem vollständig randomisierten Versuchsplan erreichen möchten wie mit einem vollständig randomisierten Blockdesign, dann muss gelten

$$\frac{\sigma_{\text{RCB}}^2}{r} = \frac{\sigma_{\text{CRD}}^2}{n}$$

Sind σ_{RCB}^2 und σ_{CRD}^2 bekannt, so gilt

$$\frac{n}{r} = \frac{\sigma_{\text{CRD}}^2}{\sigma_{\text{RCB}}^2}$$

Wir bezeichnen das Verhältnis der beiden geschätzten Varianzen auch als *relative Effizienz*:

$$RE = \frac{\hat{\sigma}_{\text{CRD}}^2}{\hat{\sigma}_{\text{RCB}}^2}$$

Durch die relative Effizienz können wir das Verhältnis $\frac{n}{r}$ bestimmen. Würde im Falle der Augentropfen die relative Effizienz 2 betragen, so müssten in einem vollständig randomisierten Versuchsplan doppelt so viele Versuchseinheiten benutzt werden, um dieselbe Genauigkeit wie in einem Blockdesign zu erreichen.

6.4.3. Faktorielle Experimente mit zwei Faktoren

Sind bei einem Experiment mehrere Faktoren von Interesse, dann sollte ein *faktorielles Experiment* angewendet werden. Dabei wird das Experiment mit Kombinationen der Stufen der betrachteten Faktoren durchgeführt. Wir betrachten den einfachsten Fall eines faktoriellen Experiments, welches nur zwei Faktoren enthält.

Beispiel 6.4.7

Bei Flugzeugen wird Grundierungsfarbe auf die Aluminiumoberflächen aufgetragen. Es gibt dabei zwei Verfahren, wie die Grundierungsfarbe aufgetragen wird: einerseits durch Eintauchen, andererseits durch Besprühen der Oberflächen. Der Zweck dieser Grundierung liegt darin, dass die Haftung der eigentlichen Farbe verbessert wird.

Die Ingenieure, die für diesen Prozess verantwortlich sind, möchten wissen, ob sich die drei unterschiedlichen Grundierungstypen und die zwei Verfahren, die Farben aufzutragen, in den Hafteigenschaften unterscheiden. Die Ingenieure könnten nun mit dem vertrauten Gruppenmittelmodell ihre Fragestellung beantworten. Allerdings werden sie Mühe haben, Fragen an beide Behandlungsmethoden zu beantworten:

1. Ist der Effekt der Grundierungsfarbe abhängig vom verwendeten Verfahren, diese Farbe aufzutragen?
(→ *Interaktionseffekt zwischen Grundierungsfarbe und Verfahren*)
2. Wie gross ist der Effekt der Grundierungsfarbe gemittelt über alle Verfahren?
(→ *Haupteffekt* der Grundierungsfarbe)
3. Wie gross ist der Effekt des Verfahrens gemittelt über alle Grundierungstypen?
(→ *Haupteffekt* vom Verfahren)

Sie führen deshalb ein faktorielles Experiment mit zwei Faktoren durch, deren Resultate in Tabelle 7.1 aufgeführt sind.

Grundierungstyp	Eintauchen	Besprühen
A	4.0, 4.5, 4.3	5.4, 4.9, 5.6
B	5.6, 4.9, 5.4	5.8, 6.1, 6.3
C	3.8, 3.7, 4.0	5.5, 5.0, 5.0

Table 6.9.: Die Zielgröße Haftungsfestigkeit des Datensatzes **Grundierungsfarben** ist aufgeführt, und zwar in Abhängigkeit vom Grundierungstyp und von den Grundierungsmethoden.

Es wurden also für jede Stufe der Faktoren jeweils 3 Messungen vorgenommen. Auch hier wurden die Versuche in zufälliger Reihenfolge durchgeführt, um den Einfluss von unbekannten Störfaktoren zu minimieren.

Kapitel 6. Varianz-Analyse

Die Daten aus der Tabelle 7.1 sind graphisch in Abbildung 7.1 dargestellt. Dabei wurden als Plotpunkte jeweils die *Mittelwerte* aus den einzelnen Kombinationen der Faktorstufen genommen. (zu R)

```
Farbe = DataFrame({  
    "Grund": np.repeat(["A", "B", "C"], 6),  
    "Methode": np.tile(np.repeat(["Eintauchen", "Bespruehen"], 3), 3),  
    "Y": np.array([4, 4.5, 4.3, 5.4, 4.9, 5.6, 5.6, 4.9, 5.4, 5.8, 6.1, 6.3, 3.8, 3.2])  
})  
  
interaction_plot(x=Farbe["Grund"], trace=Farbe["Methode"], response=Farbe["Y"])  
  
plt.xlabel("Grundierungstypen")  
plt.ylabel("Mittelwerte Haltungsfestigkeit")  
  
plt.show()
```

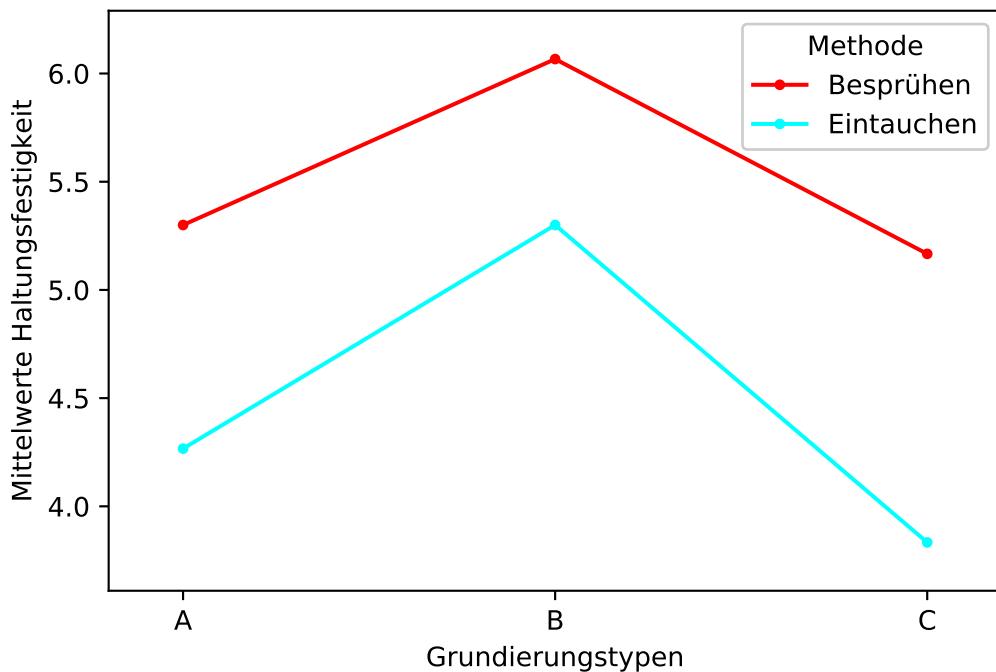


Abbildung 6.10.: Graph der durchschnittlichen Haftungsfestigkeit für den Datensatz **Grundierungsfarben** und zwar in Abhängigkeit der Grundierungsmethode und des Grundierungstyps.

Mit der Python-Funktion `interaction_plot()` wird eine Interaktionsgraphik erstellt, wobei entlang der *y*-Achse die Zielgrösse (`response`) und entlang der *x*-Achse der durch `x` festgelegte Faktor aufgetragen werden. Für jede Stufe des in `trace` festgelegten Faktors wird dann eine Linie gezogen.

Aufgrund des Interaktionsplots gibt es keinen Hinweis, dass das Verfahren, Grundierungsfarben aufzutragen, mit der Art der Grundierungsfarbe interagiert. Mit anderen Worten: egal, welche Grundierungsfarbe wir auftragen, das Verfahren, Ein-tauchen oder Besprühen, hat denselben Einfluss auf die Haftfestigkeit.

□

In diesem Beispiel wurden zwei *primäre Faktoren* betrachtet, wobei der einzige Unterschied zum randomisierten Block-Design darin besteht, dass mehrere Messungen für alle Faktorstufen gemacht wurden und die zweite Faktorvariable als eine primäre Variable aufgefasst wird.

Betrachten wir nämlich Abbildung 7.1, dann stellen wir fest, dass sich die Effekte für die Grundierungsmethoden zu den Grundierungstypen addieren (*Additivität*). Die Grundierungstypen verhalten sich in diesem Beispiel also wie Blöcke im randomisierten Block-Design.

Dass dies nicht immer der Fall sein muss, zeigt folgendes Beispiel.

Beispiel 6.4.8 Elritzen

Die Schädlichkeit von Cyanid-Verunreinigungen im Wasser wurde untersucht, indem für vier Konzentrationsstufen (0.16, 0.8, 4 und 20 mg/l) bei zwei Wassertemperaturen (15 und 25 °C) die Überlebenszeiten für jeweils drei Elritzen gemessen wurden.

Es wurden also auch hier jeweils drei Messungen für alle möglichen Faktorstufen gemacht, und die *Mediane* dieser Messungen sind in Tabelle 6.10 aufgeführt.

Temperatur	Konzentration			
	0.16	0.8	4	20
15 °C	46	17	6	7
25 °C	13	6	4	5

Table 6.10.: Überlebenszeiten von Elritzen in Abhängigkeit der Cyanid-Konzentration bei zwei Wassertemperaturen.

Wir stellen diese Tabelle graphisch als Interaktionsplot in Abbildung 6.11 dar.

```
E1 = DataFrame({
    "Konz": np.repeat(["A", "B", "C", "D"], 6),
    "Temp": np.tile(np.repeat(["15C", "25C"], 3), 4),
    "Y": np.array([82, 46, 16, 20, 13, 7, 20, 14, 17, 6, 7, 5, 8, 6, 5, 4, 3, 5, 10])
})
```

Kapitel 6. Varianz-Analyse

```
interaction_plot(x=El ["Konz"], trace=El ["Temp"], response=El ["Y"],  
legendtitle="Methode")  
  
plt.xlabel("Cyanid-Konzentration")  
plt.ylabel("Mediane Experimente")  
  
plt.show()
```

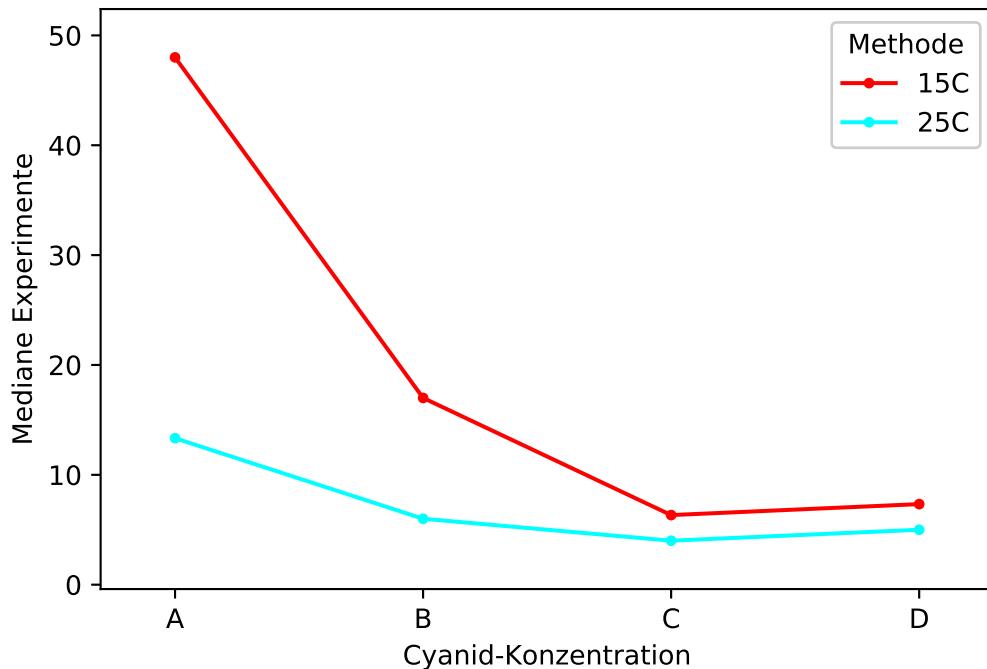


Abbildung 6.11.: "Überlebenszeiten von Elritzen in Abhängigkeit der Cyanid-Konzentration

Es zeigt sich deutlich, dass die Fische bei den beiden Temperaturen nach ähnlichem Muster reagieren, aber ein additives Modell passt nicht, da die Linienzüge nach rechts zusammenlaufen.

Wir müssen hier auch die *Wechselwirkung* zwischen Wassertemperatur und Cyanidkonzentration betrachten. Während bei tiefen Cyanidkonzentrationen die Wassertemperatur einen grossen Einfluss auf die Überlebenszeit der Elritzen ausübt, ist bei hohen Cyanidkonzentrationen der Effekt der Wassertemperatur auf die Überlebenszeit der Elritzen nicht mehr stark. Die Wassertemperatur und die Cyanidkonzentration scheinen also nicht unabhängig voneinander einen Effekt auf die Überlebenszeit der Elritzen zu haben, sondern es scheint eine Wechselwirkung vorzuliegen.

□

Wenn das additive Modell nicht gilt, sprechen wir allgemein von *Wechselwirkungen* oder *Interaktionen*: Wir können dies so interpretieren, dass die i -te Stufe des Faktors A und die j -te Stufe des Faktors B nicht jede für sich wirken, sondern auch die Wirkung der andern beeinflussen.

Totaleffekte und totale Interaktionseffekte

Wir betrachten ein Experiment mit zwei Faktoren A und B mit a bzw. b Stufen. Dann hat das Experiment ab Versuchskombinationen.

Im Folgenden möchten wir das Konzept der Wechselwirkung anhand von einfachen Beispielen besser zu verstehen versuchen. Insbesondere werden wir die Begriffe *Totaleffekt* und *totalen Interaktionseffekt* einführen.

Der Effekt eines Faktors ist definiert als die Änderung der Zielvariable, die bei Änderung der Stufe dieses Faktors auftritt. Dieser Effekt heisst *Totaleffekt*.

Beispiel 6.4.9

Um die Effekte zu untersuchen, betrachten wir ein (hypothetisches) faktorielles Experiment mit zwei Faktoren A und B .

Diese Faktoren haben jeweils zwei Stufen A_{hoch} , A_{tief} und B_{hoch} , B_{tief} . Die Daten sind in Tabelle 6.11 aufgeführt.

		Faktor B	
		B_{tief}	B_{hoch}
Faktor A	A_{tief}	10	20
	A_{hoch}	30	40

Table 6.11.: Faktorielles Experiment mit zwei Faktoren mit jeweils zwei Stufen

Der *Totaleffekt* $e(A)$ von Faktor A ist der Unterschied zwischen den durchschnittlichen Werten der Zielvariablen bei A_{hoch} und den durchschnittlichen Werten der Zielvariablen bei A_{tief} :

$$e(A) = \frac{30 + 40}{2} - \frac{10 + 20}{2} = 20$$

Das heisst, wechselt der Faktor A von „tief“ zu „hoch“, so ändert sich die Zielvariable im Mittel um 20 Einheiten.

Auf analoge Weise können wir den Totaleffekt $e(B)$ von Faktor B bestimmen:

$$e(B) = \frac{20 + 40}{2} - \frac{10 + 30}{2} = 10$$

Diese Resultate waren auch zu erwarten, wenn wir Tabelle 6.11 genauer anschauen. Eine Änderung von Faktor A von „tief“ zu „hoch“ hat bei beiden Stufen von B eine Zunahme von 20 Einheiten zur Folge. Das heisst, die Zunahme ist *unabhängig* von den Stufen von Faktor B .

Die entsprechende Überlegung bei Faktor B geht analog: Eine Änderung von Faktor B von „tief“ zu „hoch“ hat bei beiden Stufen von A eine Zunahme von 10 Einheiten zur Folge. Das heisst, die Zunahme ist *unabhängig* von den Stufen von Faktor A .

Wir können die Daten aus Tabelle 6.11 auch graphisch darstellen (siehe Abbildung 6.12).

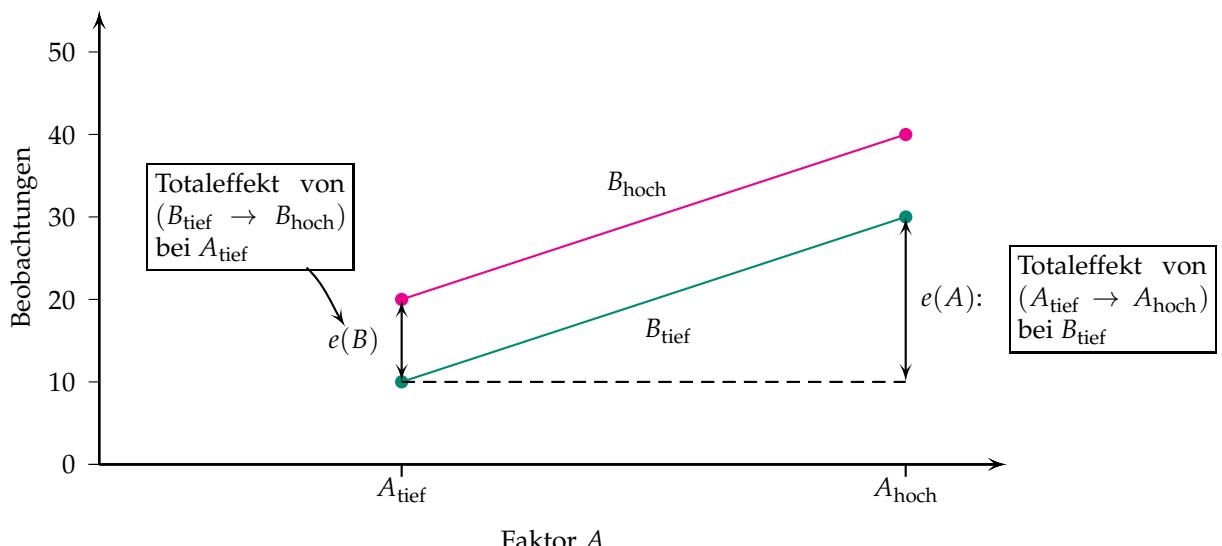


Abbildung 6.12.: Faktorielles Experiment mit zwei Faktoren und je zwei Stufen. Totaleffekt $e(B)$ ist gleich gross auf der Stufe A_{tief} wie auf der Stufe A_{hoch} . Totaleffekt $e(A)$ ist gleich gross auf der Stufe B_{tief} wie auf der Stufe B_{hoch} .

Wir sehen hier, dass die Graphen parallel verlaufen und damit durch Addition ineinander übergehen. Deshalb sprechen wir auch von *Additivität* der Daten.

□

Diese Additivität im Beispiel vorher tritt immer auf, wenn eine Änderung der Stufe von Faktor A die gleiche Änderung der Zielvariablen auf allen Stufen der anderen Faktoren zur Folge hat.

In einigen Experimenten ist aber der Unterschied der Zielvariablen zwischen den Stufen eines Faktors *nicht* gleich auf allen Stufen der anderen Faktoren. Wir sprechen dann von *Interaktion* zwischen den Faktoren.

Beispiel 6.4.10

Wir betrachten wieder ein hypothetisches Experiment, dessen Daten in Tabelle 6.12 aufgeführt sind.

		Faktor B	
		B _{tief}	B _{hoch}
Faktor A	A _{tief}	10	20
	A _{hoch}	30	0

Table 6.12.: Faktorielles Experiment mit zwei Faktoren und Interaktion.

Auf der Stufe B_{tief} ändert der Faktor A den Wert der Zielgröße um

$$30 - 10 = 20$$

und auf Stufe B_{hoch} um

$$0 - 20 = -20$$

Da der Effekt von Faktor A von der Stufe von Faktor B abhängt, sprechen wir von *Interaktion* zwischen den Faktoren A und B.

Wir können auch noch den Totaleffekt des Faktors A berechnen:

$$e(A) = \frac{30 + 0}{2} - \frac{20 + 10}{2} = 0$$

Wir könnten daraus schliessen, dass der Faktor A überhaupt keinen Einfluss hat. Allerdings haben wir gesehen, dass der Faktor A *auf verschiedenen Stufen* von Faktor B durchaus Einfluss hat.

Somit ist die Kenntnis des Interaktionsfaktors AB (hier keine Multiplikation!) wichtiger als diejenige des Hauptfaktors A. Deswegen können bei Vorhandensein von Interaktionsfaktoren die Hauptfaktoren keine grosse Bedeutung haben.

Wir können leicht den *totalen Interaktionseffekt* $e(AB)$ berechnen. Der *totale Interaktionseffekt* bei zwei Faktoren mit je zwei Stufen ist die Hälfte der Differenz zwischen dem Totaleffekt eines Faktors berechnet auf der Stufe „hoch“ des anderen Faktors und dem Totaleffekt desselben Faktors berechnet auf der Stufe „tief“ des anderen Faktors.

Für Tabelle 6.11 gilt für den Interaktionseffekt: der Totaleffekt des Faktors A auf der Stufe „hoch“ des Faktors B ist $40 - 20$. Der Totaleffekt des Faktors A auf der Stufe „tief“ des Faktors B ist $30 - 10$.

$$e(AB) = \frac{1}{2}((40 - 20) - (30 - 10)) = 0$$

Kapitel 6. Varianz-Analyse

Dies war auch zu erwarten, da die Daten additiv sind und mit keinem Interaktionseffekt zu rechnen ist.

Bei Tabelle 6.12 sieht es schon anders aus: der Totaleffekt des Faktors A auf der Stufe „hoch“ des Faktors B ist $0 - 20$. Der Totaleffekt des Faktors A auf der Stufe „tief“ des Faktors B ist $30 - 10$. Folglich ist der totale Interaktionseffekt gegeben durch:

$$e(AB) = \frac{1}{2}((0 - 20) - (30 - 10)) = -20$$

Der Interaktionseffekt ist hier gross, was ebenfalls zu erwarten war. Wir bemerken, dass wir dasselbe Resultat erhalten, wenn wir die Differenz zwischen dem Totaleffekt des Faktors B auf der Stufe „hoch“ des Faktors A und dem Totaleffekt des Faktors B auf der Stufe „tief“ des Faktors A betrachten:

$$e(BA) = \frac{1}{2}((0 - 30) - (20 - 10)) = -20$$

Tabelle 6.12 ist graphisch in Abbildung 6.13 dargestellt.

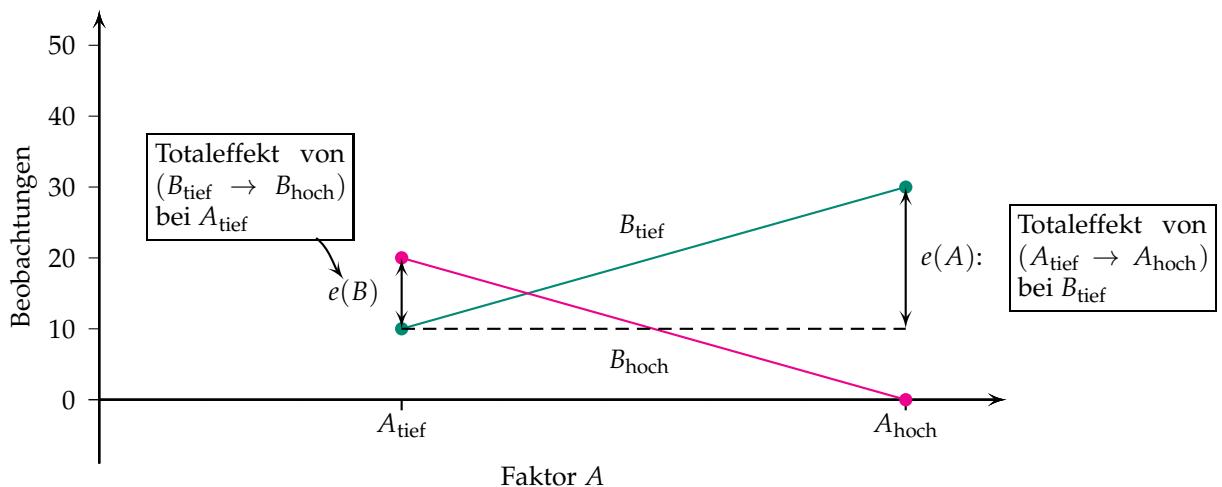


Abbildung 6.13.: Faktorielles Experiment mit zwei Faktoren und Interaktion. Der Totaleffekt $e(B)$ hängt von der Stufe von A ab. Ebenso hängt der Totaleffekt $e(A)$ von der Stufe von B ab.

Aufgrund der nicht parallel verlaufenden Linien ist deutlich erkennbar, dass keine Additivität vorhanden ist, und damit können wir nicht mehr von der Unabhängigkeit der beiden Faktoren ausgehen. Würden die Linien parallel verlaufen, so wäre die Differenz der Totaleffekte null.

□

Kapitel 6. Varianz-Analyse

Der totale Interaktionseffekt ist folglich ein Mass für die *Abweichung* von der Additivität.

Allgemein: Faktorielle Experimente mit zwei Faktoren

Ein faktorielles Experiment enthalte zwei Faktoren: Faktor A hat a Stufen und entsprechend gibt es b Stufen für Faktor B . Das Experiment hat n Wiederholungen oder Replikate (siehe Tabelle 6.13). Jede Wiederholung enthält alle $a \cdot b$ möglichen Stufenpaare. Hier steht Y_{ijk} für die k -te Beobachtung bei Stufe i für Faktor A und bei Stufe j für Faktor B .

		Faktor B				Durchschnitt
		1	2	...	b	
Faktor A	1	$y_{111}, y_{112}, \dots, y_{11n}$	$y_{121}, y_{122}, \dots, y_{11n}$...	$y_{1b1}, y_{1b2}, \dots, y_{1bn}$	$\bar{y}_{1..}$
	2	$y_{211}, y_{212}, \dots, y_{21n}$	$y_{221}, y_{222}, \dots, y_{21n}$...	$y_{2b1}, y_{2b2}, \dots, y_{2bn}$	$\bar{y}_{2..}$
	:					
	a	$y_{a11}, y_{a12}, \dots, y_{a1n}$	$y_{a21}, y_{a22}, \dots, y_{a1n}$...	$y_{ab1}, y_{ab2}, \dots, y_{abn}$	$\bar{y}_{a..}$
Durchschnitt		$\bar{y}_{\bullet 1..}$	$\bar{y}_{\bullet 2..}$...	$\bar{y}_{\bullet b..}$	$\bar{y}_{\bullet ..}$

Table 6.13.: Datenschema für faktorielles Experiment mit zwei Faktoren A (a Stufen) und B (b Stufen) und n Wiederholungen.

Allgemein: Modell mit zwei Faktoren

Das allgemeine Modell für zwei Faktoren, mit Berücksichtigung von Wechselwirkung, kann geschrieben werden als

$$Y_{ijk} = \mu_{ij} + \varepsilon_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}$$

Wir bezeichnen α_i als den *Haupteffekt* von Faktor A auf der Stufe i und β_j als *Haupteffekt* von Faktor B auf der Stufe j . Die Bezeichnung $(\alpha\beta)_{ij}$ hat hier *nicht* die Bedeutung eines Produktes, sondern bezieht sich auf die Wechselwirkung zwischen α und β . Der Wechselwirkungseffekt ist die Abweichung vom Haupteffektenmodell.

Haupteffekte und Zwei-Faktor-Wechselwirkungseffekte für ein Modell mit zwei Faktoren mit jeweils zwei Stufen sind in Abbildung 6.14 dargestellt.

Man braucht für die Wechselwirkungen $(\alpha\beta)_{ij}$ wieder Nebenbedingungen, damit das Modell identifizierbar wird:

$$\sum_i (\alpha\beta)_{ij} = 0 \quad \text{für alle } j \quad \text{und} \quad \sum_j (\alpha\beta)_{ij} = 0 \quad \text{für alle } i$$

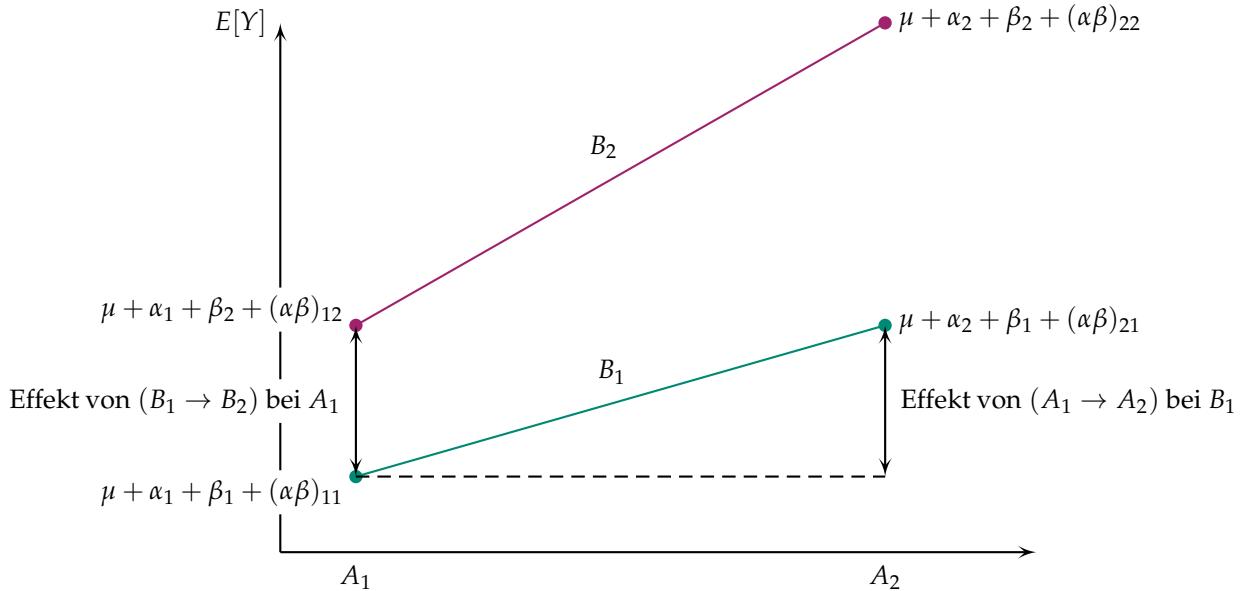


Abbildung 6.14.: Haupteffekte und Zwei-Faktor-Wechselwirkungseffekte für ein Modell mit zwei Faktoren mit jeweils zwei Stufen.

Wir auferlegen dem Modell wieder folgende Nebenbedingungen

$$\sum_{i=1}^a \alpha_i = 0, \quad \sum_{j=1}^b \beta_j = 0$$

Parameterschätzung

Mit ähnlichen Überlegungen wie im letzten Abschnitt lassen sich auch die Schätzungen für die Parameter im allgemeinen Modell für zwei Faktoren herleiten:

$$\hat{\mu} = \bar{Y} \dots, \quad \hat{\alpha}_i = \bar{Y}_{i..} - \bar{Y} \dots, \quad \hat{\beta}_j = \bar{Y}_{.j..} - \bar{Y} \dots, \quad \widehat{(\alpha\beta)}_{ij} = \bar{Y}_{ij..} - \bar{Y}_{i..} - \bar{Y}_{.j..} + \bar{Y} \dots$$

wobei der Punkt im Index immer anzeigt, über welchen Index gemittelt werden soll. Aus der Parameterschätzung für den Wechselwirkungseffekt, $\widehat{(\alpha\beta)}_{ij}$, geht hervor, dass dieser gerade die Abweichung vom Haupteffektenmodell darstellt.

Beispiel 6.4.11

Betrachten wir das Beispiel **Elritzen**, so entnehmen wir die Werte der Behandlungseffekte folgender **Python**-Ausgabe ([zu R](#))

Kapitel 6. Varianz-Analyse

```

E1 = DataFrame({
  "Konz": np.repeat(["A", "B", "C", "D"], 6),
  "Temp": np.tile(np.repeat(["15C", "25C"], 3), 4),
  "Y": np.array([82, 46, 16, 20, 13, 7, 20, 14, 17, 6, 7, 5, 8, 6, 5, 4, 3, 5, 10, 7, 5,
}))

fit = ols("Y~C(Konz, Sum)*C(Temp, Sum)", data=E1).fit()

fit.params

## Intercept                      13.375000
## C(Konz, Sum) [S.A]            17.291667
## C(Konz, Sum) [S.B]            -1.875000
## C(Konz, Sum) [S.C]            -8.208333
## C(Temp, Sum) [S.15C]          6.291667
## C(Konz, Sum) [S.A]:C(Temp, Sum) [S.15C] 11.041667
## C(Konz, Sum) [S.B]:C(Temp, Sum) [S.15C] -0.791667
## C(Konz, Sum) [S.C]:C(Temp, Sum) [S.15C] -5.125000
## dtype: float64

```

Aus dem **Python**-Output entnehmen wir, dass der globale Mittelwert gegeben ist durch $\mu = 13.38$, dass der *Haupteffekt* der Behandlung der Konzentration auf Stufe A 17.29 beträgt (Notation: α_1) und dass der Haupteffekt der Temperatur auf Stufe 15 °C gegeben ist durch 6.29 (Notation: β_1).

Der Effekt aufgrund der *Wechselwirkung* von der Konzentrationsstufe A mit der Temperatur auf Stufe 15 °C beträgt 11.04 (Notation: $(\alpha\beta)_{11}$).

Der Haupteffekt ist nichts anderes als die mittlere Änderung (Effekt), wenn wir in der obigen **Python**-Ausgabe von einer Zeile zur nächsten gehen: Der Effekt von der Konzentration auf Stufe A in Bezug zum globalen Mittelwert μ ist der Haupteffekt $\alpha_1 = 17.29$. Der Effekt von der Temperatur auf der Stufe 15 °C in Bezug auf die Konzentrationsstufe A beträgt $\beta_1 = 6.29$.

Die Differenz zum Haupteffekten-Modell ist durch den Wechselwirkungseffekt zwischen Konzentrationsstufe A und Temperatur auf der Stufe 15 °C durch $(\alpha\beta)_{11} = 11.04$ gegeben. Somit lautet der vorhergesagte Wert der Zielgrösse auf der Konzentrationsstufe A und mit der Temperatur 15 °C:

$$\begin{aligned}
 E[Y_{11k}] &= \mu + \alpha_1 + \beta_1 + (\alpha\beta)_{11} \\
 &= 13.38 + 17.29 + 6.29 + 11.04 \\
 &= 48.0
 \end{aligned}$$

Wir können nun dieselbe Überlegung wiederholen, indem wir die Spalte mit der Konzentrationsstufe B durchgehen.

□

Replikate

Man kann die Wechselwirkungseffekte $(\alpha\beta)_{ij}$ zwar auch dann schätzen, wenn man nur eine Beobachtung für jede Kombination (i, j) hat - also falls die Anzahl Replikate n gleich eins ist. Allerdings kann dann kein Fehler angegeben werden, da die Anzahl der zu schätzenden Parameter gerade gleich der Anzahl Beobachtungen ist (perfekter Fit, somit verschwinden die Residuen).

Um einen Fehler anzugeben, sind für jede Stufenkombination (i, j) mindestens 2 Messungen (Messwiederholungen, Replikate) nötig, also $n > 1$. Dies ist auch in der Varianzanalysetabelle 6.14 ersichtlich: Falls $n = 1$ ist, dann hat die mittlere Quadratsumme MS_E genau $ab(n - 1) = 0$ Freiheitsgrade.

Hypothesentest

Als Grundlage für statistische Tests kann man die Varianzanalyse-Tabelle 6.14 nochmals um eine weitere Zeile für die Wechselwirkung erweitern (Tabelle ??).

Quelle	Quadratsumme	Freiheitsgrade	Mittleres Quadrat	Teststatistik
Faktor A	SS_A	$DF_A = a - 1$	MS_A	MS_A/MS_E
Faktor B	SS_B	$DF_B = b - 1$	MS_B	MS_B/MS_E
Faktor W	SS_W	$DF_W = (a - 1)(b - 1)$	MS_W	MS_W/MS_E
Fehler	SS_E	$DF_E = ab(n - 1)$	MS_E	
Total	SS_T	$DF_T = abn - 1$		

Table 6.14.: Varianzanalyse-Tabelle für die Zweiweg-Varianzanalyse mit Wechselwirkung.

Die Quadratsummen sind definiert durch

$$SS_A = b \cdot c \cdot \sum_{i=1}^a \hat{\alpha}_i^2$$

und

$$SS_B = a \cdot c \cdot \sum_{j=1}^b \hat{\beta}_j^2$$

und

$$SS_E = \sum_{i,j,k} (Y_{ijk} - \bar{Y}_{...})^2 = \sum_{i,j,k} r_{ijk}^2$$

und

$$SS_W = SS_T - SS_A - SS_B - SS_E$$

Kapitel 6. Varianz-Analyse

Für die Nullhypothese

$$H_0 : (\alpha\beta)_{11} = (\alpha\beta)_{12} = \dots = (\alpha\beta)_{ab} = 0$$

nämlich dass die Effekte der Wechselwirkung alle null sind, bietet sich das Verhältnis der Mittleren Quadrate

$$F = \frac{MS_W}{MS_E} = \frac{SS_W/DF_W}{SS_E/DF_E}$$

als eine natürliche Testgrösse an, wobei die Anzahl Freiheitsgrade der Residuen

$$DF_E = ab(c - 1)$$

und

$$DF_W = (a - 1)(b - 1)$$

ist. Falls die Fehler normalverteilt sind mit gleicher Varianz σ^2

$$\varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

hat F unter der Nullhypothese wieder eine F -Verteilung mit DF_W und DF_E Freiheitsgraden, und zwar unabhängig davon, ob Effekte der Faktoren A und B vorhanden sind oder nicht.

Bemerkungen:

- i. Die *Varianzanalyse* folgt aus der Zerlegung

$$\begin{aligned} (Y_{ijk} - \bar{Y}_{...}) &= (\bar{Y}_{i..} - \bar{Y}_{...}) + (\bar{Y}_{..j} - \bar{Y}_{...}) \\ &\quad + (Y_{ij.} - \bar{Y}_{i..} - \bar{Y}_{..j} + \bar{Y}_{...}) + (Y_{ijk} - \bar{Y}_{ij.}) \end{aligned}$$

Dies kann dann geschrieben werden als

$$\begin{aligned} \underbrace{\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^n (Y_{ijk} - \bar{Y}_{...})^2}_{SS_T} &= \underbrace{\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^n (\bar{Y}_{i..} - \bar{Y}_{...})^2}_{SS_A} + \underbrace{\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^n (\bar{Y}_{..j} - \bar{Y}_{...})^2}_{SS_B} \\ &\quad + \underbrace{\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^n (Y_{ij.} - \bar{Y}_{i..} - \bar{Y}_{..j} + \bar{Y}_{...})^2}_{SS_{AB}} \\ &\quad + \underbrace{\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^n (Y_{ijk} - \bar{Y}_{ij.})^2}_{SS_E} \end{aligned}$$

Auf der linken Seite des Gleichheitszeichens haben wir

$$abn - 1 = N - 1$$

Freiheitsgrade, auf der rechten Seite des Gleichheitszeichens sind es deren

$$(a - 1) + (b - 1) + (a - 1)(b - 1) + ab(n - 1) = abn - 1$$

- ii. Beim Blockdesign darf es keine Wechselwirkungen geben, da ein Block mit der Behandlung nicht interagieren kann. Gäbe es in den Daten Hinweise auf Interaktionen, dann wären die Blockfaktoren nicht homogen. Diese Inhomogenität müsste identifiziert und bereinigt werden. Anschliessend kann das Experiment wiederholt werden.

Beispiel 6.4.12

Die Frage, ob solche Wechselwirkungen im Beispiel Elritzen vorliegen, lässt sich nun formell mit einem Test prüfen. Aus dem **Python**-Ouput lesen wir ab, dass für die Wechselwirkung ein P-Wert von 0.1034 resultiert und folglich die Null-Hypothese, dass alle Wechselwirkungsterme $(\alpha\beta)_{ij}$ null sind, auf dem 5 % Niveau nicht verworfen werden kann. ([zu R](#))

```
fit = ols ("Y~C(Konz, Sum)*C(Temp, Sum)", data=El).fit()

anova_lm(fit)

##                                df      sum_sq     mean_sq       F    PR(>F)
## C(Konz, Sum)                3.0   2531.125000  843.708333  5.843867  0.006803
## C(Temp, Sum)                1.0    950.041667  950.041667  6.580375  0.020754
## C(Konz, Sum):C(Temp, Sum)  3.0   1050.458333  350.152778  2.425301  0.103439
## Residual                     16.0  2310.000000  144.375000      NaN      NaN
```

Das Ergebnis des F -Tests für den Interaktionsterm ergibt einen P-Wert von 0.1034, somit können wir die Nullhypothese nicht verworfen. Dieses Resultat stimmt allerdings mit der explorativen Analyse aus der Abbildung 6.11 nicht überein.

Wo könnte hier der Grund für diese Diskrepanz liegen? Zur Beantwortung dieser Frage führen wir eine Residuenanalyse durch. In Abbildung 6.15 ist der Residuenplot aufgeführt : die Residuen werden gegen die geschätzten Gruppenmittelwerte $\hat{\mu}_{ij}$ aufgezeichnet.

Aus der trichterförmigen Struktur des Tukey-Anscombe-Diagramms in Abbildung 6.15 ist eindeutig sichtbar, dass die Varianz der Fehlerterme nicht konstant ist und dass also die Modellannahme bezüglich Konstanz der Varianz der Fehler ε_{ijk} nicht erfüllt ist. Somit ist unser Testresultat nicht aussagekräftig.

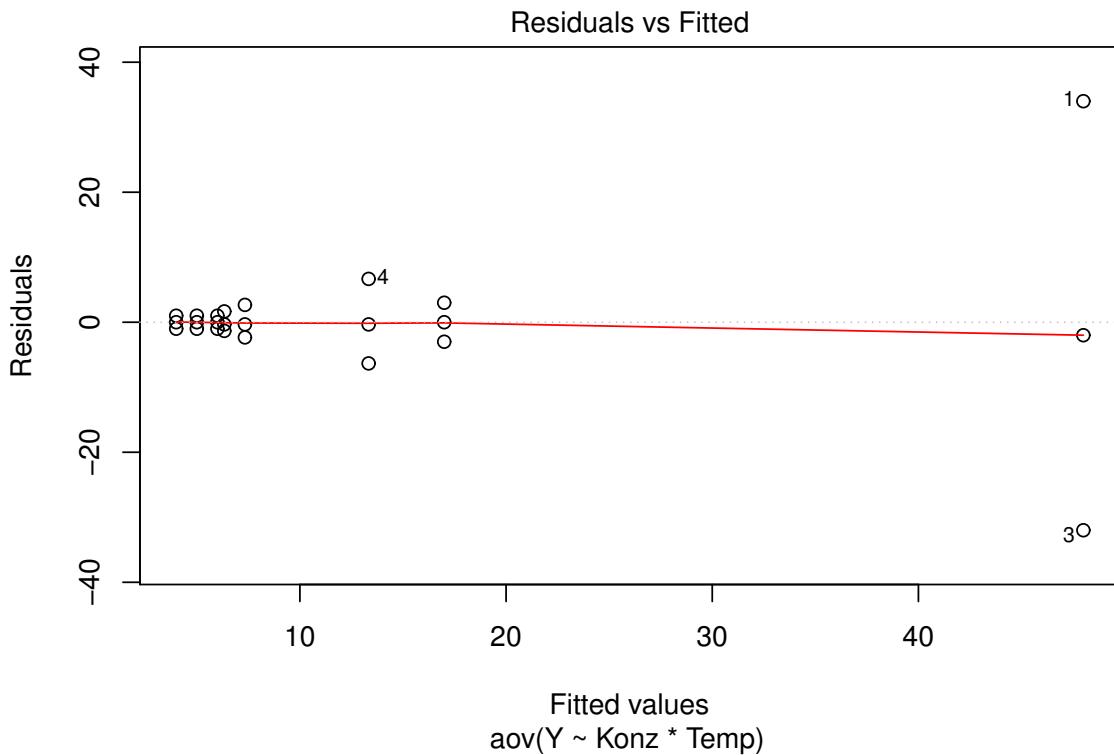


Abbildung 6.15.: Residuenplot für das Beispiel Elritzen, wobei die Residuen werden gegen die geschätzten Gruppenmittelwerte $\hat{\mu}_{ij}$ aufgezeichnet werden. Die Varianz der Residuen scheint nicht konstant zu sein.

Wir führen eine Variablentransformation durch, nämlich wir betrachten die Inverse der Zielgröße, also $\tilde{Y} = 1/Y$. Das resultierende Tukey-Anscombe-Diagramm ist in Abbildung 6.16 dargestellt.

Dieses Tukey-Anscombe-Diagramm sieht nun in Ordnung aus. Betrachten wir nun den Hypothesentest für den Interaktionsterm, so ergibt sich aufgrund der **Python**-Ausgabe ein P-Werte von 0.618020 für den Interaktionsterm. ([zu R](#))

```
E11 = DataFrame({
    "Konz": np.repeat(["A", "B", "C", "D"], 6),
    "Temp": np.tile(np.repeat(["15C", "25C"], 3), 4),
    "Y": 1/np.array([82, 46, 16, 20, 13, 7, 20, 14, 17, 6, 7, 5, 8, 6, 5, 4, 3, 5, 1])
})

fit = ols("Y~Konz*Temp", data=E11).fit()

anova_lm(fit)
```

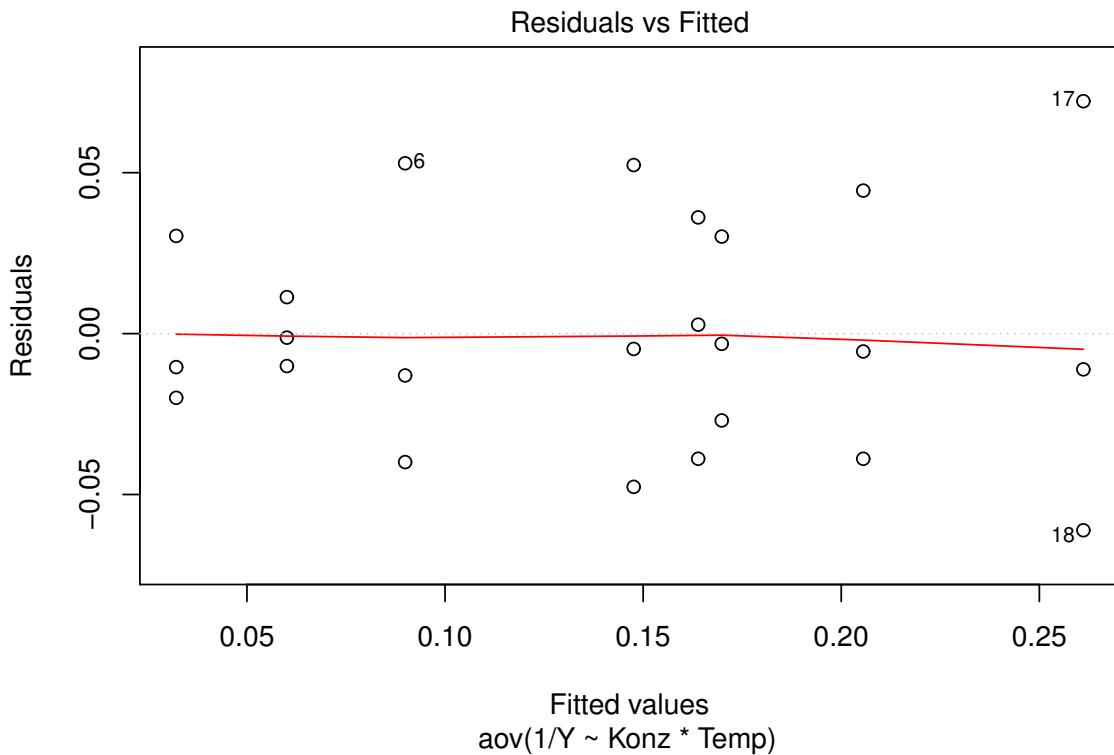


Abbildung 6.16.: Residuenplot für das Beispiel Elritzen mit transformierter Zielgröße $\tilde{y} = 1/y$.

	df	sum_sq	mean_sq	F	PR (>F)
## Konz	3.0	0.080704	0.026901	15.196851	0.000061
## Temp	1.0	0.039050	0.039050	22.059946	0.000242
## Konz:Temp	3.0	0.003241	0.001080	0.610307	0.618020
## Residual	16.0	0.028323	0.001770	NaN	NaN

Daraus schliessen wir, dass es hier *keinen* Interaktionseffekt gibt. Dies bestätigt auch der Interaktionsplot in Abbildung 6.17.

□

Beispiel 6.4.13

Für das Beispiel **Grundierungsfarben** ist es wohl kaum überraschend, dass der Interaktionsterm keinen Einfluss hat. Der **Python**-Output lautet ([zu R](#))

Kapitel 6. Varianz-Analyse

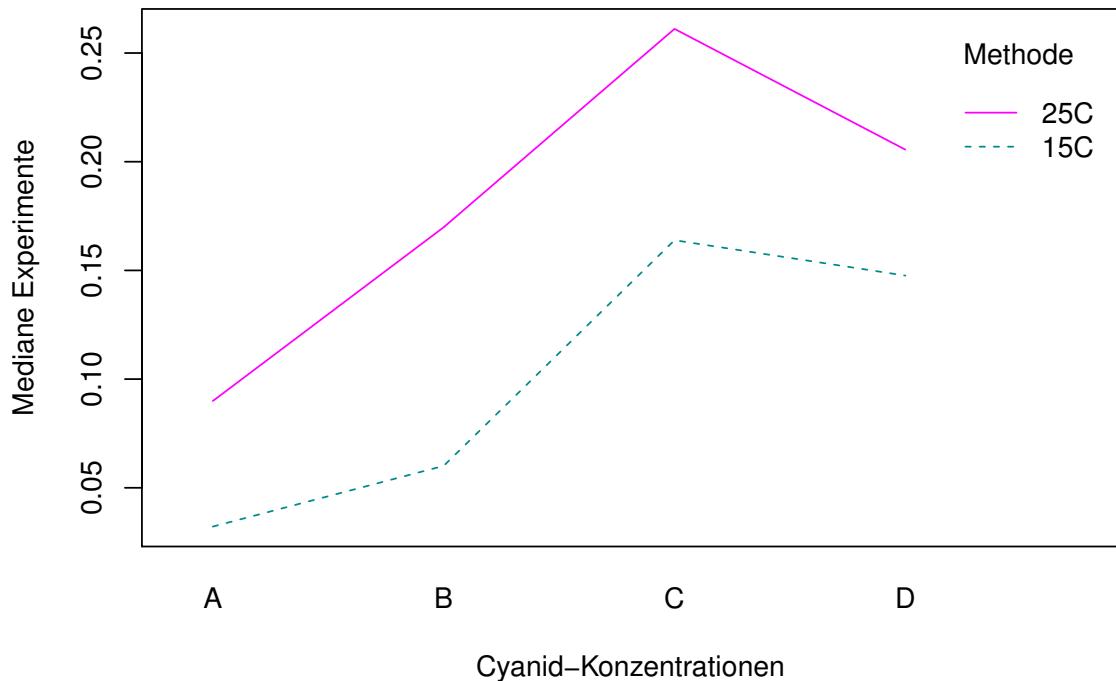


Abbildung 6.17.: Überlebenszeiten von Elritzen in Abhängigkeit der Cyanid-Konzentration mit transformierter Zielgrösse.

```
fit = ols("Y~Grund*Methode", data=Farbe).fit()

anova_lm(fit)

##                df    sum_sq   mean_sq      F    PR(>F)
## Grund          2.0  4.581111  2.290556 27.858108 0.0000031
## Methode        1.0  4.908889  4.908889 59.702703 0.0000005
## Grund:Methode 2.0  0.241111  0.120556  1.466216 0.269342
## Residual       12.0  0.986667  0.082222      NaN      NaN
```

Der P -Wert ist mit 0.269 so gross, dass die Nullhypothese *nicht* verworfen wird. Wir schliessen daraus, dass also keine Interaktion vorhanden ist. Wir wollen die Modellannahmen noch durch eine Residuenanalyse überprüfen (siehe Abbildung 6.15).

In diesen Residuendiagrammen deutet nichts auf eine Verletzung der Modellannahmen hin.

□

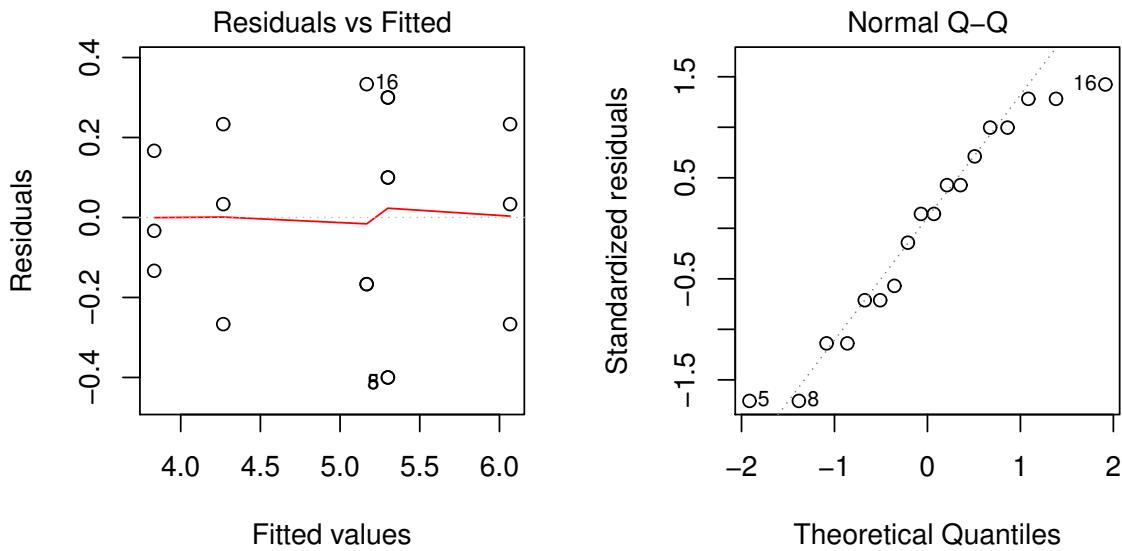


Abbildung 6.18.: Residuen- und Normalplot für den Datensatz **Grundierungsfarben**.

6.4.4. Vorgehensweise bei der Analyse von ANOVA-Tabellen

Typischerweise werden ANOVA-Tabellen von unten nach oben analysiert. Wir beginnen also mit dem F -Test für den Interaktionsterm. Falls wir die Nullhypothese in Bezug auf den F -Test des Interaktionseffekts verwerfen, müssen wir davon ausgehen, dass ein Interaktionseffekt in unserem Modell vorhanden ist.

In diesem Fall sollte nicht mit dem Test der Haupteffekte fortgefahrene werden, sondern es sollten die Haupteffekte individuell in einer Einweg-Faktorenanalyse untersucht werden. Zusammen mit einem Interaktionsplot lässt sich untersuchen, ob der Interaktionseffekt durch eine einzelne Gruppe zustandekommt. Mit einer Residuenanalyse und entsprechenden Variablentransformationen verschwindet unter Umständen der Interaktionseffekt.

Falls der Interaktionseffekt nicht signifikant ist, kann mit der Analyse der Haupteffekte fortgefahrene werden.

Beispiel 6.4.14

Wir betrachten im Folgenden einen Datensatz, in welchem die Fruchtbarkeit von Schnecken untersucht wurde, und zwar in Abhängigkeit der Jahreszeit (Sommer oder

Kapitel 6. Varianz-Analyse

Frühling) und der Dichte der eierlegenden Schnecken (6, 12 oder 24 Schnecken pro Masche). Die Zielvariable zeichnet die Anzahl gelegter Eier auf. Pro Behandlung gab es 3 Messungen. (zu R)

```
snails = DataFrame({
"season": np.repeat(["spring", "summer"], 9),
"density": np.tile(np.repeat(["6", "12", "24"], 3), 2),
"Y": np.array([1.17, 0.50, 1.67, 1.50, 0.83, 1.00, 0.67, 0.67, 0.75,
               4.00, 3.83, 3.83, 3.33, 2.58, 2.75, 2.54, 1.83, 1.63])
})

fit = ols("Y~season*density", data=snails).fit()

anova_lm(fit)

##                df      sum_sq   mean_sq       F    PR (>F)
## season          1.0    17.130756  17.130756  119.373466  1.364839e-07
## density         2.0     4.001011   2.000506   13.940266  7.422016e-04
## season:density 2.0     1.689144   0.844572   5.885293  1.655191e-02
## Residual        12.0    1.722067   0.143506      NaN           NaN
```

Da der Interaktionseffekt zwischen Jahreszeit und Schneckendichte signifikant ist, wollen wir als erstes einen Interaktionsplot zur genaueren Analyse herbeiziehen.

```
from pandas import DataFrame
import pandas as pd
import numpy as np
import seaborn as sns
import scipy.stats as st

from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm
from statsmodels.graphics.factorplots import interaction_plot
from statsmodels.stats.outliers_influence import summary_table
import matplotlib.pyplot as plt
from scipy import stats
from statsmodels.stats.multicomp import pairwise_tukeyhsd, MultiComparison
from patsy.contrasts import Sum

snails = DataFrame({
"season": np.repeat(["spring", "summer"], 9),
"density": np.tile(np.repeat(["6", "12", "24"], 3), 2),
"Y": np.array([1.17, 0.50, 1.67, 1.50, 0.83, 1.00, 0.67, 0.67, 0.75,
               4.00, 3.83, 3.83, 3.33, 2.58, 2.75, 2.54, 1.83, 1.63])
```

Kapitel 6. Varianz-Analyse

```
} )  
  
plt.figure(figsize=(7, 3))  
  
interaction_plot(x=snails["density"], trace=snails["season"],  
response=snails["Y"])  
  
plt.show()
```

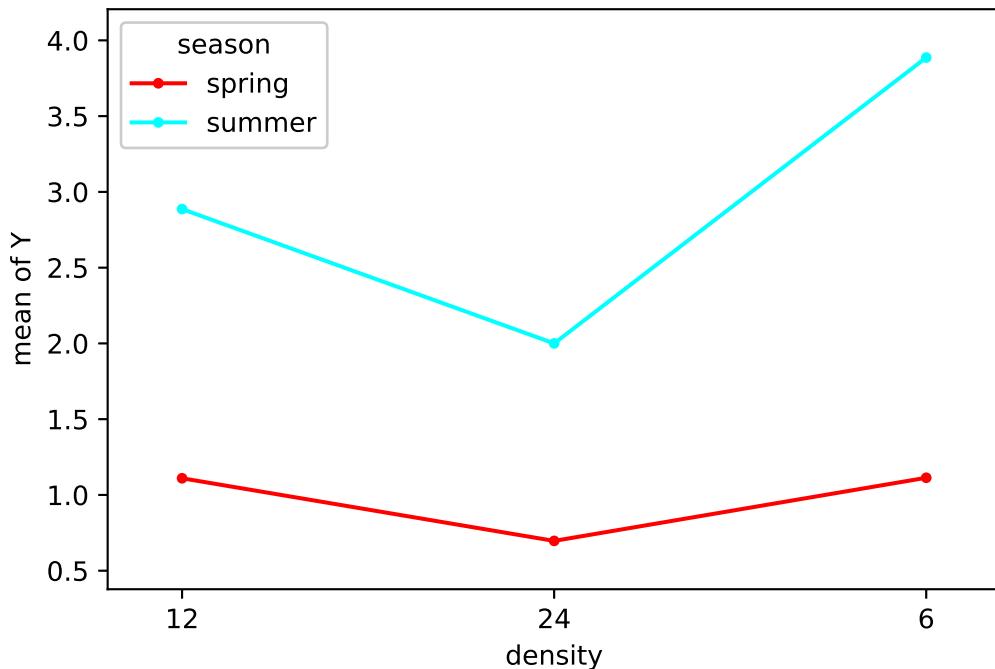


Abbildung 6.19.: Interaktionsplot zwischen Faktor season und density für den Datensatz `snails`.

Da die Linien im Interaktionsplot nicht parallel verlaufen, besteht ein Hinweis auf einen Interaktionseffekt. Insbesondere scheint die Kombination Frühling und eine Schneekendichte von 6 pro Masche für die nichtparallele Struktur im Interaktionsplot verantwortlich zu sein.

Nun betrachten wir für die Jahreszeiten individuell die Modelle. ([zu R](#))

```
snails = DataFrame({  
    "season": np.repeat(["spring", "summer"], 9),  
    "density": np.tile(np.repeat(["6", "12", "24"], 3), 2),  
    "Y": np.array([1.17, 0.50, 1.67, 1.50, 0.83, 1.00, 0.67, 0.67, 0.75,  
        4.00, 3.83, 3.83, 3.33, 2.58, 2.75, 2.54, 1.83, 1.63])  
})
```

Kapitel 6. Varianz-Analyse

```
snails_spring = snails[snails["season"]=="spring"]

fit_spring = ols("Y~density", data=snails_spring).fit()

anova_lm(fit_spring)

snails_summer = snails[snails["season"]=="summer"]

fit_summer = ols("Y~density", data=snails_summer).fit()

anova_lm(fit_summer)

##          df    sum_sq   mean_sq        F    PR(>F)
## density    2.0  0.344467  0.172233  1.103903  0.390636
## Residual   6.0  0.936133  0.156022      NaN       NaN
##          df    sum_sq   mean_sq        F    PR(>F)
## density    2.0  5.345689  2.672844  20.405123  0.002106
## Residual   6.0  0.785933  0.130989      NaN       NaN
```

Wir schliessen daraus, dass es im Sommer signifikante Unterschiede in Bezug auf die Menge der gelegten Schneckeneier für die jeweiligen Gruppen mit unterschiedlicher Dichte gibt. Im Frühling sind die Unterschiede jedoch nicht signifikant. Offenbar hat die Jahreszeit einen Effekt auf die Menge gelegter Eier bei unterschiedlichen Schneckenarten.



Kapitel 7.

Einführung in die Zeitreihen

7.1. Einleitung

Wir beginnen dieses Kapitel¹ mit einem klassischen Beispiel einer *Zeitreihe*.

Beispiel 7.1.1

Die folgende Tabelle 7.1 zeigt die Passagierzahlen der Fluggesellschaft PAN AM (1927-1991) von 1949 bis 1960.

#	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
## 1949	112	118	132	129	121	135	148	148	136	119	104	118
## 1950	115	126	141	135	125	149	170	170	158	133	114	140
## 1951	145	150	178	163	172	178	199	199	184	162	146	166
## 1952	171	180	193	181	183	218	230	242	209	191	172	194
## 1953	196	196	236	235	229	243	264	272	237	211	180	201
## 1954	204	188	235	227	234	264	302	293	259	229	203	229
## 1955	242	233	267	269	270	315	364	347	312	274	237	278
## 1956	284	277	317	313	318	374	413	405	355	306	271	306
## 1957	315	301	356	348	355	422	465	467	404	347	305	336
## 1958	340	318	362	348	363	435	491	505	404	359	310	337
## 1959	360	342	406	396	420	472	548	559	463	407	362	405
## 1960	417	391	419	461	472	535	622	606	508	461	390	432

Table 7.1.: Passagierzahlen in Tausenden

Diese Daten sind in dieser Auflistung etwas unübersichtlich, da die Entwicklung der Passagierzahlen nicht klar ersichtlich ist. Dass die Fluggesellschaft 1960 mehr Passagiere hatte als 1949 ist nicht weiter überraschend. Aber wie verlief die Steigerung dieser Zahlen über die Jahre konkret?

¹Dieses Kapitel folgt Kapitel 1 des Lehrbuches *Introductory Time Series with R* by Paul S.P. Cowpertwait and Andrew V. Metcalfe, Springer 2009.

Kapitel 7. Einführung in die Zeitreihen

Wir wollen zunächst die Daten graphisch auf eine übersichtliche Art und Weise darstellen (siehe Abbildung 7.1).

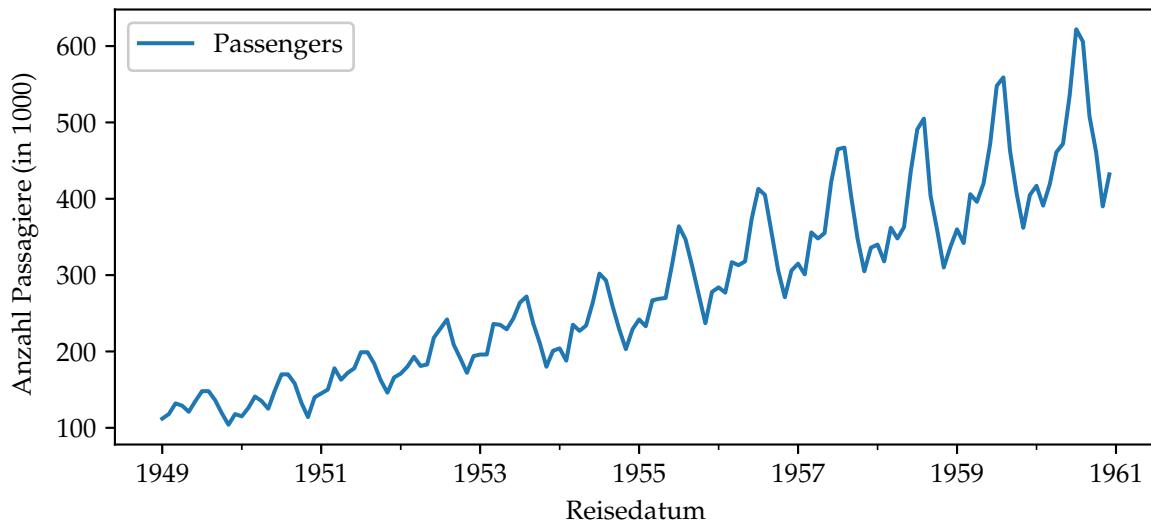


Abbildung 7.1.: Plot der Passagierzahlen aus Tabelle 7.1

Aufgrund der Darstellung in Abbildung 7.1 erkennen wir folgende Muster:

- Die Passagierdaten steigen jährlich. Wir sprechen in diesem Fall von einem *Trend*.
- Innerhalb eines Jahres gibt es auch Unterschiede. In der Ferienzeit wurde mehr geflogen als während des Rests des Jahres. Wir sprechen in diesem Fall von *Saisonalität*.
- Es fällt auf, dass die Beobachtungen *nicht* unabhängig voneinander sind. Benachbarte Werte sind ähnlich. Wir sprechen von *serieller Korrelation*.

