

# **Statistik1**

Sebastian Sauer

2024-08-30

# Inhaltsverzeichnis

|   |           |
|---|-----------|
| <b>1. Willkommen!</b>   | <b>3</b>  |
| 1.1. Freude am Lernen . . . . .                               | 3         |
| 1.2. Es geht um Ihren Lernerfolg . . . . .                    | 4         |
| 1.2.1. Lernziele . . . . .                                    | 4         |
| 1.2.2. Was lerne ich hier und wozu ist das gut? . . . . .     | 5         |
| 1.2.3. Was ist hier das Erfolgsgeheimnis? . . . . .           | 6         |
| 1.2.4. Motivieren Sie mich! . . . . .                         | 6         |
| 1.2.5. Voraussetzungen . . . . .                              | 7         |
| 1.2.6. Überblick . . . . .                                    | 7         |
| 1.3. Software: R . . . . .                                    | 8         |
| 1.4. Zum Autor . . . . .                                      | 8         |
| 1.5. Nomenklatur . . . . .                                    | 8         |
| 1.5.1. Griechische Buchstaben . . . . .                       | 8         |
| 1.6. Zitation . . . . .                                       | 9         |
| 1.7. Reproduzierbarkeit . . . . .                             | 9         |
| <b>I. Organisatorisches</b>                                   | <b>10</b> |
| <b>2. Prüfung</b>   | <b>11</b> |
| 2.1. Prüfungleistung . . . . .                                | 11        |
| 2.2. Zum Prognosewettbewerb . . . . .                         | 11        |
| 2.3. Prüfungsrelevanter Stoff . . . . .                       | 11        |
| 2.4. Wie kann ich mich auf die Prüfung vorbereiten? . . . . . | 11        |
| 2.5. Allgemeine Prüfungscheinweise . . . . .                  | 12        |
| 2.6. Lieblingsfehler . . . . .                                | 12        |
| 2.7. Fazit . . . . .  | 12        |
| <b>II. Vorbereiten</b>  | <b>13</b> |
| <b>3. Rahmen</b>  | <b>14</b> |
| 3.1. Lernsteuerung . . . . .                                  | 14        |
| 3.1.1. Standort im Lernpfad . . . . .                         | 14        |
| 3.1.2. Lernziele . . . . .                                    | 14        |
| 3.1.3. Einstieg . . . . .                                     | 15        |
| 3.1.4. Erfolsgrezept . . . . .                                | 15        |
| 3.2. Was ist Statistik und wozu ist sie gut? . . . . .        | 15        |
| 3.2.1. Daten zusammenfassen . . . . .                         | 16        |

|  |           |
|--|-----------|
| 3.2.2. Unterschiedlichkeit messen . . . . .                | 17        |
| 3.3. Was ist das Ziel Ihrer Analyse? . . . . .             | 18        |
| 3.3.1. Arten von Zielen . . . . .                          | 18        |
| 3.3.2. Forschungsfrage . . . . .                           | 19        |
| 3.3.3. Aus der Forschung: Smartphone-Brain-Drain . . . . . | 20        |
| 3.3.4. Der Prozess der Datenanalyse . . . . .              | 21        |
| 3.4. Was sind Daten? . . . . .                             | 21        |
| 3.4.1. Was ist eine Variable? . . . . .                    | 22        |
| 3.4.2. Beobachtungseinheit . . . . .                       | 23        |
| 3.4.3. Wert . . . . .                                      | 23        |
| 3.4.4. Tidy Data . . . . .                                 | 24        |
| 3.4.5. Je mehr, desto besser (?) . . . . .                 | 27        |
| 3.5. Arten von Variablen . . . . .                         | 28        |
| 3.5.1. Nach Position in der Forschungsfrage . . . . .      | 28        |
| 3.5.2. Nach dem Skalenniveau . . . . .                     | 29        |
| 3.5.3. Beispiele für Skalenniveaus . . . . .               | 29        |
| 3.6. Modelle . . . . .                                     | 33        |
| 3.6.1. Vorher . . . . .                                    | 34        |
| 3.6.2. Nachher . . . . .                                   | 34        |
| 3.7. Praxisbezug . . . . .                                 | 34        |
| 3.8. Wie man mit Statistik lügt . . . . .                  | 35        |
| 3.9. Fazit . . . . .                                       | 35        |
| 3.10. Aufgaben . . . . .                                   | 35        |
| 3.11. Vertiefung . . . . .                                 | 36        |
| 3.11.1. Excel für Könner . . . . .                         | 36        |
| 3.11.2. Sind wir süchtig nach dem Handy? . . . . .         | 36        |
| 3.11.3. Datenprofi plaudert aus dem Nähkästchen . . . . .  | 36        |
| 3.12. Literaturhinweise . . . . .                          | 37        |
| 3.13. Literatur . . . . .                                  | 37        |
| <b>4. Daten einlesen</b>                                   | <b>38</b> |
| 4.1. Lernsteuerung . . . . .                               | 38        |
| 4.1.1. Standort im Lernpfad . . . . .                      | 38        |
| 4.1.2. Lernziele . . . . .                                 | 38        |
| 4.1.3. Überblick . . . . .                                 | 38        |
| 4.1.4. Ab diesem Kapitel benötigen Sie R . . . . .         | 38        |
| 4.1.5. Begleitvideos . . . . .                             | 39        |
| 4.2. Errrstkontakt . . . . .                               | 39        |
| 4.2.1. Warum R? . . . . .                                  | 39        |
| 4.2.2. R und Reproduzierbarkeit . . . . .                  | 40        |
| 4.2.3. R & RStudio . . . . .                               | 41        |
| 4.3. Installation von R und RStudio . . . . .              | 43        |
| 4.3.1. Installation von R . . . . .                        | 43        |
| 4.3.2. Installation von RStudio Desktop . . . . .          | 43        |
| 4.3.3. RStudio Cloud . . . . .                             | 43        |
| 4.4. RStudio starten, nicht R . . . . .                    | 44        |

|           |                                    |           |
|-----------|------------------------------------|-----------|
| 4.5.      | R-Pakete                           | 45        |
| 4.5.1.    | Was sind R-Pakete?                 | 45        |
| 4.5.4.    | Pakete installieren                | 45        |
| 4.5.2.    | Viele Pakete                       | 46        |
| 4.5.3.    | Es kommen viele dazu               | 46        |
| 4.5.5.    | Pakete starten                     | 48        |
| 4.6.      | Mit R arbeiten                     | 48        |
| 4.6.1.    | Projekte in R                      | 48        |
| 4.6.2.    | Skriptdateien                      | 49        |
| 4.6.3.    | Quarto-Dokumente                   | 50        |
| 4.7.      | Errisch für Einsteiger             | 50        |
| 4.7.1.    | Variablen                          | 50        |
| 4.7.2.    | Funktionen (“Befehle”)             | 53        |
| 4.7.3.    | Unsere erste statistische Funktion | 54        |
| 4.7.4.    | Vektorielles Rechnen               | 57        |
| 4.7.5.    | R-Quiz                             | 58        |
| 4.7.6.    | Ich brauche R-Hilfe!               | 58        |
| 4.8.      | Mit Daten arbeiten                 | 59        |
| 4.8.1.    | Wo sind meine Daten?               | 59        |
| 4.8.2.    | Gebräuchliche Datenformate         | 60        |
| 4.8.3.    | Aufgabe                            | 60        |
| 4.8.4.    | Lösung                             | 60        |
| 4.8.5.    | Daten importieren                  | 61        |
| 4.8.6.    | Dataframes                         | 63        |
| 4.8.7.    | Tabellen in R betrachten           | 64        |
| 4.9.      | Logikprüfung                       | 64        |
| 4.10.     | Praxisbezug                        | 65        |
| 4.11.     | Aufgaben                           | 66        |
| 4.12.     | Vertiefung                         | 67        |
| 4.12.1.   | Varianten zu <code>read.csv</code> | 67        |
| 4.12.2.   | Importieren von Excel-Tabellen     | 68        |
| 4.12.3.   | Der Dollar-Operator                | 69        |
| 4.12.4.   | R-Zertifikat bei LinkedIn          | 70        |
| 4.12.5.   | R-Funktionen verschachteln         | 70        |
| 4.12.6.   | R und Friends updaten              | 71        |
| 4.12.7.   | Benötigte R-Pakete                 | 71        |
| 4.12.8.   | Benötigte Daten                    | 71        |
| 4.13.     | Literaturhinweise                  | 72        |
| 4.14.     | Literatur                          | 72        |
| <b>5.</b> | <b>Daten umformen</b>              | <b>73</b> |
| 5.1.      | Lernsteuerung                      | 73        |
| 5.1.1.    | Standort im Lernpfad               | 73        |
| 5.1.2.    | Lernziele                          | 73        |
| 5.1.3.    | Benötigte R-Pakete                 | 73        |
| 5.1.4.    | Benötigte Daten                    | 73        |

|  |    |
|--|----|
| 5.1.5. R-Code zum Copy-Pasten . . . . .  | 74 |
| 5.1.6. Frag den Bot . . . . .  | 74 |
| 5.1.7. Zum Einstieg . . . . .  | 74 |
| 5.2. Datenjudo . . . . .   | 74 |
| 5.2.1. Die Wahrheit über Data Science . . . . .                                  | 74 |
| 5.2.2. Praxisbezug: Aus dem Alltag des Data Scientisten . . . . .                | 75 |
| 5.2.3. Mach's einfach . . . . .  | 75 |
| 5.3. Die Verben des Datenjudos . . . . .   | 77 |
| 5.3.1. Tabelle sortieren: <code>arrange</code> . . . . .                         | 78 |
| 5.3.2. Zeilen filtern: <code>filter</code> . . . . .                             | 79 |
| 5.3.3. Spalten auswählen mit <code>select</code> . . . . .                       | 80 |
| 5.3.4. Spalten zu einer Zahl zusammenfassen mit <code>summarise</code> . . . . . | 81 |
| 5.3.5. Tabelle gruppieren . . . . .  | 82 |
| 5.3.6. Aufgabe . . . . .   | 83 |
| 5.3.7. Lösung . . . . .  | 83 |
| 5.3.8. Spalten verändern mit <code>mutate</code> . . . . .                       | 84 |
| 5.3.9. Aufgabe . . . . .   | 85 |
| 5.3.10. Lösung . . . . .   | 85 |
| 5.3.11. Aufgabe . . . . .  | 85 |
| 5.3.12. Lösung . . . . .   | 85 |
| 5.3.13. Zeilen zählen mit <code>count</code> . . . . .                           | 86 |
| 5.3.14. Fazit: Verben am Fließband . . . . .                                     | 87 |
| 5.4. Die Pfeife . . . . .  | 88 |
| 5.4.1. Russische Puppen . . . . .  | 88 |
| 5.4.2. Die Pfeife zur Rettung . . . . .  | 89 |
| 5.5. Beispiele für Forschungsfragen . . . . .                                    | 91 |
| 5.5.1. Forschungsfrage 1 . . . . .   | 92 |
| 5.5.2. Frage . . . . .   | 92 |
| 5.5.3. Antwort . . . . .   | 92 |
| 5.5.4. Forschungsfrage 2 . . . . .   | 92 |
| 5.5.5. Frage . . . . .   | 92 |
| 5.5.6. Antwort . . . . .   | 93 |
| 5.5.7. Forschungsfrage 3 . . . . .   | 93 |
| 5.5.8. Frage . . . . .   | 93 |
| 5.5.9. Antwort . . . . .   | 94 |
| 5.6. Praxisbezug . . . . .   | 94 |
| 5.7. Wie man mit Statistik lügt . . . . .  | 94 |
| 5.8. Fallstudien . . . . .   | 95 |
| 5.8.1. Die Pinguine . . . . .  | 95 |
| 5.8.2. Weitere Fallstudien . . . . .   | 96 |
| 5.9. Aufgaben . . . . .  | 97 |
| 5.10. Vertiefung . . . . .   | 98 |
| 5.10.1. Tidydatatutor . . . . .  | 98 |
| 5.10.2. Fortgeschrittenes R . . . . .  | 98 |
| 5.10.3. Hilfe?! Erbie! . . . . .   | 98 |
| 5.10.4. Zertifikate und Online-Kurse . . . . .                                   | 99 |

|  |            |
|--|------------|
| 5.11. Exkurs . . . . .                                   | 99         |
| 5.12. Literaturhinweise . . . . .                        | 100        |
| 5.13. Literatur . . . . .                                | 100        |
| <b>III. Modellieren</b>                                  | <b>101</b> |
| <b>6. Daten verbildlichen</b>                            | <b>102</b> |
| 6.1. Lernsteuerung . . . . .                             | 102        |
| 6.1.1. Standort im Lernpfad . . . . .                    | 102        |
| 6.1.2. Lernziele . . . . .                               | 102        |
| 6.1.3. Benötigte R-Pakete . . . . .                      | 102        |
| 6.1.4. Benötigte Daten . . . . .                         | 102        |
| 6.1.5. R-Code zum Copy-Pasten . . . . .                  | 103        |
| 6.1.6. Quiz zum Einstieg . . . . .                       | 103        |
| 6.1.7. Wozu das alles? . . . . .                         | 103        |
| 6.2. Ein Dino sagt mehr als 1000 Worte . . . . .         | 103        |
| 6.2.1. Datendiagramm . . . . .                           | 104        |
| 6.2.2. Ein Bild hat nicht so viele Dimensionen . . . . . | 104        |
| 6.3. Nomenklatur von Datendiagrammen . . . . .           | 106        |
| 6.4. Verteilungen verbildlichen . . . . .                | 107        |
| 6.4.1. Verteilung: nominale Variable . . . . .           | 107        |
| 6.4.2. Aufgabe . . . . .                                 | 110        |
| 6.4.3. Lösung . . . . .                                  | 110        |
| 6.4.4. Verteilung: quantitative Variable . . . . .       | 111        |
| 6.4.5. Aufgabe . . . . .                                 | 113        |
| 6.4.6. Lösung . . . . .                                  | 113        |
| 6.4.7. Aufgabe . . . . .                                 | 116        |
| 6.4.8. Lösung . . . . .                                  | 116        |
| 6.4.9. Normalverteilung . . . . .                        | 116        |
| 6.5. Zusammenhänge verbildlichen . . . . .               | 117        |
| 6.5.1. Zusammenhang: nominale Variablen . . . . .        | 117        |
| 6.5.2. Aufgabe . . . . .                                 | 120        |
| 6.5.3. Lösung . . . . .                                  | 120        |
| 6.5.4. Zusammenhang: metrisch . . . . .                  | 121        |
| 6.5.5. Aufgabe . . . . .                                 | 126        |
| 6.5.6. Lösung . . . . .                                  | 126        |
| 6.6. Unterschiede verbildlichen . . . . .                | 127        |
| 6.6.1. Unterschied: nominale Variablen . . . . .         | 127        |
| 6.6.2. Unterschied: quantitative Variablen . . . . .     | 127        |
| 6.6.3. Aufgabe . . . . .                                 | 131        |
| 6.6.4. Lösung . . . . .                                  | 131        |
| 6.6.5. Aufgabe . . . . .                                 | 132        |
| 6.6.6. Lösung . . . . .                                  | 132        |
| 6.7. So lügt man mit Statistik . . . . .                 | 133        |
| 6.7.1. Achsen manipulieren . . . . .                     | 133        |

|   |            |
|---|------------|
| 6.7.2. Scheinkorrelation . . . . .                  | 134        |
| 6.8. Praxisbezug . . . . .                          | 135        |
| 6.9. Vertiefung . . . . .                           | 135        |
| 6.9.1. Schicke Diagramme . . . . .                  | 135        |
| 6.9.2. Farbwahl . . . . .                           | 138        |
| 6.10. Aufgaben . . . . .                            | 138        |
| 6.11. Literaturhinweise . . . . .                   | 139        |
| 6.12. Literatur . . . . .                           | 139        |
| <b>7. Punktmodelle 1</b>                            | <b>140</b> |
| 7.1. Lernsteuerung . . . . .                        | 140        |
| 7.1.1. Standort im Lernpfad . . . . .               | 140        |
| 7.1.2. Lernziele . . . . .                          | 140        |
| 7.1.3. Benötigte R-Pakete . . . . .                 | 140        |
| 7.1.4. Benötigte Daten . . . . .                    | 140        |
| 7.2. Mittelwert als Modell . . . . .                | 141        |
| 7.2.1. Frage . . . . .                              | 141        |
| 7.2.2. Antwort . . . . .                            | 141        |
| 7.2.3. Auswahl . . . . .                            | 145        |
| 7.2.4. Antwort . . . . .                            | 145        |
| 7.2.5. Der Mittelwert als lineares Modell . . . . . | 147        |
| 7.3. Median als Modell . . . . .                    | 148        |
| 7.3.1. Aufgabe . . . . .                            | 152        |
| 7.3.2. Lösung . . . . .                             | 152        |
| 7.4. Quantile . . . . .                             | 153        |
| 7.5. Lagemaße . . . . .                             | 154        |
| 7.4.1. Histogramm . . . . .                         | 155        |
| 7.4.2. Dichtediagramm . . . . .                     | 155        |
| 7.4.3. 25%-Schritte: Quartile . . . . .             | 156        |
| 7.4.4. 10%-Schritte: Dezile . . . . .               | 156        |
| 7.4.5. 1%-Schritte: Perzentile . . . . .            | 156        |
| 7.5.1. Gruppierte Lagemaße . . . . .                | 157        |
| 7.6. Wie man mit Statistik lügt . . . . .           | 159        |
| 7.7. Vertiefung . . . . .                           | 160        |
| 7.8. Aufgaben . . . . .                             | 160        |
| 7.9. Literaturhinweise . . . . .                    | 161        |
| 7.10. Literatur . . . . .                           | 161        |
| <b>8. Modellgüte</b>                                | <b>162</b> |
| 8.1. Lernsteuerung . . . . .                        | 162        |
| 8.1.1. Standort im Lernpfad . . . . .               | 162        |
| 8.1.2. Lernziele . . . . .                          | 162        |
| 8.1.3. Benötigte R-Pakete . . . . .                 | 162        |
| 8.1.4. Benötigte Daten . . . . .                    | 162        |
| 8.1.5. Zum Einstieg . . . . .                       | 162        |

|           |   |            |
|-----------|---|------------|
| 8.2.      | Warum Sie die Streuung Ihrer Daten kennen sollten . . . . . | 163        |
| 8.2.1.    | Die Schlankheitspille von Prof. Weiss-Ois . . . . .         | 163        |
| 8.2.2.    | Wie man seine Kuh über den Fluss bringt . . . . .           | 163        |
| 8.3.      | Woran erkennt man ein gutes Modell? . . . . .               | 164        |
| 8.4.      | Streuungsmaße . . . . .                                     | 166        |
| 8.4.1.    | Der mittlere Abweichungsbalken . . . . .                    | 166        |
| 8.4.2.    | Der Interquartilsabstand . . . . .                          | 168        |
| 8.4.5.    | Streuungsmaße für Normalverteilungen . . . . .              | 168        |
| 8.4.3.    | Histogramm . . . . .  | 169        |
| 8.4.4.    | Dichtediagramm . . . . .                                    | 169        |
| 8.4.6.    | Varianz . . . . .   | 170        |
| 8.4.7.    | Die Standardabweichung . . . . .                            | 173        |
| 8.5.      | Streuung als Modellfehler . . . . .                         | 174        |
| 8.6.      | $z$ -Transformation . . . . .                               | 175        |
| 8.7.      | Fazit . . . . .   | 178        |
| 8.8.      | Aufgaben . . . . .  | 178        |
| 8.8.1.    | Datenwerk . . . . .   | 178        |
| 8.8.2.    | Aufgabe . . . . .   | 179        |
| 8.8.3.    | Lösung: Daten importieren . . . . .                         | 179        |
| 8.8.4.    | Lösung: Daten aufbereiten . . . . .                         | 179        |
| 8.8.5.    | Fallstudie zur Lebenszufriedenheit . . . . .                | 179        |
| 8.9.      | Literaturhinweise . . . . .                                 | 180        |
| 8.10.     | Literatur . . . . .   | 180        |
| <b>9.</b> | <b>Punktmodelle 2</b>                                       | <b>181</b> |
| 9.1.      | Lernsteuerung . . . . .                                     | 181        |
| 9.1.1.    | Standort im Lernpfad . . . . .                              | 181        |
| 9.1.2.    | Lernziele . . . . .   | 181        |
| 9.1.3.    | Benötigte R-Pakete . . . . .                                | 181        |
| 9.1.4.    | Benötigte Daten . . . . .                                   | 181        |
| 9.1.5.    | Zum Einstieg . . . . .                                      | 182        |
| 9.2.      | Zusammenfassen zum Zusammenhang . . . . .                   | 182        |
| 9.2.1.    | Beispiele für Zusammenhänge . . . . .                       | 182        |
| 9.3.      | Abweichungsrechtecke . . . . .                              | 183        |
| 9.3.1.    | Noten und Abweichungsrechtecke . . . . .                    | 183        |
| 9.3.2.    | Kovarianz . . . . .   | 185        |
| 9.3.3.    | Die Kovarianz ist schwer zu interpretieren . . . . .        | 188        |
| 9.4.      | Korrelation . . . . .                                       | 188        |
| 9.4.1.    | Korrelation als mittleres $z$ -Produkt . . . . .            | 188        |
| 9.4.2.    | Korrelation mit R berechnen . . . . .                       | 189        |
| 9.4.3.    | Korrelation $\neq$ Kausation . . . . .                      | 190        |
| 9.4.4.    | Korrelation misst nur linearen Zusammenhang . . . . .       | 191        |
| 9.5.      | Wie man mit Statistik lügt . . . . .                        | 191        |
| 9.5.1.    | Range-Restriktion . . . . .                                 | 191        |
| 9.6.      | Fallbeispiel . . . . .                                      | 192        |
| 9.7.      | Vertiefung . . . . .  | 195        |

|   |            |
|---|------------|
| 9.8. Aufgaben . . . . .   | 196        |
| 9.9. Halbzeitquiz . . . . .   | 196        |
| 9.10. Fallstudien . . . . .   | 196        |
| 9.11. Literaturhinweise . . . . .                                     | 197        |
| 9.12. Literatur . . . . .   | 197        |
| <b>10. Geradenmodelle 1</b>   | <b>198</b> |
| 10.1. Lernsteuerung . . . . .   | 198        |
| 10.1.1. Standort im Lernpfad . . . . .                                | 198        |
| 10.1.2. Lernziele . . . . .   | 198        |
| 10.1.3. Benötigte R-Pakete . . . . .                                  | 198        |
| 10.1.4. Benötigte Daten . . . . .                                     | 198        |
| 10.2. Vorhersagen . . . . .   | 199        |
| 10.2.1. Vorhersagen ohne Prädiktor . . . . .                          | 199        |
| 10.2.2. Nullmodell (Punktmodell) . . . . .                            | 200        |
| 10.2.3. Vorhersagen mit Prädiktor . . . . .                           | 201        |
| 10.3. Geradenmodelle . . . . .  | 202        |
| 10.3.1. Achsenabschnitt und Steigung definieren eine Gerade . . . . . | 202        |
| 10.3.2. Spezifikation eines Geradenmodells . . . . .                  | 205        |
| 10.3.3. Vorhersagefehler . . . . .                                    | 207        |
| 10.3.4. Berechnung der Modellkoeffizienten . . . . .                  | 208        |
| 10.4. R-Quadrat als Maß der Modellgüte . . . . .                      | 208        |
| 10.5. Interpretation eines Regressionsmodells . . . . .               | 210        |
| 10.5.1. Modellgüte . . . . .  | 210        |
| 10.5.2. Koeffizienten . . . . .                                       | 210        |
| 10.6. Wie man mit Statistik lügt . . . . .                            | 211        |
| 10.7. Fallbeispiel MarioKart . . . . .                                | 212        |
| 10.7.1. Der Datenwahrsager legt los . . . . .                         | 212        |
| 10.7.2. Vertiefung . . . . .  | 213        |
| 10.8. Fallstudie Immobilienpreise . . . . .                           | 215        |
| 10.8.1. Hintergrund . . . . .   | 215        |
| 10.8.2. Benötigte R-Pakete . . . . .                                  | 215        |
| 10.8.3. Daten . . . . .   | 216        |
| 10.8.4. Prognosedatei . . . . .                                       | 216        |
| 10.8.5. Daten importieren von der Festplatte . . . . .                | 217        |
| 10.8.6. Ein erster Blick in die Daten . . . . .                       | 217        |
| 10.8.7. Ein erstes Vorhersagemodell . . . . .                         | 218        |
| 10.8.8. Vorhersagen im Test-Datensatz mit m2 . . . . .                | 221        |
| 10.8.9. Einreichen! . . . . .   | 221        |
| 10.8.10. Fazit . . . . .  | 222        |
| 10.9. Aufgaben . . . . .  | 222        |
| 10.10. Literaturhinweise . . . . .                                    | 223        |
| 10.11. Literatur . . . . .  | 223        |

|  |            |
|--|------------|
| <b>11. Geradenmodelle 2</b>  | <b>224</b> |
| 11.1. Lernsteuerung . . . . .  | 224        |
| 11.1.1. Standort im Lernpfad . . . . .   | 224        |
| 11.1.2. Lernziele . . . . .  | 224        |
| 11.1.3. Benötigte R-Pakete . . . . .   | 224        |
| 11.1.4. Benötigte Daten . . . . .  | 224        |
| 11.2. Forschungsbezug: Gläserne Kunden . . . . .                               | 225        |
| 11.3. Wetter in Deutschland . . . . .  | 225        |
| 11.3.1. metrische UV . . . . .   | 227        |
| 11.3.2. UV zentrieren . . . . .  | 228        |
| 11.3.3. Binäre UV . . . . .  | 231        |
| 11.3.4. Nominale UV . . . . .  | 235        |
| 11.3.5. Binäre plus metrische UV . . . . .                                     | 239        |
| 11.3.6. Interaktion . . . . .  | 243        |
| 11.4. Modelle mit vielen UV . . . . .  | 245        |
| 11.4.1. Zwei metrische UV . . . . .  | 245        |
| 11.4.2. Viele UV ins Modell? . . . . .   | 246        |
| 11.5. Fallbeispiel zur Prognose . . . . .                                      | 247        |
| 11.5.1. Modell “all-in” . . . . .  | 247        |
| 11.5.2. Modell “all-in”, ohne Titelspalte . . . . .                            | 249        |
| 11.6. Vertiefung: Das Aufteilen Ihrer Daten . . . . .                          | 252        |
| 11.6.1. Analyse- und Assessment-Sample . . . . .                               | 252        |
| 11.6.2. Train- vs. Test-Sample . . . . .                                       | 253        |
| 11.7. Praxisbezug . . . . .  | 254        |
| 11.8. Wie man mit Statistik lügt . . . . .                                     | 254        |
| 11.8.1. Pinguine drehen durch . . . . .  | 254        |
| 11.8.2. Analyse 1: Gesamtdaten . . . . .                                       | 255        |
| 11.8.3. Analyse 2: Aufteilung in Arten (Gruppen) . . . . .                     | 256        |
| 11.8.4. Vorsicht bei der Interpretation von Regressionskoeffizienten . . . . . | 257        |
| 11.9. Fazit . . . . .  | 258        |
| 11.10 Fallstudien . . . . .  | 259        |
| 11.10.1 New Yorker Flugverspätungen 2023 . . . . .                             | 259        |
| 11.10.2 Filmerlöse . . . . .   | 259        |
| 11.11 Vertiefung . . . . .   | 259        |
| 11.12 Aufgaben . . . . .   | 260        |
| 11.13 Literaturhinweise . . . . .  | 260        |
| 11.14 Literatur . . . . .  | 260        |
| <b>IV. Abschluss</b>   | <b>261</b> |
| <b>12. Abschluss</b>   | <b>262</b> |
| 12.1. Lernsteuerung . . . . .  | 262        |
| 12.1.1. Standort im Lernpfad . . . . .   | 262        |
| 12.1.2. Lernziele . . . . .  | 262        |
| 12.1.3. Benötigte R-Pakete . . . . .   | 262        |

|  |            |
|--|------------|
| 12.1.4. Benötigte Daten . . . . .          | 262        |
| 12.2. Herzlichen Glückwunsch! . . . . .    | 262        |
| 12.3. Wie geht's weiter? . . . . .         | 263        |
| 12.4. Aufgabensammlungen . . . . .         | 263        |
| 12.5. Quizze . . . . .                     | 263        |
| 12.6. Fallstudien . . . . .                | 264        |
| 12.6.1. Datenvisualisierung . . . . .      | 264        |
| 12.6.2. Explorative Datenanalyse . . . . . | 264        |
| 12.6.3. Lineare Modelle . . . . .          | 265        |
| 12.7. FAQ . . . . .                        | 266        |
| 12.7.1. SD berechnen . . . . .             | 266        |
| 12.7.2. count vs. filter . . . . .         | 266        |
| 12.7.3. 1000 . . . . .                     | 267        |
| 12.8. Literaturhinweise . . . . .          | 267        |
| <b>13. Definitionen</b>                    | <b>268</b> |
| <b>Literatur</b>                           | <b>270</b> |

# 1. Willkommen!

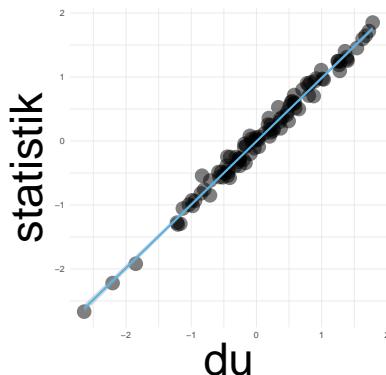


Abbildung 1.1.: Statistik und Du: Guter Fit!

## 1.1. Freude am Lernen

Dieses Buch führt in die Statistik ein; es soll Freude am Lernen bereiten und konzentriert sich auf ein Thema: Vorhersagen mittels moderner statistischen Methoden. Es wendet sich an Studierende ohne Vorkenntnisse in Statistik. Es gibt schon viele Statistikbücher auf dieser Welt, braucht es da noch eines? Kritische studentische Stimmen würden vielleicht anmerken, dass schon ein Statistikbuch eines zu viel wäre. Ja, es gibt viele Statistikbücher, aber (meines Wissens) keines in deutscher Sprache, dass Freude beim Lernen vermittelt, sich auf statistische Vohersage-Modellierung konzentriert und moderne Werkzeuge einsetzt. Freude am Lernen, beim Angstgegner Statistik, wie soll das gehen? Viele Verständnisschwierigkeiten röhren daher, dass die Lehrbücher kompliziert geschrieben sind. Solcher Schreibweise liegt wohl die Überlegung zugrunde, dass die Konzepte präzise und umfangreich erläutert sein müssen. Meiner Ansicht nach wird da das Ziel mit dem Weg verwechselt: Am Anfang darf eine Erklärung ruhig etwas grober und detailärmer sein, überblickt man die Materie einigermaßen, kann man sich mit den Details vertraut machen, was Präzision und Tiefe verlangt. Darüber hinaus verwendet das Buch eine lockere Sprache, das Leben ist schon ernst genug. Für einen Komfort beim Lesen wurde gesorgt: Lernziele, Definitionen, Beispiele, Übungen, Hinweise, Fehlerquellen, Tipps, Literatur, Verweise und mehr wurden hervorgehoben. An Erklärbildern wurde nicht gespart.

Der Inhalt des Buches ist ganz auf statistische Modelle zur Vorhersage ausgerichtet. "Statistische Modelle" ist ein sperriger Begriff, aber er sagt nur, dass es darum geht, fachliche Fragen in statistisch greifbare Bausteine zu gießen. Ein Beispiel: Studentin Anna fragt sich, ob sie die Prüfung besteht, wenn Sie 42 Stunden büffelt? Student Bert meint, dass motivierte Studis am meisten vom Lernen profitieren. Studentin Carla ist hingegen überzeugt, dass Lernen nix bringt, sondern dass die Intelligenz allein für den Prüfungserfolg verantwortlich sei. Damit haben wir drei (noch recht unpräzise wissenschaftliche)

Modelle. Die Statistik hat nun die Aufgabe, präzise Antworten zu liefern, wofür Zahlen hilfreich sind. Wenn Anna, Bert und Carla ihre Überlegungen fachlich präzisieren und dann in statistische Sprache übersetzen, können wir mit Antworten rechnen, manchmal sogar präzise. Was nicht heißt, dass diese Antworten immer richtig oder nützlich sind. Tja, das Leben ist nicht leicht.