Dieser Datensatz wurde ursprünglich dazu verwendet, um zukünftige Passagierzahlen vorherzusagen, damit der Kauf von Flugzeugen und die Nachfrage von Flugpersonal durch die Fluggesellschaft geplant werden konnte.

□

Viele reale Messungen und Datenerfassungen resultieren in Datenmengen, die seriell korrelieren. Einige Beispiele solcher Situationen:

Kapitel 7. Einführung in die Zeitreihen

- Temperatur, Druck, akustische Emissionen, Vibrationen, usw., die von Motoren, Generatoren, Kompressoren, usw. erzeugt werden und im Rahmen von Überwachung aufgezeichnet werden. Diese Größen ändern sich nicht schlagartig. Die Daten korrelieren seriell.
- Börse: Aktienpreise, Wechselkurse, usw. werden am Ende eines Handelstages aufgezeichnet. Diese ändern sich zwar zufällig von Tag zu Tag, aber in der Regel nicht extrem (ausser in Ausnahmesituationen).
- Umweltbeobachtungen: Temperaturen, Feuchtigkeit, Pollenkonzentration, Verschmutzung, Niederschläge, die bei einer bestimmten Wetterstation aufgezeichnet wurden. Die Temperatur von heute (z.B. 20 °C in Luzern) wird sich nicht (oder kaum) auf –10 °C morgen senken. Sie wird wohl bei um die 20 °C bleiben.
- Bundesamt für Statistik: Bevölkerungsgröße, Einkommen, Anzahl Unfälle, ändern sich nicht extrem von Jahr zu Jahr.

Diese Art von Daten nennt man *Zeitreihen*. Gewöhnlich gibt es mehrere Ziele, die man mit Zeitreihen erreichen möchte:

- *Deskriptive Analyse*:

Mit Hilfe von Übersichtsstatistiken und Visualisierungen können wir die grundlegenden Eigenschaften von Zeitreihen verstehen.

- *Modellierung und Interpretation*:

Durch die Modellierung des Prozesses, welcher einer Zeitreihe zugrundeliegt, erhalten wir ein tieferes Verständnis. Aufgrund eines Modells können Vertrauensintervalle und statistische Tests erstellt werden. Ebenso wird häufig die sequentielle Abhängigkeit einer Zeitreihe quantifiziert.

- *Zerlegung*:

Die hauptsächlichen Eigenschaften einer Zeitreihe sind einerseits *Saisonalität*, insbesondere periodische Muster, die in den Daten direkt mit der Zeitachse korrelieren. Auf der anderen Seite haben wir einen *Trend*, eine allmähliche Änderung des Mittelwertes der Zeitreihe. Ein wichtiges Ziel der Zeitreihenanalyse ist es, diese beiden Effekte zu trennen und eine entsprechende Zerlegung der Zeitreihe anzustreben.

- *Vorhersage*:

Mit Hilfe eines Modelles können wir zukünftige Werte einer Zeitreihe vorhersagen.

- *Regression:*

Oft versucht man, eine Zeitreihe (*Zielvariable*) durch mehrere andere Zeitreihen (*Prädiktoren*) zu erklären. Diese Idee ist auch in industriellen Anwendungen weitverbreitet. So kann es in einem Multi-Sensor-Setup vorkommen, dass ein entweder teurer oder schwierig zu installierender Sensor durch ein Modell ersetzt wird, in welchem die Werte von anderen Sensoren verwendet werden. Dies wird *virtual sensoring* oder aber *soft sensoring* genannt.

In diesem Modul werden wir die ersten 4 Punkte behandeln. Bevor wir in die mathematischen Konzepte von Zeitreihen eintauchen, wollen wir einige typische Beispiele von Zeitreihen betrachten.

7.2. Beispiele

Auf das Beispiel mit den Flugpassagieren aus Beispiel 7.1.1 werden wir noch einige Male zu sprechen kommen. Hier weitere Beispiele:

Beispiel 7.2.1

Das Kyoto Protokoll ist ein Zusatz zum Rahmenübereinkommen der Vereinten Nationen über Klimaveränderungen. Es ist am 11. Dezember 1997 beschlossen worden und am 16. Februar 2005 in Kraft getreten.

Die Argumente für die Reduzierung der Treibhausgase hängen von einer Kombination aus Erkenntnissen beruhend Wissenschaft, Wirtschaft und Zeitreihenanalyse ab. Entscheidungen, die in den nächsten Jahren gemacht werden, haben Einfluss auf die Zukunft unseres Planeten.

Abbildung 7.2 zeigt die jährliche Emission vom Treibhausgas CO₂ in der Schweiz von 1858 bis 2013².

Es ist deutlich, dass keine saisonalen Effekte vorhanden sind, da jährliche Durchschnitte ausgewertet wurden. Aber ein eigenartiger Trend ist erkennbar. Die Emissionen haben nach dem 2. Weltkrieg zwischen 1950 und 1970 stark zugenommen.

Im gleichen Sinne und ähnlich besorgniserregend sind die Daten zur globalen Temperaturerwärmung. In Abbildung 7.3 sind die jährlichen durchschnittlichen Temperaturanomalien bezüglich des Mittels zwischen 1910 und 2000 in Europa aufgezeichnet³.

²Daten aus CAIT Climate Data Explorer. 2017. Washington, DC: World Resources Institute. Erhältlich online unter: <http://cait.wri.org>

³Daten aus NOAA National Centers for Environmental information, Climate at a Glance: Global Time Series, veröffentlicht März 2017, abgerufen am 16 April, 2017 von <http://www.ncdc.noaa.gov/cag/>

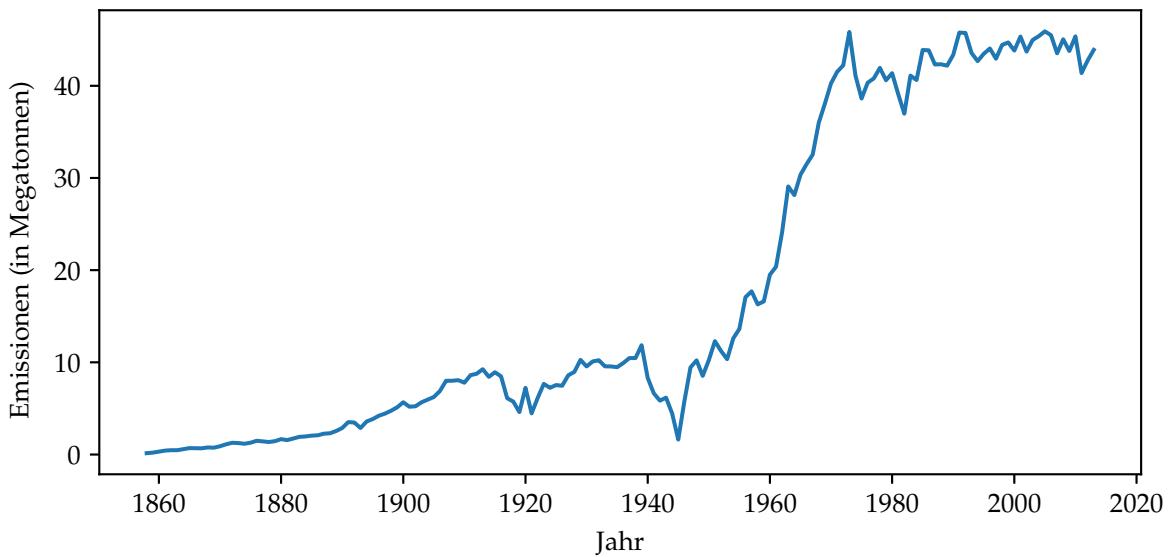


Abbildung 7.2.: CO₂-Emissionen in der Schweiz.

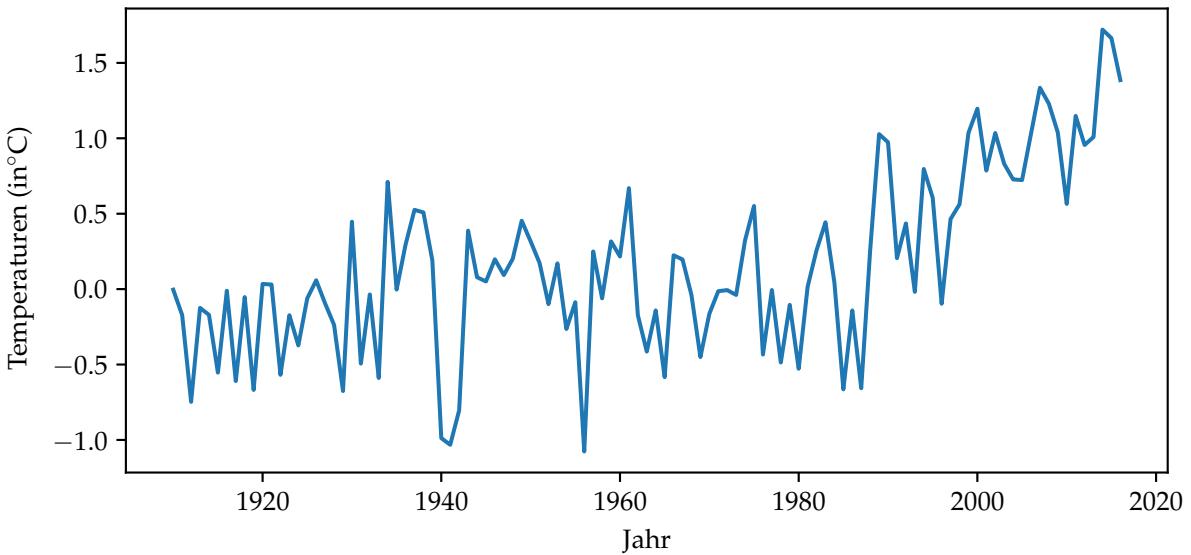


Abbildung 7.3.: Jährliche Temperaturenanomalien in Europa bezüglich des Durchschnittes zwischen 1910 und 2000.

Es gibt einen Aufwärtstrend, der um 1980 beginnt. Die Zeitreihe sagt allerdings nichts über die *Gründe* der Temperaturzunahme aus. Halten wir uns aber Abbildung 7.2 vor Augen, so ist eine Korrelation zwischen dem CO₂-Ausstoss und der globalen

Temperaturerwärmung unbestreitbar. Das heisst aber *nicht*, dass es einen kausalen Zusammenhang geben muss.

□

Beispiel 7.2.2

Der Datensatz [AirQualityUCI](#) enthält 9358 Fälle von stündlichen Messungen aus einer Reihe von 5 Metalloxid Sensoren, die in einem Multisensorgerät für chemische Luftqualität eingebettet sind⁴.

Dieses Gerät wurde auf Strassenhöhe in einer extrem verschmutzten Gegend einer italienischen Stadt aufgestellt. Die Daten wurden vom März 2004 bis Februar 2005 aufgezeichnet. Insgesamt wurden 13 verschiedene Größen gemessen.

In Abbildung 7.4 sind die Konzentration von Benzol (C_6H_6), die Lufttemperatur und die relative Luftfeuchtigkeit (in °C) an einem Tag aufgezeichnet.

Die Temperatur ist in der Nacht am tiefsten und am frühen Nachmittag am grössten, was nicht überraschend ist. Bei der Luftfeuchtigkeit ist es gerade umgekehrt. Die Benzolkonzentration ist um 9 und 18 Uhr am grössten. Diese Zeiten entsprechen den Rushhours. Dies sind Anzeichen einer Saisonalität. In Abbildung 7.5 sind die Werte über eine zweiwöchige Periode aufgezeichnet.

□

Beispiel 7.2.3

Sehr typische Beispiele für Zeitreihen sind Aktienindizes, Wechselkurse, usw. in der Wirtschaft. Aktienindizes werden oft analysiert und für Vorhersagen verwendet. Es ist allerdings unmöglich, den Trend eines Aktienkurses vorherzusagen. In diesem Beispiel betrachten wir den Aktienkurs von Tesla (siehe Abbildung 8.1).

Die Tagesabschlüsse betreffen 1112 aufeinanderfolgende Handelstage, begonnen am 19 Oktober 2012. Wie wir sehen, nahm der Aktienindex von Tesla beginnend am März bis Juni 2013 sehr stark zu. Um den Februar 2016 gab es einen (vorübergehenden) Zusammenbruch des Kurses, der wiederum von einem starken Anstieg gefolgt wurde. Vergleichen wir die Trends mit den Ankündigungen von Tesla, so scheint der Anstieg mit der Ankündigung des Models 3 im April 2016 zu korrelieren.

⁴S. De Vito et al., *On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario*, Sensors and Actuators B: Chemical, 129(2), Pages 750-757.

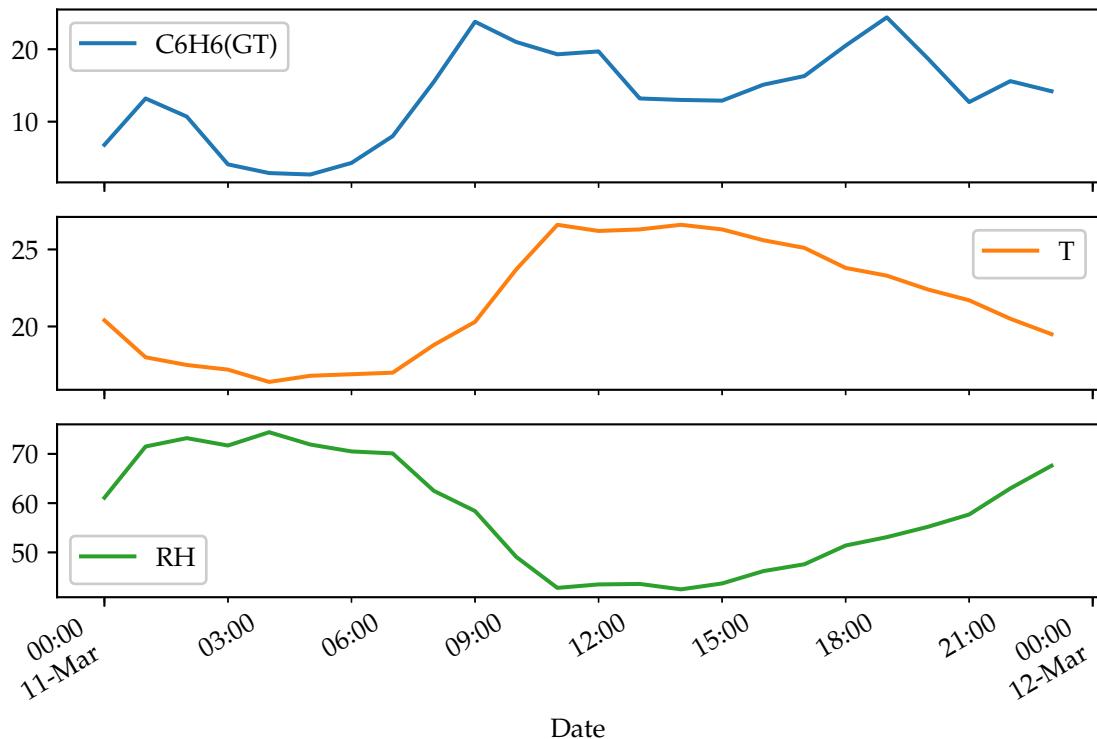


Abbildung 7.4.: Städtische Luftqualitätsmessungen am 11. März 2004

Anstatt des blossen Aktienkurses können wir auch den sogenannten *log-return* angeben. Dies sind die Veränderungen des Logarithmus des Index von Tag zu Tag. Der log-return für den Tesla-Aktienkurs ist in Abbildung 7.7 aufgezeichnet.

Die log-returns sind eine Näherung der *relativen* Änderung (in Prozent) bezüglich des vorhergehenden Handelstages (mehr dazu in Beispiel 7.4.3). Es ist offensichtlich, dass kein Trend mehr vorhanden ist. Wir werden sehen, dass die Daten unkorreliert sind. Als Konsequenz sind Vorhersagen des log-returns, die auf historischen Daten basieren, nutzlos.

□

7.3. Zeitreihen mit pandas

Das Paket **pandas** eignet sich besonders gut, um Zeitreihen zu untersuchen.

Kapitel 7. Einführung in die Zeitreihen

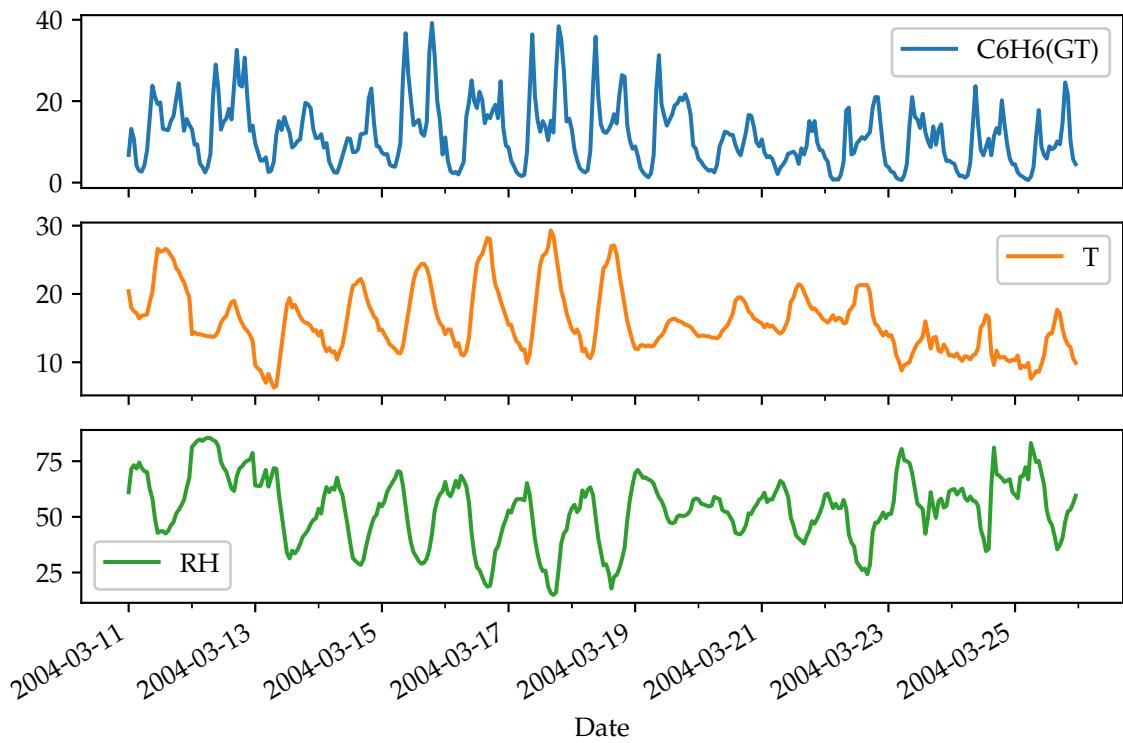


Abbildung 7.5.: Städtische Luftqualitätsmessungen am 11.-25. März 2004

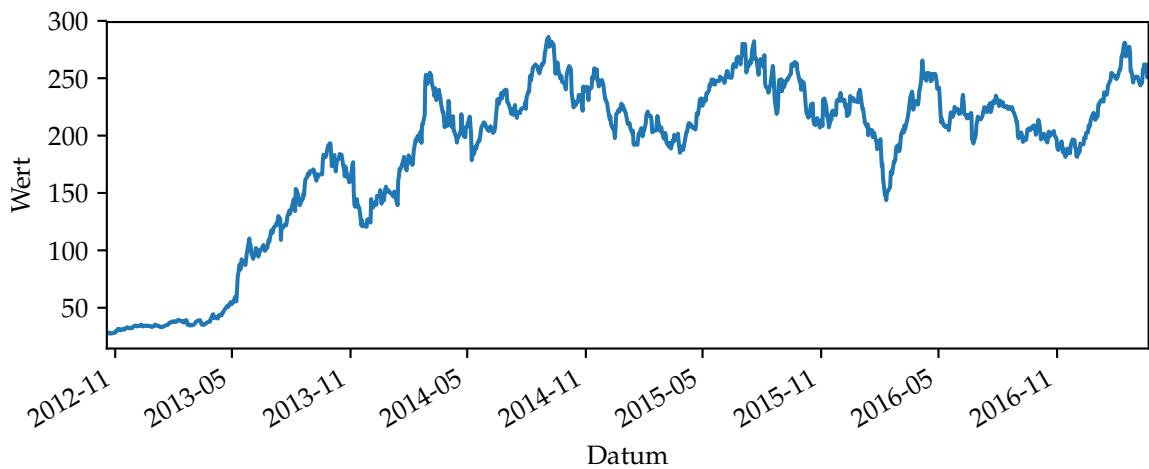


Abbildung 7.6.: Tagesabschlüsse des Tesla Aktienindex.

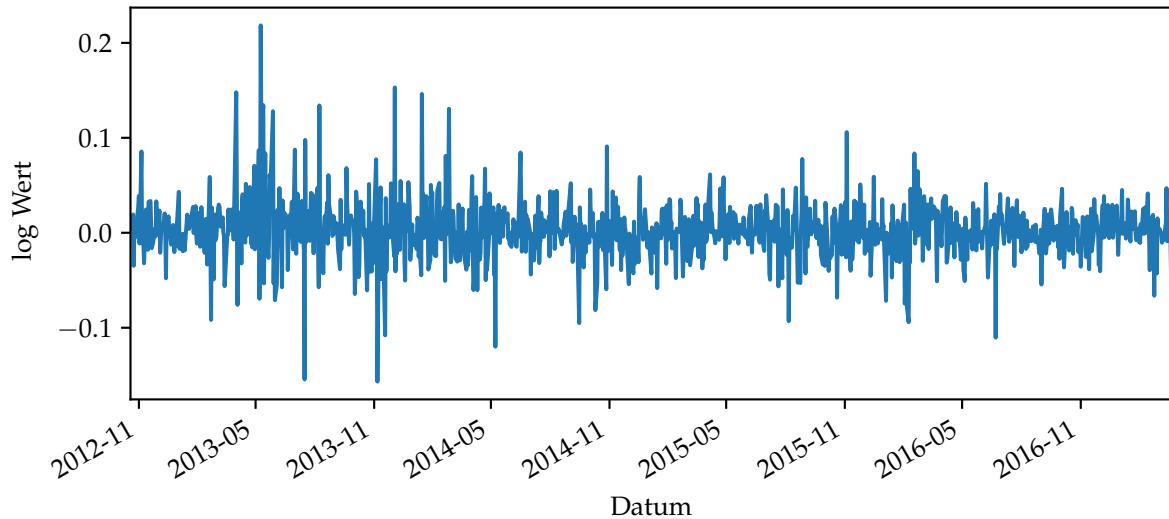


Abbildung 7.7.: Log-Return des Tesla Aktienkurses

Beispiel 7.3.1

Wir wollen das Beispiel 7.1.1 der Flugpassagiere auf **Python**-Befehle untersuchen. ([zu R](#))

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

AirP = pd.read_csv("AirPassengers.csv")

AirP.head()

AirP["TravelDate"] = pd.DatetimeIndex(AirP["TravelDate"])

AirP.set_index("TravelDate", inplace = True)

AirP.head()

AirP.plot()

plt.xlabel("Reisedatum")
plt.ylabel("Anzahl Passagiere (in 1000)")

plt.show()
```

Kapitel 7. Einführung in die Zeitreihen

Wir gehen den Code Schritt für Schritt durch. Zunächst lesen wir die Datei ein. Der Stern beim Einlesen der Datei steht für den Pfad, der angegeben werden muss. Mit `.head()` zum Beispiel können wir überprüfen, ob die Daten korrekt eingelesen wurden.

```
AirP = pd.read_csv("../ ../../Themen/Time_Series_Introduction/Skript_de/Daten/AirPassengers.csv")  
  
AirP.head()  
  
##      TravelDate  Passengers  
## 0    1/1/1949       112  
## 1    2/1/1949       118  
## 2    3/1/1949       132  
## 3    4/1/1949       129  
## 4    5/1/1949       121
```

Die Spalte `TravelDate` muss zuerst in ein Datumformat umgeformt werden, welches `pandas` versteht. Die geschieht mit dem Befehl `DatetimeIndex`

```
AirP[ "TravelDate" ] = pd.DatetimeIndex(AirP[ "TravelDate" ])
```

Nun wollen wir diese Daten noch als Index (Bezeichnung der Zeilen) übernehmen:

```
AirP.set_index("TravelDate", inplace = True)
```

Der Anfang der Tabelle sieht dann wie folgt aus:

```
AirP.head()  
  
##                  Passengers  
## TravelDate  
## 1949-01-01       112  
## 1949-02-01       118  
## 1949-03-01       132  
## 1949-04-01       129  
## 1949-05-01       121
```

Nun können wir die Zeitreihe graphisch darstellen [7.1](#):

```
AirP.plot()  
  
plt.xlabel("Reisedatum")  
plt.ylabel("Anzahl Passagiere (in 1000)")  
  
plt.show()
```



Beispiel 7.3.2

Wir untersuchen die vierteljährliche Bierproduktion in Australien (in Megaliter) zwischen März 1956 und Juni 1994⁵ (zu R)

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

AusBeer = pd.read_csv("*AustralianBeer.csv", sep = ";", header = 0)

AusBeer1 = AusBeer.copy()

AusBeer1.head()

AusBeer1["Quarter"] = pd.DatetimeIndex(AusBeer["Quarter"])

AusBeer1.set_index("Quarter", inplace = True)

AusBeer1.head()

AusBeer1.describe()
```

```
##      Quarter    megalitres
## 0    1956Q1      284.4
## 1    1956Q2      212.8
## 2    1956Q3      226.9
## 3    1956Q4      308.4
## 4    1957Q1      262.0
##                  megalitres
## Quarter
## 1956-01-01      284.4
## 1956-04-01      212.8
## 1956-07-01      226.9
## 1956-10-01      308.4
## 1957-01-01      262.0
##                  megalitres
## count    154.000000
## mean     408.267532
```

⁵Die Daten sind erhält unter <http://datamarket.com/data/list/?q=provider:tsdl> und wird vom australischen Amt für Statistik bereitgestellt.

Kapitel 7. Einführung in die Zeitreihen

```
## std      97.598588
## min     212.800000
## 25%    325.425000
## 50%    427.450000
## 75%    466.950000
## max    600.000000
```

Die Befehl `.describe()` listet der Reihe nach

- die Anzahl Werte
- das arithmetische Mittel
- die Standardabweichung
- den minimalen Wert
- das untere Quartil
- den Median
- das obere Quartil
- den maximalen Wert

auf.

Häufig möchte man blass eine Teilmenge der Zeitreihe auswählen, zum Beispiel die Zeit zwischen September 1980 und März 1994. In Abbildung 7.8 sehen wir das saisonale Verhalten, wo wir Peaks gegen Ende des Jahres haben. Dies entspricht dem australischen Sommer. Des weiteren stellen wir einen Langzeittrend bei der Produktionsssteigerung zwischen 1960 und der Mitte der 1970er Jahre und eine darauffolgende Stagnation. ([zu R](#))

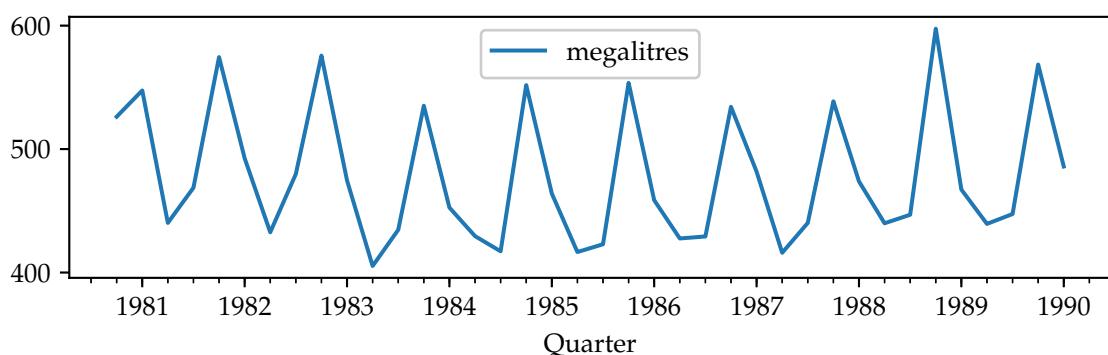


Abbildung 7.8.: Vierteljährliche Bierproduktion in Australien

Diese Auswahl erfolgt durch, wobei das amerikanische Zeitformat benutzt wird.

```
AusBeer1.loc["1980-9" : "1990-3", :].plot()
```



Multivariate Zeitreihen

Wir illustrieren hier einige wichtige Ideen und Konzepte der multivariaten Zeitreihen. Dabei werden verschiedene Zeitreihen über denselben Zeitraum gemeinsam betrachtet.

Beispiel 7.3.3

Der vierteljährliche Stromverbrauch (in Millionen kWh in Australien wird betrachtet und mit der schon bekannten vierteljährlichen Bierproduktion verglichen. (zu R)

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

AusBeer = pd.read_csv("*AustralianBeer.csv", sep = ";", header = 0)

AusEl = pd.read_csv("*AustralianElectricity.csv", sep = ";")

Aussie = AusBeer.copy()

# Hier wird der Datensatz um eine Spalte kilowatt erweitert
Aussie["kilowatt"] = AusEl["kilowatt"]

Aussie["Quarter"] = pd.DatetimeIndex(Aussie["Quarter"])

Aussie.set_index("Quarter", inplace = True)

Aussie.plot(subplots = True)

plt.show()
```

Die Plots in Abbildung 7.9 zeigen einen wachsenden Trend für beide Größen, die unter anderem wohl auf die steigende Bevölkerungszahl in Australien von ungefähr 10 Millionen auf ungefähr 18 Millionen über die gleiche Periode zurückzuführen ist.

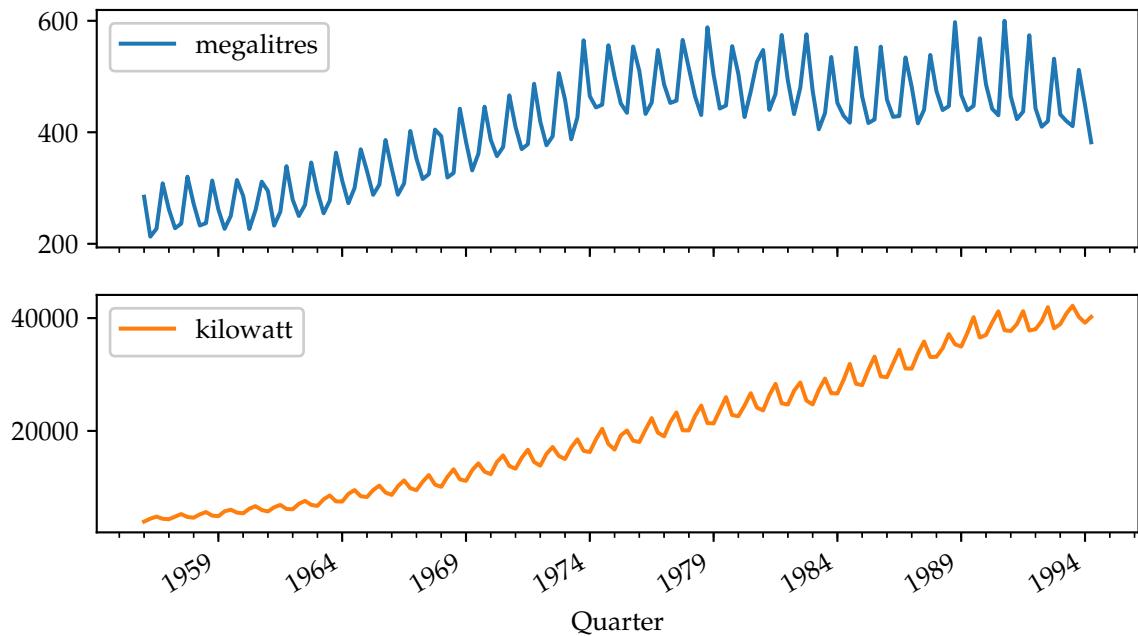


Abbildung 7.9.: Bier- und Elektrizitätsverbrauch in Australien von 1959-1994

Es ist allerdings beachtenswert, dass die Produktion der Elektrizität um den Faktor 7 zunahm, während sich die Bevölkerung kaum verdoppelt hat.

□

7.4. Elementare Transformationen, Visualisierung und Zerlegung von Zeitreihen

Die Analyse von Zeitreihen beginnt in der Regel mit der Beschreibung mittels Kennzahlen, Transformation und Visualisierung der Daten. Dies ist *keine* Modellbildung der Daten, und damit können wir auch noch keine richtiggehenden Vorhersagen machen oder Vertrauensintervalle bilden. Allerdings kann man mit diesen Techniken wichtige Einsichten und ein tieferes Verständnis der Daten gewinnen.

In diesem Abschnitt werden wir die wichtigsten Datentransformationen bezüglich Zeitreihen beschreiben. Dazu stellen wir einen Werkzeugkasten mit den wichtigsten Visualisierungen bereit, um Zeitreihen zu untersuchen. Schliesslich werden wir Zeitreihen in Trend, saisonale sowie irreguläre Komponenten zerlegen. Insbesondere

lernen wir am Ende des Kapitels das STL-Verfahren zur Zerlegung einer Zeitreihe kennen.

7.4.1. Datentransformationen

In vielen Situationen ist es wünschenswert oder sogar notwendig, Zeitreihen zu transformieren, bevor wir Modelle zur Vorhersage erstellen und für neue Daten anwenden. Insbesondere machen viele Methoden folgende Annahmen:

- *Normalverteilung* oder zumindest *symmetrische Verteilung* der Daten
- einen *linearer Trend* zwischen den Daten und der Zeit
- Eine zeitlich *konstante Varianz*

So ist es beispielsweise für sehr schiefe oder heteroskedastische (nicht konstante Varianz) Daten oft besser, nicht die ursprüngliche Zeitreihe

$$\{x_1, x_2, \dots\}$$

zu verwenden, sondern die Zeitreihe zu transformieren:

$$\{g(x_1), g(x_2), \dots\}$$

Eine Familie von Transformationen, die besonders geeignet ist, um Schiefe und Varianz zu korrigieren, sind die *Box-Cox-Transformationen*:

Box-Cox-Transformationen

Für eine Zeitreihe mit positiven Werten

$$\{x_1, x_2, \dots\}$$

sind die Box-Cox-Transformationen definiert durch

$$g(x) = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \text{falls } \lambda \neq 0 \\ \log(x) & \text{falls } \lambda = 0. \end{cases}$$

Ziel ist es, den Parameter λ , so zu wählen, dass die gewünschten Eigenschaften erfüllt sind. Wir illustrieren dies an den **AirPassengers** Daten.

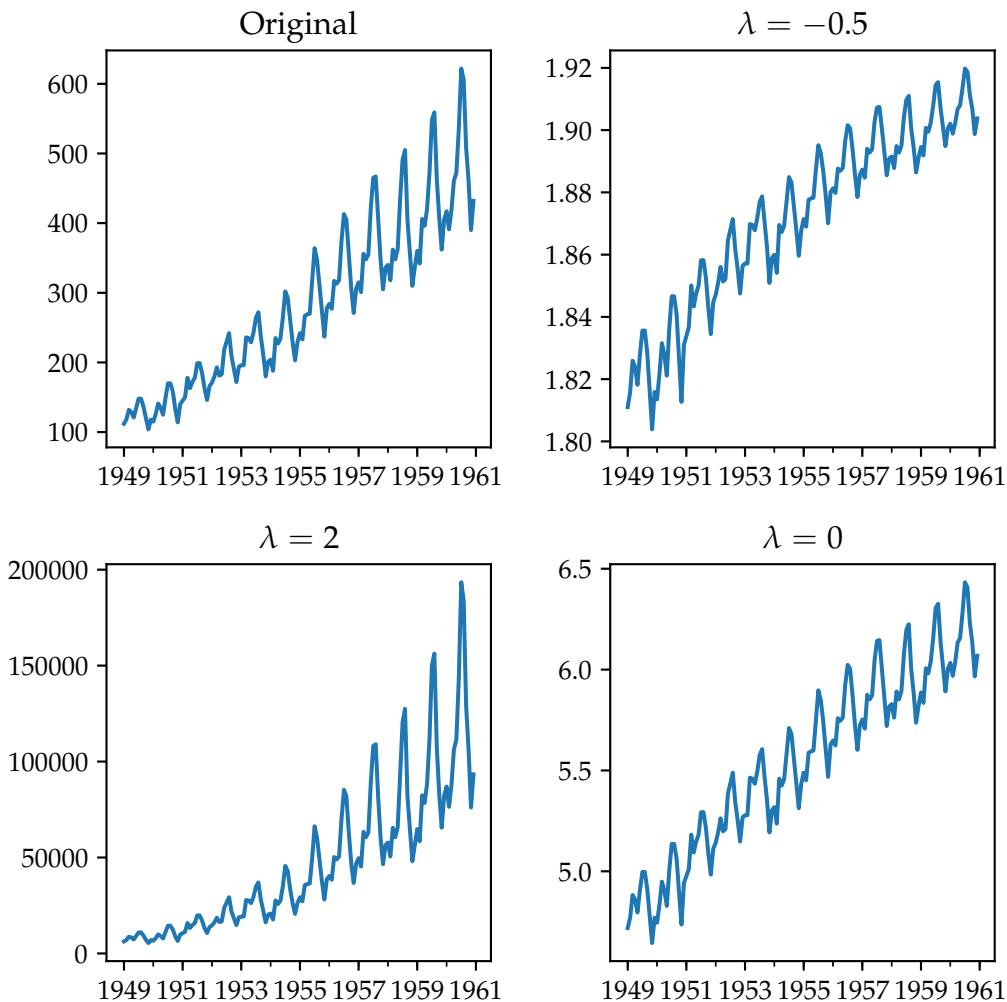


Abbildung 7.10.: Box-Cox-Transformationen für verschiedene Werte für λ .

Beispiel 7.4.1

Wir wenden Box-Cox-Transformationen für verschiedene Werte von λ auf den Datensatz **AirPassengers** an. In Abbildung 7.10 ist dies dargestellt.

Wir sehen, dass in den ursprünglichen Daten ein Trendeffekt und Saisonalität offensichtlich zu Tage tritt. Die Intensität des saisonalen Einflusses (d.h. die Varianz) ist mit den Jahren aber zunehmend. Für $\lambda = 0$ erhalten wir ein stabiles Bild: wir haben einen linearen Trend und einen homogenen saisonalen Effekt. (zu R)

```
def boxcox(x, lambd):
    return np.log(x) if (lambd==0) else (x**lambd-1)/lambd

AirP["l_2"] = boxcox(AirP["Passengers"], 2)
```

Kapitel 7. Einführung in die Zeitreihen

```
AirP["l_0"] = boxcox(AirP["Passengers"], 0)
AirP["l_-05"] = boxcox(AirP["Passengers"], -0.5)

plt.subplot(221)
AirP["Passengers"].plot()
plt.title("Original")
plt.xlabel("")

plt.subplot(222)
AirP["l_-05"].plot()
plt.title("lambda = -0.5")
plt.xlabel("")

plt.subplot(223)
AirP["l_2"].plot()
plt.title("lambda = 2")
plt.xlabel("")

plt.subplot(224)
AirP["l_0"].plot()
plt.title("lambda = 0")
plt.xlabel("")

plt.show()
```

□

Die Box-Cox-Familie von Transformationen kommt einer Modifikation der *Werte* einer Zeitreihe gleich. Manchmal ist es aber notwendig, die *Zeitachse* zu transformieren. Wir werden nur das einfachste Beispiel einer Zeittransformation, nämlich die *Zeitverschiebung* oder das sogenannte *shifting* kennenlernen. Eine allgemeinere Version einer Zeitverschiebung wird oft bei Spracherkennung verwendet (Audiosigale sind nur eine spezielle Form von Zeitreihen), die *warping* genannt wird.

Zeitverschiebungstransformation (time-shift)

Sei

$$\{x_1, x_2, \dots\}$$

eine Zeitreihe.

1. Die Zeitverschiebung durch einen *lag* von $k \in \mathbb{Z}$ ist definiert durch

$$g(x_i) = x_{i-k}$$

2. Für den Spezialfall $k = 1$ heisst die Zeitverschiebung *backshift*

$$B(x_i) = x_{i-1}$$

Anders ausgedrückt führt eine Zeitverschiebung dazu, dass wir k Schritte in der Zeitreihe zurückgehen (falls $k > 0$) oder k Schritte vorwärts (falls $k < 0$).

Beispiel 7.4.2

Für den Datensatz **AirPassengers** wenden wir die Zeitverschiebung für verschiedene Werte für k an. Dazu nehmen wir die **shift**-Funktion von **pandas** in Anspruch (siehe Abbildung 7.11). (zu **R**)

Die **shift**-Funktion funktioniert allerdings gerade umgekehrt, wie die übliche Definition der *lag*-Funktion. Für einen Vorwärtsshift müssen wir positive und für einen Rückwärtsshift negative Werte eingeben.

```
AirP["s_4"] = AirP["Passengers"].shift(-4)
AirP["s_-5"] = AirP["Passengers"].shift(5)

AirP["Passengers"].plot()
AirP["s_4"].plot()
AirP["s_-5"].plot()

plt.legend(["Original", "zurückverschoben", "vorverschoben"])
plt.show()
```

□

Die Rückwärtszeitverschiebung wird angewendet, falls wir *Differenzen* von Zeitreihen berechnen, da

$$x_i - x_{i-1} = x_i - B(x_i)$$

Insbesondere wird oft das Berechnen von Differenzen mit Box-Cox-Transformationen kombiniert.

Beispiel 7.4.3

Die *log-returns* einer (finanziellen) Zeitreihe sind definiert durch

$$y_i = \log(x_i) - \log(x_{i-1}) = \log\left(\frac{x_i}{x_{i-1}}\right) = \log\left(\frac{x_i - x_{i-1}}{x_{i-1}} + 1\right) \approx \frac{x_i - x_{i-1}}{x_{i-1}}$$

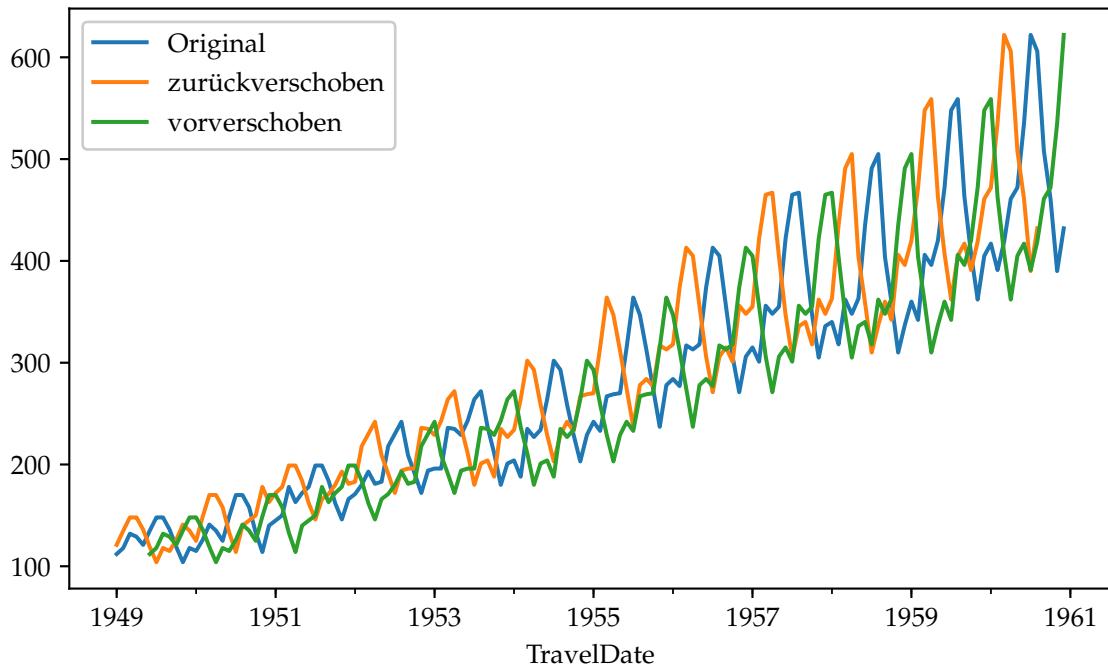


Abbildung 7.11.: Zeitverschiebung für $k = 4$ und $k = -5$

Die letzte Gleichung folgt aus der Taylor-Reihen Entwicklung der Logarithmus:

$$\log(s+1) = s - \frac{s^2}{2} + \dots$$

Mit anderen Worten nähert die log-return Zeitreihe y_i den relativen Anstieg der Zeitreihe von x_{i-1} bis x_i zu jedem Zeitpunkt an.

Diese Grösse wird oft für finanzielle Anwendungen studiert: Die ursprüngliche Reihe

$$\{x_1, x_2, \dots\}$$

könnte ein scheinbar signifikantes Muster aufweisen, aber die Reihe

$$\{y_1, y_2, \dots\}$$

ist oft sehr zufällig. Das heisst, die Änderungen sind von Tag zu Tag zufällig.

Im Beispiel 7.2.3 haben wir den Aktienindex von Tesla mit dem zugehörigen log-return untersucht. Abbildung 8.1 zeigt den Aktienindex, die Abbildung 7.7 den log-return. Die zweite Abbildung schaut sehr zufällig aus trotz der gelegentlichen starken Fluktuationen, die sehr typisch für diese Art von Daten sind. Analysten

(und andere) versuchen die Wartezeit zwischen solchen Fluktuationen zu modellieren. Das heisst, sie möchten zumindest mit einer gewissen Wahrscheinlichkeit den nächsten Peak vorhersagen.

□

7.4.2. Visualisierungen

Ein sehr wichtiger Teil der deskriptiven Statistik ist die geeignete Visualisierung einer Datenmenge (und das gilt nicht nur Zeitreihen). Der *Zeitreihenplot* ist normalerweise der erste Schritt in der Analyse von Zeitreihen. Wir haben schon einige Beispiele von Zeitreihenplots im Abschnitt 7.1 besprochen. Wir haben beobachtet, dass diskrete Zeitpunkte durch Linien verbunden werden, obwohl sie in der Tat nicht stetig sind. Dies macht `.plot()` von `pandas` automatisch.

Wenn wir eine Zeitreihe plotten, dann hängt die Interpretierbarkeit stark von der Anzahl und der Glattheit der Daten ab. Zeitreihen über grosse Zeiträume und vielen Datenpunkten sollten entsprechend aufgeteilt oder zusammengefasst werden. Dies soll im folgenden Beispiel illustriert werden.

Beispiel 7.4.4

In diesem Beispiel führen die Luftqualität aus Beispiel 7.2.2 weiter aus. Wir haben dort stündliche Messungen von verschiedenen Sensoren, wobei wir uns nur auf die Temperatur konzentrieren. Die Abbildung 7.12 zeigt den vollständigen Plot. ([zu R](#))

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

AirQ = pd.read_csv("AirQualityUCI.csv", sep = ";", decimal = ",")

AirQ1 = AirQ.copy()

# pandas kennt das Zeitformat in der Tabelle nicht:
# Punkt muss durch - ersetzt werden
AirQ1["Time"] = AirQ1["Time"].str.replace(".", "-")

AirQ1["Date"] = pd.DatetimeIndex(AirQ1["Date"] + " " + AirQ1["Time"])

AirQ1.set_index("Date", inplace = True)

# Einige Wert der Temperatur sind -200. Diese Zeilen werden weggelassen
```

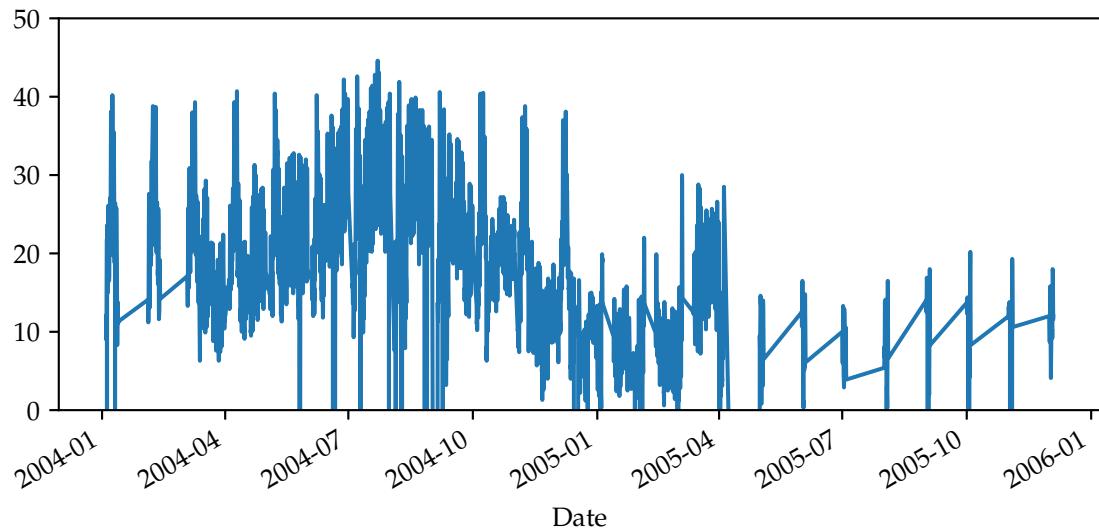


Abbildung 7.12.: Temperatur über das gesamte Intervall

```
AirQ1 = AirQ1[AirQ1["T"] > -20]
AirQ1["T"].plot()
plt.show()
```

In der Tabelle sind einige Werte mit -200 aufgeführt, die wir als Sensorfehler interpretieren können.

Wir fokussieren uns auf 20 Tage, damit wir das Verhalten der Temperatur in grösserem Detail betrachten können. Abbildung 7.13 zeigt diese Daten. ([zu R](#))

```
AirQ4 = AirQ1.loc["2004-3-10" : "2004-3-30", "T"]
AirQ4.plot()
```

□

Im nächsten Beispiel interessiert uns, wie wir eine Datenaggregation durch den `.boxplot()`-Befehl visualisieren können.

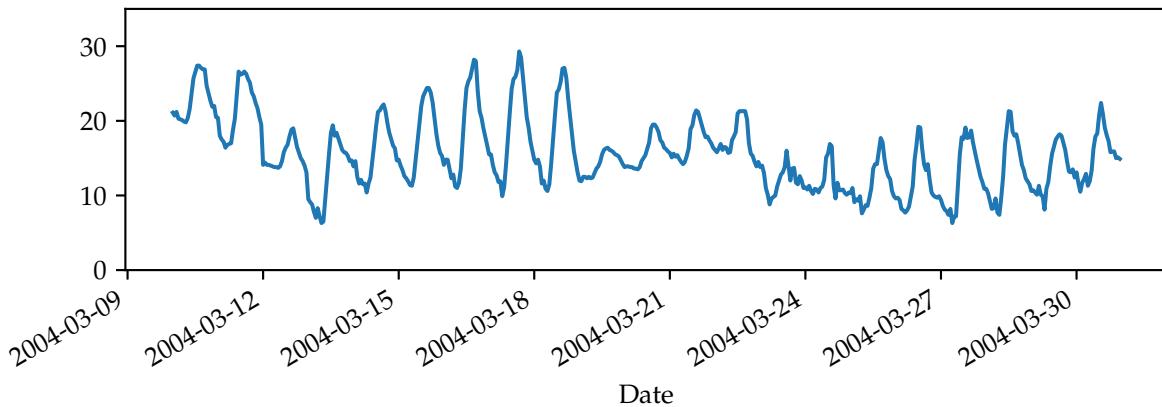


Abbildung 7.13.: Temperatur über 20 Tage im März

Beispiel 7.4.5

Wir betrachten nochmals die Daten am Ende von Beispiel 7.4.4. Die Zeitreihe **AirQ4** aus Beispiel 7.4.4 enthält die stündlichen Lufttemperaturen über einen Zeitraum von 20 Tagen im März 2004 in einer italienischen Stadt. Wir wollen nun die Daten für jede Stunde über diese 20 Tage betrachten. In Abbildung 7.14 sehen wir die Boxplots dieser Daten, die nach Stunde gruppiert sind. (zu R)

```
AirQ1.boxplot("T", by = "Time", rot = 45)
plt.show()
```

Die Option **by = "Time"** erreicht, dass für jede Stunde ein Boxplot erzeugt wird.

□

Ein nützlicher Ansatz, um graphisch die Korrelation von aufeinanderfolgenden Beobachtungen zu visualisieren, ist der *lagged scatterplot*. Dabei wird die ursprüngliche Zeitreihe gegen eine zeitverschobene Zeitreihe aufgezeichnet, also die Datenpunkte (x_i, x_{i-k}) . Dies wird in **pandas** mit der **lag_plot**-Funktion erreicht.

Beispiel 7.4.6

Wir betrachten wieder die über einen Zeitraum von 20 Tagen gemessene Lufttemperatur in einer italienischen Stadt. Wir wenden den lag-plot für $k = 1$ und $k = 10$ an. (zu R)

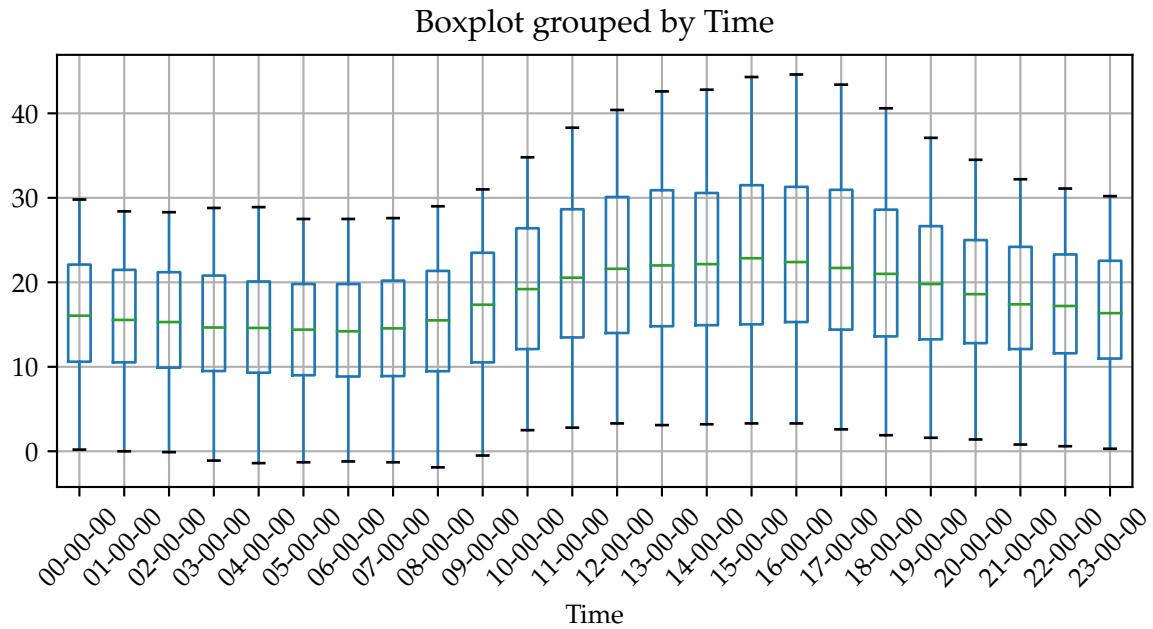


Abbildung 7.14.: Temperatur über 20 Tage im März gruppiert nach Stunden.

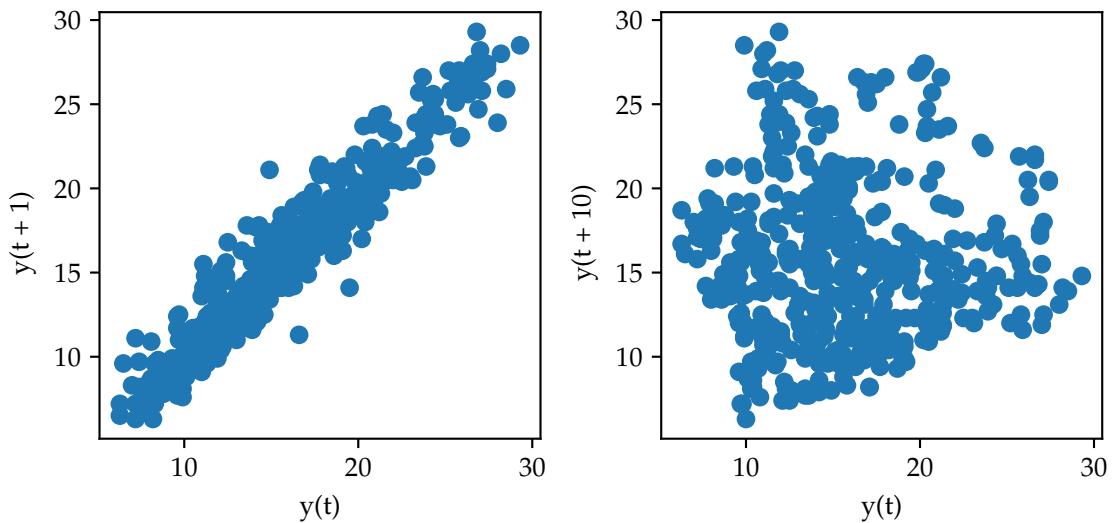


Abbildung 7.15.: Lagged scatterplots für $k = 1$ (links) und $k = 10$ (rechts)

```
from pandas.plotting import lag_plot
```

```
plt.subplot(121)
lag_plot(AirQ4)

plt.subplot(122)
lag_plot(AirQ4, 10)

plt.show()
```

Wir sehen in Abbildung 7.15, dass das Streudiagramm mit lag 1 ein lineares Muster zeigt. Dies deutet auf eine Korrelation zwischen benachbarten stündlichen Temperaturen hin. Dies war auch zu erwarten, da sich die Temperaturen innerhalb einer Stunde kaum dramatisch ändern.

Der lag von 10 Stunden zeigt ein ziemlich unspezifisches Streudiagramm. Hier ist keine Korrelation von Temperaturen vorhanden, die 10 Stunden auseinander liegen.

□

7.4.3. Zerlegung von Zeitreihen

Die ersten Beispiele in Abschnitt 7.1 zeigten, dass viele Zeitreihen dominiert werden durch einen Trend und/oder saisonale Effekte. Die Modelle in diesem Abschnitt basieren deswegen auf diesen Komponenten.

Ein einfaches additives Zerlegungsmodell ist gegeben durch

$$x_k = m_k + s_k + z_k$$

wobei

- k der Zeitindex
- x_k die beobachteten Daten
- m_k der Trend
- s_k der saisonale Effekt
- z_k ein Fehlerterm

sind. Der Fehlerterm ist im Allgemeinen eine Folge aus *korrelierten* zufälligen Variablen mit Mittelwert 0.

In diesem Abschnitt wollen wir die beiden wichtigsten Ansätze umreissen, wie wir den Trend m_k und die saisonalen Effekte abschätzen können. Wie wir beim Datensatz

AirPassengers in Beispiel 7.1.1 gesehen haben, nimmt der saisonale Effekt mit dem Trend zu. In diesem Fall ist ein multiplikatives geeigneter als ein lineares Modell :

$$x_k = m_k \cdot s_k + z_k$$

Wenn das Rauschen auch noch multiplikativ ist

$$x_k = m_k \cdot s_k \cdot z_k$$

dann ist der Logarithmus von x_k wieder linear

$$\log(x_k) = \log(m_k) + \log(s_k) + \log(z_k)$$

Bewegendes Mittel (moving average)

Der *moving average filter* ist eine sehr einfache Methode, den Trend m_k unter dem saisonalen Effekt s_k abzuschätzen:

Moving average filter

Sei

$$\{x_1, x_2, \dots, x_n\}$$

eine Zeitreihe und $p \in \mathbb{N}$.

Dann ist der *moving average filter* der Länge p definiert durch:

- Falls p ungerade, dann $p = 2l + 1$ und die gefilterte Folge ist definiert durch:

$$g(x_i) = \frac{1}{p}(x_{i-l} + \dots + x_i + \dots + x_{i+l})$$

- Falls p is gerade, dann $p = 2l$ und die gefilterte Folge ist definiert durch:

$$g(x_i) = \frac{1}{p} \left(\frac{1}{2}x_{i-l} + x_{i-l+1} + \dots + x_i + \dots + x_{i+l-1} + \frac{1}{2}x_{i+l} \right)$$

Der Wert p wird bezeichnet als *Fensterbreite* oder *window width*.

In anderen Worten ersetzt der moving average filter den i -ten Wert der Zeitreihe durch den Mittelwert der nächsten p Nachbarn. Falls p ungerade ist, so ist das Fenster symmetrisch um x_i . Für einen gerades p konstruiert man ein Fenster der Länge $p + 1$ (was dann ungerade ist), zählt dann aber jeweils nur die Hälften an den Endpunkten zum Mittelwert

Falls eine Zeitreihe die Frequenz p hat (zum Beispiel $p = 12$ für monatliche Daten), dann kann die Trendkomponente der Zeitreihe durch einen moving average filter mit einer Fensterbreite p abgeschätzt werden. Da wir an jedem Zeitpunkt genau über eine ganze Periode mitteln, so verschwinden die saisonalen Effekte und was übrig bleibt, ist die Trendkomponente. Dies resultiert im Trendschätzer \hat{m}_k .

Beispiel 7.4.7

Für den Datensatz **AirPassenger** schätzen wir den Trend mit dem moving average filter. In **pandas** wird dies mittels des Befehls **rolling(window=12).mean()** (siehe Abbildung 7.18) (zu **R**)

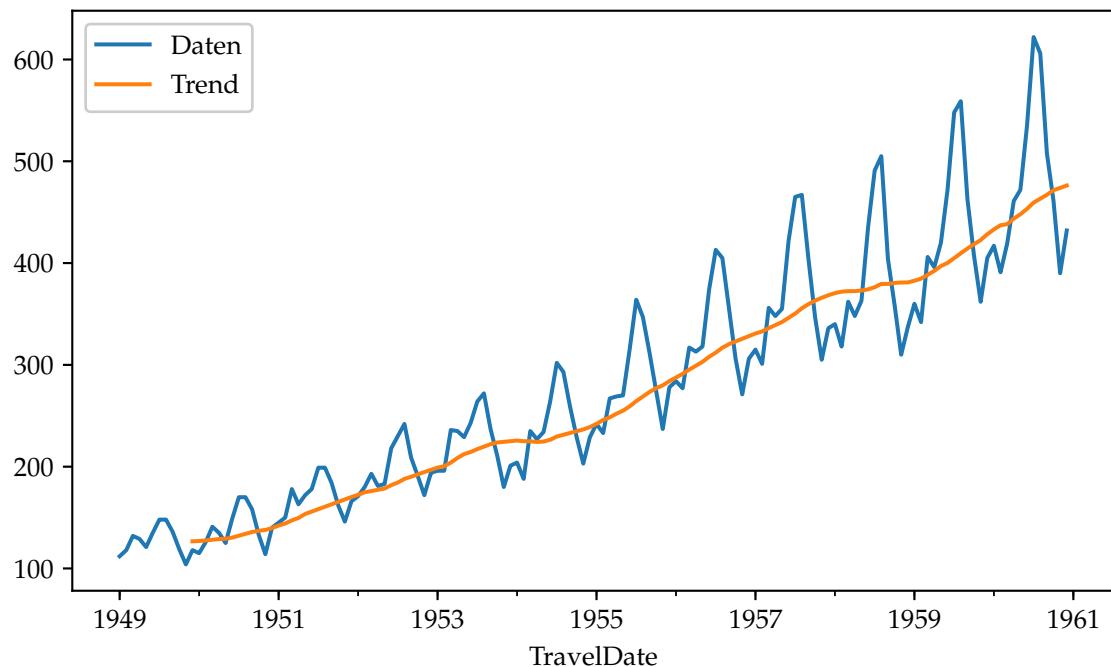


Abbildung 7.16.: Schätzung des Trends

```
AirP["Trend"] = AirP["Passengers"].rolling(window = 12).mean()

AirP["Passengers"].plot()

AirP["Trend"].plot()

plt.legend(["Daten", "Trend"])

plt.show()
```

Der geschätzte Trend \hat{m}_k zeigt keine saisonalen Fluktuationen.

□

Um den saisonalen additiven Effekt abzuschätzen, berechnen wir

$$\hat{s}_k = x_k - \hat{m}_k$$

Nun wird die Zeitreihe \hat{s}_k für jeden Punkt in einem Zyklus (Monat) gemittelt, und wir erhalten eine einzige Schätzung für jeden Zykluspunkt (Monat).

Beispiel 7.4.8

Wir verwenden wieder den Datensatz **AirPassengers**, und subtrahieren den geschätzten Trend von Beispiel 7.4.7 (siehe Abbildung 7.17)

```
AirP["Season"] = AirP["Passengers"] - AirP["Trend"]

AirP["Season"].plot()

plt.show()
```

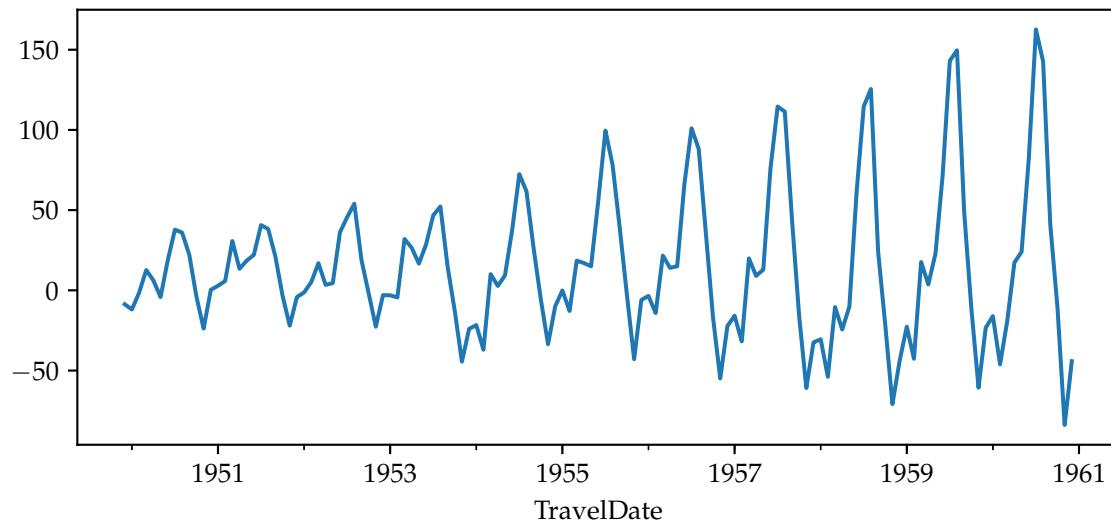


Abbildung 7.17.: Der saisonale Effekt resultiert durch Subtraktion des Trends von der ursprünglichen Zeitreihe.

Wir wollen nun die durchschnittliche Saisonalität bestimmen. Dazu wird der Mittelwert aller entsprechenden Monate berechnet (siehe Abbildung 7.18). (zu R)

Kapitel 7. Einführung in die Zeitreihen

```
# AirP["Season"] wird in eine Matrix umgewandelt
# mit den Monaten als Spalten (Jahre als Zeilen)

AirP2 = AirP["Season"].values.reshape((12, 12))

# Entlang der Spalten (axis=0) wird der Mittelwert genommen
# nanmean bedeutet, die NaN werden ignoriert

ave = np.nanmean(AirP2, axis = 0)

# Der Vektor ave wird verzwölft,
# damit er wieder die gleiche Länge hat, wie AirP["Season"]

AirP["Season_ave"] = np.tile(A = ave, reps = 12)

AirP["Season_ave"].plot()

plt.show()
```

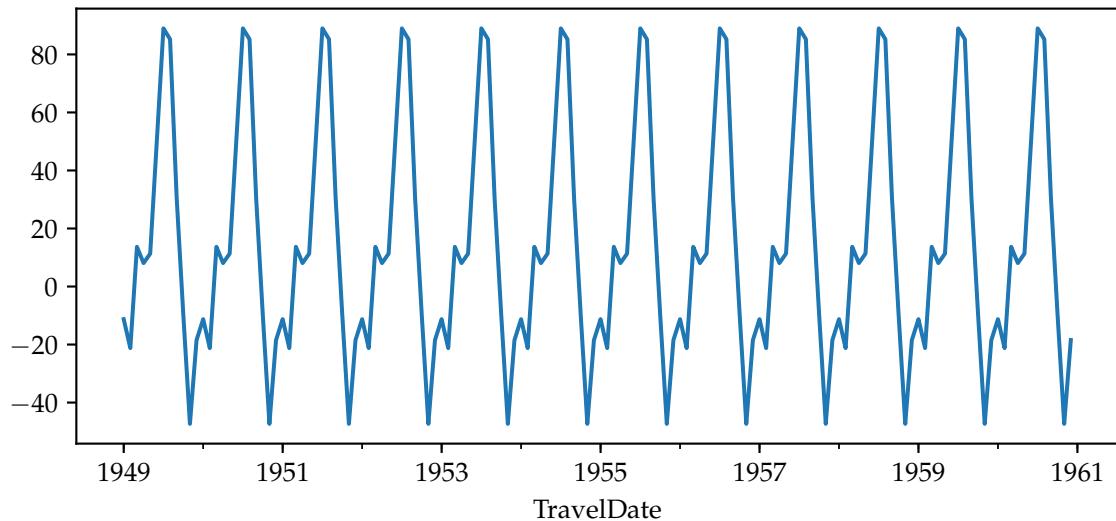


Abbildung 7.18.: Gemittelte Saisonalität \hat{s}_i

□

Schlussendlich subtrahieren wir die Schätzungen für Trend und Saisonalität von der ursprünglichen Zeitreihe und erhalten den Restterm (oder die *Residuen*)

$$\hat{r}_i = x_i - \hat{m}_i - \hat{s}_i$$

Kapitel 7. Einführung in die Zeitreihen

Der Restterm sollte aus (möglicherweise korrelierten) Zufallsvariablen ohne Struktur/Periodizität bestehen.

Beispiel 7.4.9

Wir verwenden wieder den Datensatz **AirPassengers** und subtrahieren den geschätzten Trend und Saisonalität von Beispiel 7.4.7 und 7.4.8 (siehe Abbildung 7.19) (zu R)

```
AirP["Residual"] = AirP["Season"] - AirP["Season_ave"]

AirP["Residual"].plot()

plt.show()
```

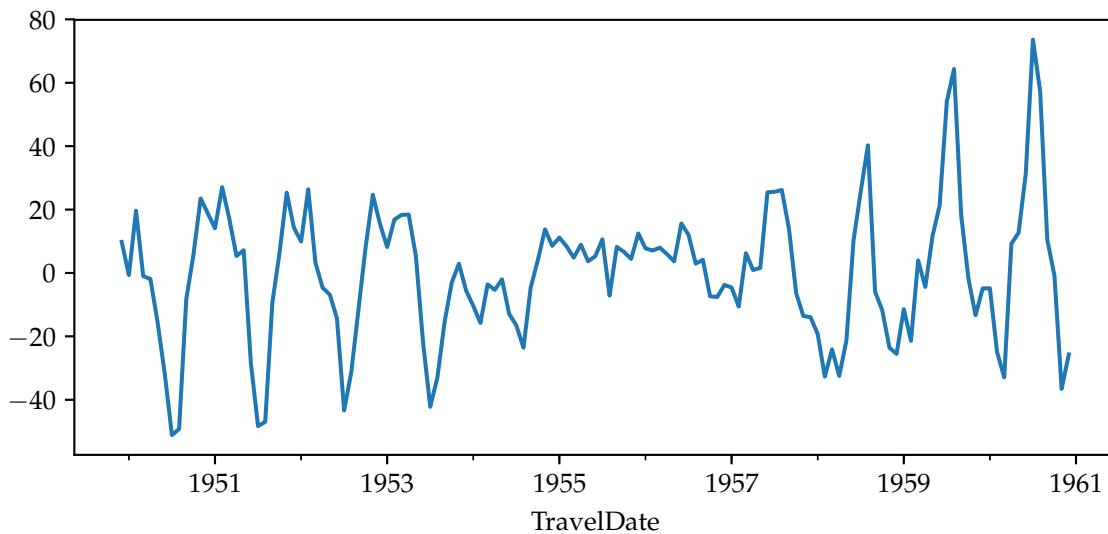


Abbildung 7.19.: Restterm (Residuen) \hat{r}

Die Abbildungen 7.10, 7.18 und 7.19 lassen sich mit einem einzigen Befehl erzeugen (siehe Abbildung 7.20) (zu R)

```
from statsmodels.tsa.seasonal import seasonal_decompose

seasonal_decompose(AirP["Passengers"], model = "additive", freq = 12).plot()

plt.show()
```

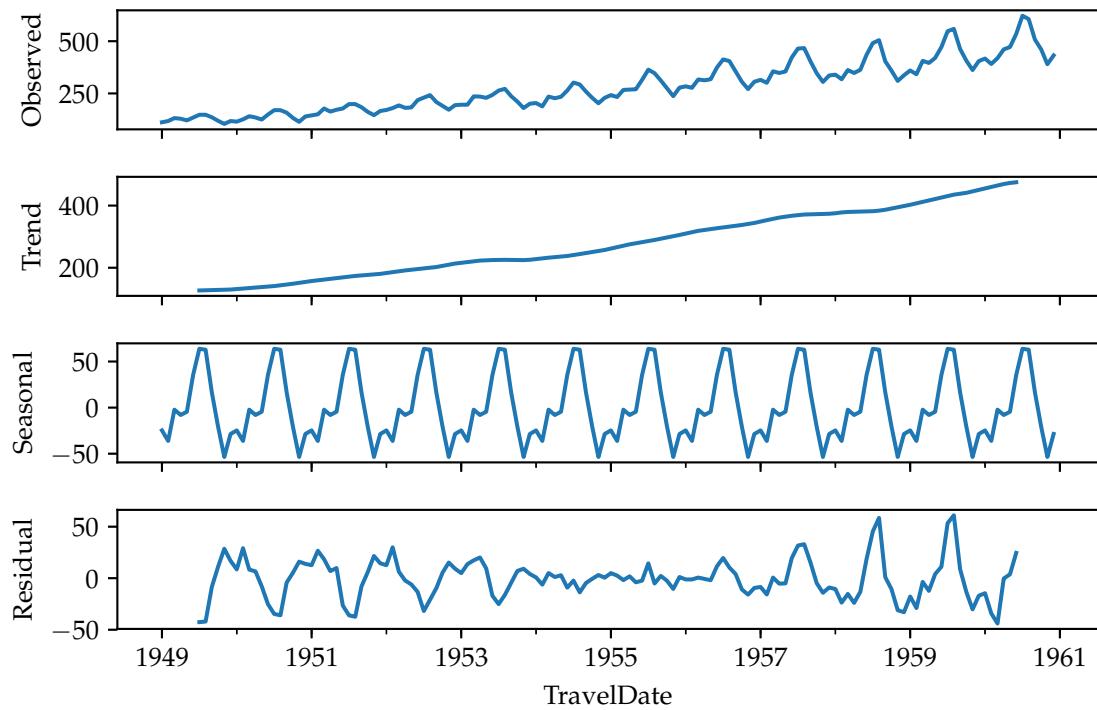


Abbildung 7.20.: Zerlegung einer additiven Zeitreihe

Der Restterm in Abbildung 7.19 und 7.20 unten zeigt offensichtlich ein nichtzufälliges Verhalten des Restterms $\hat{\tau}$. Der Grund dafür ist, dass das lineare Zerlegungsmodell hier nicht passt.

Wir können nun versuchen, die Schritte oben mit dem *Logarithmus* der **AirPassengers** durchzuführen, was einem multiplikativen Modell entspricht (siehe Abbildung 7.21). (zu R)

```
seasonal_decompose(np.log(AirP["Passengers"]), model = "add").resid.plot()
plt.show()
```

Wir sehen, dass im geschätzten Restterm beruhend auf den log-Daten in 7.21 der nichtzufällige Teil wesentlich vermindert wurde.

□

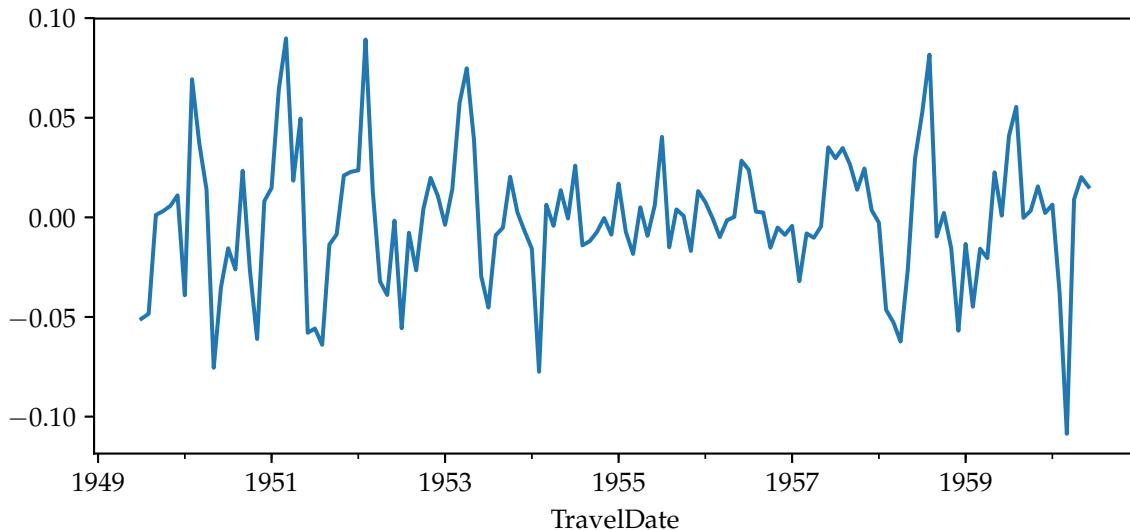


Abbildung 7.21.: Restterm für den Logarithmus der Daten

STL-Verfahren (Seasonal Decomposition of Time Series by Loess)

Das in der Funktion `seasonal_decompose()` implementierte und oben beschriebene Verfahren ist zwar sehr einfach, hat aber Nachteile:

- Das Verfahren ist nicht robust in Bezug auf Ausreisern in den Daten.
- Es wird angenommen, dass die saisonale Komponente über die Zeit konstant ist.

Die state-of-the-art Methode zur Zerlegung von Zeitreihen, welche nicht von den obengenannten Problemen beeinträchtigt wird, lautet *seasonal decomposition of time series by loess* (STL). Es ist in den Grundsätzen insofern ähnlich zur bisher verwendeten Methode, als die Trendkomponente durch Glätten der Zeitreihe und die saisonalen Effekte durch Subtraktion der Trenkomponente von der ursprünglichen Zeitreihe geschätzt werden. Die drei grundlegenden Unterschiede sind:

1. Das STL-Verfahren ist iterativ. Insbesondere werden Ausreisser im geschätzten Restterm detektiert, und deren Einfluss auf die Zerlegung der Zeitreihe wird durch Gewichtung entsprechend abgeschwächt.
2. Der moving average Filter wird durch *loess* Regression ersetzt, was eine grössere Flexibilität ermöglicht und in der Regel bessere Resultate erzielt als der moving average Filter. Loess Regression bezeichnet eine Form von lokaler (in der Regel linearer) Regression, was bedeutet, dass die Regressionskurve durch die Punkte $(x_1, y_1), \dots, (x_n, y_n)$ an einem Punkt x bloss aufgrund der Beobachtungen in der

näheren Umgebung von x geschätzt wird. Die Breite des Regressionsfensters ist ein Parameter bei der loess Regression. Falls die Fensterbreite grösser als die Spannweite der Beobachtungen ist, so entspricht loess Regression der standard-mässigen Regression.

3. Die saisonale Komponente wird nicht als konstant angenommen. Das STL-Verfahren berücksichtigt sogenannte *Subzyklen*, d.h., Unterreihen an jeder Position eines saisonalen Zyklus. Betrachten wir als Beispiel eine monatliche Zeitreihe, also eine Zeitreihe mit Periode 12, so entspricht zum Beispiel der Monat Januar einem Subzyklus. Dieser Subzyklus wird durch loess geglättet und kann sich über die Zeit ändern.

Es gibt bei der STL-Zerlegung einer Zeitreihe mehrere Parameter, die optimiert werden müssen, wobei `period` sicher die wichtigste ist. In **Python** ist das STL-Verfahren im Paket `stldecompose` implementiert, was im folgenden Beispiel illustriert wird.

Beispiel 7.4.10

Für den (logarithmierten) Datensatz **AirPassenger** ist die STL-Zerlegung in Abbildung 7.22 dargestellt. ([zu R](#))

```
import numpy as np
from stldecompose import decompose

decompose(np.log(AirP["Passengers"]), period = 12).plot();
```

Bemerkungen:

- i. Mit `pip install stldecompose` kann das Paket `stldecompose` in **Python** in der Kommandozeile installiert werden.



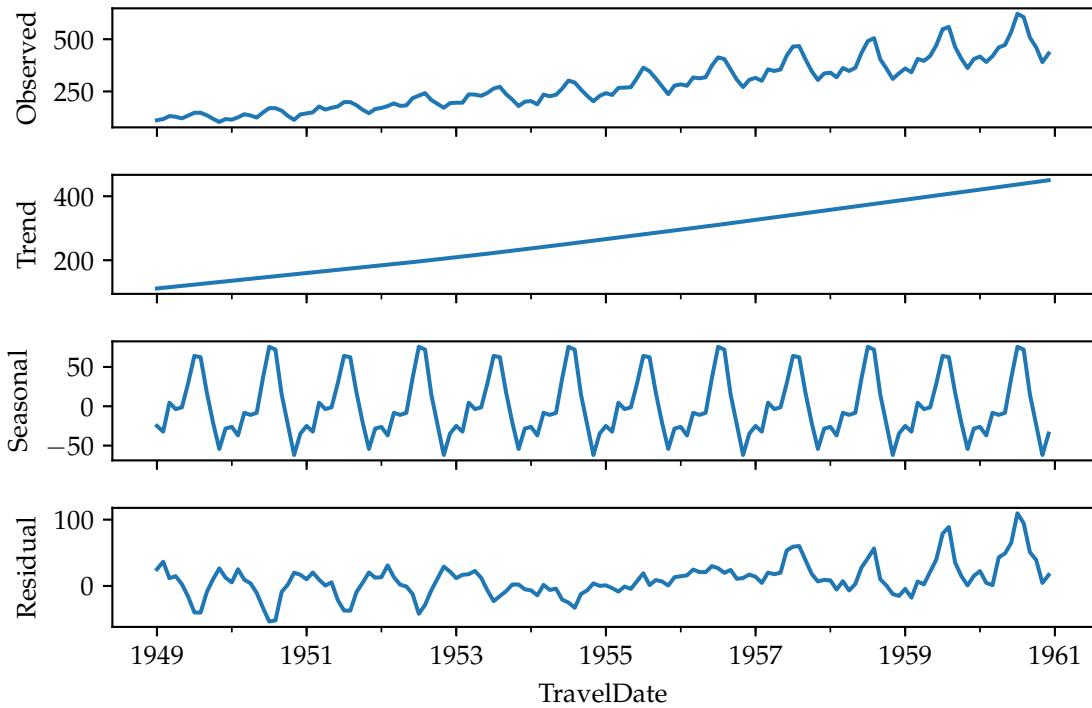


Abbildung 7.22.: STL-Zerlegung für den (logarithmierten) `AirPassenger`-Datensatz.

Konzeptionelle Lernziele

Sie sollten fähig sein, . . .

- die wichtigsten Ziele der Zeitreihenanalyse aufzulisten und mindestens drei Anwendungsbereiche von Zeitreihenanalyse zu nennen.
- den Unterschied zwischen Zeitreihendaten und geläufigen Datentabellen, die z.B. mit der linearen Regression modelliert werden, zu erklären.
- die Familie von Box-Cox-Transformationen zu definieren und qualitativ zu erklären, wie diese Transformationen einen Zeitreihendatensatz ändern.
- die additive Zerlegung einer Zeitreihe mit Hilfe von Moving Average zu erklären.
- das Konzept der STL-Zerlegung einer Zeitreihe in groben Zügen zu erklären.
- den log-return einer Zeitreihe zu berechnen und das Konzept von log-return zu erklären.

Computer-Basierte Lernziele

Sie sollten fähig sein, ...

- Zeitreihen mit Hilfe von **pandas** in **Python** einzulesen, Zeitfenster der Zeitreihe zu definieren und graphisch darzustellen.
- Zeitreihen mit passenden Methoden wie lagged scatterplots darzustellen oder nach Stunden, Monaten oder Jahren zu gruppieren und mit entsprechenden Boxplots darzustellen.
- die Funktionen **seasonal_decompose()** und **stl()** so anzuwenden, dass Zeitreihen zerlegt werden können.

Kapitel 8.

Mathematische Modelle für Zeitreihen

The true logic of this world is in
the calculus of probabilities.

(James Clerk Maxwell)

Im vorhergehenden Kapitel haben wir Zeitreihen als Beobachtungen von Daten eingeführt, die auf natürliche Weise chronologisch geordnet werden können. Wir haben dabei Konzepte für Transformationen, Visualisierungen und Zerlegungen kennengelernt. Des weiteren haben wir Zeitreihen mit Berechnungen und Darstellungen in **Python** studiert.

Wir gehen nun einen Schritt weiter und *modellieren* Zeitreihen¹. Zuerst betrachten wir aber ein paar einführende Beispiele.

8.1. Vom Random Walk zum Thermischen Rauschen von elektrischen Widerständen

8.1.1. Random Walk

Ein *Random Walk*, auch Zufallsbewegung oder Irrfahrt genannt, ist ein mathematisches Modell für eine Bewegung, bei der die einzelnen Schritte zufällig erfolgen. Es handelt sich um einen stochastischen Prozess in diskreter Zeit mit unabhängigen und identisch verteilten Zuwächsen.

¹Dieses Kapitel folgt den Abschnitten 1.3 – 1.6 des Lehrbuches *Time Series Analysis and Its Applications* von Robert H. Shumway und David S. Stoffner, Springer 2011.

Beispiel 8.1.1 Aktienkurs

Aktienkurse können nicht vorhergesagt werden. Durch Random-Walks zum Beispiel können die zufälligen Kursanstiege oder Kurszerfälle von Tag zu Tag (diskrete Zeit) modelliert werden.

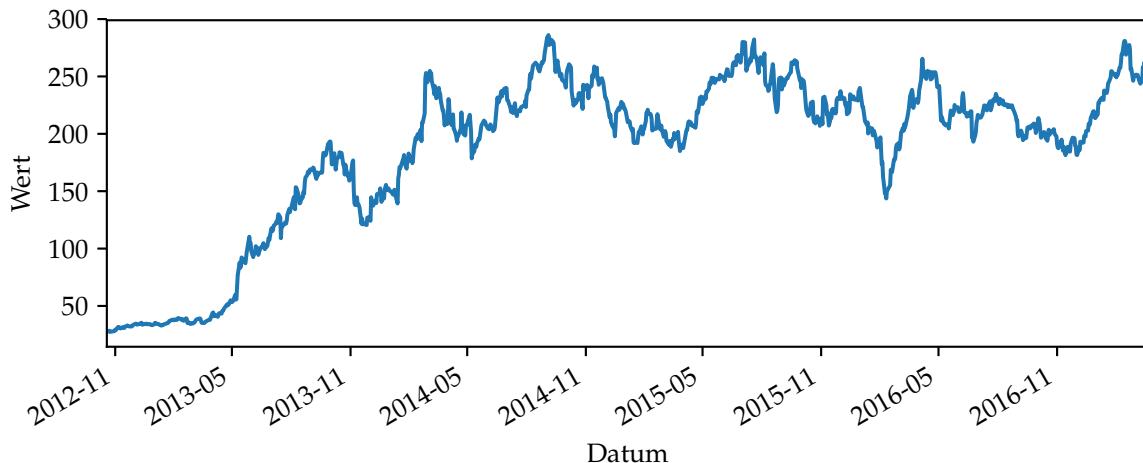


Abbildung 8.1.: Tagesabschlüsse des Tesla Aktienindex.

□

Beispiel 8.1.2 Weisses Rauschen

In einem Leiter, an dem keine Spannung angelegt ist, bewegen sich die Elektronen trotzdem, und zwar in zufälliger Art und Weise aufgrund der Wärme. Dies führt zu einer kleinen Spannung im Leiter, die dauernd zufällig ändert. Wir sprechen in diesem Fall von *weissem Rauschen*.

Auch diesen Prozess können wir mit Hilfe eines Random-Walks modellieren.

□

Beispiel 8.1.3 Galton-Brett

Ein weiteres Beispiel für einen Random Walk stellt das sogenannte Galtonsche Nagelbrett dar. Die Sprungwahrscheinlichkeit der Kugel auf einem Nagel nach links oder nach rechts ist die gleiche, nämlich 50 %. Die Verteilung der Kugeln nach N Sprüngen folgt einer Binomialverteilung.

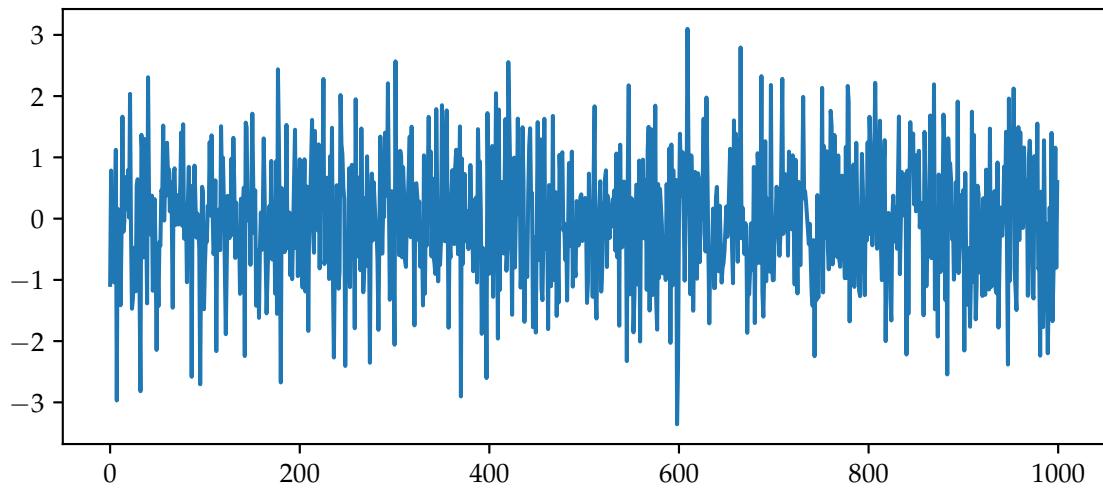


Abbildung 8.2.: Weisses Rauschen.

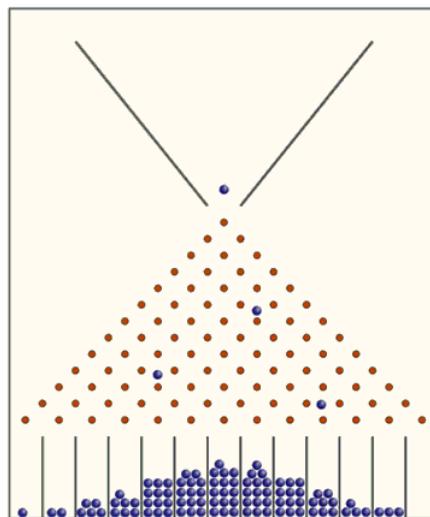


Abbildung 8.3.: Galton-Nagelbrett

□

Um den Begriff des *Random Walks* besser zu verstehen, stellen wir uns einen Betrunkenen vor, der aus einer Bar in eine Gasse tritt und versucht, nach Hause zu gehen. Da er etwas durcheinander ist, geht nicht jeder seiner Schritte in die richtige Richtung. Wir gehen davon aus, dass jeder seiner Schritte die Grösse Δx hat und mit Wahrscheinlichkeit p nach rechts geht und mit Wahrscheinlichkeit $q = 1 - p$ nach links. Wenn $p > q$ ist, geht er häufiger nach rechts als nach links.

Der Weg des Betrunkenen ist ein Beispiel für einen Zufallspfad („random walk“) auf

einem eindimensionalen Gitter mit der Gitterkonstanten Δx (siehe Abbildung 8.4).

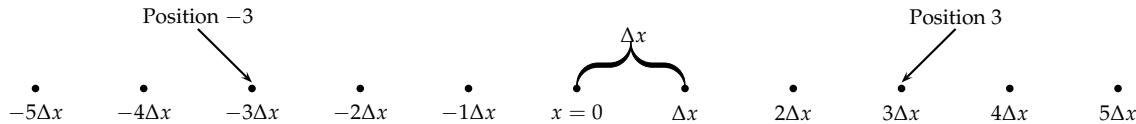


Abbildung 8.4.: Schritte auf einem eindimensionalen Gitter bei einem Random-Walk.

Wir nehmen an, dass unser Betrunkener am Ort $x = 0$ seinen nächtlichen Spaziergang beginnt. Bezeichnen wir mit X die Anzahl Schritte, die unser Betrunkener nach rechts geht, dann ist

$$X \sim \text{Bin}(N, p)$$

wobei N die Gesamtzahl Schritte ist.

Nun möchten wir wissen, wie gross die Wahrscheinlichkeit ist, dass sich unser Betrunkener nach N Schritten an der Position m (in Einheiten von Δx) auf dem Gitter befindet. Man beachte, dass m auch negativ sein kann.

Wir bezeichnen mit M_N die Zufallsvariable für die Position unseres Spaziergängers nach N Schritten auf dem Gitter (in Einheiten von Δx). Wir fragen also, wie gross die Wahrscheinlichkeit

$$P(M_N = m)$$

ist, dass der Pfad nach N Schritten an der Position m ist.

Weiter bezeichnen wir mit S_r die Anzahl Schritte nach rechts und mit S_l die Anzahl Schritte nach links. Nach unseren Voraussetzungen muss dann gelten

$$S_r + S_l = N \quad \text{und} \quad S_r - S_l = m$$

Dies ist ein lineares Gleichungssystem mit zwei Gleichungen und den beiden Unbekannten S_r und S_l . Wir lösen nach S_r und S_l auf, indem wir die beiden obigen Gleichungen addieren und subtrahieren:

$$S_r = \frac{N + m}{2} \quad \text{und} \quad S_l = \frac{N - m}{2}$$

Um nach N Schritten bei m zu sein, müssen wir also $(N + m)/2$ von den N Schritten nach rechts und $(N - m)/2$ Schritte nach links gehen. Die Zahl $m + N$ muss gerade sein: wenn also N gerade ist, ist auch m gerade; ist N ungerade, so ist es m .

Die Anzahl Schritte nach rechts bestimmt die Position M_N eindeutig. Wir können die Wahrscheinlichkeit, dass sich der Betrunkene nach insgesamt N Schritten bei $M_N =$

m befindet, berechnen, indem wir die Zufallsvariable X (Anzahl Schritte nach rechts) durch $(N + m)/2$ ersetzen:

$$P(M_N = m) = \binom{N}{\frac{N+m}{2}} p^{(N+m)/2} q^{(N-m)/2}$$

Wenn N gross ist (dies ist bei einem betrunkenen Bargänger normalerweise der Fall), erhalten wir unter Verwendung des Zentralen Grenzwertsatzes eine Normalverteilung. Denn eine binomialverteilte Zufallsvariable kann als die Summe von N unabhängigen Bernoulli-verteilten Zufallsvariablen aufgefasst werden, und für grosse N kann die Verteilung der Summe von beliebig verteilten Zufallsvariablen als Normalverteilung angenähert werden (siehe Abschnitt 3.4.3 zur Normalapproximation).

Folglich ist die Wahrscheinlichkeit, dass sich der Bargänger nach N Schritten bei $M_N = m$ befindet:

$$P(M_N = m) = \frac{2}{\sqrt{2\pi\sigma^2}} e^{-\frac{(m-\mu)^2}{2\sigma^2}} \quad (8.1)$$

Der Vorfaktor 2 ist so gewählt, dass die Summe über alle m -Werte eins ergibt, die Punktswahrscheinlichkeiten also normiert sind. Hierbei muss man beachten, dass für (un)gerade N nur (un)gerade Werte von m auftauchen, so dass der Abstand zweier benachbarter Werte von m den Betrag 2 hat:

$$\Delta m = \pm 2$$

Diese Schrittgrösse steht im Zähler des Vorfaktors, so dass die Summe von $P(M_N = m)$ über alle m mit $\Delta m = \pm 2$ eins ergibt.

Es handelt sich bei

$$P(M_N = m)$$

immer noch um eine diskrete Wahrscheinlichkeitsverteilung: $P(M_N = m)$ ist die Punktswahrscheinlichkeit für die diskrete Zufallsvariable M_N .

Wie bestimmen wir nun die Werte der Parameter μ und σ in der Gleichung (8.1)?

Als Erwartungswert der Normalverteilung für die Zufallsvariable M_N nehmen wir den Erwartungswert der Binomialverteilung: N mal die mittlere Positionsänderung eines Schrittes (in Einheiten von Δx),

$$\begin{aligned} \mu &= E(M_N) = N \cdot E(M_1) = N \left(1 \cdot P(M_1 = +1) - 1 \cdot P(M_1 = -1) \right) \\ &= N(p - q) \\ &= N(p - (1 - p)) = N(2p - 1) \end{aligned}$$

wobei M_1 Bernoulli-verteilt ist.

Kapitel 8. Mathematische Modelle für Zeitreihen

Die Varianz der Normalverteilung ist N mal die Varianz eines Schrittes

$$\begin{aligned}\sigma^2(M_N) &= \text{Var}(M_N) \\ &= N \cdot \text{Var}(M_1) \\ &= N \left((1 - E(M_1))^2 P(M_1 = +1) + (-1 - E(M_1))^2 P(M_1 = -1) \right) \\ &= N \left((1 - (p - q))^2 p + (-1 - (p - q))^2 q \right) \\ &= 4Npq\end{aligned}$$

Also ist die Wahrscheinlichkeit, dass sich unser nächtlicher Spaziergänger nach N Schritten an der Position m befindet

$$\begin{aligned}P(M_N = m) &= \frac{2}{\sqrt{2\pi\sigma^2}} e^{-\frac{(m-\mu)^2}{2\sigma^2}} \\ &= \frac{2}{\sqrt{8\pi Npq}} e^{-\frac{(m-N(p-q))^2}{8Npq}}\end{aligned}$$

Wir können uns nun fragen, wie weit unser Betrunkener nach N Schritten im Durchschnitt kommt. Die Antwort ist durch den Erwartungswert der Position, also

$$\mu = E[M_N] = N(2p - 1)$$

gegeben.

Falls unser Spaziergänger so betrunken ist, dass jeder seiner Schritte mit gleicher Wahrscheinlichkeit nach rechts oder links geht, also $p = 1/2$ ist, dann ist seine erwartete Position null. Dies hat einerseits damit zu tun, dass der Barbesucher mit grösster Wahrscheinlichkeit wieder bei der Bar landet, andererseits aber, wenn er mal vorangekommen ist, gleich oft – über viele Barbesuche gemittelt – eine gewisse Distanz nach rechts wie nach links zurückgelegt hat. Diese Abstände heben sich dann auf.

Wollen wir das Vorankommen unseres Barbesuchers beschreiben, müssen wir die Vorzeichen der zurückgelegten Distanz „neutralisieren“. Wir könnten die mittlere zurückgelegte Distanz des Spaziergängers von der Startposition mit dem Erwartungswert des Absolutwertes von M_N , also

$$E[|M_N|]$$

messen. Es ist allerdings vorteilhafter, das mittlere Quadrat der Verschiebung zu bestimmen, also

$$E[(M_N - M_0)^2]$$

wobei $M_N - M_0$ die Verschiebung von der Anfangsposition M_0 bezeichnet. Im Fall von $p = 1/2$ und $M_0 = 0$ ist dies aber identisch mit

$$E[(M_N - M_0)^2] = E[(M_N - 0)^2] = E[(M_N - \mu)^2] = \sigma^2 = 4Npq$$

Also für unseren betrunkenen Spaziergänger mit $p = 1/2$ ist

$$\mathbb{E}[(M_N - M_0)^2] = N$$

Um die Einheit von Distanz zu bekommen, kann man die Wurzel vom *mittleren Verschiebungskvadrat*

$$\sqrt{\langle (M_N - M_0)^2 \rangle} \equiv \sqrt{\mathbb{E}[(M_N - M_0)^2]}$$

betrachten, wobei die Notation $\langle \cdot \rangle$ eine Mittelung oder die Erwartungswertoperation bezeichnet.

Im Falle von $p = q = 1/2$ und $M_0 = 0$ finden wir

$$\sqrt{\langle (M_N - M_0)^2 \rangle} = \sqrt{N}$$

Bei $N = 1000$ Schritten der Schrittweite 1 m hätte unser Betrunkener eine *quadratisch gemittelte Distanz* von 34 m zurückgelegt.

Beispiel 8.1.4

Ein weiteres Beispiel eines Stochastischen Prozesses ist das *thermische Rauschen* oder *Widerstandsräuschen*.

Thermisches Rauschen kommt in jedem elektrischen Leiter vor und wird durch die ungeordnete Wärmebewegung der Ladungsträger hervorgerufen (Brownsche Bewegung, siehe Abschnitt 8.7). In einem Ohmschen Widerstand, wie er in Abbildung 8.5 gezeigt wird, tritt an den Anschlüssen durch eine zufällige Ansammlung von Elektronen sporadisch eine Rauschspannung auf, selbst wenn kein Strom durch den Leiter fliessst.

Ein typischer Verlauf einer Rauschspannung $v(t)$ ist in Abbildung 8.5 aufgetragen. Die auftretenden Spannungen liegen unter üblichen Bedingungen in der Größenordnung von Mikrovolts. Wir haben es mit einem Zufallssignal (engl. random signal) oder einem „stochastischen Signal“ zu tun, da wir aus der Vergangenheit des Signals den zukünftigen Verlauf nicht vorhersagen können, und sich ein einmal aufgetretener Signalverlauf in einem wiederholten Experiment nicht wiederholen lässt.

Stochastische Signale können nur mit statistischen Größen oder Mittelwerten beschrieben werden. Bei einem thermischen Rauschsignal ist der lineare Mittelwert der Spannung null, da ja im Mittel kein Strom fliessen kann.

□

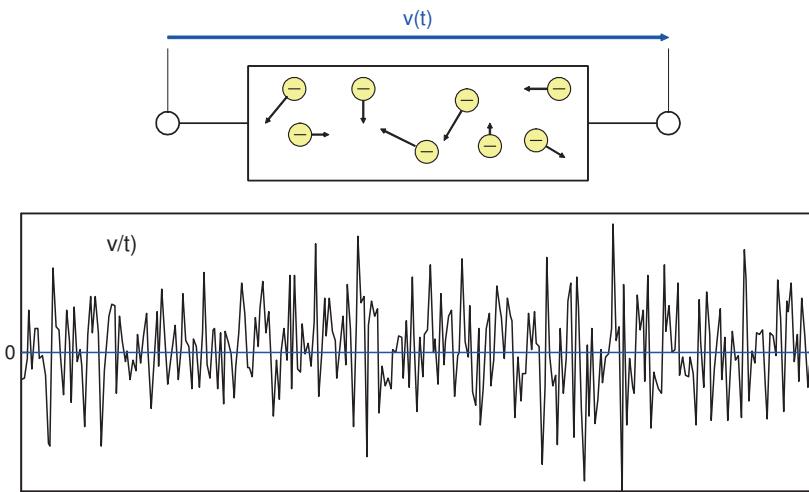


Abbildung 8.5.: (oben): Die thermische Bewegung der Elektronen in einem Widerstand erzeugt eine zufällige Rauschspannung an den Klemmen. (unten): Zufälliger oder „stochastischer“ Verlauf einer Rauschspannung über die Zeit.

8.2. Charakteristische Größen von stochastischen Signalen

Ein Signal wird *stochastisch* genannt, wenn es zufällige Werte annimmt. Wir betrachten im folgenden zwei stochastische Signale $S_1(t)$ und $S_2(t)$, dargestellt in Abbildung 8.6.

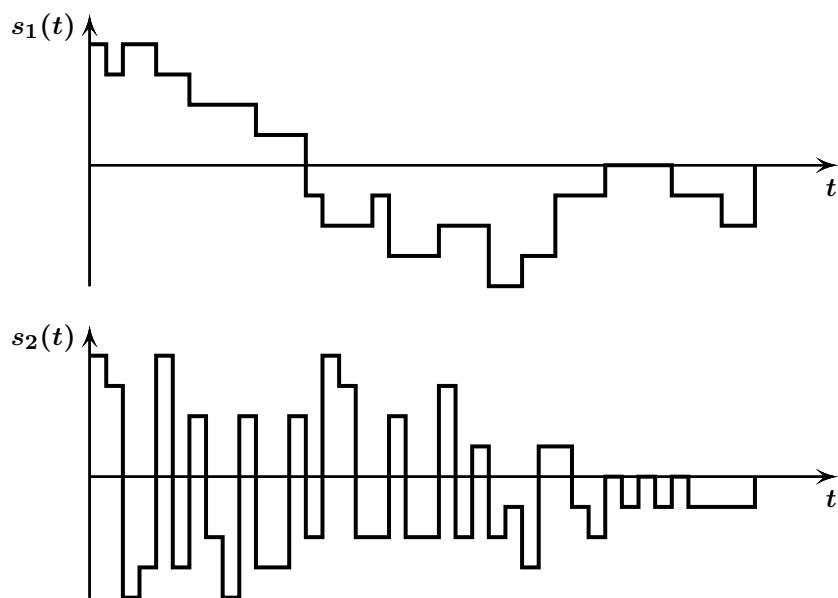


Abbildung 8.6.: Zwei diskrete stochastische Signale $S_1(t)$ und $S_2(t)$.

Der Einfachheit halber nehmen wir an, dass beide Signale nur eine endliche Anzahl von Amplitudenniveaus annehmen können, nämlich

$$\{0, \pm 1, \pm 2, \pm 3, \pm 4\}$$

Es handelt sich also bezüglich Amplitude um einen sogenannten *diskreten Zufallsprozess*. Weiterhin können die Signale nur bei Vielfachen von T ihre Amplitude ändern, d.h. dieser Zufallsprozess ist auch *zeitdiskret*.

Im Gegensatz zu deterministischen Signalen kann im Falle von stochastischen Signalen aufgrund der Kenntnis eines Signalwertes zu einer Zeit t_0 nicht mit Sicherheit auf den Signalwert zur Zeit $t_0 + \Delta T$, $\Delta T > 0$ geschlossen werden.

Deswegen müssen andere Beschreibungsmethoden angewandt werden. Eine Möglichkeit besteht darin, dass man angibt, wie häufig im Mittel ein bestimmter Amplitudenwert auftritt. In Abbildung 8.7 sind die relativen Häufigkeiten des Auftretens der Signalwerte s von $S_1(t)$ und $S_2(t)$ dargestellt.

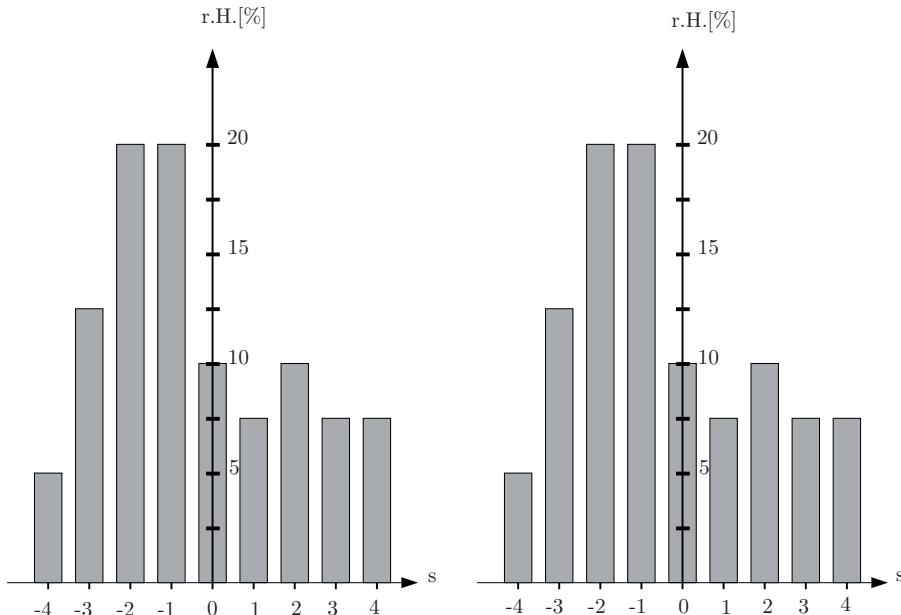


Abbildung 8.7.: Relative Häufigkeiten der Signalwerte (Wahrscheinlichkeitsverteilung) von: a) $S_1(t)$, b) $S_2(t)$.

Stellen wir uns die Frage, wie viel Prozent der Signalwerte des Signals $S_1(t)$ liegen unterhalb eines Schwellenwertes z , so führt dies zur kumulativen Verteilungsfunktion, dargestellt in Abbildung 8.8.

In den Abbildungen 8.7 sind die relativen Häufigkeiten der Amplituden in einem endlichen Zeitintervall des Signals dargestellt. Für eine sehr grosse Anzahl Amplitudenwerte bzw. ein unbeschränktes Zeitintervall können wir diese relativen Häufigkeiten im Grenzfall als Auftrittswahrscheinlichkeiten auffassen.

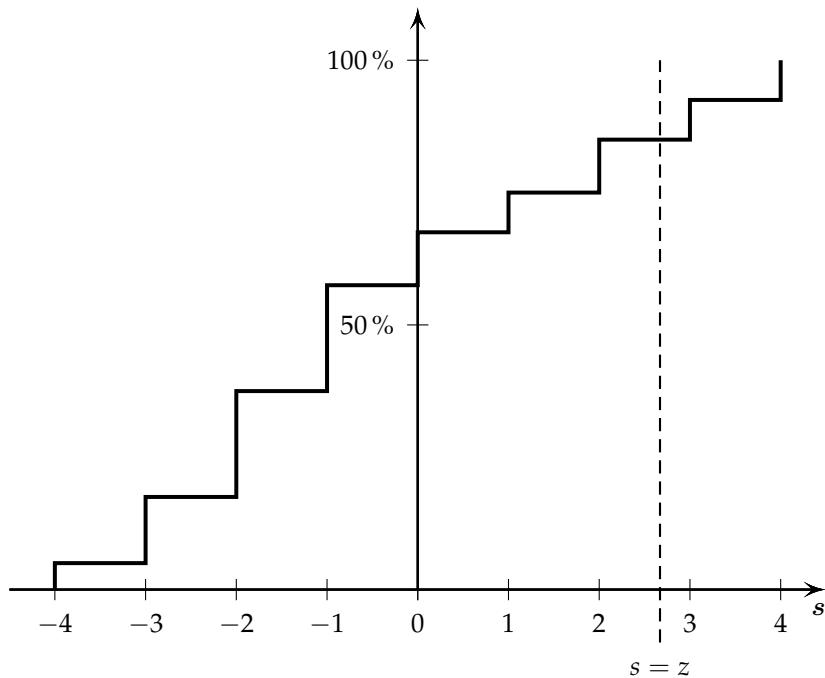


Abbildung 8.8.: Prozentualer Anteil der Signalwerte (kumulative Wahrscheinlichkeitsverteilung) von $S_1(t)$ unterhalb des Schwellenwertes z .

Die Darstellung der Amplitudenwerte in Form eines Histogramms (siehe Abbildung 8.7) und als kumulative Verteilungsfunktion (siehe Abbildung 8.8) liefern in verschiedener Form die gleiche Information über ein Signal.

Die beiden Darstellungen werden als *Wahrscheinlichkeitsverteilungen* bezeichnet und zur Beschreibung von stochastischen Signalen verwendet. Aus Abbildung 8.7 erkennen wir, dass die relativen Häufigkeiten (bzw. Auftrittswahrscheinlichkeiten) der Signalwerte für die Signale $S_1(t)$ und $S_2(t)$ gleich sind.

Wenn wir allerdings die beiden Signale $S_1(t)$ und $S_2(t)$ betrachten, so erscheint anschaulich klar, dass der frequenzmässige Aufbau der Signale sehr unterschiedlich ist, da $S_1(t)$ lediglich schwach, $S_2(t)$ hingegen stark oszilliert. Mit der Angabe der Auftrittswahrscheinlichkeiten haben wir die Signale also noch nicht ausreichend charakterisiert.

Es fehlt noch ein Mass für den „inneren Zusammenhang“ eines Signals, also für die Stärke der Schwankung. Zwei relativ nahe beieinanderliegende Signalwerte von $S_1(t)$ sind wahrscheinlich ungefähr gleich gross.

Diese Aussage ist für das Signal $S_2(t)$ nicht richtig, d.h. benachbarte Signalwerte hängen im Signal $S_1(t)$ offenbar stärker zusammen als im Signal $S_2(t)$. Für diesen „inneren Zusammenhang“ eines Signals existiert ebenfalls ein Mass, nämlich die *Autokorrelationsfunktion*. Benachbarte Signalwerte von $S_1(t)$ sind stärker korreliert als

solche von $S_2(t)$. Die Autokorrelationsfunktionen werden wir im Abschnitt ?? näher betrachten.

8.3. Wahrscheinlichkeitsverteilungsfunktion von Stochastischen Prozessen

Wir betrachten *stochastische Signale* als Musterfunktionen von Zufallsprozessen. Es gilt die folgende Definition (Abbildung 8.9):

Stochastischer Prozess

Ein Zufallsprozess oder stochastischer Prozess $S(t)$ ist durch ein *Ensemble* von Musterfunktionen $\{s_1(t), s_2(t), \dots, s_N(t)\}$ gegeben.

Eine *Realisierung* ergibt sich durch die zufällige Auswahl einer Musterfunktion $s_i(t)$ mit $1 \leq i \leq N$ des Ensembles.

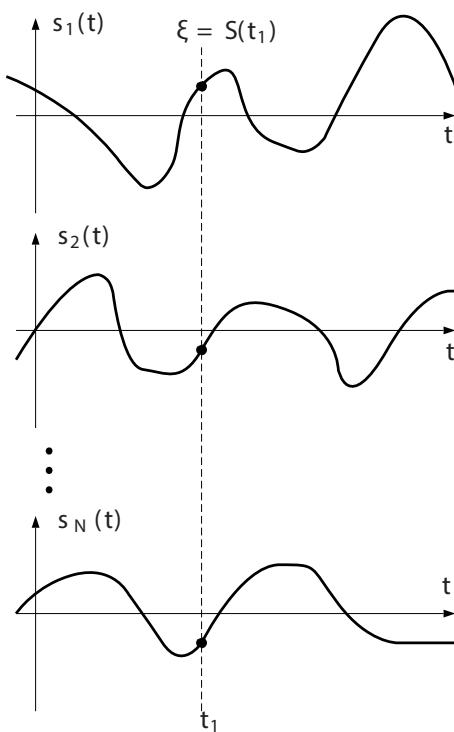


Abbildung 8.9: Ensemble von Musterfunktionen eines kontinuierlichen Zufallsprozesses.

Der Zufallsprozess $S(t)$ kann nun mit Hilfe von Wahrscheinlichkeitsverteilungsfunktionen beschrieben werden.

Kapitel 8. Mathematische Modelle für Zeitreihen

Dazu betrachten wir den (gemessenen) Funktionswert $s(t_1)$ des Prozesses $S(t)$ zum Zeitpunkt $t = t_1$ als Realisierung der Zufallsvariable $\xi = S(t_1)$ (Abbildung 8.9), wobei es sich hier im Gegensatz zu den in Abbildung 8.6 dargestellten Zufallsprozessen um Musterfunktionen eines *kontinuierlichen* Zufallsprozesses handelt.

Über das Ensemble der Musterfunktionen kann man die Wahrscheinlichkeit ermitteln, mit der die Werte $s(t_1)$ im Bereich $-\infty < \xi \leq s$ anzutreffen sind:

$$P(\xi \leq s) = F(s)$$

wobei $F(s)$ die kumulative Wahrscheinlichkeitsverteilungsfunktion ist. Sie ist analog zu der in Abbildung 8.8 gezeigten Treppenkurve; sie ist in diesem Fall aber eine stetige Funktion. Da ihr Funktionswert eine Wahrscheinlichkeit ist, gilt

$$0 \leq F(s) \leq 1$$

Da das Ereignis $\{\xi \leq s\}$ grösser wird, wenn s grösser wird, ist $F(s)$ monoton wachsend mit s . Die beiden Ereignisse $\xi \leq s$ und $s < \xi \leq (s + \Delta s)$ schliessen sich gegenseitig aus. Deshalb dürfen die zugehörigen Wahrscheinlichkeiten addiert werden:

$$P(\xi \leq s) + P(s < \xi \leq (s + \Delta s)) = P(\xi \leq (s + \Delta s))$$

Damit gilt auch:

$$P(s < \xi \leq (s + \Delta s)) = F(s + \Delta s) - F(s) \geq 0$$

Dividieren wir obigen Ausdruck durch Δs , so führt Grenzwertbildung auf die Definition der *Wahrscheinlichkeitsdichtefunktion* $f(s)$:

$$f(s) := \lim_{\Delta s \rightarrow 0} \frac{P(s < \xi \leq (s + \Delta s))}{\Delta s} = \frac{dF(s)}{ds} \geq 0 \quad (8.2)$$

Man beachte ferner, dass $f(s)$ ggf. dimensionsbehaftet ist mit $[f(s)] = s^{-1}$, also ist z.B. $[f(s)] = \text{Sekunde}^{-1}$, sofern s die Zeit gemessen in Sekunden ist. Die Dichtefunktion hat die wichtige Eigenschaft

$$\int_{-\infty}^{\infty} f(s) \, ds = 1$$

Sind die zu einem stochastischen Signal gehörenden Funktionen $f(s)$ bzw. $F(s)$ bekannt, so lassen sich systemtechnisch wichtige Größen ermitteln. Zum Beispiel kann bei bekannter Wahrscheinlichkeitsdichte $f(s)$ der Werte $\{s(t_1)\}$ ausgesagt werden, dass sich diese mit der Wahrscheinlichkeit

$$P_{ab} = \int_a^b f(s) \, ds$$

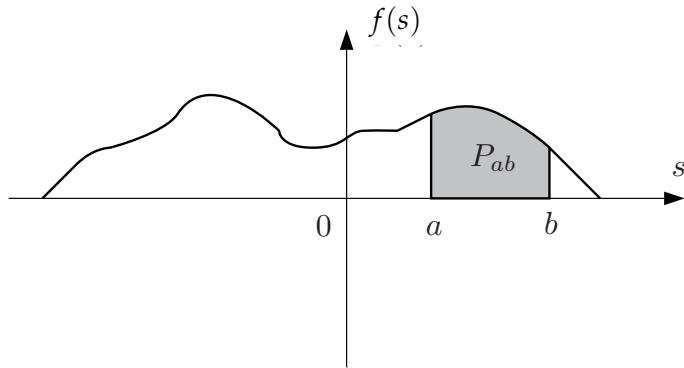


Abbildung 8.10.: Interpretation der Wahrscheinlichkeitsdichtefunktion.

im Bereich $a \leq s(t_1) \leq b$ aufhalten. Anschaulich ist also die Fläche unter der Kurve $f(s)$ von $s = a$ bis $s = b$ gleich der Wahrscheinlichkeit, mit der die Zufallsvariable ξ einen Wert zwischen a und b annimmt (siehe Abbildung 8.10).

Bis anhin wurde lediglich der Funktionswert eines stochastischen Signals zum Zeitpunkt $t = t_1$ betrachtet. Allgemein sind Wahrscheinlichkeitsverteilungsfunktion und Wahrscheinlichkeitsdichtefunktion zeitabhängig, d.h. man hat $F(s, t)$ und $f(s, t)$.

Bei vielen technisch relevanten Zufallsprozessen sind die statistischen Eigenschaften unabhängig von der Zeit. Diese Eigenschaft nennt man *Stationarität*:

Stationärer stochastischer Prozess

Ein Zufallsprozess heisst *stationär*, wenn die kumulative Wahrscheinlichkeitsverteilungsfunktion oder die Wahrscheinlichkeitsdichtefunktion zeitunabhängig sind:

$$\begin{aligned} F(s, t) &= F(s) \\ f(s, t) &= f(s) \end{aligned}$$

Bemerkungen:

- i. Diese Definition von Stationarität wird auch als starke Stationarität bezeichnet. Ein stochastischer Prozess $X(t)$ heisst *stark stationär*, wenn die Verteilung von $X(t + s)$ nicht von der Verschiebung s abhängt.
- ii. Ein stochastischer Prozess heisst *schwach stationär*, wenn der Erwartungswert konstant ist und die Varianz endlich ist.

Beispiel 8.3.1

Ein Beispiel für einen stationären Zufallsprozess ist das *thermische Rauschen* $V(t)$ in einem elektrischen Widerstand, siehe Abbildung 8.5.

Betrachten wir die Verteilung der Rauschamplituden $V(t)$, z.B. die Werte der Spannung v aus Abbildung 8.5, dann stellen wir erstens fest, dass sich die Verteilung zeitlich nicht ändert und zweitens, dass sie einer Normalverteilung folgt:

$$f(v) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(v-\mu)^2}{2\sigma^2}}$$

wobei μ den Erwartungswert und σ^2 die Varianz bzw. σ die Standardabweichung der Zufallsvariablen $V(t) = V$ (Rauschamplitude) bezeichnet.

Würden wir die Verteilung der Rauschamplituden in einem elektrischen Widerstand in einem Jahr mit der Verteilung der momentanen Rauschamplituden vergleichen, würden wir keinen Unterschied feststellen können.

Anders verhält es sich bei einem *Diffusionsprozess*: in diesem Fall ändert sich die Wahrscheinlichkeitsdichte für die Position x eines Teilchens in Abhängigkeit der Zeit t gemäss

$$f(x; t) = \frac{1}{\sqrt{4\pi D t}} e^{-\frac{(x-vt)^2}{4Dt}}$$

unter der Annahme, dass es sich zum Zeitpunkt $t = 0$ an der Stelle $x = 0$ befand. In diesem Fall haben wir also einen nicht-stationären stochastischen Prozess.

□

Für eine beliebige Wahrscheinlichkeitsdichtefunktion $f(s, t)$ berechnen sich Erwartungswert und Varianz gemäss

$$\mu(t) = \int_{-\infty}^{\infty} s \cdot f(s, t) \, ds \quad (8.3)$$

$$\sigma^2(t) = \int_{-\infty}^{\infty} (s - \mu)^2 f(s, t) \, ds \quad (8.4)$$

Die Berechnungen gemäss (8.3) und (8.4) können als Mittelwertbildung über das Ensemble $\{s_1(t), s_2(t), \dots, s_N(t)\}$ bzw. über die Schar des Zufallsprozesses $S(t)$ angesehen werden.

Ist der Zufallsprozess $S(t)$ stationär, so sind die Scharmittelwerte

$$\mu_S(t) = E[S(t)]$$

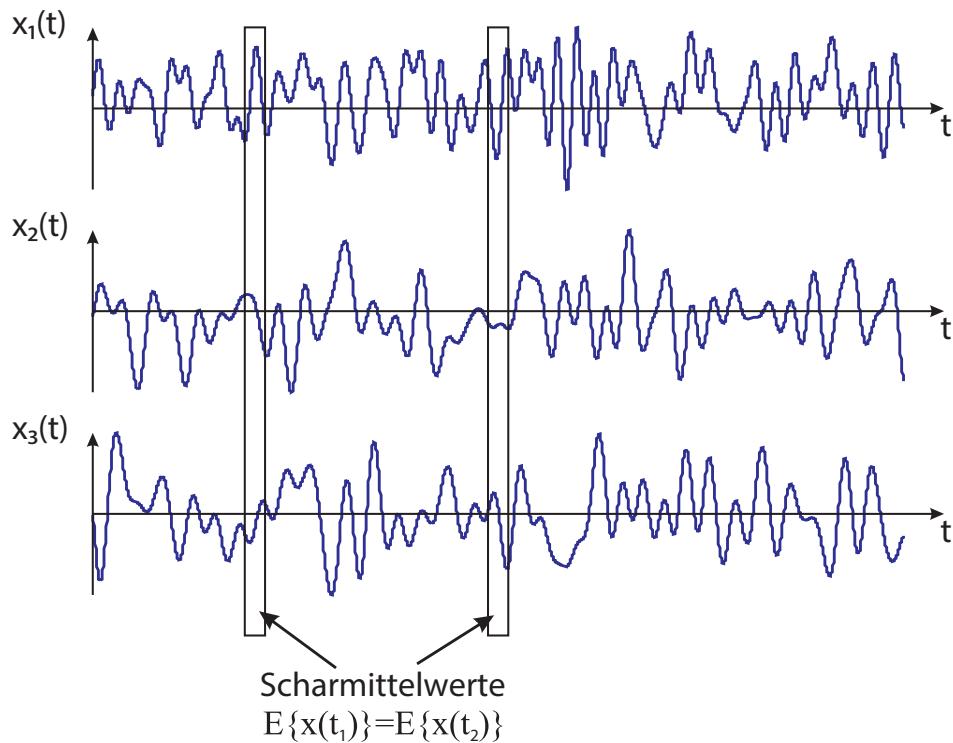


Abbildung 8.11.: Bestimmung des Erwartungswertes über eine Schar von Musterfunktionen $\{x_1(t), x_2(t), \dots, x_N(t)\}$ eines stationären stochastischen Prozesses $X(t)$. In diesem Fall sind die Scharmittelwerte $\mu_X(t) = E[X(t)]$ zeitunabhängig.

zeitunabhängig (siehe Abbildung 8.11). Man bezeichnet $\mu_S(t)$ und $\sigma_S^2(t)$ deshalb auch als *Scharmittelwert*, resp. *Scharvarianz*. Sie können berechnet werden, wenn eine mathematische Beschreibung der Wahrscheinlichkeitsdichtefunktion $f(s, t)$ vorliegt.

8.3.1. Ergodizität²

In der Praxis wird jedoch oft nur eine einzige Realisierung des Prozesses beobachtet, und man kennt die Wahrscheinlichkeitsdichtefunktion nicht. In diesem Fall lassen sich μ_S und σ_S^2 nur bestimmen, wenn die betrachteten Zufallsprozesse eine weitere Regularität aufweisen, die als *Ergodizität* bezeichnet wird (siehe Abbildung 8.13). Dazu definieren wir zuerst den Zeitmittelwert:

Zeitmittelwert

²Dieses Kapitel ist nicht prüfungsrelevant.

Für einen stochastischen Prozess $S(t)$ definieren wir den Zeitmittelwert

$$\langle s(t) \rangle_T = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=-\frac{T}{2}}^{\frac{T}{2}} s(t) dt$$

Geometrisch können wir den Zeitmittelwert wie folgt interpretieren: Die Fläche unter der Kurve des Stochastischen Signals $S(t) > 0$ dividieren wir durch die Länge der Integrationszeit.

D.h. wir erhalten die Höhe des Rechtecks mit der gleichen Fläche wie die Fläche unter der Kurve des Stochastischen Prozesses $S(t)$ (siehe Abbildung 8.12).

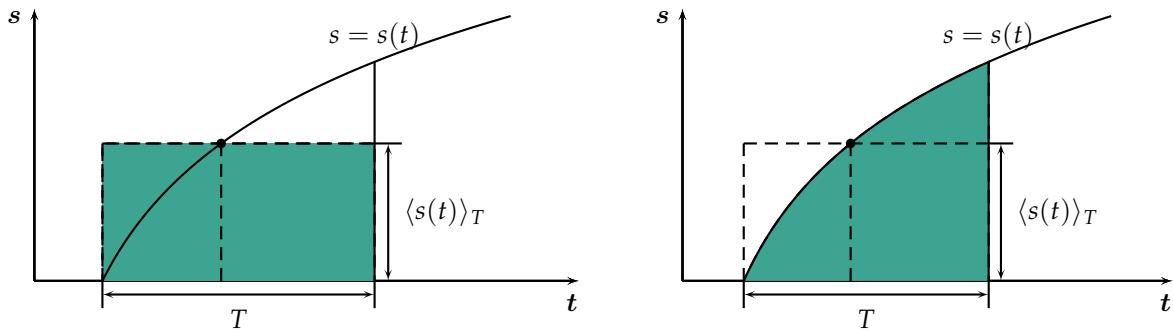


Abbildung 8.12.: Gleich grosse Flächen

Beispiel 8.3.2

Sei A eine auf dem Intervall $[0, 1]$ uniform verteilte Zufallsvariable. Wir definieren den stochastischen Prozess

$$S(t) = A$$

Eine Realisierung dieses stochastischen Prozesses besteht dann darin, dass eine auf dem Intervall $[0, 1]$ uniform verteilte Zufallszahl „gezogen“ wird und zu jedem Zeitpunkt diesen Wert besitzt.

Es handelt sich also um einen *stationären* stochastischen Prozess. Der Zeitmittelwert ist dann gegeben durch

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=-\frac{T}{2}}^{\frac{T}{2}} A dt = A \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=-\frac{T}{2}}^{\frac{T}{2}} dt = A$$

Kapitel 8. Mathematische Modelle für Zeitreihen

Da A gerade die Höhe des Rechtecks ist, und A eine Zufallsvariable ist, ist der Zeitmittelwert für jede Realisierung des Zufallsprozesses $S(t)$ verschieden.

□

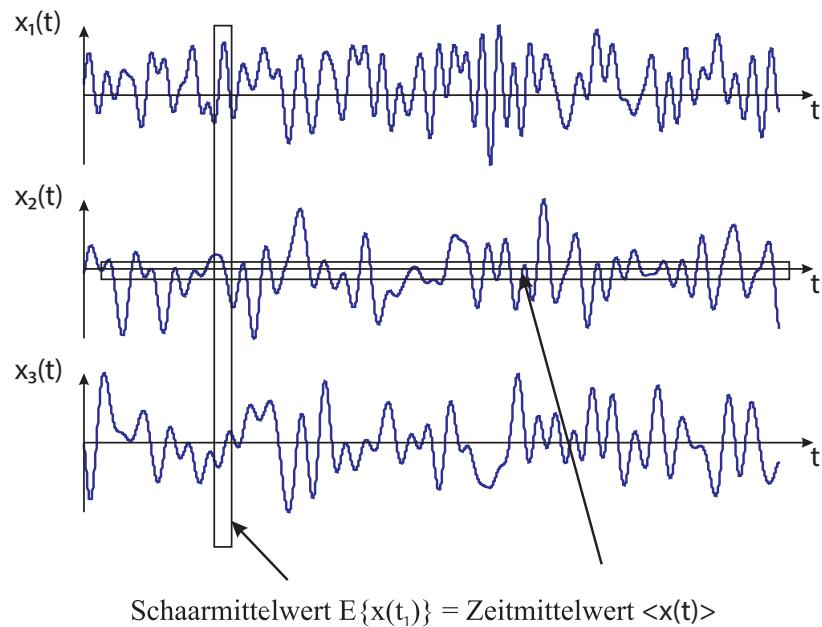


Abbildung 8.13.: Falls die Scharmittelwerte bei einem stationären stochastischen Prozess $X(t)$ mit den Zeitmittelwerten übereinstimmen, handelt es sich um einen *ergodischen* Prozess.

Ergodischer stochastischer Prozess

Ein stationärer Zufallsprozess $S(t)$ heisst *ergodisch*, wenn Zeit- und Scharmittelwerte übereinstimmen: d. h. für jede Realisierung $s(t)$ des Zufallsprozesses $S(t)$ muss gelten

$$\mu_S = \int_{s=-\infty}^{\infty} s \cdot f(s) \, ds \stackrel{!}{=} \langle s(t) \rangle_T = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=-\frac{T}{2}}^{\frac{T}{2}} s(t) \, dt = m_s$$

$$\sigma_S^2 = \int_{s=-\infty}^{\infty} (s - \mu_S)^2 \cdot f(s) \, ds \stackrel{!}{=} \langle (s(t) - m_s)^2 \rangle_T = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=-\frac{T}{2}}^{\frac{T}{2}} (s(t) - m_s)^2 \, dt$$

wobei $\langle \cdot \rangle_T$ die zeitliche Mittelwertbildung bezeichnet.