Mit Blick auf den Spagat zwischen Wissen und Können irrt das Buch (sein Autor) zugunsten der Seite des Könnens. Ich wollte lieber befähigen, spannende Probleme zu lösen, als tiefen theoretischen Einblick zu vermitteln. Meine Hoffnung ist, dass die Freude am Können beflügelt, sich im nächsten Schritt tiefer mit der Materie zu beschäftigen. Ist es nicht auch so im Alltag [([lovett\\_applying\\_2000?](#))]? Was Freude macht, wo sich Erfolge einstellen, dort arbeiten wir gerne weiter.

Da sich das Buch auf ein Thema, Modellierung, konzentriert, bleiben andere Themen außen vor, vor allem Inferenzstatistik. Vielleicht freut sich die eine oder der andere, von diesem Thema verschont zu sein. Ich denke, dass Modellierung für die Forschung und für die Praxis ein zentraler Gedanke ist; für zwei große Themen erscheint mir dieses Buch zu eng.

Wenn Sie Fragen oder Feedback haben, bin ich für Ihre Hinweise dankbar. Stellen Sie sie gerne hier ein: <https://github.com/sebastiansauer/statistik1/issues>.

Ich wünsche Ihnen viel Freude und Erfolg beim Statistik lernen!

Ihr

Sebastian Sauer

## 1.2. Es geht um Ihren Lernerfolg

Meister Yoda rät: Lesen Sie die folgenden Hinweise (Abbildung 1.2).

Quelle: [Imgflip Memengenerator](#)

### 1.2.1. Lernziele

- Die Studenten sind mit wesentlichen Methoden der explorativen Datenanalyse vertraut und können diese selbstständig anwenden.
- Die Studenten können gängige Forschungsfragen in lineare Modelle übersetzen, diese auf echte Datensätze anwenden und die Ergebnisse interpretieren.

Kurz gesagt: Das ist ein Grundkurs in Daten zähmen.

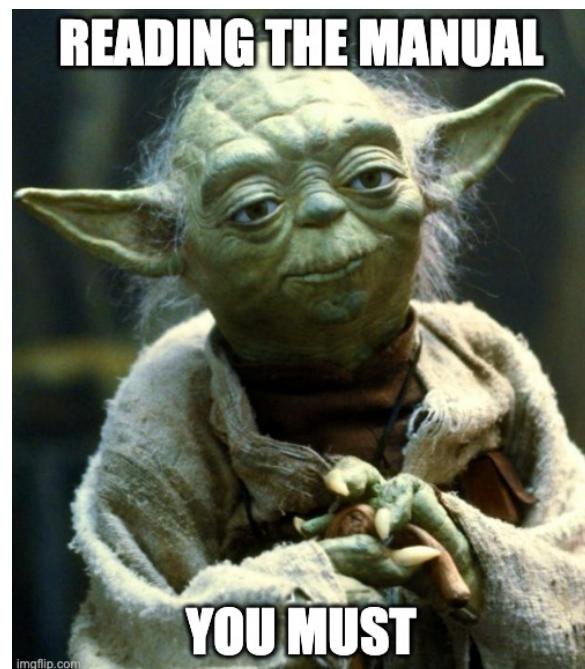


Abbildung 1.2.: Lesen Sie die folgenden Hinweise im eigenen Interesse

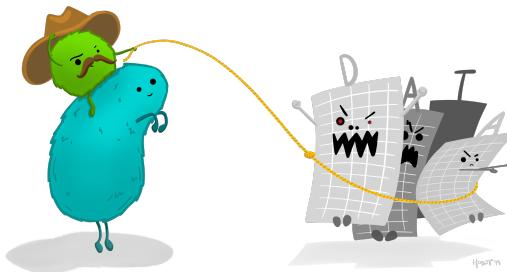


Abbildung 1.3.: Daten zähmen

Bildquelle: Allison Horst, CC-BY

### 1.2.2. Was lerne ich hier und wozu ist das gut?

*Was lerne ich hier?*

Sie lernen das *Handwerk der Datenanalyse* mit einem Schwerpunkt auf Vorhersage. Anders gesagt: Sie lernen, *Daten aufzubereiten* und aus Daten *Vorhersagen* abzuleiten. Zum Beispiel: Kommt ein Student zu Ihnen und sagt "Ich habe 42 Stunden für die Klausur gelernt, welche Note kann ich in der Klausur erwarten?". Darauf Ihre Antwort: "Auf Basis meiner Daten und meines Modells müsstest du

eine 2.7 schreiben!”<sup>1</sup>. Außerdem lernen Sie, wie man die Güte einer Vorhersage auf Stichhaltigkeit prüft. Denn Vorhersagen kann man ja in jeder Eckkneipe oder beim Wahrsager bekommen. Wir wollen aber belastbare Vorhersagen und zumindest wissen, wie gut die Vorhersagen (von jemanden) bisher waren.

*Warum ist das wichtig?*

Wir wollen nicht auf Leuten vertrauen, die behaupten, sie wüssten, was für uns richtig und gut ist. Wir wollen selber die Fakten prüfen können.

*Wozu brauche ich das im Job?*

Datenanalyse spielt bereits heute in vielen Berufen eine Rolle. Tendenz stark zunehmend.

*Wozu brauche ich das im weiterem Studium?*

In Forschungsarbeiten (wie in empirischen Forschungsprojekten, etwa in der Abschlussarbeit) ist es üblich, statistische Ergebnisse hinsichtlich quantitativ zu analysieren.

*Ist Statistik nicht sehr abstrakt?*

Der Schwerpunkt dieses Kurses liegt auf Anwenden und Tun; ähnlich dem Erlernen eines Handwerks. Theorien und Abstraktionen stehen nur am Rand.

*Gibt es auch gute Jobs, wenn man sich mit Daten auskennt?*

Das Forum (2020) berichtet zu den “Top 20 job roles in increasing and decreasing demand across industries” (S. 30, Abb. 22):

1. Data Analysts und Scientists
2. AI and Machine Learning Specialists
3. Big Data Specialists

### **1.2.3. Was ist hier das Erfolgsgeheimnis?**

! Wichtig

*Dran bleiben* ist der Schlüssel zum Erfolg. Üben Sie regelmäßig. Geben Sie bei Schwierigkeiten nicht auf.



### **1.2.4. Motivieren Sie mich!**

Schauen Sie sich das Video mit einer [Ansprache zur Motivation](#) an.<sup>2</sup>

<sup>1</sup>Darauf die Studentin: “Hpmf.”

<sup>2</sup><https://youtu.be/jtNlzpcPr5Y>

### 1.2.5. Voraussetzungen

Um von diesem Kurs am besten zu profitieren, sollten Sie Folgendes mitbringen:

- Bereitschaft, Neues zu lernen
- Bereitschaft, nicht gleich aufzugeben
- Kenntnis grundlegender Methoden wissenschaftlichen Arbeitens

Was Sie *nicht* brauchen, sind besondere Mathe- oder Statistik-Vorkenntnisse.

### 1.2.6. Überblick

Abb. Abbildung 1.4 gibt einen Überblick über den Verlauf und die Inhalte des Buches. Das Diagramm hilft Ihnen zu verorten, wo welches Thema im Gesamtzusammenhang steht.

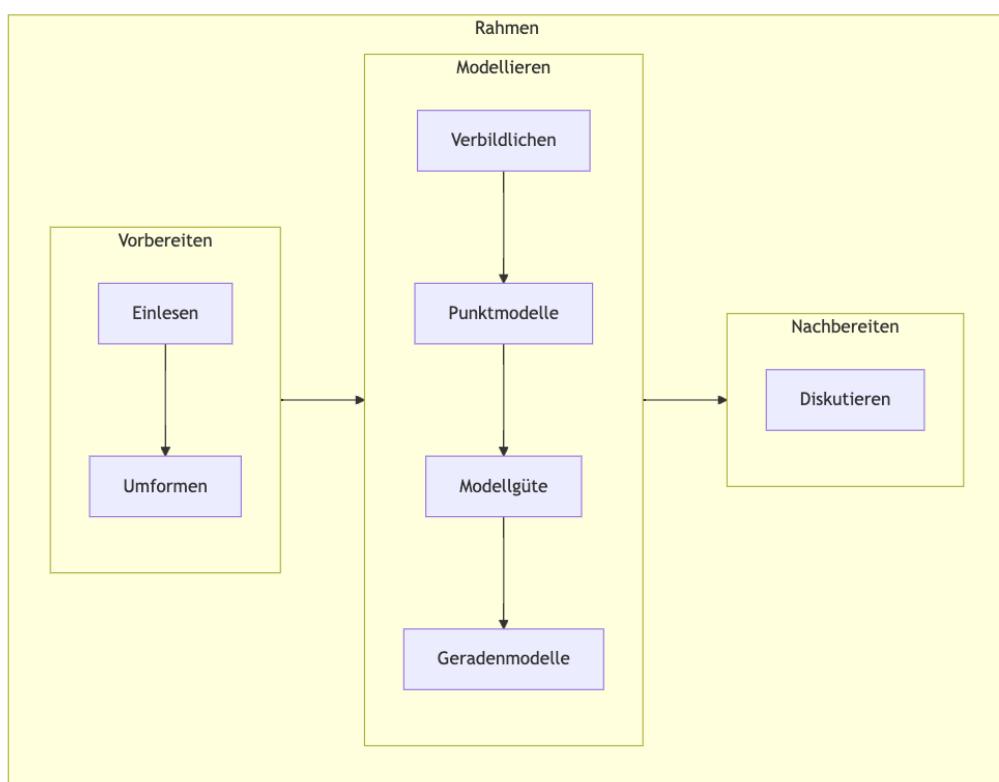


Abbildung 1.4.: Überblick über den Inhalt und Verlauf des Buches

Das Diagramm zeigt auch den Ablauf einer typischen Datenanalyse. Natürlich kann man sich auch andere sinnvolle Darstellungen dieses Ablaufs vorstellen.

### 1.3. Software: R

Sie benötigen R, RStudio und einige R-Pakete für diesen Kurs. Hier finden Sie *Installationshinweise*.<sup>3</sup>

Dieses Buch enthält “mittel” viel R. Auf fortgeschrittene R-Techniken wurde aber komplett verzichtet. Dem einen Anfänger oder der anderen Anfängerin mag es dennoch “viel Code” erscheinen. Es wäre ja auch möglich gewesen, auf R zu verzichten und stattdessen eine “Klick-Software” zu verwenden. **JASP** oder **Jamovi** sind Beispiele für tolle Software aus dieser Kategorie. Ich glaube aber, der Verzicht auf eine Skriptsprache (R) wäre ein schlechter Dienst an den Studentis. Mit Blick auf eine “High-Tech-Zukunft” sollte man zumindest mit etwas Computer-Code vertraut sein. Auf Computercode zu verzichten erschien mir daher fahrlässig für die “Zukunftsfestigkeit” der Ausbildung.

### 1.4. Zum Autor

Nähere Hinweise zum Autor dieses Buch, Sebastian Sauer, finden Sie hier.<sup>4</sup> Dort gibt es auch einen Überblick über weitere Bücher des Autors zum Themenkreis Datenanalyse.<sup>5</sup>

### 1.5. Nomenklatur

#### 1.5.1. Griechische Buchstaben

In diesem Buch werden ein paar (wenige) griechische Buchstaben verwendet, die in der Statistik üblich sind. Häufig werden *griechische* Buchstaben verwendet, um eine Grundgesamtheit (Population) zu beschreiben (die meistens unbekannt ist). Lateinische (“normale”) Buchstaben werden demgegenüber verwendet, um eine Stichprobe (Datensatz, vorliegende Daten) zu beschreiben. Tabelle 1.1 stellt diese Buchstaben zusammen mit ihrer Aussprache und Bedeutung vor.

Tabelle 1.1.: Griechische Buchstaben, die in diesem Buch verwendet werden.

| Zeichen  | Aussprache | Buchstabe | Bedeutung in der Statistik |
|----------|------------|-----------|----------------------------|
| $\beta$  | beta       | b         | Regressionskoeffizient     |
| $\mu$    | mü         | m         | Mittelwert                 |
| $\sigma$ | sigma      | s         | Streuung                   |
| $\Sigma$ | Sigma      | S         | Summenzeichen              |
| $\rho$   | rho        | r         | Korrelation (nach Pearson) |

Mehr griechische Buchstaben finden sich z.B. in Wikipedia.<sup>6</sup>

<sup>3</sup><https://hinweisbuch.netlify.app/hinweise-software>

<sup>4</sup><https://sebastiansauer-academic.netlify.app/>

<sup>5</sup><https://sebastiansauer-academic.netlify.app/#ebooks>

<sup>6</sup>[https://de.wikipedia.org/wiki/Griechisches\\_Alphabet](https://de.wikipedia.org/wiki/Griechisches_Alphabet)

## 1.6. Zitation

Bitte zitieren Sie dieses Buch wie folgt:

Sauer, S. (2024). *Statistik1*. <https://statistik1.netlify.app/>

Hier sind die maschinenlesbaren Zitationsinfos (Bibtex-Format), die Sie in Ihre Literatursoftware importieren können:

```
@book{sauer_statistik1,
  title = {Statistik1},
  rights = {CC-BY-NC},
  url = {https://statistik1.netlify.app/},
  author = {Sauer, Sebastian},
  date = {2024},
}
```

Hier ist die DOI:

[10.5281/zenodo.10082517](https://doi.org/10.5281/zenodo.10082517)

## 1.7. Reproduzierbarkeit

Die verwendeten R-Pakete sind mit `renv` dokumentiert.<sup>7</sup> Der Quellcode ist [in diesem Github-Repo](#) dokumentiert.<sup>8</sup>

Dieses Dokument wurde erzeugt am/um: 2024-08-30 15:30:28.

---

<sup>7</sup><https://rstudio.github.io/renv/index.html>

<sup>8</sup><https://github.com/sebastiansauer/statistik1>

**Teil I.**

**Organisatorisches**

## 2. Prüfung

Die folgenden Hinweise sind dem [Hinweisbuch](#) des Autors entnommen. Lesen Sie auch die übrigen Hinweise dort.<sup>1</sup>

### 2.1. Prüfungleistung

Die Prüfungsleistung besteht aus einer Hauptleistung (keine Bonusleistung).

Die Hauptleistung besteht aus einer Projektarbeit im Form eines *Prognosewettbewerbs*.

### 2.2. Zum Prognosewettbewerb

Im [Hinweisbuch](#) finden Sie [Hinweise zur Prüfung](#).<sup>2</sup>

### 2.3. Prüfungsrelevanter Stoff

Beachten Sie die [Hinweise zum prüfungsrelevanten Stoff](#).<sup>3</sup>

### 2.4. Wie kann ich mich auf die Prüfung vorbereiten?

[Hier](#) finden Sie Hinweise zur Prüfungsvorbereitung.<sup>4</sup>

---

<sup>1</sup><https://hinweisbuch.netlify.app/>

<sup>2</sup><https://hinweisbuch.netlify.app/080-hinweise-pruefung-prognosewettbewerb-frame>

<sup>3</sup><https://hinweisbuch.netlify.app/010-hinweise-pruefung-allgemein-frame#pr%C3%BCfungsrelevanter-stoff>

<sup>4</sup><https://hinweisbuch.netlify.app/150-hinweise-pruefungsvorbereitung-frame>

## **2.5. Allgemeine Prüfungscheinweise**

Die folgenden Hinweise gelten grundsätzlich, d.h. soweit nicht anders in der jeweiligen Prüfung bzw. der jeweiligen Aufgabe angegeben. Nichtbeachten von Prüfungshinweisen kann zu Punkteabzug oder Nichtbestehen führen. Lesen Sie sich diese Hinweise im eigenen Interesse sorgfältig durch. Kenntnis dieser Hinweise wird bei der Begutachtung vorausgesetzt.

Für eine einfachere Kommunikation kontaktieren Sie mich per E-Mail bei Fragen, die nur Sie betreffen. Bei Fragen von allgemeinem Interesse (z.B. "Bis wann müssen wir die Arbeit abgeben?") nutzen Sie bitte (sofern verfügbar) das Kursforum, damit die Kommilitonen auch von dem Austausch profitieren.

Beachten Sie die [allgemeinen Prüfungshinweise](#).<sup>5</sup>

## **2.6. Lieblingsfehler**

Vermeiden Sie diese [häufigen Fehler im Prognosewettbewerb](#).<sup>6</sup>

## **2.7. Fazit**

✿✿✿VIEL ERFOLG!✿✿✿

---

<sup>5</sup><https://hinweisbuch.netlify.app/010-hinweise-pruefung-allgemein-frame>

<sup>6</sup><https://hinweisbuch.netlify.app/170-beispiele-fehler-prognosewettbewerb-frame>

**Teil II.**

## **Vorbereiten**

# 3. Rahmen

## 3.1. Lernsteuerung

### 3.1.1. Standort im Lernpfad

Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

Abbildung 3.1 zeigt, dass unser Vorgehen in diesem Buch einem Fließband gleicht: Schritt für Schritt, in der richtigen Reihenfolge, vom Anfang bis Ende, erarbeiten wir unser “Datenprodukt”.<sup>1</sup>

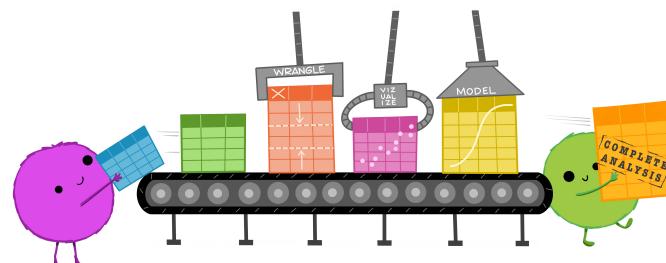


Abbildung 3.1.: Datenanalyse als eine Abfolge am Fließband

### 3.1.2. Lernziele

- Sie können eine Definition von Statistik wiedergeben.
- Sie können eine Definition von Daten wiedergeben.
- Sie können den Begriff Tidy-Daten erläutern.
- Sie können Beispiele für verschiedene Skalenniveaus nennen.

<sup>1</sup>Quelle: Allison Horst, CC-by, <https://github.com/allisonhorst/stats-illustrations>

### 3.1.3. Einstieg

**Übungsaufgabe 3.1** (Hallo, Statistik). Gehen Sie in eine kleine Gruppe zusammen (3-4 Personen). Stellen Sie sich anhand der Schlagworte einander vor:

1. Name
2. (wissenschaftliche) Interessen
3. Erwartung an diesen Kurs

**Übungsaufgabe 3.2** (Frag jetzt). Die Lehrkraft stellt Ihnen ein Forum zur Verfügung, auf dem Sie *anonym* Fragen an die Lehrkraft richten können (z.B. auf [frag.jetzt](#)).

Stellen Sie dort Ihre Fragen ein; voten Sie die Fragen Ihrer Kommilitonis auf oder ab. Die Lehrkraft beantwortet dann die Fragen mit den meisten Upvotes.

### 3.1.4. Erfolsgrezept

Ihren Lernerfolg kann man als von drei Faktoren abhängig betrachten: 1) Ihrer Lehrkraft, 2) Ihrer Mitarbeit im Unterricht und 3) Ihrem Eigenstudium zuhause (Vor- bzw. Nachbereitung des Unterrichts), s. Abbildung 3.2.



Abbildung 3.2.: Ihr Lernerfolg besteht aus drei Komponenten: Der Lehrkraft, Ihrer Konzentration im Unterricht und Ihrer Vor- bzw. Nachbereitung zuhause.

Eine gute Lehrkraft ist wie der Funke, der eine (Lern-)Flamme entzündet. Aber es braucht Brennmaterial, einen Holzscheit, das ist Ihre Konzentration im Unterricht. Ein Holzscheit alleine ist aber nicht nachhaltig, es braucht viel mehr Brennmaterial, um das Feuer am Leben zu halten: Ihr Eigenstudium, das stetige, selbständige Beschäftigen mit der Materie.

## 3.2. Was ist Statistik und wozu ist sie gut?

**Übungsaufgabe 3.3** (Nutzen von Statistik). ♀ Fragen Sie ChatGPT, was der Nutzen von Statistik ist. Diskutieren Sie die Antwort.

#### **i Hinweis**

In diesem Buch werden die Begriffe *Statistik*, *Datenanalyse* und *Data Science* synonym verwendet.

Hm. Vielleicht fragen wir mal ChatGPT, was es unter Statistik versteht:

- Was ist Statistik? Bitte definiere verständlich.
- Statistik ist eine Wissenschaft, die sich mit der Sammlung, Analyse, Interpretation, Präsentation und Übertragung von Daten beschäftigt. Ziel ist es, Muster und Trends in den Daten zu erkennen und verlässliche Schlussfolgerungen über eine größere Gruppe (z.B. Bevölkerung) auf der Basis einer Stichprobe (Teilmenge) zu treffen. Die Statistik hilft uns bei der Beantwortung von Fragen, indem sie uns dabei hilft, die Unsicherheit in den Daten zu messen und zu verstehen<sup>2</sup>.

Es gibt mehrere Definition von Statistik; hier ist eine.

**Definition 3.1** (Statistik). Statistik fasst Daten zusammen, um wesentliche Informationen den Daten zu entnehmen und beschreibt die Ungewissheit unserer Schlüsse (Poldrack 2023); Kaplan (2009) .□

Betrachten wir die zwei Bestimmungsstücke einer Definition von Statistik genauer:

1. Daten zusammenfassen
2. Ungewissheit beschreiben

### 3.2.1. Daten zusammenfassen

Abbildung 3.3 verdeutlicht das Prinzip des Zusammenfassens von Daten. Anschaulich gesprochen: Eine Menge von Zahlen wird zu einer einzelnen Zahl “zusammengedampt”. Eine einzelne Zahl ist wesentlich besser zu verstehen als eine große Menge von Zahlen. Bei vielen Zahlen würde man den Überblick verlieren.

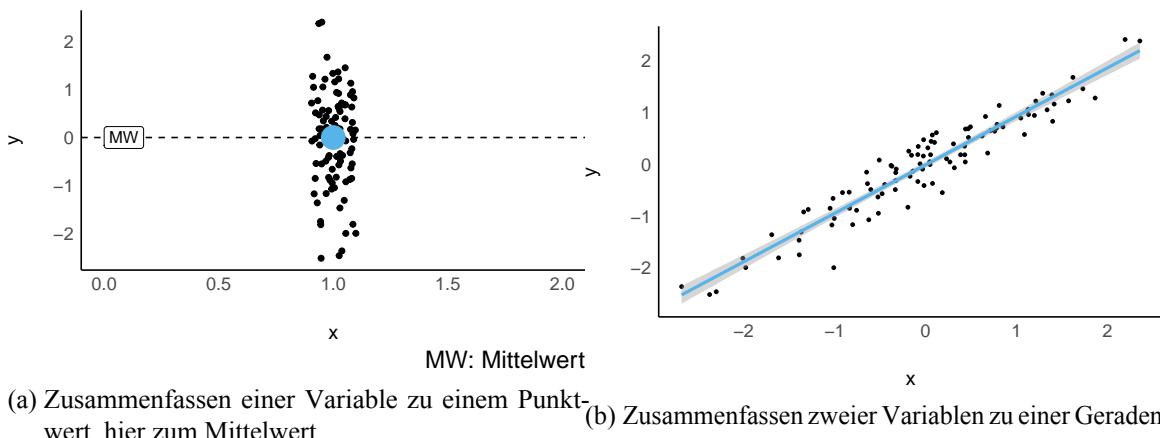


Abbildung 3.3.: Daten zusammenfassen

---

<sup>2</sup>Release 2023-Jan

### 3.2.2. Unterschiedlichkeit messen

Eine allgegenwärtige Tatsache ist, dass die Dinge der Welt sich unterscheiden, etwa, dass Exemplare einer Gattung sich unterscheiden. So sind nicht alle Menschen gleich groß, nicht alle Bücher gleich lang oder nicht alle Tage gleich warm.

Ein zentrales Vorgehen bei statistischen Analysen ist es, die *Unterschiedlichkeit der Dinge* zu beschreiben, präziser gesagt: die *Variation zu quantifizieren*. Betrachten wir dazu das Beispiel in s. Abbildung 3.4.

#### Variation in der Körpergröße

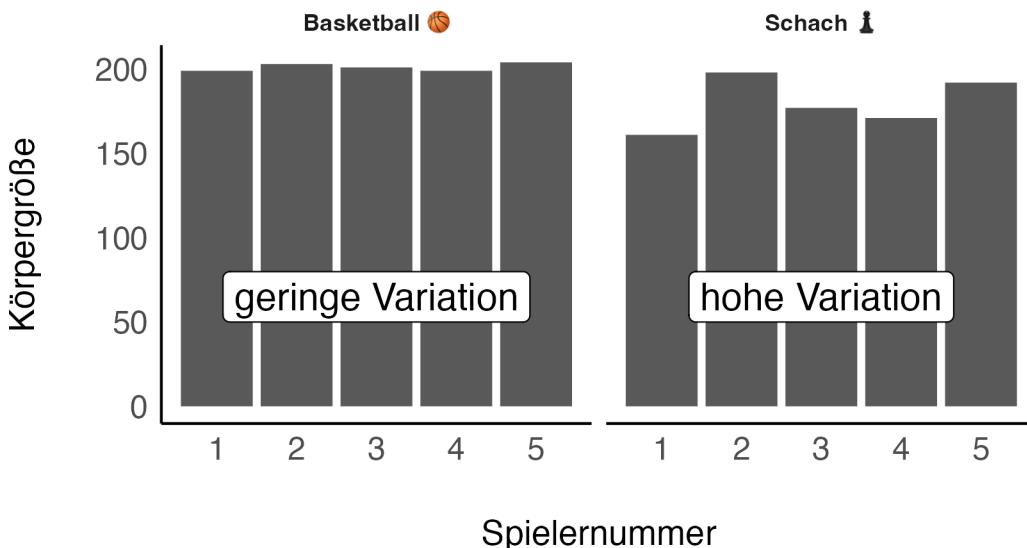


Abbildung 3.4.: Wenig Variation in der Körpergröße bei den Basketballern. Alles lange Kerle. Viel Variation bei den Schachspielern: Manche sind klein, ander groß.

Bei den Basketballern gibt es *geringe* Variation in der Körpergröße - alle sind groß, ähnlich groß. Bei den Schachspielern gibt es (im Verhältnis) *hohe* Variation: Einige Personen sind groß, andere klein.

Die Variation (auch “Variabilität” genannt) kann man auch gut so darstellen wie in s. Abbildung 3.5 gezeigt.

Eine *Abweichung* (auch *Residuum*) genannt, zeigt hier die Differenz von Mittelwert und dem Wert der Körpergröße bei der jeweiligen Person. Wenn wir allgemein von einer Person  $i$  sprechen, Das Merkmal *Körpergröße* mit  $X$  bezeichnen und den Mittelwert der Körpergröße als  $\bar{x}$  (“x quer”), dann können wir knapp und präzise das Residuum der  $i$ -ten Person mit  $r_i$  bezeichnen und entsprechend definieren.

**Definition 3.2** (Residuum). Das Residuum des Merkmals  $X$  der  $i$ -ten Beobachtung ist definiert als die Differenz vom Wert  $x_i$  und einem Referenzwert, etwa dem Mittelwert,  $\bar{x}$ :

$$r_i = x_i - \bar{x}. \square$$

## Abweichung vom Mittelwert der Körpergröße pro Team

Basketball: Wenig Variation; Schach: Viel Variation

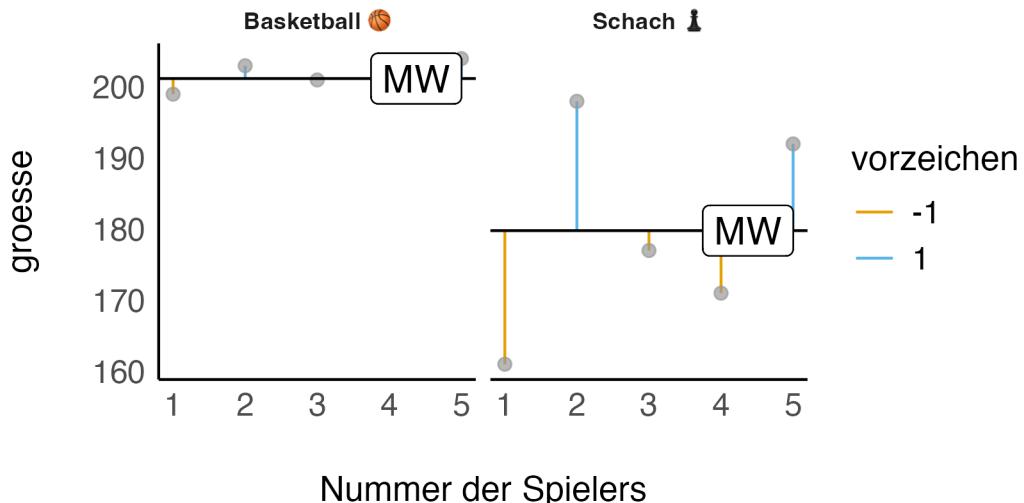


Abbildung 3.5.: Die Abweichungen der einzelnen Personen von der mittleren Körpergröße ihres Teams

### 3.3. Was ist das Ziel Ihrer Analyse?

#### 3.3.1. Arten von Zielen



Abbildung 3.6.: Zielarten einer Datenanalyse

Beispiele für die einzelnen Zielarten der Datenanalyse:

- *Beschreiben*: "Wie groß ist der Gender-Paygap in der Branche X im Zeitraum Y?"
- *Vorhersagen*: Wenn ich 100 Stunden auf die Statistikklausur lernen, welche Note kann ich dann erwarten?
- *Erklären*: Wie viel bringt mir das Lernen auf die Statistikklausur?

**Übungsaufgabe 3.4.** Benennen Sie Beispiele für die drei Zielarten von Datenanalysen! □

### 3.3.2. Forschungsfrage

Eine Forschungsfrage ist die Leitfrage Ihrer Analyse. Sie definiert, was Sie herausfinden wollen. Häufig sind Forschungsfragen so aufgebaut:

Hat X einen Einfluss auf Y?

Eine Forschungsfrage weist häufig folgende Struktur auf, s. Abbildung 3.7.

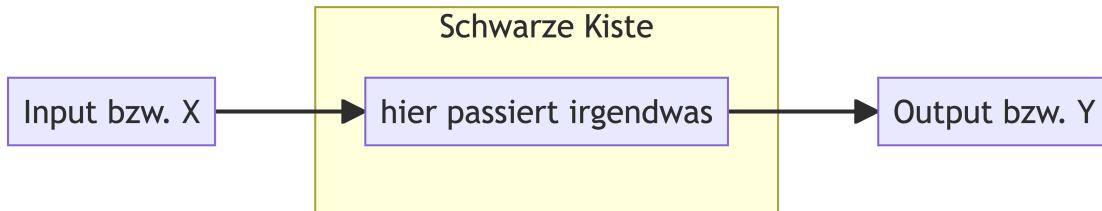


Abbildung 3.7.: Struktur einer Forschungsfrage

**Beispiel 3.1** (Forschungsfrage 1).

Hat Lernen (X) einen Einfluss auf den Prüfungserfolg (Y)? Verringert Joggen (X) die Menge des Hüftgolds (Y)? Um welchen Betrag erhöht sich der Umsatz (Y), wenn wir 1000€ mehr Werbung ausgeben? (X)□

**Beispiel 3.2** (Forschungsfrage 2). Nach dem Studium haben Sie bei einem großen Online-Auktionshaus angeheuert. Da Sie angaben, sich im Studium intensiv etwas mit Statistik beschäftigt zu haben, hat man Sie in die F&E-Abteilung<sup>3</sup> gesteckt. Heute ist es Ihre Aufgabe, Auktionen zur Spielekonsole **Wii** zu untersuchen,<sup>4</sup> genauer gesagt, geht es um das Spiel **MarioKart**.<sup>5</sup> Ihre Forschungsfrage lautet:

Welche Produktmerkmale stehen mit einem hohen Verkaufserlös in Zusammenhang?□

**Beispiel 3.3** (Handynutzung und Konzentrationsfähigkeit). Eine Forschungsfrage könnte lauten zum Thema Handynutzung:

Verringert intensive Handynutzung die Konzentrationsfähigkeit? □

**Beispiel 3.4.**

<sup>3</sup>Forschung und Entwicklung

<sup>4</sup><https://www.nintendo.de/Wii/Wii-94559.html>

<sup>5</sup>y[https://www.nintendo.de/Spiele/Wii/Mario-Kart-Wii-281848.html#\\_bersicht](https://www.nintendo.de/Spiele/Wii/Mario-Kart-Wii-281848.html#_bersicht)

### 3.3.3. Aus der Forschung: Smartphone-Brain-Drain

(ward\_brain\_2017?) untersuchten die Forschungsfrage, ob die bloße Gegenwart eines Handies (z.B. wenn es vor Ihnen auf dem Tisch liegt) dazu führt, dass man abgelenkt wird und daher schlechtere kognitive Leistungen zeigt.