Beispiel 8.3.3

Sei A eine auf dem Intervall $[0, 1]$ uniform verteilte Zufallsvariable. Wir definieren den Stochastischen Prozess

$$S(t) = A$$

Den Zeitmittelwert haben wir bereits zu $\langle S(t) \rangle_T = A$ berechnet. Für den Scharmittelwert erhalten wir

$$\mu_S(t) = E[S(t)] = E[A] = \frac{1-0}{2} = \frac{1}{2}$$

Da Scharmittelwert und Zeitmittelwert nicht identisch sind, ist der Prozess $S(t)$ nicht ergodisch.

□

Beispiel 8.3.4

Wir betrachten den stochastischen Prozess $X(t)$, der gegeben ist durch

$$X(t) = A \cos(\omega t + \Theta)$$

wobei A und ω konstant sind und Θ eine gleichmässig verteilte Zufallsvariable über das Intervall $[-\pi, \pi]$ ist. Der Mittelwert $\mu(t)$ berechnet sich wie folgt

$$\mu(t) = E(X(t)) = \int_{-\infty}^{\infty} A \cos(\omega t + \vartheta) f_{\Theta}(\vartheta) \, d\vartheta$$

wobei die Dichtefunktion gegeben ist durch

$$f_{\Theta}(\vartheta) = \begin{cases} \frac{1}{2\pi} & -\pi \leq \vartheta \leq \pi \\ 0 & \text{sonst} \end{cases}$$

Somit ergeben sich für den Mittelwert

$$\mu(t) = \frac{A}{2\pi} \int_{-\pi}^{\pi} \cos(\omega t + \vartheta) d\vartheta = 0$$

und für die Varianz

$$\sigma(t)^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} (A \cos(\omega t + \vartheta) - 0)^2 d\vartheta = \frac{A^2}{2}$$

Da der Mittelwert und die Varianz des betrachteten stochastischen Prozesses $X(t)$ unabhängig von der Zeit t sind, handelt es sich um einen *stationären Zufallsprozesse*.

Der zeitliche Mittelwert von $X(t)$ ergibt

$$\langle x(t) \rangle_{T_0} = \frac{A}{T_0} \int_{-T_0/2}^{T_0/2} \cos(\omega t + \vartheta) dt = 0$$

wobei $T_0 = 2\pi/\omega$. Eine ähnliche Rechnung ergibt für die Varianz

$$\langle \sigma^2(t) \rangle_{T_0} = \frac{A^2}{2}$$

woraus wir schliessen, dass $X(t)$ nicht nur stationär, sondern auch *ergodisch* ist (Scharmittelwerte sind identisch mit den zeitlichen Mittelwerten).

Da $X(t)$ periodisch ist, brauchen wir bei der Berechnung des zeitlichen Mittelwertes blos über eine Periode zu integrieren.

□

Beispiel 8.3.5

Die Maxwell-Boltzmann-Verteilung ist eine Wahrscheinlichkeitsverteilung der statistischen Physik und spielt in der kinetischen Gastheorie eine wichtige Rolle. Sie beschreibt die statistische Verteilung des Betrags $v = |\vec{v}|$ der Teilchengeschwindigkeiten in einem idealen Gas und ist gegeben durch

$$f(v) = 4\pi \left(\frac{m}{2\pi k_B T} \right)^{3/2} v^2 \exp \left(-\frac{mv^2}{2k_B T} \right)$$

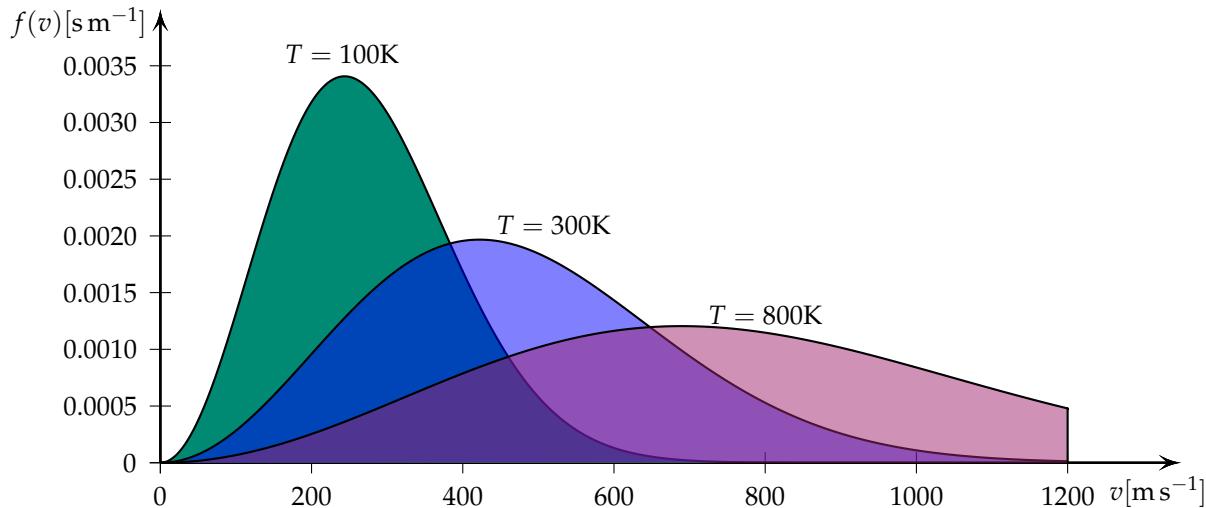


Abbildung 8.14.: Maxwell-Boltzmann Geschwindigkeitsverteilung (Geschwindigkeitsbetrag) für das zweiatomige Stickstoffmolekül (N_2) für drei verschiedene Temperaturen: $T = 100\text{ K}$ (grün), $T = 300\text{ K}$ (blau) und $T = 800\text{ K}$ (violett).

wobei k_B die Boltzmann-Konstante und T die Temperatur in Kelvin und m die Masse des Gasmoleküls bezeichnet. Die Geschwindigkeitsverteilung ist dargestellt in der Abbildung 8.14.

Die mittlere Geschwindigkeit der Gasmoleküle ergibt sich zu

$$\mu_v = \int_0^\infty v f(v) dv = \sqrt{\frac{8k_B T}{\pi m}}$$

Die für die statistische Mechanik grundlegende *Ergodenhypothese* besagt, dass der Zeitmittelwert gleich dem Scharmittelwert einer Messgrösse ist. Im Falle der sich zeitlich ändernden Geschwindigkeit $v(t)$ eines Gasmoleküls ergibt der Zeitmittelwert:

$$\langle v(t) \rangle_T = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t=-\frac{T}{2}}^{\frac{T}{2}} v(t) dt \stackrel{!}{=} \mu_v = \sqrt{\frac{8k_B T}{\pi m}}$$

Um die zeitlich gemittelte Geschwindigkeit eines Gasmoleküls zu bestimmen, brauchen wir also bloss über die Geschwindigkeiten aller in einem idealen Gas befindlichen Gasmoleküle zu einem beliebigen Zeitpunkt zu mitteln.

□

8.4. Mathematische Konzepte für Zeitreihen

Neben der beschreibenden Statistik, d.h., zusammenfassende Kenngrößen und Graphiken, ist das hauptsächliche Ziel der Zeitreihenanalyse, ein mathematisches Modell zu entwickeln, das eine plausible Beschreibung der Versuchsdaten liefert, wie wir sie in Abschnitt 7.1 kennengelernt haben.

Um die Beschreibung des Charakters dieser scheinbar zufällig fluktuierenden Daten in einen statistischen Kontext zu stellen, nehmen wir an, dass Zeitreihen Realisierungen von zeitlich indexierten Zufallsvariablen sind, wie wir dies bereits in Abschnitt 8.3 dargelegt hatten.

In diesem Kapitel werden wir die Theorie der stochastischen Prozesse nun systematisch ausbauen.

Zeitreihen und diskrete stochastische Prozesse

Sei T eine Menge von Zeitpunkten, die gleichweit auseinanderliegen

$$T = \{t_1, t_2, \dots\}$$

1. Ein *diskreter stochastischer Prozess* ist eine Menge von Zufallsvariablen

$$\{X_1, X_2, \dots\}$$

Jede einzelne Zufallsvariable X_i hat eine eindimensionale Verteilungsfunktion F_i und kann zur Zeit t_i beobachtet werden.

2. Eine *Zeitreihe*

$$\{x_1, x_2, \dots\}$$

ist eine Realisierung eines diskreten stochastischen Prozesses $\{X_1, X_2, \dots\}$.

In anderen Worten ist der Wert x_i eine Realisierung der Zufallsvariable X_i , die zur Zeit t_i gemessen wird.

Es ist wichtig zwischen einer Zeitreihe (einer konkreten Beobachtung von Werten) und dem stochastischen Prozess als theoretischem Konstrukt zu unterscheiden, der den zugrundeliegenden Mechanismus der Zeitreihe modelliert und die Werte erzeugt.

Wir illustrieren dies an einem einfachen, wichtigen und bereits bekannten Beispiel.

Beispiel 8.4.1

Nehmen wir an, dass sich eine Person mit konstanter Geschwindigkeit vom Koordinatenursprung in x -Richtung bewegt. Bei jedem Schritt entscheidet die Person zufällig

Kapitel 8. Mathematische Modelle für Zeitreihen

lig, ob sie 1 m nach links oder nach rechts geht. Dies ist der einfachste Fall eines *Random Walk*.

Das probabilistische Modell für diesen Random Walk wäre

1. Wir wählen n unabhängige Bernoulli-Zufallsvariablen

$$D_1, \dots, D_n$$

die die Werte -1 und 1 mit gleicher Wahrscheinlichkeit von $p = 0.5$ annehmen.

2. Wir definieren die Zufallsvariable

$$X_i = D_1 + \dots + D_i$$

für jedes i zwischen 1 und n .

Dann ist

$$X_1, X_2, \dots$$

ein diskreter stochastischer Prozess, der den Random Walk modelliert.

Der folgende **Python**-Code berechnet einen besonderen Fall dieses Prozesses, d.h. eine Zeitreihe $\{x_1, x_2, \dots\}$. Jedesmal, wenn der Code ausgeführt wird, erscheint eine neue Zeitreihe wie in Abbildung 8.15. (zu **R**)

```
import matplotlib.pyplot as plt
import numpy as np

d = np.random.choice(a=[-1,1], size=10000, replace=True)

x = np.cumsum(d)

plt.plot(x)

plt.xlabel("Random Walk")
plt.ylabel("y-Abweichung in [m]")

plt.show()
```

Aus der Definition des Prozesses ist klar, dass die folgende rekursive Definition äquivalent ist:

$$X_i = X_{i-1} + D_i, \quad X_0 = 0$$

Falls in jedem Schritt eine fixe Konstante δ zur Zeitreihe addiert wird, also

$$Y_i = \delta + Y_{i-1} + D_i, \quad Y_0 = 0$$

dann erhalten wir eine Zeitreihe mit einem *Drift*.

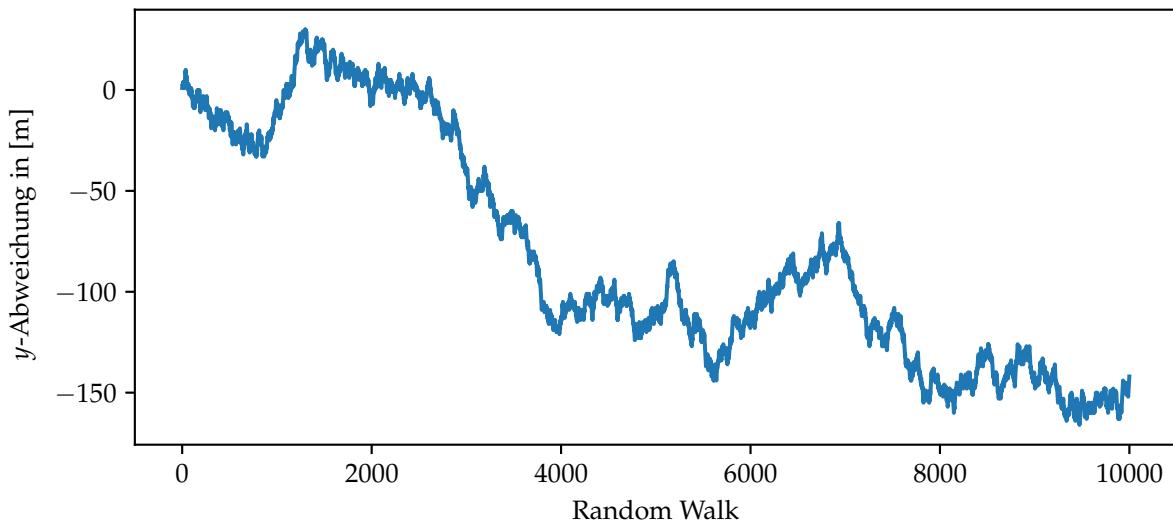


Abbildung 8.15.: Eine Zeitreihe als Beobachtung eines Random Walk.

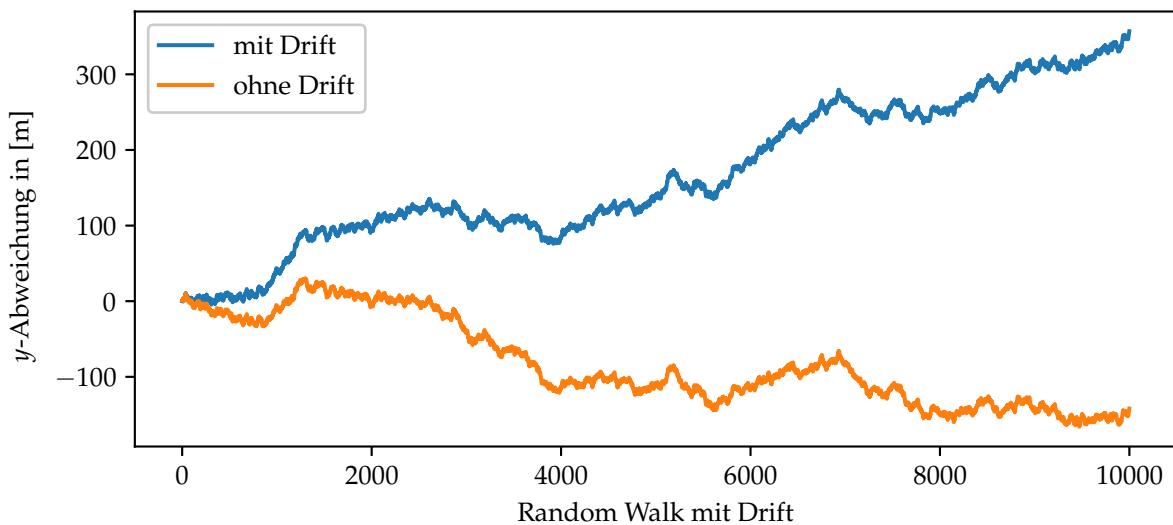


Abbildung 8.16.: Zeitreihe mit und ohne Drift

In Abbildung 8.16 sehen wir eine beobachtete Zeitreihe eines solchen Prozesses. Der Random Walk mit Drift-Modellen wird verwendet, um den Trend einer Zeitreihe zu modellieren.

Wir simulieren diesen Prozess mit einer `for`-Schleife. ([zu R](#))

```

np.random.seed(35)
d = np.random.choice(a=[-1,1], size=10000, replace=True)
delta = 5*10**(-2)
x = np.cumsum(d)

y = np.zeros(10000)

for i in range(1,10000):
    y[i] = delta+y[i-1]+d[i]

plt.plot(y)
plt.plot(x)
plt.xlabel("Random Walk mit Drift")
plt.ylabel("y-Abweichung in [m]")

plt.show()

```

□

Eine Zeitreihe

$$\{x_1, x_2, \dots, x_n\}$$

kann als *eine* Realisierung der multivariaten Zufallsvariablen

$$\{X_1, X_2, \dots, X_n\}$$

aufgefasst werden. Modellierung und Vorhersagen für Zeitreihen kommt dementsprechend der Analyse der Daten von *einer* Beobachtung gleich, was ohne weitere Annahmen über die Zeitreihe unmöglich ist. Wir werden später mit solchen Annahmen arbeiten, aber wir werden zuerst ein Beispiel betrachten, welches ohne diese Annahmen auskommt und somit nicht vorhersehbar ist: das *weisse Rauschen* (*white noise*).

Beispiel 8.4.2

Der Prozess des weissen Rauschens besteht aus unabhängigen, gleich verteilten Zufallsvariablen

$$\{W_1, W_2, \dots, W_n\}$$

wobei alle W_i einen Mittelwert 0 und eine Varianz σ^2 haben. In Abbildung ?? ist weisses Rauschen dargestellt. (zu R)

```

w = np.random.normal(size = 1000)
plt.plot(w)

plt.show()

```

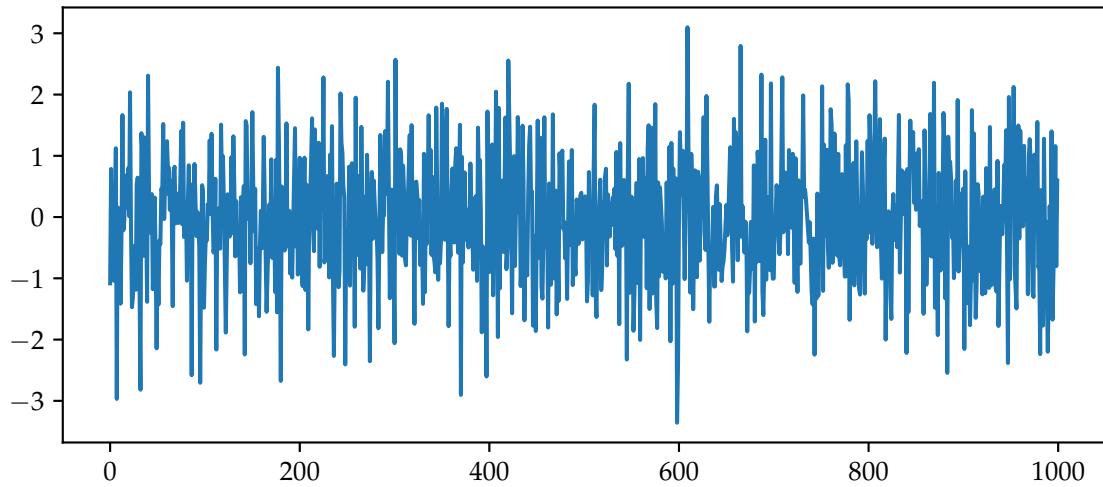


Abbildung 8.17.: Eine Realisierung eines Prozesses des weissen Rauschen.

Sind die Zufallsvariablen W_i zusätzlich normalverteilt, dann sprechen wir von einem *Gauss'schen weissen Rauschen*.

Diese Modelle beschreiben das Rauschen bei Ingenieurproblemen. Der Begriff *weiss* wurde in Analogie zum weissen Licht eingeführt und deutet an, dass alle möglichen periodischen Oszillationen in der Zeitreihe mit gleicher Stärke vorhanden sind.

□

Die Beobachtungen in einem Prozess des weissen Rauschens sind unkorreliert und können mit den gewöhnlichen statistischen Methoden behandelt werden.

Wir fahren mit zwei weiteren Beispielen von diskreten stochastischen Prozessen fort, die aus einem weissen Rauschen erzeugt werden können und die ein *seriell korreliertes* Verhalten aufweisen.

Beispiel 8.4.3

Wenn wir ein *sliding window filter* auf den Prozess des weissen Rauschens

$$\{W_1, W_2, \dots, W_n\}$$

im Beispiel 8.4.2 anwenden, dann erhalten wir einen *moving average*-Prozess. Wählen wir ein Fenster der Länge 3, dann erhalten wir

$$V_i = \frac{1}{3}(W_{i-1} + W_i + W_{i+1})$$

Wir wählen

$$V_1 = W_1 \quad \text{und} \quad V_2 = 0.5(V_1 + V_2)$$

Der resultierende Prozess ist glatter, d.h. die Oszillationen höherer Ordnung werden ausgeglättet. (siehe Abbildung 8.4.3) (zu R)

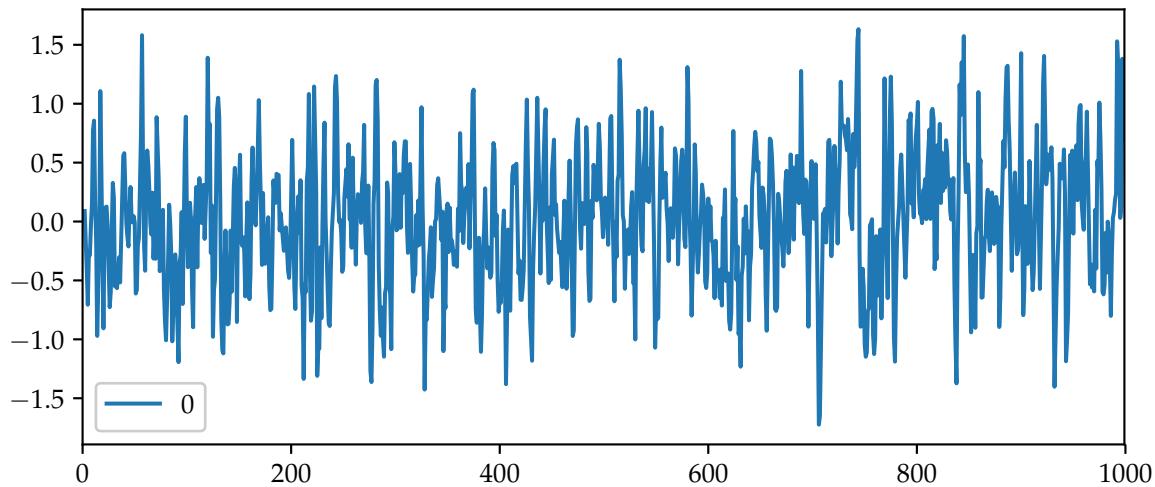


Abbildung 8.18.: Eine Realisierung eines moving average mit Fensterlänge 3.

```
w = DataFrame(np.random.normal(size=1000))

w.rolling(window=3).mean().plot()

plt.show()
```

□

Viele Beispiele von Anwendungsproblemen, wie akustische Zeitreihen in der Sprachanalyse, enthalten dominante oszillierende Komponenten, die sinusförmiges Verhalten aufweisen. Ein mögliches Beispiel um solche quasiperiodischen Daten zu erzeugen, sind *autoregressive Zeitreihen*.

Beispiel 8.4.4

Wir betrachten wieder einen Prozess des weissen Rauschens

$$\{W_1, W_2, \dots, W_n\}$$

im Beispiel 8.4.2. Wir definieren dann rekursive die folgende Reihe

$$X_i = 1.5X_{i-1} - 0.9X_{i-2} + W_i$$

In anderen Worten wird der Wert für den Zeitpunkt i modelliert als Linearkombination der letzten beiden Werte addiert mit einer zufälligen Komponente. So ein Prozess wird *autoregressiv* genannt.

Die Definition der Anfangsbedingungen sind subtil, da der ganze Prozess stark von diesen abhängt. Wir werden vorläufig die Frage der Anfangsbedingungen ignorieren.

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

np.random.seed(35)
d = np.random.choice(a=[-1,1], size=500, replace=True)
y = np.zeros(500)
for i in range(2,500):
    y[i] = 1.5*y[i-1]-0.9*y[i-2]+d[i]
plt.plot(y)
plt.xlabel("Autoregressiver Prozess")
plt.ylabel("y-Abweichung")
plt.title("AR(2)-Prozess")
plt.show()
```

Abbildung 8.19 zeigt eine Realisierung des autoregressiven Prozesses oben. Das oszillierende Verhalten kommt deutlich zum Vorschein.

□

In den Beispielen oben haben wir den Gebrauch von verschiedenen Kombinationen von Zufallsvariablen zur Erzeugung von Zeitreihe motiviert, und zwar zum Zwecke, Anwendungsprobleme damit nachzuahmen. Es ist wichtig, das statistische Verhalten solcher Modelle zu verstehen, um deren Genauigkeit abzuschätzen.

In der Definition eines diskreten stochastischen Prozess $\{X_1, X_2, \dots\}$ haben wir die Existenz einer Verteilungsfunktion $F_i(x)$ für alle Beobachtungen X_i in diesem Prozess postuliert, also

$$P(X_i \leq x) = F_i(x)$$

Die Kenntnis der einzelnen Verteilungen reicht aber nicht, um das serielle Verhalten eines Prozesse zu verstehen, da die Beobachtungen gegenseitig voneinander *abhängen*. Es ist bekannt, dass die vollständige probabilistische Struktur eines solchen

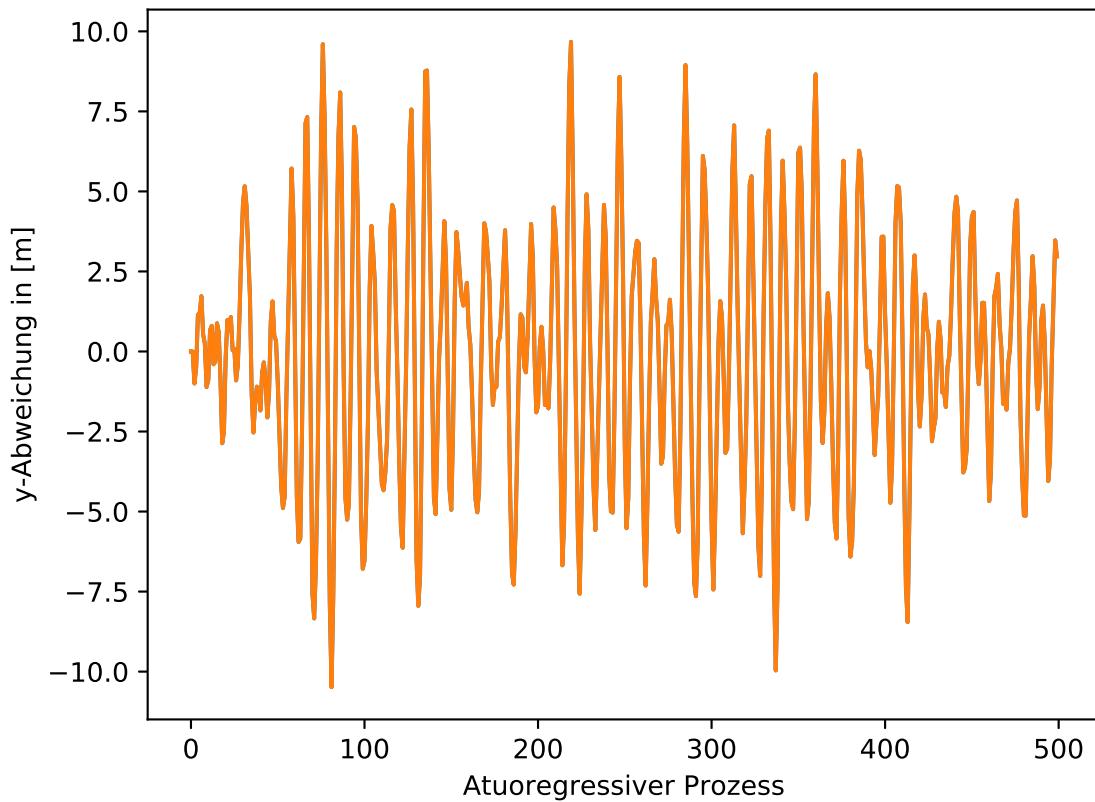


Abbildung 8.19.: Eine Realisierung eines autoregressiven Prozesses beruhend auf den zwei vorangehenden Werten.

Prozesses durch die *gemeinsame Verteilung aller endlichen Ansammlungen* $\{X_{i_1}, \dots, X_{i_n}\}$ aller Beobachtungen gegeben ist. Wir müssen als eine Funktionen F finden, so dass

$$P(X_{i_1} \leq x_1, \dots, X_{i_n} \leq x_n) = F(x_1, \dots, x_n)$$

für alle möglichen Indizes i_1, \dots, i_n .

In der Praxis brauchen wir uns nicht um solche multivariate Verteilungen zu kümmern. Die meiste Information in diesen gemeinsamen Verteilungen können durch Mittelwerte, Varianz und Kovarianz beschrieben werden. Wir werden uns nun auf diese Größen konzentrieren.

8.5. Serielle Korrelation

In Kapitel 7.4.3 haben wir Zeitreihen zerlegt, und zwar in eine Trendkomponente, eine saisonale Komponente und schliesslich in einen Restterm. Sobald wir den Trend und die saisonale Komponente einer Zeitreihe identifiziert haben, können wir diese von der Zeitreihe entfernen. Im Falle der additiven Zerlegung aus 7.4.3 ergibt sich durch Subtraktion der Trend- und der saisonalen Komponente ein zufälliger Restterm der zerlegten Zeitreihe. Diese zufällige Komponente braucht dann aber nicht notwendigerweise aus unabhängigen Zufallsvariablen zu bestehen.

In vielen Fällen sind die aufeinanderfolgenden Variablen miteinander korreliert. Wenn wir im nächsten Kapitel Vorhersagen für Zeitreihen machen möchten, dann können solche Vorhersagen dramatisch verbessert werden, indem wir solche Korrelationen identifizieren.

Des weiteren benötigen wir auch Schätzungen der Korrelation, falls wir ein realistisches Modell für die Zeitreihe konstruieren wollen. Die Korrelationsstruktur eines Zeitreihenmodells ist definiert durch die Autokorrelationsfunktion, und wir schätzen diese durch eine beobachtete Zeitreihe.

Plots von serieller Korrelation (sogenannte „Korrelogramme“, siehe später) werden ebenso umfangreich in Anwendungen der Signalverarbeitung verwendet. Das Paradigma ist ein zugrundeliegendes deterministisches Signal, das durch Rauschen verschlechtert wurde. Signale von Jachten, Flugzeugen oder Raumsonden sind Beispiele. Ziel ist es, das Rauschen zu entfernen, um das ursprüngliche Signal wiederherzustellen.

8.5.1. Mittelwertsfolge und Varianzfolge

Neben den individuellen (also *Rand*-Verteilungen F_i einer Zufallsvariable X_i eines diskreten stochastischen Prozesses definieren wir zuerst die sogenannten ersten und zweiten Momente, um den ganzen Prozess zu analysieren. Wir beginnen mit der **Mittelwertsfolge**.

Mittelwertsfolge

Die Mittelwertsfolge

$$\{\mu(1), \mu(2), \dots\}$$

(oder Mittelwertsfunktion) eines diskreten stochastischen Prozess $\{X_1, X_2, \dots\}$ ist definiert durch die Folge der Mittelwerte:

$$\mu(i) = E[X_i]$$

Beispiel 8.5.1

Wir berechnen die Mittelwertsfolgen für zwei Beispiele aus Abschnitt 8.4.

1. Falls W_i ein Prozess des weissen Rauschens bezeichnet, dann ist $E[X_i] = 0$ für alle $i \geq 1$. Nehmen wir die Mittelwerte in diesem Prozess, so ändert sich folglich nichts am Mittelwert und die Mittelwertsfolgen in einem moving average Prozess wie in Beispiel 8.4.3 ist somit 0.
2. Ist X_i ein Random Walk mit Drift, also $X_0 = 0$

$$X_i = \delta + X_{i-1} + W_i$$

dann erhalten wir

$$\begin{aligned} E[X_1] &= \delta + E[X_0] + E[W_1] = \delta \\ E[X_2] &= \delta + E[X_1] + E[W_2] = 2\delta \\ E[X_3] &= \delta + E[X_2] + E[W_3] = 3\delta \\ &\vdots \end{aligned}$$

Das bedeutet, dass

$$\mu(i) = i\delta$$

□

Der Erwartungswert in dieser Definition ist der Durchschnitt zum Zeitpunkt i über ein *Ensemble* von allen möglichen Zeitreihen, die durch das zugehörige Zeitreihenmodell produziert werden können (siehe Abbildung 8.20).

Wenn wir ein Modell für die Zeitreihe haben, so können wir die Zeitreihe simulieren. Sind allerdings historische Daten vorhanden, so haben wir nur diese Zeitreihe. Alles was wir dann tun können, ohne Annahme einer mathematischen Struktur für den Trend, ist den Mittelwert durch die zugehörige Beobachtung des Stichprobenpunktes selbst zu schätzen.

In der Praxis machen wir Schätzungen für den Trend und die saisonalen Effekte unserer Daten, die wir dann von den Daten entfernen (mit `stldecompose`). Was übrig bleibt, ist die zufällige Komponente. Dann wird ein Zeitreihenmodell mit konstanten Mittelwerten oft angemessen sein. Falls die Mittelwertsfolge konstant ist, sagen wir, dass die Zeitreihe *stationär* im Mittelwert ist.

Die Varianzfolge für ein Zeitreihenmodell, das stationär im Mittelwert ist, lautet

$$\sigma^2(i) = E[(X_i - \mu)^2]$$

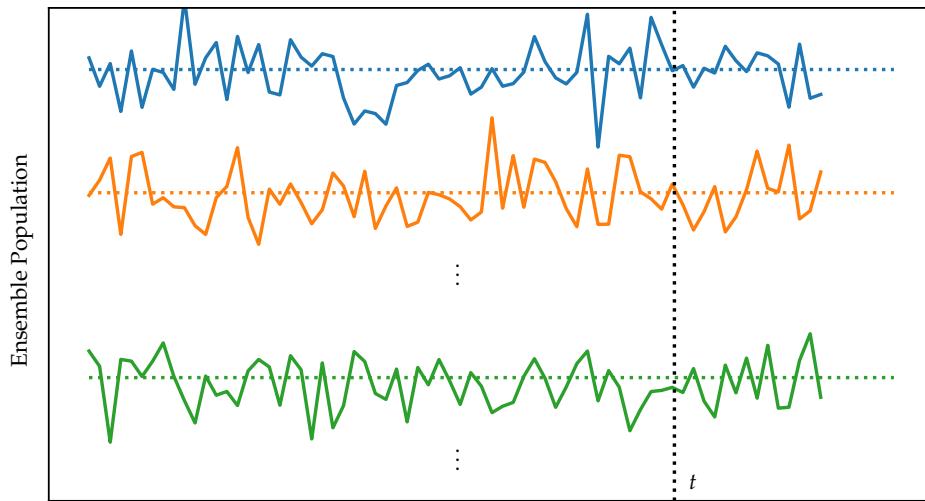


Abbildung 8.20.: Mittelwert über alle möglichen Zeitreihen zum Zeitpunkt t .

Diese kann im Prinzip verschiedene Werte zu jeder Zeit annehmen. Wir können die Varianz für einen Zeitpunkt allerdings nicht schätzen, wenn nur eine Zeitreihe vorliegt.

Unter der Annahme, dass das Modell stationär in der Varianz ist, kann die Populationsvarianz σ^2 durch die Stichprobenvarianz

$$\text{var}_x = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

geschätzt werden.

8.5.2. Autokovarianz und Autokorrelation

Bevor wir *Autokovarianz* und *Autokorrelation* allgemein definieren, repetieren wir das Konzept von *empirischer Kovarianz* und *empirischer Korrelation*.

Empirische Kovarianz und Korrelation

Empirische Kovarianz und Korrelation

Für Stichproben $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ lautet die *empirische Kovarianz*:

$$\text{cov}_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

Falls $x = y$, so gilt

$$\text{cov}_{xx} = \frac{\sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})}{n - 1}$$

und dies ist gerade die *empirische Varianz* von x :

$$\text{cov}_{xx} = \text{var}_x = s_x^2$$

wobei s_x^2 die empirische Varianz bezeichnet.

Wir wollen uns überzeugen, dass die Kovarianz in der Tat den linearen Zusammenhang erfasst.

Beispiel 8.5.2

In Abbildung 8.21 sind einige Datenpunkte eingezeichnet, die mehr oder weniger einer Geraden folgen. Es wurden noch die zu den Koordinatenachsen parallelen Geraden \bar{x} und \bar{y} eingezeichnet.

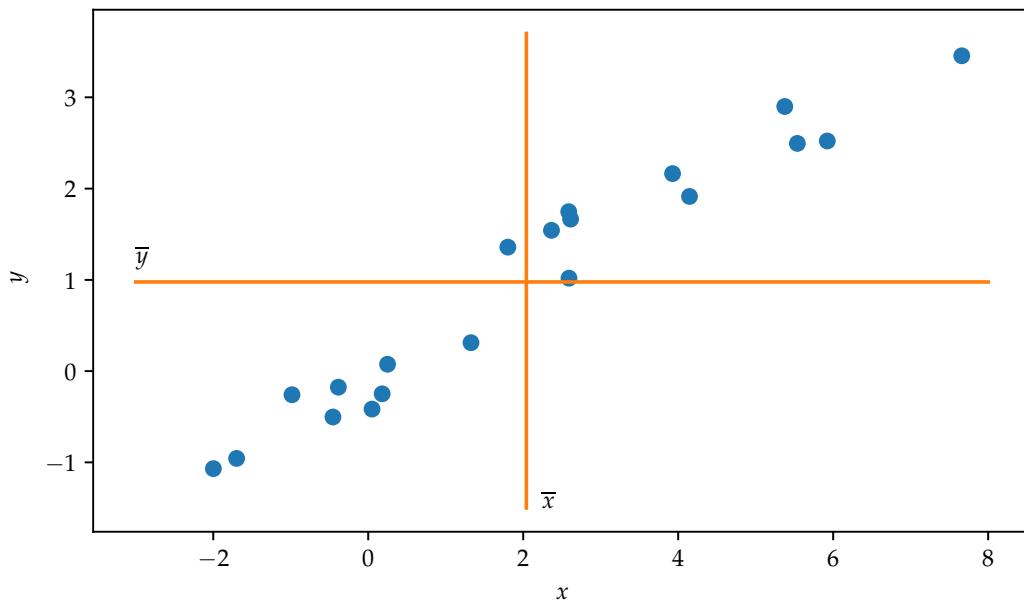


Abbildung 8.21.: Punkte, die fast auf einer Geraden liegen.

Zur Veranschaulichung subtrahieren wir von den x -Koordinaten den Mittelwert \bar{x} und von den y -Koordinaten den Mittelwert \bar{y} . Wir erhalten dann Abbildung 8.22.

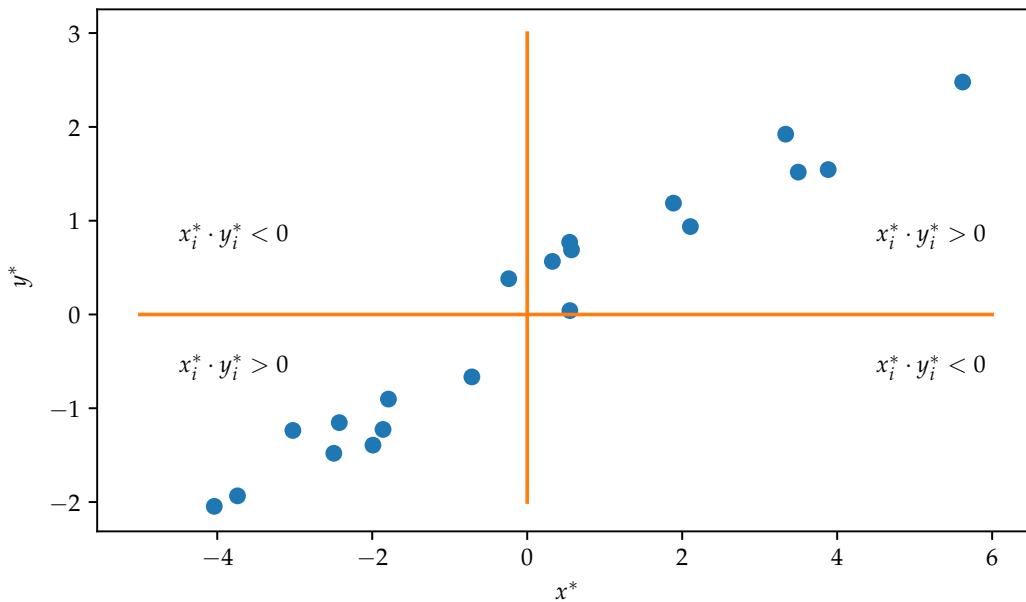


Abbildung 8.22.: Punkte, die fast auf einer Geraden liegen.

Die empirische Kovarianz für diese Punkte lautet nun

$$\text{cov}_{x^*y^*} = \frac{\sum_{i=1}^n x_i^* y_i^*}{n - 1}$$

Im Zähler werden also die Produkte $x_i^* y_i^*$ aufaddiert. Im I. und III. Quadranten sind diese Produkte positiv, im II. und IV. Quadranten negativ.

Da in unserem Beispiel die Punkte praktisch alle im I. und III. Quadranten liegen, so wird $\text{cov}_{x^*y^*}$ sicher positiv:

$$\text{cov}_{x^*y^*} > 0$$

Liegen die Punkte eher auf einer fallenden Geraden, so liegen die Punkte meistens im II. und IV. Quadranten. Der Wert von $\text{cov}_{x^*y^*}$ wird dann sicher negativ:

$$\text{cov}_{x^*y^*} < 0$$

Was passiert nun, wenn die Punkte keinen Zusammenhang aufweisen (siehe Abbildung 8.23)?

In diesem Fall heben sich die Produkte $x_i^* y_i^*$ über alle Punkte aufaddiert in etwa auf, da die Hälfte aller Punkte im I. und III. Quadranten (Produkte positiv) und die andere

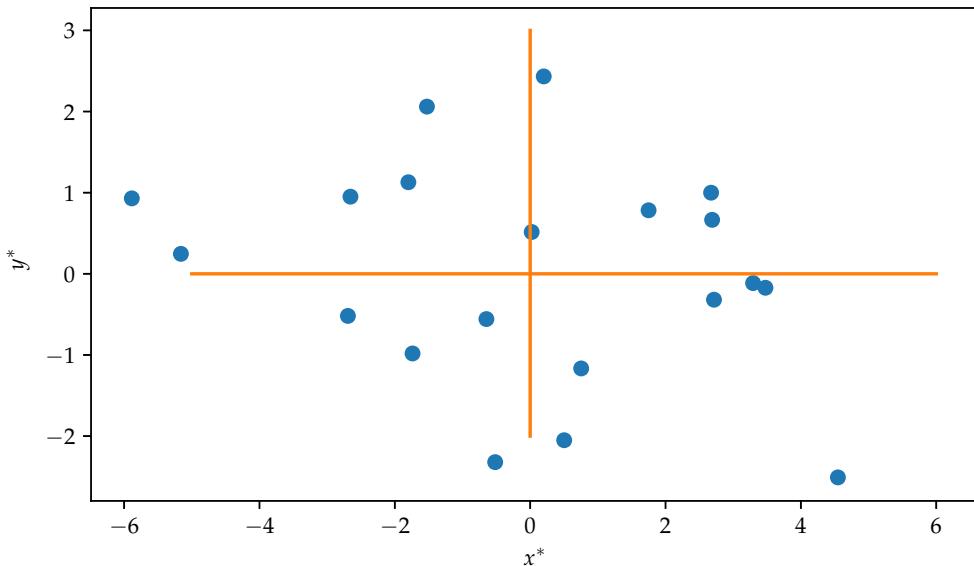


Abbildung 8.23.: Von den Koordinaten wurden die jeweiligen Mittelwerte subtrahiert.

Hälften im II. und IV. Quadranten (Produkte negativ) liegen. Zudem sind die Produkte betragsmässig ähnlich. Damit gilt

$$\text{cov}_{x^*y^*} \approx 0$$

Wie sieht es aus, wenn wir einen quadratischen Zusammenhang haben (siehe Abbildung 8.24)?

Auch hier heben sich die Beträge der Produkte links und rechts von der y -Achse auf. Es gilt

$$\text{cov}_{x^*y^*} \approx 0$$

Die Kovarianz erkennt also nur *lineare* Zusammenhänge.

□

Falls es keinen linearen Zusammenhang gibt, so kann die Kovarianz 0 sein, obwohl ein nichtlinearer Zusammenhang besteht. Wir sollten uns *nie* ausschliesslich auf den Wert der Kovarianz verlassen, sondern *immer* die Plots auf nichtlineare Zusammenhänge überprüfen.

Beispiel 8.5.3

Benzoapryrene ist ein krebsförderndes Kohlenwasserstoffmolekül, welches das Produkt von unvollständiger Verbrennung ist. Eine Quelle von Benzoapryrene und Kohlenmonoxid sind Autoabgase. Colucci und Begeman (1971) analysierten 16 Luftproben, die am Herald Square in Manhattan genommen wurden.

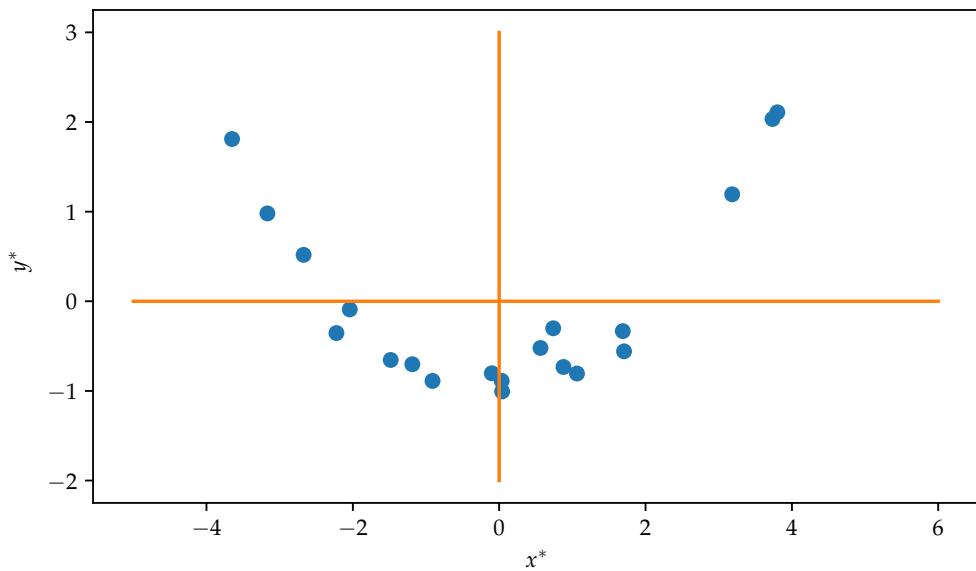


Abbildung 8.24.: Punkte, die fast auf einer Geraden liegen.

Sie zeichneten die Kohlenmonoxidkonzentration (in parts per million) und die Benzapyrenekonzentration (in Mikrogramm pro 1000 Kubikmeter) für jede Probe auf. In der Datei **Herald.dat** sind die Daten aufgelistet:

```
## -c:5: FutureWarning: read_table is deprecated, use read_csv instead, passing
##           CO  Benzoa
## 0    2.8      0.5
## 1   15.5      0.1
## 2   19.0      0.8
## 3    6.8      0.9
## 4    5.5      1.0
```

Wir stellen nun diese Daten in Abbildung 8.25 graphisch dar.

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from pandas import Series, DataFrame

df = pd.read_table("Herald.dat")

df.plot(kind="scatter", x="CO", y="Benzoa")

plt.plot((2.5, 20), (df["Benzoa"].mean(), df["Benzoa"].mean()), c="orange")

plt.plot((df["CO"].mean(), df["CO"].mean()), (0, 10), c="orange")
```

```
plt.ylabel("Benzoapyrene")
plt.show()
```

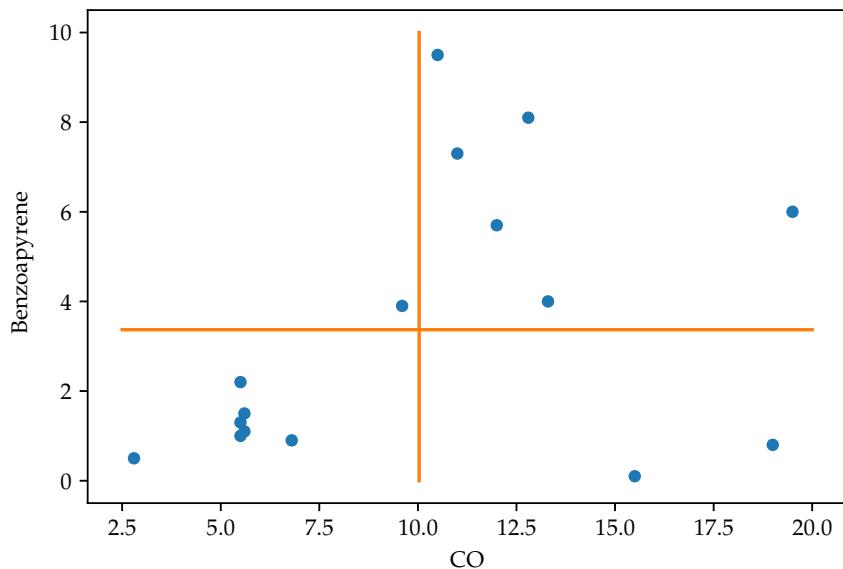


Abbildung 8.25.: Punkte, die fast auf einer Geraden liegen.

Mit **Python** können wir nun die empirische Kovarianz der Herald Square Paare berechnen.

```
df.cov()

## -c:5: FutureWarning: read_table is deprecated, use read_csv instead, passing
##                      CO      Benzoa
## CO      25.811625  5.511042
## Benzoa  5.511042  9.331625
```

Der Output von **Python** ist eine sogenannte *Kovarianzmatrix*. Der für uns wichtige Wert ist in Zeile **CO** und Spalte **Benzoa** (oder umgekehrt) aufgeführt. Es gilt also

$$\text{cov}_{\text{df}["\text{CO}"]\text{df}["\text{Benzoa}"]} = 5.51$$

Der Wert 25.8 in der Kovarianzmatrix entspricht der Varianz von **CO**. Entsprechend ist der Wert 9.33 die Varianz von **Benzoa**.

□

Das Problem an Kovarianz ist, dass sie schlecht oder kaum zu interpretieren ist. So hat die Kovarianz im Beispiel 8.5.3 die Einheit

$$\text{ppm} \cdot \mu\text{g}/1000\text{m}^3$$

was fast unmöglich zu interpretieren ist.

Um dies zu korrigieren, führen wir mit der *empirische Korrelation* ein neues dimensionsloses Mass ein, das den linearen Zusammenhang zwischen Stichprobenpaaren (x_i, y_i) misst. Dies erreichen wir, indem wir die Kovarianz standardisieren. Dabei wird die Kovarianz durch die Standardabweichung von x und y geteilt. Die Korrelation nimmt dann Werte zwischen -1 und 1 an, wobei ein Wert von 0 *keinen linearen Zusammenhang* aufzeigt.

Empirische Korrelation

Die empirische Korrelation r für die Koordinatenpaare (x_i, y_i) ist wie folgt definiert:

$$r_{xy} = \frac{\text{cov}_{xy}}{s_x \cdot s_y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n - 1) \cdot s_x \cdot s_y}$$

wobei s_x und s_y die empirischen Standardabweichungen von den Stichproben x_i und y_i bezeichnen.

Ist $r_{xy} = 1$, so liegen alle Punkte auf einer steigenden Geraden. Für $r_{xy} = -1$ liegen die Punkte alle auf einer fallenden Geraden.

Mit **Python** berechnen wir die Korrelation mit dem Zusatz `.corr()`.

Beispiel 8.5.4

Für den Datensatz **Herald** berechnet sich die Korrelation wie folgt:

```
df.corr()
```

```
## -c:5: FutureWarning: read_table is deprecated, use read_csv instead, passing
## Traceback (most recent call last):
##   File "<string>", line 5, in <module>
##   File "/usr/local/lib/python3.6/site-packages/pandas/io/parsers.py", line 70
##       return _read(filepath_or_buffer, kwds)
##   File "/usr/local/lib/python3.6/site-packages/pandas/io/parsers.py", line 42
##       parser = TextFileReader(filepath_or_buffer, **kwds)
##   File "/usr/local/lib/python3.6/site-packages/pandas/io/parsers.py", line 89
##       self._make_engine(self.engine)
##   File "/usr/local/lib/python3.6/site-packages/pandas/io/parsers.py", line 11
```

```

##     self._engine = CParserWrapper(self.f, **self.options)
## File "/usr/local/lib/python3.6/site-packages/pandas/io/parsers.py", line 18
##     self._reader = parsers.TextReader(src, **kwds)
## File "pandas/_libs/parsers.pyx", line 387, in pandas._libs.parsers.TextRea
## File "pandas/_libs/parsers.pyx", line 705, in pandas._libs.parsers.TextRea
## FileNotFoundError: [Errno 2] File b'../../../../Themen/Time_Series_Introduction'

```

Wir erhalten die Korrelationsmatrix. Wieder ist der Eintrag für Zeile **CO** und Spalte **Benzoa** entscheidend. Es gilt:

$$r_{\text{df}["\text{CO}"], \text{df}["\text{Benzoa}"]} = 0.3551$$

Obwohl die Korrelation klein ist, gibt es trotzdem eine physikalische Erklärung für die Korrelation, da beide Stoffe das Produkt von unvollständiger Verbrennung sind.

□

Eine Korrelation von 0.36 ergibt typischerweise einen leichten visuellen Eindruck, dass y sich vergrössert, wenn x vergrössert wird. Die Punkte streuen allerdings stark.

Empirische Autokovarianz und Autokorrelation

Der Mittelwert und die Varianz spielen eine wichtige Rolle im Studium von statistischen Verteilungen, da sie zwei Schlüsseleigenschaften von Verteilungen zusammenfassen, nämlich die zentrale Lage und die Streuung. In ähnlicher Weise spielt beim Studium von Zeitreihen die *serielle Korrelation* eine wichtige Rolle. Eine Korrelation einer Variable X_i mit sich selbst zu verschiedenen Zeiten wird *Autokorrelation* oder *serielle Korrelation* genannt.

Empirische Autokovarianzfunktion und Autokorrelationsfunktion

Die *empirische Autokovarianzfunktion* g_k als Funktion vom lag k ist definiert als

$$g_k = \frac{1}{n} \sum_{i=1}^{n-k} (x_i - \bar{x})(x_{i+k} - \bar{x})$$

wobei $g_{-k} = g_k$ für $h = 0, 1, 2, \dots, n-1$.

Die *empirische Autokorrelationsfunktion* mit lag k ist dann definiert durch

$$r_k = \frac{g_k}{g_0}$$

Bemerkungen:

- i. Die Autokovarianzfunktion hängt von der Skala der Zeitreihe ab, weshalb es sehr mühsam ist, die Kovarianzstruktur von zwei Zeitreihen miteinander zu vergleichen. Die Autokorrelationsfunktion ist daher wesentlich praktischer, da diese in einer Graphik mit normalisierter y -Achse resultiert.
- ii. Die Autokovarianz g_0 mit lag 0 ist die Varianz, allerdings mit Nenner n anstatt $n - 1$. Es wird immer der Nenner n zur Berechnung von g_k verwendet, obwohl es im Zähler nur $n - k$ Summanden hat.

Wir wollen die Berechnungen in **Python** an folgendem Beispiel aufzeigen.

Beispiel 8.5.5

Wir haben eine Zeitreihe, die die Wellenhöhe (in mm relativ zum Level von ruhigem Wasser) angibt. Gemessen wird jeweils im Zentrum eines Wellentanks.

Das Stichprobenintervall war 0.1 s und es wurde über einen Zeitraum von 39.7 s gemessen. Die Wellen wurden durch einen Wellengenerator erzeugt, der durch eine Pseudozufallszahlengenerator gesteuert wurde. Ziel war, Wellengang im Meer zu simulieren. Es gibt keinen Trend und keine saisonale Periode. Somit ist es vernünftig anzunehmen, dass die Zeitreihe die Realisierung eines stationären Prozesses ist.

```
## -c:5: FutureWarning: read_table is deprecated, use read_csv instead, passing
##      waveht
## 0      367
## 1      407
## 2     -255
## 3     -515
## 4     -500
```

Wir stellen nun diese Daten in Abbildung 8.26 graphisch dar. In dieser Abbildung sehen wir zunächst, dass es keine Ausreisser gibt.

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from pandas import Series, DataFrame

df = pd.read_table("wave.dat")

df.plot()

plt.xlabel("Time")
plt.ylabel("Wave height (mm)")

plt.show()
```

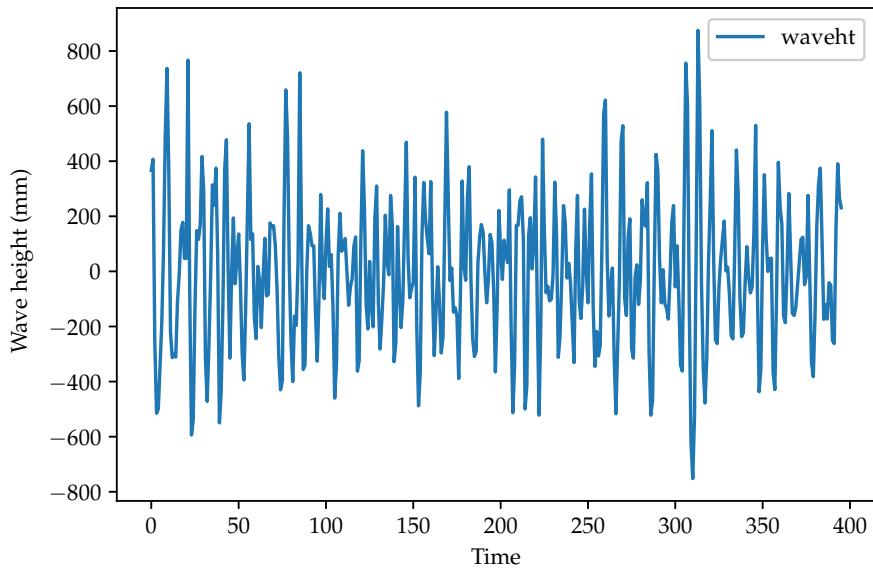


Abbildung 8.26.: Messresultate der Wellenhöhe über eine Zeitspanne von 39.7 s.

Die Graphik in Abbildung 8.27 zeigt nur die ersten 60 Wellenlängen. Wir sehen, dass aufeinanderfolgende Werte sehr ähnlich sind, obwohl die Wellen zufällig erzeugt wurden. Die Form gleicht einem bewegten Meer mit einer Quasi-Periode, aber es gibt keine feste Frequenz.

```
df.loc[0:59, :].plot()
plt.xlabel("Time")
plt.ylabel("Wave height (mm)")
plt.show()
```

Die Autokorrelationskoeffizienten von `waveht` sind in einem Vektor `acf(df["waveht"])` gespeichert.

```
from statsmodels.tsa.stattools import acf
acf(df["waveht"])[1]
```

Für den lag 1 ist die Autokorrelation für `waveht`

```
acf(df["waveht"])[1]
## -c:3: FutureWarning: read_table is deprecated, use read_csv instead, passing
## 0.4702563961883794
```

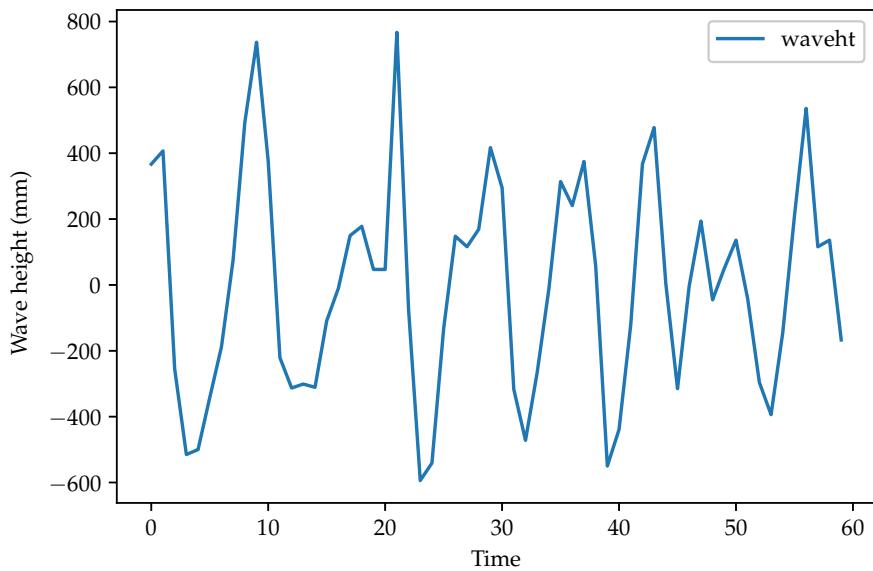


Abbildung 8.27.: Messresultate der Wellenhöhe über eine Zeitspanne von 6 s.

Ein Streudiagramm wie in Abbildung 8.25 ergänzt die Berechnung des Korrelationskoeffizienten und macht uns allenfalls auf nichtlineare Muster aufmerksam. In ähnlicher Weise können wir Streudiagramme für einen festen lag zeichnen. Dazu verwenden wir den `lag_plot` aus Beispiel 7.4.6. Für lag 1 ist das Streudiagramm in Abbildung 8.28 dargestellt.

```
from pandas.plotting import lag_plot

df = pd.read_table("wave.dat")

lag_plot(df,1)

plt.xlabel("x_t")
plt.ylabel("x_(t+k)")

plt.show()
```

Es zeigt sich, dass Wellenhöhen, die durch 0.1 s getrennt sind, ähnliche Höhen aufweisen. In Abbildung 8.29 sind Streudiagramme für weitere lag-Werte aufgeführt.

Für lag 2 beobachten wir eine sehr schwache Korrelation, für lag 3 eine sichtbare negative Korrelation. Bei lag 5 und lag 10 streuen die Diagramme sehr stark. Dies wird auch durch die Autokorrelationswerte des `acf`-Vektors bestätigt.

```
acf(df["waveht"])[[2,3,5,10]]
```

```
## -c:3: FutureWarning: read_table is deprecated, use read_csv instead, passing
```

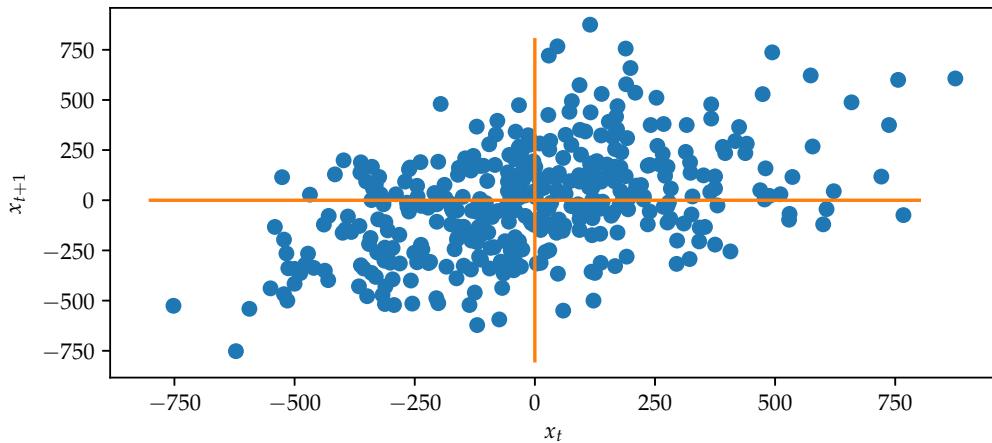


Abbildung 8.28.: Paare von Wellenhöhen durch lag 1 (0.1 s) getrennt.

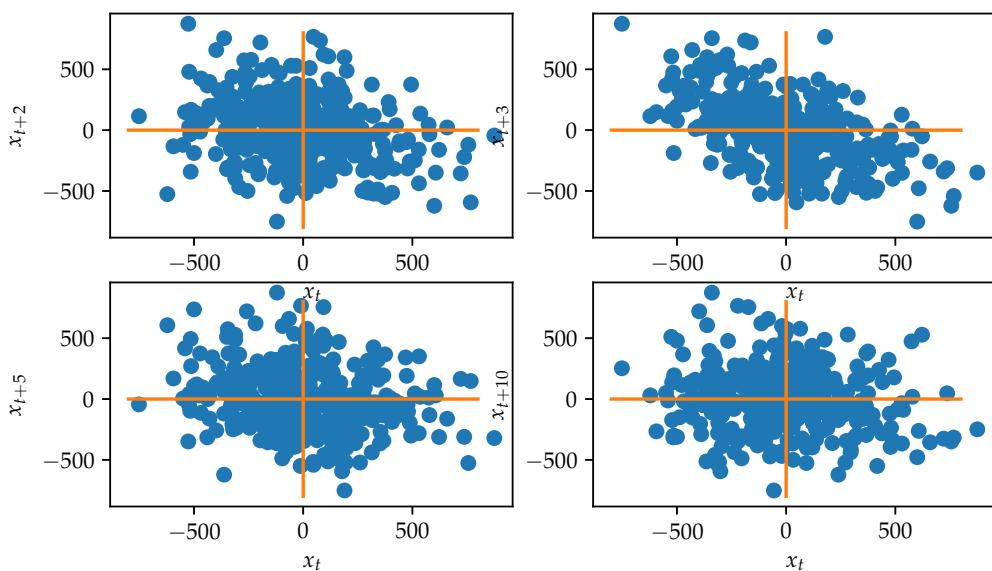


Abbildung 8.29.: Streudiagramm von Wellenhöhen mit lag 2, 3, 5, 10.

```
## [-0.26291153 -0.49891702 -0.21499293 -0.07431329]
```

□

Korrelogramm

Die Bibliothek **statsmodels** hat noch eine Plot-Funktion, die die Autokorrelationsswerte r_k in Abhängigkeit vom lag k aufzeichnet.

Beispiel 8.5.6

Für unseren Datensatz **wave.dat** ist das Korrelogramm in Abbildung 8.30 dargestellt.