Leider schreiben die Autoren Ihre Hypothese nicht glasklar, aber implizit ist obige Hypothese herauszulesen:

First, smartphones may redirect the orientation of conscious attention away from the focal task and toward thoughts or behaviors associated with one's phone. Prior research provides ample evidence that ... this digital distraction adversely affects both performance ... and enjoyment.

Später formulieren Sie Ihre Hypothese noch genauer:

In two experiments, we test the hypothesis that the mere presence of one's own smartphone reduces available cognitive capacity.

Die Ergebnisse unterstützen Ihre Hypothese, s. Abbildung 3.8. Im Diagramm ist ersichtlich, dass die kognitive Leistung (Y-Achse) sowohl in der Kapazität des Arbeitsgedächtnisses (links) als auch in der fluiden Intelligenz (rechts) am geringsten ist, wenn das Handy auf dem Schreibtisch (Desk) liegt. Am besten ist die kognitive Leistung, wenn das Handy nicht im Raum ist. □

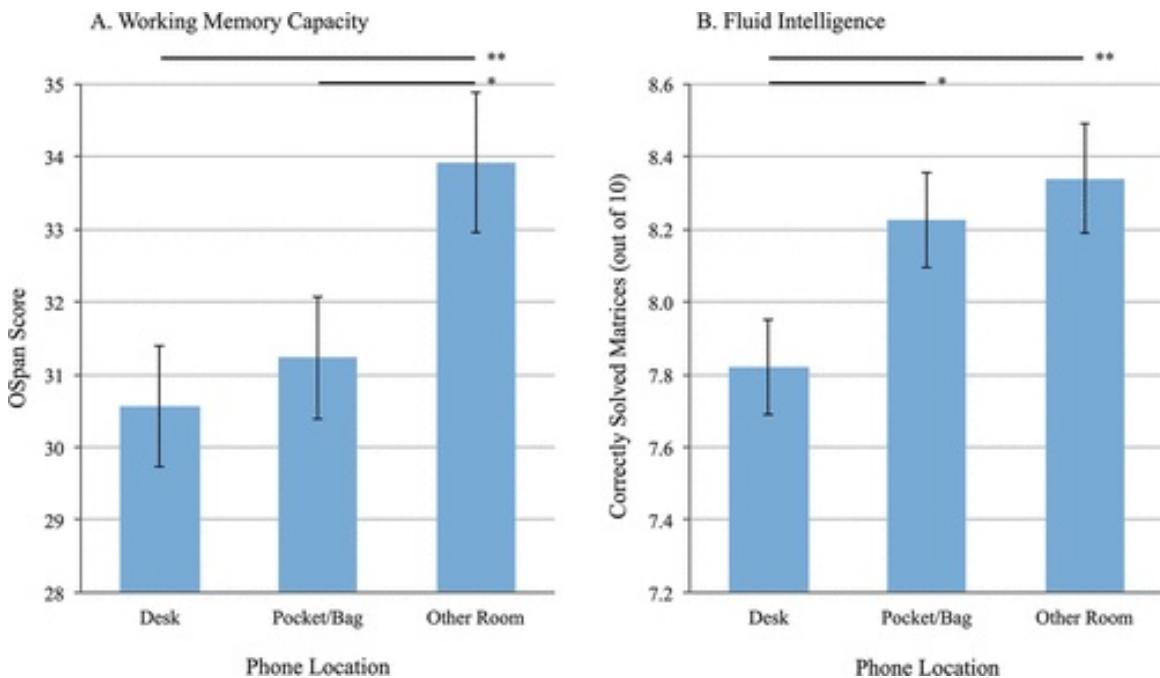


Abbildung 3.8.: Handy in Sichtweite verringert die kognitiven Ressourcen

**Übungsaufgabe 3.5.** Benennen Sie X und Y in Beispiel 3.4! □

**Übungsaufgabe 3.6.** Fragen Sie einen Bot (z.B. ChatGPT) zum Stand der Forschung hinsichtlich der Braindrain-Forschungsfrage. Diskutieren Sie die Antwort, auch in ihren Grenzen. □

### 🔥 Vorsicht

Es ist ein häufiger Fehler, in der Forschungsfrage zu formulieren "X führt zu Y", aber in der Analyse keine Methode zu verwenden, die geeignet ist, kausale Zusammenhänge aufzudecken. Es reicht nicht, dass man z.B. einen (negativen) Zusammenhang zwischen der Häufigkeit von Smartphone-Nutzung und Konzentrationsfähigkeit findet (Schwaiger und Tahir 2022), um zu sagen: "Daddeln macht dumm!". Es könnte ja z.B. auch umgekehrt sein. Platt gesagt: "Dummheit führt zu Daddeln". Weitere Erklärungen sind möglich. Vorsicht also mit (vor)schnellen Aussagen zu kausalen Abhängigkeiten.

### 3.3.4. Der Prozess der Datenanalyse

Datenanalyse ist eine Art des Problemlösens. Anders gesagt, man macht es nicht zum Spaß<sup>6</sup>, sondern um ein Ziel zu erreichen, d.h. ein Problem zu lösen. Daher analysiert man nicht gleich zu Anfang wild drauf los. Zunächst 1) klärt man das Problem und das Ziel. Dann 2) plant man das Vorgehen, z.B. welche Daten man erheben möchte. Als nächstes 3) erhebt man die Daten und bereitet sie auf. Schließlich kann man sie 4) endlich analysieren. Aber Daten sprechen nicht für sich, man muss sie 5) interpretieren und Schlüsse daraus ziehen. Dazu gehört auch, dass man die Schwächen der eigenen Analyse kritisch beleuchtet, vgl. Abbildung 3.9. Diesen Ablauf nennt man auch das PPDAC-Modell (MacKay und Oldford 2000):

- P: *Problem* (Problem und Ziel und Sachgegenstand verstehen)
- P: *Plan* (Vorgehen planen)
- D: *Data* (Daten erheben und aufbereiten)
- A: *Analysis* (Daten analysieren)
- C: *Conclusions* (Schlussfolgerungen ziehen; Daten interpretieren )

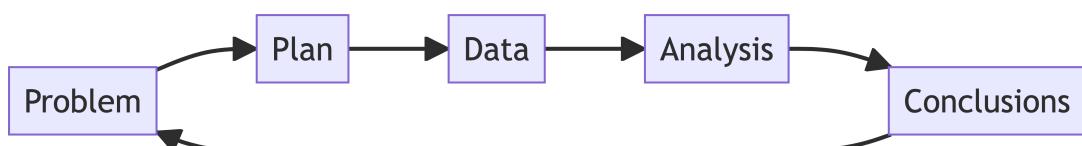


Abbildung 3.9.: Datenanalyse als Prozess: Das PPDAC-Modell

### 3.4. Was sind Daten?

**Definition 3.3** (Hallo, Daten). Daten kann man als eine geordnete Folge von Zeichen definieren. □

<sup>6</sup>jedenfalls nicht alle von uns

Daten kommen häufig in Tabellenform vor; so sind sie (oft) am besten zu untersuchen, s. Tabelle 3.1.

Tabelle 3.1.: So sehen Daten aus.

| id | name  | note |
|----|-------|------|
| 1  | Anna  | 1.3  |
| 2  | Berta | 2.3  |
| 3  | Carla | 3.0  |

Die erste Spalte `id` ist nur eine laufende Nummer. Sie dient dazu, die einzelnen Beobachtungen (hier Studenten) identifizieren zu können und birgt ansonsten keine Information. Beispiele für ID-Variablen sind z.B. Matrikulationsnummer, Personalausweisnummern oder Bestellnummern.

**Beispiel 3.5** (Daten zur Forschungsfrage 2). Hier ist ein Auszug der Daten zur Tabelle `mariokart`, s. Tabelle 3.2.

Tabelle 3.2.: Auszug aus der Tabelle `mariokart`

| duration | n_bids | cond | start_pr | ship_pr | total_pr | to_ship_sp  | seller_rate | stock_photo | wheels |
|----------|--------|------|----------|---------|----------|-------------|-------------|-------------|--------|
| 3        | 20     | new  | 0.99     | 4.0     | 52       | standard    | 1580        | yes         | 1      |
| 7        | 13     | used | 0.99     | 4.0     | 37       | first-Class | 365         | yes         | 1      |
| 3        | 16     | new  | 0.99     | 3.5     | 46       | first-Class | 998         | no          | 1      |
| 3        | 18     | new  | 0.99     | 0.0     | 44       | standard    | 7           | yes         | 1      |
| 1        | 20     | new  | 0.01     | 0.0     | 71       | media       | 820         | yes         | 2      |
| 3        | 19     | new  | 0.99     | 4.0     | 45       | standard    | 270144      | yes         | 0      |

Eine Erklärung aller Variablen des Datensatzes `mariokart` findet sich [hier](#). □

**Definition 3.4** (Data-Dictionary). Eine Erklärung, was die Namen einer Datentabelle bedeuten, nennt man *Code Book or Data Dictionary*. □

### 3.4.1. Was ist eine Variable?

**Definition 3.5** (Variable). Eine Variable ist ein Platzhalter, der für ein Merkmal steht, das verschiedene Werte annehmen kann. □

Man kann sich eine Variable wie einen Behälter vorstellen, auf dem mit einem Stift geschrieben steht, was für eine Art Inhalt darin ist, s. Abbildung 3.10.

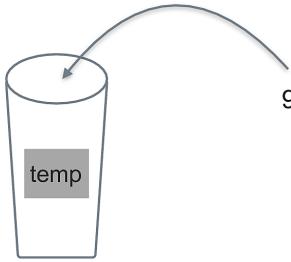


Abbildung 3.10.: Wir definieren eine Variable “temp” mit dem Inhalt “9”

### 3.4.2. Beobachtungseinheit

**Definition 3.6** (Beobachtungseinheit). Beobachtungseinheiten sind die Dinge, die wir untersuchen (beobachten). Beobachtungseinheiten sind die Träger von Variablen.  $\square$

In Tabelle 3.1 gibt es drei Variablen: `id`, `Name` und `Note`. Es gibt auch drei Beobachtungseinheiten: *Anna*, *Berta* und *Carla*.

### 3.4.3. Wert

**Definition 3.7** (Wert). Ein *Wert* ist der Inhalt einer Variablen.  $\square$

In Abbildung 3.10 ist der Wert von `temp` 9. In Tabelle 3.1 hat die Variable `name` drei Elemente: *Anna*, *Berta*, *Carla*. Der Wert des 2. Elements ist *Berta*.

**Definition 3.8** (Ausprägung). Als *Ausprägungen* bezeichnet man die verschiedenen Werte einer Variablen.  $\square$

**Beispiel 3.6.** In einer Studie wurden zehn Probanden untersucht. Die Variable `geschlecht` dokumentiert die Geschlechter der Personen:

```
geschlecht <- c("Mann", "Frau", "Frau", "Frau", "Mann",
              "Frau", "Mann", "Mann", "divers", "Frau")
geschlecht
## [1] "Mann"    "Frau"    "Frau"    "Frau"    "Mann"    "Frau"
## [6] "Mann"    "Mann"    "divers"  "Frau"
```

In dieser Variable (die aus 10 Werten besteht) finden sich drei Ausprägungen: *divers*, *Frau*, *Mann*.  $\square$

### 💡 Tipp

Gerade haben Sie etwas Computer-Syntax gesehen, genauer gesagt, Befehle aus der Programmiersprache *R*. Bisher haben wir diese Befehle nicht kennengelernt. Sie verstehen Sie vermutlich (nicht ganz). Ignorieren Sie diese Befehle einfach erstmal.

#### 3.4.4. Tidy Data

**Definition 3.9** (Tidy Data). Unter *Tidy-Data* (tidy data, “Normalform”) versteht man eine Tabelle, in der jede Zeile eine Beobachtungseinheit darstellt, jede Spalte eine Variable und jede Zelle der Tabelle einen Wert, s. Abbildung 3.11a. (Zusätzlich ist noch eine “Kopfzeile” erlaubt, in der die Namen der Variablen stehen.)□

Tabelle 3.1 ist ein Beispiel für Tidy-Data. Abbildung 3.11a zeigt ein Sinnbild für Tidy-Data (Wickham und Gromelund 2018). Und Abbildung 3.11b erläutert das Tidy-Prinzip genauer.

| country     | year | cases | population |
|-------------|------|-------|------------|
| Afghanistan | 2010 | 45    | 3071       |
| Afghanistan | 2011 | 66    | 30360      |
| Brazi       | 1970 | 37    | 1735362    |
| Brazi       | 2010 | 688   | 148988     |
| Chile       | 1970 | 2158  | 1275272    |
| Chile       | 2010 | 2995  | 120593     |

| country     | year | cases | population |
|-------------|------|-------|------------|
| Afghanistan | 2010 | 45    | 3071       |
| Afghanistan | 2011 | 66    | 30360      |
| Brazi       | 1970 | 37    | 1735362    |
| Brazi       | 2010 | 688   | 148988     |
| Chile       | 1970 | 2158  | 1275272    |
| Chile       | 2010 | 2995  | 120593     |

| country     | year | cases | population |
|-------------|------|-------|------------|
| Afghanistan | 2010 | 45    | 3071       |
| Afghanistan | 2011 | 66    | 30360      |
| Brazi       | 1970 | 37    | 1735362    |
| Brazi       | 2010 | 688   | 148988     |
| Chile       | 1970 | 2158  | 1275272    |
| Chile       | 2010 | 2995  | 120593     |

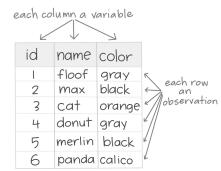
(a) Tidy-Data-Sinnbild. Image Credit: Hadley Wickham

“**TIDY DATA** is a standard way of mapping the meaning of a dataset to its structure.”

—HADLEY WICKHAM

#### In tidy data:

- each variable forms a column
- each observation forms a row
- each cell is a single measurement



Wickham, H. (2014). Tidy Data. Journal of Statistical Software 59 (10). DOI: 10.18637/jss.v059.i10

(b) Was ist Tidy-Data?. Image Credit: Allision Horst

Abbildung 3.11.: Stay Tidy!

### ❗ Wichtig

Für eine statistische Analyse ist es oft sinnvoll, dass die Daten im Tidy-Format vorliegen.

Der Vorteil des Tidy-Formats ist es, dass man weiß, wie die Daten aufgebaut sind. Außerdem können Statistikprogramme oft mit dieser Form am besten umgehen, s. Abbildung 3.12.

Das Tidy-Format wird auch als “langes” Format bezeichnet.

Abbildung 3.13 zeigt einen Datensatz in der “langen” Form, also tidy, und den gleichen Datensatz, umformatiert in der “breiten” Form, nicht-tidy.

Quelle: Garrick Aden-Buie, 2018, CC0-1.0 license, <https://www.garrickadenbuie.com/project/tidyexplain/>, CC-BY-SA

**Beispiel 3.7.** Im Folgenden sind eine Nicht-Tidy-Tabelle (Tabelle 3.3) und eine Tidy-Tablle (Tabelle 3.4) dargestellt.

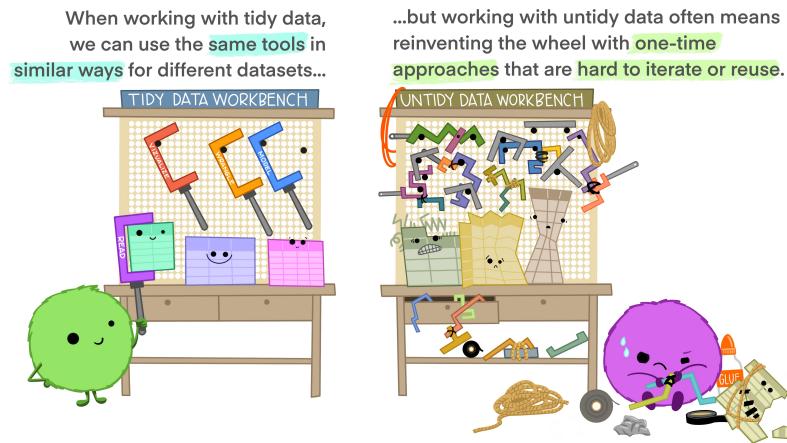


Abbildung 3.12.: Immer schön Ordnung halten... Image credit: Allision Horst, <https://github.com/allisonhorst/stats-illustrations>, CC-BY

| wide |   |   |   | long |     |     |
|------|---|---|---|------|-----|-----|
| id   | x | y | z | id   | key | val |
| 1    | a | c | e | 1    | x   | a   |
| 2    | b | d | f | 2    | x   | b   |

Abbildung 3.13.: Links: Eine Tabelle mit Format “wide” - nicht “tidy”. Rechts: Das “Langformat” (“long”) ist “tidy”.

### 3.4.4.2. Langformat (tidy)

#### 3.4.4.1. Breitformat

Tabelle 3.3.: Beispiel für eine NICHT-Tidy-Tabelle (Breitformat)

| Produkt | Umsatz_2021 | Umsatz_2022 | Umsatz_2023 |
|---------|-------------|-------------|-------------|
| Hammer  | 10          | 11          | 12          |
| Nägel   | 15          | 10          | 5           |

Tabelle 3.4.: Beispiel für eine Tidy-Tabelle (Langformat)

| Produkt | Jahr | Umsatz |
|---------|------|--------|
| Hammer  | 2021 | 10     |
| Hammer  | 2022 | 11     |
| Hammer  | 2023 | 12     |
| Nägel   | 2021 | 15     |
| Nägel   | 2022 | 10     |
| Nägel   | 2023 | 5      |

**Übungsaufgabe 3.7.** Suchen Sie ein Beispiel für eine Konfiguration einer Tabelle im Long- vs. Wide-Format.

💡 Wozu braucht man das Tidy-Format?

💡 In vielen Software-Programmen der Datenanalyse weißt man z.B. der X- oder Y-Variablen eine Spalte einer Tabelle zu. Möchte man etwa die Veränderung des Umsatzes im Verlauf der Jahre visualisieren oder analysieren, so braucht es die Spalten ‘Jahr’ und ‘Umsatz’, also ein Tidy-Format.

Abbildung 3.14 stellt auf Basis einer “Tidy-Tabelle” (Tabelle 3.4) ein Diagramm dar. Ohne Tidy-Daten wäre dieses Diagramm nicht (so einfach) zu erstellen gewesen.

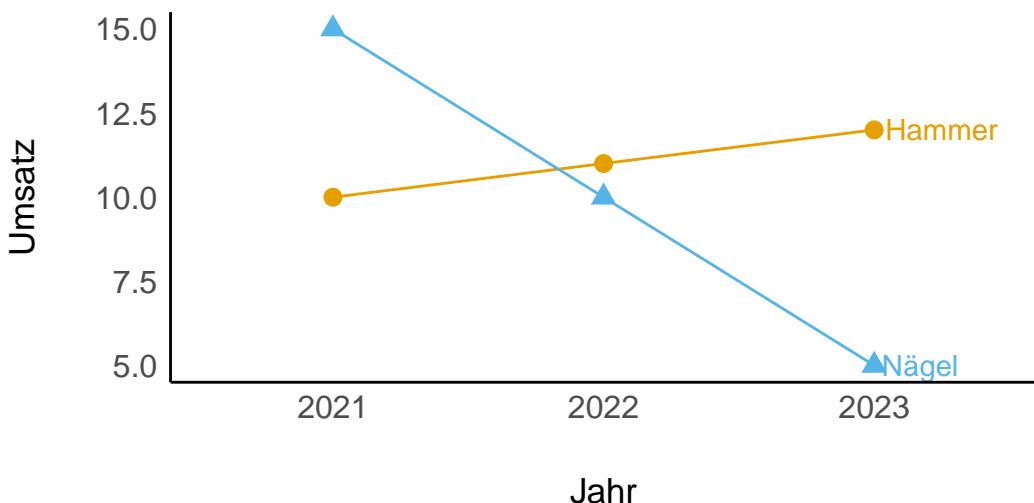


Abbildung 3.14.: Beispiel für eine Visualisierung auf Basis einer Tidy-Tabelle, vgl. Tabelle 3.4

### 3.4.5. Je mehr, desto besser (?)

Was Daten betrifft, könnte man behaupten: "Viel hilft viel" oder "Je mehr, desto besser". Natürlich unter sonst gleichen Umständen<sup>7</sup>. Viel Datenmüll ist natürlich nicht besser als ein paar knappe, wasserdichte Fakten!

**Beispiel 3.8.** Um Ihre eigene Lehraktivität zu organisieren, wollen Sie sich ein Bild machen, wie viel Ihre Nebensitzer im Hörsaal so lernen. Sie blicken nach links und fragen "wie viel lernst du so?". Sie blicken nach rechts und wiederholen die Frage gerichtet an den rechtsneben sitzenden Kommilitonen. Dann addieren Sie die zwei Zahlen (unter der Annahme, dass Sie zwei Zahlen bekommen haben), und teilen durch zwei, um den Mittelwert zu erhalten.

Ein kritischer Geist könnte anmerken, dass Sie besser die Untersuchung nicht gemacht hätten (auch wenn Sie, vielleicht ohne zu wollen, eine statistische Untersuchung angestellt haben). Denn bei so wenigen befragten Personen ist die Ungenauigkeit Ihrer Schätzung der typischen Lernzeit bei Studenten einfach zu hoch.□

Abbildung 3.15 veranschaulicht, dass man einen Mittelwert genauer schätzen kann, wenn man auf eine größere Stichprobe zurückgreift. Das Teilbild links zeigt den Mittelwert einer Stichprobe mit  $n = 20$  Beobachtungen. Das Teilbild rechts zeigt den Mittelwert einer Stichprobe mit  $n = 200$  Beobachtungen (jeweils aus der gleichen Grundgesamtheit). Wie man sieht, ist im linken Teilbild die Streuung (Variation) höher als im rechten Teilbild:

! Wichtig

Mehr Daten = genauere Ergebnisse (unter sonst gleichen Umständen) □

**Übungsaufgabe 3.8** (Live-Experiment zum Effekt der Stichprobengröße). In diesem Live-Experiment untersuchen wir den Effekt der *Stichprobengröße* auf die Streuung des Mittelwerts in der *Stichprobe*. Streuen die Ergebnisse mehr in kleinen Stichproben als in großen? Probieren wir es aus!

In diesem Experiment werfen Sie (in kleinen Gruppen) eine Münze (auf faire Art und Weise) und notieren das Ergebnis (Kopf oder Zahl). Uns interessiert dabei die Frage, ob die Ergebnisse bei kleinen Stichproben ( $n=5$  Münzwürfe) anders streuen als in großen Stichproben ( $n=20$  Münzwürfe).

Sie brauchen nur experimentierfreudige Partner (Kleingruppen mit 2-4 Personen), eine faire Münze und dann kann's los gehen! [Klicken Sie hier, um mit dem Experiment zu starten.](#)<sup>8</sup>

Die Daten aller Versuche können Sie [hier](#) einsehen.<sup>9</sup> □

**Beispiel 3.9** (Dorfschulen machen die schlauesten Schüler!). In einer Pressemitteilung sei zu lesen, dass die besten Schüler in den Dorfschulen zu finden seien<sup>10</sup>. Mit etwas Recherche finden Sie heraus, dass diese Aussage für belastbaren Daten beruht: Tatsächlich sind die Notendurchschnitte

<sup>7</sup>Ceteris paribus, auf Latein, hört sich gleich viel schlauer an

<sup>8</sup><https://forms.gle/q4F1DrbgfhLAiH1s5>

<sup>9</sup><https://tinyurl.com/3w8ke2n2>

<sup>10</sup>Das ist eine fiktive Geschichte

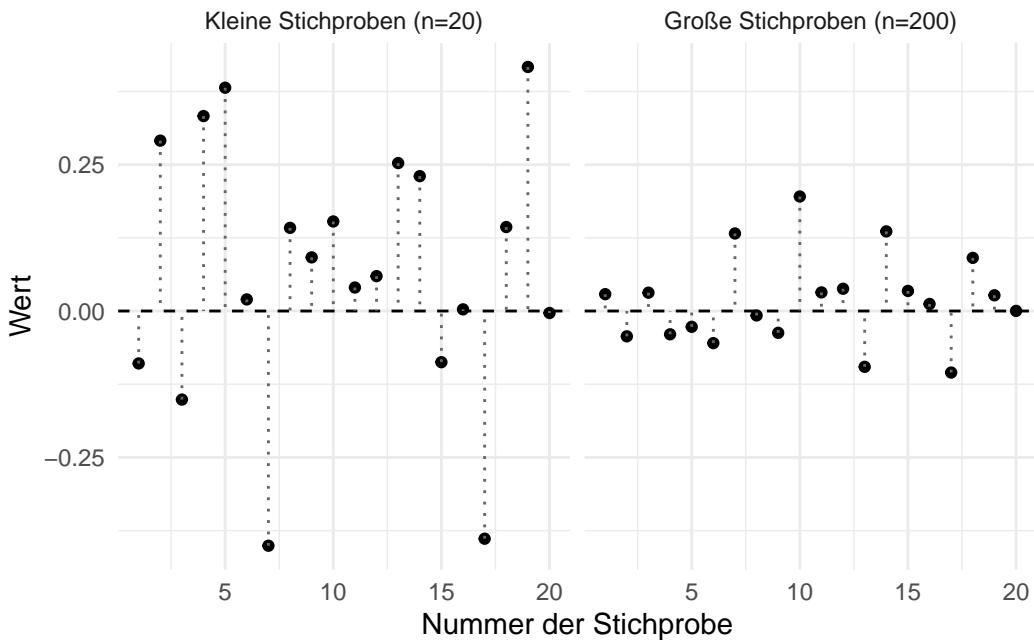


Abbildung 3.15.: Schätzgenauigkeit als Funktion der Stichprobengröße. Jeder Punkt stellt eine Stichprobe dar, entweder mit  $n=20$  (links) oder mit  $n=200$  (rechts). Kleine Stichproben (links) haben im Schnitt eine größere Abweichung vom wahren Mittelwert als größere Stichproben (rechts).

auf den kleinen Dorfschulen deutlich besser als in den großen Schulen in der Stadt. Also stimmt die Behauptung der Pressemitteilung? Die gute Landluft lässt das Hirn wachsen? Sie recherchieren noch etwas weiter in den Daten. Dann fällt Ihnen auf: Die *schlechtesten* Schüler kommen auch aus den Dorfschulen! Eine statistische Erklärung bietet sich an: In den Dorfschulen gibt es nur wenig Kinder und kleine Klassen – die Stichproben sind also klein. Bei kleinen Stichproben gibt es viel Variation um den Mittelwert herum, s. Abbildung 3.15, und zwar nach oben (guter Notenschnitt) und nach unten (schlechter Notenschnitt). □

## 3.5. Arten von Variablen

### 3.5.1. Nach Position in der Forschungsfrage

Angenommen, Ihre Forschungsfrage lautet:

Hat Lernen einen Einfluss auf den Prüfungserfolg?

In dem Fall gilt:

- *Lernen* ist die Inputvariable/X-Variable/Ursache/unabhängige Variable (UV)
- *Prüfungserfolg* ist die Outputvariable/Y-Variable/Wirkung/abhängige Variable (AV)

Abbildung 3.16 stellt diese beiden “Positionen” einer Variable dar. Die erste Position ist vor dem Pfeil. Die zweite Position ist nach dem Pfeil.

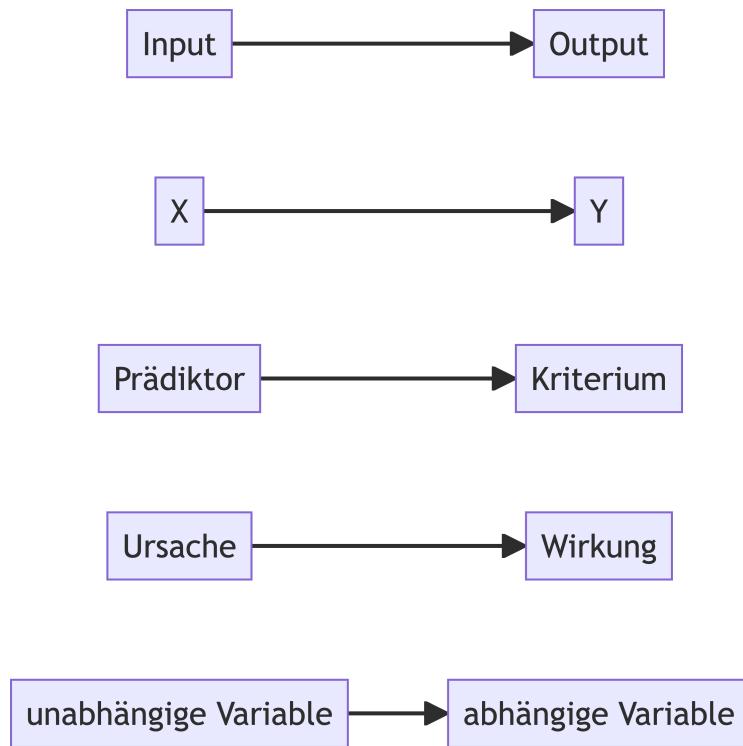


Abbildung 3.16.: Synonyme Bezeichnungen für Input- und Output-Variablen einer Forschungsfrage

**Übungsaufgabe 3.9.** Überlegen Sie sich eine Forschungsfrage, die eine UV und eine AV enthält. Sagen Sie einer/em Kommilitonen diese Forschungsfrage und fragen Sie, was die UV und die AV ist. Bei richtiger Antwort belohnen Sie großzügig. □

### 3.5.2. Nach dem Skalenniveau

**Definition 3.10** (Skalenniveau). Der Begriff *Skalenniveau* wird verwendet, um die Art und Menge der Information, die in Variablen enthalten ist, zu benennen. Diese Klassifikation basiert auf den Eigenschaften der Daten und den mathematischen Operationen, die sinnvoll auf diese Daten angewendet werden können. □

Abbildung 3.17 gibt einen Überblick über typisch verwendete Skalenniveaus.

### 3.5.3. Beispiele für Skalenniveaus

Beispiele zu den Skalenniveaus sind in Tabelle 3.5 aufgeführt. □

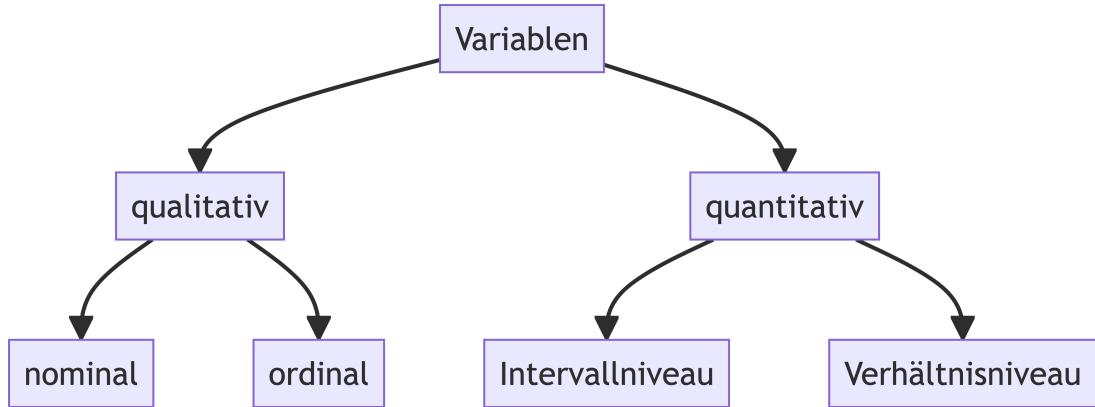


Abbildung 3.17.: Skalenniveaus

Tabelle 3.5.: Beispiele für Skalenniveaus

| Variable                      | Skalenniveau    |
|-------------------------------|-----------------|
| Haarfarbe                     | Nominalskala    |
| Augenfarbe                    | Nominalskala    |
| Geschlecht                    | Nominalskala    |
| Automerke                     | Nominalskala    |
| Partei                        | Nominalskala    |
| Lieblingsessen                | Ordinalskala    |
| Medaillen beim 100-Meter-Lauf | Ordinalskala    |
| Uniranking                    | Ordinalskala    |
| IQ                            | Intervallskala  |
| Extraversion                  | Intervallskala  |
| Temperatur in Celcius         | Intervallskala  |
| Temperatur in Fahrenheit      | Intervallskala  |
| Temperatur in Kelvin          | Verhältnisskala |
| Körpergröße                   | Verhältnisskala |
| Geschwindigkeit               | Verhältnisskala |
| Länge                         | Verhältnisskala |

Jenachdem über welches Skalenniveau eine Variable verfügt, sind verschiedenen Rechenoperationen erlaubt, s. Tabelle 3.6.

Tabelle 3.6.: Erlaubte Rechenoperationen nach Skalenniveau

| Skalenniveau    | Quantitativ | Gleichheit | Reihenfolge | Addition | Multiplikation |
|-----------------|-------------|------------|-------------|----------|----------------|
| Nominalniveau   | nein        | ja         | nein        | nein     | nein           |
| Ordinalniveau   | nein        | ja         | ja          | nein     | nein           |
| Intervallniveau | ja          | ja         | ja          | ja       | nein           |

Tabelle 3.6.: Erlaubte Rechenoperationen nach Skalenniveau

| Skalenniveau     | Quantitativ | Gleichheit | Reihenfolge | Addition | Multiplikation |
|------------------|-------------|------------|-------------|----------|----------------|
| Verhältnisniveau | ja          | ja         | ja          | ja       | ja             |

Was soll das bedeuten, “Rechenoperationen”?

Schauen wir uns für jedes Skalenniveau ein “Rechenbeispiel” an.

*Nominalskala:* Die Variable *Geschlecht* ist nominalskaliert. Das bedeutet, dass ihre Ausprägungen *Frau* und *Mann* z.B. nicht (sinnvoll) addiert oder sonstwie “verrechnet” werden können. Man könnte, z.B. um das Eintippen zu erleichtern, Frauen mit 1 kodieren und Männer mit 2. Damit darf man aber nicht rechnen! Nicht addieren, multiplizieren ... Es macht keinen Sinn zu sagen: “Ich habe eine Frau und einen Mann in meiner Tabelle, das ist im Schnitt ein diverses Geschlecht, weil der Mittelwert von 1 und 2 ist 1,5!”

Die *einige* “Rechenoperation”, die man auf der Nominalskala machen darf, ist die Prüfung auf *Gleichheit*: Mann kann feststellen, ob ein Objekt gleich zu einem anderen ist oder unterschiedlich. Also ob zwei Personen das gleiche Geschlecht haben oder von unterschiedlichem Geschlecht sind. Anders ausgedrückt:

- $\text{FRAU} \neq \text{MANN}$
- $\text{FRAU} = \text{FRAU}$
- $\text{MANN} = \text{MANN}$

*Ordinalskala:* Diese Skala entspricht einer Rangordnung. Eine Rangordnung ist etwa die geordnete Abfolge Ihres Leibgerichte<sup>11</sup>. Etwas “formaler” ausgedrückt:

- Pizza  $\succ$  Spagethi  $\succ$  Schnitzel

Das komische Zeichen  $\succ$  soll heißen: “Ist auf meiner Liste von Leibgerichten weiter oben, mag ich lieber”. Man kann aber *nicht* sagen, “Ich mag aber Pizza um 42% mehr als die Spagethi und die wieder um 73% mehr als ein Schnitzel!”. Zumindest kann man das nicht ohne weitere Informationen und Annahmen. Es gibt also Dinge auf der Welt, die man leicht in eine Rangordnung bringen kann, aber die man nur schwer in der Größe der Unterschiede bemessen kann. Das ist die Ordinalskala.

### ! Wichtig

Die Ordinalskale erlaubt, Objekte zu ordnen (hinsichtlich eines Merkmals). Die Abstände zwischen den Objekten können nicht quantifiziert werden. □

*Intervallskala:* Das ist vielleicht eine Überraschung für Sie: Wenn es heute  $10^\circ\text{C}$  hat und morgen  $5^\circ\text{C}$  – dann ist es heute *nicht* doppelt so warm wie morgen. Ja, 10 ist das Doppelte von 5. Aber  $10^\circ\text{ Celcius}$  ist *nicht* doppelt so warm wie  $20^\circ\text{ Celcius}$ . Wenn Sie das verwundert: Das ist normal, so geht es vielen Leuten, wenn sie das zum ersten Mal hören. Der Grund, dass es nicht erlaubt ist, Verhältnisse (wie doppelt/halb so viel etc.) auf der Celcius-Skala zu bilden, ist, dass der Nullpunkt der Skala,  $0^\circ\text{ C}$ , kein

<sup>11</sup>1. Pizza, 2. Spagethi, 3. Schnitzel

echter, physikalischer Nullpunkt ist. Bei  $0^\circ\text{C}$  liegt eben nicht Null Wärmeenergie vor. Stattdessen wurde eine Wärmemengen gewählt, die für uns Menschen ganz praktisch, da augenfällig ist: der Gefrierpunkt von Wasser. Was bei der Intervallskala erlaubt ist, ist das Addieren (und Subtrahieren): heute  $10^\circ\text{C}$ , morgen  $5^\circ\text{C}$ , das ist ein Unterschied von  $5^\circ\text{C}$ . Oder: Im Schnitt waren es  $7,5^\circ\text{C}$ , das ist genau in der Mitte von  $5$  und  $10^\circ\text{C}$ . Abbildung 3.18 versinnbildlicht die Intervallskala.

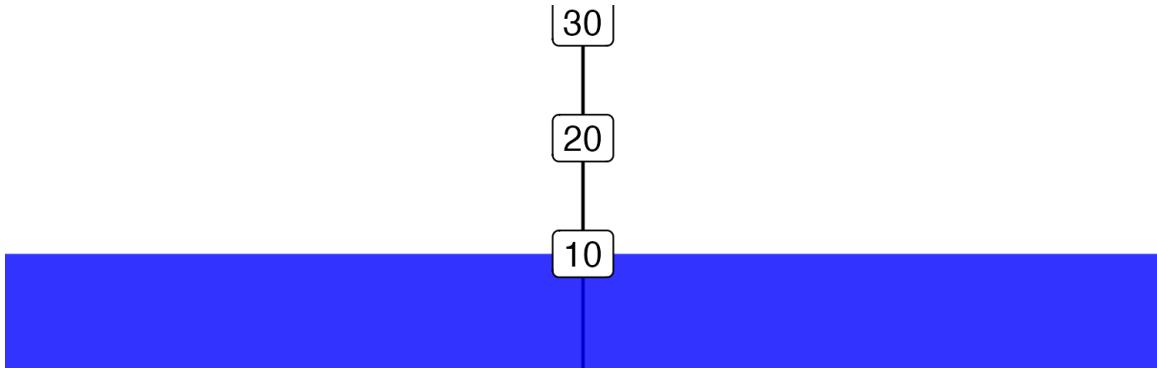


Abbildung 3.18.: Ein Metermaß steckt im Wasser. Auf dem Metermaß können wir die aufgedruckten Zahlen ablesen. Aber wir wissen nicht, ob der Metermaß auf dem Boden steht. Wir wissen demnach nicht, ob der vom Metermaß angegebene Nullpunkt der wahre Nullpunkt (Meeresboden) ist.

*Verhältnisskala:* Eine Verhältnisskala ist das, was man sich gemeinhin unter einer metrische Variable vorstellt: Man kann “normal” rechnen, alle Rechenoperationen sind erlaubt. Zuzüglich zu denen, die auch in anderen, “niedrigeren” Skalenniveaus erlaubt sind, ist das das Bilden von Verhältnissen – Multiplizieren, s. Abbildung 3.19.

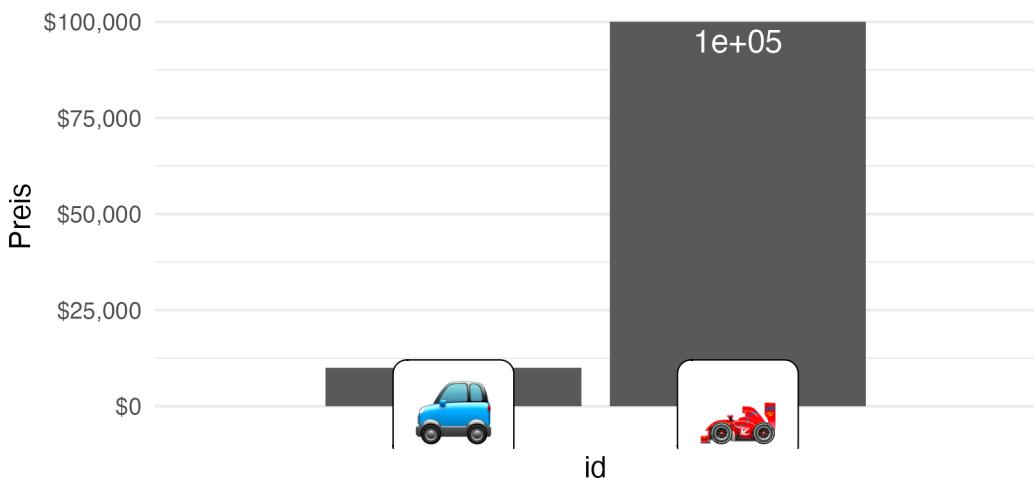


Abbildung 3.19.: Puh! Der rote Flitzer ist 10 Mal so teuer wie die blaue Möhre. Kohlen zusammenkratzen.

In [diesem Video](#) gibt es noch ausführlichere Erklärung zum Thema Skalenniveaus.

Außerdem können quantitative Variablen untergliedert werden in:

- *stetige* Variablen, das sind Variablen, bei denen man zwischen zwei Ausprägungen immer noch eine weitere quetschen kann. So gibt es eine Wert für die Körpergröße zwischen 1.60 m und 1.61 m. Und einen Wert zwischen 1.601 m und 1.602 m, etc.
- diskrete Variablen, das sind metrische Variablen, die nur bestimmte Ausprägungen haben, häufig sind das die natürlichen Zahlen: 1, 2, .... Ein Beispiel wäre die Anzahl der Kinder in einer Familie.

 Tipp

Fragen nach Skalenniveaus gehören zu den Lieblingsprüfungsfragen in diesem Themenbereich. Sie sind gut beraten, sich gerade mit dieser Frage intensiver zu beschäftigen. Auch in thematisch angrenzenden Fächern wird immer wieder die Frage nach dem Skalenniveau aufgeworfen. Das zeigt natürlich auch die hohe Relevanz des Themas.

**Übungsaufgabe 3.10.** Überlegen Sie sich für einige Variablen die Skalenniveaus und befragen Sie dann eine:n Kommilitonen dazu. □

### 3.6. Modelle

Woran denken Sie beim Wort “Modell”? Vielleicht an Spielzeugautos, s. Abbildung 3.20.



Abbildung 3.20.: Matchbox-Autos sind Modelle für Autos

**Definition 3.11** (Modelle). Modelle sind ein vereinfachtes Abbild der Realität, eine *Repräsentation* (Kaplan 2009).□

**Beispiel 3.10** (Beispiele für Modelle). Puppen sind Modelle für Babys, Landkarten für Landstriche und [das Atommodell von Nils Bohr](#) ist ein Modell für Atome.<sup>12</sup>□

Auch in der Statistik nutzen wir Modelle. Helfen Sie Prof. Weiss-Ois: Er blickt nicht durch. Gerne würde er wissen, wie viele Stunden seine Studenten auf die Prüfung lernen. Aber mit so vielen

### 3.6.1. Vorher

12, 8, 10, 11, 10, 9, 13, 9, 14, 9, 12, 14, 7, 9,  
9, 11, 9, 4, 5, 12, 9, 6, 9, 12, 13, 9, 9, 6, 10, 8



### 3.6.2. Nachher

9.6



- (a) Oh jeh, so viele Zahlen! Ich check nix! Wie  
viel lernen denn jetzt meine Studis?!
- (a) Yeah, jetzt weiß ich, wie viel die Studis so typischerweise lernen. Viel zu wenig natürlich!

Zahlen kann er nicht umgehen ... Geben Sie ihm ein Modell: Sagen Sie ihm, wie lang die Studis typischerweise lernen (sagen Sie ihm ein einfacher den Mittelwert der Lernzeiten).<sup>13</sup>

Der Nutzen von Modellen ist, dass sie komplexe Sachverhalte vereinfachen und damit oft überhaupt erst dem Verständnis oder einer Untersuchung zugänglich machen: Modelle ermöglichen Verständnis. In der Datenanalyse bzw. Statistik<sup>14</sup> fassen Sie oft viele Daten prägnant zusammen, z.B. zu einer einzelnen Kennzahl. Das Verrückte an Modellen ist, dass man Informationen wegwirft, um eine (andere, hoffentlich nützlichere) Information zu bekommen (Stigler 2016). Weniger ist mehr?!

## 3.7. Praxisbezug

Wir leben im Datenzeitalter; Daten durchdringen alle Bereiche des beruflichen, gesellschaftlichen und privaten Lebens. Die Datenanalyse hat sich in den letzten Jahren massiv verändert, s. Abbildung 3.23.

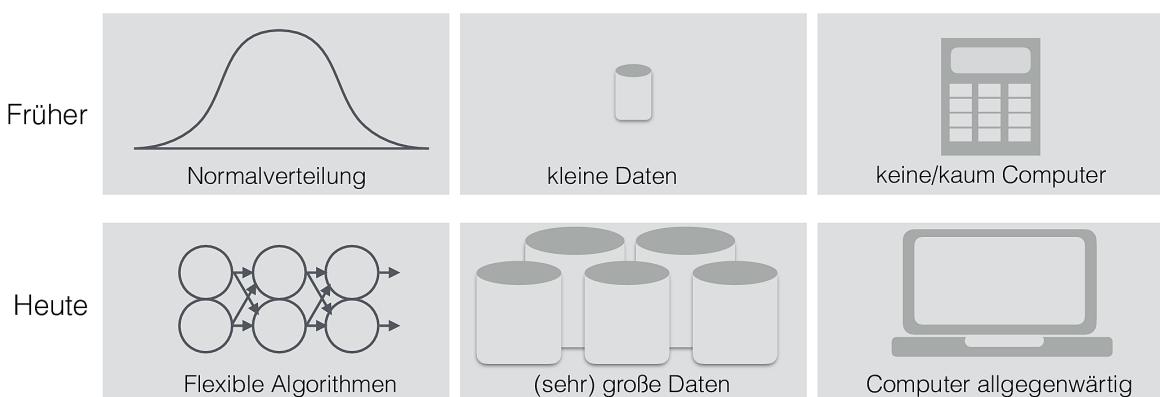


Abbildung 3.23.: Forschung früher und heute

Diese Entwicklung ist durchaus auch kritisch zu betrachten. Mit der wachsenden Bedeutung von Daten wächst in gleichem Maße die Bedeutung von Datenanalyse. Denn Daten ohne Sinn sind nutzlos.

<sup>12</sup>[https://de.wikipedia.org/wiki/Bohrsches\\_Atommodell](https://de.wikipedia.org/wiki/Bohrsches_Atommodell)

<sup>13</sup>Bildquelle: Icon unter Flaticon licence, Autor: iconixar, <https://www.flaticon.com/free-icons/professor>

<sup>14</sup>die beiden Begriffe werden hier weitgehend synonym gebraucht

Aus diesem Grund kann man sagen, dass Datenanalyse (und damit auch Statistik als eine spezielle Art von Datenanalyse) zu stark nachgefragten Jobs gehören.

Laut dem [Entgeltatlas der Bundesagentur für Arbeit](#) liegt ein typisches Gehalt von Data Scientisten bei knapp 6000 € pro Monat (in der Altersgruppe von 25 bis 54)<sup>15</sup>. Laut dem [Gehaltsreporter](#) liegt das Einstiegsgehalt dieser Berufsgruppe bei knapp 50.000€ pro Jahr.<sup>16</sup>

### 3.8. Wie man mit Statistik lügt

Das *File-Drawer-Problem*: Sie haben ein tolles Experiment durchgeführt, viel Arbeit, viel Stress, endlich geschafft, puh. Von den 20 Variablen (als AV, s. Kapitel 3.5), die Sie untersucht haben, zeigt nur 1 einen interessanten Effekt, leider. 1 von 20, das hört sich nicht so toll an. Wäre es da nicht “elegant”, die 19 Variablen ohne schönen Effekt einfach in der Schublade liegen zu lassen bis zum Sankt-Nimmerleins-Tag? Dann könnten Sie stattdessen als Ergebnis nur die eine Variable mit schönen Ergebnis präsentieren, ganz ohne widersprechende Befunde.

Dieser Versuchung nicht zu erliegen, kann schwer sein. Es ist aber gefährlich, missliebige Ergebnisse zu verschweigen: Die anderen Menschen bekommen dann ein falsches Bild der Ergebnislage; man spricht von [Publikationsbias](#).<sup>17</sup> Wer Ergebnisse verschweigt, verzerrt die insgesamte Befundlage (Rothstein 2014).

### 3.9. Fazit

Die Aufgabe von Statistik ist es, durch Zusammenfassen von Daten Modelle zu bilden, die es uns einfacher machen, schwierige Sachverhalte zu verstehen. Zentral ist dabei, die Analyse von Variabilität der Daten. Daten kommen in verschiedenen Varianten vor, typischerweise in Tabellenform, möglichst im Tidy-Format.

### 3.10. Aufgaben

Die Webseite [datenwerk.netlify.app](#) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

1. [variation01](#)
2. [Def-Statistik01](#)
3. [tidy1](#)
4. [Skalenniveau1a](#)
5. [Ziele-Statistik](#)

<sup>15</sup>Abrufdatum: 1.2.23; <https://web.arbeitsagentur.de/entgeltatlas/beruf/129987>

<sup>16</sup><https://gehaltsreporter.de/gehaelter-von-a-bis-z/it/data-scientist/>

<sup>17</sup><https://de.wikipedia.org/wiki/Publikationsbias>

6. [variation02](#)
7. [Skalenniveau1b](#)
8. [tidydata1](#)

## 3.11. Vertiefung

### 3.11.1. Excel für Könner

In vielen Organisationen werden Exceltabellen für bestimmte Zwecke der Datenverarbeitung verwendet. Excel<sup>18</sup> hat bestimmte Stärken und Vorteile, aber auch gewisse Nachteile und Schwäche; das liegt z.T. daran, dass Excel für bestimmte Aufgaben besser und für andere weniger gut geeignet ist. Wenn man mit Excel arbeitet, wiederholen sich erfahrungsgemäß immer wieder die gleichen Fehler bzw. suboptimalen Vorgehensweise zum Aufbau einer Exceltabelle.

[Dieser Artikel](#) von Broman und Woo (2018) zeigt anhand einiger praktischer Tipps, wie man Exceltabellen so aufbaut, dass Fehler minimiert werden.

**Übungsaufgabe 3.11** (Fassen Sie den Artikel von Broman und Woo (2018) zusammen). Die Lehrkraft teilt Sie dazu in Gruppen ein und weist jeder Gruppe einen Abschnitt des Artikels zu. Fassen Sie das *Wesentliche* (und nur das Wesentliche) an einem geeigneten Ort zusammen (z.B. auf einem Miro-Board). □

### 3.11.2. Sind wir süchtig nach dem Handy?

Sind Sie süchtig nach Ihrem Handy? Lassen Sie uns eine kleine Studie dazu live im Hörsaal durchführen. Füllen Sie [diese Umfrage](#) zum Thema Smartphonse-Sucht aus (anonym und kein Muss).<sup>19</sup> Kernstück der Umfrage ist die Smartphone-Sucht-Skala (Kwon u. a. 2013). Eine Studie fand, dass ca. ein Siebtel der Studierenden süchtig nach ihrem Smartphone sind (Haug u. a. 2015); demnach könnte dem Thema eine hohe Bedeutsamkeit zukommen.

Wir werden die Daten im weiteren Verlauf auswerten. □

### 3.11.3. Datenprofi plaudert aus dem Nähkästchen

Inspiration von einer Praktikerin der Datenanalyse: Caitlin Hudon verrät [in diesem Video](#), welche Fehler Sie sie in den acht Jahren ihrer Berufserfahrung gemacht hat und was sie daraus gelernt hat.<sup>20</sup>

<https://www.youtube.com/watch?v=O5IP6XcopdQ&list=PL9HYL-VRX0oQchs7dqFICoxMgnvFO10tC&index=15&t=1s>

---

<sup>18</sup>und ähnliche Programme

<sup>19</sup><https://forms.gle/PP8yb6Ubqq3JU78F9>

<sup>20</sup><https://youtu.be/O5IP6XcopdQ?si=7UsS6xbeYjnorGhx>

### **3.12. Literaturhinweise**

Einen Einblick in die Fundamente statistischer Analyse bietet Stigler (2016). Cetinkaya-Rundel und Hardin (2021), stellen grundlegende Konzepte der Analyse von Daten im Kapitel 1, “Hello data”, vor. Downey (2023) illustriert statistische Überraschungsmoment auf unterhaltsame, und vor allem: sofataugliche Art.

### **3.13. Literatur**

## 4. Daten einlesen

### 4.1. Lernsteuerung

#### 4.1.1. Standort im Lernpfad

Abb. Abbildung 1.4 den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

#### 4.1.2. Lernziele

- Sie können R und RStudio starten.
- Sie können R-Pakete installieren und starten.
- Sie können Variablen in R zuweisen und auslesen.
- Sie können Daten in R importieren.
- Sie können den Begriff *Reproduzierbarkeit* definieren.

#### 4.1.3. Überblick

Abbildung 1.4 zeigt Ihnen, wo auf unserer Reise durch die Datenanalyse sich dieses Kapitels verorten lässt.

Abbildung 4.1 zeigt den typischen Lernverlauf in Zusammenhang mit Datenanalyse (und R) an: Es gibt Höhen und Tiefen. Die wechseln sich ab. Das ist ganz normal!

#### 4.1.4. Ab diesem Kapitel benötigen Sie R

Bitte stellen Sie sicher, dass Sie R rechtzeitig einsatzbereit haben. Weiter unten in diesem Kapitel finden Sie Installationshinweise (Kapitel 4.3). Falls Sie dieses Kapitel zum ersten Mal bzw. sich noch nicht mit R auskennen, werden Sie vielleicht einige Inhalte begegnen, die Sie noch nicht gleich verstehen. Keine Sorge, das ist normal. Mit etwas Übung wird Ihnen bald alles schnell von der Hand gehen.

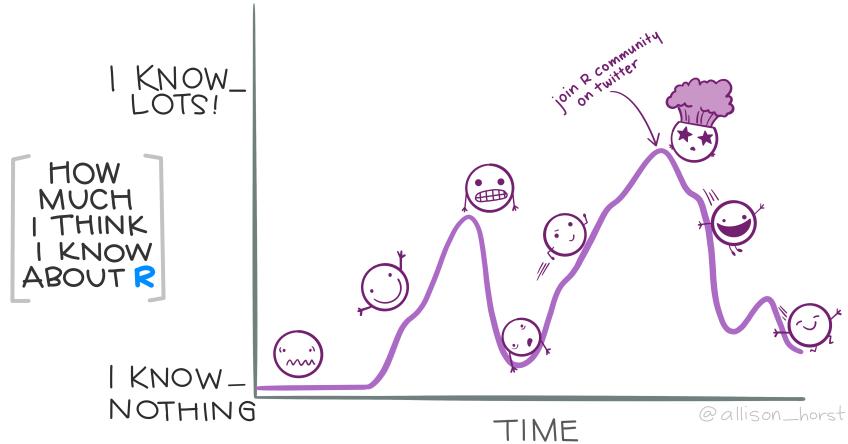


Abbildung 4.1.: Life is a roller-coaster. You just have to ride it. Image credit: Allison Horst; <https://github.com/allisonhorst/stats-illustrations>, CC-BY

#### 4.1.5. Begleitvideos

Schauen Sie sich mal den YouTube-Kanal [@sebastiansauerstatistics<sup>1</sup>](https://www.youtube.com/@sebastiansauerstatistics) an und dort die Playlist "R"<sup>2</sup>. Dort finden Sie einige Videos zum Thema R.

## 4.2. Errstkontakt

### 4.2.1. Warum R?

Gründe, die für den Einsatz von R sprechen:

1. R ist kostenlos, andere Softwarepakete für Datenanalyse sind teuer.
2. R und R-Befehle sind quelloffen, d.h. man kann sich die zugrundeliegenden Computerbefehle anschauen. Jeder kann prüfen, ob R vernünftig arbeitet. Alle können beitragen.
3. R hat die neuesten Methoden.
4. R hat eine große Community.
5. R ist maßgeschneidert für Datenanalyse.

Allerdings gibt es auch abweichende Meinungen, s. Abbildung 4.2.

<sup>1</sup><https://www.youtube.com/@sebastiansauerstatistics>

<sup>2</sup>[https://www.youtube.com/playlist?list=PLRR4REmBgpIEaIyeNBgNGPgmhQJ\\_T1y8\\_](https://www.youtube.com/playlist?list=PLRR4REmBgpIEaIyeNBgNGPgmhQJ_T1y8_)

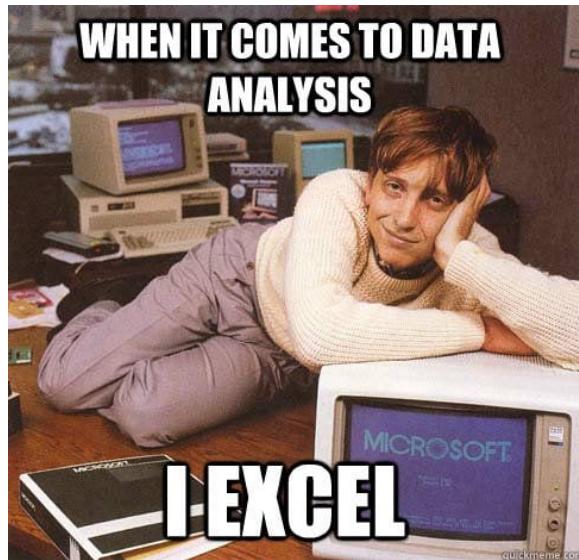


Abbildung 4.2.: Manche finden Excel cooler als R, nicht wahr, Bill Gates?

#### 4.2.2. R und Reproduzierbarkeit

**Definition 4.1** (Reproduzierbarkeit). Ein (wissenschaftlicher) Befunde ist reproduzierbar, wenn andere Analystis mit dem gleichen experimentellen Setup zum gleichen Ergebnis (wie in der ursprünglichen Analyse) kommen (Plessner 2018). □

Definition 4.1 ist, etwas überspitzt, in Abbildung 4.3 wiedergegeben.



Abbildung 4.3.: Daten + Syntax + genaue Beschreibung der Messungen = reproduzierbar

**Beispiel 4.1** (Aus der Forschung: Reproduzierbarkeit in der Psychologie).

💡 Wie ist es um unsere Wissenschaft, Psychologie, bestellt? Haben die Befunde Hand und Fuß?

Obels u. a. (2020) haben die Reproduzierbarkeit in psychologischen Studien untersucht. Sie berichten folgendes Ergebnis

We examined data and code sharing for Registered Reports published in the psychological literature from 2014 to 2018 and attempted to independently computationally reproduce the main results in each article. Of the 62 articles that met our inclusion criteria, 41 had data available, and 37 had analysis scripts available. Both data and code for 36 of the articles were shared. We could run the scripts for 31 analyses, and we reproduced the main results for 21 articles. □

#### 4.2.3. R & RStudio

Wenn wir sagen, “wir arbeiten mit R”, dann heißt das in unserem Fall “wir arbeiten mit R und mit RStudio”.



Ismay und Kim (2020) zeigen eine schöne Analogie, was der Unterschied von *R* und *RStudio* ist, s. Abbildung 4.4.<sup>3</sup>

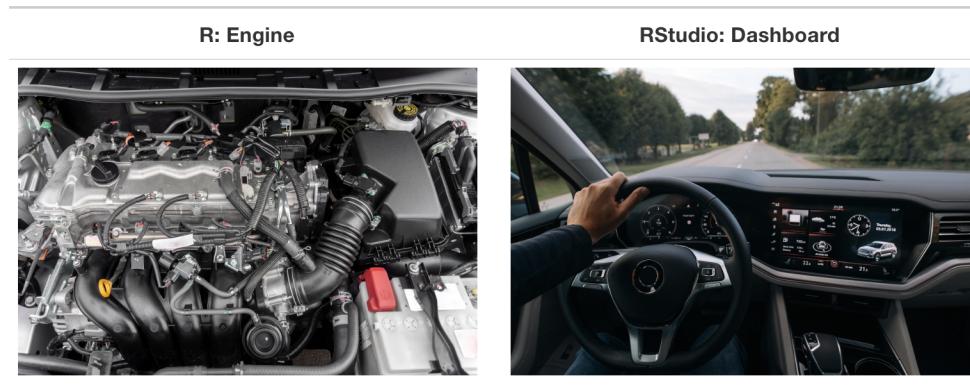


Abbildung 4.4.: R vs. RStudio: R macht die Arbeit, RStudio ist für Komfort und Übersicht

Kurz gesagt: Das eigentlich Arbeiten besorgt R. Für den Komfort und die Schönheit ist RStudio zuständig. Auch eine Art von Arbeitsteilung!

Hier sehen Sie einen Screenshot von der Oberfläche von RStudio, s. Abbildung 4.5.

<sup>3</sup>Streng genommen ist RStudio für die Datenanalyse irrelevant, aber RStudio ist praktisch, Sie werden es nicht missen wollen.

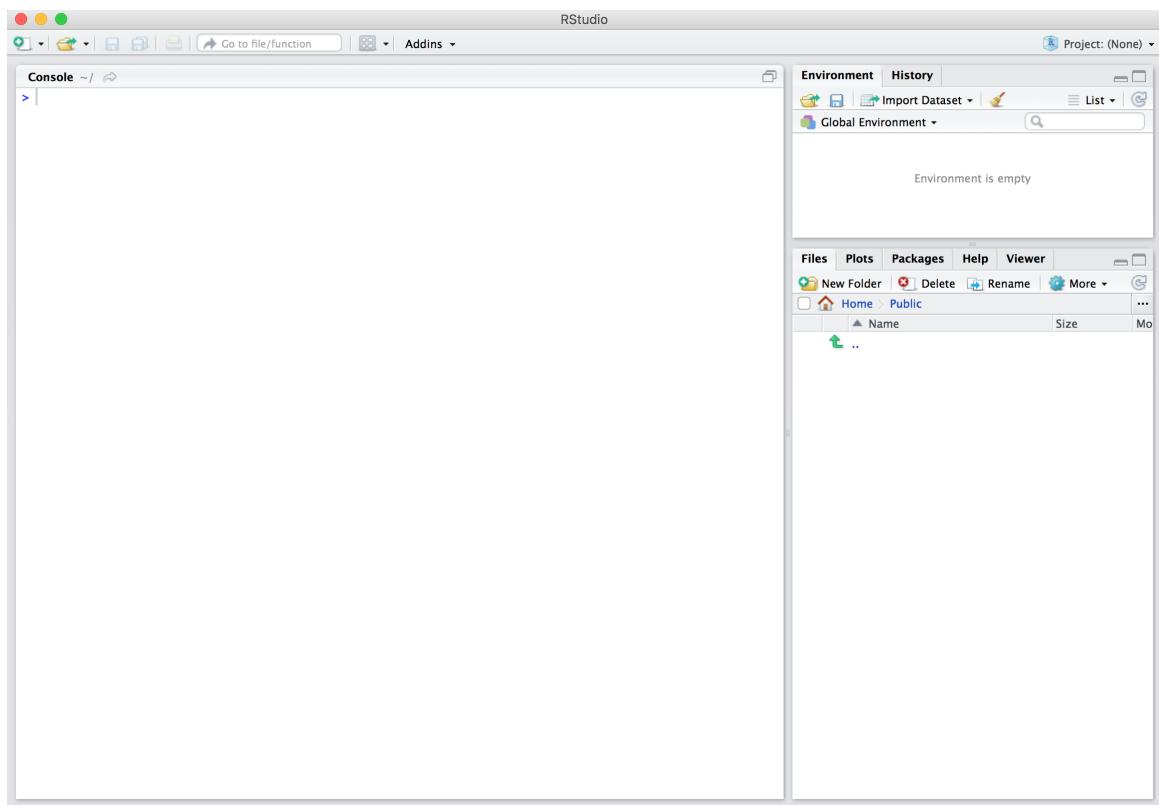


Abbildung 4.5.: So sieht RStudio aus

## 4.3. Installation von R und RStudio

### 4.3.1. Installation von R

R ist ein Softwarepaket für statistische Berechnungen<sup>4</sup>. Laden Sie es für Ihr Betriebssystem herunter:

- Windows
- MacOS
- Linux

Mehr Infos zu R finden Sie unter <https://cloud.r-project.org/>.<sup>5</sup>

Wenn Sie die Installationsdatei heruntergeladen haben, öffnen Sie diese Datei (Doppelklick) und Sie werden durch die Installation geführt.<sup>6</sup>

### 4.3.2. Installation von RStudio Desktop

RStudio ist eine *graphische Benutzeroberfläche* (graphical user interface, GUI) für R, plus ein paar Goodies<sup>7</sup>.