```
from statsmodels.graphics.tsplots import plot_acf

plot_acf(df, lags=25)

plt.show()
```

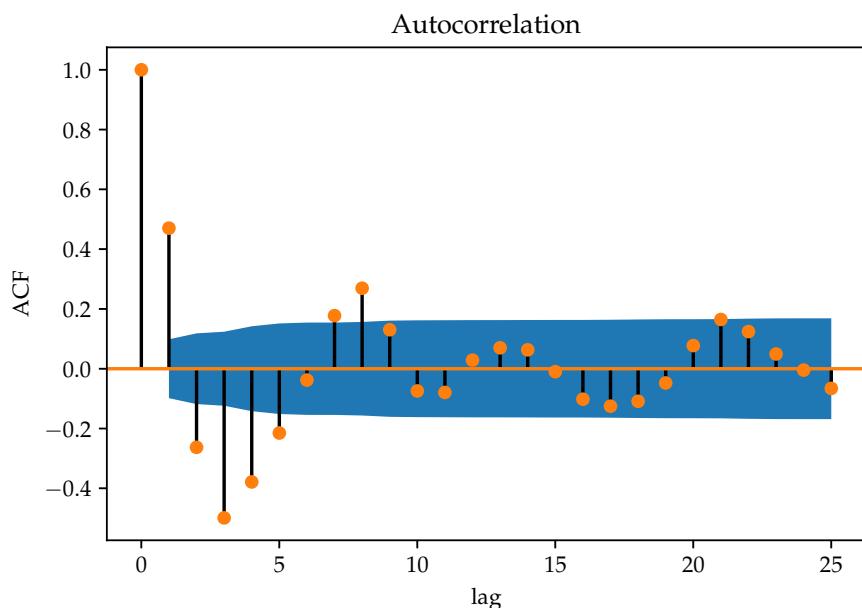


Abbildung 8.30.: Korrelogramm für die Wellenhöhen als Funktion vom lag k .

□

Im Allgemeinen haben Korrelogramme folgende Merkmale:

- Entlang der x -Achse ist der lag k aufgeführt und entlang der y -Achse die Autokorrelationskoeffizienten r_k für jeden lag. Einheit des lags ist das Stichprobenintervall (0.1 s beim Beispieldatensatz **wave.dat**). Die Korrelation ist dimensionslos, somit gibt es keine Einheit für die y -Achse.

Kapitel 8. Mathematische Modelle für Zeitreihen

- Der Autokorrelationskoeffizient für den lag 0 ist immer 1 und im Plot ganz links aufgeführt. Deren Einbeziehung hilft uns, die anderen Autokorrelationskoeffizienten mit dem theoretischen Maximum zu vergleichen.
- Die blaue Fläche entspricht dem Vertrauensintervall für alle Autokorrelationskoeffizienten r_k für weisses Rauschen. Es kann gezeigt werden, dass im Falle von weissem Rauschen, also

$$r_k = 0$$

für $k \neq 0$, die Verteilung eines Autokorrelationskoeffizienten r_k annähernd normalverteilt mit Mittelwert $-\frac{1}{n}$ und Standardabweichung

$$\sigma_r = \frac{1}{\sqrt{n}}$$

ist. Dies kann nun dafür verwendet werden, um zu überprüfen, ob die Zeitreihe von weissem Rauschen stammt. Falls die Zeitreihe weiss ist, dann sollten Autokorrelationskoeffizienten in rund 95 % der Fälle im Intervall

$$-\frac{1}{n} \pm \frac{2}{\sqrt{n}}$$

liegen. Diese Begrenzungslinien, die parallel zur x -Achse verlaufen und die die blaue Fläche einschliessen, werden automatisch in **Python** durch die Funktion **plot_acf** eingezeichnet.

Liegt ein Autokorrelationskoeffizient r_k für einen gegebenen lag k ausserhalb dieser Linien, so verwerfen wir auf dem 5 % Signifikanzniveau die Nullhypothese $r_k = 0$, also dass die Zeitreihe weisses Rauschen ist. Allerdings sollten wir vorsichtig im Falle von multiplen Hypothesentests sein. Auch wenn in Tat und Wahrheit $r_k \neq 0$ für alle lags k , so erwarten wir, dass 5 % der Autokorrelationskoeffizienten r_k ausserhalb dieser Linien liegen.

- Im Falle einer Zeitreihe mit saisonaler Periode, z.B. bei monatlichen Zeitreihen, deutet eine signifikante Autokorrelation der Restreihe beim lag 12 darauf hin, dass die saisonale Anpassung nicht genügend erreicht wurde.
- Eine lag 1 Autokorrelation von 0.1 impliziert, dass eine lineare Abhängigkeit von x_i und x_{i-1} nur $0.1^2 = 0.01 = 1\%$ der Variabilität von x_i erklärt. Es ist ein oft gemachter Fehlschluss, dass statistisch signifikante Resultate als wichtig betrachtet werden, obwohl sie fast keine praktische Bedeutung haben.
- Falls das Korrelogramm eine klare Struktur hat, wie z.B. eine gedämpfte Cosinuswelle, dann ist dies typisch für Korrelogramme von Zeitreihen, die durch ein autoregressives Modell 2. Ordnung erzeugt werden. Wir werden solche Modelle später behandeln.

Theoretische Autokovarianz und Autokorrelation

Nun wollen wir uns den theoretischen Definitionen von Autokovarianz und Autokorrelation zuwenden.

Autokovarianz und Autokorrelation

Sei $\{X_1, X_2, \dots\}$ ein diskreter stochastischer Prozess.

1. Die *Autokovarianz* γ_X ist definiert durch

$$\gamma_X(i, j) = \text{Cov}(X_i, X_j) = E[(X_i - \mu(i))(X_j - \mu(j))]$$

2. Die *Autokorrelation* ρ_X ist definiert durch

$$\rho_X(i, j) = \frac{\gamma_X(i, j)}{\sqrt{\gamma_X(i, i)\gamma_X(j, j)}}$$

Falls der Kontext klar ist, schreiben wir den Index X nicht. Eine wichtige Eigenschaft sowohl der Autokovarianz und der Autokorrelation ist die Symmetrie, also insbesondere

$$\gamma(i, j) = \gamma(j, i)$$

Die Autokovarianz misst die *lineare Abhängigkeit* von zwei Punkten im selben Prozess, beobachtet zu verschiedenen Zeitpunkten. Falls die Zeitreihe sehr glatt ist, dann ist die Autokovarianz gross, auch wenn i und j weit auseinander liegen.

Man beachte, dass wenn

$$\gamma(i, j) = 0$$

dies nur bedeutet, dass X_i und X_j nicht linear abhängig sind, sie können aber trotzdem nicht linear verknüpft sein. Für $i = j$ wird die Autokovarianz zur Varianz von X_i .

Die Autokorrelation kann im gleichen Sinne beschrieben werden, diese ist allerdings normalisiert, heisst also

$$\rho(i, j) \in [-1, 1]$$

Gibt es also einen linearen Zusammenhang zwischen X_i und X_j , dann ist

$$\rho(X_i, X_j) = \pm 1$$

Genauer: Falls

$$X_i = \beta_0 + \beta_1 X_j$$

dann ist die Autokorrelation 1 falls $\beta_1 > 0$, ansonsten -1 . Die Autokorrelation gibt ein grobes Mass an, wie die Reihe zur Zeit i durch den Wert der Reihe zur Zeit j vorhergesagt werden kann.

Wir können die Autokovarianz und die Autokorrelation für einige Prozesse in Abschnitt 8.4 berechnen.

Beispiel 8.5.7

- Der Prozess des weissen Rauschens aus Beispiel 8.4.2 hat die Autokovarianzfunktion

$$\gamma(i, j) = \begin{cases} 0 & \text{falls } i \neq j \\ \sigma^2 & \text{falls } i = j \end{cases}$$

Entsprechend ist die Autokorrelation 1 falls $i = j$ und 0 sonst.

- Wir berechnen die Autokovarianz des moving average Prozesses in Beispiel 8.4.3. Aus der Definition der Autokovarianz ist klar, dass

$$\gamma(i, j) = \text{Cov}(X_i, X_j) = \text{Cov}\left(\frac{1}{3}(W_{i-1} + W_i + W_{i+1}), \frac{1}{3}(W_{j-1} + W_j + W_{j+1})\right).$$

Falls $i = j$, dann

$$\begin{aligned} \text{Cov}(X_i, X_i) &= \frac{1}{9} \text{Cov}(W_{i-1} + W_i + W_{i+1}, W_{i-1} + W_i + W_{i+1}) \\ &= \frac{1}{9} (\text{Cov}(W_{i-1}, W_{i-1}) + \text{Cov}(W_i, W_i) + \text{Cov}(W_{i+1}, W_{i+1})) \\ &= \frac{3\sigma^2}{9} \end{aligned}$$

Dies folgt aus der Tatsache, dass W_i, W_{i-1} und W_{i+1} gegenseitig unkorreliert sind. Für $i+1 = j$ finden wir analog

$$\begin{aligned} \text{Cov}(X_i, X_{i+1}) &= \frac{1}{9} \text{Cov}(W_{i-1} + W_i + W_{i+1}, W_i + W_{i+1} + W_{i+2}) \\ &= \frac{1}{9} (\text{Cov}(W_i, W_i) + \text{Cov}(W_{i+1}, W_{i+1})) \\ &= \frac{2\sigma^2}{9} \end{aligned}$$

Zusammenfassend

$$\gamma(i, j) = \begin{cases} \frac{3\sigma^2}{9} & \text{falls } i = j \\ \frac{2\sigma^2}{9} & \text{falls } |i - j| = 1 \\ \frac{\sigma^2}{9} & \text{falls } |i - j| = 2 \\ 0 & \text{else} \end{cases}$$

Dies zeigt, dass Glättung des weissen Rauschens eine nichttriviale Autokovarianzstruktur einführt. Es ist bemerkenswert, dass die Autokovarianz nur vom Abstand der Beobachtungen abhängt, aber nicht von deren Wert.

Kapitel 8. Mathematische Modelle für Zeitreihen

Wir berechnen schliesslich noch die Autokorrelation

$$\rho(i, j) = \frac{\gamma(i, j)}{\sqrt{\gamma(i, i)\gamma(j, j)}} = \frac{\gamma(i, j)}{\gamma(i, i)}$$

Wir erhalten

$$\rho(i, j) = \begin{cases} 1 & \text{falls } i = j \\ \frac{2}{3} & \text{falls } |i - j| = 1 \\ \frac{1}{3} & \text{falls } |i - j| = 2 \\ 0 & \text{else} \end{cases}$$

Wir können dieses Resultat auch empirisch überprüfen.

```
import matplotlib.pyplot as plt
import pandas as pd
from pandas import DataFrame
from pandas import Series
import numpy as np
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.tsa.stattools import acf
w = DataFrame(np.random.normal(size=1000))

MA = DataFrame(w.rolling(window=3).mean()).dropna()
plot_acf(MA, lags=12, c="C1")
plt.vlines(x=2.1, ymin=0, ymax=1/3, color="red", linestyle='--', label="Gesamt")
plt.vlines(x=1.1, ymin=0, ymax=2/3, color="red", linestyle='--')
plt.vlines(x=0.1, ymin=0, ymax=1, color="red", linestyle='--')

plt.legend()
```

Wie wir in Abbildung 8.31 erkennen können, liegen ab dem dritten lag alle Autokorrelationskoeffizienten im 95 % Vertrauensband.

3. Schliesslich berechnen wir noch die Autokovarianz des Random Walks in Beispiel 8.4.1. Wir erinnern uns, dass wir den Random Walk Prozess X_i definiert haben als Summe von unabhängigen Bernoulli Zufallsvariablen $X_i = D_1 + \dots + D_i$ jede mit Wahrscheinlichkeit $p = 0.5$. Damit ist die Varianz für jedes D_i

$$\sigma^2 = p(1 - p) = 0.25$$

Damit finden wir

$$\gamma(i, j) = \text{Cov} \left(\sum_{k=0}^i D_k, \sum_{l=0}^j D_l \right) = \max(i, j)\sigma^2$$

Wir bemerken, dass die Autokovarianz des Random Walks nicht nur vom Unterschied der Beobachtungen, sondern auch von den Zeitpunkten i und j abhängt.

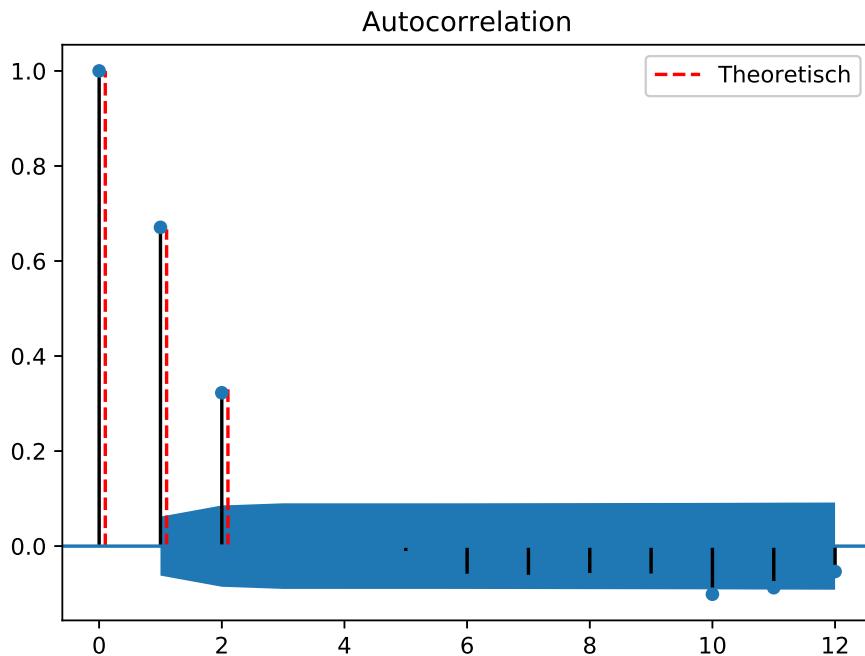


Abbildung 8.31.: Korrelogramm für den Moving Average Prozess mit den theoretischen und geschätzten Autokorrelationskoeffizienten.

Insbesondere ist die Varianz des Prozesses zur Zeit i

$$\text{Var}(X_i) = i\sigma^2$$

und nimmt somit mit der Zeit zu.

Die Autokorrelationsfunktion des Random Walk können wir nun einfach berechnen:

$$\rho(i, j) = \frac{\gamma(i, j)}{\sqrt{\gamma(i, i)\gamma(j, j)}} = \frac{\max(i, j)}{\sqrt{i \cdot j}}$$

□

Oft möchte man die lineare Abhängigkeit von zwei verschiedenen diskreten Prozessen X_i und Y_j messen. Das kann wertvoll sein, wenn wir beispielsweise die Vorhersage von X_i durch die Werte von Y_j messen wollen. Dies wird durch die folgenden Größen erreicht.

Kreuzkovarianz und Kreuzkorrelation

Seien $\{X_1, X_2, \dots\}$ und $\{Y_1, Y_2, \dots\}$ diskrete stochastische Prozesse.

1. Die Kreuzkovarianzfolge zwischen den beiden Prozessen ist definiert durch

$$\gamma_{XY}(i, j) = \text{Cov}(X_i, Y_j) = E[(X_i - \mu_X(i))(Y_j - \mu_Y(j))]$$

2. Die Kreuzkorrelationsfolge ist definiert durch

$$\rho_{XY}(i, j) = \frac{\gamma_{XY}(i, j)}{\sqrt{\gamma_X(i, i)\gamma_Y(j, j)}}$$

Die Konzepte der Mittelwert- und Kovarianz-/Korrelationsfolgen, die wir hier eingeführt haben, gelten komplett allgemein. Wir haben aber schon bemerkt, dass einige Prozesse eine gewisse Art von Regularität zeigen, die durch die Mittelwerte dieser Größen ausgedrückt werden können.

8.6. Stationarität

Wie wir gesehen haben, können wir Zeitreihen als *einzelne* Realisierung der multivariaten Zufallsvariable $\{X_1, X_2, \dots\}$ auffassen, welche wir in diesem Fall als stochastischen Prozess bezeichnet haben. Wir wissen allerdings schon aus der elementaren Statistik, dass wir keine ernsthafte statistische Analyse auf bloss einer einzigen Beobachtung aufbauen können. Deshalb brauchen wir ein Konzept der Regularität, welches uns erlaubt, aus einer einzelnen Zeitreihe Rückschlüsse auf den darunterliegenden Prozess zu machen. Diese Regularität ist die Stationarität.

Strikte Stationarität

Ein diskreter stochastischer Prozess heisst *strikt stationär*, falls für jede endliche Ansammlung $\{X_{i_1}, \dots, X_{i_n}\}$ und jede Verschiebung $h \in \mathbb{Z}$ die verschobene Ansammlung

$$\{X_{i_1+h}, \dots, X_{i_n+h}\}$$

der gleichen Verteilung folgt. Oder anders ausgedrückt:

$$P(X_{i_1} \leq c_1, \dots, X_{i_n} \leq c_n) = P(X_{i_1+h} \leq c_1, \dots, X_{i_n+h} \leq c_n)$$

für alle c_1, \dots, c_n .

Die Definition der strikten Stationarität läuft darauf hinaus, dass jede Auswahl von Beobachtungen der Zeitreihe eine typische Realisierung der Zufallsvariablen des Prozesses ist, wobei das Muster für jeden Zeitpunkt dasselbe ist. Grob gesagt bedeutet dies, dass sich der stochastische Charakter des Prozesse über die Zeit nicht ändert.

Die Annahme der strikten Stationarität ist oft zu stark für viele Anwendungen und schwer abzuschätzen aufgrund eines einzigen Datensatzes. Die Definition der strikten Stationarität impliziert insbesondere für den Spezialfall $n = 1$, dass

$$P(X_i \leq c) = P(X_j \leq c)$$

für alle i, j und für alle c . Wir können auch sagen, dass die Randverteilungen $F_i = F$ des Prozesses zusammenfallen und somit zur Folge haben, dass

$$\mu_X(i) = \mu_X(j)$$

In anderen Worten ist die *Mittelwertsfolge konstant*.

Wir betrachten nun weiter den Fall $n = 2$. Dann impliziert die Definition der strikten Stationarität, dass

$$P(X_i \leq c_1, X_j \leq c_2) = P(X_{i+h} \leq c_1, X_{j+h} \leq c_2)$$

Die gemeinsame Verteilung für jedes Paar von Zeitpunkten in diesem Prozess hängt nur von der Zeitdifferenz ab und somit gilt für die Autokovarianz

$$\gamma(i, j) = \gamma(i + h, j + h)$$

Diese beiden Schlussfolgerungen aus der strikten Stationarität genügen für die meisten Anwendungen, damit man ein vernünftigeres statistisches Modell einführen kann. Dazu wählen wir eine schwächere Form der Stationarität.

Schwache Stationarität

Ein stochastischer Prozess X_i heisst *schwach stationär* falls

1. die Mittelwertsfolgen $\mu_X(i)$ konstant sind und nicht vom Zeitindex i abhängen und
2. die Autokovarianzfolgen $\gamma_X(i, j)$ hängen von i und j nur durch die Differenz $|i - j|$ ab.

Aus der Diskussion oben ist klar, dass jede strikt stationäre Zeitreihe auch schwach stationär ist. Die Umkehrung ist allgemein nicht wahr. Man kann aber zeigen, dass für Gauss'sche Prozesse, also wenn jede endliche Auswahl der Zufallsvariablen im Prozess eine gemeinsame Normalverteilung hat, dass dann die beiden Begriffe der Stationarität äquivalent sind.

Da die Autokovarianz/-korrelation für (schwache) Stationarität nur vom *Zeitunterschied (lag)* $h = i - j$ abhängt, können wir diese Folgen als Funktionen von h selbst betrachten:

$$\begin{aligned}\gamma(h) &= \gamma(i, i + h) \\ \rho(h) &= \rho(i, i + h)\end{aligned}$$

Offensichtlich gilt

$$\gamma(h) = \gamma(-h)$$

so dass wir nur Werte $h = 0, 1, \dots$ betrachten müssen.

Beispiel 8.6.1

Wir betrachten den moving average Prozesses aus Beispiel 8.4.3. Es ist klar, dass die Mittelwertsfunktion

$$\mu(i) = \mu = 0$$

konstant ist und aus den Berechnungen in Beispiel 8.5.7 sehen wir, dass die Autokovarianz nur vom Zeitunterschied abhängt:

$$\gamma(h) = \begin{cases} \frac{3\sigma^2}{9} & \text{falls } h = 0 \\ \frac{2\sigma^2}{9} & \text{falls } |h| = 1 \\ \frac{\sigma^2}{9} & \text{falls } |h| = 2 \\ 0 & \text{else.} \end{cases}$$

Somit ist der Prozess des moving average schwach stationär.

□

8.6.1. Auf Stationarität testen

Sind wir in der Praxis mit Zeitreihen konfrontiert, so liegt uns diese in der Regel bloss als einzige Beobachtung des zugrundeliegenden diskreten stochastischen Prozesses vor. Stationarität stellt allerdings eine Eigenschaft des letzteren dar, und wir müssen *testen* oder zumindest *abschätzen*, ob der zugrundeliegende Prozess stationär ist oder nicht.

Ein typisches Beispiel ist eine Zeitreihenzerlegung, wie wir sie in Abschnitt 7.4.3 untersucht haben. Zerlegen wir eine Zeitreihe in eine Trend- und Saisonalitätskomponente, dann folgt durch Subtraktion dieser beiden Komponenten von der Zeitreihe die sogenannte *Restreihe*. Eine typische Annahme für die Modellierung eines solchen

stochastischen Prozesses besteht in der (schwachen) Stationarität der Restreihe. Wie können wir dies aber aus den Daten schliessen?

Der erste und einfachste Typ von Test, den wir zum Prüfen von Stationarität anwenden können, besteht darin, die Daten graphisch darzustellen. Wir halten dann Ausschau nach Hinweisen für einen Trend in der Mittelwertsfolge, Varianz- und Autokorrelationsfolge und der Saisonalität. Ist dies der Fall, so können wir nicht mehr von Stationarität ausgehen, und wir versuchen mit Hilfe von Datentransformation, Stationarität herzustellen.

Beispiel 8.6.2

Wenn wir Abbildung 7.1 der monatlichen Flugpassagierzahlen betrachten, so ist ein klarer Trend und ein saisonales Muster erkennbar. Es ist *nicht* vernünftig zu behaupten, dass diese Zeitreihe eine Realisierung eines stationären stochastischen Prozesses ist.

□

Eine weitere Möglichkeit besteht darin, die Mittelwerts- und Autokovarianzfolgen für verschiedene Fensterweiten zu berechnen und das Verhalten zu vergleichen. Wenn es dramatische "Anderungen gibt, dann kann die Hypothese der Stationarität abgelehnt werden.

Wir können die Autokorrelationsfunktion für beliebige Zeitreihen berechnen, insbesondere auch für *deterministische Signale*. Einige Resultate für deterministische Signale sind hilfreich, um Muster in Autokorrelationsfunktionskurven von Zeitreihen zu erklären, die wir nicht als Realisierungen von stationären Prozessen betrachten können:

- Wir konstruieren eine Zeitreihe, die nur aus einem Trend besteht, beispielsweise die natürlichen Zahlen von 1 bis 1000. Dann nimmt die Autokorrelationsfunktionskurve langsam und fast linear ausgehend von 1 ab.
- Nehmen wir eine grosse Anzahl Zyklen von diskreten sinusförmigen Wellen mit beliebiger Amplitude und Phase, dann ist die Autokorrelationsfunktionskurve eine diskrete Cosinusfunktion mit der gleichen Periode.
- Wir konstruieren eine Zeitreihe, die aus einer zufälligen Folge von p Zahlen besteht, die sehr oft wiederholt werden. Das Korrelogramm hat dann dominierende Spitzen von fast 1 mit lag p .

Üblicherweise zeigt sich im Falle einer langsam abfallenden Autokorrelationsfunktion, dass es einen Trend in der Zeitreihe gibt. Die entsprechenden Autokorrelationsfunktionskoeffizienten sind in der Regel gross und positiv, da benachbarte Werte der Zeitreihe sehr ähnlich sind.

Wir werden in den Übungen Autokorrelationsfunktionskurven von deterministischen Signalen studieren.

Beispiel 8.6.3

Für den Datensatz **AirPassengers** ist das Korrelogramm in Abbildung 8.32 dargestellt.

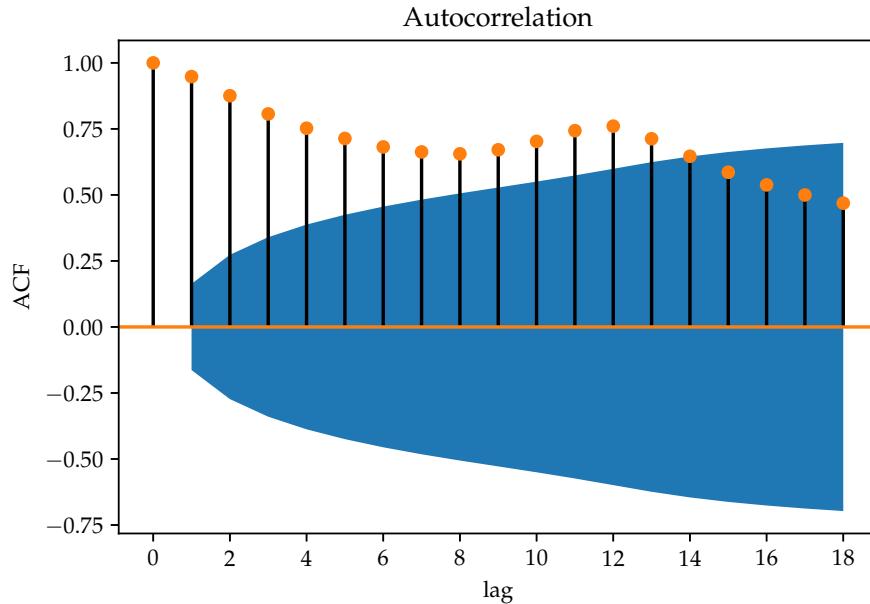


Abbildung 8.32.: Korrelogramm für den Datensatz **AirPassengers** für die ersten 18 Monate.

Gibt es saisonale Variationen, so werden saisonale Spitzen dem Muster in der Autokorrelationskurve überlagert. Der jährliche Zyklus erscheint im **AirPassengers**-Korrelogramm als Zyklus derselben Periode und überlagert die graduelle Abnahme der Autokorrelationsfunktion. Somit erhalten wir nach einem Jahr bei lag 12 wieder ein Maximum. Dies widerspiegelt einen positiven linearen Zusammenhang zwischen Paaren von Variablen $\{x_i, x_{i+12}\}$, die durch ein Jahr getrennt sind. Umgekehrt tendieren Werte, die durch 6 Monate getrennt sind, zu einem negativen Zusammenhang, da der saisonale Effekt annähernd sinusförmig ist. Beispielweise tendieren höhere Werte in den Sommermonaten von tieferen Werten in den Wintermonaten gefolgt zu werden. Eine Senke erscheint folglich bei etwa lag 6.

□

Obwohl dies ein typisches Beispiel für saisonale Variationen ist, die durch sinusförmige Kurven angenähert werden können, mögen andere Zeitreihen andere Muster haben. Bespielsweise tendieren Zeitreihen für Verkaufszahlen in Modegeschäften

aufgrund des grossen Umsatzes vor Weihnachten dazu, eine einzige Spitze im Korrelogramm zu haben.

Wollen wir Trends und saisonale Muster in Zeitreihen erkennen, so müssen wir uns nicht notwendigerweise auf das Korrelogramm verlassen, um diese zu identifizieren. Der hauptsächliche Nutzen des Korrelogramms liegt darin, Autokorrelation zu entdecken, *nachdem* Trend und Saisonalität entfernt wurden.

Beispiel 8.6.4

Im Code unten wird für die Zeitreihe **AirPassengers** der Trend und der saisonale Effekt entfernt.

```
from statsmodels.tsa.seasonal import seasonal_decompose
remainder = seasonal_decompose(AirP["Passenger"], model="multiplicative").resid[6:138]
```

Um die Zufallskomponente und das Korrelogramm zu zeichnen, müssen wir uns in Erinnerung rufen, dass wir als Konsequenz des centred moving average mit Periode 12 für die Schätzung des Trendes, die ersten und letzten 6 Einträge nicht berechnen können. Somit fehlen diese auch für die Zufallskomponente und werden in **Python** mit **NA** gespeichert.

Die Zufallskomponente ist in Abbildung 8.33 dargestellt.

```
remainder.plot()
```

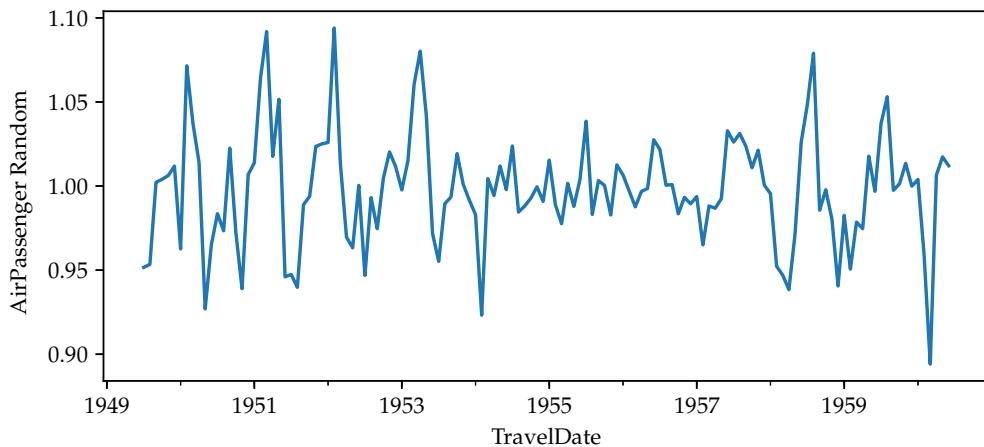


Abbildung 8.33.: Restterm der Zeitreihe **AirPassengers**, nach Entfernung von Trend und Saisonalität.

Die Zufallskomponente ist in Abbildung 8.34 graphisch dargestellt.

```
plot_acf(remainder, lags = 21)
```

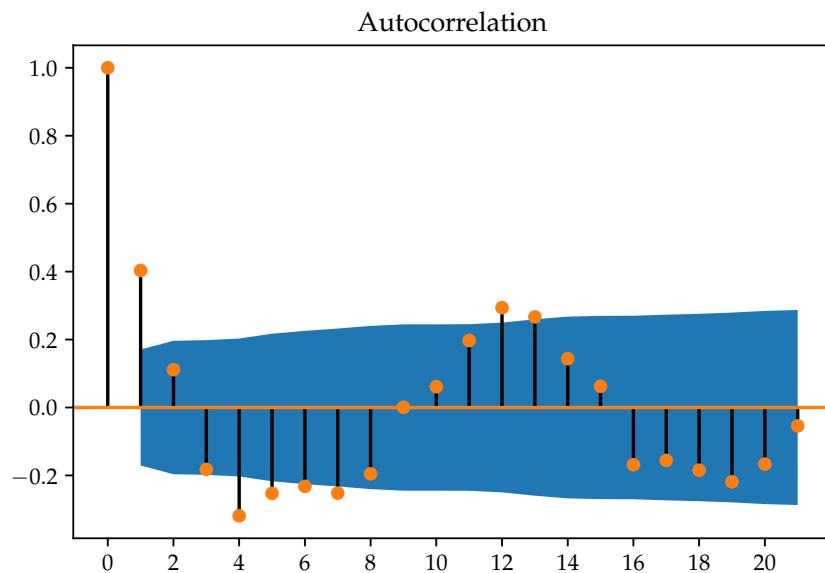


Abbildung 8.34.: Korrelogramm der Zufallskomponente für `AirPassengers`.

Das Korrelogramm suggeriert entweder eine gedämpfte Cosinusschwingung, die charakteristisch ist für autoregressive Modelle 2. Ordnung (siehe später) oder dass die saisonale Anpassung nicht vollständig effektiv war. Letztere Erklärung ist unwahrscheinlich, weil die Zerlegung 12 unabhängige monatliche Indizes schätzte. Wenn wir dies noch genauer untersuchen, so stellen wir fest, dass die Standardabweichung der ursprünglichen Reihe vom Juli bis Juni etwa 109 beträgt:

```
AirP["Passengers"][6:138].std()
## 109.41868429698121
```

Entfernen wir von der ursprünglichen Reihe noch den Trend, so ergibt sich für die Standardabweichung der Wert 41:

```
trend = seasonal_decompose(AirP["Passengers"], model="multiplicative").trend[6:138]
(AirP["Passengers"][6:138]-trend).std()
## 41.1149061835257
```

Die Standardabweichung nach Anpassung der saisonalen Komponente ist nur noch 0.03

```
res = seasonal_decompose(AirP["Passengers"], model="multiplicative").resid[6:138]

res.std()

## 0.033388395802509976
```

Die Reduktion der Standardabweichung zeigt folglich, dass die saisonale Anpassung sehr effektiv war.

□

8.7. Brownsche Bewegung³

Der Begriff *Brownsche Bewegung* bezieht sich auf die 1827 vom Biologen Robert Brown durchgeführten Arbeiten, in welchen dieser die zufällige Bewegung von in Wasser schwimmenden Pollen beobachtete.

Einstein lieferte 1905 eine Erklärung dafür: die Zitterbewegung der Pollen wird durch fortwährende Stöße mit sich zufällig bewegenden Wassermolekülen verursacht (siehe Abbildung 8.35). Dies war in jener Zeit tatsächlich noch ein gewichtiges Argument für die Existenz von Atomen und Molekülen, die im 19. Jahrhundert noch heftig umstritten gewesen ist.

Und gleichzeitig passte Einsteins Beschreibung zur molekularen Theorie der Wärme. Je wärmer beispielsweise Wasser ist, um so grösser ist die mittlere Geschwindigkeit, mit der die Wassermoleküle ungeordnet umherflitzen und damit Stöße verursachen.

Die mathematische Theorie der Brownschen Bewegung wurde von Louis Bachelier im Jahre 1900 in seiner Doktorarbeit „Theorie de la speculation“ entwickelt, die Random Walks mit der Entwicklung von Börsenkursen in Verbindung brachte.

Der Durchbruch kam jedoch, als Albert Einstein 1905 ohne Kenntnis von Bacheliers Arbeiten und unabhängig von ihm Marian Smoluchowski (1906), den stochastischen Prozess der Brownschen Bewegung (auch Wiener-Prozess genannt) in seiner heutigen Gestalt definierte.

Einen Beweis für die wahrscheinlichkeitstheoretische Existenz des Prozesses blieb Einstein allerdings schuldig. Dieser gelang erst 1923 dem US-amerikanischen Mathematiker Norbert Wiener.

³Dieses Kapitel ist hochrelevant, aber nicht Teil des Prüfungsstoffes.

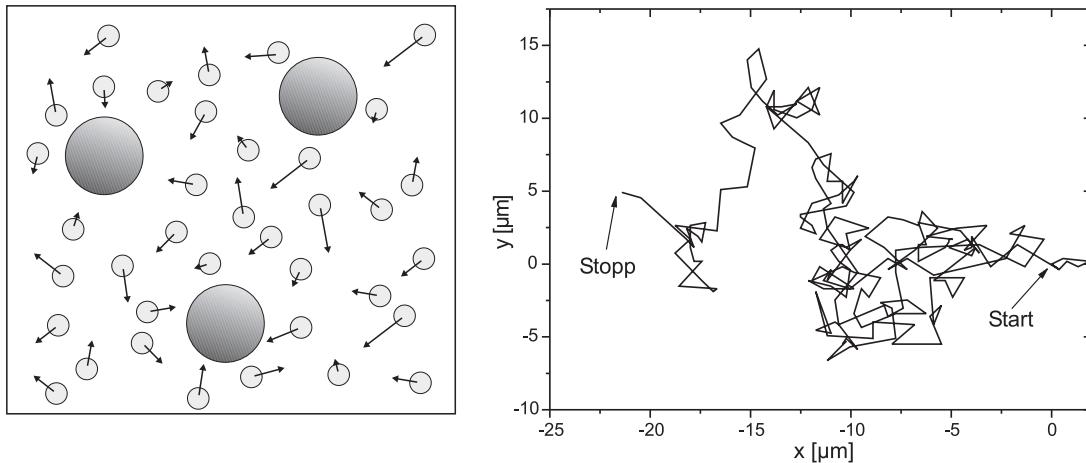


Abbildung 8.35.: Links: Modell der Brownschen Bewegung. Die Moleküle des umgebenden Mediums stoßen aufgrund ihrer thermischen Energie mit den suspendierten Partikeln zusammen, wodurch sich diese auf völlig unregelmässigen Bahnen bewegen. Rechts: Gemessene Bahn eines einzelnen Partikels.

Brownsche Bewegung ergibt sich aus einem *Random Walk*, wobei die Zeitvariable kontinuierlich gemacht wird und die Schrittweiten als normalverteilte Zufallsvariablen aufgefasst werden.

Anstelle eines Betrunkenen betrachten wir dazu nun Partikel in Wasser. Geben wir zum Beispiel einen Tintentropf in Wasser, so diffundieren die Tintenmoleküle im Wasser und verteilen sich gleichmäßig.

Nehmen wir an, in der Flüssigkeit befinden sich n Tintenmoleküle zum Zeitpunkt $t = 0$ an der Stelle x_0 , dann kann die zeitabhängige Teilchendichte als

$$n(x, t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}} \quad (8.5)$$

beschrieben werden, wobei D die von der Temperatur T des Wassers abhängige Diffusionskonstante bezeichnet (siehe Abbildung 8.36). Aufgrund der Diffusion der Tintenmoleküle wird die Teilchendichtekurve mit grösser werdendem t immer breiter.

Wir können den Zusammenhang des makroskopisch beschriebenen Phänomens der *Diffusion* mit dem mikroskopischen Phänomen der *Brownschen Bewegung* eines Partikels verstehen, indem wir zu einer Kontinuumsbeschreibung des Random Walks übergehen.

Wir fassen also die Bewegung des Tintenmoleküls im Wasser als einen Random Walk auf (in Analogie zum Barbesucher), lassen nun aber die Schrittänge Δx sowie den zeitlichen Abstand Δt zwischen zwei Schritten immer kleiner werden.

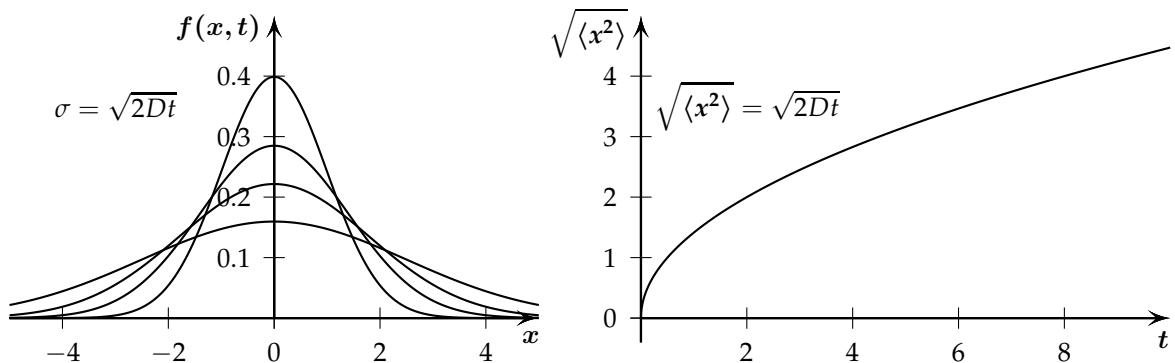


Abbildung 8.36.: Links: Zeitabhängige Wahrscheinlichkeitsdichte mit dem Mittelwert $\langle x \rangle = x_0 = 0$ und der Varianz $\sigma^2 = \langle x^2 \rangle - \langle x \rangle^2 = 2Dt$. Da die Varianz zeitabhängig ist, wird die Verteilung mit zunehmender Zeit immer breiter. Rechts: Mittleres Verschiebungsquadrat $\langle x^2 \rangle$ als Funktion der Zeit.

Wir führen nun den Grenzübergang $\Delta x \rightarrow 0, \Delta t \rightarrow 0$ bei konstantem D und konstanter v durch. Dies bedeutet, dass wir die beiden „makroskopischen“ Charakteristika des Zufallspfades, die mittlere Driftgeschwindigkeit v und die Diffusionskonstante D vorgeben, aber Raum und Zeit kontinuierlich machen. Bei diesem Prozess schrumpft Δx mit der Wurzel von Δt , denn bei konstantem D gilt $(\Delta x)^2 \propto \Delta t$ oder $\Delta x \propto \sqrt{\Delta t}$. Andererseits muss bei konstantem v gelten, dass die Parameter p und q sich so ändern, dass $p - q \propto \Delta x$ ist. Für $\Delta x \rightarrow 0$ nähern sich also beide Parameter p und q dem Wert $1/2$ an, wobei v konstant bleibt. Für $\Delta x \rightarrow 0$ wird aber auch die Punktswahrscheinlichkeit $P(X = m\Delta x; N\Delta t)$ null, was wir bei einer kontinuierlichen Wahrscheinlichkeitsverteilung natürlich erwarten. Nun entspricht der Ausdruck $P(X = m\Delta x; N\Delta t)$ für die diskrete Zufallsvariable X der Wahrscheinlichkeit

$$\begin{aligned} P(x < X \leq x + \Delta x; t) &= F(x + \Delta x, t) - F(x, t) \\ &= \Delta F(x, t) \\ &= \frac{2\Delta x}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}}, \end{aligned}$$

also der Differenz der kumulativen Wahrscheinlichkeitsfunktion. Wir erhalten dann folgende Wahrscheinlichkeitsdichte für die Position x des Brownschen Partikels zur Zeit t

$$\frac{\Delta F(x, t)}{2\Delta x} \xrightarrow{\Delta x \rightarrow 0} f(x; t) = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}}, \quad (8.6)$$

Die Position des Brownschen Partikels zur Zeit $t = 0$ bezeichnen wir mit x_0 . In unserer Herleitung aus dem Random Walk haben wir stets $x_0 = 0$ angenommen. Falls dies nicht der Fall ist, ersetzen wir x einfach durch $x - x_0$. Der Parameter D wird **Diffusionskonstante** und v wird **mittlere Driftgeschwindigkeit** genannt.

Nehmen wir einfacheitshalber an, dass die Driftgeschwindigkeit verschwindet (falls $p = q$, dann ist $v = 0$) und die Ausgangsposition x_0 nicht null sein muss, dann ergibt

sich die Teilchendichtefunktion (D.1) nun, indem wir die n Brownschen Teilchen mit $f(x; t)$ multiplizieren:

$$n(x, t) = n \cdot f(x; t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}} \quad (8.7)$$

Diffusionsgleichung

Die Teilchendichte

$$n(x, t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}}$$

löst die **Diffusionsgleichung**

$$\frac{\partial n(x, t)}{\partial t} = D \frac{\partial^2 n(x, t)}{\partial x^2}, \quad (8.8)$$

mit der Randbedingung, dass sich alle Teilchen zum Zeitpunkt $t = 0$ bei $x = 0$ befinden. Der Random Walk in der Kontinuumsbeschreibung wird **Brownsche Bewegung** genannt und kann also als **Diffusionsprozess** aufgefasst werden.

Die beiden Terme auf der rechten Seite von Gleichung (D.5) haben eine einfache anschauliche Bedeutung: Der erste Term (mit v) verschiebt die Wahrscheinlichkeitsdichte $f(x, t)$ mit Geschwindigkeit v nach rechts (wenn v positiv ist; sonst wird die Verteilung nach links geschoben). Man nennt diesen Term den **Driftterm**. Für $p = q$ ist $v = 0$, und der Driftterm fällt weg. Der Term auf der rechten Seite von Gleichung (D.5) macht die Verteilung breiter; er wird deshalb **Diffusionsterm** genannt. Physikalisch stellen wir uns einen Tintentropfen vor, der in heißes Wasser gegeben wird: durch Diffusion verteilt sich die Tinte im Wasser aufgrund der Stöße zwischen den sich zufällig bewegenden Wassermolekülen und Tintenmolekülen.

Wir erhalten dann folgende Wahrscheinlichkeitsdichte für die Position x des Brownschen Partikels zur Zeit t

$$f(x; t) = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}} \quad (8.9)$$

Die Position des Brownschen Partikels zur Zeit $t = 0$ bezeichnen wir mit x_0 . In unserer Herleitung aus dem Random Walk haben wir stets $x_0 = 0$ angenommen.

Falls dies nicht der Fall ist, ersetzen wir x einfach durch $x - x_0$. Der Parameter D wird **Diffusionskoeffizient** und v wird **mittlere Driftgeschwindigkeit** genannt.

Gleichung (8.9) ist natürlich eine zeitabhängige Normalverteilung mit Mittelwert vt und Varianz $2Dt$ (siehe Abbildung 8.36). Die mittlere Verschiebung $E[X]$ des Brown-

schen Partikels ergibt

$$\langle x \rangle \equiv E[X] = \int_{-\infty}^{\infty} x f(x, t) dt = vt$$

Im Folgenden nehmen wir einfacheitshalber an, dass die Driftgeschwindigkeit null ist. Die mittlere Verschiebung $\langle x \rangle$ ist in diesem Fall also null (oder x_0).

Es macht daher mehr Sinn, das mittlere Verschiebungsquadrat $E[X^2]$ als Mass für den mittleren zurückgelegten Abstand zu betrachten:

$$\langle x^2 \rangle \equiv E[X^2] = \int_{-\infty}^{\infty} x^2 f(x, t) dx = 2Dt$$

was der Varianz σ^2 der zeitabhängigen Wahrscheinlichkeitsdichte $f(x, t)$ entspricht. Damit können wir das wichtige Ergebnis unserer Untersuchung wie folgt formulieren:

Einstein-Smoluchowski-Gleichung

Der *quadratisch gemittelte Abstand* ($\equiv \sqrt{\langle x^2 \rangle}$) eines Partikels vom Ursprungsort, nimmt mit der Quadratwurzel der Zeit t zu:

$$\sqrt{\langle x^2 \rangle} = \sqrt{2Dt}$$

Der quadratisch gemittelte Abstand eines Teilchens als Funktion der Zeit t ist in Abbildung 8.36 dargestellt.

Diese Gleichung heisst *Einstein-Smoluchowski-Gleichung*.

Bisher haben wir die Brownsche Bewegung nur in einer Dimension untersucht. Unser Ergebnis lässt sich aber sehr einfach auf mehrere Dimensionen übertragen.

Findet die Brownsche Bewegung in zwei Dimensionen statt, so gilt für das mittlere Verschiebungsquadrat $\langle r^2 \rangle$:

$$\langle r^2 \rangle = \langle x^2 \rangle + \langle y^2 \rangle$$

Da die Brownsche Bewegung isotrop (alle Richtungen sind gleichberechtigt) ist, liefert jeder Summand den Beitrag $2Dt$ und somit ist

$$\langle r^2 \rangle = 4Dt$$

bzw. im Dreidimensionalen:

$$\langle r^2 \rangle = 6Dt$$

Der Parameter D wird als *Diffusionskoeffizient* bezeichnet und ist ein Mass für die Beweglichkeit des Partikels im umgebenden Medium.

Kapitel 8. Mathematische Modelle für Zeitreihen

Nach Einstein und Stokes ist der Diffusionskoeffizient gegeben durch

$$D = \frac{kT}{6\pi\eta a}$$

wobei k die Boltzmann-Konstante, T die Temperatur der Flüssigkeit, η die Viskositätskonstante und a den Radius des kugelförmigen Brownschen Partikels bezeichnet.

Die Einstein-Smoluchowski-Gleichung für ein Brownsches Teilchen mit Radius a in einer Flüssigkeit mit Temperatur T und Viskosität η in drei Dimensionen lautet also

$$\sqrt{\langle r^2 \rangle} = \sqrt{6Dt} = \sqrt{\frac{kTt}{\pi\eta a}}$$

Nehmen wir an, in der Flüssigkeit befinden sich n Brownsche Partikel, dann kann die zeitabhängige Teilchendichte als

$$n(x, t) = n \cdot f(x; t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}} \quad (8.10)$$

geschrieben werden, wobei $f(x; t)$ die in Gleichung (8.9) gegebene Wahrscheinlichkeitsdichte ist. Dies ist also die Teilchendichtefunktion, die die Diffusionsgleichung löst⁴.

⁴Eine etwas ausführlichere Diskussion befindet sich in Kapitel D.1 des Anhangs.

Anhang A.

R-Code

(zu Python)

```
methodeA <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03,
             80.04, 79.97, 80.05, 80.03, 80.02, 80, 80.02)

sort(methodeA)

## [1] 79.97 79.98 80.00 80.02 80.02 80.02 80.03 80.03
## [9] 80.03 80.04 80.04 80.04 80.05
```

Beispiel 2.1.4

(zu Python)

```
methodeA <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03,
             80.04, 79.97, 80.05, 80.03, 80.02, 80, 80.02)

mean(methodeA)

## [1] 80.02077
```

Beispiel 2.1.6

(zu Python)

```
var(methodeA)

## [1] 0.000574359

sd(methodeA)

## [1] 0.02396579
```

Beispiel 2.1.7

(zu **Python**)

```
median(methodeA)

## [1] 80.03

methodeB <- c(80.02, 79.94, 79.98, 79.97, 79.97, 80.03,
            79.95, 79.97)

median(methodeB)

## [1] 79.97
```

Beispiel 2.1.10

(zu **Python**)

```
# Syntax fuer das untere Quartil: p=0.25

quantile(methodeA, probs = 0.25)

##      25%
## 80.02

quantile(methodeB, probs = 0.25)

##      25%
## 79.965

# Syntax fuer das obere Quartil: p=0.75

quantile(methodeA, probs = 0.75)

##      75%
## 80.04
```

Beispiel 2.1.11

(zu Python)

```
quantile(methodeA, probs = 0.25, type = 1)

##      25%
## 80.02

quantile(methodeA, probs = 0.25, type = 2)

##      25%
## 80.02

quantile(methodeA, probs = 0.25, type = 3)

## 25%
## 80

quantile(methodeA, probs = 0.25, type = 4)

##      25%
## 80.005

quantile(methodeA, probs = 0.25, type = 5)

##      25%
## 80.015

quantile(methodeA, probs = 0.25, type = 6)

##      25%
## 80.01

quantile(methodeA, probs = 0.25, type = 7)

##      25%
## 80.02

quantile(methodeA, probs = 0.25, type = 8)

##      25%
## 80.01333

quantile(methodeA, probs = 0.25, type = 9)

##      25%
## 80.01375
```

Anhang A. R-Code

(zu Python)

```
quantile(methodeB, probs = 0.25, type = 1)

##    25%
## 79.95

quantile(methodeB, probs = 0.25, type = 2)

##    25%
## 79.96

quantile(methodeB, probs = 0.25, type = 3)

##    25%
## 79.95

quantile(methodeB, probs = 0.25, type = 4)

##    25%
## 79.95

quantile(methodeB, probs = 0.25, type = 5)

##    25%
## 79.96

quantile(methodeB, probs = 0.25, type = 6)

##    25%
## 79.955

quantile(methodeB, probs = 0.25, type = 7)

##    25%
## 79.965

quantile(methodeB, probs = 0.25, type = 8)

##    25%
## 79.95833

quantile(methodeB, probs = 0.25, type = 9)

##    25%
## 79.95875
```

Beispiel 2.1.12

(zu Python)

```
IQR(methodeA, type = 2)

## [1] 0.02
```

Beispiel 2.1.14

(zu Python)

```
quantile(methodeA, probs = 0.1)

##      10%
## 79.984

quantile(methodeA, probs = 0.7)

##      70%
## 80.034
```

Beispiel 2.1.15

(zu Python)

```
noten <- c(4.2, 2.3, 5.6, 4.5, 4.8, 3.9, 5.9, 2.4, 5.9,
       6, 4, 3.7, 5, 5.2, 4.5, 3.6, 5, 6, 2.8, 3.3, 5.5, 4.2,
       4.9, 5.1)

quantile(noten, seq(from = 0.2, to = 1, by = 0.2))

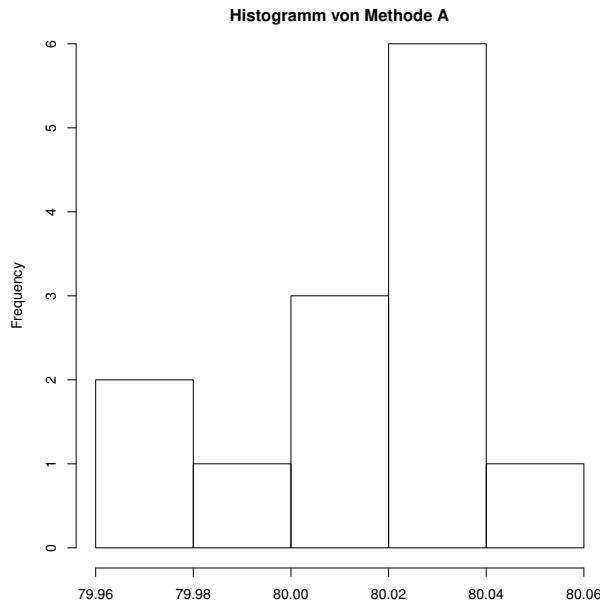
## 20% 40% 60% 80% 100%
## 3.66 4.26 4.98 5.54 6.00
```

Anhang A. R-Code

Beispiel 2.1.17

(zu **Python**)

```
hist(methodeA, main = "Histogramm von Methode A")
```

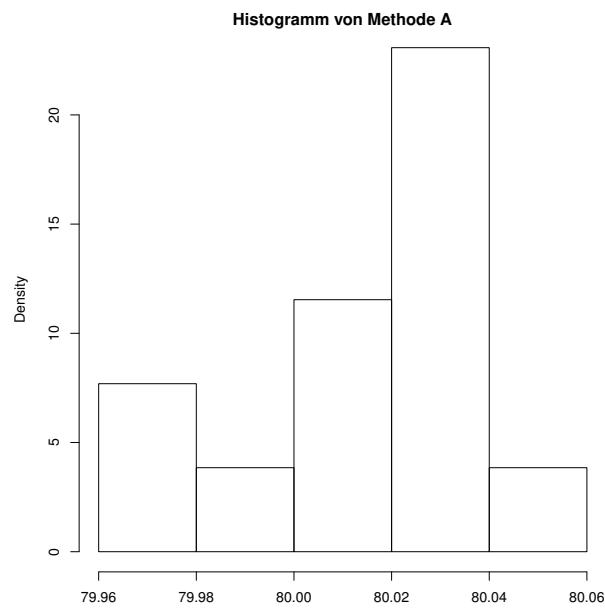


Beispiel 2.1.18

(zu **Python**)

```
hist(methodeA, freq = F, main = "Histogramm von Methode A")
```

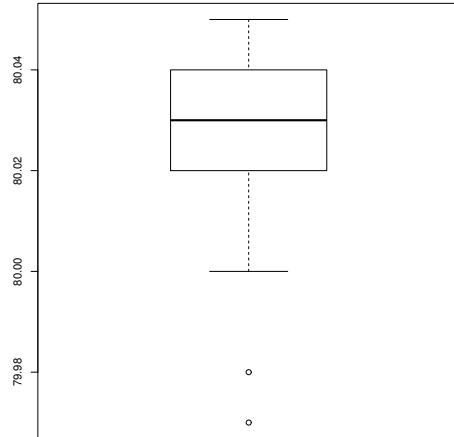
Anhang A. R-Code



Beispiel 2.1.19

(zu Python)

```
boxplot(methodeA)
```

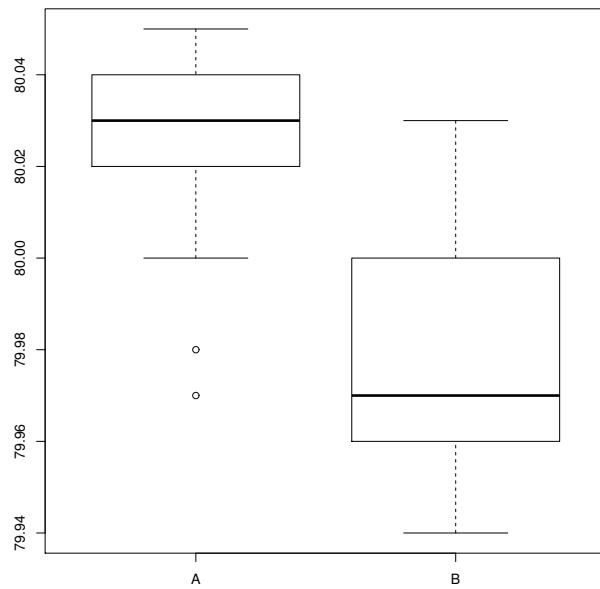


Beispiel 2.1.20

(zu Python)

Anhang A. R-Code

```
boxplot(methodeA, methodeB, xaxt = "n", xlab = "Methode")
axis(1, at = c(1, 2), labels = c("A", "B"))
```

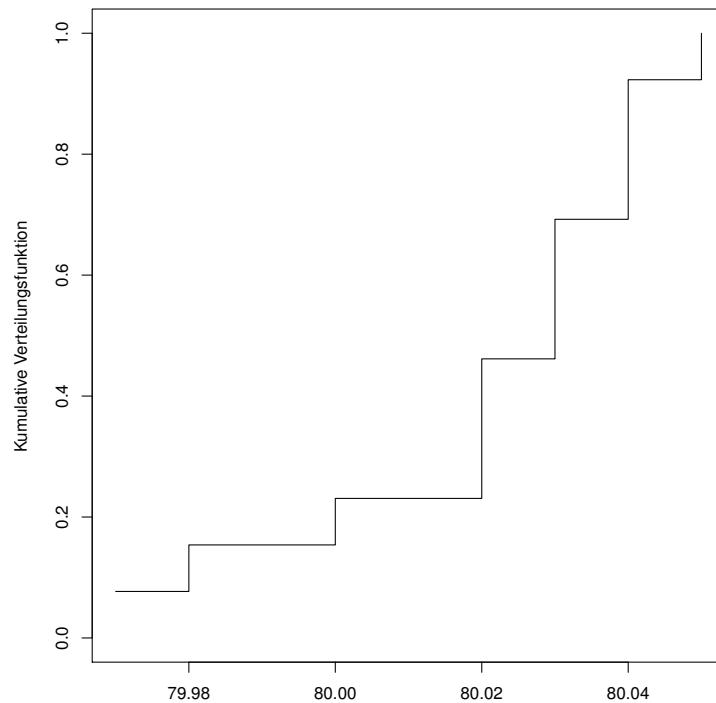


Beispiel 2.1.23

(zu [Python](#))

```
plot(sort(methodeA), (1:length(methodeA))/length(methodeA),
      type = "s", ylim = c(0, 1), ylab = "Kumulative Verteilungsfunktion",
      xlab = "Methode A")
```

Anhang A. R-Code



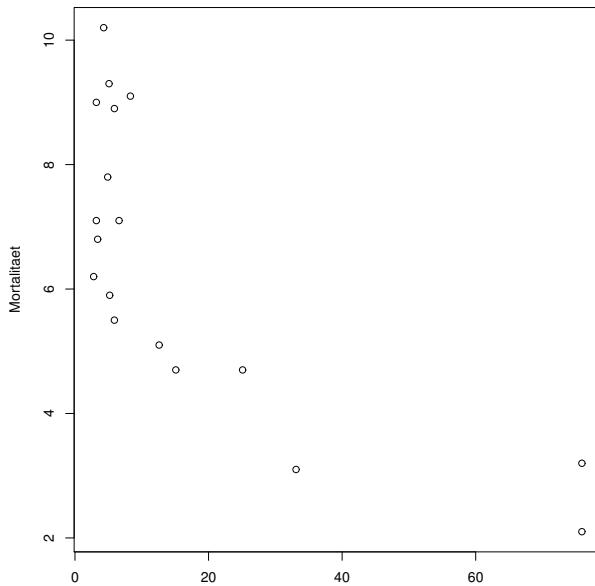
Beispiel 2.2.2

(zu [Python](#))

```
wein <- c(2.8, 3.2, 3.2, 3.4, 4.3, 4.9, 5.1, 5.2, 5.9, 5.9,
       6.6, 8.3, 12.6, 15.1, 25.1, 33.1, 75.9, 75.9)
mort <- c(6.2, 9, 7.1, 6.8, 10.2, 7.8, 9.3, 5.9, 8.9, 5.5,
        7.1, 9.1, 5.1, 4.7, 4.7, 3.1, 3.2, 2.1)

plot(wein, mort, xlab = "Weinkonsum (Liter pro Jahr und Person)",
     ylab = "Mortalitaet")
```

Anhang A. R-Code



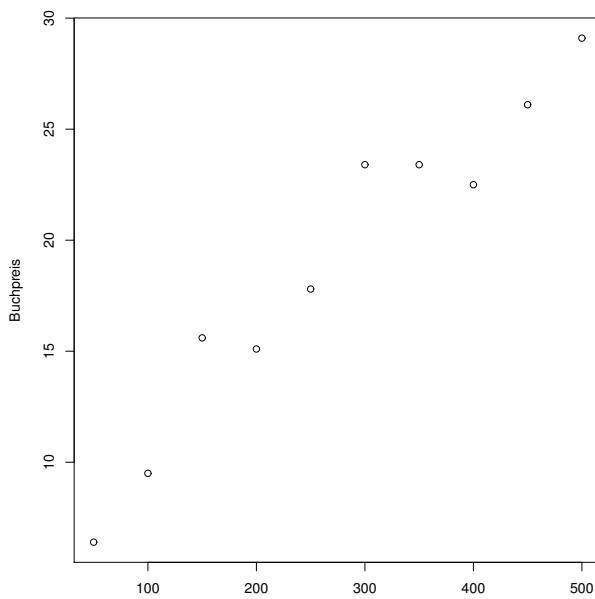
Beispiel 2.2.3

(zu Python)

```
seitenzahl <- c(seq(50, 500, 50))

buchpreis <- c(6.4, 9.5, 15.6, 15.1, 17.8, 23.4, 23.4, 22.5,
             26.1, 29.1)

plot(seitenzahl, buchpreis, xlab = "Seitenzahl", ylab = "Buchpreis")
```



Beispiel 2.2.4

(zu **Python**)

```
seitenzahl <- c(seq(50, 500, 50))

buchpreis <- c(6.4, 9.5, 15.6, 15.1, 17.8, 23.4, 23.4, 22.5,
             26.1, 29.1)

lm(buchpreis ~ seitenzahl)

##
## Call:
## lm(formula = buchpreis ~ seitenzahl)
##
## Coefficients:
## (Intercept)    seitenzahl
##       6.04000      0.04673
```

Beispiel 2.2.5

(zu **Python**)

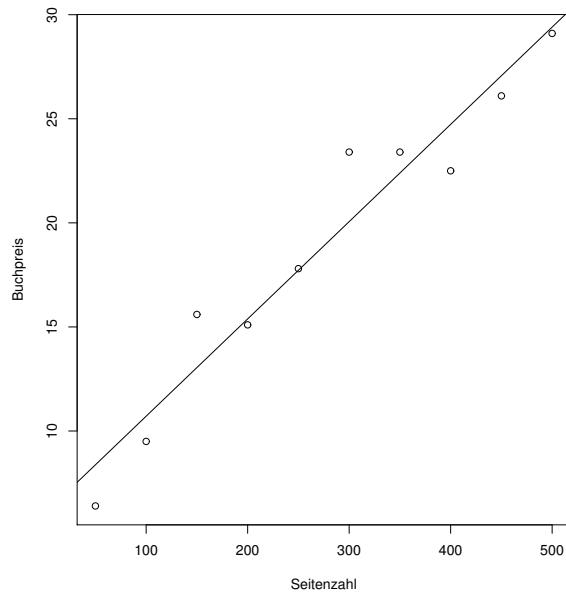
```
seite <- c(seq(50, 500, 50))

preis <- c(6.4, 9.5, 15.6, 15.1, 17.8, 23.4, 23.4, 22.5,
          26.1, 29.1)

plot(seite, preis, xlab = "Seitenzahl", ylab = "Buchpreis")

abline(lm(preis ~ seite))
```

Anhang A. R-Code



Beispiel 2.2.9

(zu Python)

```
cor(seitenzahl, buchpreis)  
## [1] 0.9681122
```

Beispiel 2.3.1

(zu Python)

```
library(tidyr)  
df <- data.frame(rownames = c("John", "Paul", NA, "Wale",  
    "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,  
    NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,  
    10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,  
    4), Value = c(55, 84, NA, 90, 63, 15, 46))  
df  
  
##   rownames Age Sex Goals Assists Value  
## 1      John  21   M     5       7    55  
## 2      Paul  23 <NA>    10       4    84  
## 3    <NA>   NA <NA>     NA       NA    NA
```

Anhang A. R-Code

```
## 4      Wale   19     M    19      9    90
## 5      Mary   25     F     5      7    63
## 6      Carli NA     F     0      6    15
## 7      Steve  15     M     7      4    46
```

(zu Python)

```
library(tidyr)
df <- data.frame(rownames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
drop_na(df)

##   rownames Age Sex Goals Assists Value
## 1   John   21   M     5       7    55
## 4   Wale   19   M    19      9    90
## 5   Mary   25   F     5       7    63
## 7   Steve  15   M     7       4    46
```

(zu Python)

```
df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
ind <- apply(df, 1, function(x) all(is.na(x)))
df_drop <- df[!ind, ]
df_drop

##   colnames Age Sex Goals Assists Value
## 1   John   21   M     5       7    55
## 2   Paul   23 <NA>   10      4    84
## 4   Wale   19   M    19      9    90
## 5   Mary   25   F     5       7    63
## 6   Carli NA   F     0       6    15
## 7   Steve  15   M     7       4    46
```

(zu Python)

Anhang A. R-Code

```
df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
df[, colSums(!is.na(df)) != nrow(df)]
```

	colnames	Age	Sex	Goals	Assists	Value
## 1	John	21	M	5	7	55
## 2	Paul	23	<NA>	10	4	84
## 3	<NA>	NA	<NA>	NA	NA	NA
## 4	Wale	19	M	19	9	90
## 5	Mary	25	F	5	7	63
## 6	Carli	NA	F	0	6	15
## 7	Steve	15	M	7	4	46

(zu Python)

```
df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
df_new <- cbind(df, New = rep(NA, nrow(df)))
df_new
```

	colnames	Age	Sex	Goals	Assists	Value	New
## 1	John	21	M	5	7	55	NA
## 2	Paul	23	<NA>	10	4	84	NA
## 3	<NA>	NA	<NA>	NA	NA	NA	NA
## 4	Wale	19	M	19	9	90	NA
## 5	Mary	25	F	5	7	63	NA
## 6	Carli	NA	F	0	6	15	NA
## 7	Steve	15	M	7	4	46	NA

(zu Python)