Laden Sie zunächst die *Desktop-Version* von RStudio herunter für Ihr Betriebssystem (Windows, MacOS, Linux) vom Anbieter (Posit) herunter.<sup>8</sup>

Wenn Sie die Installationsdatei heruntergeladen haben, öffnen Sie diese Datei (Doppelklick) und Sie werden durch die Installation geführt.<sup>9</sup>

### 4.3.3. RStudio Cloud

#### 4.3.3.1. RStudio Cloud als Alternative zu RStudio

RStudio Cloud<sup>10</sup> ist ein Webdienst von Posit/RStudio (zum Teil kostenlos), also *RStudio online*: Man kann damit online mit R arbeiten. Die Oberfläche ist praktisch identisch zur Desktop-Version, s. Abbildung 4.6. Sie können es als Alternative zur Installation von RStudio auf Ihrem Computer verwenden. Ein Vorteil von RStudio Cloud ist, dass man als Nutzer *nichts installieren* muss und dass es *auch auf Tablets* läuft (im Gegensatz zur Desktop-Version von RStudio). Ein Nachteil ist, dass es etwas langsamer ist und nur für ein gewisses Zeitvolumen kostenlos. Sie müssen sich ein Konto anlegen, um den Dienst nutzen zu können.

---

<sup>4</sup>Mehr Infos finden sich hier: [https://de.wikipedia.org/wiki/R\\_%28Programmiersprache%29](https://de.wikipedia.org/wiki/R_%28Programmiersprache%29)

<sup>5</sup>Wenn Sie gefragt werden, dass Sie einen “Mirror” auswählen sollen, heißt das, Sie sollen einen Computer (Server) wählen, von dem Sie R herunterladen. Der sollte möglichst nicht zu weit weg stehen, dann spart es vielleicht etwas Zeit und Bandbreite.

<sup>6</sup>Sie benötigen Admin-Rechte auf Ihrem Computer.

<sup>7</sup>in Form einer *integrierten Entwicklungsumgebung* (integrated development environment, IDE: [https://en.wikipedia.org/wiki/Integrated\\_development\\_environment](https://en.wikipedia.org/wiki/Integrated_development_environment))

<sup>8</sup><https://posit.co/download/rstudio-desktop/>

<sup>9</sup>Sie benötigen u.U. Admin-Rechte auf Ihrem Computer.

<sup>10</sup><https://rstudio.cloud/>; neuerdings auch “Posit Cloud” genannt

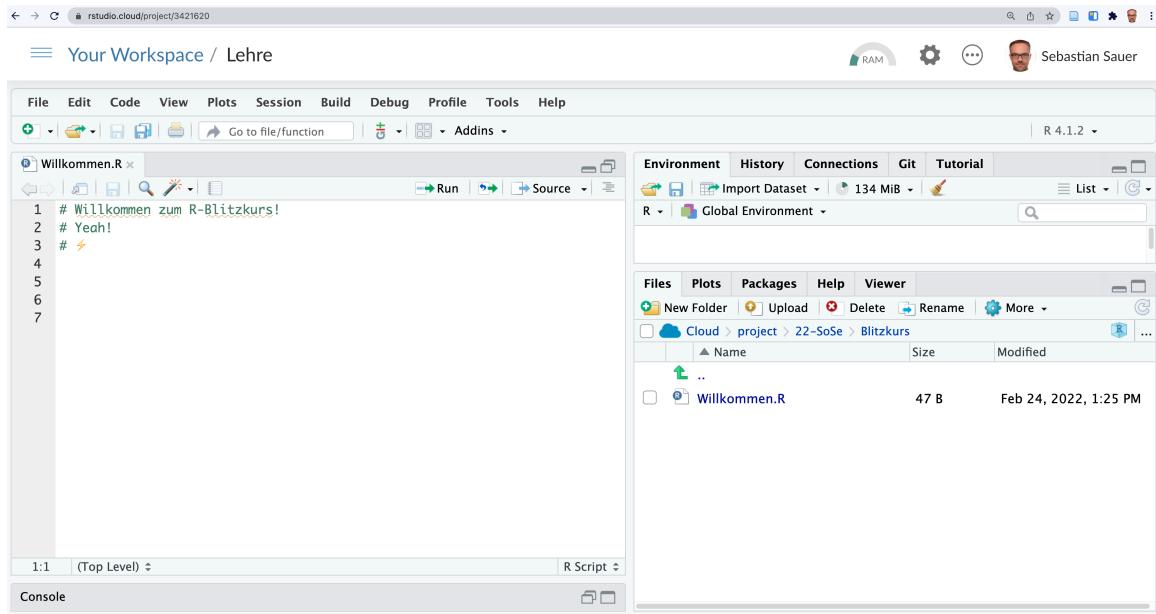


Abbildung 4.6.: So sieht RStudio Cloud aus. Genau wie RStudio Desktop

#### 4.3.3.2. Vertiefung

Wenn Ihr Dozent Ihnen einen Projektordner bzw. einen Link dazu bereitstellt, ist das komfortabel, da der Dozent dann schon Pakete installieren, Daten bereitstellen und andere Nettigkeit vorbereiten kann für Sie. Allerdings müssen Sie den Projektordner in Ihrem Konto abspeichern, wenn Sie etwas speichern möchten, da Sie vermutlich keine Schreibrechte im Projektordner Ihres Dozenten haben. Klicken Sie dazu auf “Save a permanent copy”, s. Abbildung 4.7.



Abbildung 4.7.: Einen Projektordner im eigenen Konto abspeichern, um Schreibrechte zu haben

Sie können auch von der Cloud exportieren, also Ihre Syntaxdatei herunterladen. Klicken Sie dazu im Reiter “Files” auf More > Export . . .

## 4.4. RStudio starten, nicht R

Wir verwenden beide Programme (R und RStudio). Aber wir *öffnen nur* RStudio. RStudio findet selbständig R und öffnet dieses “heimlich”. Öffnen Sie nicht noch extra R (sonst wäre R zweifach geöffnet).

Anstelle von *RStudio Desktop* (auf Ihrem Computer/Desktop) können Sie auch die *RStudio Cloud* (die Online-Version ) starten.

## 4.5. R-Pakete

### 4.5.1. Was sind R-Pakete?

Typisch für R ist sein modularer Aufbau: Man kann eine große Zahl an Erweiterungen (“Pakete”, engl. *packages*) installieren, alle kostenlos. In R Paketen “wohnen” R-Befehle, also Dinge, die R kann, “Skills” sozusagen. Außerdem können in R-Paketen auch Daten bereitgestellt werden. Damit man die Inhalte eines R-Pakets nutzen kann, muss man es zuerst installieren und dann starten.

Man kann sich daher ein R-Paket vorstellen wie ein Buch: Wenn R es gelesen hat, dann kennt es die Inhalte. Diese Inhalte könnten irgendwelche Formeln, also Berechnungen sein. Es könnte aber die “Bauanleitung” für ein schönes Diagramm sein.

Ist ein spezielles R-Paket auf Ihrem Computer installiert, so können Sie diese Funktionalität nutzen.

Die Zahl an diesen “Paketen” ist groß; zur Verdeutlichung s. Abbildung 4.8.

*Erweiterungen* kennt man von vielen Programmen, sie werden auch *Add-Ons*, *Plug-Ins* oder sonstwie genannt. Man siehe zur Verdeutlichung Erweiterungen beim Broswer Chrome, Abbildung 4.9.

Die Anzahl der R-Pakete ist groß; allein auf dem “offiziellen Web-Store” (nennt sich “CRAN”) von R gibt es ca. 20,000 Pakete (vgl. Abbildung 4.8b); [Stand: 2022](#); [Quelle](#)). Und es kommen immer mehr dazu.

### 4.5.4. Pakete installieren

Wie jede Software muss man Pakete (Erweiterungen für R) erst einmal installieren, bevor man sie verwenden kann. Ja, einmal installieren reicht.

Das geht komfortabel, wenn man beim Reiter *Packages* auf *Install* klickt (s. Abbildung 4.10) und dann den Namen des zu installierenden Pakets eingibt.

💡 Welche R-Pakete sind denn schon installiert?

Im Reiter *Packages* können Sie nachschauen, welche Pakete auf Ihrem Computer schon installiert sind. Diese Pakete brauchen Sie logischerweise dann *nicht* noch mal installieren, s. Abbildung 4.11.

Alternativ können Sie zum Installieren von Paketen auch den Befehl `install.packages()` verwenden. Also zum Beispiel `install.packages("tidyverse")` um das Paket `tidyverse` zu installieren.

💡 Ja, aber welche R-Pakete “soll” ich denn installieren, welche brauch ich denn?

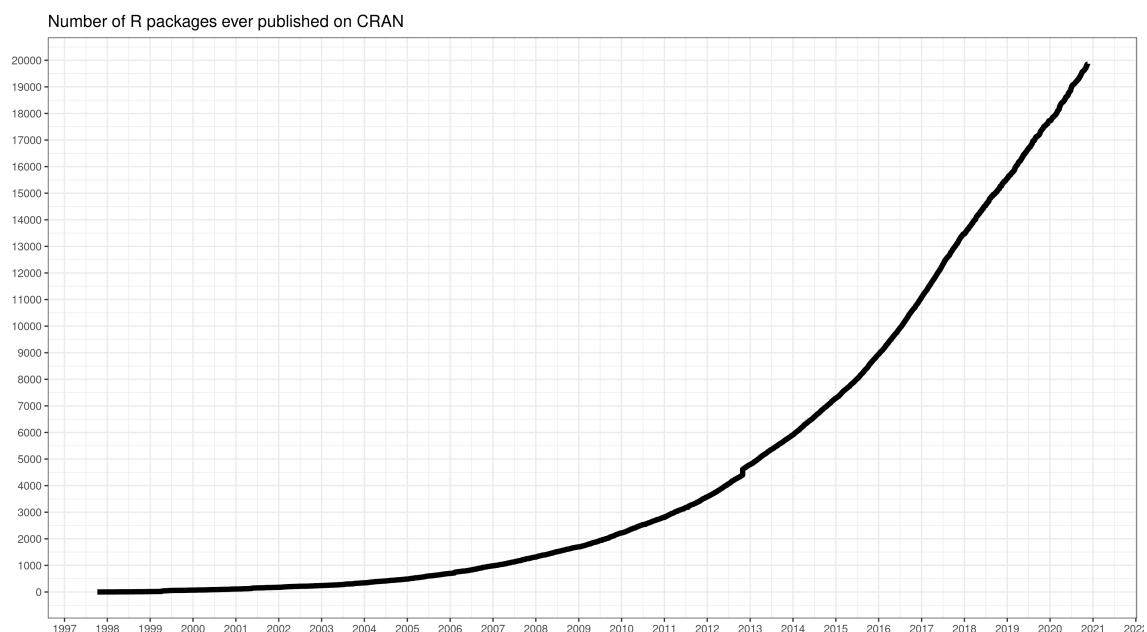
Im Moment sollten Sie die folgenden Pakete installiert haben:

#### 4.5.2. Viele Pakete



(a) Containerschiff mit vielen Paketen, Corey Seeman, CC-BY-NC 20, Flickr.com

#### 4.5.3. Es kommen viele dazu



(b) Die Anzahl der R-Pakete ist exponentiell gewachsen  
Es gibt viele R-Pakete.

Abbildung 4.8.

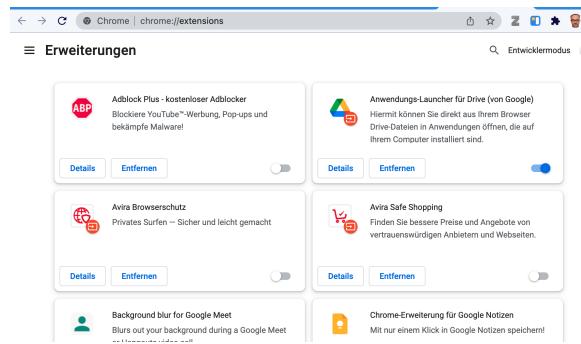


Abbildung 4.9.: Erweiterungen beim Browser Chrome

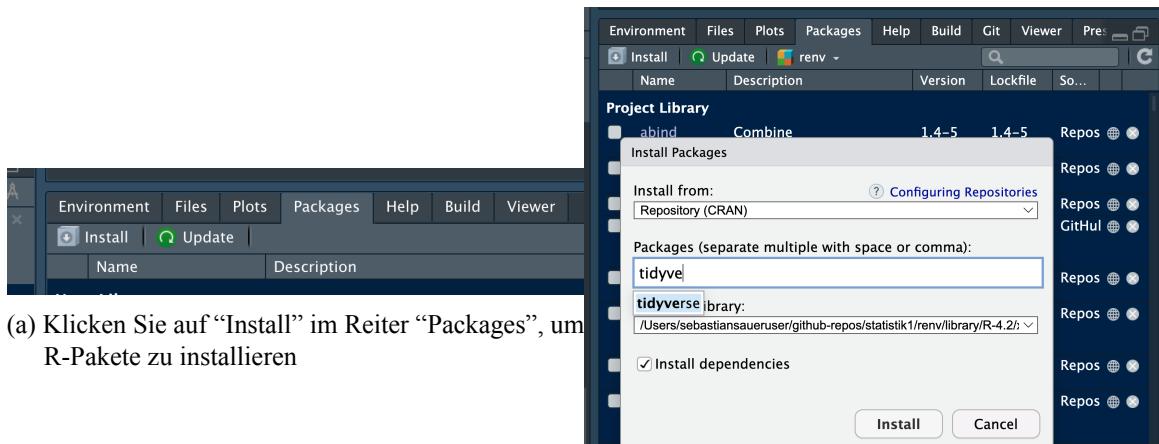


Abbildung 4.10.: So installiert man Pakete in R.

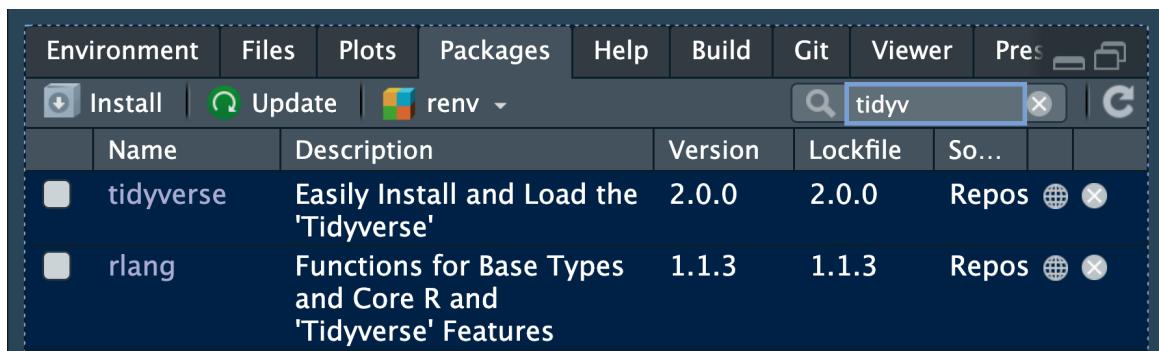


Abbildung 4.11.: So sehen Sie, ob ein R-Paket auf Ihrem System installiert ist

- tidyverse
- easystats

Wenn Sie die noch nicht installiert haben sollten, dann können Sie das jetzt ja nachholen.<sup>11</sup>

### Vorsicht

Bevor Sie ein R-Paket (oder überhaupt irgendwelche Software) installieren/updaten, sollten Sie das R-Paket schließen/beenden. Sonst schrauben Sie an einem elektrischen Gerät herum, das noch unter Strom steht (nicht gut). Die einfachste Art, alle Pakete zu beenden ist, Session > Restart R zu klicken (in RStudio).□

## 4.5.5. Pakete starten

Wenn Sie ein Softwareprogramm – nichts anderes sind R-Pakete – installiert haben, müssen Sie es noch *starten*.

Merke: Ein bestimmtes R-Paket muss man nur *einmalig installieren*. Aber man muss es *jedes Mal neu starten*, wenn man R (bzw. RStudio) startet.

Sie erkennen leicht, ob ein Paket gestartet ist, wenn Sie ein Häkchen vor dem Namen des Pakets in der Paketliste (Reiter *Packages*) sehen, s. Abbildung Abbildung 4.10a.<sup>12</sup>

## 4.6. Mit R arbeiten

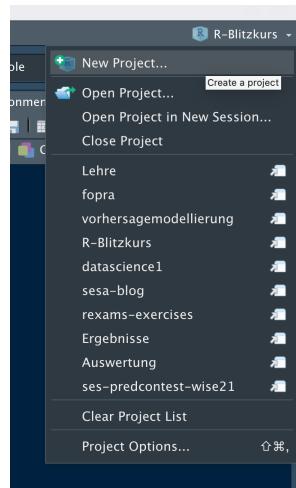
### 4.6.1. Projekte in R

Ein *Projekt* in RStudio (s. Abbildung 4.12) ist letztlich ein Ordner, der als “Basis” für eine Reihe von Dateien verwendet wird. Sagen wir, das Projekt heißt cool\_stuff. RStudio legt uns diesen Ordner an einem von uns gewählten Platz auf unserem Computer an. Das ist ganz praktisch, weil man dann sagen kann “Hey R, nimmt die Datei ‘daten.csv’”, ohne einen Pfad anzugeben. Vorausgesetzt, die Datei liegt auch im Projektordner (cool\_stuff).

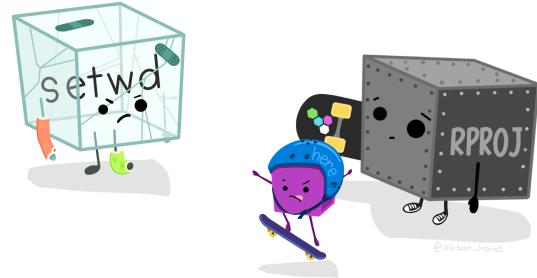
Projekte kann anlegen mit Klick auf das Icon, das einen Quader mit dem Buchstaben R darin anzeigt (s. Abbildung 4.12a). RStudio-Projekte machen Ihr Leben leichter (s. Abbildung 4.12).

<sup>11</sup>Übrigens sind tidyverse und easystats Pakete, die nur dafür da sind, mehrere Pakete zu installieren. So gehören z.B. zu tidyverse die Pakete ggplot (Daten verbildlichen) und dplyr (Datenjudo). Damit wir nicht alle Pakete einzeln installieren und starten müssen, bietet uns das Paket tidyverse den Komfort, alle die Pakete dieser “Sammlung” auf einmal zu starten. Praktisch.

<sup>12</sup>Dieses Video [https://www.youtube.com/watch?v=YeJ9xzKQ3yI&list=PLRR4REmBgpIEalyeNBgNGPgmhQJ\\_T1y8\\_&index=26](https://www.youtube.com/watch?v=YeJ9xzKQ3yI&list=PLRR4REmBgpIEalyeNBgNGPgmhQJ_T1y8_&index=26) verdeutlicht den Unterschied zwischen *Installation* und *Starten* eines R-Pakets.



(a) RStudio-Projekte, Beispiele



(b) RStudio-Projekte sind viel sicherer als das Arbeitsverzeichnis von Hand zu wählen oder mit Pfaden herumzubasteln. Image credit: Allision Horst

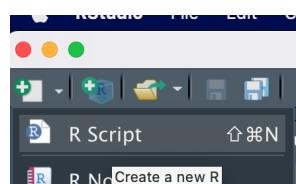
Abbildung 4.12.: Nutzen Sie RStudio-Projekte, das macht Ihr Leben leichter.

#### 4.6.2. Skriptdateien

Die R-Befehle (“Syntax”) schreiben Sie am besten in eine speziell dafür vorgesehene Textdatei in RStudio. Eine Sammlung von (R-)Computer-Befehlen nennt man auch ein *Skript*, daher spricht man auch von einer *Skriptdatei*.

##### 4.6.2.1. So öffnen Sie eine neue Skriptdatei

Um eine neue R-Skriptdatei zu öffnen, klicken Sie auf das Icon, das ein weißes Blatt mit einem grünen Pluszeichen zeigt, s. Abbildung 4.13.



(a) So erstellen Sie eine neue Skriptdatei

Abbildung 4.13.: Es gibt verschiedene Wege, um eine neue R-Skript-Datei in RStudio zu öffnen.

##### 4.6.2.2. So speichern Sie Ihre Skriptdatei

Vergessen Sie nicht zu *speichern*, wenn Sie ein tolles Skript geschrieben haben. Dafür gibt es mehrere Möglichkeiten:

1. Tastaturkürzel *Strg+S*

2. Menü: File > Save
3. Klick auf das Icon mit der Diskette, s. Abbildung 4.13.

#### 4.6.2.3. So öffnen Sie eine Skriptdatei

Eine Skriptdatei können Sie in typischer Manier *öffnen*:

1. Strg+O
2. Klick auf das Icon mit der Akte und dem grünen Pfeil (vgl. Abbildung 4.13)
3. Menü: File > Open File...

#### 4.6.3. Quarto-Dokumente

**Quarto** ist ein Programm zum Erstellen von Texten, in das man R-Syntax einfügen kann. Die Ausgaben der R-Befehle werden dann direkt im Dokument eingebunden. Abbildung 4.14 zeigt ein Beispiel für ein Quarto-Dokument.

##### i Hinweis

Quarto ist eine komfortable und leistungsfähige Methode, um Dokumente mit R-Syntax zu schreiben. Sie sind aber nicht verpflichtet, Quarto zu nutzen. Stattdessen können Sie Ihre Syntax auch in Skriptdateien schreiben. □

Wenn Sie Quarto nutzen möchten, müssen Sie es zunächst installieren, d.h. [herunterladen](#). Dann können Sie in RStudio Quarto-Dateien erstellen.<sup>13</sup> Ein neues Quarto-Dokument können Sie erstellen mit Klick auf *File > New File > Quarto Document ...*.<sup>14</sup>

#### 4.7. Errisch für Einsteiger

##### i Hinweis

Sie finden den R-Code für jedes Kapitel [hier](#). □

##### 4.7.1. Variablen

In jeder Programmiersprache kann man Variablen definieren, so auch in R:

---

<sup>13</sup><<https://quarto.org/docs/get-started/>>

<sup>14</sup>Dieses Video [https://youtu.be/\\_f3latmOhew](https://youtu.be/_f3latmOhew) gibt Ihnen Einstiegshilfe in Quarto.

The screenshot shows the Quarto IDE interface. On the left, the 'Source' panel contains R code and text. The main 'Edit' panel displays the rendered document:

## Hello, Quarto

### Meet Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>.

### Meet the penguins

The penguins data from the `palmerpenguins` package contains size measurements for `nrow(penguins)` penguins from three species observed on three islands in the Palmer Archipelago, Antarctica.

The plot below shows the relationship between flipper and bill lengths of these penguins.

```
{r}
#| label: load-packages
#| include: false

library(tidyverse)
library(palmerpenguins)
```

The right side of the interface shows a plot titled "Flipper and bill length Dimensions for penguins at Palmer Station LTER". The x-axis is "bill\_mm" and the y-axis is "flipper\_mm", both ranging from 50 to 60. Data points are colored by species: Adelie (red), Gentoo (blue), and Chinstrap (purple). A legend on the right identifies the species.

Abbildung 4.14.: Dokumente schreiben mit Quarto

```
richtige_antwort = 42
falsche_antwort = 43
typ = "Antwort"
ist_korrekt = TRUE
```

Alternativ zum Gleichheitszeichen = können Sie auch (synonym) den Zuweisungspfeil <- verwenden. Beides führt zum gleichen Ergebnis. Allerdings ist der Zuweisungspfeil präziser, und sollte daher bevorzugt werden.

Der Zuweisungspfeil <- bzw. das Gleichheitszeichen = definiert eine neue *Variable* (oder überschreibt den Inhalt, wenn die Variable schon existiert).<sup>15</sup>

```
richtige_antwort <- 42
falsche_antwort <- 43
typ <- "Antwort"
ist_korrekt <- TRUE
```

<sup>15</sup>Dieses Video [https://www.youtube.com/watch?v=TKQk-tEF9YQ&list=PLRR4REmBgpIEaIyeNBgNGPgmhQJ\\_T1y8\\_&index=28](https://www.youtube.com/watch?v=TKQk-tEF9YQ&list=PLRR4REmBgpIEaIyeNBgNGPgmhQJ_T1y8_&index=28) und dieses Video [https://www.youtube.com/watch?v=Nal0m\\_AmMwg&list=PLRR4REmBgpIEaIyeNBgNGPgmhQJ\\_T1y8\\_&index=48](https://www.youtube.com/watch?v=Nal0m_AmMwg&list=PLRR4REmBgpIEaIyeNBgNGPgmhQJ_T1y8_&index=48) geben eine Einführung in das Definieren von Variablen in R

Sie können sich eine Variable wie einen Becher oder Behälter vorstellen, der bestimmte Werte enthält. Auf dem Becher steht (mit Edding geschrieben) der Name des Bechers. Natürlich können Sie die Werte aus dem Becher entfernen und sie durch neue ersetzen (vgl. Abbildung 4.15).

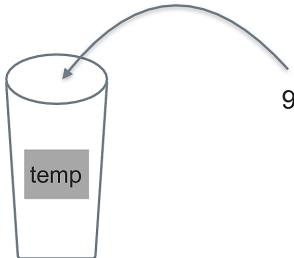


Abbildung 4.15.: Variablen zuweisen

R kann übrigens auch rechnen. Probieren Sie es doch gleich mal hier aus!

```
die_summe <- falsche_antwort + richtige_antwort
```

Aber was ist jetzt der Wert, der “Inhalt” der Variable `die_summe`?

Um den Wert, d.h. den Inhalt einer Variablen in R *auszulesen*, geben wir einfach den Namen des Objekts ein:

```
die_summe  
## [1] 85
```

Was passiert wohl, wenn wir `die_summe` jetzt wie folgt definieren?

```
die_summe <- falsche_antwort + richtige_antwort + 1
```

Wer hätt's geahnt:

```
die_summe  
## [1] 86
```

Variablen können auch “leer” sein:

```
alter <- NA  
alter  
## [1] NA
```

NA steht für *not available*, nicht verfügbar und macht deutlich, dass hier ein Wert fehlt.

💡 Wozu brauche ich bitte fehlende Werte?!

Fehlende Werte sind ein häufiges Problem in der Praxis. Vielleicht hat sich die befragte Person geweigert, ihr Alter anzugeben<sup>16</sup>. Oder als Sie die Daten in Ihren Computer eingeben wollten, ist Ihre Katze über die Tastatur gelaufen und alles war futsch...

#### 4.7.2. Funktionen (“Befehle”)

Das, was R kann, ist in “Funktionen” hinterlegt. Genauer gesagt ist “Befehl” eine Funktion.

**Definition 4.2** (Funktion). Eine Funktion ist eine Regel, die jedem Eingabewert (auch Argument genannt) einen Ausgabewert zuordnet. Man kann sich Funktionen als Maschinen vorstellen, die Eingabedaten in Ausgabedaten umwandeln, vgl. Abbildung 4.16. □

##### 4.7.2.1. Eine erste Funktion: Vektoren erstellen

Ein Beispiel für eine solche Funktion könnte sein: “Berechne den Mittelwert dieser Datenreihe” (schauen wir uns gleich an).

Das geht so:

```
Antworten <- c(42, 43)
```

Der Befehl `c` (`c` wie *combine*) fügt mehrere Werte zusammen zu einer “Liste” (einem Vektor)<sup>17</sup>.

**Definition 4.3** (Vektor). Als *Vektor* bezeichnen wir eine geordnete Folge von Werten. In R kann man sie mit der Funktion `c()` erstellen. Die Werte eines Vektors bezeichnet man als *Elemente*. □

Mit dem Zuweisungspfeil geben wir diesem Vektor einen Namen, hier `Antworten`. Dieser Vektor besteht aus zwei Werten, zuerst 42, dann kommt 43.

**Beispiel 4.2** (Beispiele für Vektoren). Vektoren können (praktisch) beliebig lang sein, z.B. drei Elemente.

```
x <- c(1, 2, 3)
y <- c(2, 1, 3)  # x und y sind ungleich (Reihenfolge der
                 ↴ Werte)
z <- c(3.14, 2.71)
namen <- c("Anni", "Bert", "Charli") # Text-Vektor
```

<sup>16</sup>Datenschutz!

<sup>17</sup>Streng genommen sollte man nicht von einer Liste sprechen, da es in R noch einen anderen Objekttyp gibt, der `list` heißt, und eine verallgemeinerte Form eines Vektors ist.

Zwei wichtige Typen von Vektoren sind numerische Vektoren (reelle Zahlen; in R auch als *numeric* oder *double* bezeichnet) und Textvektoren, in R auch als *String* oder *character* bezeichnet.

**Beispiel 4.3.** Weitere Beispiel für Funktionen sind:

- “Erstelle eine Liste (Vektor) von Werten”.
- “Lade dieses R-Paket.”
- “Gib den größten Wert dieser Datenreihe aus.” □

#### 4.7.3. Unsere erste statistische Funktion

Jetzt wird's ernst. Jetzt kommt die Statistik. □ Berechnen wir also unsere erste statistische Funktion: Den Mittelwert. Puh.

```
mean (Antworten)
## [1] 42.5
```

Sie hätten *Antworten* auch durch `c(42, 43)` ersetzen können, so haben Sie ja schließlich die Variable gerade definiert.

R arbeitet so einen “verschachtelten” Befehl *von innen nach außen* ab:

Start: `mean (Antworten)`

↓

Schritt 1: `mean (c(42, 43))`

↓

Schritt 2: `42.5`

##### 4.7.3.1. Schema einer Funktion

Abbildung 4.16 stellt eine Funktion schematisch dar.

##### 4.7.3.2. Argumente einer Funktion

Eine Funktion hat einen oder mehrere *Inputs* (s. Abbildung 4.16), das sind Daten oder Verarbeitungshinweise, die man in die Funktion *fun eingibt*, bevor sie loslegt. Eine Funktion hat immer (genau) eine *Ausgabe* (Output), in der das Ergebnis einer Funktion ausgegeben wird.

**Definition 4.4** (Argumente einer Funktion). Die “Trichter” einer (R-)Funktion, in denen man die Eingaben “einfüllt”, nennt man auch *Argumente*. □

So hat die Funktion `mean ()` z.B. folgende Argumente, s. Listing 4.1.

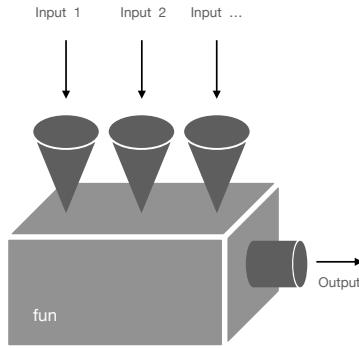


Abbildung 4.16.: Schema einer Funktion

---

**Listing 4.1** Die Argumente der R-Funktion `mean`

---

```
mean(x, trim = 0, na.rm = FALSE, ...)
```

---

- `x`: das ist der Vektor, für den der Mittelwert berechnet werden soll
- `trim = 0`: Sollen die extremsten Werte von `x` lieber “abgeschnitten” werden, also nicht in die Berechnung des Mittelwerts einfließen?
- `na.rm = FALSE`: Wie soll mit fehlenden Werten NA umgegangen werden? Im Standard liefert `mean`<sup>18</sup> NA zurück. R schwenkt sozusagen die rote Fahne, um zu signalisieren, Achtung, Mensch, hier ist irgendwas nicht in Ordnung. Setzt man aber `na.rm = TRUE`, dann entfernt (remove, rm) R die fehlenden Werte und berechnet den Mittelwert.
- `...` heißt “sonstiges Zeugs, das manchmal eine Rolle spielen könnte”; darum kümmern wir uns jetzt nicht.

Einige Argumente haben einen *Standardwert* bzw. eine *Voreinstellung* (engl. *default*). So wird bei der Funktion `mean` im Standard nicht getrimmt (`trim = 0`) und fehlende Werte werden nicht entfernt (`na.rm = FALSE`).

---

<sup>18</sup>und viele andere arithmetische Funktionen in R

### **i Hinweis**

Wenn ein R-Befehl ein Argument mit Voreinstellung hat, brauchen Sie dieses Argument *nicht* zu befüllen. In dem Fall wird auf den Wert der Voreinstellung zurückgegriffen. Argumente ohne Voreinstellung – wie `x` bei `mean()` – müssen Sie aber auf jeden Fall mit einem Wert befüllen. Man würde also `mean` zumeist so aufrufen: `mean(x)`. □

Bei jedem R-Befehl haben die Argumente eine bestimmte Reihenfolge, etwa bei `mean():mean(x, trim = 0, na.rm = FALSE, ...)`.

(Nur) wenn man die Argumente in ihrer vorgegebenen Reihenfolge anspricht, muss man *nicht* den Namen des Arguments anführen:

`mean(Antworten, 0, FALSE)`

Hält man sich aber nicht an die vorgebene Reihenfolge, so weiß R nicht, was zu tun ist und flüchtet sich in eine Fehlermeldung:

```
mean(Antworten, FALSE, 0) # FALSCH, DON'T DO IT
## Error in mean.default(Antworten, FALSE, 0): 'trim' must be
#<-- numeric of length one
```

Wenn man die Namen der Argumente anspricht, ist die Reihenfolge egal:

```
mean(na.rm = FALSE, x = Antworten) # ok
mean(trim = 0, x = Antworten, na.rm = TRUE) # ok
```

Übrigens: Leerzeichen sind R fast immer egal. Aus Gründen der Übersichtlichkeit sollte man aber Leerzeichen verwenden. In diesen Fällen sind Leerzeichen nicht erlaubt:

- `<-`
- `<=` etc.
- Variablennamen

#### **4.7.3.3. Achtung bei fehlenden Werten**

Sagen wir, wir haben einen fehlenden Wert in unseren Daten:

```
Antworten <- c(42, 43, NA)
Antworten
## [1] 42 43 NA
```

Wenn wir jetzt den Mittelwert berechnen wollen, quittiert R das mit einem schnöden `NA`. `NA` steht für *not available*, ist also ein Hinweis, dass Werte fehlen.

```
mean (Antworten)
## [1] NA
```

R meint es gut mit Ihnen<sup>19</sup>. Stellen Sie sich vor, dass R Sie auf dieses Problem aufmerksam machen möchte:

💡 Achtung, lieber Herr und Gebieter, du hast nicht mehr alle Latten am Zaun, will sagen, alle Daten im Vektor!

(Danke, R.)

Möchten Sie aber lieber R dieses Verhalten austreiben, so befüllen Sie das Argument `na.rm` mit dem Wert `TRUE`.<sup>20</sup>

```
mean (Antworten, na.rm = TRUE)
## [1] 42.5
```

**Übungsaufgabe 4.1** (Geben Sie neue Bedeutungen an, was “NA” noch bedeuten könnte!).

💡 Wie wäre es mit “nebulöse Anomalie” oder “nix-checkender Angeber” oder “nölder Automat”.

💡 Hm...

□

#### 4.7.4. Vektorielles Rechnen

**Definition 4.5** (Vektorielles Rechnen). Das Rechnen mit Vektoren in R bezeichnen wir als *vektorielles Rechnen*. □

Vektorielles Rechnen ist ein praktische Angelegenheit, man kann z.B. folgende Dinge einfach in R ausrechnen.

Gegeben sei `x` als Vektor `(1, 2, 3)`. Dann können wir die Differenz (Abweichung) jedes Elements von `x` zum Mittelwert von `x` komfortabel so ausrechnen:

```
x - mean (x)
## [1] -1  0  1
```

Etwas fancier ausgedrückt: Wir haben die Funktion mit Namen “Differenz” (“Minus-Rechnen”) auf jedes Element von `x` angewandt. Im Einzelnen haben wir also folgenden drei Differenzen ausgerechnet:

<sup>19</sup>> 💡 Naja, manchmal.

<sup>20</sup>`na.rm` steht für *remove* die NA, also fehlenden Werte

1 – 2  
 2 – 2  
 3 – 2

Diese drei Rechenschritte sind symbolisch in Abbildung 4.17 dargestellt.

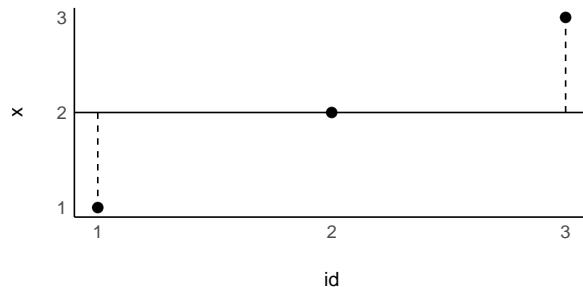


Abbildung 4.17.: Schema des vektoriellen Rechnens: Eine Funktion wird auf jedes Element eines Vektors angewandt. Hier:  $1-2=-1$ ;  $2-2=0$ ;  $3-2=1$

#### 4.7.5. R-Quiz

**Übungsaufgabe 4.2.** Ihre R-Muskeln sind gestählt? Oder doch noch nicht so ganz ausdefiniert? Macht nichts! Trainieren Sie sich mit dem R-Quiz auf der Datenwerk-Webseite<sup>21</sup>! □

#### 4.7.6. Ich brauche R-Hilfe!

- *Wo finde ich Hilfe zu einer bestimmten Funktion, z.B. fun ()?* Geben Sie dazu folgenden R-Befehl ein: `help(fun)`. Alternativ geben Sie den Namen der Funktion in RStudio im Suchfeld beim Reiter Help ein.
- *Wenn ich ein R-Paket installiere, fragt mich R manchmal, ob ich auch Pakete installieren, will, die "kompiliert" werden müssen. Soll ich das machen?*? Nein, das ist nicht nötig; geben Sie "no" ein.
- *In welchem Paket wohnt meine R-Funktion?* Suchen Sie nach der Funktion auf der Webseite RDocumentation<sup>22</sup>.
- *Ich weiß nicht, wie der R-Befehl funktioniert!* Vermutlich haben andere Ihr Problem auch, und meistens hat irgendwer das Problem schon gelöst. Am besten suchen Sie mal auf Stackoverflow<sup>23</sup>.
- *Ich muss mal grundlegend verstehen, wozu ein bestimmten R-Paket gut ist. Was tun?* Lesen Sie die Dokumentation ("Vignette") eines R-Pakets durch. Für das Paket dplyr bekommen Sie so einen Überblick über die verfügbaren Vignetten dieses Pakets: `vignette(package = "dplyr")`. Dann suchen Sie sich aus der angezeigten Liste eine Vignette raus; mit

<sup>21</sup><https://datenwerk.netlify.app/posts/r-quiz/r-quiz>

<sup>22</sup><https://www.rdocumentation.org/>

<sup>23</sup><https://www.stackoverflow.com>

`vignette("rowwise")` können Sie sich dann die gewünschte Vignette (z.B. `rowwise`) anzeigen lassen.

- *Oh nein, ich seh rot, das heißt, R zeigt mir irgendwas in roter Schrift an. Ist jetzt was kaputt?* Keine Sorge, R ist in seiner Ausgabe nicht sparsam mit roter Farbe. Solange es nicht als Fehlermeldung (ERROR) erscheint, ist es meist kein Problem.
- *R hat sich aufgehängt oder bringt einen Fehler an einer Stelle, wo sonst alles funktioniert hat.* Probieren Sie auf jeden Fall mal das AEG-Prinzip (Aus-Ein-Gut): sprich R neu starten.
- *Ich suche schon seit einer Stunde einen Fehler und find ihn nicht. Ich habe schon verschiedene Gegenstände vor Wut an die Wand geworfen. Was soll ich tun?* Machen Sie eine Pause. Doch, das ist ernst gemeint. Meine Erfahrung: Mit etwas Abstand wird der Kopf klarer und man findet das Problem viel einfacher.<sup>24</sup>
- *Irgendwie reagiert R komisch, vielleicht hat es sich aufgehängt?* Starten Sie R neu. Klicken Sie auf `Session > Restart R`.
- *Ich muss mal klar Schiff machen und alle (oder einige) Variablen löschen. Wie wird ich das Zeug wieder los?* Beim Neustart von R werden alle Objekte (Variablen) gelöscht. Einzelne Objekte können Sie selektiv löschen mit dem Befehl `rm`, so löscht `rm(mariokart)` das Objekt namens `mariokart`.

### Vorsicht

R ist penibel: So sind `name` und `Name` zwei verschiedene Variablen für R. Groß- und Kleinschreibung wird von R streng beachtet! Hingegen ist es R egal, ob Sie zur besseren Übersichtlichkeit Leerzeichen in Ihre Syntax tippen. Ausnahme sind spezielle Operatoren wie `<-` oder `<=`.

Eine gute Nachricht: Wenn R etwas von `WARNING` (bzw. Warnung) sagt, können Sie das zumeist ignorieren. Eine *Warnung* ist kein Fehler (ERROR) und meistens nicht gravierend oder nicht dringend. Ihre Syntax läuft trotzdem durch. Im Zweifel ist Googeln eine gute Idee. Nur wenn R von `Error` spricht, ist es auch ein Fehler und Ihre Syntax läuft nicht durch.□

## 4.8. Mit Daten arbeiten

### 4.8.1. Wo sind meine Daten?

Damit Sie eine Datendatei importieren können, müssen Sie wissen, wo die Datei ist. Schauen wir uns zwei Möglichkeiten an, wo Ihre Datei liegen könnte.

1. Irgendwo im Internet<sup>25</sup>
2. Irgendwo auf Ihrem Computer, z.B. in Ihrem R-Projektordner

In beiden Fällen wird der “Aufenthaltsort” der Datei durch den *Pfad*<sup>26</sup> und den Namen der Datei definiert.

<sup>24</sup>Und manchmal ist einem das Problem danach schlichtweg egal.

<sup>25</sup>z.B. hier: <https://vincentarelbundock.github.io/Rdatasets/csv/openintro/mariokart.csv>

<sup>26</sup>Der Pfad einer Datei sagt, in welchem Ordner und Unterordner und Unter-Unterordner die gesuchte Datei liegt. Ein Pfad könnte z.B. so aussehen: “/Users/sebastiansaueruser/github-repos/statistik1/”.

### **i Hinweis**

Wir werden in diesem Kurs häufiger mit dem Daten `mariokart` arbeiten; Sie finden ihn [hier](#).<sup>27</sup>

#### **4.8.2. Gebräuchliche Datenformate**

Daten werden in verschiedenen Formaten im Computer abgespeichert; Tabellen häufig als

- Excel-Datei
- CSV-Datei

In der Datenanalyse ist das gebräuchlichste Format für Daten in Tabellenform die *CSV-Datei*. Das hat den Grund, weil dieses Format technisch schön einfach ist. Für uns Endverbraucher tut das nichts groß zur Sache, die CSV-Datei beherbergt einfach eine brave Tabelle in einer *Textdatei*, sonst nichts.

In diesem Buch werden wir mit einem Datensatz namens `mariokart` arbeiten; hallo Mario (s. Abbildung 4.18)!



Abbildung 4.18.: Hallo, Mario

**Übungsaufgabe 4.3** (CSV-Datei von innen).

#### **4.8.3. Aufgabe**

Öffnen Sie die CSV-Datei `mariokart.csv` mit einem *Texteditor* (nicht mit Word und auch nicht mit Excel). Schauen Sie sich gut an, was Sie dort sehen und erklären Sie die Datenstruktur.

#### **4.8.4. Lösung**

Eine CSV-Datei repräsentiert eine Datentabelle. Eine Spaltengrenze wird mittels eines Kommas dargestellt (man kann auch andere Zeichen wählen, um Spalten voneinander abzugrenzen).

<sup>27</sup>Auf dieser Webseite <https://vincentarelbundock.github.io/Rdatasets/articles/data.html> finden Sie eine große Zahl an Datensätzen. Nur für den Fall, dass Ihnen langweilig ist.

## 4.8.5. Daten importieren

### 4.8.5.1. Importieren von einem R-Paket

Ihr Datensatz schon in einem R-Paket gespeichert, können Sie ihn aus diesem R-Paket starten. Das ist die bequemste Option. Zum Beispiel “wohnt” der Datensatz `mariokart` im R-Paket `openintro`.

#### 💡 Tipp

Ein häufiger Fehler ist, dass man vergisst, dass man zuerst ein R-Paket installieren muss, bevor man es nutzen kann. Auf der anderen Seite muss man ein R-Paket (wie andere Software auch) nur ein Mal installieren – das Paket muss man ein Paket nach jedem Neustart von RStudio mit `library()` starten.

```
data("mariokart", package = "openintro")
```

### 4.8.5.2. Importieren von einer Webseite

Hier ist eine Möglichkeit, Daten (in Form einer Tabelle) von einer Webseite (URL) in R zu importieren:

```
mariokart <- read.csv(paste0(  
  "https://vincentarelbundock.github.io/Rdatasets/",  
  "csv/openintro/mariokart.csv"))
```

Es ist egal, welchen Namen Sie der Tabelle geben. Ich nehme oft `d`, `d` die Daten. Außerdem ist `d` kurz, muss man nicht so viel tippen. Auf der anderen Seite ist `d` nicht gerade präzise und vielsagend.

Werfen wir einen Blick in die Tabelle (engl. *to glimpse*):

```
glimpse(d)  
## #> Rows: 143  
## #> Columns: 12  
## #> $ id <dbl> 150377422259, 260483376854, 3204323429~  
## #> $ duration <int> 3, 7, 3, 3, 1, 3, 1, 1, 3, 7, 1, 1, 1, ~  
## #> $ n_bids <int> 20, 13, 16, 18, 20, 19, 13, 15, 29, 8, ~  
## #> $ cond <fct> new, used, new, new, new, new, used, n~  
## #> $ start_pr <dbl> 0.99, 0.99, 0.99, 0.99, 0.01, 0.99, 0.~  
## #> $ ship_pr <dbl> 4.00, 3.99, 3.50, 0.00, 0.00, 4.00, 0.~  
## #> $ total_pr <dbl> 51.55, 37.04, 45.50, 44.00, 71.00, 45.~  
## #> $ ship_sp <fct> standard, firstClass, firstClass, stan~  
## #> $ seller_rate <int> 1580, 365, 998, 7, 820, 270144, 7284, ~
```

```
## $ stock_photo <fct> yes, yes, no, yes, yes, yes, yes, yes, yes, ~
## $ wheels      <int> 1, 1, 1, 1, 2, 0, 0, 2, 1, 1, 2, 2, 2, ~
## $ title       <fct> "~~~ Wii MARIO KART & WHEEL ~ NINTE~
```

Hier findet sich eine Erklärung des Datensatzes.

#### 4.8.5.3. Importieren von Ihrem Computer in RStudio Desktop

Gehen wir davon aus, dass sich die Datendatei im gleichen Ordner wie die R-Datei<sup>28</sup> befindet, in der Sie den Befehl zum Importieren schreiben. Dann können Sie die Datei einfach so importieren:

```
d <- read.csv("mariokart.csv")
```

Dieses Video erklärt die Schritte des Importierens einer Datendatei von Ihrem Computer.<sup>29</sup>

#### 4.8.5.4. Importieren von Ihrem Computer in RStudio Cloud

Das Importieren in von Ihrem Computer zu RStudio Cloud ist identisch zum Importieren von Ihrem Computer in RStudio Desktop. Nur dass Sie die Datendatei vorab hochladen müssen, schließlich ist RStudio Cloud in der Cloud und nicht auf Ihrem Computer. Klicken Sie dazu auf das Icon Upload im Reiter Files, s. Abbildung 4.19.

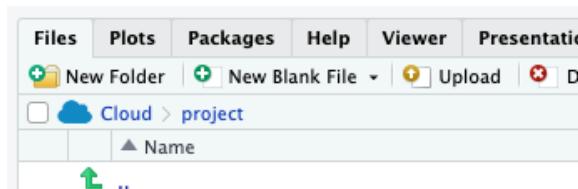


Abbildung 4.19.

Wählen Sie am besten den Ordner als Ziel, in dem sich auch die R-Datei, von der aus Sie den Befehl zum Daten importieren schreiben, befindet.

#### i Hinweis

Es gibt verschiedene Formate, in denen (Tabellen-)Dateien in einem Computer abgespeichert werden. Die gebräuchlichsten sind CSV und Excel. Es gibt auch mehrere R-Befehle, um Daten in R zu importieren, z.B. `read.csv()` oder `data_read()`. Praktischerweise kann der R-Befehl `data_read()` viele verschiedene Formate automatisch einlesen, so dass wir uns nicht weiter um das Format kümmern brauchen. Der Vorteil von `read.csv` ist, dass Sie kein Extra-Paket installiert bzw. gestartet haben müssen.

<sup>28</sup> .R- oder .qmd-Datei

<sup>29</sup>[https://youtu.be/B\\_nuN-M0pQM](https://youtu.be/B_nuN-M0pQM)

#### 4.8.5.5. Daten importieren per Klick

RStudio Desktops GUI (Benutzeroberfläche) erlaubt es Ihnen auch, Daten per Klick, also ohne R-Befehle, zu importieren, s. Abbildung 4.20.

Sie können über diese Maske sowohl CSV-Dateien, Excel-Dateien oder Daten-Dateien aus anderen Statistik-Programmen (z.B. SPSS) importieren auf diese Weise.

Zur Erinnerung: CSV-Dateien sind Textdateien, wählen Sie in dem Fall also From Text. Ich empfehle die Variante From Text (readr) . . .

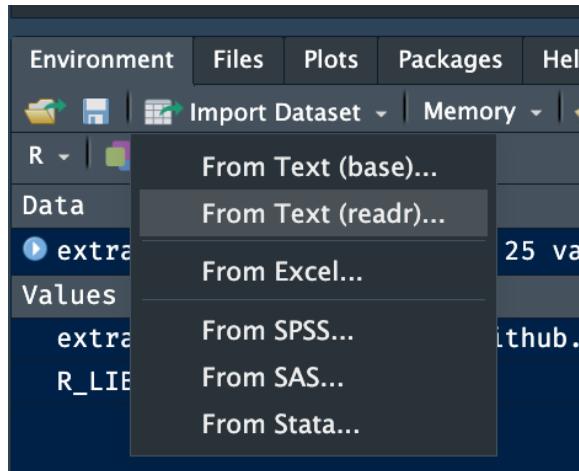


Abbildung 4.20.: Daten importieren per Klick

In der folgenden Maske können Sie unter `Browse` die zu importierende Datendatei auswählen. Mit Klick auf `Import` wird die Datei schließlich in R importiert.

#### 4.8.6. Dataframes

Eine in R importierte Tabelle (mit bestimmten Eigenschaften) heißt *Dataframe*. Dataframes sind in der Datenanalyse von großer Bedeutung.

**Definition 4.6** (Dataframe). Ein Dataframe (data frame; auch “Tibble” genannt<sup>30</sup>) ist ein Datenobjekt in R zur Darstellung von Tabellen. Dataframes bestehen aus einer oder mehreren Spalten. Spalten haben einen Namen, sozusagen einen “Spaltenkopf”. Alle Spalten müssen die gleiche Länge haben; anschaulich gesprochen ist eine Tabelle (in R) rechteckig. Jede Spalte einzeln betrachtet kann als Vektor aufgefasst werden. □

Tabelle 3.2 ist die Tabelle mit den Mario Kart-Daten; etwas präziser gesprochen ein Dataframe mit Namen `mariokart`. Übrigens ist Tabelle 3.2 in Normalform (Tidy-Format), vgl. Definition 3.9.

---

<sup>30</sup>von “tbl” wie Table

### **i Hinweis**

Geben Sie den Namen eines Dataframes ein, um sich den Inhalt anzeigen zu lassen. Beachten Sie, dass Sie die Daten auf diese Weise nur anschauen, nicht ändern können. □

#### **4.8.7. Tabellen in R betrachten**

Wenn Sie in R z.B. die Tabelle `mariokart` in einer Excel-typischen Ansicht betrachten wollen, klicken Sie am besten auf das Tabellen-Icon im Reiter *Environment*, gleich neben dem Namen `mariokart`, s. Abbildung 4.21.



Abbildung 4.21.: Per Klick auf das Tabellen-Icon können Sie eine Tabellenansicht der Tabelle `mariokart` öffnen

Alternativ öffnet der Befehl `View(mariokart)` die gleiche Ansicht.

#### **4.9. Logikprüfung**

💡 Wer will schon wieder wen prüfen?!

In diesem Abschnitt schauen wir uns *Logikprüfungen* an: Wir lassen R prüfen, ob eine Variable einen bestimmten Wert hat oder größer/kleiner als ein Referenzwert ist.

Definieren wir zuerst eine Variable, `x`.

```
x <- 42
```

Dann fragen wir R, ob diese Variable den Wert 42 hat.

```
x == 42
## [1] TRUE
```

👋 Hallo, Mensch. Ja, diese Variable hat den Wert 42.

(Danke, R.)

Möchte man mit R prüfen, ob eine Variable `x` einen bestimmten Wert ("Inhalt") hat, so schreibt man:

```
x == Wert.
```

### ! Wichtig

Man beachte das *doppelte* Gleichheitszeichen. Zur Prüfung auf Gleichheit muss man das doppelte Gleichheitszeichen verwenden.

### 🔥 Vorsicht

Ein beliebter Fehler ist es, bei der Prüfung auf Gleichheit, nur ein Gleichheitszeichen zu verwenden, z.B. so: `x = 73`. Mit einem Gleichheitszeichen prüft man aber *nicht* auf Gleichheit, sondern man definiert die Variable oder bestimmt ein Funktionsargument, s. Kapitel 4.7.1. □

Tabelle 4.1 gibt einen Überblick über wichtige Logikprüfungen in R.<sup>31</sup>

Tabelle 4.1.: Logische Prüfungen in R

| Prüfung.auf              | R-Syntax   |
|--------------------------|--|
| Gleichheit               | <code>x == Wert</code>                           |
| Ungleichheit             | <code>x != Wert</code>                           |
| Größer als Wert          | <code>x &gt; Wert</code>                         |
| Größer oder gleich Wert  | <code>x &gt;= Wert</code>                        |
| Kleiner als Wert         | <code>x &lt; Wert</code>                         |
| Kleiner oder gleich Wert | <code>x &lt;= Wert</code>                        |
| Logisches UND            | <code>(x &lt; Wert1) &amp; (x &gt; Wert2)</code> |
| Logisches ODER           | <code>(x &lt; Wert1)   (x &gt; Wert2)</code>     |

## 4.10. Praxisbezug

💡 R in der Praxis wirklich genutzt? Oder ist R nur der Traum von (vielleicht verwirrten) Profs im Elfenbeinturm?

Schauen wir uns dazu die Suchanfragen bei [stackoverflow.com](https://stackoverflow.com) an, dem größten FAQ-Forum für Software-Entwicklung. Wir vergleichen Suchanfragen mit dem Tag `[r]` zu Suchanfragen mit dem Tag `[spss]`<sup>32</sup>. Die Ergebnisse sind in Abbildung Abbildung 4.22 dargestellt.

Das ist grob gerechnet ein Faktor von 200 (der Unterschied von R zu SPSS). Dieses Ergebnis lässt darauf schließen, dass R in der Praxis viel mehr als Excel gebraucht wird.

💡 Aber ist R wirklich ein Werkzeug, das mir im Job hilft?

Viele Firmen weltweit nutzen R zur Datenanalyse.<sup>33</sup>

<sup>31</sup>Um das Zeichen für das logische ODER, `|` auf einer Mac-Tastatur zu erhalten, drückt man `Option+7`.

<sup>32</sup>Durchgeführt am 2022-02-24, 17:21 CET

<sup>33</sup>wie diese Liste zeigt: <https://www.quora.com/Which-organizations-use-R?share=1> zeigt

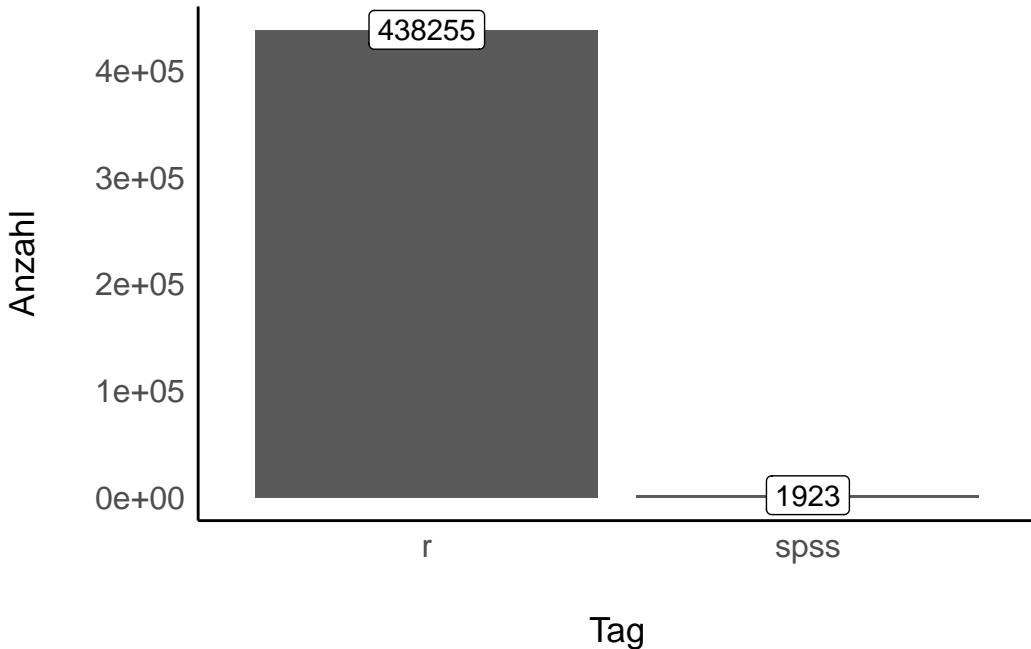


Abbildung 4.22.: Suchanfragen nach R bzw SPSS, Stand 2022-02-24

💡 R ist *der* Place-to-be für die Datenanalyse.

💡 Aber ist Datenanalyse wirklich etwas, womit ich in Zukunft einen guten Job bekomme?

Berufe mit Bezug zu Daten, Datenanalyse oder, allgemeiner, Künstlicher Intelligenz (artificial intelligence) gehören zu den am meisten wachsenden Berufen:

Artificial intelligence (AI) continues to make a strong showing on our Emerging Jobs lists, which is no surprise. Many jobs that have risen up as a result of AI in fields like cybersecurity and data science and because it's so pervasive many roles may demand more knowledge of AI than you may think. For example, real estate and business development roles. (Quelle: LinkedIn<sup>34</sup>)

## 4.11. Aufgaben

**Übungsaufgabe 4.4** (Statistik-Meme). Suchen Sie ein schönes Meme zum Thema Statistik, Datenanalyse und Data Science. [Hier](#) ist ein Startpunkt. □

Die Webseite [datenwerk.netlify.app](https://datenwerk.netlify.app) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

1. Typ-Fehler-R-01

<sup>34</sup><https://blog.linkedin.com/2019/december/10/the-jobs-of-tomorrow-linkedin-s-2020-emerging-jobs-report>

2. Typ-Fehler-R-02
3. Typ-Fehler-R-03
4. Typ-Fehler-R-04
5. Typ-Fehler-R-06a
6. Typ-Fehler-R-07
7. Typ-Fehler-R-08-name-clash
8. Logikpruefung1
9. Logikpruefung2
10. there-is-no-package
11. Wertberechnen2
12. Wertzuweisen\_mc
13. argumente
14. import-mtcars
15. Wertzuweisen
16. Wertpruefen
17. wrangle1
18. repro1-sessioninfo
19. mw-berechnen

Prüfen Sie Ihr Wissen mit [diesem Quiz!](#)<sup>35</sup>

Noch nicht genug? Checken Sie alle Aufgaben mit dem Tag R auf dem Datenwerk aus.<sup>36</sup>

#### **i Hinweis**

Die Webseite [Datenwerk](#) stellt eine Reihe von Aufgaben zum Thema Statistik bereit. Zu jeder Aufgabe sind ein oder mehrere Schlagwörter (Tags) zugeordnet. Wenn Sie auf ein Schlagwort klicken, sehen Sie die Liste der Aufgaben mit diesem Schlagwort. Es kann aber sein, dass Sie einige Aufgabe nicht lösen können, da Wissen vorausgesetzt wird, das Sie (noch) nicht haben. Lassen Sie sich davon nicht ins Boxhorn jagen. Ignorieren Sie solche Aufgaben fürs Erste. □

## 4.12. Vertiefung

### 4.12.1. Varianten zu `read.csv`

Hier ist eine weitere Möglichkeit, um Daten von einem Ordner (egal ob dieser sich im Internet oder auf Ihrem Computer befindet) einzulesen, stellt die Funktion `data_read` bereit:

```
library(easystats) # Das Paket muss installiert sein
d <- data_read(paste0(
  "https://vincentarelbundock.github.io/Rdatasets/",
  "csv/openintro/mariokart.csv"))
```

<sup>35</sup><https://datenwerk.netlify.app/posts/r-quiz/r-quiz>

<sup>36</sup><https://datenwerk.netlify.app/#category=R>

Der Unterschied ist, dass `data_read` viele Formate von Daten (Excel, CSV, SPSS, ...) verkraftet, wohingegen `read.csv` nur Standard-CSV einlesen kann.

Schauen wir uns die letzte R-Syntax en Detail an:

```
Hey R,  
hol das "Buch" easystats aus der Bücherei und lies es  
definiere als "d" die Tabelle,  
die du unter der angegebenen URL findest.
```

In R gibt es oft viele Möglichkeiten, ein Ziel zu erreichen. Zum Beispiel haben wir hier den Befehl `data_read()` verwendet, um Daten zu importieren. Andere, gebräuchliche Befehle, die CSV-Dateien importieren, heißen `read.csv()` (aus dem Standard-R, kein Extra-Paket nötig) und `read_csv()` (aus dem Meta-Paket `{tidyverse}`).

#### 4.12.2. Importieren von Excel-Tabellen

Mit der Funktion `data_read` aus `{easystats}` kann man viele verschiedene Datenformate importieren, auch Excel-Tabellen (`.xls`, `.xlsx`).

Als Beispiel betrachten wir den Datensatz `extra` aus dem R-Paket `{pradadata}`<sup>37</sup>. In diesem Datensatz werden die Ergebnisse einer Umfrage zu den Korrelaten von Extraversion beschrieben. Details zu der zugrundeliegenden Studie finden Sie hier: <https://osf.io/4kgzh>.

Ein Daten-Dictionary findet sich [hier](#).<sup>38</sup>

Laden Sie die Excel-Datei herunter. Angenommen, Sie speichern die Excel-Datei in einem Unterordner namens `daten` Ihres aktuellen Projektordners. Dann können Sie die Daten so importieren:

```
library(easystats)  
extra <- data_read("daten/extra.xls")
```

Allerdings kann `data_read` keine Dateien aus dem Internet importieren, was praktisch wäre. Stattdessen muss die Datei lokal auf Ihrer Festplatte liegen.