```
df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
df_new <- cbind(df, New = rep(NA, nrow(df)))
thresh <- 2
df_new[, !(nrow(df_new) - colSums(!is.na(df_new)) > thresh)]
```

Anhang A. R-Code

```
##   colnames Age  Sex Goals Assists Value
## 1      John  21    M     5      7    55
## 2      Paul  23 <NA>    10      4    84
## 3     <NA>   NA <NA>    NA     NA    NA
## 4      Wale  19    M    19      9    90
## 5      Mary  25    F     5      7    63
## 6     Carli  NA    F     0      6    15
## 7     Steve  15    M     7      4    46
```

Beispiel 2.3.2

(zu Python)

```
library(Hmisc)

## Loading required package: lattice
## Loading required package: Formula
## Loading required package: ggplot2
##
## Attaching package: 'Hmisc'
## The following objects are masked from 'package:base':
## 
##     format.pval, units

df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
                             "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
                             NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
                             10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
                             4), Value = c(55, 84, NA, 90, 63, 15, 46))
impute(df[, "Age"], mean)

##      1      2      3      4      5      6      7
## 21.0 23.0 20.6* 19.0 25.0 20.6* 15.0

# or simply
df$Age[is.na(df$Age)] <- mean(df$Age, na.rm = T)
df
```

Anhang A. R-Code

```
##   colnames Age  Sex Goals Assists Value
## 1     John 21.0    M      5       7     55
## 2     Paul 23.0 <NA>     10      4     84
## 3    <NA> 20.6 <NA>     NA      NA     NA
## 4     Wale 19.0    M     19       9     90
## 5     Mary 25.0    F      5       7     63
## 6    Carli 20.6    F      0       6     15
## 7    Steve 15.0    M      7       4     46
```

Beispiel 2.3.3

(zu Python)

```
library(DMwR)

## Loading required package: grid

df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
knnOutput <- knnImputation(df[, !names(df) %in% "medv"],
  k = 3)
knnOutput

##   colnames      Age Sex      Goals Assists      Value
## 1     John 21.00000  M  5.000000 7.000000 55.00000
## 2     Paul 23.00000  M 10.000000 4.000000 84.00000
## 3    Carli 21.66667  M  9.666667 7.666667 69.33333
## 4     Wale 19.00000  M 19.000000 9.000000 90.00000
## 5     Mary 25.00000  F  5.000000 7.000000 63.00000
## 6    Carli 20.54846  F  0.000000 6.000000 15.00000
## 7    Steve 15.00000  M  7.000000 4.000000 46.00000
```

Beispiel 2.3.4

(zu Python)

Anhang A. R-Code

```
library(mice)

## 
## Attaching package: 'mice'
## The following object is masked from 'package:tidyverse':
##     complete
## The following objects are masked from 'package:base':
##     cbind, rbind

df <- data.frame(colnames = c("John", "Paul", NA, "Wale",
  "Mary", "Carli", "Steve"), Age = c(21, 23, NA, 19, 25,
  NA, 15), Sex = c("M", NA, NA, "M", "F", "F", "M"), Goals = c(5,
  10, NA, 19, 5, 0, 7), Assists = c(7, 4, NA, 9, 7, 6,
  4), Value = c(55, 84, NA, 90, 63, 15, 46))
# perform mice imputation, based on random forests.
miceMod <- mice(df[, !names(df) %in% "medv"] , method = "rf")

## 
## iter imp variable
##   1   1 colnames  Age  Sex  Goals  Assists  Value
##   1   2 colnames  Age  Sex  Goals  Assists  Value
##   1   3 colnames  Age  Sex  Goals  Assists  Value
##   1   4 colnames  Age  Sex  Goals  Assists  Value
##   1   5 colnames  Age  Sex  Goals  Assists  Value
##   2   1 colnames  Age  Sex  Goals  Assists  Value
##   2   2 colnames  Age  Sex  Goals  Assists  Value
##   2   3 colnames  Age  Sex  Goals  Assists  Value
##   2   4 colnames  Age  Sex  Goals  Assists  Value
##   2   5 colnames  Age  Sex  Goals  Assists  Value
##   3   1 colnames  Age  Sex  Goals  Assists  Value
##   3   2 colnames  Age  Sex  Goals  Assists  Value
##   3   3 colnames  Age  Sex  Goals  Assists  Value
##   3   4 colnames  Age  Sex  Goals  Assists  Value
##   3   5 colnames  Age  Sex  Goals  Assists  Value
##   4   1 colnames  Age  Sex  Goals  Assists  Value
##   4   2 colnames  Age  Sex  Goals  Assists  Value
##   4   3 colnames  Age  Sex  Goals  Assists  Value
##   4   4 colnames  Age  Sex  Goals  Assists  Value
##   4   5 colnames  Age  Sex  Goals  Assists  Value
##   5   1 colnames  Age  Sex  Goals  Assists  Value
##   5   2 colnames  Age  Sex  Goals  Assists  Value
##   5   3 colnames  Age  Sex  Goals  Assists  Value
```

Anhang A. R-Code

```
## 5 4 colnames Age Sex Goals Assists Value
## 5 5 colnames Age Sex Goals Assists Value

complete(miceMod)

## colnames Age Sex Goals Assists Value
## 1 John 21 M 5 7 55
## 2 Paul 23 M 10 4 84
## 3 Steve 15 M 5 6 46
## 4 Wale 19 M 19 9 90
## 5 Mary 25 F 5 7 63
## 6 Carli 25 F 0 6 15
## 7 Steve 15 M 7 4 46
```

Beispiel 3.2.1

(zu **Python**)

```
dunif(x = 5, min = 1, max = 10)

## [1] 0.1111111
```

(zu **Python**)

```
punif(q = 5, min = 1, max = 10)

## [1] 0.4444444
```

(zu **Python**)

```
punif(q = 4.8, min = 1, max = 10) - punif(q = 1.2, 1, 10)

## [1] 0.4
```

(zu **Python**)

```
rrunif(n = 5, min = 1, max = 10)

## [1] 5.387321 8.893714 8.935790 4.543805 6.160926
```

Bemerkung: Da hier Zufallszahlen generiert werden, sind **R**- und **Python**-Output auch nicht gleich.

Beispiel 3.2.5

(zu Python)

```
pexp(q = 4, rate = 3)  
  
## [1] 0.9999939
```

Beispiel 3.2.7

(zu Python)

```
1 - pnorm(q = 130, mean = 100, sd = 15)  
  
## [1] 0.02275013
```

(zu Python)

```
qnorm(p = 0.025, mean = 100, sd = 15)  
  
## [1] 70.60054  
  
qnorm(p = 0.975, mean = 100, sd = 15)  
  
## [1] 129.3995
```

(zu Python)

```
qnorm(p = c(0.025, 0.975), mean = 100, sd = 15)  
  
## [1] 70.60054 129.39946
```

(zu Python)

```
pnorm(q = 115, mean = 100, sd = 15) - pnorm(85, 100, 15)  
  
## [1] 0.6826895
```

Beispiel 3.2.8

(zu Python)

```
pnorm(1.13)

## [1] 0.8707619
```

(zu Python)

```
qnorm(0.791)

## [1] 0.8098959
```

(zu Python)

```
pnorm(-0.2)

## [1] 0.4207403

1 - pnorm(0.2)

## [1] 0.4207403
```

Beispiel 3.4.4

(zu Python)

```
werte <- c(0, 10, 11)
ew <- sum(werte * 1/3)
ew

## [1] 7
```

(zu Python)

```
var.X <- sum((werte - ew)^2 * 1/3)
var.X

## [1] 24.66667
```

Anhang A. R-Code

(zu Python)

```
sd.X <- sqrt(var.X)
sd.X

## [1] 4.966555
```

(zu Python)

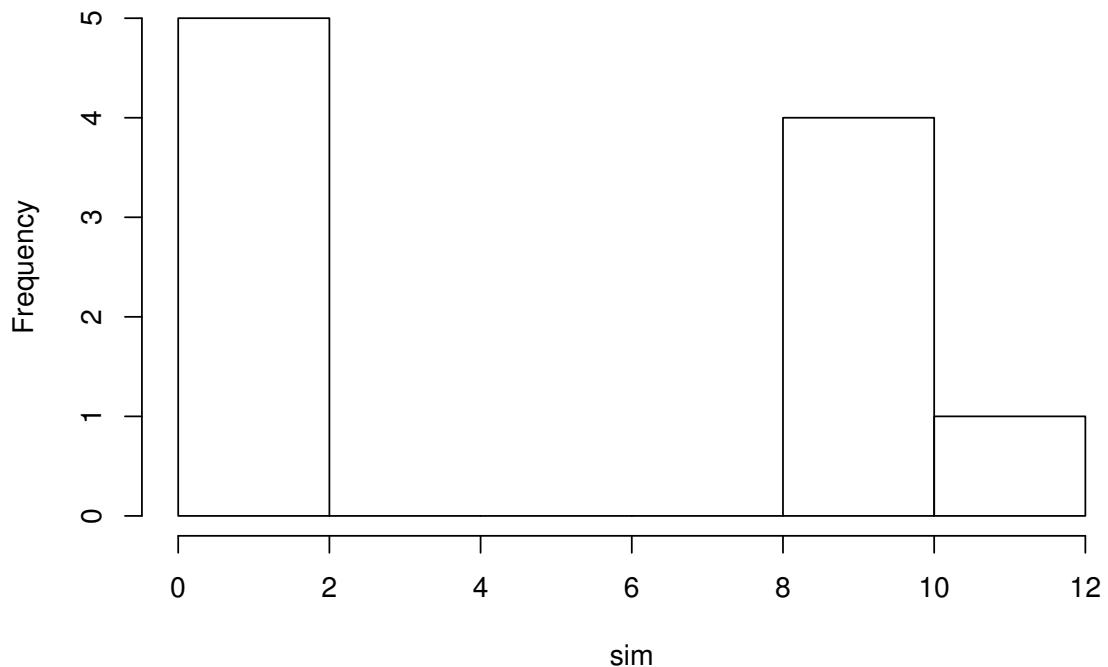
```
# zieht 10-mal aus der Menge {0,10,11} einen Wert mit
# gleicher W'keit
sim <- sample(werte, 10, replace = T)

# Vektor mit 10 Werten
sim

## [1] 0 0 0 10 11 0 0 10 10 10

# Histogramm mit diesen 10 Werten
hist(sim)
```

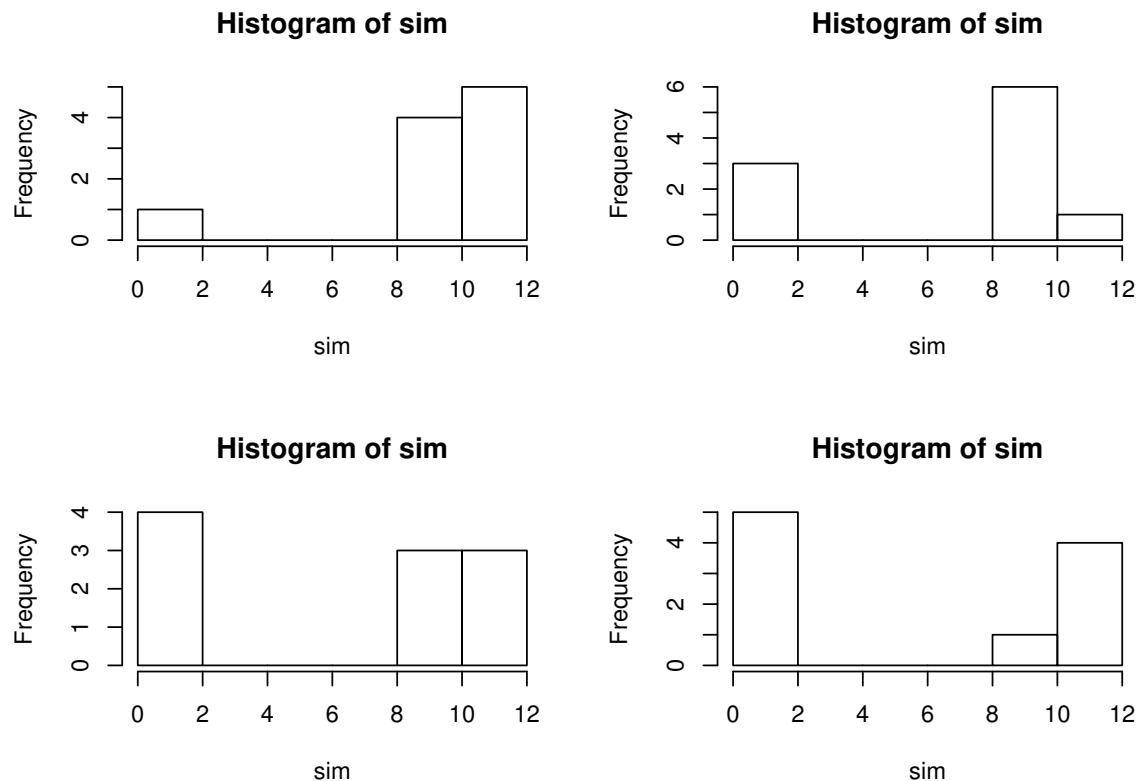
Histogram of sim



Anhang A. R-Code

(zu Python)

```
par(mfrow = c(2, 2))
for (i in 1:4) {
  sim <- sample(werte, 10, replace = T)
  hist(sim)
}
```



(zu Python)

```
sim.1 <- sample(werte, 10, replace = T)
sim.1

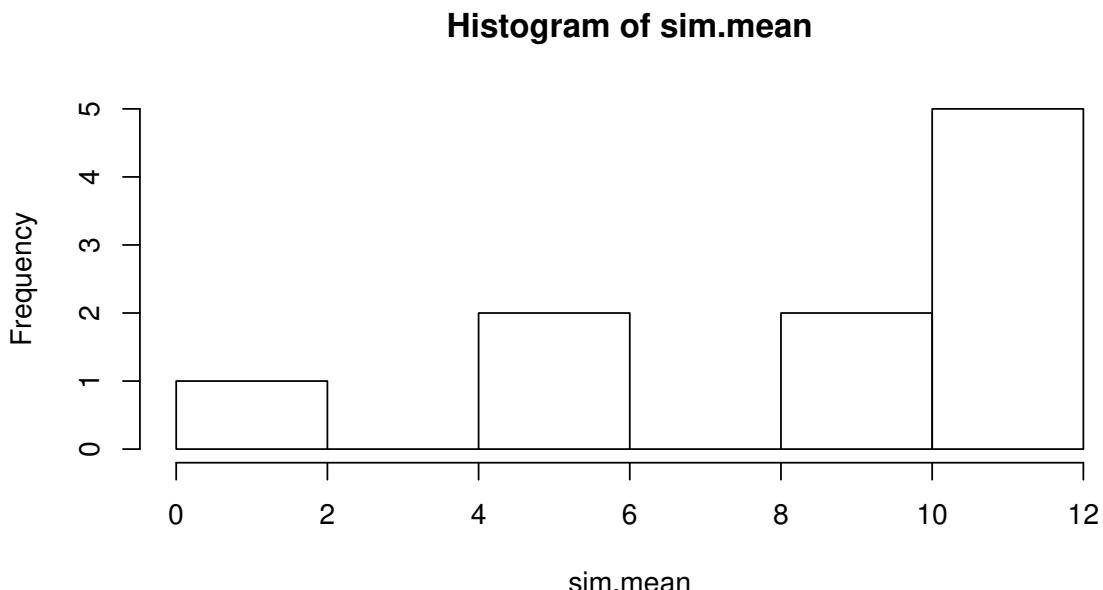
## [1] 0 10 11 11 10 10 11 11 0 11

sim.2 <- sample(werte, 10, replace = T)
sim.2

## [1] 10 10 10 10 10 11 11 0 0 10
```

Anhang A. R-Code

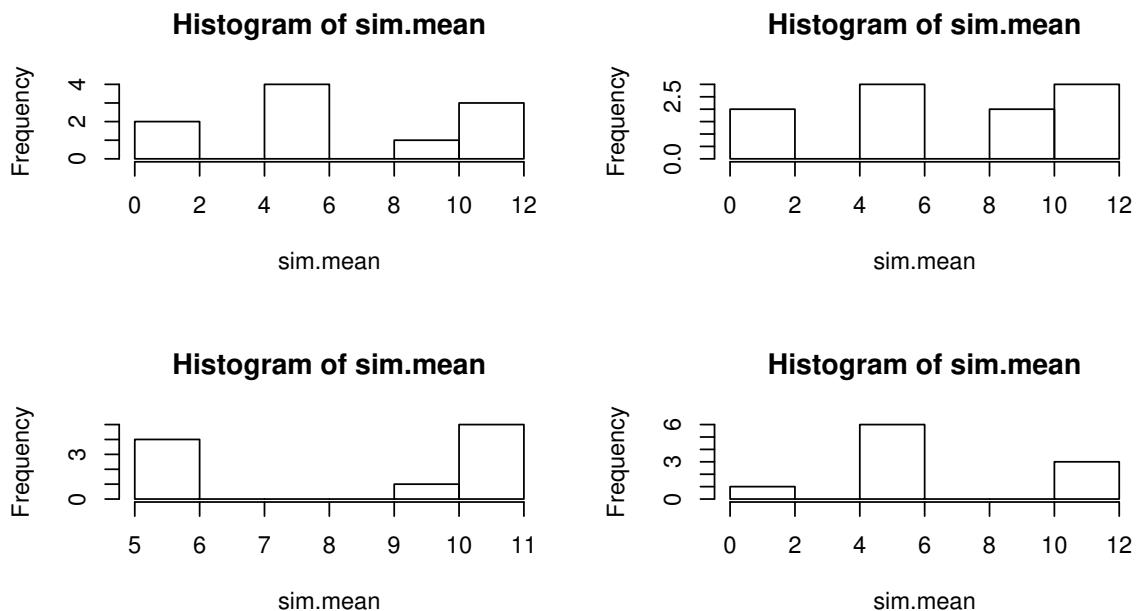
```
sim.mean <- (sim.1 + sim.2)/2  
sim.mean  
  
## [1] 5.0 10.0 10.5 10.5 10.0 10.5 11.0 5.5 0.0 10.5  
  
hist(sim.mean)
```



(zu **Python**)

```
par(mfrow = c(2, 2))  
for (i in 1:4) {  
  sim.1 <- sample(werte, 10, replace = T)  
  sim.2 <- sample(werte, 10, replace = T)  
  sim.mean <- (sim.1 + sim.2)/2  
  hist(sim.mean)  
}
```

Anhang A. R-Code



(zu Python)

```

sim.1 <- sample(werte, 10, replace = T)
sim.1

## [1] 0 11 11 11 11 0 0 0 10 10

sim.2 <- sample(werte, 10, replace = T)
sim.2

## [1] 11 10 0 11 10 11 0 10 0 0

sim.3 <- sample(werte, 10, replace = T)
sim.3

## [1] 10 0 10 11 10 0 11 10 10 11

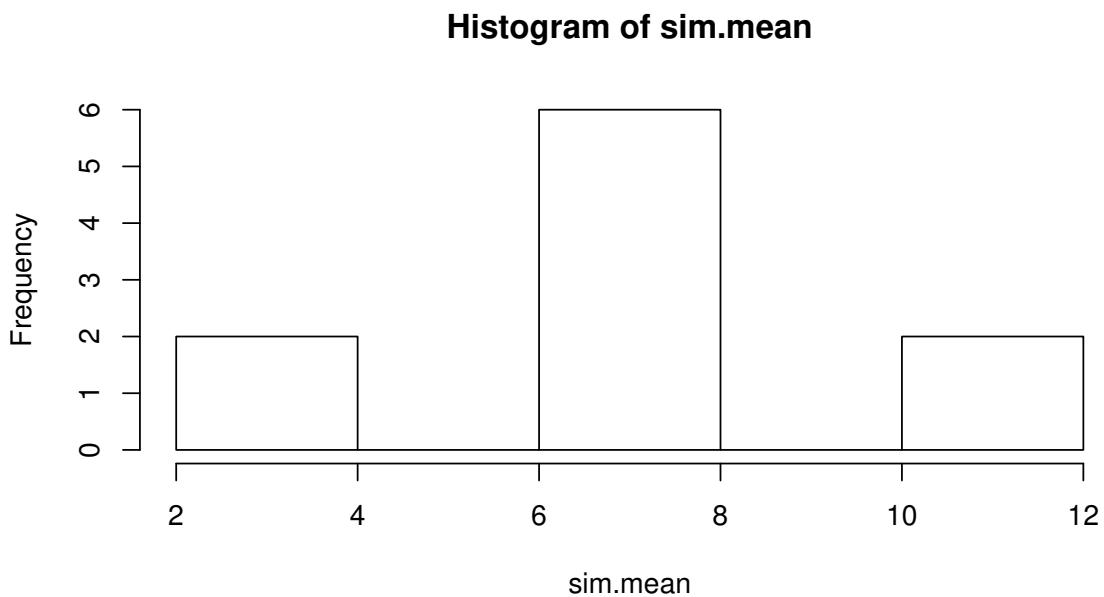
sim.mean <- (sim.1 + sim.2 + sim.3)/3
round(sim.mean, 2)

## [1] 7.00 7.00 7.00 11.00 10.33 3.67 3.67 6.67
## [9] 6.67 7.00

hist(sim.mean)

```

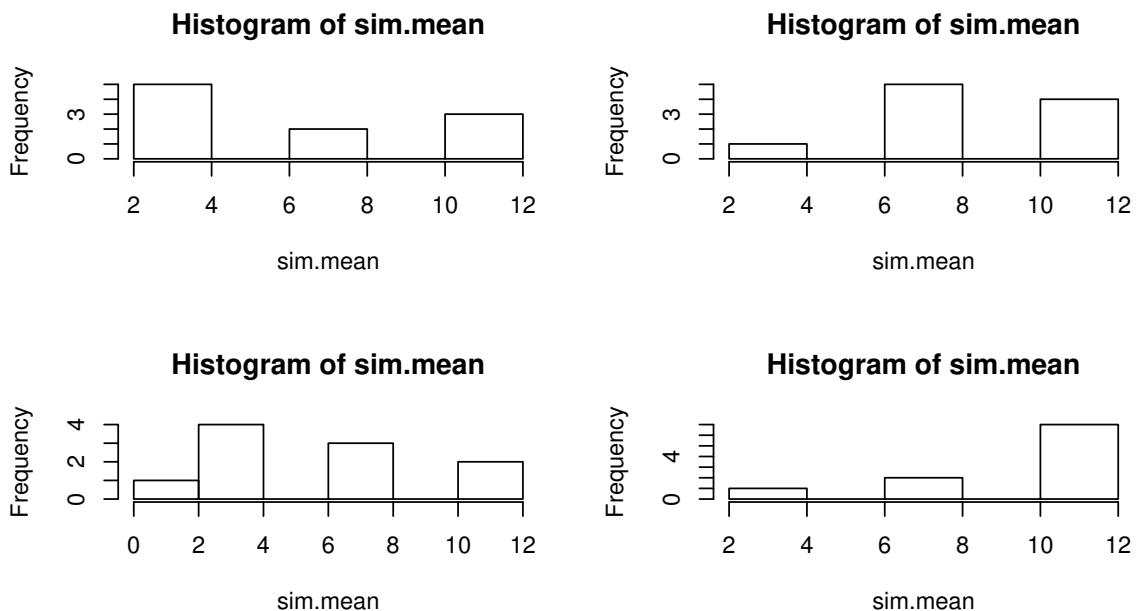
Anhang A. R-Code



(zu **Python**)

```
par(mfrow = c(2, 2))
for (i in 1:4) {
  sim.1 <- sample(werte, 10, replace = T)
  sim.2 <- sample(werte, 10, replace = T)
  sim.3 <- sample(werte, 10, replace = T)
  sim.mean <- (sim.1 + sim.2 + sim.3)/3
  hist(sim.mean)
}
```

Anhang A. R-Code



(zu Python)

```
par(mfrow = c(2, 2))

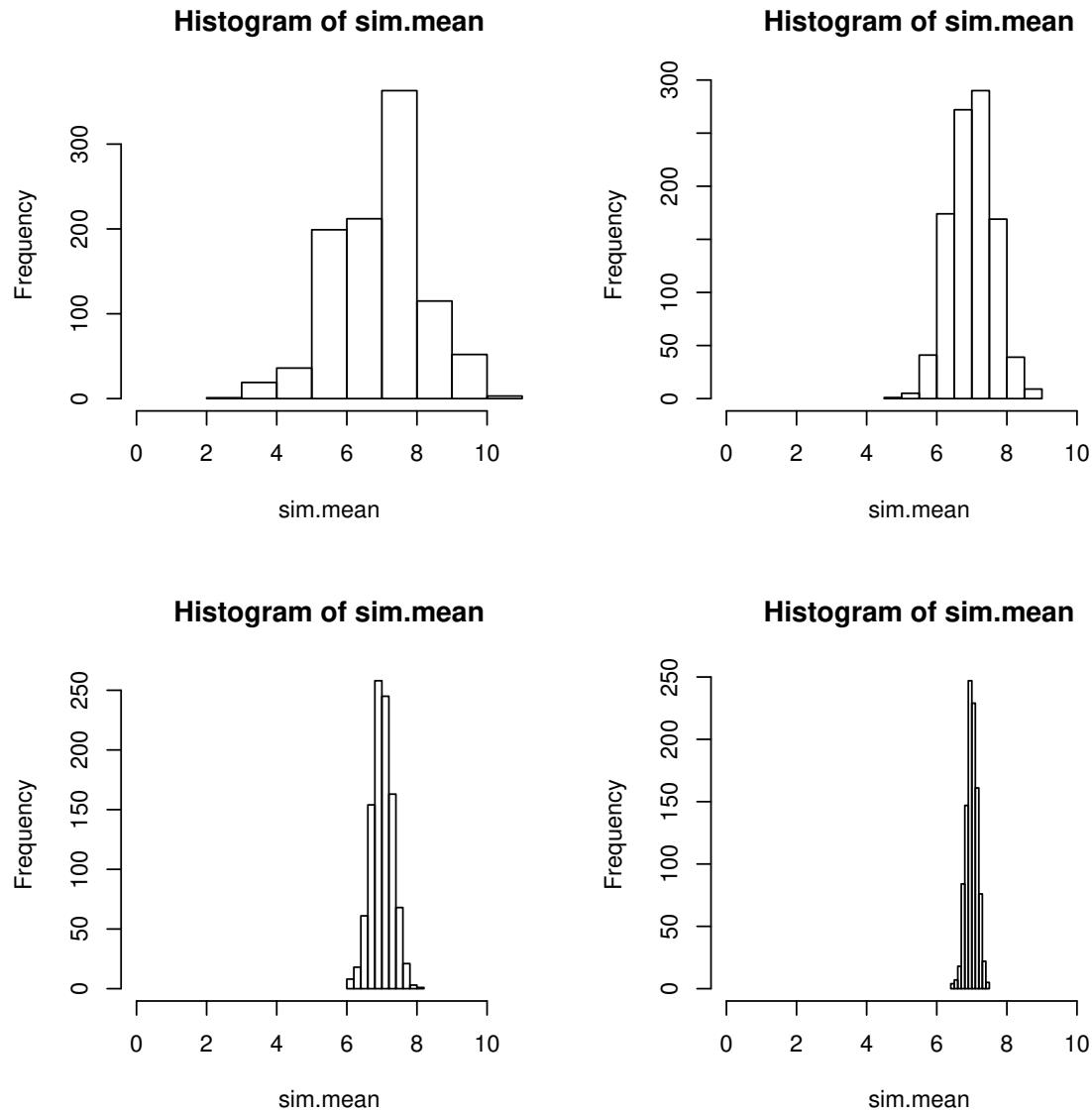
n <- 16
# X_1, ..., X_n simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11))

n <- 64
# X_1, ..., X_n simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11))

n <- 256
# X_1, ..., X_n simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11))
```

Anhang A. R-Code

```
n <- 1024
#  $X_1, \dots, X_n$  simulieren und in einer  $n$ -spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11))
```



(zu [Python](#))

Anhang A. R-Code

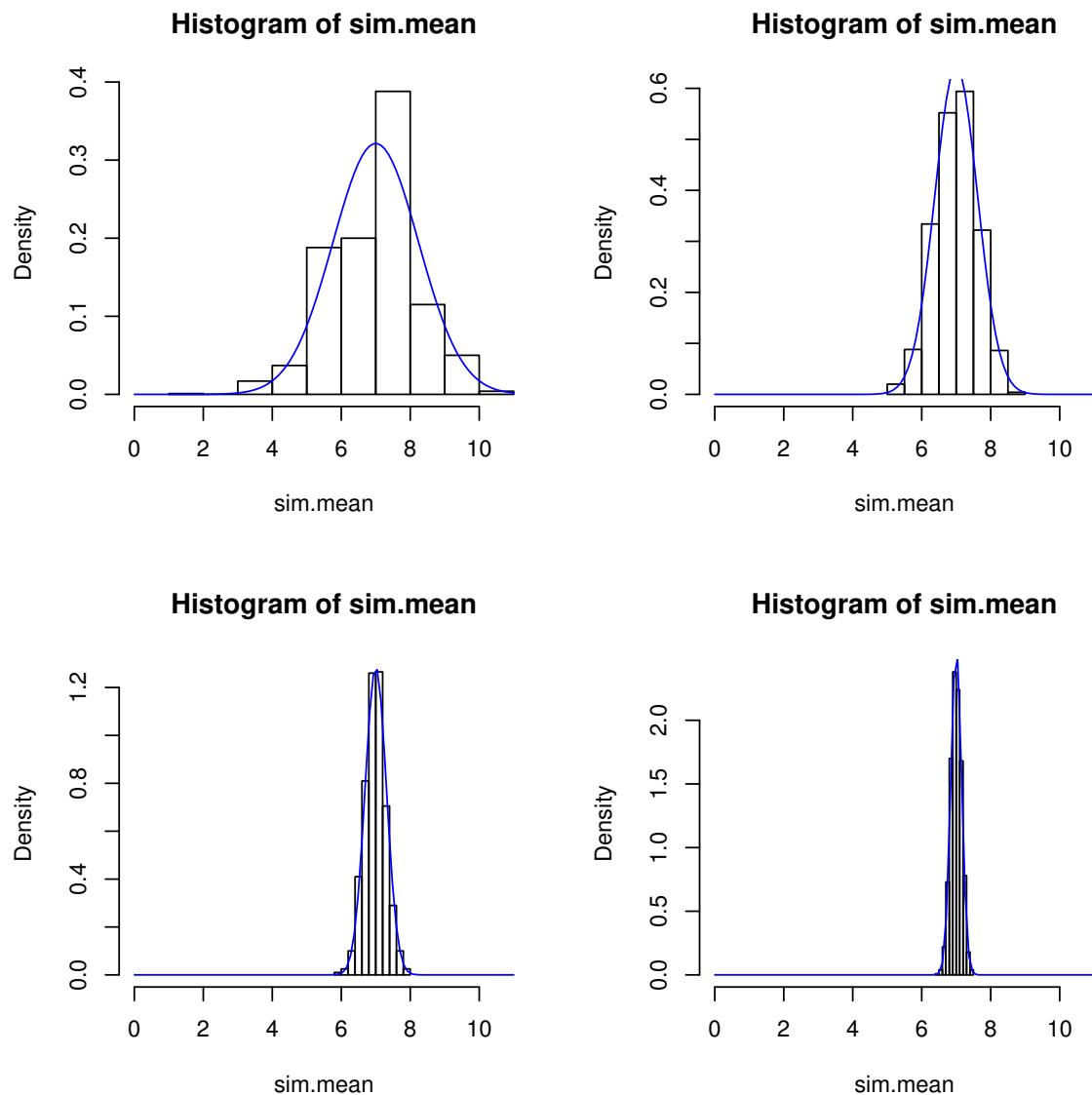
```
par(mfrow = c(2, 2))

n <- 16
#  $X_1, \dots, X_n$  simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11), freq = F)
curve(dnorm(x, 7, sd.X/sqrt(n)), col = "blue", add = T)

n <- 64
#  $X_1, \dots, X_n$  simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11), freq = F)
curve(dnorm(x, 7, sd.X/sqrt(n)), col = "blue", add = T)

n <- 256
#  $X_1, \dots, X_n$  simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11), freq = F)
curve(dnorm(x, 7, sd.X/sqrt(n)), col = "blue", add = T)

n <- 1024
#  $X_1, \dots, X_n$  simulieren und in einer n-spaltigen Matrix
# (mit 1000 Zeilen) anordnen
sim <- matrix(sample(werte, n * 1000, replace = TRUE), ncol = n)
# In jeder Matrixzeile Mittelwert berechnen
sim.mean <- apply(sim, 1, "mean")
hist(sim.mean, xlim = c(0, 11), freq = F)
curve(dnorm(x, 7, sd.X/sqrt(n)), col = "blue", add = T)
```



Beispiel 3.4.5

(zu [Python](#))

```
pnorm(5100, 5000, sqrt(2500))  
## [1] 0.9772499
```

(zu [Python](#))

Anhang A. R-Code

```
pbinom(5100, 10000, 0.5)

## [1] 0.9777871
```

Beispiel 4.1.1

(zu Python)

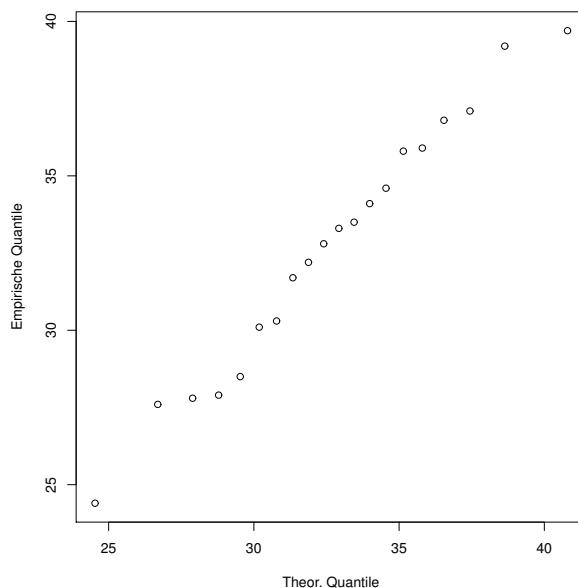
```
x <- c(24.4, 27.6, 27.8, 27.9, 28.5, 30.1, 30.3, 31.7, 32.2,
      32.8, 33.3, 33.5, 34.1, 34.6, 35.8, 35.9, 36.8, 37.1,
      39.2, 39.7)

alphak <- (seq(1, length(x), by = 1) - 0.5)/length(x)

quantile.theor <- qnorm(alphak, mean = mean(x), sd = sd(x))

quantile.empir <- sort(x)

qqplot(quantile.theor, quantile.empir, xlab = "Theor. Quantile",
       ylab = "Empirische Quantile")
```

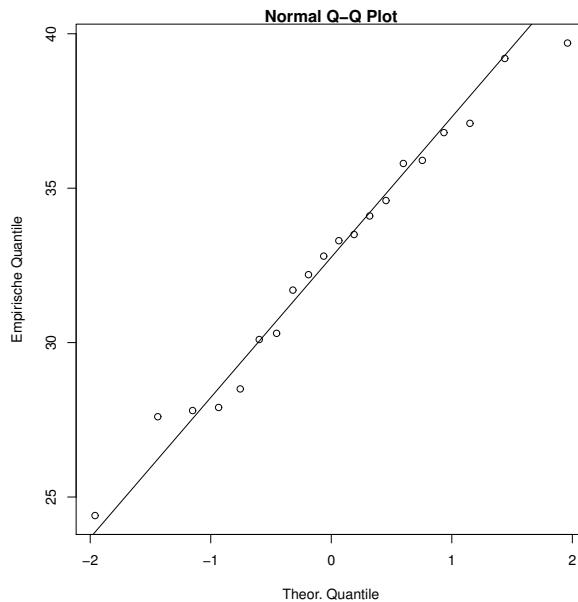


(zu Python)

Der R-Befehl ist hier `qqnorm(...)`. Der Befehl `qqline(...)` zeichnet die gerade Linie.

Anhang A. R-Code

```
x <- c(24.4, 27.6, 27.8, 27.9, 28.5, 30.1, 30.3, 31.7, 32.2,  
      32.8, 33.3, 33.5, 34.1, 34.6, 35.8, 35.9, 36.8, 37.1,  
      39.2, 39.7)  
  
qqnorm(x, xlab = "Theor. Quantile", ylab = "Empirische Quantile")  
qqline(x)
```



Beispiel 4.3.2

Da in diesem Beispiel Zufallsgeneratoren verwendet werden, sind die Resultate oft leicht unterschiedlich.

(zu [Python](#))

```
methodeA <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03,  
            80.04, 79.97, 80.05, 80.03, 80.02, 80, 80.02)  
  
mean(methodeA)  
  
## [1] 80.02077  
  
sd(methodeA)  
  
## [1] 0.02396579
```

Anhang A. R-Code

(zu Python)

```
set.seed(1)
methodeA.sim1 <- rnorm(n = 6, mean = 80, sd = 0.02)

methodeA.sim1

## [1] 79.98747 80.00367 79.98329 80.03191 80.00659
## [6] 79.98359

mean(methodeA.sim1)

## [1] 79.99942

sd(methodeA.sim1)

## [1] 0.0188596
```

(zu Python)

```
set.seed(10)
for (i in 1:5) {
  methodeA.sim1 <- rnorm(n = 6, mean = 80, sd = 0.02)
  cat(mean(methodeA.sim1), sd(methodeA.sim1), "\n")
}

## 79.99516 0.01314937
## 79.99468 0.02128554
## 80.00143 0.01416658
## 79.98611 0.0257259
## 79.98815 0.008686601
```

(zu Python)

```
set.seed(1450070)
methodeA.sim2 <- rnorm(n = 6, mean = 80, sd = 0.02)

methodeA.sim2

## [1] 80.05403 80.03896 80.03671 80.06336 80.01052
## [6] 80.04372
```

Anhang A. R-Code

```
mean(methodeA.sim2)

## [1] 80.04122

sd(methodeA.sim2)

## [1] 0.01804572
```

(zu Python)

```
set.seed(384)
methodeA.sim3 <- rnorm(n = 6, mean = 80, sd = 0.02)

methodeA.sim3

## [1] 80.00420 79.95783 79.96086 79.95553 79.97645
## [6] 79.99413

mean(methodeA.sim3)

## [1] 79.97483

sd(methodeA.sim3)

## [1] 0.02046691
```

Beispiel 4.3.8

(zu Python)

```
pnorm(q = 79.98, mean = 80, sd = 0.02/sqrt(6))

## [1] 0.007152939
```

(zu Python)

```
qnorm(p = c(0.025, 0.975), mean = 80, sd = 0.02/sqrt(6))

## [1] 79.984 80.016
```

Beispiel 4.3.9

(zu Python)

(zu Python)

```
1 - pnorm(q = 80.04, mean = 80, sd = 0.02/sqrt(6))  
## [1] 4.816785e-07
```

Beispiel 4.3.10

(zu Python)

```
qnorm(p = c(0.025, 0.975), mean = 500, sd = 1/sqrt(100))  
## [1] 499.804 500.196
```

Beispiel ??

(zu Python)

```
pnorm(q = 171.54, mean = 180, sd = 10/sqrt(8))  
## [1] 0.008359052
```

Beispiel 4.3.12

(zu Python)

```
pnorm(q = 79.98, mean = 80, sd = 0.02/sqrt(2))  
## [1] 0.0786496
```

(zu Python)

Anhang A. R-Code

```
pnorm(q = 79.98, mean = 80, sd = 0.02/sqrt(4))  
## [1] 0.02275013
```

(zu Python)

```
pnorm(q = 79.98, mean = 80, sd = 0.02/sqrt(6))  
## [1] 0.007152939
```

(zu Python)

```
pnorm(q = 79.98, mean = 80, sd = 0.02/sqrt(8))  
## [1] 0.002338867
```

Beispiel 4.3.14

(zu Python)

```
qnorm(p = c(0.025, 0.975), 80, 0.01/sqrt(13))  
## [1] 79.99456 80.00544
```

Beispiel 4.3.16

(zu Python)

```
x <- c(5.9, 3.4, 6.6, 6.3, 4.2, 2, 6, 4.8, 4.2, 2.1, 8.7,  
      4.4, 5.1, 2.7, 8.5, 5.8, 4.9, 5.3, 5.5, 7.9)  
  
sd(x)  
  
## [1] 1.883802
```

(zu Python)

Diese Funktion steht in R nicht zur Verfügung.

(zu Python)

Anhang A. R-Code

```
x <- c(5.9, 3.4, 6.6, 6.3, 4.2, 2, 6, 4.8, 4.2, 2.1, 8.7,
      4.4, 5.1, 2.7, 8.5, 5.8, 4.9, 5.3, 5.5, 7.9)

mean.x <- mean(x)

sigma.x <- sd(x)

t <- (mean.x - 5) / (sigma.x/sqrt(length(x)) )

pt(t, df = length(x))

## [1] 0.6923238
```

Bei **R** muss die Testvariable standardisiert werden, bei **Python** nicht.

Beispiel 4.3.17

(zu **Python**)

```
qt(p = 0.975, df = 12)

## [1] 2.178813
```

(zu **Python**)

Diese Funktion steht in **R** nicht zur Verfügung.

(zu **Python**)

```
x <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04,
      79.97, 80.05, 80.03, 80.02, 80, 80.02)

t.test(x, alternative = "two.sided", mu = 80, conf.level = 0.95)

##
## One Sample t-test
##
## data: x
## t = 3.1246, df = 12, p-value = 0.008779
## alternative hypothesis: true mean is not equal to 80
## 95 percent confidence interval:
## 80.00629 80.03525
## sample estimates:
```

Anhang A. R-Code

```
## mean of x  
## 80.02077
```

Beispiel 4.4.5

Da in diesem und den folgenden Beispielen der Zufallsgenerator verwendet wird, sind die Resultate leicht anders als bei **Python**.

(zu **Python**)

```
x = c(80.02, 79.94, 79.98, 79.97, 79.97, 80.03, 79.95, 79.97)  
n = length(x)  
  
nboot <- 1  
  
tmpdata = sample(x, n * nboot, replace = TRUE)  
tmpdata  
  
## [1] 79.98 79.98 79.97 79.97 79.94 79.97 79.97 80.03
```

(zu **Python**)

```
mean(tmpdata)  
  
## [1] 79.97625
```

Beispiel 4.4.6

(zu **Python**)

```
nboot <- 20  
  
tmpdata <- sample(x, n * nboot, replace = TRUE)  
  
bootstrapsample <- matrix(tmpdata, nrow = n, ncol = nboot)  
  
xbarstar <- colMeans(bootstrapsample)  
  
xbarstar
```

Anhang A. R-Code

```
## [1] 79.97625 79.98125 79.97500 79.97625 79.98250
## [6] 79.97625 79.99125 79.98125 79.98125 79.96750
## [11] 79.98625 79.96750 79.97750 79.97750 79.99500
## [16] 79.96000 79.98625 79.97500 79.98500 79.97125

sort(xbarstar)

## [1] 79.96000 79.96750 79.96750 79.97125 79.97500
## [6] 79.97500 79.97625 79.97625 79.97625 79.97750
## [11] 79.97750 79.98125 79.98125 79.98125 79.98250
## [16] 79.98500 79.98625 79.98625 79.99125 79.99500
```

(zu **Python**)

```
d <- quantile(xbarstar, probs = c(0.025, 0.975))

cat ("Vertrauensintervall: ", d, "\n")

## Vertrauensintervall: 79.96356 79.99322
```

(zu **Python**)

```
nboot <- 10000

tmpdata <- sample(x, n * nboot, replace = TRUE)

bootstrapsample <- matrix(tmpdata, nrow = n, ncol = nboot)

xbarstar <- colMeans(bootstrapsample)

d <- quantile(xbarstar, probs = c(0.025, 0.975))

cat ("Vertrauensintervall: ", d, "\n")

## Vertrauensintervall: 79.96 80
```

Beispiel 4.4.10

(zu **Python**)

Anhang A. R-Code

```
x <- rnorm(n = 100, mean = 40, sd = 6)
n <- length(x)

nboot <- 20

tmpdata <- sample(x, n * nboot, replace = TRUE)
bootstrapsample <- matrix(tmpdata, nrow = n, ncol = nboot)

xbarstar <- colMeans(bootstrapsample)

d <- quantile(xbarstar, probs = c(0.025, 0.975))

cat("Vertrauensintervall: ", d, "\n")

## Vertrauensintervall: 39.37542 40.96497
```

(zu Python)

```
n <- 100
nboot <- 1000
x <- rnorm(n = n * nboot, mean = 40, sd = 6)

samp <- matrix(x, ncol = nboot)
dim(samp)

## [1] 100 1000

k <- 0

for (i in 1:nboot) {
  y <- samp[, i]
  xbar <- mean(y)
  tmpdata = sample(y, n * nboot, replace = TRUE)
  bootstrapsample = matrix(tmpdata, nrow = n, ncol = nboot)
  xbarstar = colMeans(bootstrapsample)
  deltastar <- xbarstar - xbar
  d <- quantile(deltastar, probs = c(0.025, 0.975))
  if ((xbar - d[2] <= 40) & (40 <= xbar - d[1])) {
    k <- k + 1
  }
}

k

## [1] 951
```

Anhang A. R-Code

(zu Python)

```
n <- 100

nboot <- 1000

x <- rnorm(n = n * nboot, mean = 40, sd = 6)

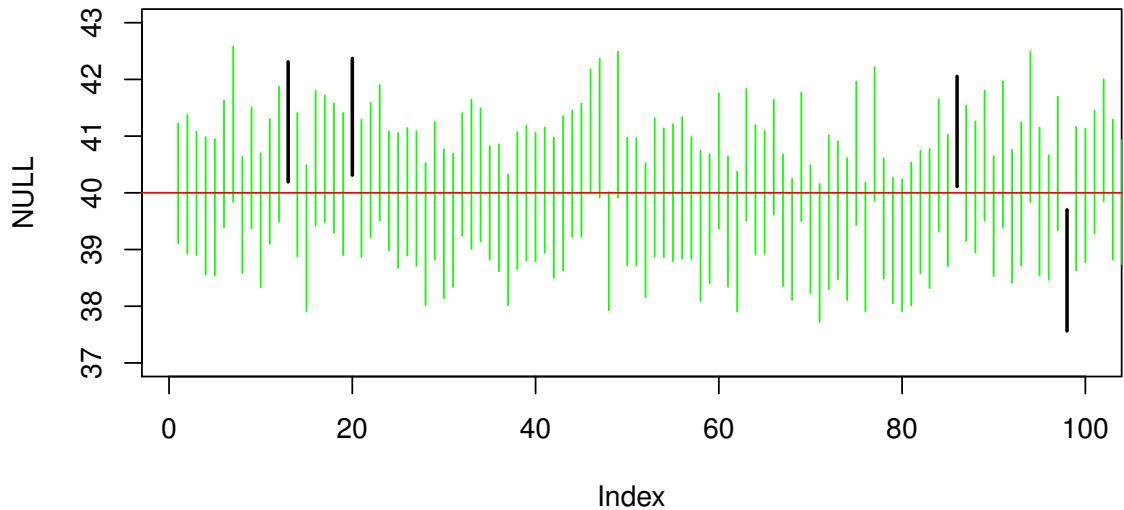
samp <- matrix(x, ncol = nboot)

k <- 0

plot(x = NULL, xlim = c(1, 100), ylim = c(37, 43))

for (i in 1:nboot) {
  y <- samp[, i]
  xbar <- mean(y)
  tmpdata = sample(y, n * nboot, replace = TRUE)
  bootstrapsample = matrix(tmpdata, nrow = n, ncol = nboot)
  xbarstar = colMeans(bootstrapsample)
  deltastar <- xbarstar - xbar
  d <- quantile(deltastar, probs = c(0.025, 0.975))
  lines(c(i, i), c(xbar - d[2], xbar - d[1]), col = "green")
  if (((xbar - d[2] <= 40) & (40 <= xbar - d[1])) == FALSE) {
    lines(c(i, i), c(xbar - d[2], xbar - d[1]), col = "black",
          lwd = 2)
  }
}

abline(h = 40, col = "red")
```



Beispiel 4.4.11

(zu Python)

```
qnorm(p = c(0.025, 0.975))  
  
## [1] -1.959964 1.959964
```

Beispiel 4.4.12

(zu Python)

```
qnorm(p = c(0.025, 0.975), mean = 6, sd = 2)  
  
## [1] 2.080072 9.919928
```

Beispiel 4.4.13

(zu Python) Da es sich hier um eine Zufallszahl handelt, entspricht der Wert unten nicht demjenigen im Skript.

```
rnorm(n = 1, mean = 6, sd = 2)

## [1] 8.453865
```

(zu Python)

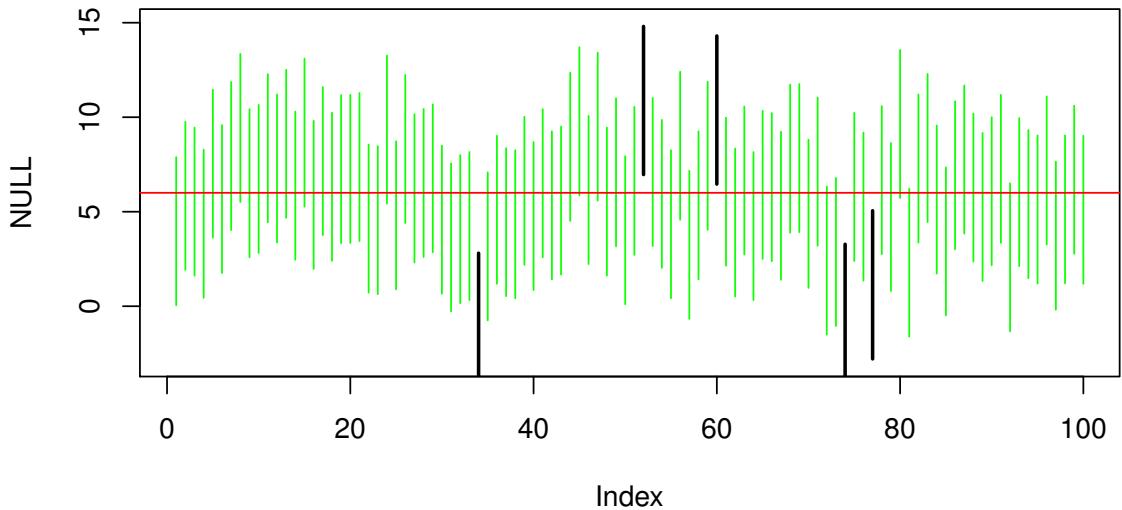
```
x <- rnorm(n = 100, mean = 6, sd = 2)

yu <- x - 1.96 * 2
yo <- x + 1.96 * 2

plot(x = NULL, xlim = c(1, 100), ylim = c(-3, 15))

for (i in 1:100) {
  lines(c(i, i), c(yu[i], yo[i]), col = "green")
  if (((yu[i] <= 6) & (6 <= yo[i]))) == FALSE) {
    lines(c(i, i), c(yu[i], yo[i]), col = "black", lwd = 2)
  }
}

abline(h = 6, col = "red")
```



Beispiel 4.4.14

(zu [Python](#))

```
qnorm(p = c(0.005, 0.995))  
  
## [1] -2.575829 2.575829
```

Beispiel 4.4.15

(zu [Python](#))

Dieser Befehl existiert in [R](#) nicht und muss von Hand berechnet werden.

Beispiel 4.4.16

(zu [Python](#))

```
qt(p = 0.975, df = 12)  
  
## [1] 2.178813
```

Anhang A. R-Code

(zu Python)

Dieser Befehl existiert in R nicht.

Das Vertrauensintervall ist aber im Output von `t.test` enthalten

```
x <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04,
      79.97, 80.05, 80.03, 80.02, 80, 80.02)

t.test(x, alternative = "two.sided", mu = 80, conf.level = 0.95)$conf

## [1] 80.00629 80.03525
## attr(,"conf.level")
## [1] 0.95
```

Beispiel 4.5.1

(zu Python)

```
qbinom(p = c(0.025, 0.975), size = 13, prob = 0.5)

## [1] 3 10
```

(zu Python)

```
pbinom(q = 3, size = 13, prob = 0.5)

## [1] 0.04614258

pbinom(q = 10, size = 13, prob = 0.5)

## [1] 0.9887695
```

(zu Python)

```
1 - pbinom(q = 10, size = 13, prob = 0.5)

## [1] 0.01123047
```

(zu Python)

Anhang A. R-Code

```
binom.test(x = 11, n = 13, p = 0.5)

##
## Exact binomial test
##
## data: 11 and 13
## number of successes = 11, number of trials = 13,
## p-value = 0.02246
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.5455289 0.9807933
## sample estimates:
## probability of success
##                 0.8461538
```

Beispiel 4.5.2

(zu **Python**)

```
x <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04,
      79.97, 80.05, 80.03, 80.02, 80, 80.02)

wilcox.test(x, mu = 80, alternativ = "two.sided")

##
## Wilcoxon signed rank test with continuity
## correction
##
## data: x
## V = 69, p-value = 0.0195
## alternative hypothesis: true location is not equal to 80
```

Beispiel 4.6.4

(zu **Python**)

```
vorher <- c(25, 25, 27, 44, 30, 67, 53, 53, 52, 60, 28)

nachher <- c(27, 29, 37, 56, 46, 82, 57, 80, 61, 59, 43)

t.test(nachher, vorher, alternative = "two.sided", mu = 0,
       paired = TRUE, conf.level = 0.95)
```

Anhang A. R-Code

```
##  
## Paired t-test  
##  
## data: nachher and vorher  
## t = 4.2716, df = 10, p-value = 0.001633  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 4.91431 15.63114  
## sample estimates:  
## mean of the differences  
## 10.27273
```

(zu **Python**)

Dieser Befehl steht in **R** nicht zur Verfügung.

Beispiel 4.6.7

(zu **Python**)

```
x <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04,  
      79.97, 80.05, 80.03, 80.02, 80, 80.02)  
  
y <- c(80.02, 79.94, 79.98, 79.97, 80.03, 79.95, 79.97)  
  
t.test(x, y, alternative = "two.sided", mu = 0, paired = FALSE,  
       conf.level = 0.95)  
  
##  
## Welch Two Sample t-test  
##  
## data: x and y  
## t = 2.8399, df = 9.3725, p-value = 0.01866  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 0.008490037 0.073048425  
## sample estimates:  
## mean of x mean of y  
## 80.02077 79.98000
```

Beispiel 4.6.8

(zu Python)

```
x <- c(79.98, 80.04, 80.02, 80.04, 80.03, 80.03, 80.04, 79.97, 80.05,
      80.03, 80.02, 80, 80.02)

y <- c(80.02, 79.94, 79.98, 79.97, 80.03, 79.95, 79.97)

wilcox.test(x, y, alternative = "two.sided", mu = 0)

##
## Wilcoxon rank sum test with continuity
## correction
##
## data: x and y
## W = 76.5, p-value = 0.01454
## alternative hypothesis: true location shift is not equal to 0
```

Beispiel 6.2.2

(zu Python)

```
reissfestigkeit <- data.frame(HC = rep(c("5%", "10%", "15%",
                                         "20%"), each = 6), Strength = c(7, 8, 15, 11, 9, 10,
                                         12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19,
                                         25, 22, 23, 18, 20))

reissfestigkeit

##      HC Strength
## 1    5%       7
## 2    5%       8
## 3    5%      15
## 4    5%      11
## 5    5%       9
## 6    5%      10
## 7   10%      12
## 8   10%      17
## 9   10%      13
## 10  10%      18
## 11  10%      19
## 12  10%      15
```

Anhang A. R-Code

```
## 13 15%      14
## 14 15%      18
## 15 15%      19
## 16 15%      17
## 17 15%      16
## 18 15%      18
## 19 20%      19
## 20 20%      25
## 21 20%      22
## 22 20%      23
## 23 20%      18
## 24 20%      20

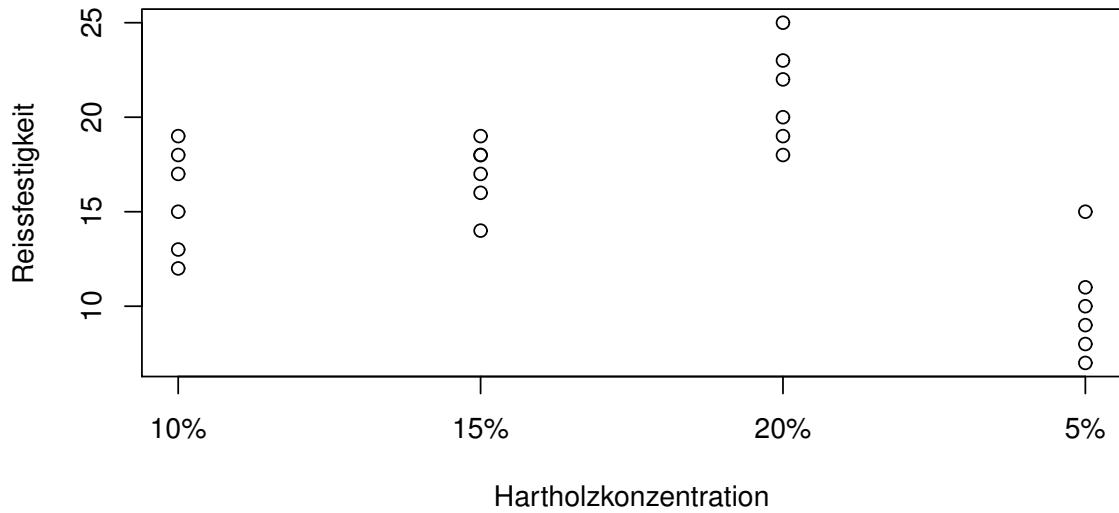
pairwise.t.test(reissfestigkeit$Strength, reissfestigkeit$HC,
                 p.adj = "none", paired = FALSE)

##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  reissfestigkeit$Strength and reissfestigkeit$HC
##
##          10%     15%     20%
## 15% 0.37611 -       -
## 20% 0.00131 0.01037 -
## 5%   0.00101 0.00012 2.6e-07
##
## P value adjustment method: none
```

(zu **Python**)

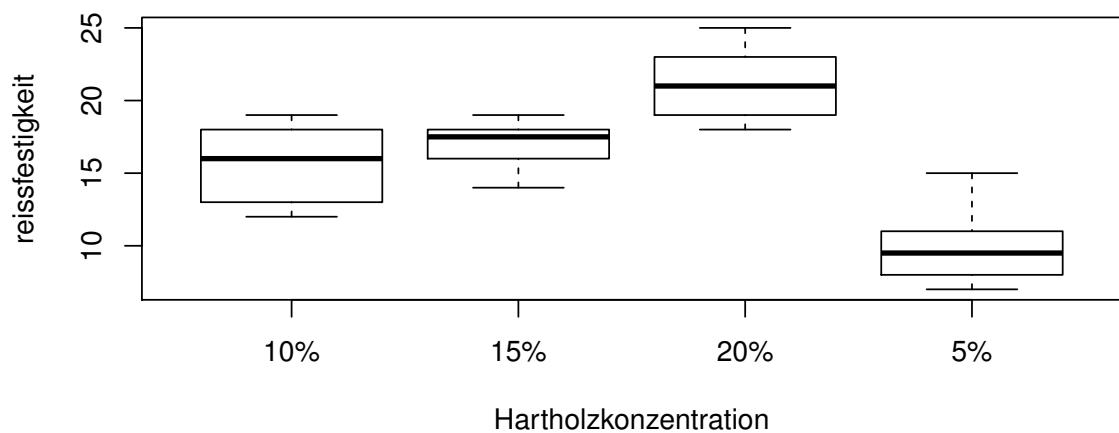
```
reissfestigkeit <- data.frame(HC = rep(c("5%", "10%", "15%",
                                         "20%"), each = 6), Strength = c(7, 8, 15, 11, 9, 10,
                                         12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19,
                                         25, 22, 23, 18, 20))
stripchart(Strength ~ HC, data = reissfestigkeit, pch = 1,
           vertical = TRUE, xlab = "Hartholzkonzentration", ylab = "Reissfestigkeit")
```

Anhang A. R-Code



(zu Python)

```
reissfestigkeit <- data.frame(HC = rep(c("5%", "10%", "15%",  
"20%"), each = 6), Strength = c(7, 8, 15, 11, 9, 10,  
12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19,  
25, 22, 23, 18, 20))  
boxplot(Strength ~ HC, data = reissfestigkeit, xlab = "Hartholzkonzentration",  
ylab = "reissfestigkeit", vertical = TRUE)
```



Beispiel 6.2.4

(zu Python)

```
reissfestigkeit <- data.frame(HC = rep(c("5%", "10%", "15%",
  "20%"), each = 6), Strength = c(7, 8, 15, 11, 9, 10,
  12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19,
  25, 22, 23, 18, 20))
fit <- aov(Strength ~ HC, data = reissfestigkeit)
## Bestimmung der Koeffizienten
dummy.coef(fit)

## Full coefficients are
##
## (Intercept):      15.95833
## HC:                10%          15%          20%
##                   -0.2916667  1.0416667  5.2083333
##
## (Intercept):
## HC:                  5%
##                   -5.9583333
```

(zu Python)

```
reissfestigkeit <- data.frame(HC = rep(c("5%", "10%", "15%",
  "20%"), each = 6), Strength = c(7, 8, 15, 11, 9, 10,
  12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19,
  25, 22, 23, 18, 20))
fit <- aov(Strength ~ HC, data = reissfestigkeit)
predict(fit, newdata = data.frame(HC = c("5%", "10%", "15%",
  "20%")), interval = "confidence", level = 0.95)

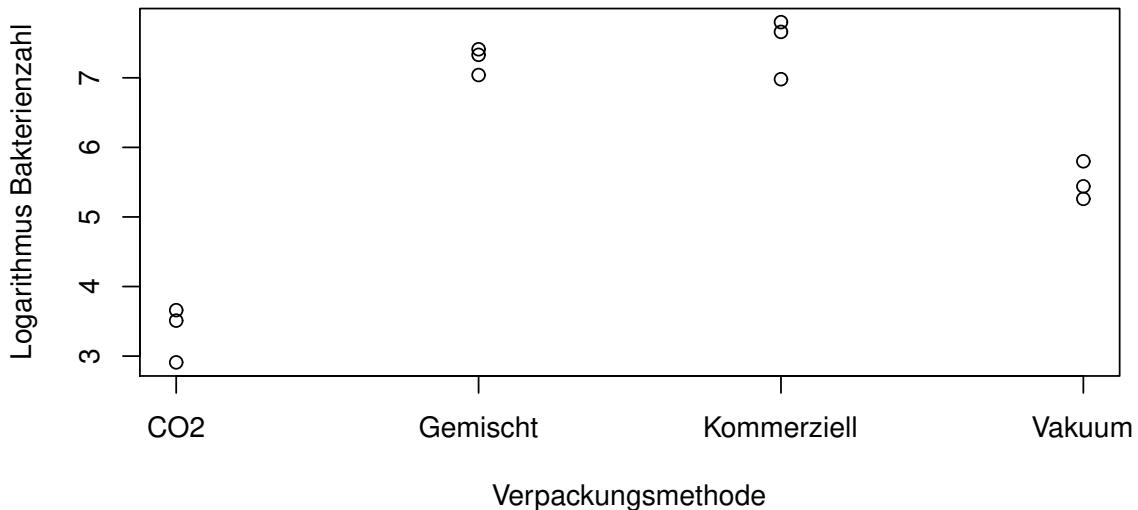
##           fit      lwr      upr
## 1 10.00000 7.827469 12.17253
## 2 15.66667 13.494136 17.83920
## 3 17.00000 14.827469 19.17253
## 4 21.16667 18.994136 23.33920
```

Beispiel 6.2.5

(zu Python)

Anhang A. R-Code

```
meat <- data.frame(steak.id = c(1, 6, 7, 12, 5, 3, 10, 9,
  2, 8, 4, 11), treatment = rep(c("Kommerziell", "Vakuum",
  "Gemischt", "CO2"), each = 3), y = c(7.66, 6.98, 7.8,
  5.26, 5.44, 5.8, 7.41, 7.33, 7.04, 3.51, 2.91, 3.66))
stripchart(y ~ treatment, data = meat, pch = 1, vertical = TRUE,
  xlab = "Verpackungsmethode", ylab = "Logarithmus Bakterienzahl")
```



(zu [Python](#))

```
meat <- data.frame(steak.id = c(1, 6, 7, 12, 5, 3, 10, 9,
  2, 8, 4, 11), treatment = rep(c("Kommerziell", "Vakuum",
  "Gemischt", "CO2"), each = 3), y = c(7.66, 6.98, 7.8,
  5.26, 5.44, 5.8, 7.41, 7.33, 7.04, 3.51, 2.91, 3.66))
levels(meat$treatment)

## [1] "CO2"          "Gemischt"      "Kommerziell"
## [4] "Vakuum"

fit <- aov(y ~ treatment, data = meat)
## Bestimmung der Koeffizienten
dummy.coef(fit)