Wenn Sie allerdings "remote", also aus dem Internet, eine Excel-Datei importieren möchten, so können Sie das mit `import` aus dem R-Paket `{rio}` tun:

```
library(rio)  
extra_path <- paste0(  
  "https://github.com/sebastiansauer/statistik1/",  
  "raw/main/daten/extra.xls")  
extra <- import(extra_path)
```

<sup>37</sup><https://github.com/sebastiansauer/pradadata>

<sup>38</sup><https://github.com/sebastiansauer/statistik1/raw/main/daten/extra-dictionary.md>

### **i Hinweis**

CSV-Dateien werden auf vielen Computern als eine Datei erkannt, die Excel öffnen kann und das auch tut, wenn man eine CSV-Datei doppelklickt. Dennoch ist das CSV-Format keine Datei im Excel-Format, sondern eine einfache Text-Datei, die auch mit jedem Text-Editor geöffnet und bearbeitet werden kann. □

Alternativ können Sie in RStudio auch Excel-Dateien *ohne* R-Code importieren, s. Abbildung 4.20.

### **4.12.3. Der Dollar-Operator**

In Beispiel 4.2 hatten wir Vektoren definiert. Solche Vektoren fliegen sozusagen frei in Ihrem Environment herum (Schauen Sie mal dort nach!) Die Spalten einer Tabelle sind aber auch Vektoren, nur eben nicht frei im Environment, sondern in eine Tabelle eingebunden.

Möchte man diese Vektoren direkt ansprechen, so kann man das mit dem sog. *Dollar-Operator* \$ tun.

Angenommen, Sie möchten sich die Verkaufspreise (`total_pr`) aus der Tabelle `mariokart` herausziehen, dann können Sie das mit dem Dollar-Operator tun:

```
mariokart$total_pr
## [1] 51.55 37.04 45.50 44.00 71.00 45.00 37.02
## [8] 53.99 47.00 50.00 54.99 56.01 48.00 56.00
## [15] 43.33 46.00 46.71 46.00 55.99 326.51 31.00
## [22] 53.98 64.95 50.50 46.50 55.00 34.50 36.00
## [29] 40.00 47.00 43.00 31.00 41.99 49.49 41.00
## [36] 44.78 47.00 44.00 63.99 53.76 46.03 42.25
## [43] 46.00 51.99 55.99 41.99 53.99 39.00 38.06
## [50] 46.00 59.88 28.98 36.00 51.99 43.95 32.00
## [57] 40.06 48.00 36.00 31.00 53.99 30.00 58.00
## [64] 38.10 118.50 61.76 53.99 40.00 64.50 49.01
## [71] 47.00 40.10 41.50 56.00 64.95 49.00 48.00
## [78] 38.00 45.00 41.95 43.36 54.99 45.21 65.02
## [85] 45.75 64.00 36.00 54.70 49.91 47.00 43.00
## [92] 35.99 54.49 46.00 31.06 55.60 40.10 52.59
## [99] 44.00 38.26 51.00 48.99 66.44 63.50 42.00
## [106] 47.00 55.00 33.01 53.76 46.00 43.00 42.55
## [113] 52.50 57.50 75.00 48.92 45.99 40.05 45.00
## [120] 50.00 49.75 47.00 56.00 41.00 46.00 34.99
## [127] 49.00 61.00 62.89 46.00 64.95 36.99 44.00
## [134] 41.35 37.00 58.98 39.00 40.70 39.51 52.00
## [141] 47.70 38.76 54.51
```

Der Dollar-Operator trennt den Namen der Tabelle vom Namen der Spalte.

Natürlich können Sie mit dem resultierenden Vektor beliebig weiterarbeiten, etwa ihn in einem anderen Vektor speichern oder eine Funktion anwenden:

```
verkaufspreise <- mariokart$total_pr
mean(verkaufspreise)
## [1] 49.88049
mean(mariokart$total_pr) # synonym zur obigen Zeile
## [1] 49.88049
```

#### 4.12.4. R-Zertifikat bei LinkedIn

Sie können bei LinkedIn<sup>39</sup> (oder anderen Anbietern ein Zertifikat) erhalten, das Ihre R-Kenntnisse dokumentiert.

#### 4.12.5. R-Funktionen verschachteln

Das Kombinieren von Funktionen kann kompliziert werden:

---

##### **Listing 4.2** Verschachtelte Funktionen

---

```
x <- c(1, 2, 3)
sum(abs(mean(x)-x))
## [1] 2
```

---

Die Funktion `abs(x)` gibt den (Absolut-)Betrag von `x` zurück (entfernt das Vorzeichen, mit anderen Worten).

Verschachtelte Ausdrücke lesen sich von innen nach außen (und werden in dieser Reihenfolge abgearbeitet). Für unser Beispiel (Listing 5.2):

1. Berechne den Mittelwert von `x`
2. Ziehe vom Mittelwert jeweils die Elemente von `x` ab
3. Nimm vom Ergebnis jeweils den Absolutwert
4. Summiere diese Werte

Kurz gesagt: Hier haben wir die mittlere Absolutabweichung der Elemente von `x` zum Mittelwert ausgerechnet.

---

<sup>39</sup><https://www.linkedin.com/help/linkedin/answer/a510481>

#### 4.12.6. R und Friends update

Irgendwann werden Ihr R, Ihr RStudio und Ihre R-Pakete veraltet sein, s. Abbildung 4.23. Installieren Sie dann einfach die neue Version von R und RStudio wie oben beschreiben, s. Kapitel 4.3.

So updaten Sie Ihre R-Pakete: Klicken Sie im Reiter `Packages` (in RStudio) und dort auf den Button `Update`.<sup>40</sup> Denken Sie daran, dass Sie die Software (R, RStudio, R-Paket), die Sie updaten/-installieren, nicht laufen darf.

**i** Hinweis

Ihre R-Pakete sollten aktuell sein. Klicken Sie beim Reiter `Packages` auf “Update”, um Ihre R-Pakete zu aktualisieren. Arnold Schwarzenegger rät, Ihre R-Pakete aktuell zu halten, s. Abbildung 4.23<sup>41</sup>.

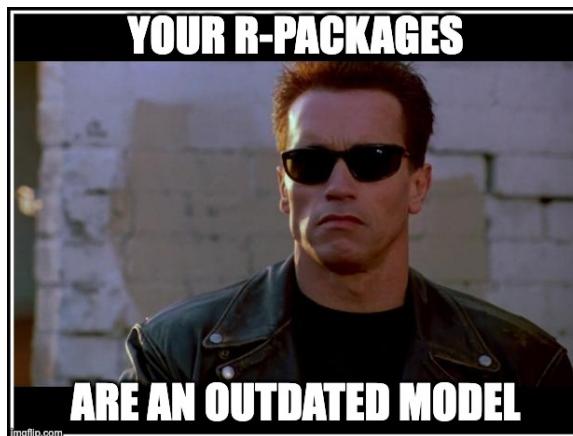


Abbildung 4.23.: R-Pakete sollten stets aktuell sein, so Arnold Schwarzenegger

#### 4.12.7. Benötigte R-Pakete

In diesem Kapitel benötigen Sie folgendes R-Paket:

```
library(openintro) # Datensatz `mariokart`
```

#### 4.12.8. Benötigte Daten

Sie benötigen in diesem Kapitel den Datensatz `mariokart`, der entweder online<sup>42</sup> oder über R-Paket `openintro` importiert werden kann:

<sup>40</sup>Wenn die Anzahl der zu aktualisierenden Pakete groß ist, dann besser nicht alle auswählen, sondern nur ein paar. Dann die nächsten paar Pakete usw.

<sup>41</sup>Bildquelle: <https://imgflip.com/memegenerator>

<sup>42</sup>über diese Internetadresse: <https://vincentarelbundock.github.io/Rdatasets/csv/openintro/mariokart.csv>

#### 4.12.8.1. Import via Download

```
mariokart <- read.csv(paste0(  
  "https://vincentarelbundock.github.io/Rdatasets/",  
  "csv/openintro/mariokart.csv"))
```

#### 4.12.8.2. Import via R-Paket

```
# Das Paket 'openintro' muss installiert sein:  
data(mariokart, package = "openintro")
```

### 4.13. Literaturhinweise

“Warum R? Warum, R?” heißt ein Kapitel in Sauer (2019), das einiges zum Pro und Contra von R ausführt. In Kapitel 3 in der gleichen Quelle finden sich viele Hinweise, wie man R startet; In Kapitel 4 werden Grundlagen von “Errisch” erläutert; Kapitel 5 führt in Datenstrukturen von R ein (schon etwas anspruchsvoller). Alternativ bietet [Kapitel 1](#) von Ismay und Kim (2020) einen guten und sehr anwenderfreundlichen Überblick. Das Buch hat auch den Vorteil, dass es komplett frei online verfügbar ist. Vergleichbar dazu ist Cetinkaya-Rundel und Hardin (2021), vielleicht einen Tick formaler; auf jeden Fall genau das richtige Niveau für Bachelor-Statistik in angewandten nicht-technischen Studiengängen.

Natürlich gibt es viele Online-Kurse zu R, die aber teilweise kostenpflichtig sind<sup>43</sup>.

### 4.14. Literatur

<sup>43</sup>Ein Beispiel ist der Kurs *Getting Started with RStudio*, <https://www.coursera.org/projects/getting-started-rstudio> (Kursdauer: 1 Stunde)

# 5. Daten umformen

## 5.1. Lernsteuerung

### 5.1.1. Standort im Lernpfad

Abb. Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 5.1.2. Lernziele

- Sie können folgende Verben des Datenjudo anwenden: `arrange`, `filter`, `select`, `summarise`, `group_by`, `mutate`.
- Sie können R-Befehle mit der “Pfeife” verketten.

### 5.1.3. Benötigte R-Pakete

```
library(tidyverse)
library(easystats)
```

### 5.1.4. Benötigte Daten

```
mariokart <- paste0(
  "https://vincentarelbundock.github.io/Rdatasets/",
  "csv/openintro/mariokart.csv")

mariokart <- read.csv(mariokart_path)
```

**Übungsaufgabe 5.1.** Importieren Sie den Datensatz `mariokart`. Probieren Sie alle vorgestellten Verben des Datenjudos aus, zunächst in dem Sie einfach den R-Code kopieren und selber ausführen.□

### **5.1.5. R-Code zum Copy-Pasten**

Sie finden den R-Code für jedes Kapitel [hier](#). □

### **5.1.6. Frag den Bot**

💡 Erkläre einem Fünfjährigen, warum Statistik nützlich ist.

🤖 Okay, hör zu. Stell dir vor, du hast viele Spielzeugautos und du willst herausfinden, welches Auto am meisten bespielt wird. Du kannst jeden Tag zählen, wie oft du jedes Auto benutzt hast. Aber das wäre sehr mühsam und könnte schnell durcheinander gebracht werden. Stattdessen kannst du mit Statistik eine einfachere Methode finden. Du kannst zum Beispiel eine Liste machen und jedes Auto mit einer bestimmten Farbe markieren. Dann kannst du jeden Tag einfach zählen, wie oft du jede Farbe verwendet hast. Am Ende kannst du herausfinden, welches Auto am meisten bespielt wurde, indem du siehst, welche Farbe am meisten verwendet wurde. Das ist ein Beispiel dafür, wie Statistik uns dabei helfen kann, Muster in großen Datensätzen zu finden und Entscheidungen zu treffen, ohne alles einzeln zu zählen.

Quelle: ChatGTP 3.5, 2023-02-09

### **5.1.7. Zum Einstieg**

Vielleicht fordert Sie die Lehrkraft zu einem Einstiegsquiz auf, etwas mittels der Plattform [antwortet.jetzt](#).<sup>1</sup> Alternativ überlegen Sie sich selber 10 Quiz-Aufgaben zum Stoff des letzten Kapitels.

## **5.2. Datenjudo**

**Definition 5.1** (Datenjudo). Mit *Datenjudo* meint man den Prozess der Aufbereitens, Umformens oder Zusammenfassen von Daten, sowohl für einzelne Beobachtungen (Zeilen einer Datentabelle) oder Variablen (Spalten einer Datentabelle) oder einer ganzen Datentabelle. □

### **5.2.1. Die Wahrheit über Data Science**

Denkt man an Data Science, stellt man sich coole Leute vor (in San Francisco oder Berlin), die an abgefahrenen Berechnungen mit hoch komplexen statistischen Modellen für gigantische Datenmengen basteln. Tatsächlich besteht ein großer Teil der Arbeit aus dem Aufbereiten von Daten.

---

<sup>1</sup><https://antworte.jetzt/>

### 5.2.2. Praxisbezug: Aus dem Alltag des Data Scientisten

Laut dem [Harvard Business Review](#) allerdings, verbringen diese Leute “80%” ihrer Zeit mit dem *Aufbereiten* von Daten (Bowne-Anderson 2018).<sup>2</sup> Ja: mit uncoolen Tätigkeiten wie Tippfehler aus Datensätzen entfernen oder die Daten überhaupt nutzbar und verständlich zu machen.

Das zeigt zumindest, dass das Aufbereiten von Daten a) wichtig ist und b) dass man allein damit schon weit kommen kann. Eine gute Nachricht ist (vielleicht), dass das Aufbereiten von Daten keine aufwändige Mathematik verlangt, stattdessen muss man ein paar Handgriffe und Kniffe kennen. Daher passt der Begriff *Datenjudo* vielleicht ganz gut. Kümmern wir uns also um das Aufbereiten bzw. Umformen von Daten, um das Datenjudo. □□□

**Beispiel 5.1.** Beispiele für typische Tätigkeiten des Datenjudos sind:

- Zeilen *filtern* (z. B. nur Studenten des Studiengangs X)
- Zeilen *sortieren* (z. B. Studenten mit guten Noten in den oberen Zeilen)
- Spalten *wählen* (z. B. 100 weitere Produkte ausblenden)
- Spalten in eine Zahl *zusammenfassen* (z. B. Notenschnitt der 1. Klausur)
- Tabelle *gruppieren* (z. B. Analyse getrennt nach Standorten)
- Werte aus einer Spalte *verändern* oder *neue Spalte* bilden (z. B. Punkte in Prozent-Richtige umrechnen).
- ... □

### 5.2.3. Mach's einfach

Es gibt einen (einfachen) Trick, wie man umfangreiche Datenaufbereitung elegant geregelt kriegt, klingt fast zu schön, um wahr zu sein (s. Abbildung 5.1).

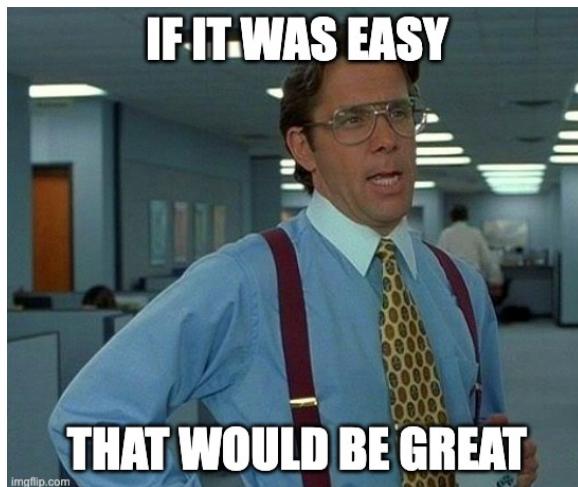


Abbildung 5.1.: Mach's einfach. Made at imgflip.com, Meme Generator

<sup>2</sup><https://hbr.org/2018/08/what-data-scientists-really-do-according-to-35-data-scientists>

Der Trick besteht darin, komplexe Operationen in mehrere einfache Teilschritte zu zergliedern<sup>3</sup>. Man könnte vom “Lego-Prinzip” sprechen, s. Abbildung 5.2. Im linken Teil von Abbildung 5.2 sieht man ein (recht) komplexes Gebilde. Zerlegt man es aber in seine Einzelteile, so sind es deutlich einfachere geometrische Objekte wie Dreiecke oder Quadrate (rechter Teil des Diagramms).



Abbildung 5.2.: Das Lego-Prinzip

Damit Sie es selber einfach machen können, müssen Sie selber Hand anlegen. Importieren Sie daher den Datensatz `mariokart`, z.B. so:

```
mariokart <- read.csv(mariokart_path)

glimpse(mariokart)
## Rows: 143
## Columns: 13
## $ rownames      <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,
##   ↵ 13, 14, 15, 16, 17, ~
## $ id            <dbl> 1.5e+11, 2.6e+11, 3.2e+11, 2.8e+11,
##   ↵ 1.7e+11, 3.6e+11, 1.2e~
## $ duration      <int> 3, 7, 3, 3, 1, 3, 1, 1, 3, 7, 1, 1, 1,
##   ↵ 1, 7, 7, 3, 3, 1, 7~
## $ n_bids        <int> 20, 13, 16, 18, 20, 19, 13, 15, 29, 8,
##   ↵ 15, 15, 13, 16, 6, ~
## $ cond          <chr> "new", "used", "new", "new", "new",
##   ↵ "new", "used", "new", ~
## $ start_pr      <dbl> 0.99, 0.99, 0.99, 0.99, 0.01, 0.99,
##   ↵ 0.01, 1.00, 0.99, 19.9~
## $ ship_pr       <dbl> 4.0, 4.0, 3.5, 0.0, 0.0, 4.0, 0.0, 3.0,
##   ↵ 4.0, 4.0, 3.0, 0.0~
## $ total_pr      <dbl> 52, 37, 46, 44, 71, 45, 37, 54, 47, 50,
##   ↵ 55, 56, 48, 56, 43~
## $ ship_sp        <chr> "standard", "firstClass", "firstClass",
##   ↵ "standard", "media~
## $ seller_rate   <int> 1580, 365, 998, 7, 820, 270144, 7284,
##   ↵ 4858, 27, 201, 4858, ~
```

<sup>3</sup>Genau darin besteht das Wesen einer Analyse: die Zerlegung eines Objekts in seine Bestandteile.

```

## $ stock_photo <chr> "yes", "yes", "no", "yes", "yes",
#> "yes", "yes", "yes", "ye~
## $ wheels      <int> 1, 1, 1, 1, 2, 0, 0, 2, 1, 1, 2, 2, 2,
#> 2, 1, 0, 1, 1, 2, 2~
## $ title       <chr> "~~~ Wii MARIO KART & WHEEL ~
#> NINTENDO Wii ~ BRAND NEW ~

```

**Beispiel 5.2.** Sie arbeiten immer noch bei dem großen Online-Auktionshaus. Mittlerweile haben Sie sich den Ruf des “Datenguru” erworben. Vielleicht weil Sie behauptet haben, Data Science sei zu 80% Datenjudo, das hat irgendwie Eindruck geschindet... Naja, jedenfalls müssen Sie jetzt mal zeigen, dass Sie nicht nur schlaue Sprüche draufhaben, sondern auch die Daten ordentlich abbürsten können. Sie analysieren dafür im Folgenden den Datensatz `mariokart`. Na, dann los.□

### 5.3. Die Verben des Datenjudos

Im R-Paket `{dplyr}`, das wiederum Teil des R-Pakets `{tidyverse}` ist, gibt es eine Reihe von R-Befehlen, die das Datenjudo in eine Handvoll einfacher Verben unterbrechen.<sup>4</sup> Die wichtigsten Verben des Datenjudos schauen wir uns im Folgenden an.

Wir betrachten dazu im Folgenden einen einfachen (Spielzeug-)Datensatz, an dem wir zunächst die Verben des Datenjudos vorstellen, s. Tabelle 5.1.

Tabelle 5.1.: Ein einfacher Datensatz von schlichtem Gemüt

|  | id | name   | gruppe | note |
|--|----|--------|--------|------|
|  | 1  | Anni   | A      | 2.7  |
|  | 2  | Berti  | A      | 2.7  |
|  | 3  | Charli | B      | 1.7  |

#### ! Wichtig

Die Verben des Datenjudos wohnen im Paket `{dplyr}`, welches gestartet wird, wenn Sie `library(tidyverse)` eingeben. Falls Sie vergessen, das Paket `{tidyverse}` zu starten, dann funktionieren diese Befehle nicht.□

#### i Hinweis

Zur Erinnerung: In RStudio können Sie per Klick auf das kleine Tabellen-Icon im Bereich *Environment* die Tabellenansicht einer Tabelle öffnen, s. Kapitel 4.8.7. □

<sup>4</sup>Falls Sie das R-Paket `{tidyverse}` noch nicht installiert haben sollten, wäre jetzt ein guter Zeitpunkt dafür.

### 5.3.1. Tabelle sortieren: `arrange`

*Sortieren* der Zeilen ist eine einfache, aber häufige Tätigkeit des Datenjudos, s. Abbildung 5.3.



| <b>id</b> | <b>name</b> | <b>gruppe</b> | <b>note</b> | <b>arrange()</b> | <b>id</b> | <b>name</b> | <b>gruppe</b> | <b>note</b> |
|-----------|-------------|---------------|-------------|------------------|-----------|-------------|---------------|-------------|
| 1         | Anni        | A             | 2.7         |                  | 3         | Charli      | B             | 1.7         |
| 2         | Berti       | A             | 2.7         |                  | 1         | Anni        | A             | 2.7         |
| 3         | Charli      | B             | 1.7         |                  | 2         | Berti       | A             | 2.7         |

Abbildung 5.3.: Sinnbild für das Sortieren einer Tabelle mit `arrange()`

**Beispiel 5.3** (Was sind die höchsten Preise?). Sie wollen mal locker anfangen. Daher stellen Sie sich folgende Frage: Was sind denn eigentlich die höchsten Preise, für die das Spiel *Mariokart* über den Online-Ladentisch geht? Die Spalte des Verkaufspreis heißt offenbar `total_pr` (s. Tabelle `mariokart`). In Excel kann die Spalte, nach der man die Tabelle sortieren möchte, einfach anklicken. Ob das in R auch so einfach geht? Die Funktion `arrange()` macht es uns ziemlich einfach:

```
arrange(mariokart, total_pr)
```

| total_pr | start_pr |
|----------|----------|
| 29       | 0.99     |
| 30       | 0.01     |
| 31       | 0.99     |
| 31       | 1.99     |
| 31       | 30.00    |
| 31       | 0.01     |

Übersetzen wir die R-Syntax ins Deutsche:

Hey R,  
arrangiere (sortiere) `mariokart` nach der Spalte `total\_pr`.

Gar nicht so schwer.□

Übrigens wird in `arrange()` per Voreinstellung aufsteigend sortiert. Setzt man ein Minus vor der zu sortierenden Spalte, wird umgekehrt, also *absteigend* sortiert:

```
mario_sortiert <- arrange(mariokart, -total_pr)
```

**Übungsaufgabe 5.2.** Sortieren Sie die Mariokart-Daten absteigend nach der Anzahl der beigelegten Lenkräder.□

### 5.3.2. Zeilen filtern: `filter`

#### 5.3.2.1. Nur bestimmte Zeilen behalten

Zeilen *filtern* bedeutet, dass man nur *bestimmte Zeilen* (Beobachtungen) *behalten* möchte, die restlichen Zeilen brauchen wir nicht, weg mit ihnen. Wir haben also ein Filterkriterium im Kopf, anhand dessen wir die Tabelle filtern, s. Abbildung 5.4.

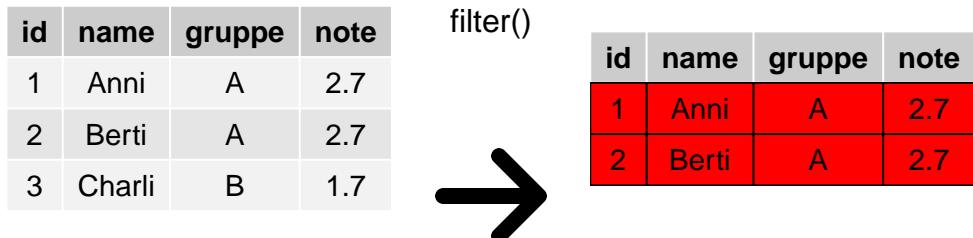


Abbildung 5.4.: Sinnbild für das Filtern einer Tabelle mit `filter()`

**Beispiel 5.4** (Ob ein Foto für den Verkaufspreis nützlich ist?). Als nächstes kommt Ihnen die Idee, mal zu schauen, ob Auktionen mit Photo der Ware einen höheren Verkaufspreis erzielen als Auktionen ohne Photo.

```
mariokart_neu <- filter(mariokart, stock_photo == "yes")
```

Sie filtern also die Tabelle so, dass *nur* diese Auktionen im Datensatz verbleiben, welche ein Photo haben, mit anderen Worten, Auktionen (Beobachtungen) bei denen gilt: `stock_photo == TRUE`.□

#### 5.3.2.2. Komplexeres Filtern

Angestachelt von Ihren Erfolgen möchten Sie jetzt komplexere Hypothesen prüfen: Ob wohl Auktionen von *neuen* Spielen und zwar *mit* Photo einen höheren Preis erzielen als die übrigen Auktionen?

Anders gesagt haben Sie zwei Filterkriterien im Blick: Neuheit `cond` und Photo `stock_photo`. Nur diejenigen Auktionen, die *sowohl* Neuheit *als auch* Photo erfüllen, möchten Sie näher untersuchen (Filtern mit dem logischen UND):

```
mario_filter1 <- filter(mariokart, stock_photo == "yes" & cond  
                           == "new")
```

Hm. Was ist mit den Auktionen, die *entweder* über ein Photo verfügen *oder auch* neu sind, oder beides (Filtern mit dem logischen ODER)?

```
mario_filter2 <- filter(mariokart,
                        stock_photo == "yes" | cond == "new")
```

Zur Erinnerung: Logische Operatoren sind in Kapitel 4.9 erläutert.

Hier könnte man noch viele interessante Hypothesen prüfen, denken Sie sich und tun das auch ...

**Übungsaufgabe 5.3.** Filtern Sie die Spiele mit nur einem Lenkrad und ohne Versandkosten.□

**Übungsaufgabe 5.4.** Filtern Sie die Spiele mit nur einem Lenkrad, die einen überdurchschnittlichen Verkaufspreis erzielen. Tipp: Nutzen Sie die Funktion `describe_distribution(name_der_tabelle)`, um den Mittelwert einer Variable des Datensatzes zu erfahren (diese Funktion wohnt im R-Paket `easystats`).□

### 5.3.3. Spalten auswählen mit `select`

Eine Tabelle mit vielen Spalten kann schnell unübersichtlich werden. Da lohnt es sich, eine alte goldene Regel zu beachten: Mache die Dinge so einfach wie möglich, aber nicht einfacher. Wählen wir also *nur* die Spalten aus, die uns interessieren und entfernen wir die restlichen, s. Abbildung 5.5.

| <b>id</b> | <b>name</b> | <b>gruppe</b> | <b>note</b> |
|-----------|-------------|---------------|-------------|
| 1         | Anni        | A             | 2.7         |
| 2         | Berti       | A             | 2.7         |
| 3         | Charli      | B             | 1.7         |

select()

| <b>id</b> | <b>note</b> |
|-----------|-------------|
| 1         | 2.7         |
| 2         | 2.7         |
| 3         | 1.7         |

Abbildung 5.5.: Sinnbild für das Auswählen von Spalten mit `select()`

**Beispiel 5.5** (Fokus auf nur zwei Spalten). Ob wohl gebrauchte Spiele deutlich geringere Preise erzielen im Vergleich zu neuwertigen Spielen? Sie entschließen sich, mal ein Stündchen auf die relevanten Daten zu starren.

```
mario_select1 <- select(mariokart, cond, total_pr)
```

Aha (?)□

Der Befehl `select` erwartet als Input eine Tabelle und gibt (als Output) eine Tabelle zurück – genau wie die meisten anderen Befehle des Datenjudos. Auch wenn Sie nur eine Spalte auswählen, bleibt es eine Tabelle, eben eine Tabelle mit nur einer Spalte.

`select` erlaubt Komfort; Sie können Spalten auf mehrere Arten auswählen, z.B.

```
select(mariokart, 1, 2) # Spalte 1 und 2
select(mariokart, 2:5) # Spalten 2 *bis* 5
select(mariokart, -1) # Alle Spalte *aber nicht* Spalte 1
```

**Übungsaufgabe 5.5.** Wählen Sie die Spalten `total_pr`, `cond` sowie die zweite Spalte der Tabelle `mariokart` aus!<sup>5</sup> □

Vertiefte Informationen zum Auswählen von Spalten mit `select` findet sich [hier](#).<sup>6</sup>

#### 5.3.4. Spalten zu einer Zahl zusammenfassen mit `summarise`

So eine lange Spalte mit Zahlen – mal ehrlich: wer blickt da schon durch? Viel besser wäre es doch, die Spalte `total_pr` zu einer Zahl zusammenzufassen, das ist doch viel handlicher. Kurz entschlossen fassen Sie die Spalte `total_pr`, den Verkaufspreis, zum Mittelwert zusammen, s. Abbildung 5.6.

The diagram illustrates the process of summarizing data. On the left, there is a data frame with four columns: `id`, `name`, `gruppe`, and `note`. The data consists of three rows:

| <code>id</code> | <code>name</code> | <code>gruppe</code> | <code>note</code> |
|-----------------|-------------------|---------------------|-------------------|
| 1               | Anni              | A                   | 2.7               |
| 2               | Berti             | A                   | 2.7               |
| 3               | Charli            | B                   | 1.7               |

To the right of the data frame, the `summarise()` function is shown, followed by a large arrow pointing to the result. The result is a single value labeled `note_mw` with a value of `2.4`.

Abbildung 5.6.: Spalten zu einer einzelnen Zahl zusammenfassen mit `summarise()`

**Beispiel 5.6** (Was ist der mittlere Verkaufspreis?). Mit `summarise`, s. Listing 5.1, können wir den mittleren Verkaufspreis der MarioKart-Spiele berechnen.

---

**Listing 5.1** Die R-Funktion `summarise` fasst einen Vektor zu einer Zahl zusammen

---

```
mariokart_mittelwert <- summarise(mariokart,
                                    preis_mw = mean(total_pr))
mariokart_mittelwert
```

---

$$\overline{\overline{\text{preis\_mw}}} \\ \underline{\underline{50}}$$

Aha! Etwa 50€ erzielt so eine Auktion im Schnitt.□

<sup>5</sup>`select(mariokart, total_pr, cond, 2)`

<sup>6</sup>[https://tidyverse.org/reference/tidyr\\_tidy\\_select.html](https://tidyverse.org/reference/tidyr_tidy_select.html)

Übersetzen wir Listing 5.1 vom Errischen ins Deutsche:

Hey R, fasse die Zeilen von total\_pr aus mariokart zu einer Zahl zusammen, und zwar mit Hilfe des Mittelwerts. Die resultierende Tabelle nennen wir mariokart\_mittelwert, sehr kreativ. Und die resultierende Spalte, die einzige inmariokart\_mittelwert, nennen wirpreis\_mw<sup>7</sup>.

Ein bisschen abstrakter gesprochen, fasst summarise also eine Spalte zu einer (einzelnen) Zahl zusammen, s. Gleichung 5.1.<sup>7</sup> Auf welche Art zusammengefasst werden soll, z.B. anhand des Mittelwerts oder Maximalwerts, muss noch zusätzlich innerhalb von summarise angegeben werden.



**Übungsaufgabe 5.6.** Identifizieren Sie den höchsten Kaufpreis eines Mario Kart-Spiels!<sup>8</sup> □

### 5.3.5. Tabelle gruppieren

Es ist ja gut und schön, zu wissen, was so ein Spiel im Schnitt kostet. Aber viel interessanter wäre es doch, denken Sie sich, zu wissen, ob die neuen Spiele im Schnitt mehr kosten als die alten? Ob R Ihnen so etwas ausrechnen kann?

👉 Ich tue fast alles für dich. ❤️

Also gut, R, dann gruppiere die Tabelle, s. Abbildung 5.7.

Durch das Gruppieren wird die Tabelle in “Teiltabellen” – entsprechend der Gruppen – aufgeteilt. Das sieht man der R-Tabelle aber nicht wirklich an. Aber alle nachfolgenden Berechnungen werden für jede Teiltabelle einzeln ausgeführt.

**Beispiel 5.7** (Mittlerer Preis pro Gruppe). Gruppieren alleine liefert Ihnen zwei (oder mehrere) Teiltabellen, etwa neue Spiele (Gruppe 1, new) vs. gebrauchte Spiele (Gruppe 2, used). Mit anderen Worten: Wir gruppieren anhand der Variable cond.

```
mariokart_gruppiert <- group_by(mariokart, cond)
```

<sup>7</sup>Eine Alternative, um eine Spalte zu einer Zahl zusammenzufassen, bietet der “Dollar-Operator” (\$): mean(mariokart\$total\_pr). Der Dollar-Operator trennt hier die Tabelle von der Spalte: tibble\$spalte. Im Gegensatz zu den Verben des Tidyverse (die immer einer Tabelle zurückliefern), liefert der Dollar-Operator einen Vektor (Spalte) zurück. (Diese wird von mean dann zu einer einzelnen Zahl zusammengefasst.)