## Full coefficients are
##
## (Intercept):      5.9
## treatment:        CO2  Gemischt Kommerziell Vakuum
##                   -2.54     1.36      1.58   -0.40
```

Anhang A. R-Code

(zu Python)

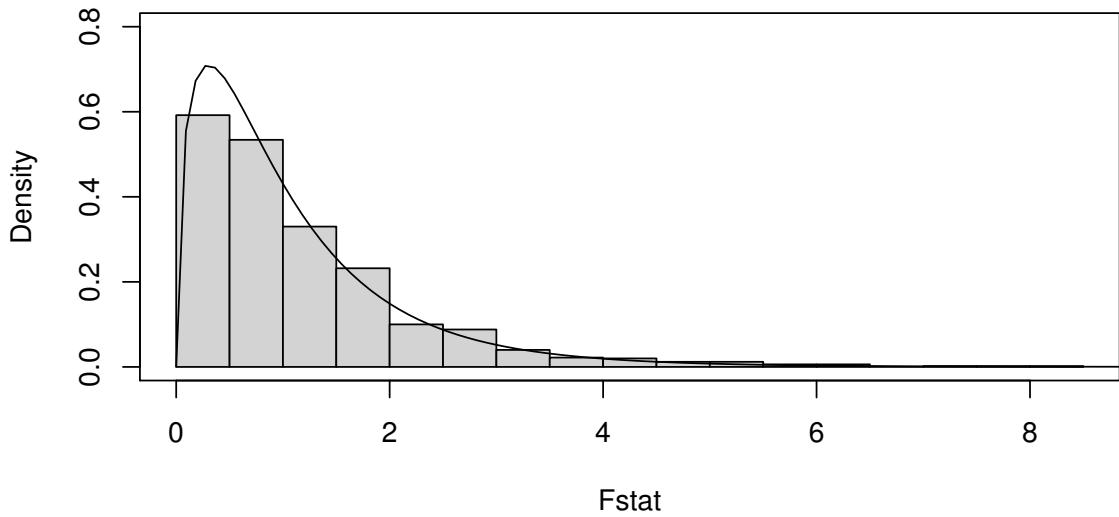
```
meat <- data.frame(steak.id = c(1, 6, 7, 12, 5, 3, 10, 9,
  2, 8, 4, 11), treatment = rep(c("Kommerziell", "Vakuum",
  "Gemischt", "CO2"), each = 3), y = c(7.66, 6.98, 7.8,
  5.26, 5.44, 5.8, 7.41, 7.33, 7.04, 3.51, 2.91, 3.66))
fit <- aov(y ~ treatment, data = meat)
predict(fit, newdata = data.frame(treatment = c("Kommerziell",
  "Vakuum", "Gemischt", "CO2")), interval = "confidence",
  level = 0.95)

##      fit      lwr      upr
## 1 7.48 7.026844 7.933156
## 2 5.50 5.046844 5.953156
## 3 7.26 6.806844 7.713156
## 4 3.36 2.906844 3.813156
```

Beispiel 6.2.7

(zu Python)

```
n <- 24
g <- 4
m <- 6
Fstat <- NULL
for (i in 1:1000) {
  reissfestigkeit.sim <- rnorm(n, mean = 15.958, sd = 2.55)
  reissfestigkeit.mat <- matrix(reissfestigkeit.sim, ncol = 4)
  grand.mean <- mean(reissfestigkeit.sim)
  group.mean <- apply(reissfestigkeit.mat, MARGIN = 2,
    mean)
  MSG <- m * sum((group.mean - grand.mean)^2)/(g - 1)
  MST <- sum(apply((apply(reissfestigkeit.mat, MARGIN = 1,
    function(x) x - group.mean))^2, MARGIN = 1, sum))/(n -
    g)
  Fstat[i] <- MSG/MST
}
hist(Fstat, freq = FALSE, col = "lightgrey", breaks = 25,
  main = "", ylim = c(0, 0.8))
box()
curve(df(x, df1 = 3, df2 = 20), from = 0, to = max(Fstat) +
  1, add = T, ylim = c(0, 0.8))
```



Beispiel 6.2.8

(zu Python)

```
reissfestigkeit <- data.frame(HC = rep(c("5%", "10%", "15%", "20%"), each = 6), Strength = c(7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18, 19, 17, 16, 18, 19, 25, 22, 23, 18, 20))
fit <- aov(Strength ~ HC, data = reissfestigkeit)
summary(fit)

##                               Df  Sum Sq Mean Sq F value    Pr(>F)
## HC                  3   382.8  127.60   19.61 3.59e-06 ***
## Residuals        20   130.2     6.51
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel 6.2.9

(zu Python)

Anhang A. R-Code

```
meat <- data.frame(steak.id = c(1, 6, 7, 12, 5, 3, 10, 9,
  2, 8, 4, 11), treatment = rep(c("Kommerziell", "Vakuum",
  "Gemischt", "CO2"), each = 3), y = c(7.66, 6.98, 7.8,
  5.26, 5.44, 5.8, 7.41, 7.33, 7.04, 3.51, 2.91, 3.66))
fit <- aov(y ~ treatment, data = meat)
summary(fit)

##                                Df  Sum Sq Mean Sq F value    Pr(>F)
## treatment          3  32.87  10.958   94.58 1.38e-06 ***
## Residuals         8    0.93    0.116
## ---
## Signif. codes:
## 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel 6.4.1

(zu Python)

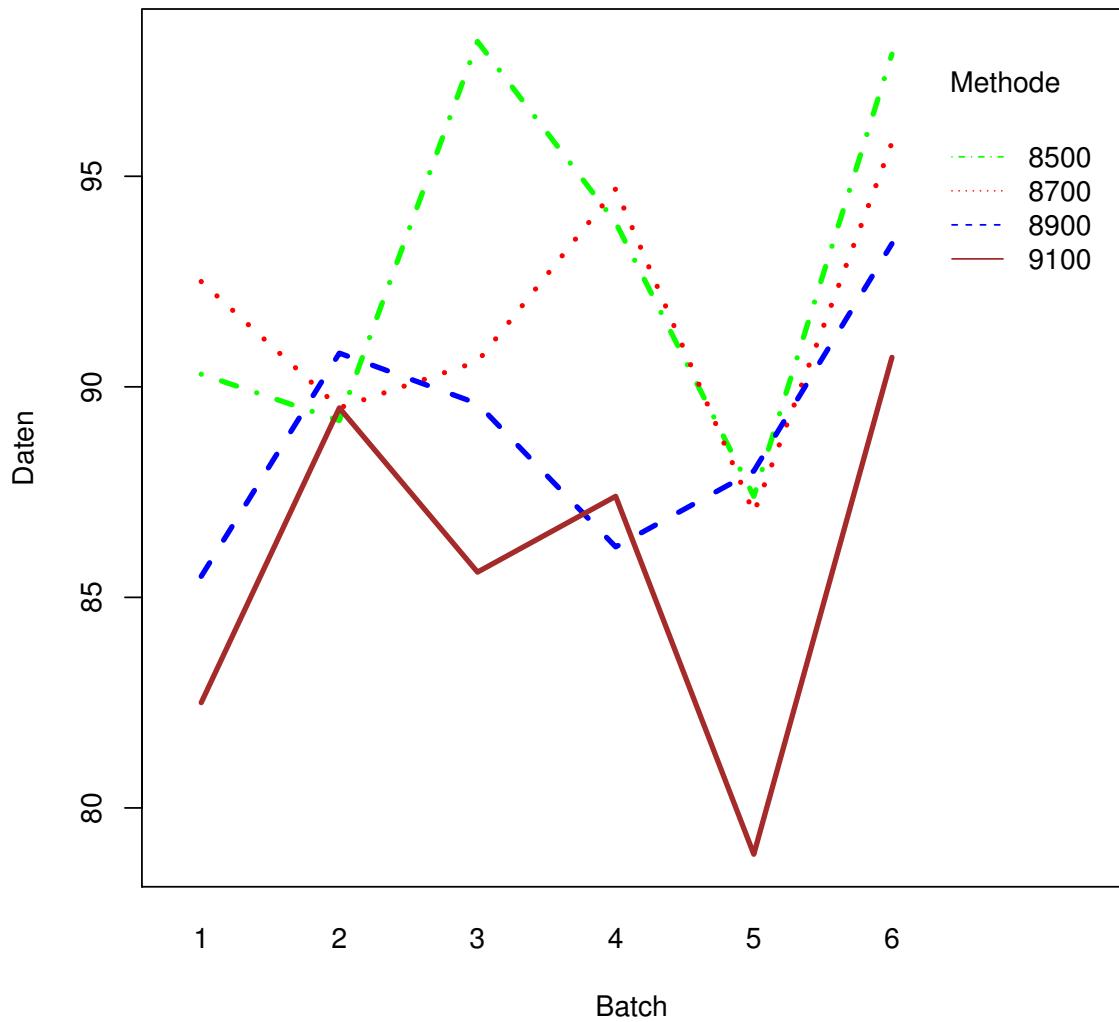
```
sample(1:24)

##  [1] 5 17 13 4 19 18 3 15 8 9 16 21 10 2 24 14 23 11 20 1
## [21] 6 7 22 12
```

Beispiel 6.4.2

(zu Python)

```
Daten <- data.frame(Batch <- rep(c("1", "2", "3", "4", "5", "6"),
  4), Methode <- rep(c("8500", "8700", "8900", "9100"), c(6, 6,
  6, 6)), Y <- c(90.3, 89.2, 98.2, 93.9, 87.4, 97.9, 92.5, 89.5,
  90.6, 94.7, 87, 95.8, 85.5, 90.8, 89.6, 86.2, 88, 93.4, 82.5,
  89.5, 85.6, 87.4, 78.9, 90.7))
interaction.plot(x.factor = Daten$Batch, trace.factor = Daten$Methode,
  response = Daten$Y, lwd = 3, col = c("green", "red", "blue", "brown"),
  xlab = "Batch", ylab = "Daten", trace.label = "Methode")
```



Beispiel 6.4.4

(zu [Python](#))

```
Daten <- data.frame(Batch = factor(rep(c(1, 2, 3, 4, 5, 6), 4)), Methode = factor(c(8500, 8700, 8900, 9100), c(6, 6, 6, 6))), Y <- c(90.3, 89.2, 98.2, 93.9, 87.4, 97.9, 92.5, 89.5, 90.6, 94.7, 87, 95.8, 85.5, 90.8, 89.6, 86.2, 88, 93.4, 82.5, 89.5, 85.6, 87.4, 78.9, 90.7))
# Damit sich die Behandlungseffekte zu null aufsummieren
options(contrasts = c("contr.sum", "contr.sum"))
fit <- aov(Y ~ Methode + Batch, data = Daten)
```

Anhang A. R-Code

```
dummy.coef(fit)

## Full coefficients are
##
## (Intercept):          89.79583
## Methode:              8500        8700        8900        9100
##                         3.0208333  1.8875000 -0.8791667 -4.0291667
## Batch:                 1           2           3           4
##                      -2.09583333 -0.04583333  1.20416667  0.75416667
##
## (Intercept):
## Methode:
## 
## Batch:                  5           6
##                      -4.47083333  4.65416667
```

Beispiel 6.4.5

(zu Python)

```
Daten <- data.frame(Batch = factor(rep(c(1, 2, 3, 4, 5, 6), 4)), Methode = factor(
  8700, 8900, 9100), c(6, 6, 6, 6)))
Y <- c(90.3, 89.2, 98.2, 93.9,
  87.4, 97.9, 92.5, 89.5, 90.6, 94.7, 87, 95.8, 85.5, 90.8, 89.6,
  86.2, 88, 93.4, 82.5, 89.5, 85.6, 87.4, 78.9, 90.7))
# Damit sich die Behandlungseffekte zu null aufsummieren
options(contrasts = c("contr.sum", "contr.sum"))
fit <- aov(Y ~ Methode + Batch, data = Daten)
summary(fit)

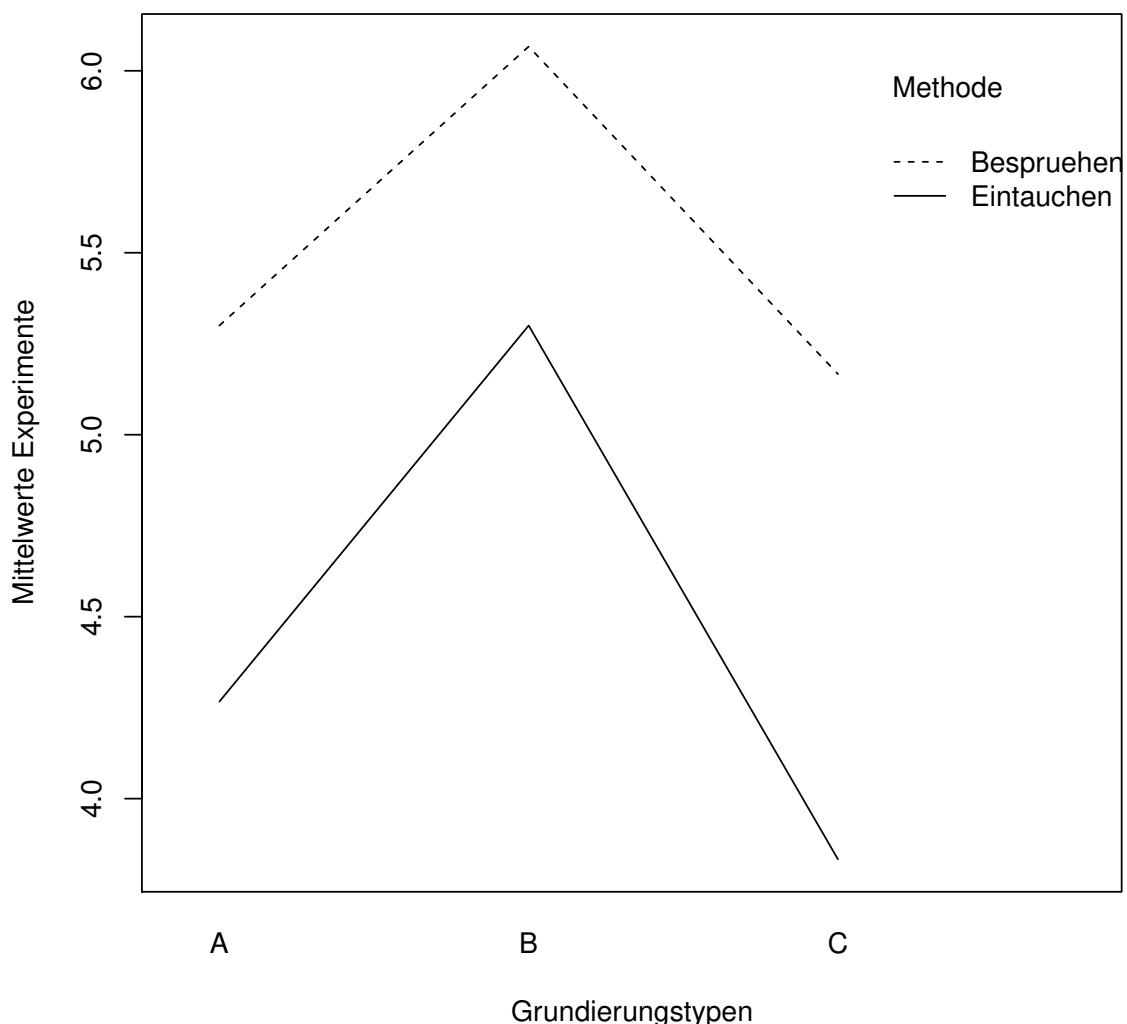
##                               Df Sum Sq Mean Sq F value    Pr(>F)
## Methode                  3 178.2   59.39   8.107 0.00192 ***
## Batch                     5 192.2   38.45   5.249 0.00553 ***
## Residuals                 15 109.9    7.33
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel 7.1.1

(zu Python)

Anhang A. R-Code

```
Farbe <- data.frame(Grund = factor(rep(c("A", "B", "C"), c(6, 6, 6))),  
Methode = factor(rep(rep(c("Eintauchen", "Bespruehen"), c(3, 3)),  
3)), Y <- c(4, 4.5, 4.3, 5.4, 4.9, 5.6, 5.6, 4.9, 5.4, 5.8,  
6.1, 6.3, 3.8, 3.7, 4, 5.5, 5, 5))  
options(contrasts = c("contr.sum", "contr.sum"))  
interaction.plot(x.factor = Farbe$Grund, trace.factor = Farbe$Methode,  
response = Farbe$Y, trace.label = "Methode", pch = 20, xlab = "Grundierungstypen",  
ylab = "Mittelwerte Experimente")
```



Beispiel 6.4.11

Anhang A. R-Code

(zu Python)

```
El <- data.frame(Konz = factor(rep(c("A", "B", "C", "D"), c(6, 6, 6, 6))), Temp = factor(rep(rep(c("15C", "25C"), c(3, 3)), 4))), Y <- c(82, 46, 16, 20, 13, 7, 20, 14, 17, 6, 7, 5, 8, 6, 5, 4, 3, 5, 10, 7, 5, 6, 4, 5))
options(contrasts = c("contr.sum", "contr.sum"))
El.aov <- aov(Y ~ Konz * Temp, data = El)
dummy.coef(El.aov)

## Full coefficients are
##
## (Intercept):          13.375
## Konz:                  A          B          C          D
##                      17.291667 -1.875000 -8.208333 -7.208333
## Temp:                  15C        25C
##                      6.291667 -6.291667
## Konz:Temp:             A:15C      B:15C      C:15C      D:15C
##                      11.0416667 -0.7916667 -5.1250000 -5.1250000
## 
## (Intercept):
## Konz:
## 
## Temp:
## 
## Konz:Temp:             A:25C      B:25C      C:25C      D:25C
##                      -11.0416667  0.7916667  5.1250000  5.1250000
```

Beispiel 6.4.12

(zu Python)

```
El <- data.frame(Konz = factor(rep(c("A", "B", "C", "D"), c(6, 6, 6, 6))), Temp = factor(rep(rep(c("15C", "25C"), c(3, 3)), 4))), Y <- c(82, 46, 16, 20, 13, 7, 20, 14, 17, 6, 7, 5, 8, 6, 5, 4, 3, 5, 10, 7, 5, 6, 4, 5))
El.aov <- aov(Y ~ Konz * Temp, data = El)
summary(El.aov)

##              Df Sum Sq Mean Sq F value Pr(>F)
## Konz          3   2531   843.7   5.844 0.0068 **
## Temp          1     950    950.0   6.580 0.0208 *
## Konz:Temp     3   1050   350.2   2.425 0.1034
```

Anhang A. R-Code

```
## Residuals   16   2310   144.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(zu Python)

```
E1 <- data.frame(Konz = factor(rep(c("A", "B", "C", "D"), c(6, 6,
  6, 6))), Temp = factor(rep(rep(c("15C", "25C"), c(3, 3)), 4)),
  Y <- c(82, 46, 16, 20, 13, 7, 20, 14, 17, 6, 7, 5, 8, 6, 5, 4,
  3, 5, 10, 7, 5, 6, 4, 5))
options(contrasts = c("contr.sum", "contr.sum"))
E1.aov.T <- aov(Y ~ Konz * Temp, data = E1)
summary(E1.aov.T)

##                               Df  Sum Sq Mean Sq F value    Pr(>F)
## Konz                  3  0.08070 0.02690     15.20 6.07e-05 ***
## Temp                  1  0.03905 0.03905     22.06 0.000242 ***
## Konz:Temp              3  0.00324 0.00108      0.61 0.618020
## Residuals             16  0.02832 0.00177
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel 6.4.13

(zu Python)

```
Farbe <- data.frame(Grund = factor(rep(c("A", "B", "C"), c(6, 6, 6))),
  Methode = factor(rep(rep(c("Eintauchen", "Bespr\\"uhnen"), c(3,
  3)), 3)), Y <- c(4, 4.5, 4.3, 5.4, 4.9, 5.6, 5.6, 4.9, 5.4,
  5.8, 6.1, 6.3, 3.8, 3.7, 4, 5.5, 5, 5))
options(contrasts = c("contr.sum", "contr.sum"))
Farbe.aov <- aov(Y ~ Grund * Methode, data = Farbe)
summary(Farbe.aov)

##                               Df  Sum Sq Mean Sq F value    Pr(>F)
## Grund                  2  4.581   2.291  27.858 3.10e-05 ***
## Methode                 1  4.909   4.909  59.703 5.36e-06 ***
## Grund:Methode          2  0.241   0.121   1.466    0.269
## Residuals                12  0.987   0.082
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel 6.4.14

(zu Python)

```
## Eingabe der Daten ##
season <- factor(rep(c("Spring", "Summer"), each = 6))
density <- factor(rep(c(6, 12, 24), each = 3))
y <- c(1.17, 0.5, 1.67, 1.5, 0.83, 1, 0.67, 0.67, 0.75, 4, 3.83, 3.83,
      3.33, 2.58, 2.75, 2.54, 1.83, 1.63)

design <- expand.grid(density = factor(c(6, 12, 24)), season = c("Spring",
    "Summer"))
snails <- data.frame(design[rep(1:nrow(design), each = 3), ], y = y)

## Wir fitten Zweiweg-ANOVA-Modell mit Interaktion

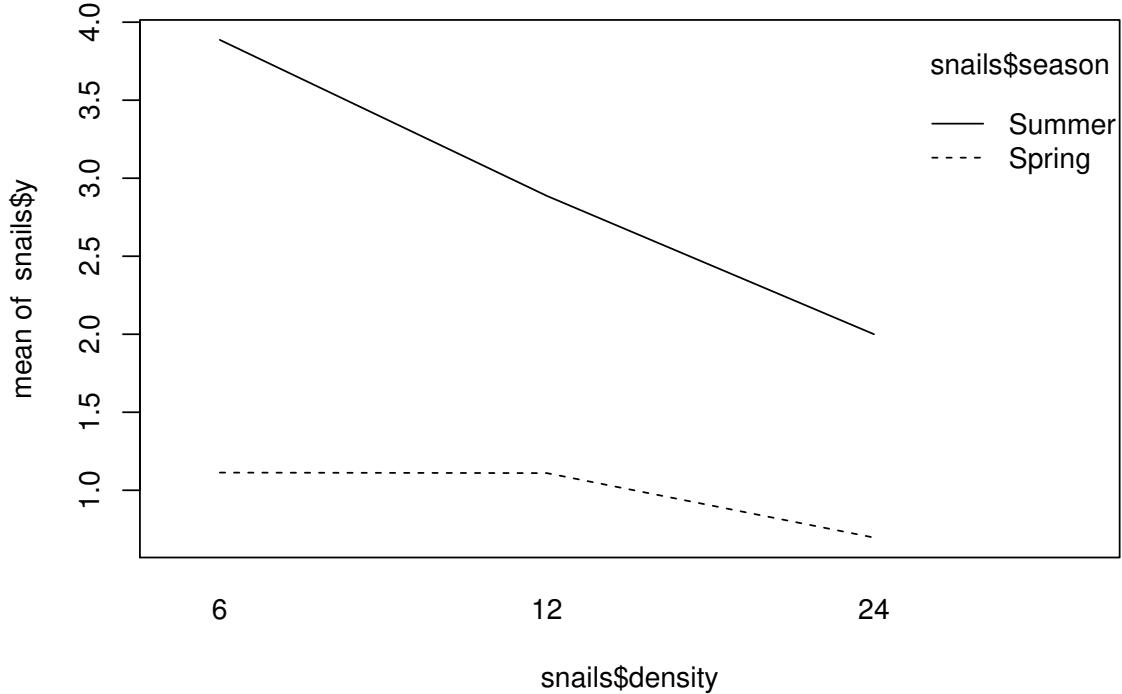
fit <- aov(y ~ season * density, data = snails)
summary(fit)

##                                Df Sum Sq Mean Sq F value    Pr(>F)
## season                  1 17.131 17.131 119.373 1.36e-07 ***
## density                  2   4.001   2.001  13.940 0.000742 ***
## season:density           2   1.689   0.845   5.885 0.016552 *
## Residuals                12   1.722   0.144
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(zu Python)

```
## Wir betrachten den Interaktionsplot #####
interaction.plot(snails$density, snails$season, snails$y)
```

Anhang A. R-Code



(zu **Python**)

```
## Wir passen ein Modell pro Saison an ##
fit.spring <- aov(y ~ density, data = subset(snails, season == "Spring"))
fit.summer <- aov(y ~ density, data = subset(snails, season == "Summer"))

summary(fit.spring)

##                               Df Sum Sq Mean Sq F value Pr(>F)
## density                  2 0.3445  0.1722   1.104   0.391
## Residuals                 6 0.9361  0.1560

summary(fit.summer)

##                               Df Sum Sq Mean Sq F value    Pr(>F)
## density                  2  5.346   2.673   20.41 0.00211 ***
## Residuals                 6  0.786   0.131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel 7.3.1

(zu [Python](#))

Der Datensatz **AirPassengers** ist in **R** selbst enthalten.

```
data(AirPassengers)
AirPassengers

##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949 112 118 132 129 121 135 148 148 136 119 104 118
## 1950 115 126 141 135 125 149 170 170 158 133 114 140
## 1951 145 150 178 163 172 178 199 199 184 162 146 166
## 1952 171 180 193 181 183 218 230 242 209 191 172 194
## 1953 196 196 236 235 229 243 264 272 237 211 180 201
## 1954 204 188 235 227 234 264 302 293 259 229 203 229
## 1955 242 233 267 269 270 315 364 347 312 274 237 278
## 1956 284 277 317 313 318 374 413 405 355 306 271 306
## 1957 315 301 356 348 355 422 465 467 404 347 305 336
## 1958 340 318 362 348 363 435 491 505 404 359 310 337
## 1959 360 342 406 396 420 472 548 559 463 407 362 405
## 1960 417 391 419 461 472 535 622 606 508 461 390 432
```

Und er ist schon eine Zeitreihe:

```
class(AirPassengers)

## [1] "ts"
```

Wir können den Output analysieren.

```
start(AirPassengers)

## [1] 1949     1

end(AirPassengers)

## [1] 1960    12

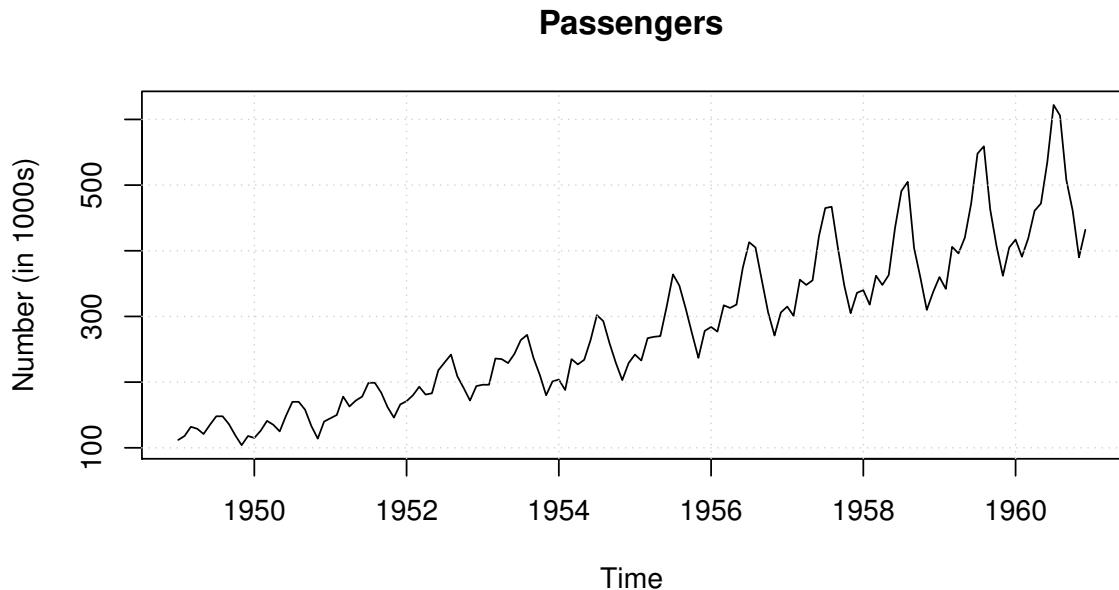
frequency(AirPassengers)

## [1] 12
```

Anhang A. R-Code

Und der Plot:

```
plot(AirPassengers, main = "Passengers", ylab = "Number (in 1000s)")  
grid()
```



Beispiel 7.3.2

(zu Python)

```
X.beer = read.table("../.../Themen/Time_Series_Introduction/Skript_de/Daten/Au  
sep = ";", header = T)  
X.beer.ts = ts(X.beer[, 2], start = c(1956, 1), end = c(1994, 2),  
frequency = 4)  
summary(X.beer.ts)  
  
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.  
##    212.8    325.4   427.4    408.3   466.9    600.0
```

(zu Python)

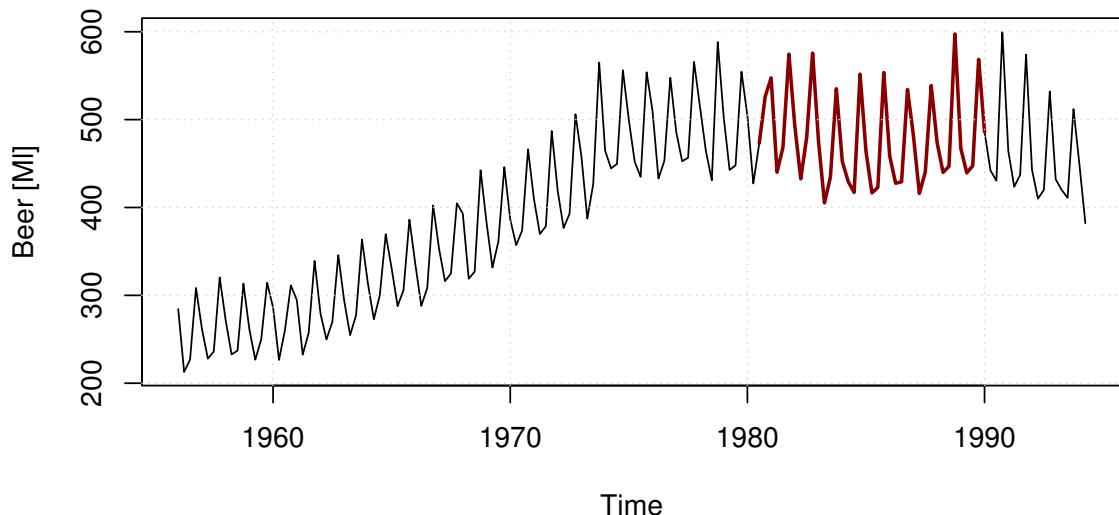
```
plot(X.beer.ts, ylab = "Beer [Ml]", main = "Beer production in Australia")  
X.ts.w = window(X.beer.ts, start = c(1980, 3), end = c(1990, 1))  
summary(X.ts.w)
```

Anhang A. R-Code

```
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##    405.3   437.0   467.2   478.5   530.2   597.5
```

```
lines(X.ts.w, col = "darkred", lwd = 2)
grid()
```

Beer production in Australia

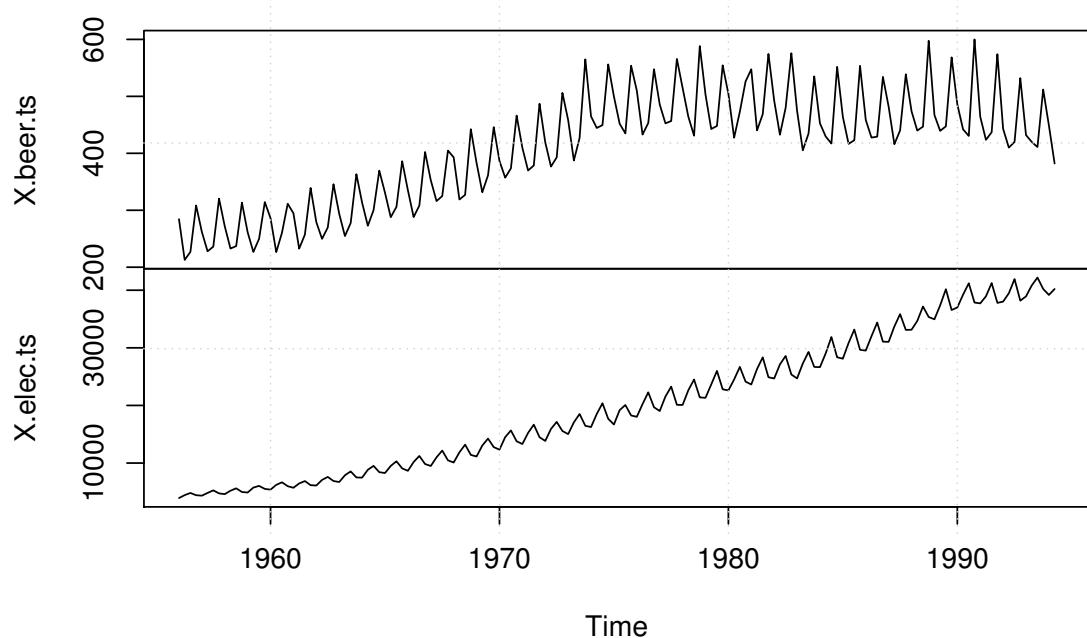


Beispiel 7.3.3

(zu [Python](#))

```
X.elec = read.table("../.../Themen/Time_Series_Introduction/Skript_de/Daten/Au...
  sep = ";", header = T)
X.elec.ts = ts(X.elec[, 2], start = c(1956, 1), end = c(1994, 2),
  frequency = 4)
X.ts = cbind(X.beer.ts, X.elec.ts)
plot(X.ts, main = "Beer and electricity production in Australia")
grid()
```

Beer and electricity production in Australia



Beispiel 7.4.1

(zu [Python](#))

```
# define the Box-Cox transformation
box.cox <- function(x, lambda) {
  if (lambda == 0)
    log(x) else (x^lambda - 1)/lambda
}
# plot the original and the transformed data
layout(matrix(c(1, 2, 3, 4), 2, 2))
plot(AirPassengers, main = "Original", ylab = "", xlab = "")
plot(box.cox(AirPassengers, 2), main = "lambda = 2", ylab = "", xlab = "")
plot(box.cox(AirPassengers, -0.5), main = "lambda = -0.5", ylab = "", xlab = "")
plot(box.cox(AirPassengers, 0), main = "lambda = 0", ylab = "", xlab = "")
```

Beispiel 7.4.2

Anhang A. R-Code

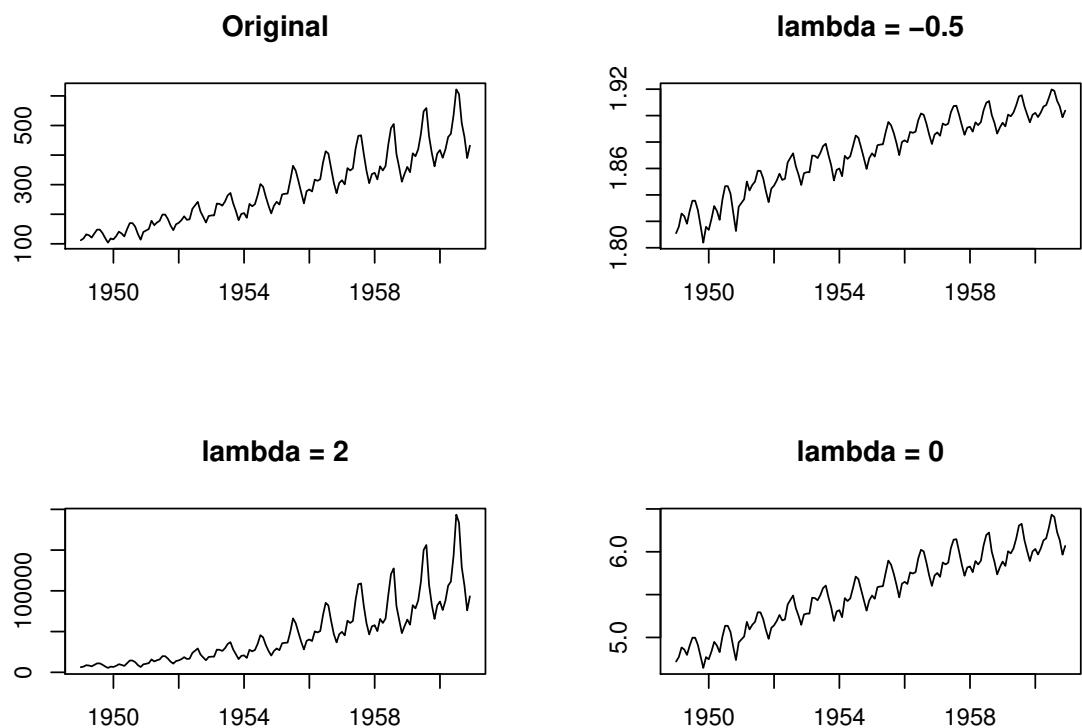


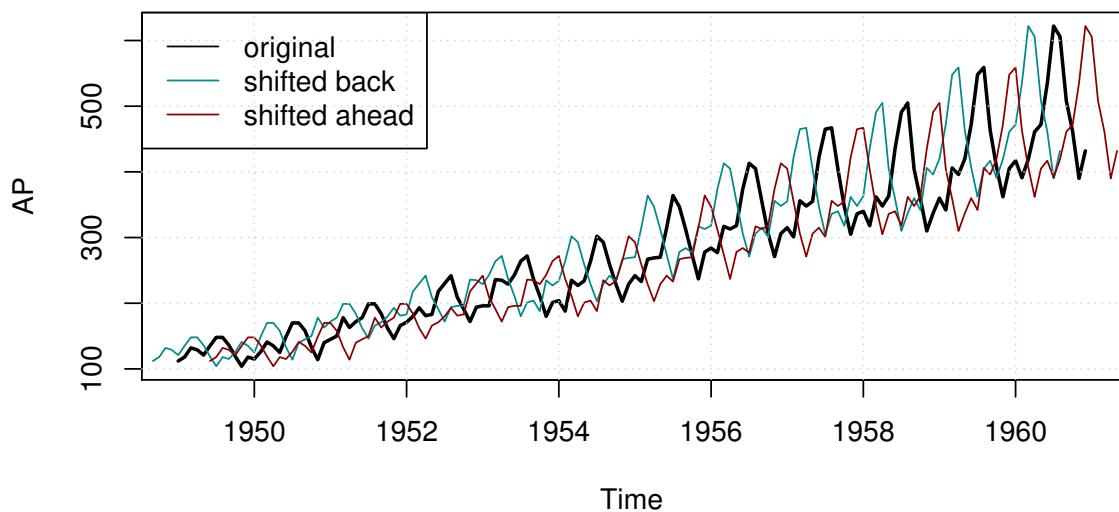
Abbildung A.1.: Box-Cox-transformations for different values of λ .

Anhang A. R-Code

(zu Python)

```
AP = AirPassengers
AP.back = lag(AP, k = 4)
AP.ahead = lag(AP, k = -5)

plot(AP, lwd = 2)
lines(AP.back, col = "darkcyan")
lines(AP.ahead, col = "darkred")
grid()
legend("topleft", legend = c("original", "shifted back", "shifted ahead"),
       lty = 1, col = c("black", "darkcyan", "darkred"))
```



Beispiel 7.4.4

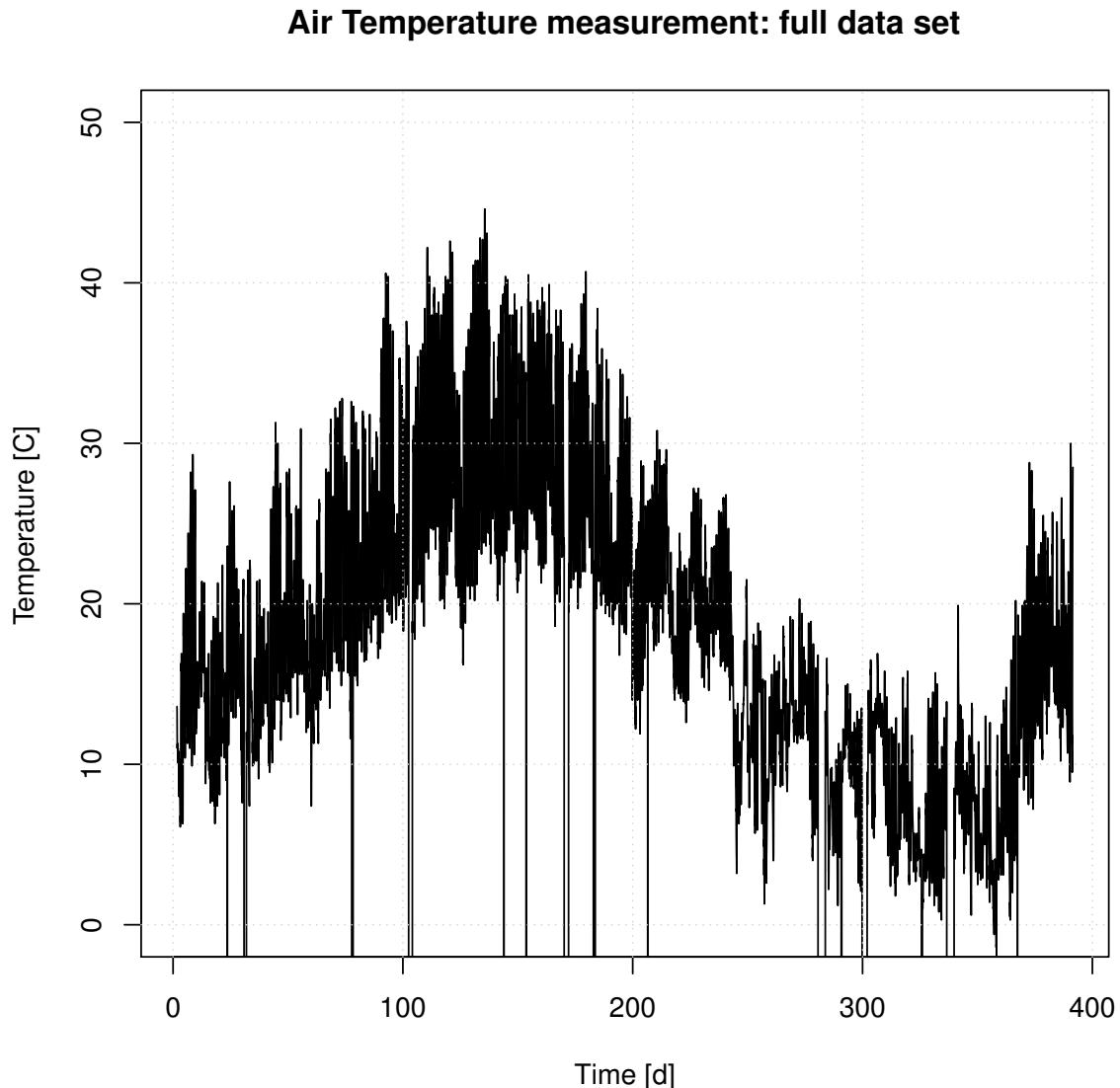
(zu Python)

```
AirData = read.table("../.../Themen/Time_Series_Introduction/Skript_de/Daten/AirPassengerData.txt",
                     sep = ";", header = T, dec = ",")
AirTmp.ts = ts(AirData[, c(13)], start = c(1949, 1), frequency = 12)
end(AirTmp.ts)

## [1] 391 14
```

Anhang A. R-Code

```
plot(AirTmp.ts, main = "Air Temperature measurement: full data set",
      ylab = "Temperature [C]", xlab = "Time [d]", ylim = c(0, 50))
grid()
```



(zu **Python**)

```
AirTmpWin.ts = window(AirTmp.ts, start = c(1, 18), end = c(20, 18))
plot(AirTmpWin.ts, main = "Air Temperature measurement: detail", ylab = "Temperature [C]",
      xlab = "Time [d]", ylim = c(0, 50))
grid()
```

Air Temperature measurement: detail

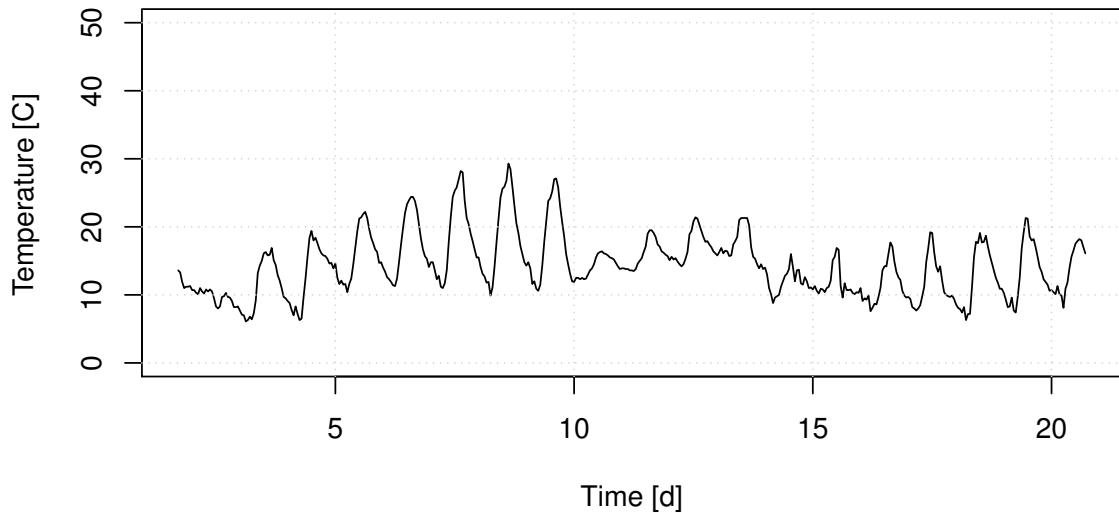


Abbildung A.2.: Hourly air temperatur of 20 consecutive days in march 2014 in an Italian city.

Beispiel 7.4.5

(zu [Python](#))

```
boxplot(AirTmpWin.ts ~ cycle(AirTmpWin.ts), col = "darkcyan", main = "Air temperatur")
```

Beispiel 7.4.6

(zu [Python](#))

```
lag.plot(AirTmpWin.ts, pch = 20, main = "")  
lag.plot(AirTmpWin.ts, pch = 20, main = "", set.lags = 10)
```

Beispiel 7.4.7

(zu [Python](#))

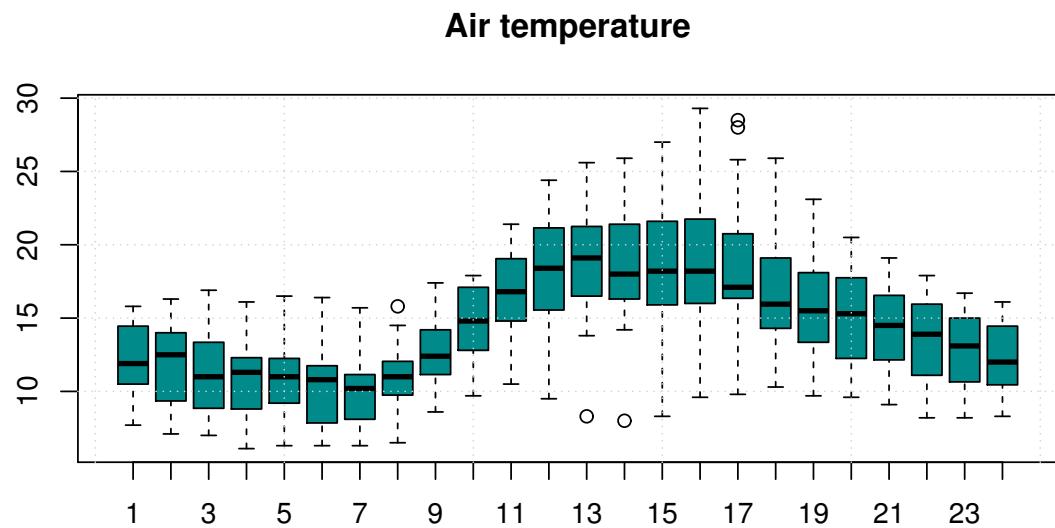


Abbildung A.3.: Grouped boxplot of air temperature data. Each box corresponds to a particular hour of the day.

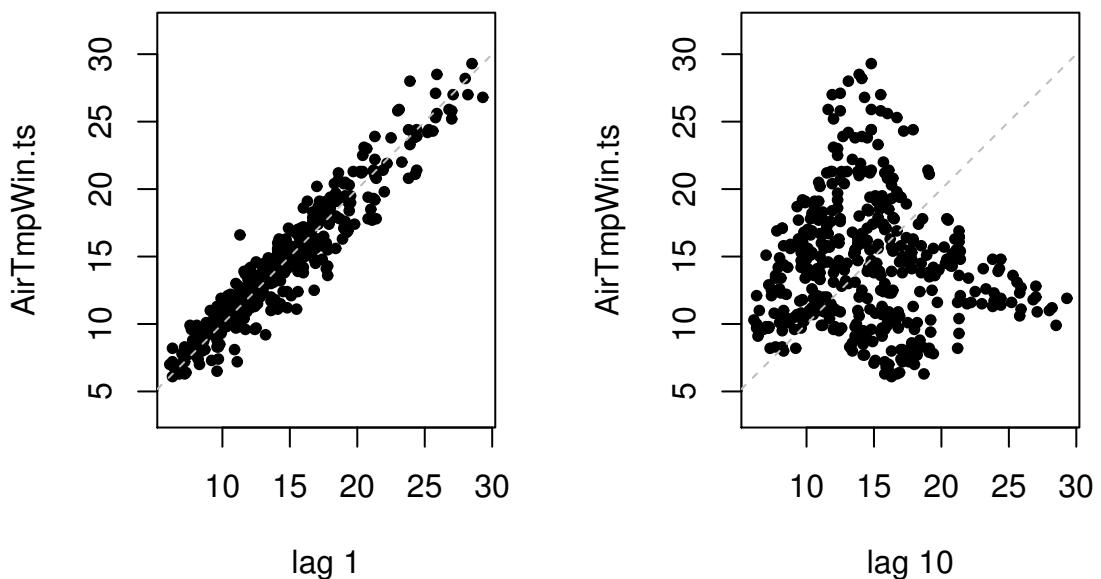
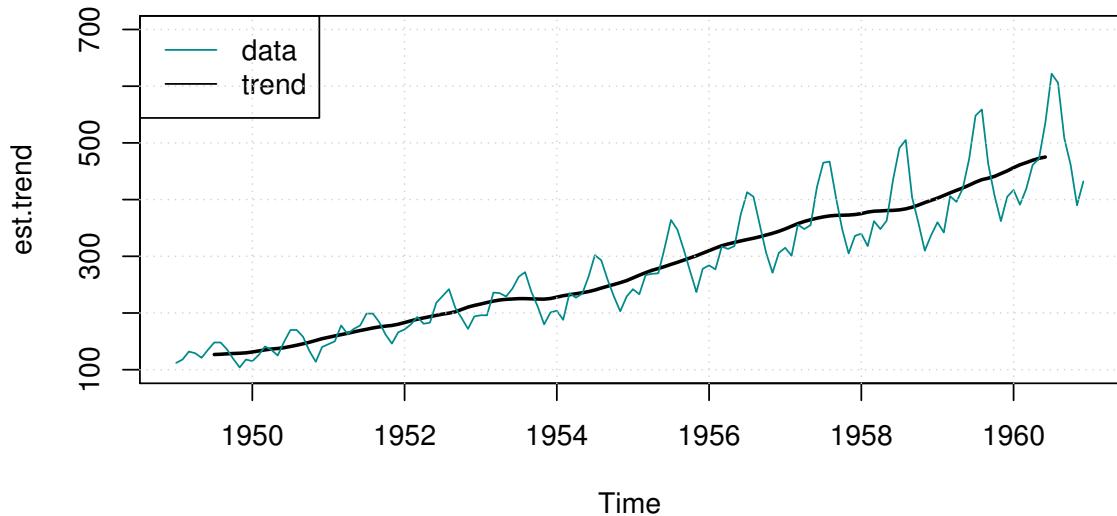


Abbildung A.4.: Lagged scatterplot of the air temperature data. The lag is chosen to be $k = 1$ (left) and $k = 10$ (right).

Anhang A. R-Code

```
weights = c(0.5, rep(1, 11), 0.5)/12
est.trend <- filter(AirPassengers, filter = weights, sides = 2)
plot(est.trend, lwd = 2, ylim = c(100, 700))
lines(AirPassengers, col = "darkcyan")
legend("topleft", legend = c("data", "trend"), lty = 1, col = c("darkcyan",
"black"))
grid()
```

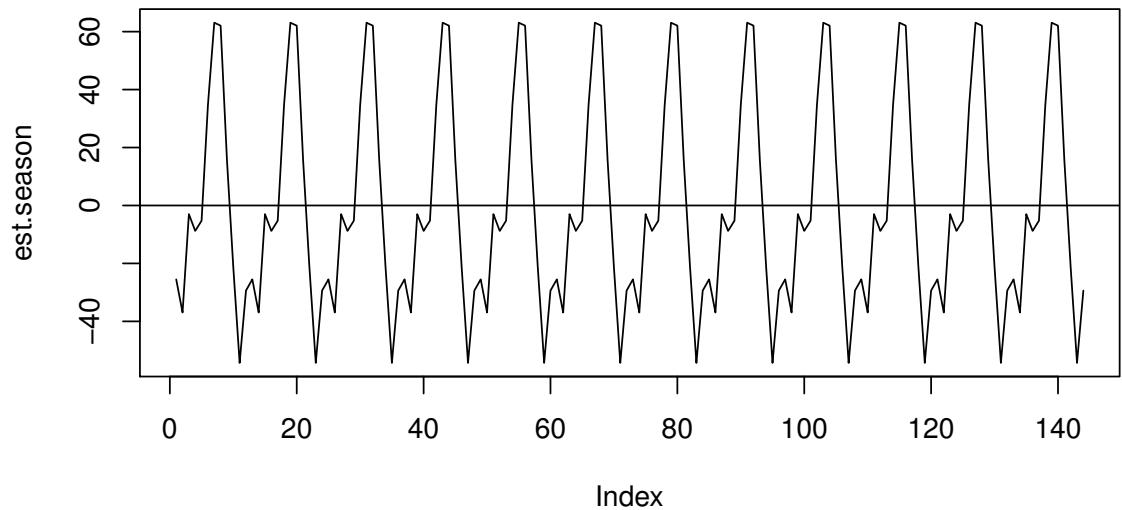


Beispiel 7.4.8

(zu Python)

```
est.season = AirPassengers - est.trend
cyc = factor(cycle(AirPassengers))

est.season.month = tapply(est.season, cyc, mean, na.rm = T)
est.season = est.season.month[cyc]
plot(est.season, type = "l")
abline(h = 0)
```

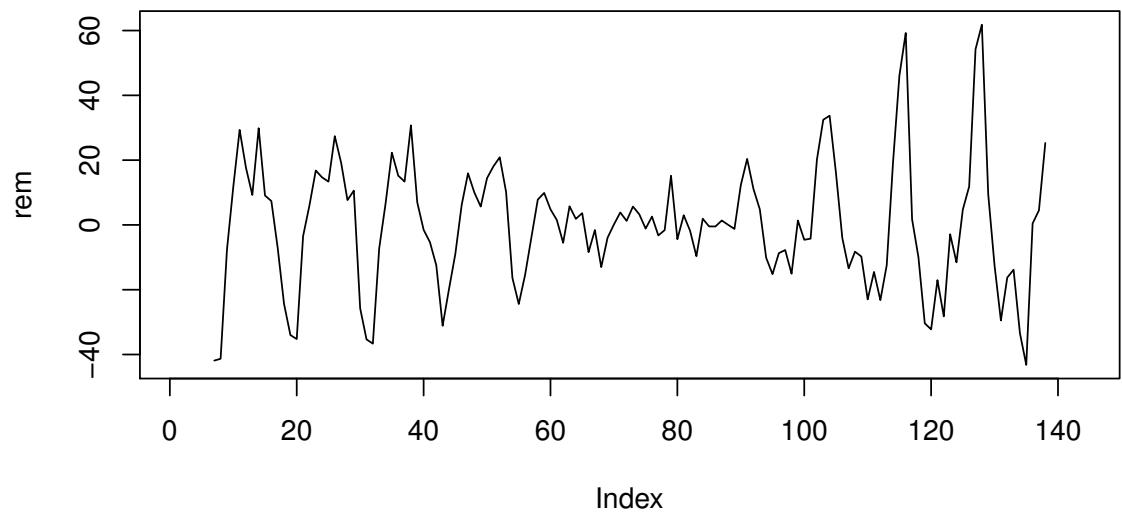


Beispiel 7.4.9

(zu **Python**)

```
est.rem = AirPassengers - est.trend - est.season  
plot(as.vector(est.rem), type = "l", ylab = "rem") #needs fix
```

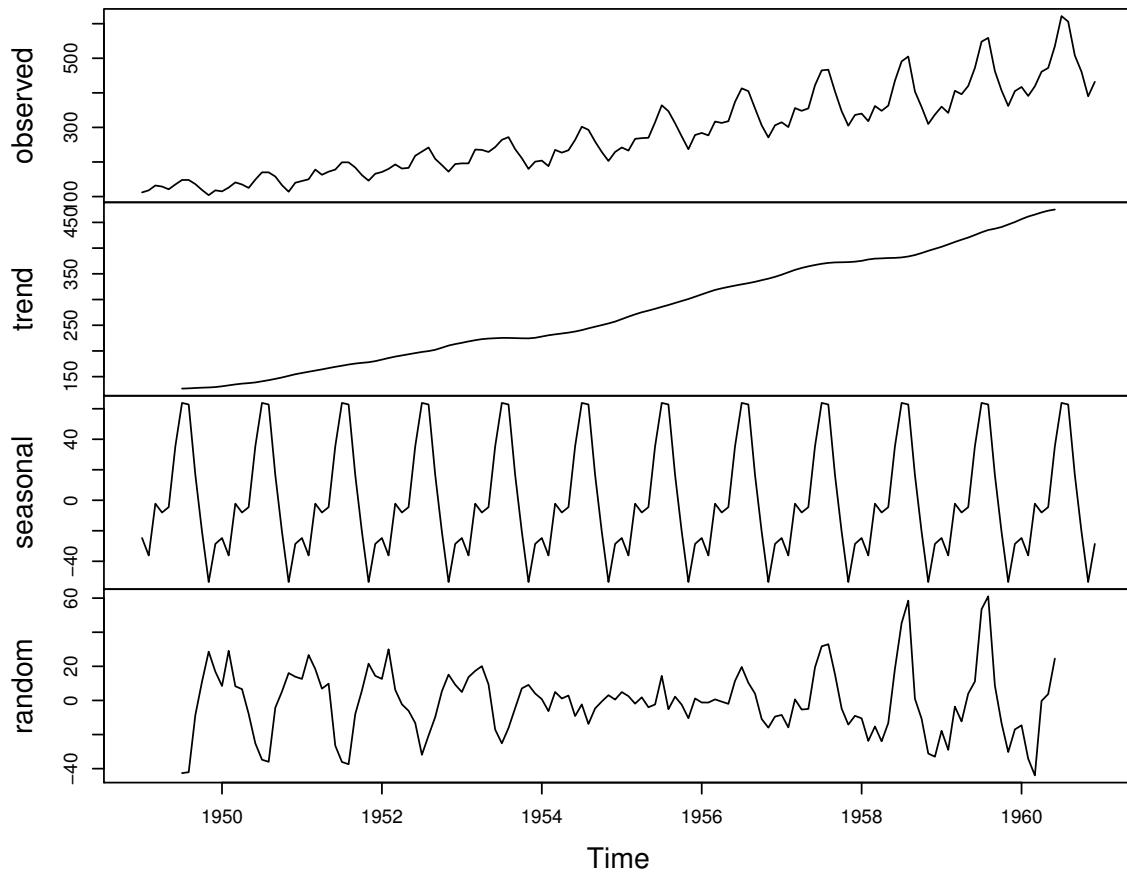
Anhang A. R-Code



(zu Python)

```
decomposed.data = decompose(AirPassengers)  
plot(decomposed.data)
```

Decomposition of additive time series



(zu Python)

```
log.data = log(AirPassengers)

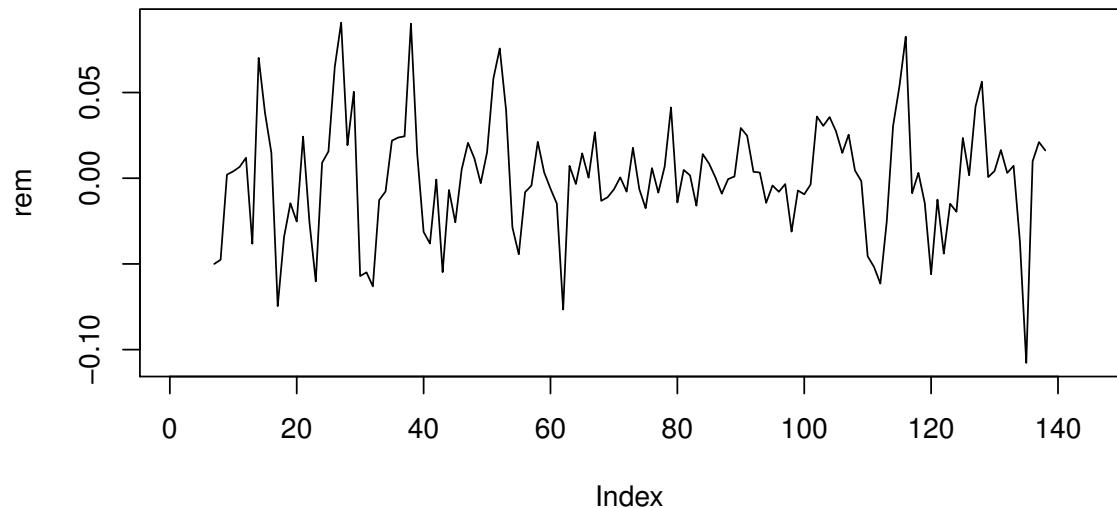
# trend estimation of log data
est.trend.log <- filter(log.data, filter = weights, sides = 2)

# seasonality estimation for log data
est.season.log = log.data - est.trend.log

est.season.month = tapply(est.season.log, cyc, mean, na.rm = T)
est.season.log = est.season.month[cyc]

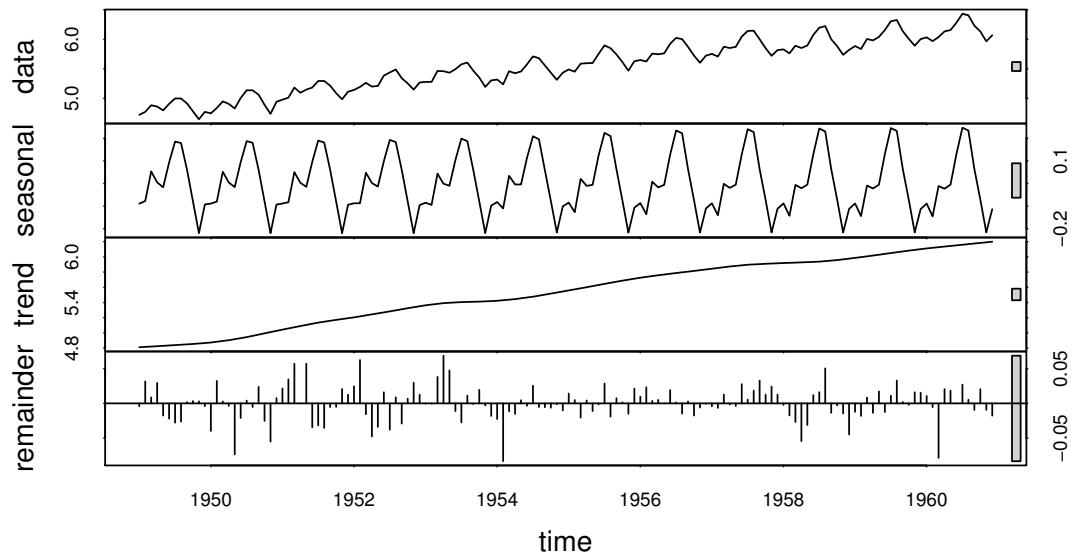
# remainder term estimation for log data
est.rem.log = log.data - est.trend.log - est.season.log
plot(as.vector(est.rem.log), type = "l", ylab = "rem") #needs fix
```

Anhang A. R-Code



(zu Python)

```
stl.fit <- stl(log(AirPassengers), s.window = 10)
plot(stl.fit)
```



Anhang B.

Aus der Normalverteilung hergeleitete Verteilungen

Die **Chi-Quadrat-Verteilung**¹ (χ^2 -Verteilung) ist eine stetige Wahrscheinlichkeitsverteilung über der Menge der positiven reellen Zahlen. Üblicherweise ist mit „Chi-Quadrat-Verteilung“ die zentrale Chi-Quadrat-Verteilung gemeint. Ihr einziger Parameter n muss eine natürliche Zahl sein und wird **Freiheitsgrad** genannt.

Sie ist eine der Verteilungen, die aus der Normalverteilung $\mathcal{N}(\mu, \sigma^2)$ hergeleitet werden kann. Hat man n Zufallsvariablen X_i , die unabhängig und standardnormalverteilt sind, so ist die Chi-Quadrat-Verteilung mit n Freiheitsgraden definiert als die Verteilung der Summe der quadrierten Zufallsvariablen $X_1^2 + \dots + X_n^2$.

B.1. Dichtefunktion der Chi-Quadrat-Verteilung

Chi-Quadrat-Verteilung

Bezeichnen wir mit Z eine standardnormalverteilte Zufallsvariable, dann wird die Verteilung von $X = Z^2$ **Chi-Quadrat-Verteilung mit einem Freiheitsgrad** genannt.

Wir haben also $X = Z^2$, wobei $Z \sim \mathcal{N}(0, 1)$. Somit haben wir

$$\begin{aligned} F_X(x) &= P(X \leq x) \\ &= P(Z^2 \leq x) \\ &= P(-\sqrt{x} \leq Z \leq \sqrt{x}) \\ &= \Phi(\sqrt{x}) - \Phi(-\sqrt{x}). \end{aligned}$$

¹Zu Risiken und Nebenwirkungen lesen Sie diesen Anhang mit Gelassenheit und fragen Sie Ihren Dozenten oder Tutoren. Im Falle eines Falles: Zu den riesigen Nebenwirkungen fressen Sie den Anhang und erschlagen Sie den irren Dozenten Ihres Tutoren.

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

Wir finden die Dichtefunktion von X , indem wir die Kumulative Verteilungsfunktion $F_X(x)$ nach x ableiten. Da $\Phi'(x) = \varphi(x)$, ergibt sich mit der Kettenregel

$$\begin{aligned} f_X(x) &= \frac{1}{2}x^{-1/2}\varphi(\sqrt{x}) + \frac{1}{2}\varphi(-\sqrt{x}) \\ &= x^{-1/2}\varphi(\sqrt{x}), \end{aligned}$$

wobei wir im letzten Schritt verwendet haben, dass φ symmetrisch bezüglich der y -Achse ist. Dies ergibt dann

$$f_X(x) = \frac{x^{-1/2}}{\sqrt{2\pi}}e^{-x/2}.$$

Wir bezeichnen diese Dichtefunktion auch als $f_{\chi_1^2}$, also der Chi-Quadrat-Verteilung mit einem Freiheitsgrad.

Wir stellen fest, dass wenn $X \sim \mathcal{N}(\mu, \sigma^2)$, dann ist $(X - \mu)/\sigma \sim \mathcal{N}(0, 1)$ und somit

$$[(X - \mu)/\sigma]^2 \sim \chi_1^2.$$

Wir stellen fest, dass die Chi-Quadrat-Verteilung mit einem Freiheitsgrad ein Spezialfall der Gammaverteilung ist.

Gammaverteilung

Die **Gammaverteilung** ist eine kontinuierliche Wahrscheinlichkeitsverteilung über der Menge der positiven reellen Zahlen. Sie ist eine direkte Verallgemeinerung der Exponentialverteilung und hat folgende Dichtefunktion

$$f(x) = \begin{cases} \frac{\lambda^\alpha}{\Gamma(\alpha)}x^{\alpha-1}e^{-\lambda x} & x > 0 \\ 0 & x \leq 0. \end{cases}$$

wobei der Ausdruck $\Gamma(\alpha)$ für den Funktionswert der Gammafunktion steht, nach der die Verteilung auch benannt ist. Die Gammafunktion ist für positive reelle Zahlen α über das Integral

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t} dt$$

definiert. Die Werte von $\Gamma(\alpha)$ kann man auch berechnen mit $\Gamma(\frac{1}{2}) = \sqrt{\pi}$, $\Gamma(1) = 1$, $\Gamma(\alpha + 1) = \alpha \cdot \Gamma(\alpha)$ mit $\alpha \in \mathbb{R}^+$.

Die Chi-Quadrat-Verteilung mit einem Freiheitsgrad ist also ein Spezialfall der Gammaverteilung, wobei $\alpha = 1/2$ und $\lambda = 1/2$.

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

Die **Momentengenerierende Funktion** $M(t)$ der Gammaverteilung ist gegeben durch:

$$\begin{aligned} M(t) &\equiv E[e^{tx}] \\ &= \int_{-\infty}^{\infty} e^{tx} f(x) dx \\ &= \int_0^{\infty} e^{tx} \frac{\lambda^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\lambda x} dx \\ &= \left(\frac{\lambda}{\lambda - t} \right)^\alpha. \end{aligned}$$

Mit der Momentengenerierenden Funktion lassen sich Erwartungswert und Varianz einer Wahrscheinlichkeitsverteilung sehr elegant berechnen:

$$M'(t = 0) = E[X] = \frac{\alpha}{\lambda}$$

und

$$M''(t = 0) = E[X^2] = \frac{\alpha(\alpha + 1)}{\lambda^2},$$

woraus sich für die Varianz folgendes Resultat ergibt

$$\begin{aligned} \text{Var}[X] &= E[X^2] - (E[X])^2 \\ &= \frac{\alpha(\alpha + 1)}{\lambda^2} - \frac{\alpha^2}{\lambda^2} \\ &= \frac{\alpha}{\lambda^2}. \end{aligned}$$

Über den kleinen Umweg der Gammaverteilung und deren Momentengenerierenden Funktion lässt sich nun die Dichte der Chi-Quadrat-Verteilung mit n Freiheitsgraden herleiten.

χ^2 -Quadrat-Verteilung mit n Freiheitsgraden

Die **χ^2 -Quadrat-Verteilung mit n Freiheitsgraden** ist definiert als die Summe aus n stochastisch unabhängiger quadrierter standardnormalverteilter Zufallsvariablen

$$X_1^2 + \cdots + X_n^2,$$

mit $X_i \sim \mathcal{N}(0, 1)$ für $i = 1, \dots, n$ und wir mit χ_n^2 bezeichnen. Die Summe „quadrierter“ Größen kann keine negativen Werte annehmen.

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

Die Dichte der Zufallsvariable

$$V = X_1^2 + \cdots + X_n^2,$$

mit X_1, \dots, X_n unabhängig und standardnormalverteilt, ergibt sich aus der Momentengenerierenden Funktion

$$\begin{aligned} M(t) &= E[e^{tV}] \\ &= E[e^{tX_1^2} \cdot e^{tX_2^2} \cdots e^{tX_n^2}] \\ &= E[e^{tX_1^2}] \cdot E[e^{tX_2^2}] \cdots E[e^{tX_n^2}], \end{aligned}$$

wobei wir im letzten Schritt verwendet haben, dass alle X_i stochastisch unabhängig sind. Nun haben wir aber bereits $E[e^{tX_i^2}]$ berechnet, und zwar ist dies nichts anderes als die Momentengenerierende Funktion der Chi-Quadrat-Verteilung mit einem Freiheitsgrad:

$$\left(\frac{\lambda}{\lambda-t}\right)^{\alpha}.$$

Somit finden wir für die Momentengenerierende Funktion der Chi-Quadrat-Verteilung

$$\begin{aligned} M(t) &= \left(\frac{\lambda}{\lambda-t}\right)^{n \cdot \alpha} \Big|_{\alpha=\frac{1}{2}, \lambda=\frac{1}{2}} \\ &= (1-2t)^{-n/2} \end{aligned}$$

Dies ist aber nun nichts anderes als die Momentengenerierende Funktion der Gamma-Verteilung mit den Parameterwerten $\alpha = n/2$ und $\lambda = 1/2$.

χ_n^2 -Verteilung mit n Freiheitsgraden

Die Dichte der χ_n^2 -Verteilung mit n Freiheitsgraden ist also gegeben durch:

$$f(v) = \begin{cases} \frac{v^{\frac{n}{2}-1} e^{-\frac{v}{2}}}{2^{\frac{n}{2}} \Gamma(\frac{n}{2})} & v > 0 \\ 0 & v \leq 0 \end{cases}$$

Der Erwartungswert der Chi-Quadrat-Verteilung mit n Freiheitsgraden ergibt sich nun direkt aus der Momentengenerierenden Funktion

$$\begin{aligned} M'(t=0) &= E[V] \\ &= -n/2 \cdot (1-2t)^{-n/2-1} \cdot (-2)|_{t=0} \\ &= n. \end{aligned}$$

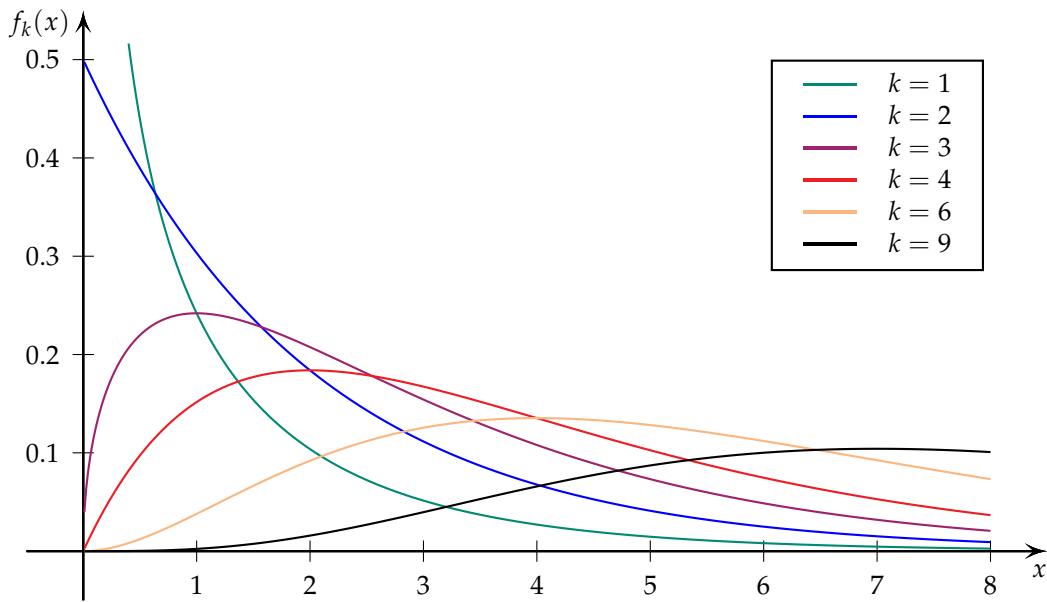


Abbildung B.1.: Densities of the chi-squared distribution with different degrees of freedom k .

Eine ähnliche Rechnung ergibt für die Varianz

$$\text{Var}[V] = 2n .$$

B.2. t -Verteilung

t -Verteilung

Es seien die Zufallsvariablen Z standardnormalverteilt, also $Z \sim \mathcal{N}(0, 1)$, und U χ_n^2 -verteilt, also $U \sim \chi_n^2$. Falls Z und U stochastisch unabhängig sind, dann folgt die Zufallsvariable

$$\frac{Z}{\sqrt{U/n}}$$

der t -Verteilung mit n Freiheitsgraden.

Die Wahrscheinlichkeitsdichte der t -Verteilung lässt sich herleiten aus der gemeinsamen Dichte der beiden unabhängigen Zufallsvariablen Z und χ_n^2 , die standardnormal, beziehungsweise χ_n^2 -verteilt sind:

$$f_{Z,\chi_n^2}(z, y) = \frac{e^{-\frac{1}{2}z^2}}{\sqrt{2\pi}} \cdot \frac{y^{\frac{n}{2}-1} e^{-\frac{1}{2}y}}{2^{\frac{n}{2}} \Gamma(\frac{n}{2})} .$$

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

Mit der Transformation

$$t = z / \sqrt{y/n}, \quad v = y,$$

bekommt man die gemeinsame Dichte von $T = Z / \sqrt{\chi_n^2/n}$ und χ_n^2 , wobei $-\infty < t < \infty$ und $0 \leq v < \infty$.

Die Jacobideterminante dieser Transformation ist:

$$\det \frac{\partial(z, y)}{\partial(t, v)} = \begin{vmatrix} \sqrt{\frac{v}{n}} & 0 \\ \diamond & 1 \end{vmatrix} = \sqrt{\frac{v}{n}}.$$

Der Wert \diamond ist unwichtig, weil er bei der Berechnung der Determinante mit 0 multipliziert wird. Die neue Dichtefunktion schreibt sich also

$$f_{T, \chi_n^2}(t, v) = \frac{e^{-\frac{1}{2}v\frac{t^2}{n}}}{\sqrt{2\pi}} \cdot \frac{1}{2^{\frac{n}{2}}\Gamma(\frac{n}{2})} v^{\frac{n}{2}-1} e^{-\frac{1}{2}v} \cdot \sqrt{\frac{v}{n}}.$$

Gesucht ist nun die Randverteilung $f(t)$ als Integral über die nicht interessierende Variable v :

$$\begin{aligned} f(t) &= \int_0^\infty f_{T, \chi_n^2}(t, v) dv \\ &= \frac{1}{\sqrt{n\pi} 2^{(n+1)/2} \Gamma(n/2)} \int_0^\infty v^{(n-1)/2} e^{-v(1+t^2/n)/2} dv \\ &= \frac{\Gamma\left(\frac{n+1}{2}\right)}{\sqrt{n\pi} \Gamma\left(\frac{n}{2}\right)} \left(1 + \frac{t^2}{n}\right)^{-\frac{n+1}{2}}. \end{aligned}$$

t-Verteilung mit n Freiheitsgraden

Die Dichtefunktion der **t-Verteilung mit n Freiheitsgraden** ist gegeben durch

$$f(t) = \frac{\Gamma\left(\frac{n+1}{2}\right)}{\sqrt{n\pi} \Gamma\left(\frac{n}{2}\right)} \left(1 + \frac{t^2}{n}\right)^{-\frac{n+1}{2}}.$$

Wenn die unabhängigen Zufallsvariablen X_1, X_2, \dots, X_n identisch normalverteilt sind mit Erwartungswert μ und Standardabweichung σ_X , dann sind der Stichprobenmittelwert gegeben durch

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$$

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

und die Stichprobenvarianz durch

$$\hat{\sigma}_X = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2.$$

Es kann gezeigt werden, dass \bar{X}_n und $\hat{\sigma}_X$ stochastisch unabhängig sind.

Die Zufallsgrösse

$$\frac{\bar{X}_n - \mu}{\sigma_X / \sqrt{n}}$$

folgt einer Standardnormalverteilung. Nun möchten wir aber wissen, welche Verteilung die Grösse

$$\frac{\bar{X}_n - \mu}{\hat{\sigma}_X / \sqrt{n}}$$

hat. Dazu schreiben wir als erstes die obige Grösse etwas um:

$$\begin{aligned} \frac{\bar{X}_n - \mu}{\hat{\sigma}_X / \sqrt{n}} &= \frac{\bar{X}_n - \mu}{\hat{\sigma}_X / \sqrt{n}} \cdot \frac{\sigma_X}{\sigma_X} \\ &= \frac{\bar{X}_n - \mu}{\sigma / \sqrt{n}} \cdot \frac{\sigma_X}{\hat{\sigma}_X} \\ &= \frac{\bar{X}_n - \mu}{\sigma_X / \sqrt{n}} \Big/ \left(\frac{\hat{\sigma}_X}{\sigma_X} \right). \end{aligned}$$

Nun stellt sich die Frage, wie $\frac{\hat{\sigma}_X}{\sigma_X}$ verteilt ist. Als erstes stellen wir fest, dass

$$\begin{aligned} (n-1) \cdot \left(\frac{\hat{\sigma}_X}{\sigma_X} \right)^2 &= (n-1) \cdot \frac{1}{\sigma_X^2} \cdot \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2 \\ &= \frac{1}{\sigma_X^2} \cdot \sum_{i=1}^n (X_i - \bar{X}_n)^2. \end{aligned}$$

Nun wissen wir, dass

$$\frac{1}{\sigma_X^2} \sum_{i=1}^n (X_i - \mu)^2 = \sum_{i=1}^n \left(\frac{X_i - \mu}{\sigma} \right)^2 \sim \chi_n^2.$$

Dieser Ausdruck lässt sich aber umformen in

$$\frac{1}{\sigma_X^2} \sum_{i=1}^n (X_i - \mu)^2 = \frac{1}{\sigma_X^2} \sum_{i=1}^n [(X_i - \bar{X}_n) + (\bar{X}_n - \mu)]^2.$$

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

Multiplizieren wir den quadratischen Term aus und benutzen, dass $\sum_{i=1}^n (X_i - \bar{X}_n)^2 = 0$, so erhalten wir

$$\underbrace{\frac{1}{\sigma_X^2} \sum_{i=1}^n (X_i - \mu)^2}_W = \underbrace{\frac{1}{\sigma_X^2} \sum_{i=1}^n (X_i - \bar{X}_n)^2}_U + \underbrace{\left(\frac{\bar{X}_n - \mu}{\sigma_X / \sqrt{n}} \right)}_V^2$$

Dies ist eine Beziehung von der Form $W = U + V$, wobei U und V stochastisch unabhängig (was etwas schwieriger zu zeigen ist). Somit gilt für die Momentengenerierenden Funktionen

$$M_W(t) = \frac{M_W(t)}{M_V(t)}.$$

Da sowohl W wie auch V χ_n^2 -verteilt sind, gilt

$$\begin{aligned} M_U(t) &= \frac{M_W(t)}{M_V(t)} \\ &= \frac{(1 - 2t)^{-n/2}}{(1 - 2t)^{-1/2}} \\ &= (1 - 2t)^{-(n-1)/2}. \end{aligned}$$

Also folgt U , respektive der Ausdruck $\frac{1}{\sigma_X^2} \sum_{i=1}^n (X_i - \bar{X}_n)^2$ einer χ_{n-1}^2 -Verteilung. $\left(\frac{\hat{\sigma}_X}{\sigma_X}\right)^2$ ist also eine χ_{n-1}^2 -verteilte Zufallsvariable dividiert durch die Anzahl Freiheitsgrade. Daraus schliessen wir, dass

$$\frac{\bar{X}_n - \mu}{\hat{\sigma}_X / \sqrt{n}}$$

wie eine Zufallsvariable $Z / \sqrt{U/(n-1)}$ verteilt ist, wobei $Z \sim \mathcal{N}(0, 1)$ und $U \sim \chi_{n-1}^2$, somit folgt $\frac{\bar{X}_n - \mu}{\hat{\sigma}_X / \sqrt{n}}$ einer t_{n-1} -Verteilung.

B.3. Chi-Quadrat Test

Beim Chi-Quadrat Test wird eine Hypothese über die unbekannte Verteilung der Zufallsvariablen X geprüft, woraus sich der Name Anpassungstest, Verteilungstest bzw. Goodness-of-fit-Test ergibt.

Wir fassen die n unabhängigen Beobachtungen x_1, \dots, x_n als Realisationen der Zufallsvariablen X auf. Es muss nun eine Intervallbildung der beobachteten Werte in disjunkte, aneinander angrenzende Klassen erfolgen. Mit k als Anzahl der Klassen ($k \geq 2$) können die Klassen allgemein wie folgt geschrieben werden:

$$(x_0^*, x_1^*), (x_1^*, x_2^*), \dots, (x_{k-1}^*, x_k^*)$$

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

bzw. (x_{j-1}^*, x_j^*) für $j = 1, \dots, k$. Es bezeichne $h(x_{j-1}^* < X \leq x_j^*) = h_j$ die beobachtete absolute Häufigkeit in der j -ten Klasse der Stichprobe für $j = 1, \dots, k$. Die Intervallbildung sollte so erfolgen, dass $h_j \geq 5$ für alle $j = 1, \dots, k$.

Wir bezeichnen mit p_j für $j = 1, \dots, k$ die Wahrscheinlichkeit, dass die Zufallsvariable X in die j -te Klasse (x_{j-1}^*, x_j^*) fällt, wenn ihr die hypothetische Verteilung $F_0(x)$ zugrunde liegt, d.h. wenn die Nullhypothese H_0 gilt:

$$p_j = P(x_{j-1}^* < X \leq x_j^* | H_0) .$$

Die Tatsache, dass die beobachteten absoluten Häufigkeiten Zufallsvariablen H_j sind, lässt sich wie folgt zeigen. Aus der Grundgesamtheit wird ein Element zufällig gezogen und festgestellt, ob das Ereignis $\{X = x_j \mid x_j \in (x_{j-1}^*, x_j^*)\}$ eingetreten ist oder nicht. Es gibt somit nur zwei mögliche Ergebnisse des Zufallsexperimentes. Die Wahrscheinlichkeit für das Eintreten des Ereignisses $\{X = x_j \mid x_j \in (x_{j-1}^*, x_j^*)\}$ beträgt bei Gültigkeit der Nullhypothese p_j und die Wahrscheinlichkeit für das Nichteintreten $1 - p_j$. Es liegt somit ein Bernoulli-Experiment vor. Das Zufallsexperiment wird n mal wiederholt, wobei die einzelnen Versuche unabhängig voneinander sind. Bei n -maliger Durchführung der Versuche interessiert die Gesamtzahl des Eintretens von $\{X = x_j \mid x_j \in (x_{j-1}^*, x_j^*)\}$, d.h. die absolute Häufigkeit H_j in der Stichprobe. Diese Häufigkeit kann von Stichprobe zu Stichprobe unterschiedlich sein, so dass H_j eine diskrete Zufallsvariable ist, die die Werte $0, \dots, n$ annehmen kann. Die Zufallsvariable H_j ist binomialverteilt und zwar bei Gültigkeit der Nullhypothese H_0 mit den Parametern n und p_j : $H_j \sim \text{Bin}(n, p_j)$. Der Erwartungswert von H_j ist $E(H_j) = n \cdot p_j$ und damit die bei Gültigkeit der Nullhypothese erwartete Häufigkeit des Ereignisses $\{X = x_j \mid x_j \in (x_{j-1}^*, x_j^*)\}$ in der Stichprobe. Die Variation der absoluten Häufigkeiten für $\{X = x_j \mid x_j \in (x_{j-1}^*, x_j^*)\}$ wird durch die Varianz $\text{Var}(H_j) = np_j(1 - p_j)$ erfasst.

Für die Konstruktion der Teststatistik wird die Abweichung der Zufallsvariable von ihrem Erwartungswert gebildet:

$$H_j - n \cdot p_j .$$

Zur Vermeidung, dass sich positive und negative Abweichungen aufheben, erfolgt eine Quadrierung: $(H_j - n \cdot p_j)^2$. Mit der Division durch die erwartete Häufigkeit $n \cdot p_j$ wird der Einfluss des Stichprobenumfangs und der Wahrscheinlichkeit p_j berücksichtigt und der unterschiedlichen Bedeutung der Abweichungen Rechnung getragen. Eine Differenz $h_j - n \cdot p_j = 5$ fällt bei $n \cdot p_j = 10$ stärker ins Gewicht als bei $n \cdot p_j = 100$. Weiter beobachten wir, dass für kleine p_j und n genügend gross gilt, dass $\text{Var}(X) \approx n \cdot p_j$ und somit

$$\frac{(H_j - n \cdot p_j)}{\sqrt{n \cdot p_j}} \sim \mathcal{N}(0, 1) .$$

Betrachten wir folgende Teststatistik

$$V = \sum_{j=1}^k \frac{(H_j - n \cdot p_j)^2}{n \cdot p_j}$$

so ist auch V eine Zufallsvariable, da die H_j Zufallsvariablen sind. Bei Gültigkeit der Nullhypothese, hinreichend grossem Stichprobenumfang n und Einhaltung der Approximationsbedingungen ($n \cdot p_j \geq 5$) ist die Teststatistik V approximativ χ^2 -verteilt mit $f = k - m - 1$ Freiheitsgraden. Dies gilt unabhängig davon, welche Verteilung unter H_0 angenommen wurde. Sind die Approximationsbedingungen nicht erfüllt, müssen vor der Anwendung des Tests benachbarte Werte bzw. Klassen zusammengefasst werden.

Bei der Ermittlung der Freiheitsgrade ist zu berücksichtigen, dass ein Freiheitsgrad grundsätzlich verloren geht, weil die beobachteten absoluten Häufigkeiten nicht unabhängig voneinander sind. Für vorgegebenen Stichprobenumfang n und aufgrund der Bedingung $\sum_j h_j = n$ folgt, dass jede Häufigkeit h_j durch die anderen $k - 1$ Häufigkeiten bestimmt ist. Weitere Freiheitsgrade gehen verloren, wenn die hypothetische Verteilung $F_0(x)$ nicht mit allen ihren Parametern bekannt ist, sondern diese Parameter aus der Stichprobe geschätzt werden müssen. Mit m als Anzahl der zu schätzenden Parameter ergibt sich die Anzahl der Freiheitsgrade zu: $f = k - m - 1$. Wurden keine Parameter geschätzt, so ist die Anzahl Freiheitsgrade gegeben durch die Anzahl Klassen minus 1.

Bei einem Signifikanzniveau α wird H_0 abgelehnt, wenn $V > \chi^2_{(1-\alpha;k-m-1)}$ gilt, wenn also der aus der Stichprobe erhaltene Wert der Prüfgrösse grösser als das $(1 - \alpha)$ -Quantil der χ^2 -Verteilung mit $k - m - 1$ Freiheitsgraden ist.

Beispiel B.3.1

In einem berühmten Artikel, den Rutherford, Chadwick und Ellis 1920 publiziert hatten, befindet sich die Tabelle B.1 mit Messwerten.

Es wurde die Anzahl der von einer radioaktiven Substanz emittierten Teilchen in einem Zeitintervall von 7.5 Sekunden gemessen. Im gesamten wurden die Anzahlen Zerfälle in 2608 Zeitintervallen beobachtet. Nun wurde postuliert, dass die Anzahl emittierter Teilchen pro Zeitintervall eine Zufallsvariable mit Poisson-Verteilung ist. Um diese Vermutung zu überprüfen, führen wir einen Chi-Quadrat-Test durch.

In diesem Fall, ist der Erwartungswert der Poisson-Verteilung unbekannt. Wir schätzen den Erwartungswert, indem wir den empirischen Mittelwert zu 3.870 ermitteln. Für D^2 mit $12 - 1 - 1 = 10$ Freiheitsgraden ergibt sich dann aus dem Datensatz der Wert 12.94. Der Verwerfungsbereich auf dem 1% Signifikanzniveau beginnt bei 23.2. Der von uns ermittelte Wert überschreitet diese Schwelle nicht, so dass wir daraus schliessen können, dass die Anzahl emittierter Teilchen pro Zeitintervall in guter Übereinstimmung mit der Poisson-Verteilung ist.

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

Anzahl Zerfälle	Beobachtet	Erwartet	$(O - E)^2 / E$	
0	57	54.40	0.12	
1	203	210.50	0.27	
2	383	407.40	1.46	
3	525	525.50	0	
4	532	508.40	1.10	
5	408	393.50	0.053	
6	273	253.80	1.45	
7	139	140.30	0.01	
8	45	67.80	7.67	
9	27	29.20	0.17	
10	10	11.30	0.15	
> 11	6	5.80	0.01	

Table B.1.: Number of decays within 7.5 seconds and number of experiments (out of 2608 total) in which the corresponding number of decays was observed.

Mit **Python** ermittelt sich das 99 %-Quantil mit

```
from scipy.stats import chi2
chi2.ppf(0.99, df=10)

## 23.209251158954356
```

der P-Wert wie folgt

```
from scipy.stats import chi2
1-chi2.cdf(12.93714, df=10)

## 0.22720766204623843
```

Der Chi-Quadrat Test kann auch folgendermassen in **Python** ausgeführt werden

```
from pandas import DataFrame
from scipy.stats import chisquare
zerf = DataFrame({
    "beobachtet" : ([57, 203, 383, 525, 532, 408,
```

Anhang B. Aus der Normalverteilung hergeleitete Verteilungen

```
    273, 139, 45, 27, 10, 6]),  
  "erwartet" : ([54.40, 210.50, 407.40, 525.50,  
 508.40, 393.50, 253.80, 140.30,  
 67.80, 29.20, 11.30, 5.80])  
}  
chisquare(zerf["beobachtet"], zerf["erwartet"])  
  
## Power_divergenceResult(statistic=12.937140958761361, pvalue=0.297453706585465
```

□

Anhang C.

Ergänzungen und Herleitung von wichtigen Beziehungen in der Varianzanalyse

Die *Varianzanalyse* beruht auf folgender Identität:

$$\sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{..})^2 = \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2 + \sum_{i=1}^g \sum_{j=1}^m (\bar{Y}_{i\bullet} - \bar{Y}_{..})^2$$

Um diese Identität herzuleiten, drücken wir die linke Seite der Gleichung wie folgt aus

$$\begin{aligned} \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{..})^2 &= \sum_{i=1}^g \sum_{j=1}^m [(Y_{ij} - \bar{Y}_{i\bullet}) + (\bar{Y}_{i\bullet} - \bar{Y}_{..})]^2 \\ &= \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2 + \sum_{i=1}^g \sum_{j=1}^m (\bar{Y}_{i\bullet} - \bar{Y}_{..})^2 \\ &\quad + 2 \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})(\bar{Y}_{i\bullet} - \bar{Y}_{..}) \\ &= \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2 + \sum_{i=1}^g \sum_{j=1}^m (\bar{Y}_{i\bullet} - \bar{Y}_{..})^2 \\ &\quad + 2 \sum_{i=1}^g \left[(\bar{Y}_{i\bullet} - \bar{Y}_{..}) \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet}) \right] \end{aligned}$$

Der letzte Term im letzten Ausdruck verschwindet, da die Summe von Abweichungen vom Mittelwert null ergibt.

Anhang C. Ergänzungen und Herleitung von wichtigen Beziehungen in der Varianzanalyse

Die grundlegende Idee, die der Varianzanalyse zugrundliegt, ist der Vergleich der Größen von folgenden Quadratsummen

$$E = \sum_{i=1}^g \sum_{j=1}^m (Y_{ij} - \bar{Y}_{i\bullet})^2$$

$$G = \sum_{i=1}^g \sum_{j=1}^m (\bar{Y}_{i\bullet} - \bar{Y}_{..})^2$$

Im Folgenden werden wir die Erwartungswerte dieser Quadratsummen ermitteln. Dazu benötigen wir folgenden Zusammenhang : Seien X_i , wobei $i = 1, 2, \dots, n$, unabhängige Zufallsvariablen mit $E[X_i] = \mu_i$ und $\text{Var}(X_i) = \sigma^2$. Dann gilt

$$E[X_i - \bar{X}]^2 = (\mu_i - \bar{\mu})^2 + \frac{n-1}{n}\sigma^2 \quad (\text{C.1})$$

wobei

$$\bar{\mu} = \frac{1}{n} \sum_{i=1}^n \mu_i$$

Um Gleichung (C.1) zu zeigen, verwenden die Beziehung $E(U)^2 = [E(U)]^2 + \text{Var}(U)$ für jede Zufallsvariable mit endlicher Varianz. Der erste Term auf der rechten Seite von Gleichung (C.1) folgt direkt. Um den zweiten Term zu verstehen, brauchen wir $\text{Var}(X_i - \bar{X})$ berechnen:

$$\text{Var}(X_i - \bar{X}) = \text{Var}(X_i) + \text{Var}(\bar{X}) - 2 \text{Cov}(X_i, \bar{X}) \quad (\text{C.2})$$

und

$$\text{Var}(X_i) = \sigma^2$$

$$\text{Var} \bar{X} = \frac{1}{n} \sigma^2$$

$$\text{Cov}(X_i, \bar{X}) = \text{Cov} \left(X_i, \frac{1}{n} \sum_{j=1}^n X_j \right) = \frac{1}{n} \sigma^2$$

Wir haben dabei benutzt, dass $\text{Cov}(X_i, X_j) = 0$, falls $i \neq j$, da die Zufallsvariablen X_i unabhängig voneinander sind. Bringt man alle diese Beziehungen zusammen, so folgt daraus Gleichung (C.1).

Anhang C. Ergänzungen und Herleitung von wichtigen Beziehungen in der Varianzanalyse

Mit Hilfe von Gleichung (C.1) können wir nun die Erwartungswerte der Quadratsummen G und E bestimmen:

$$\begin{aligned} E(E) &= \sum_{i=1}^g \sum_{j=1}^m E [Y_{ij} - \bar{Y}_{i\bullet}]^2 \\ &= \sum_{i=1}^g \sum_{j=1}^m \frac{m-1}{m} \sigma^2 \\ &= g(m-1)\sigma^2 \end{aligned}$$

In Bezug auf Gleichung (C.1) spielt hier Y_{ij} die Rolle von X_i und $\bar{Y}_{i\bullet}$ die Rolle von \bar{X} . Da $E[Y_{ij}] = E[\bar{Y}_{i\bullet}] = \mu + \tau_i$, ergibt sich für den ersten Term in Gleichung (C.1) null und somit folgt der Ausdruck oben in der zweiten Linie.

Weiter lautet der Erwartungswert von G

$$\begin{aligned} E(G) &= m \sum_{i=1}^g \sum_{j=1}^m E [Y_{i\bullet} - \bar{Y}_{\bullet\bullet}]^2 \\ &= m \sum_{i=1}^g \sum_{j=1}^m \left[\tau_i^2 + \frac{m-1}{m \cdot g} \sigma^2 \right] \\ &= m \sum_{i=1}^g \tau_i^2 + (m-1)\sigma^2 \end{aligned}$$

Hier spielt $\bar{Y}_{i\bullet}$ die Rolle von X_i und $\bar{Y}_{\bullet\bullet}$ die Rolle von \bar{X} . Des weiteren gilt $E[\bar{Y}_{i\bullet}] = \mu + \tau_i$ und $E[\bar{Y}_{\bullet\bullet}] = \mu$. Daraus folgt die zweite Linie oben.

Somit ist

$$\frac{G}{g(m-1)}$$

ein erwartungstreuer Schätzer von σ . Schreiben wir G als

$$G = \sum_{i=1}^g (m-1)s_i^2$$

wobei s_i die empirische Varianz in der i -ten Gruppe ist, dann gilt

$$\begin{aligned} \frac{G}{g(m-1)} &= \frac{\sum_{i=1}^g (m-1)s_i^2}{g(m-1)} \\ &= \frac{1}{g} \sum_{i=1}^g s_i^2 \\ &= S_{\text{Pool}}^2 \end{aligned}$$

Anhang C. Ergänzungen und Herleitung von wichtigen Beziehungen in der Varianzanalyse

Falls alle τ_i gleich null sind, dann ist der Erwartungswert von G ebenfalls gegeben durch σ^2 . In diesem Fall wird das Verhältnis von $G/[g(m - 1)]$ und $E/[(m - 1)]$ etwa eins ergeben. Falls ein τ_i hingegen verschieden von null ist, dann wird G im Vergleich zu E grösser. Als nächstes werden wir eine Teststatistik und deren Verteilung herleiten, so dass die beiden Quadratsummen miteinander verglichen werden können und die Nullhypothese, dass alle τ_i gleich null sind, getestet werden kann.

Falls die Fehlerterme unabhängig und normalverteilt mit Erwartungswert 0 und Varianz σ^2 verteilt sind, dann kann gezeigt werden, dass E/σ^2 einer Chiquadrat-Verteilung mit $g(m - 1)$ Freiheitsgraden folgt. Falls zudem alle τ_i gleich null sind, dann kann ebenfalls gezeigt werden, dass G/σ^2 einer Chiquadrat-Verteilung mit $g - 1$ Freiheitsgraden folgt. Die Teststatistik

$$F = \frac{G/(g - 1)}{E/[g(m - 1)]}$$

kann nun verwendet werden, um die folgende Nullhypothese

$$H_0 : \tau_1 = \tau_2 = \dots = \tau_g = 0$$

zu testen. Falls die Nullhypothese zutrifft, so wird die Teststatistik F nahe 1 sein. Falls die Nullhypothese nicht zutrifft, sollte der Wert der Teststatistik grösser ausfallen. Um nun entscheiden zu können, ob wir die Teststatistik verwerfen können, müssen wir die Verteilung der Teststatistik F unter der Nullhypothese kennen. Es kann gezeigt werden, dass die Nullverteilung unter der Annahme, dass die Fehler normalverteilt sind, durch eine F-Verteilung mit $(g - 1)$ und $g(m - 1)$ gegeben ist.

Anhang D.

Signaltheorie und das Wiener-Khintchine Theorem

D.1. Herleitung der Lösung der Diffusionsgleichung aus Random Walk

Wir können den Zusammenhang des makroskopisch beschriebenen Phänomens der **Diffusion** mit dem mikroskopischen Phänomen der **Brown'schen Bewegung** eines Partikels verstehen, indem wir zu einer Kontinuumsbeschreibung des Random Walks übergehen und die Schrittänge Δx sowie den zeitlichen Abstand Δt immer kleiner machen. Insbesondere können wir die zeitabhängige Teilchendichtefunktion

$$n(x, t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}} \quad (\text{D.1})$$

aus der Punktwahrscheinlichkeit für die Position des betrunkenen Bargängers

$$P(M_N = m) = \frac{2}{\sqrt{8\pi Npq}} e^{-\frac{(m-N(p-q))^2}{8Npq}} \quad (\text{D.2})$$

herleiten. Wir definieren

$$x = m\Delta x \quad \text{und} \quad t = N\Delta t$$

und führen die Parameter

$$D \equiv 2pq \frac{(\Delta x)^2}{\Delta t} \quad \text{und} \quad v \equiv (p - q) \frac{\Delta x}{\Delta t}$$

ein. D wird **Diffusionskonstante** und v wird **mittlere Driftgeschwindigkeit** genannt. Dann können wir die Normalverteilung in Gleichung (D.2) in die folgende Form bringen

$$P(X = m\Delta x; N\Delta t) = \frac{2\Delta x}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}}.$$

Anhang D. Signaltheorie und das Wiener-Khintchine Theorem

Wir führen nun den Grenzübergang $\Delta x \rightarrow 0, \Delta t \rightarrow 0$ bei konstantem D und konstantem v durch. Dies bedeutet, dass wir die beiden „makroskopischen“ Charakteristika des Zufallspfades, die mittlere Driftgeschwindigkeit v und die Diffusionskonstante D vorgeben, aber Raum und Zeit kontinuierlich machen. Bei diesem Prozess schrumpft Δx mit der Wurzel von Δt , denn bei konstantem D gilt $(\Delta x)^2 \propto \Delta t$ oder $\Delta x \propto \sqrt{\Delta t}$. Andererseits muss bei konstantem v gelten, dass die Parameter p und q sich so ändern, dass $p - q \propto \Delta x$ ist. Für $\Delta x \rightarrow 0$ nähern sich also beide Parameter p und q dem Wert $1/2$ an, wobei v konstant bleibt. Für $\Delta x \rightarrow 0$ wird aber auch die Punktwahrscheinlichkeit $P(X = m\Delta x; N\Delta t)$ null, was wir bei einer kontinuierlichen Wahrscheinlichkeitsverteilung natürlich erwarten. Nun entspricht der Ausdruck $P(X = m\Delta x; N\Delta t)$ für die diskrete Zufallsvariable X der Wahrscheinlichkeit

$$\begin{aligned} P(x < X \leq x + \Delta x; t) &= F(x + \Delta x, t) - F(x, t) \\ &= \Delta F(x, t) \\ &= \frac{2\Delta x}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}}, \end{aligned}$$

also der Differenz der kumulativen Wahrscheinlichkeitsfunktion. Wir erhalten dann folgende Wahrscheinlichkeitsdichte für die Position x des Brownschen Partikels zur Zeit t

$$\frac{\Delta F(x, t)}{2\Delta x} \xrightarrow{\Delta x \rightarrow 0} f(x; t) = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}}, \quad (\text{D.3})$$

Die Position des Brownschen Partikels zur Zeit $t = 0$ bezeichnen wir mit x_0 . In unserer Herleitung aus dem Random Walk haben wir stets $x_0 = 0$ angenommen. Falls dies nicht der Fall ist, ersetzen wir x einfach durch $x - x_0$. Der Parameter D wird **Diffusionskonstante** und v wird **mittlere Driftgeschwindigkeit** genannt.

Nehmen wir einfacheitshalber an, dass die Driftgeschwindigkeit verschwindet (falls $p = q$, dann ist $v = 0$) und die Ausgangsposition x_0 nicht null sein muss, dann ergibt sich die Teilchendichtefunktion (D.1) nun, indem wir die n Brownschen Teilchen mit $f(x; t)$ multiplizieren:

$$n(x, t) = n \cdot f(x; t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}} \quad (\text{D.4})$$

Diffusionsgleichung

Die Teilchendichte

$$n(x, t) = \frac{n}{\sqrt{4\pi Dt}} e^{-\frac{(x-x_0)^2}{4Dt}}$$

löst die **Diffusionsgleichung**

$$\frac{\partial n(x, t)}{\partial t} = D \frac{\partial^2 n(x, t)}{\partial x^2}, \quad (\text{D.5})$$

mit der Randbedingung, dass sich alle Teilchen zum Zeitpunkt $t = 0$ bei $x = 0$ befinden. Der Random Walk in der Kontinuumsbeschreibung wird **Brown'sche Bewegung** genannt und kann also als **Diffusionsprozess** aufgefasst werden.

Die beiden Terme auf der rechten Seite von Gleichung (D.5) haben eine einfache anschauliche Bedeutung: Der erste Term (mit v) verschiebt die Wahrscheinlichkeitsdichte $f(x, t)$ mit Geschwindigkeit v nach rechts (wenn v positiv ist; sonst wird die Verteilung nach links geschoben). Man nennt diesen Term den **Driftterm**. Für $p = q$ ist $v = 0$, und der Driftterm fällt weg. Der Term auf der rechten Seite von Gleichung (D.5) macht die Verteilung breiter; er wird deshalb **Diffusionsterm** genannt. Physikalisch stellen wir uns einen Tintentropfen vor, der in heißes Wasser gegeben wird: durch Diffusion verteilt sich die Tinte im Wasser aufgrund der Stöße zwischen den sich zufällig bewegenden Wassermolekülen und Tintenmolekülen.

D.2. Korrelation von Signalen und Wiener-Khintchine-Theorem

Eine der interessantesten Fragen in der Signalverarbeitung lautet: Wie ähnlich sind zwei Signale? Diese Frage stellt sich immer dann, wenn man ein bekanntes Muster in einem Signal wiederfinden will, sei es um einen Funkempfänger mit dem Trägersignal einer Übertragung zu synchronisieren, oder um in einer Sprachaufnahme Worte herauszufiltern. Ein Konzept, das hierfür zum Einsatz kommt, heißt Korrelation („Zusammenhang“).

D.3. Energie, Leistung, Korrelation

Die Energie eines Spannungssignal in einem Stromkreis mit einem bekannten Widerstand R ist

$$E = \int P(t) dt = \int U(t)I(t) dt = \frac{1}{R} \int U^2(t) dt.$$

Sie hängt also vom Quadrat der Spannung ab.

Signalenergie

In diesem Sinne definiert man die **Signalenergie** als Integral über das Quadrat eines Signals:

$$E_s = \int_{t_1}^{t_2} |s(t)|^2 dt,$$

wobei der Betrag gebildet wird, damit die Formel auch für komplexwertige Signale gültig bleibt. Falls $s(t)$ ein periodisches Signal ist, bietet es sich an, über eine Periode zu integrieren. Um einen vergleichbaren Wert zu erhalten, wenn man über unterschiedliche Zeitintervalle integriert, normiert man die Energie mit der Zeit, über die man integriert.

Signalleistung

Wir definieren deshalb die **Signalleistung** als

$$P_s = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} |s(t)|^2 dt.$$

D.3.1. Ähnlichkeitsmass für Signale

Die mittlere quadratische Abweichung zweier Signale $s(t)$ und $g(t)$ ist

$$\begin{aligned} \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} [s(t) - g(t)]^2 dt &= \lim_{T \rightarrow \infty} \frac{1}{T} \left(\int_{-T/2}^{T/2} s(t)^2 dt + \int_{-T/2}^{T/2} g(t)^2 dt - 2 \int_{-T/2}^{T/2} s(t)g(t) dt \right) \\ &= P_s + P_g - 2 \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} s(t)g(t) dt \end{aligned}$$

Beim Vergleich zweier Signale interessiert man sich nun nicht für absolute Leistungen, so dass bloss der letzte Summand im obigen Ausdruck von Interesse ist.

Kreuzkorrelationskoeffizient

Ein Mass für die Ähnlichkeit zweier reellwertiger Signale $s(t)$ und $g(t)$, das von absoluten Leistungswerten unabhängig ist, ist der **Kreuzkorrelationskoeffizient**, den wir definieren als

$$\Phi_{sg} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} s^*(t)g(t) dt,$$

wobei $s^*(t)$ das zu $s(t)$ komplex konjugierte Signal bezeichnet.

Der Kreuzkorrelationskoeffizient wird genau dann null, wenn zwischen zwei Signalen kein linearer Zusammenhang besteht. Diese Definition der Korrelation ist nur dann nützlich, wenn man eine feste zeitliche Lage der Funktionen zueinander annimmt. Können stochastische Signale auch verschoben zueinander auftreten, wie es in der Technik in der Regel der Fall ist, macht es Sinn, den Kreuzkorrelationskoeffizienten zu einer Funktion der Zeitverschiebung zu erweitern.

Kreuzkorrelationsfunktion

Wir definieren die **Kreuzkorrelationsfunktion** für stationäre stochastische Signale $s(t)$ und $g(t)$

$$\varphi_{sg}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} s^*(t)g(t + \tau) dt.$$

Den Kreuzkorrelationskoeffizienten erhält man aus der Kreuzkorrelationsfunktion ganz trivial mit $\Phi_{sg} = \varphi_{sg}(0)$. Eine der häufigeren Anwendungen dieser Definition ist der Vergleich eines Signals mit sich selbst, genannt Autokorrelation.

Autokorrelationsfunktion

Die **Autokorrelationsfunktion** (AKF) ist definiert als

$$\varphi_{ss}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T/2}^{T/2} s^*(t)s(t + \tau) dt.$$

Insbesondere gilt

$$\varphi_{ss}(0) = \Phi_{ss} = P_s.$$

Zu den Anwendungen der AKF gehört einerseits festzustellen, wie repetitiv ein Signal ist, d.h. ob Periodizitäten vorliegen. In diesem Fall hat die AKF auch an anderen Stellen als $\varphi(0)$ Werte, die von null verschieden sind. Andererseits ist nur ein Hintergrundrauschen dermassen zufällig, dass Periodizitäten nicht auftreten und sich das Rauschsignal nie wiederholt. Man kann ein störendes Rauschen also eben daran erkennen, dass die AKF an der Nullstelle einen sehr scharfen Peak hat, sonst überall verschwindet.

D.4. Korrelation und Faltung

Im folgenden nehmen wir an, dass die Signale $s(t)$ und $g(t)$ reellwertig sind, also $s^* = s(t)$. Weiter nehmen wir an, dass wir nur über ein endliches Zeitintervall T integrieren. Wir schreiben also im folgenden

$$\varphi_{sg}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} = \int_{-T/2}^{T/2} s(t)g(t + \tau) dt \approx \frac{1}{T} \int_{-T/2}^{T/2} s(t)g(t + \tau) dt = \varphi_{sg;T}(\tau)$$

Man stellt bei der Kreuzkorrelationsfunktion eine gewisse Ähnlichkeit zur Faltung fest:

$$\begin{aligned} \varphi_{sg;T}(\tau) &= \frac{1}{T} \int_{-T/2}^{T/2} s(t)g(t + \tau) dt \\ &= \frac{1}{T} \int_{-T/2}^{T/2} s(-t')g(\tau - t') dt' \\ &\quad \underbrace{\phantom{\int_{-T/2}^{T/2}}}_{s(-\tau)*g(\tau)} \end{aligned}$$

wobei wir folgende Substitution verwendet haben: $t' = -t$.

D.4.1. Kreuzkorrelationstheorem

Es bietet sich nun geradezu an, die Signale mit Hilfe der Fourier-Transformation in den Bildraum zu überführen. Im Bildraum berechnet sich die Faltung als Produkt

Anhang D. Signaltheorie und das Wiener-Khintchine Theorem

der Fourier-Transformierten der Signale. Es ergibt sich:

$$\begin{aligned}
 \Phi_{sg;T}(\omega) &\equiv \int_{-\infty}^{\infty} \varphi_{sg;T}(\tau) e^{-j\omega\tau} d\tau \\
 &= \mathcal{F}\{\varphi_{sg;T}(\tau)\} \\
 &= \frac{1}{T} \mathcal{F}\{s(-\tau) * g(\tau)\} \\
 &= \frac{1}{T} \mathcal{F}\{s(-\tau)\} \cdot \mathcal{F}\{g(\tau)\} \\
 &= \frac{1}{T} S^*(\omega) \cdot G(\omega),
 \end{aligned}$$

wobei $S^*(\omega)$ die komplex konjugierte Funktion von $S(\omega)$ bezeichnet und T die Integrationszeit für die Faltung ist. Wir haben dabei benutzt, dass für ein reellwertiges $s(t)$ gilt

$$\begin{aligned}
 \mathcal{F}\{s(-t)\} &= \int_{-\infty}^{\infty} s(-t) e^{-j\omega t} dt \\
 &= \int_{-\infty}^{\infty} s(t') e^{+j\omega t'} dt' \\
 &= S^*(\omega),
 \end{aligned}$$

Kreuzkorrelationstheorem

Das **Kreuzkorrelationsfunktionstheorem** lautet also

$$\Phi_{sg}(\omega) = \frac{1}{T} \cdot S^*(\omega) \cdot G(\omega).$$

D.5. Wiener-Khintchine Theorem

Wiener-Khintchine Theorem

Das Kreuzkorrelationstheorem ergibt im Falle der Autokorrelationsfunktion, also im Fall $s(\tau) = g(\tau)$, das Wiener-Khintchine-Theorem

$$\mathcal{F}\{\varphi_{ss;T}(\tau)\} = \Phi_{ss;T}(\omega) = \frac{1}{T} \cdot S^*(\omega) \cdot S(\omega) = \frac{1}{T} |S(\omega)|^2$$

$\frac{|S(\omega)|^2}{T}$ wird das **Leistungsdichtespektrum** des Signals $s(t)$ genannt. Die Fourier-Transformierte der Autokorrelationsfunktion $\varphi_{ss;T}(\tau)$ ergibt also das Leistungsdichtespektrum des Signals.

Der Grund, warum $\frac{|S(\omega)|^2}{T}$ Leistungsdichtespektrum genannt wird, wird aus folgender Rechnung ersichtlich

$$\begin{aligned} \varphi_{ss;T}(\tau = 0) &= \mathcal{F}^{-1}\{\Phi_{ss;T}(\omega)\} \Big|_{\tau=0} \\ &= \mathcal{F}^{-1}\left\{\frac{1}{T} |S(\omega)|^2\right\} \Big|_{\tau=0} \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{|S(\omega)|^2}{T} e^{j\omega\tau} d\omega \Big|_{\tau=0} \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{|S(\omega)|^2}{T} d\omega \end{aligned}$$

Erinnern wir uns an die Definition der Signalleistung, so ergibt sich¹

$$\varphi_{ss}(0) = P_s = \frac{1}{T} \int_{-T/2}^{T/2} |s(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{|S(\omega)|^2}{T} d\omega.$$

$\frac{1}{T} |S(\omega)|^2 d\omega$ gibt also die Teilleistung des Signals an, die auf das Frequenzband $[\omega, \omega + d\omega]$ entfällt. Wenn man ein Spannungssignal in einem realen Stromkreis betrachtet, dann hat die Autokorrelationsfunktion im Frequenzbereich die Einheit $1/\Omega \cdot V^2 s^3 = W Hz^{-1}$.

¹Dies ist im übrigen nichts anderes als das Parseval'sche Theorem.

Signalleistung

Die **Signalleistung** P_s eines stochastischen Signals $S(t)$ berechnet sich mit Hilfe des Wiener-Khintchine Theorems gemäss

$$P_s = \frac{1}{2\pi} \int_{-\infty}^{\infty} \Phi_{ss}(\omega) d\omega,$$

wobei $\Phi_{ss}(\omega)$ die Fourier-Transformierte der Autokorrelationsfunktion $\varphi_{ss}(\tau)$ ist.

Beispiel D.5.1

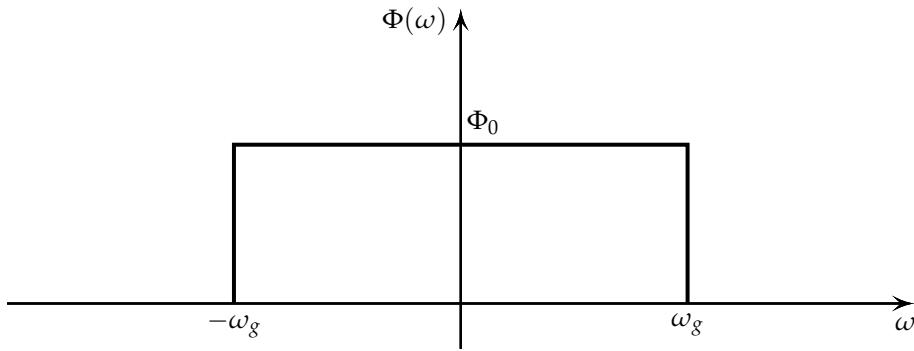


Abbildung D.1.: Leistungsdichtespektrum von weissem bandbegrenztem Rauschen.

Wir können nun das Wiener-Khintchine-Theorem auf das Beispiel des Gauss'schen Rauschsignals anwenden. Wir nehmen für das Rauschsignal $N(t)$ weisses, bei der Kreisfrequenz ω_g bandbegrenztes Rauschen an – „weiss“ bedeutet ein Leistungsdichtespektrum $\Phi(\omega)$, das konstant ist über alle vorkommenden Frequenzen, also $\Phi(\omega) = \Phi_0$. Dies trifft auf das Leistungsspektrum $\Phi(\omega)$ gemäss der Abbildung D.1 zu. Aus $\Phi(\omega)$ lässt sich nun die Autokorrelationsfunktion $\varphi(\tau)$ folgendermassen berechnen:

$$\begin{aligned} \varphi_n(\tau) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \Phi(\omega) e^{j\omega\tau} d\omega = \frac{1}{2\pi} \int_{-\omega_g}^{\omega_g} \Phi_0 e^{j\omega\tau} d\omega \\ &= \frac{\Phi_0}{2\pi} \left[\frac{e^{j\omega\tau}}{j\tau} \right]_{-\omega_g}^{\omega_g} = \frac{\Phi_0}{2j\pi\tau} \left[e^{j\omega_g\tau} - e^{-j\omega_g\tau} \right] \\ &= \Phi_0 \frac{\omega_g}{\pi} \frac{\sin(\omega_g\tau)}{\omega_g\tau}. \end{aligned} \tag{D.6}$$

□

Beispiel D.5.2

Wir betrachten im folgenden das Rechtecksignal

$$s(t) = -A\sigma(t + T_0/2) + 2A\sigma(t) - A\sigma(t - T_0/2).$$

Die Fourier-Transformierte des Signals $s(t)$ ist gegeben durch

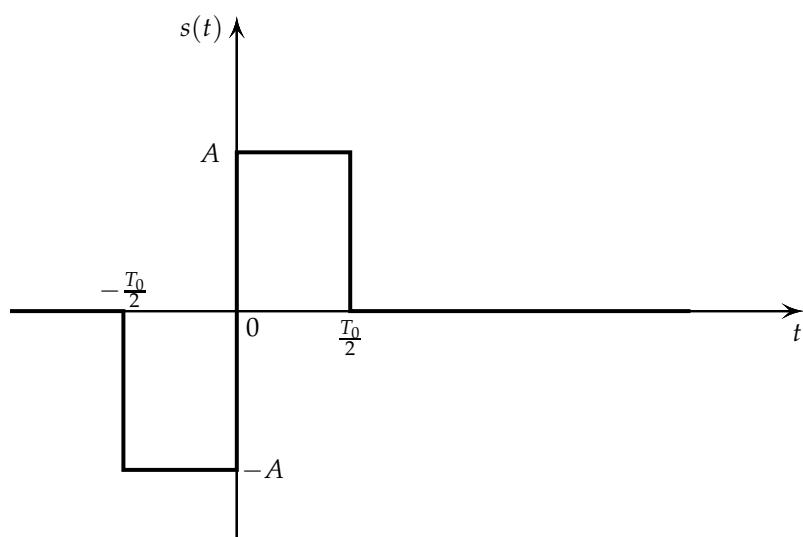


Abbildung D.2.: Rechtecksignal $s(t) = -A\sigma(t + T_0/2) + 2A\sigma(t) - A\sigma(t - T_0/2)$.

Anhang D. Signaltheorie und das Wiener-Khintchine Theorem

$$\begin{aligned}
S(\omega) &= \int_{-\infty}^{\infty} \left(-A\sigma(t + T_0/2) + 2A\sigma(t) - A\sigma(t - T_0/2) \right) e^{-j\omega t} dt \\
&= \int_{-T_0/2}^0 -A \cdot e^{-j\omega t} dt + \int_0^{T_0/2} A \cdot e^{-j\omega t} dt \\
&= \frac{A}{j\omega} \left(1 - e^{+j\omega T_0/2} \right) + \frac{A}{j\omega} \left(1 - e^{-j\omega T_0/2} \right) \\
&= \frac{A}{j\omega} \left(2 - e^{+j\omega T_0/2} - e^{-j\omega T_0/2} \right) \\
&= \frac{2A}{j\omega} \left(1 - \cos(\omega T_0/2) \right) \\
&= \frac{4A}{j\omega} \sin^2(\omega T_0/4)
\end{aligned}$$

Die Fourier-Transformierte eines reellwertigen Signals $s(-t)$ ist gegeben durch

$$\begin{aligned}
\mathcal{F}\{s(-t)\} &= \int_{-\infty}^{\infty} s(-t) e^{-j\omega t} dt \\
&= \int_{-\infty}^{\infty} s(t') e^{+j\omega t'} dt' \\
&= S^*(\omega),
\end{aligned}$$

wobei wir mit $S^*(\omega)$ die zu $S(\omega)$ komplex konjugierte Funktion bezeichnen. Somit ist

$$\begin{aligned}
\mathcal{F}\{s(-t)\} &= \frac{-A}{j\omega} \left(2 - e^{-j\omega T_0/2} - e^{+j\omega T_0/2} \right) \\
&= \frac{-2A}{j\omega} \left(1 - \cos(\omega T_0/2) \right) \\
&= \frac{-4A}{j\omega} \sin^2(\omega T_0/4)
\end{aligned}$$

Anhang D. Signaltheorie und das Wiener-Khintchine Theorem

Im Bildbereich hat die Autokorrelationsfunktion $\varphi(\tau)$ folgende Form

$$\begin{aligned}\mathcal{F}\{\varphi(\tau)\} &= \frac{1}{T_0} S^*(\omega) \cdot S(\omega) \\ &= \frac{A^2}{\omega^2 T_0} \left(2 - e^{+j\omega T_0/2} - e^{-j\omega T_0/2} \right)^2 \\ &= \frac{A^2}{\omega^2 T_0} \left(6 - 4e^{-j\omega T_0/2} - 4e^{+j\omega T_0/2} + e^{-j\omega T_0} + e^{j\omega T_0} \right).\end{aligned}$$

Transformieren wir zurück in den Zeitraum, finden wir

$$\begin{aligned}\varphi(\tau) &= \frac{A^2}{T_0} \left(-6\tau\sigma(\tau) + 4(\tau - T_0/2)\sigma(\tau - T_0/2) + 4(\tau + T_0/2)\sigma(\tau + T_0/2) \right. \\ &\quad \left. - (\tau - T_0)\sigma(\tau - T_0) - (\tau + T_0)\sigma(\tau + T_0) \right) \\ &= \begin{cases} -A^2(1 + \tau/T_0), & -T_0 \leq \tau \leq -\frac{T_0}{2} \\ +A^2(1 + 3\tau/T_0), & -T_0/2 \leq \tau \leq 0 \\ +A^2(1 - 3\frac{\tau}{T_0}), & 0 \leq \tau \leq T_0/2 \\ +A^2(-1 + \frac{\tau}{T_0}), & T_0/2 \leq \tau \leq T_0. \end{cases}\end{aligned}$$

Den Graphen sehen Sie in Abbildung D.3.

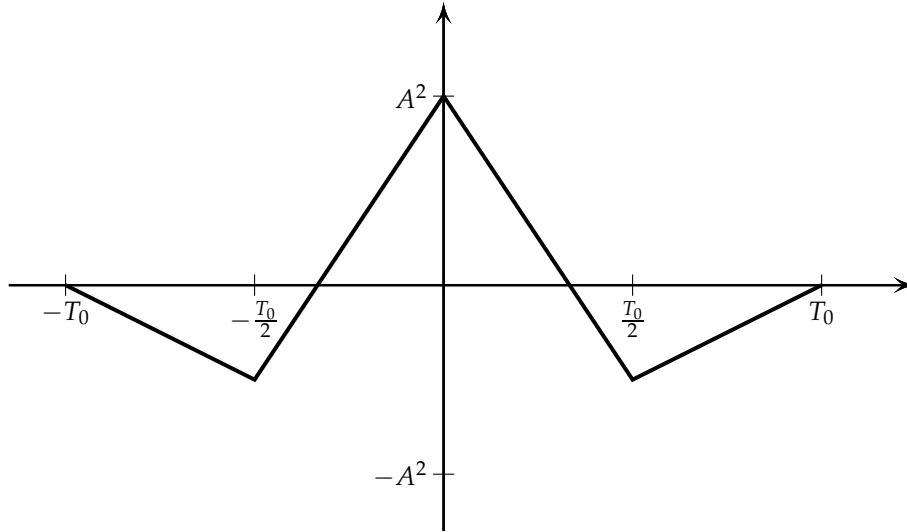


Abbildung D.3.: Rücktransformation in den Zeitraum

□

Beispiel D.5.3

Wir betrachten im folgenden das T_0 -periodische Rechtecksignal. Das T_0 -periodische

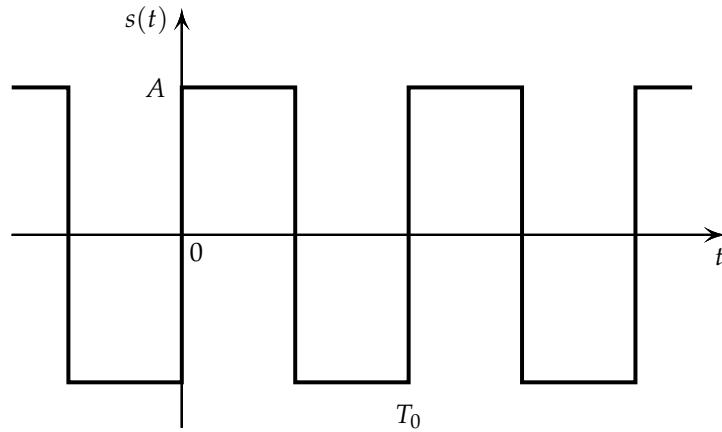


Abbildung D.4.: Das T_0 -periodische Rechtecksignal

Signal kann als Fourier-Reihe geschrieben werden:

$$s(t) = \sum_{k=-\infty}^{+\infty} c_k e^{jk\omega_0 t},$$

wobei $\omega_0 = 2\pi/T_0$ und

$$c_k = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} s(t) e^{-jk\omega_0 t} dt.$$

Aus Beispiel B folgt

$$\begin{aligned} c_k &= \frac{4A}{jk\omega_0 T_0} \sin^2(k\omega_0 T_0 / 4) \\ &= \frac{4A}{jk\omega_0 T_0} \sin^2(k\pi/2) \\ &= \begin{cases} \frac{2A}{jk\pi} & k \text{ ungerade} \\ 0 & k \text{ gerade} \end{cases} \end{aligned}$$

Anhang D. Signaltheorie und das Wiener-Khintchine Theorem

Somit gilt für jedes $k \in \mathbb{N}_0$

$$c_{2k} = 0, \quad c_{2k-1} = \frac{2A}{j\pi} \cdot \frac{1}{2k-1}.$$

Die Fourier-Reihe für das T_0 -periodische Rechtecksignal lautet also:

$$s(t) = \frac{2A}{j\pi} \sum_{k=-\infty}^{+\infty} \frac{1}{2k-1} e^{j(2k-1)\omega_0 t},$$

Die Autokorrelationsfunktion für eine T_0 -periodische Funktion ist definiert als

$$\begin{aligned} \varphi_{ss}(\tau) &= \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} s(t)s(t+\tau) dt \\ &= \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} \left(\frac{2A}{j\pi} \sum_{k=-\infty}^{+\infty} \frac{1}{2k-1} e^{j(2k-1)\omega_0 t} \right) \cdot \left(\frac{2A}{j\pi} \sum_{m=-\infty}^{+\infty} \frac{1}{2m-1} e^{j(2m-1)\omega_0(t+\tau)} \right) dt \\ &= \frac{-4A^2}{\pi^2} \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} \left(\sum_{k=-\infty}^{+\infty} \frac{1}{2k-1} \right) \cdot \left(\sum_{m=-\infty}^{+\infty} \frac{1}{2m-1} \right) e^{j(2k-1)\omega_0 t} e^{j(2m-1)\omega_0(t+\tau)} dt \\ &= \frac{-4A^2}{\pi^2} \left(\sum_{k=-\infty}^{+\infty} \frac{1}{2k-1} \right) \cdot \left(\sum_{m=-\infty}^{+\infty} \frac{1}{2m-1} \right) e^{j(2m-1)\omega_0 \tau} \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} e^{j(2k+2m-2)\omega_0 t} dt \end{aligned}$$

Nun gilt für $m, n \in \mathbb{N}_0$

$$\frac{1}{T_0} \int_{-T_0/2}^{T_0/2} e^{j(2k+2m-2)\omega_0 t} dt = \begin{cases} 0, & \text{für } 2m+2k-2 \neq 0 \\ 1, & \text{für } 2m+2k-2 = 0 \end{cases}$$

Beim Produkt der unendlichen Reihe bleiben nur die Term übrig, die $k+m-1 = 0$ erfüllen. Also finden wir

$$\begin{aligned} \varphi_{ss}(\tau) &= \frac{-4A^2}{\pi^2} \sum_{m,k: m+k=1} \left(\frac{1}{2m-1} \cdot \frac{1}{2k-1} \right) e^{j(2m-1)\omega_0 \tau} \\ &= \frac{4A^2}{\pi^2} \sum_{m=-\infty}^{\infty} \left(\frac{1}{2m-1} \right)^2 e^{j(2m-1)\omega_0 \tau}. \end{aligned}$$

Die Autokorrelationsfunktion ergibt dann die T_0 -periodische Fortsetzung von

$$\varphi(\tau) = \begin{cases} A^2(1 - 4\frac{\tau}{T_0}), & 0 \leq \tau \leq \frac{T_0}{2} \\ A^2(1 - 4\frac{\tau}{T_0}), & -\frac{T_0}{2} \leq \tau \leq 0. \end{cases}$$

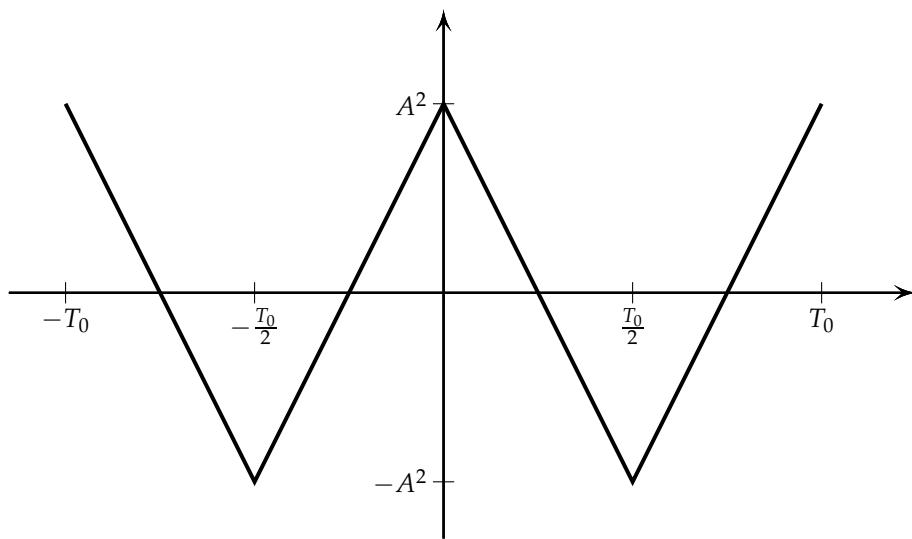


Abbildung D.5.: T_0 -periodische Fortsetzung

Den Graphen sehen Sie in Abbildung D.5.

□