<sup>8</sup>summarise(mariokart, hoechster\_preis = max(total\_pr))

The diagram illustrates the process of grouping data. On the left, a table shows three rows of data:

| <b>id</b> | <b>name</b> | <b>gruppe</b> | <b>note</b> |
|-----------|-------------|---------------|-------------|
| 1         | Anni        | A             | 2.7         |
| 2         | Berti       | A             | 2.7         |
| 3         | Charli      | B             | 1.7         |

Below this table is the text "group\_by(gruppe)". An arrow points to the right, leading to another table:

| <b>id</b> | <b>name</b> | <b>gruppe</b> | <b>note</b> |
|-----------|-------------|---------------|-------------|
| 3         | Charli      | B             | 1.7         |

The cells in the "gruppe" column for rows 1 and 2 are highlighted in blue, while the cell for row 3 is highlighted in green.

Abbildung 5.7.: Gruppieren von Datensätzen mit `group_by()`

Wenn Sie die neue Tabelle betrachte, sehen Sie wenig Aufregendes, nur einen Hinweis, dass die Tabelle gruppiert ist. Jetzt können Sie an jeder Teiltabelle Ihre weiteren Berechnungen vornehmen, etwa die Berechnung des mittleren Verkaufspreises.

```
summarise(mariokart_gruppiert, preis_mw = mean(total_pr))
```

| cond | preis_mw |
|------|----------|
| new  | 54       |
| used | 47       |

Langsam fühlen Sie sich als Datenchecker ... ☐ ☐ ♀ ☐

### Übungsaufgabe 5.7.

#### 5.3.6. Aufgabe

Berechnen Sie den mittleren und maximalen Verkaufspreis getrennt für Spiele mit und ohne Foto!

#### 5.3.7. Lösung

```
mariokart_gruppiert_foto <- group_by(mariokart, stock_photo)

mariokart_verkaufspreis_foto <-
  summarise(mariokart_gruppiert_foto,
            total_pr_avg = mean(total_pr),
```

```

total_pr_max = max(total_pr))

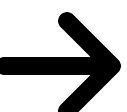
mariokart_verkaufspreis_foto

```

|     | stock_photo | total_pr_avg | total_pr_max |
|-----|-------------|--------------|--------------|
| no  |             | 54           | 327          |
| yes |             | 48           | 75           |

### 5.3.8. Spalten verändern mit `mutate`

Immer mal wieder möchte man *Spalten verändern*, bzw. deren Werte umrechnen, s. Abbildung 5.8.



| <code>id</code> | <code>name</code> | <code>gruppe</code> | <code>note</code> | <code>mutate()</code> | <code>id</code> | <code>name</code> | <code>gruppe</code> | <code>note</code> | <code>punkte</code> |
|-----------------|-------------------|---------------------|-------------------|-----------------------|-----------------|-------------------|---------------------|-------------------|---------------------|
| 1               | Anni              | A                   | 2.7               |                       | 1               | Anni              | A                   | 2.7               | 73                  |
| 2               | Berti             | A                   | 2.7               |                       | 2               | Berti             | A                   | 2.7               | 72                  |
| 3               | Charli            | B                   | 1.7               |                       | 3               | Charli            | B                   | 1.7               | 89                  |

Abbildung 5.8.: Spalten verändern/neu berechnen mit `mutate()`

**Beispiel 5.8.** Der Hersteller des Computerspiels *Mariokart* kommt aus Japan; daher erscheint es Ihnen opportun für ein anstehendes Meeting mit dem Hersteller die Verkaufspreise von Dollar in japanische Yen umzurechnen. Nach etwas Googeln finden Sie einen Umrechnungskurs von 1:133.

```

mariokart2 <- mutate(mariokart, total_pr_yen = total_pr * 133)
mariokart2 <- select(mariokart2, total_pr_yen, total_pr)
mariokart2 |> head() # nur die ersten paar Zeilen

```

|  | total_pr_yen | total_pr |
|--|--------------|----------|
|  | 6856         | 52       |
|  | 4926         | 37       |
|  | 6052         | 46       |
|  | 5852         | 44       |
|  | 9443         | 71       |
|  | 5985         | 45       |

Sicherlich werden Sie Ihre Gesprächspartner schwer beeindrucken.□

Mit `mutate` berechnen Sie eine Spalte `x` (in einer Tabelle) neu. Die Funktion, die Sie in `mutate` benennen wird für jede Zeile der Spalte `x` angewendet.

**Beispiel 5.9** (Beispiele für Funktionen für `mutate`). `mutate` eignet sich, z.B. um Spalten zu addieren, zu multiplizieren oder sonstwie zu transformieren (z.B. den Logarithmus anwenden oder den Mittelwert der Spalte von jeder Zeile abziehen). □

### Übungsaufgabe 5.8.

#### 5.3.9. Aufgabe

Rechnen Sie die Dauer der Auktionen von Tagen in Wochen um.

#### 5.3.10. Lösung

```
mariokart_duration_wochen <-  
  mutate(mariokart, duration_week = duration / 7)  
  
mariokart_duration_wochen <-  
  select(mariokart_duration_wochen, duration, duration_week)  
mariokart_duration_wochen |> head()  # nur die ersten paar  
  ↵ Zeilen
```

| duration | duration_week |
|----------|---------------|
| 3        | 0.43          |
| 7        | 1.00          |
| 3        | 0.43          |
| 3        | 0.43          |
| 1        | 0.14          |
| 3        | 0.43          |

### Übungsaufgabe 5.9.

#### 5.3.11. Aufgabe

Rechnen Sie wieder die Dauer der Auktionen von Tagen in Wochen um, aber runden Sie die Wochen auf ganze Wochen.

#### 5.3.12. Lösung

```

mariokart_duration_wochen <-
  mutate(mariokart, duration_week = duration / 7)

mariokart_duration_wochen_gerundet <-
  mutate(mariokart_duration_wochen, duration_week_gerundet =
         round(duration_week, digits = 0))

mariokart_duration_wochen_schmal <-
  select(mariokart_duration_wochen_gerundet, duration,
         duration_week, duration_week_gerundet)
mariokart_duration_wochen_schmal |> head()

```

| duration | duration_week | duration_week_gerundet |
|----------|---------------|------------------------|
| 3        | 0.43          | 0                      |
| 7        | 1.00          | 1                      |
| 3        | 0.43          | 0                      |
| 3        | 0.43          | 0                      |
| 1        | 0.14          | 0                      |
| 3        | 0.43          | 0                      |

⚠️ Statistik, wann braucht man schon sowas!?

💡 Eigentlich nur dann, wenn man die Fakten gut verstehen will, sonst nicht.

### 5.3.13. Zeilen zählen mit count

Arbeitet man mit nominalskalierten Daten, ist (fast) alles, was man tun kann, das Zeilen zählen.<sup>9</sup>

Man könnte z.B. fragen, wie viele neue und wie viele alte Spiele in der Tabelle (Dataframe) mariokart vorhanden sind.

**Beispiel 5.10.** Nach der letzten Präsentation Ihrer Analyse hat Ihre Chefin gestöhnt: “Oh nein, alles so kompliziert. Statistik! Himmel hilf! Kann man das nicht einfacher machen?” Anstelle von irgendwelchen komplizierten Berechnungen (Mittelwert?) möchten Sie ihr beim nächsten Treffen nur zeigen, wie viele Computerspiele neu und wie viele gebraucht sind (in Ihrem Datensatz). Schlichte Häufigkeiten also. Hoffentlich ist Ihre Chefin nicht wieder überfordert...

```

mariocart_counted <- count(mariokart, cond)
mariocart_counted

```

---

<sup>9</sup>Ja, das ist traurig.

| cond | n  |
|------|----|
| new  | 59 |
| used | 84 |

Aha! Es gibt mehr gebrauchte als neue Spiele.  $\square$

Jetzt könnte man noch den *Anteil* (engl. *proportion*) ergänzen: Welcher *Anteil* (der 143 Spiele in mariokart) ist neu, welcher gebraucht?

```
mutate(mariocart_counted, Anteil = n / sum(n))
```

| cond | n  | Anteil |
|------|----|--------|
| new  | 59 | 0.41   |
| used | 84 | 0.59   |

**Übungsaufgabe 5.10.** Zählen Sie Sie, wie viele Auktionen ein Foto enthalten.<sup>10</sup>  $\square$

**Übungsaufgabe 5.11.** Zählen Sie Sie, wie viele Auktionen ein Foto enthalten – innerhalb der gebrauchten Spiele und innerhalb der neuen Spiele. Anders gesagt: Teilen Sie den Datensatz sowohl nach Zustand als auch nach Foto auf und zählen Sie jeweils, wie viele Spiele/Auktionen in die jeweilige Gruppe gehören.<sup>11</sup>  $\square$

#### 5.3.14. Fazit: Verben am Fließband

die Befehle (“Verben”) des Tidyverse sind jeweils für einzelne, typische Aufgaben des Datenaufbereitens (“Datenjudo”) zuständig.

Typischerweise erwarten diese Befehle eine Tabelle ( $\square$ ) als Input und liefern eine Tabelle aus Output zurück, s. Abbildung 5.9.



Abbildung 5.9.: Tidyverse-Befehle erwarten normalerweise eine Tabelle (tibble) als Input und geben auch eine Tabelle zurück als Output

<sup>10</sup>count(mariokart, stock\_photo)

<sup>11</sup>count(mariokart, stock\_photo, cond)

## 5.4. Die Pfeife

Das ist keine Pfeife, wie René Magritte 1929 in seinem [berühmten Bild](#) schrieb, s. Abbildung 5.10.<sup>12</sup>

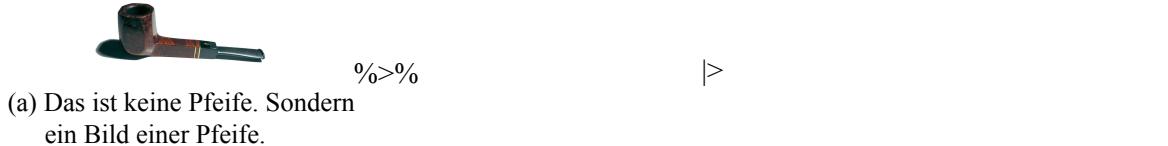


Abbildung 5.10.: So sieht die Pfeife in R aus (Jaja, das ist keine Pfeife, sondern ein Symbol einer Pfeife...). Links: Ein Bild einer Pfeife. Mitte und Rechts: Die zwei R-Symbole für eine “Pfeife” (pipe).

### 5.4.1. Russische Puppen

Computerbefehle, und im Speziellen R-Befehle kann man “aufeinander” – oder vielmehr: ineinander – stapeln, so ähnlich wie eine russische Puppe (vgl. Kapitel 4.7.3). Schauen wir uns das in einem Beispiel an. Dazu definieren wir zuerst einen Vektor  $x$  aus drei Zahlen:

```
x <- c(1, 2, 3)
```

Und dann kommt unser verschachtelter Befehl:

```
sum(x - mean(x))  
## [1] 0
```

Wie schon erwähnt, arbeitet R so einen “verschachtelten” Befehl *von innen nach außen* ab:

Start:  $\text{sum}(x - \text{mean}(x))$

↓

Schritt 1:  $\text{sum}(x - 2)$

↓

Schritt 2:  $\text{sum}(-1, 0, 1)$

↓

Schritt 3: 0. Fertig. Puh. Kompliziert.

---

**Listing 5.2** Eine wild verschachtelte Sequenz von R-Befehlen

---

```
fasse_zusammen (   
  gruppieren (   
    wähle_spalten (   
      filter_zeilen(meine_daten) )))
```

---

Soweit kann man noch einigermaßen folgen. Aber das Verschachteln kann man noch extremer machen, dann wird's wild. Schauen Sie sich mal folgende (Pseudo-)Syntax an:<sup>13</sup>

□

#### 5.4.2. Die Pfeife zur Rettung

Listing 5.2 ist schon harter Tobak, was für echte Fans. Wäre es nicht einfacher, man könnte Listing 5.2 wie folgt schreiben:

Nimm "meine\_daten" \*und dann\*  
filter gewünschte Zeilen \*und dann\*  
wähle gewünschte Spalten \*und dann\*  
teile in Subgruppen \*und dann\*  
fasse sie zusammen.

**Definition 5.2** (Pfeife). "Und dann" heißt auf Errisch %>% oder |>. Man nennt diesen Befehl "Pfeife" (engl. *pipe*). □

#### i Hinweis

Der Befehl %>% verknüpft Befehle. Der Shortcut für diesen Befehl ist Strg-Shift-M. Die Pfeife %>% "wohnt" im Paket {tidyverse}.<sup>14</sup>

Mittlerweile<sup>15</sup> ist auch im Standard-R eine Pfeife eingebaut, die sieht so aus: |>. Die eingebaute Pfeife funktioniert praktisch gleich zur anderen Pfeife %>%, hat aber den Vorteil, dass Sie nicht {tidyverse} starten müssen. Da wir {tidyverse} aber sowieso praktisch immer starten werden, bringt es uns keinen Vorteil, die neuere Pfeife des Standard-R |> zu verwenden.<sup>16</sup>

<sup>12</sup>Vgl. [https://en.wikipedia.org/wiki/The\\_Treachery\\_of\\_Images](https://en.wikipedia.org/wiki/The_Treachery_of_Images)

<sup>13</sup>Ein beliebter Fehler ist es übrigens, nicht die richtige Zahl an schließenden Klammern hinzuschreiben, z.B. fasse\_zusammen(gruppieren(wähle\_spalten(filter\_zeilen(meine\_daten)))) FALSCHES ZAHL AN KLAMMERN.

<sup>14</sup>Genauer gesagt im Paket {magrittr}, welches aber unter the hood von {tidyverse} geladen wird. Also nichts, um dass Sie sich kümmern müssten.

<sup>15</sup>Seit R 4.1

<sup>16</sup>Aber auch keinen Nachteil. Unter *Tools > Global Options...* können Sie einstellen, dass der Shortcut Strg-Shift-M die eingebaute Pfeife verwendet.

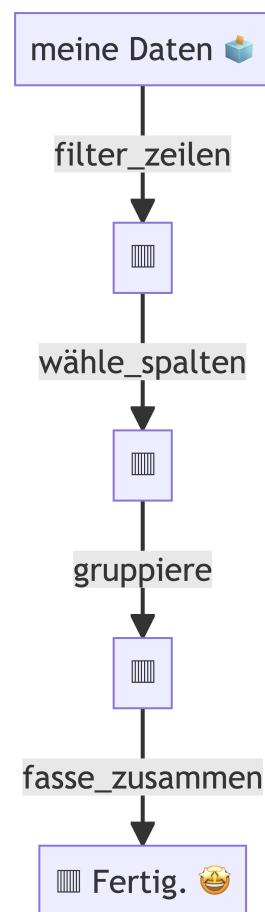


Abbildung 5.11.: Illustration für eine Pfeifensequenz, es geht vorwärts wie am Fließband.

Und jetzt kommt's: So eine Art von Befehls-Verkettung gibt es in R. Schauen Sie sich mal Listing 5.3 an:

---

**Listing 5.3** Eine Pfeifen-Befehlssequenz (Pseudo-Syntax)

```
meine_daten %>%
  filter_gewünschte_zeilen() %>%
  wähle_gewünschte_spalten() %>%
  gruppiere() %>%
  fasse_zusammen()
```

---

So eine Pfeifen-Befehlsequenz ist ein wie ein Fließband, an dem es mehrere Arbeitsstationen gibt, s. Abbildung 5.11. Unser Datensatz wird am Fließband von Station zu Station weitergereicht und an jeder Stelle weiterverarbeitet.

So könnte Ihre "Pfeifen-Sequenz" aussehen:

```
# Hey R, nimm die Tabelle "mariokart":
mariokart %>%
  # filter nur die günstigen Spiele:
  filter(total_pr < 100) %>%
  # wähle die zwei Spalten:
  select(cond, total_pr) %>%
  # gruppiere die Tabelle nach Zustand des Spiels:
  group_by(cond) %>%
  # fasse beide Gruppen nach dem mittleren Preis zusammen:
  summarise(total_pr_mean = mean(total_pr))
```

| cond | total_pr_mean |
|------|---------------|
| new  | 54            |
| used | 43            |

**! Wichtig**

Die Syntax `filter(mariokart, total_pr < 100)` und die Syntax `mariokart |> filter(total_pr < 100)` sind identisch.  
Allgemeiner:  $d \mid> f(x) = f(d, x)$ .

## 5.5. Beispiele für Forschungsfragen

**Übungsaufgabe 5.12.** Bevor Sie die Lösungen der folgenden Fallbeispiele lesen, versuchen Sie die Aufgaben selber zu lösen. Ja, ich weiß, es ist hart, nicht gleich auf die Lösungen zu schauen! □

Sie arbeiten als **Diener** strategischer Assistent der Geschäftsführerin und sind für Faktenchecks und andere Daten-Aufgaben zuständig. Heute sollen Sie zeigen, was Sie können (Schluck).

### 5.5.1. Forschungsfrage 1

#### 5.5.2. Frage

💡 Ich würde von Ihnen gerne wissen, was das teuerste Spiel ist, aber jeweils für neue und gebrauchte Spiele. Aber nur für Spiele, die mit Foto verkauft wurden!

#### 5.5.3. Antwort

```
mariokart %>%
  filter(stock_photo == "yes") %>%
  group_by(cond) %>%
  summarise(total_pr_max = max(total_pr))
```

| cond | total_pr_max |
|------|--------------|
| new  | 75           |
| used | 62           |

Die Funktion `max` liefert den größten Wert eines Vektors zurück:

```
x <- c(1, 2, 10)
max(x)
## [1] 10
```

### 5.5.4. Forschungsfrage 2

#### 5.5.5. Frage

💡 Ich würde gerne die mittlere Versandpauschale wissen, aber getrennt nach Anzahl der Lenkräder, die dem Spiel beigelegt sind. Und ich will nur Gruppen berücksichtigen, die aus mindestens 10 Spielen bestehen!

### 5.5.6. Antwort

Wenn wir die Anzahl der Spiele zählen in Abhängigkeit der beigelegten Lenkräder (wheels), bekommen wir eine Tabelle mit zwei Spalten: wheels und n. n zählt, wie viele Spiele (Zeilen) in der jeweiligen Gruppe (“Teiltabelle”) von wheels sind.

```
mariokart %>%
  count(wheels)
```

| wheels | n  |
|--------|----|
| 0      | 37 |
| 1      | 52 |
| 2      | 51 |
| 3      | 2  |
| 4      | 1  |

Aus dieser Tabellet sehen wir, dass 3 oder 4 Lenkräder nur selten (2 bzw. 1 Mal) beigelegt wurden und wir solche Spiele herausfiltern sollten, bevor wir den Mittelwert der Versankosten ausrechnen:

```
mariokart %>%
  filter(wheels < 3) %>%
  group_by(wheels) %>%
  summarise(mittlere_versandkosten = mean(ship_pr),
            anzahl_spiele = n())
```

| wheels | mittlere_versandkosten | anzahl_spiele |
|--------|------------------------|---------------|
| 0      | 2.7                    | 37            |
| 1      | 3.6                    | 52            |
| 2      | 2.9                    | 51            |

Die Funktion n () gibt die Anzahl der Zeilen pro Teiltabelle zurück.

### 5.5.7. Forschungsfrage 3

#### 5.5.8. Frage

💡 Ich würde gerne den Verkaufspreis in Yen wissen, nicht in Euro. Dann rechne mal den mittleren Verkaufspreis aus und ziehe 10% ab, die wir als Provision unseren Verkäufern zahlen müssen.

### 5.5.9. Antwort

```
mariokart %>%
  select(total_pr) %>%
  mutate(total_pr_yen = total_pr * 133) %>%
  summarise(
    preis_yen_mw = mean(total_pr_yen),
    preis_yen_mw_minus_10proz = preis_yen_mw -
      0.1*preis_yen_mw)
```

| preis_yen_mw | preis_yen_mw_minus_10proz |
|--------------|---------------------------|
| 6634         | 5971                      |

Wie man sieht kann man in `summarise` auch mehr als eine Berechnung einstellen. In diesem Fall haben wir zwei Berechnungen angestellt: Einmal den Mittelwert und einmal den Mittelwert minus 10% (des Mittelwerts).

**Übungsaufgabe 5.13** (Do It Yourself). Denken Sie sich selber ähnliche Forschungsfragen aus. Stellen Sie diese einer vertrauenswürdigen Kommilitonen bzw. einem vertrauenswürdigen Kommilitonen. DIY! Schauen Sie, ob Ihre Aufgabe richtig gelöst wird. □

## 5.6. Praxisbezug

Die Covid19-Epidemie hatte weltweit massive Auswirkungen; auch psychologischer Art wie Vereinsamung, Angst oder Depression. Eine Studie, die die psychologischen Auswirkungen von Mulukom u. a. (2020), die unter der Projekt-ID `tsjnb` bei der Open Science Foundation (OSF), <<https://osf.io/tsjnb/>>, angemeldet ist. Die Daten wurden mit R ausgewertet. Beispielahaft ist unter <https://osf.io/4b9p2> die R-Syntax zu sehen, die die Autoren zur Datenaufbereitung verwendet haben. Einen guten Teil dieser Syntax kennen Sie aus diesem Kapitel. Diese Studie ist, neben einigen vergleichbaren, ein schönes Beispiel, wie Forschung und Praxis ineinander greifen können: Angewandte Forschung als Beitrag zur Lösung eines akuten Problems, der Corona-Pandemie.

## 5.7. Wie man mit Statistik lügt

Ein (leider) immer mal wieder zu beobachtender “Trick”, um Daten zu frisieren ist, nur die Daten zu berichten, die einem in den Kram passen.

**Beispiel 5.11.** Eine Analystin 😊 möchte zeigen, dass der Verkaufspreis von MarioKart-Spielen “viel zu niedrig” ist. Es muss ein höherer Wert rauskommen, findet die Analystin. Der mittlere Verkaufspreis (im Datensatz `mariokart`) liegt bei 50 Euro.

💡 Kann man den Wert nicht ... “kreativ verbessern”? Ein paar Statistik-Tricks anwenden?

Um dieses Ziel zu erreichen, teilt die Analystin den Datensatz in Gruppen nach Anzahl der dem Spiel beigelegten Lenkräder (wheels). Dann wird der Mittelwert pro Gruppe berechnet.

```
mariokart_wheels <-  
mariokart %>%  
  group_by(wheels) %>%  
  summarise(pr_mean = mean(total_pr),  
            count_n = n()) # n() gibt die Anzahl der Zeilen  
            #   ↳ pro Gruppe an  
  
mariokart_wheels
```

| wheels | pr_mean | count_n |
|--------|---------|---------|
| 0      | 41      | 37      |
| 1      | 44      | 52      |
| 2      | 61      | 51      |
| 3      | 70      | 2       |
| 4      | 65      | 1       |

Schließlich berechnet unsere Analystin den *ungewichteten* Mittelwert über diese 5 Gruppen:

```
mariokart_wheels %>%  
  summarise(mean(pr_mean))
```

$$\overline{\text{mean(pr\_mean)}} \\ 56$$

Und das Ergebnis lautet: 56 Euro! Das ist doch schon etwas “besser” als 50 Euro.

Natürlich ist es *falsch* und irreführend, hier einen ungewichteten Mittelwert zu berechnen. Der gewichtete Mittelwert würde wiederum zum korrekten Ergebnis, 50 Euro, führen. □

## 5.8. Fallstudien

### 5.8.1. Die Pinguine

**Übungsaufgabe 5.14.** Machen Sie sich zunächst mit dem Pinguin-Datensatz vertraut. Fokussieren Sie sich auf die Zielvariable *Gewicht*. □

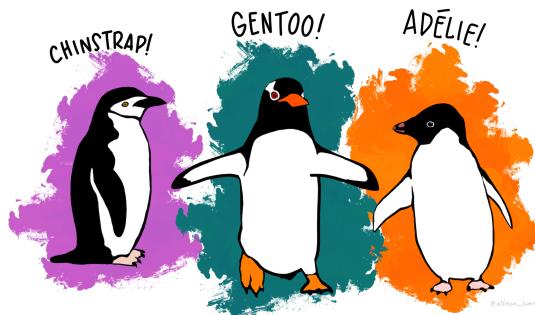


Abbildung 5.12.: Possierlich: Die Pinguine

Bearbeiten Sie die [Fallstudie zu Pinguinen](#) von Allison Horst.<sup>17</sup> Sie können die Teile auslassen, die Themen beinhalten, die *nicht* in diesem Kapitel vorgestellt wurden.

### 5.8.2. Weitere Fallstudien

Diese Fallstudie hat die Analyse von Flugverspätungen zum Thema.



The COVIDiSTRESS global survey is an international collaborative undertaking for data gathering on human experiences, behavior and attitudes during the COVID-19 pandemic. In particular, the survey focuses on psychological stress, compliance with behavioral guidelines to slow the spread of Coronavirus, and trust in governmental institutions and their preventive measures, but multiple further items and scales are included for descriptive statistics, further analysis and comparative mapping between participating countries. Round one data collection was concluded May 30. 2020. To gather comparable data swiftly from across the globe, when the Coronavirus started making a critical impact on societies and individuals, the collaboration and survey was constructed as an urgent collaborative process. Individual contributors and groups in the COVIDiSTRESS network (see below) conducted translations to each language and shared online links by their own best means in each country.

Die Daten stehen unter <https://osf.io/z39us> zur freien Verfügung. Sie können diese echten Daten eigenständig analysieren. Diese Datei beinhaltet die finalen, aufbereiteten Daten. Achtung: Die Datei ist recht groß, ca. 90 MB.

<sup>17</sup><https://allisonhorst.shinyapps.io/dplyr-learnr/#section-welcome>

## 5.9. Aufgaben

### 💡 ChatGPT

Nutzen Sie einen Chat-Bot wie ChatGPT, um sich Hilfe für die R-Syntax geben zu lassen. □

Die Webseite [datenwerk.netlify.app](https://datenwerk.netlify.app) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

1. [wrangle3](#)
2. [wrangle4](#)
3. [wrangle5](#)
4. [wrangle7](#)
5. [wrangle9](#)
6. [wrangle10](#)
7. [tidydata1](#)
8. [affairs-dplyr](#)
9. [dplyr-uebersetzen](#)
10. [haeufigkeit01](#)
11. [mariokart-mean1](#)
12. [mariokart-mean2](#)
13. [mariokart-mean3](#)
14. [mariokart-mean4](#)
15. [mariokart-max1](#)
16. [mariokart-max2](#)
17. [filter01](#)
18. [affairs-dplyr](#)
19. [summarise01](#)
20. [summarise02](#)
21. [mutate01](#)
22. [wrangle3](#)

## 5.10. Vertiefung

### 5.10.1. Tidydatatutor

Die Verben des Datenjudos werden beim “[Tidydatatutor](#)” anschaulich illustriert.<sup>18</sup>

### 5.10.2. Fortgeschrittenes R

#### Hinweis

In weiterführendem Material werden Sie immer wieder auf Inhalte treffen, die Sie noch nicht kennen, die etwa noch nicht im Unterricht behandelt wurden. Seien Sie unbesorgt: In der Regel können Sie diese Inhalte einfach auslassen, ohne den Anschluss zu verlieren. Einfach ignorieren.



Häufig ist es nützlich, die Werte einer Variablen umzukodieren, z.B. “weiblich” in “w” oder in 0. Eine gute Möglichkeit, dies in R umzusetzen, bietet der Befehl `case_when()`; der Befehl wohnt im Tidyverse. [Hier](#) - und an vielen weiteren Stellen im Internet - finden Sie ein Tutorium.<sup>19</sup>. Im Datenwerk finden Sie dazu Übungen, etwa [mutate03](#)

### 5.10.3. Hilfe?! Erbie!

R will nicht, so wie Sie wollen? Sie haben das Gefühl, R verweigert störrisch den Dienst, vermutlich rein aus Boshaftigkeit, rein um Sie zu ärgern? Ausführliches Googeln und ChatGPT befragen hat keine Lösung gebracht? Kurz, Sie brauchen die Hilfe eines kundigen Menschen?<sup>20</sup>

[Hier](#) finden Sie eine Anleitung, wie man seinen Hilfeschrei so formuliert (ruft), dass er nicht nur gehört, sondern auch verstanden wird und einen anderen Menschen veranlasst und ermöglicht Ihnen zu helfen.<sup>21</sup>

Also: Sie müssen Ihr Problem nachvollziehbar aber prägnant formulieren. Das nennt man auch ein *ERBie*, ein *einfaches, reproduzierbare Beispiel* Ihres Problems mit (R-)Syntax:

- einfach: die einfachste Syntax, die Ihr Problem bzw. die Fehlermeldung produziert. Es bietet sich an, einen einfachen, allgemein bekannten Datensatz zu verwenden, etwa `mtcars`
- reproduzierbar: Code (z.B. als Textdatei oder in einem Post), der die Fehlermeldung entstehen lässt

<sup>18</sup><(https://tidydatatutor.com>

<sup>19</sup>[https://www.statology.org/dplyr-case\\_when/](https://www.statology.org/dplyr-case_when/)

<sup>20</sup>[https://www.youtube.com/watch?v=2Q\\_ZzBGPdqE](https://www.youtube.com/watch?v=2Q_ZzBGPdqE)

<sup>21</sup><https://data-se.netlify.app/2022/01/31/erbie-einfache-reproduzierbare-beispiele-ihres-problems-mit-r-syntax/>

**Beispiel 5.12** (Beispiel für ein Erbie). *Problem:* Ich verstehe nicht, warum eine Fehlermeldung kommt

*Ziel:* Ich möchte die Automatikautos filtern (am = 0)

*Was ich schon versucht habe:* Ich habe folgende Posts gelesen ..., aber ohne Erfolg

*Erbie:*

```
data(mtcars)
library(dplyr)  # nicht "tidyverse", denn "dplyr" reicht

mtcars %>%
  filter(am = 0)  # den kürzesten Code, der Ihren Fehler
  ↵ entstehen lässt!

sessionInfo()  # gibt Infos zur R-Version etc. aus
```

Mit dem Paket `{reprex}` kann man sich R-Syntax schön formuliert ausgeben lassen. Das ist perfekt, um den Code dann in einem Forum (oder Mail) einzustellen. Dafür müssen Sie nur den Code auswählen, Strg-C drücken und dann `reprex::reprex` ausführen. Mit Strg-V können Sie die schön formatierte Syntax (sowie die Ausgabe, auch schön formatiert) dann irgendwohin pasten.

#### 💡 Tipp

Posten Sie Ihr Erbie bei <https://gist.github.com/> als “public gist”. Hier ist ein Beispiel.□

#### 5.10.4. Zertifikate und Online-Kurse

Sie können zu den Inhalten dieses Kapitels Zertifikate erwerben (teilweise kostenlos), indem Sie einen Online-Kurs absolvieren, bei z.B. folgenden Anbietern<sup>22</sup>:

- [LinkedIn: R Courses](#)
- [Google/Coursera: Data Analysis with R Programming](#)
- [Duke University/Coursera: Data Analysis with R Specialization](#)

### 5.11. Exkurs

Dall-E 2 ist eine KI, die “realistische Bilder und Kunst aus einer Beschreibung in natürlicher Sprache” erstellt.<sup>23</sup>

🤖 I'd like a mixture between robot und professor, in oil painting

<sup>22</sup>Das ist keine Werbung für spezifische Anbieter und kein umfassender Überblick und keine Kaufempfehlung.

<sup>23</sup><Dall-E 2>

💡 ... s. Abbildung 5.13



Abbildung 5.13.: Bild erzeugt von künstlicher Intelligenz, Quelle: DALL-E 2, 2023-02-09

### **i Hinweis**

Der Nutzen künstlicher Intelligenz für die Datenanalyse ist natürlich breiter: Wenn Sie sich z.B. über die Syntax eines bestimmten Befehls (oder allgemeiner: Vorhabens) nicht sicher sind, fragen Sie sich doch mal einen Bot wie ChatGPT.

## **5.12. Literaturhinweise**

Sauer (2019), Kap. 7, gibt eine Einführung in die Datenaufbereitung (mit Hilfe von R), ähnlich zu den Inhalten dieses Kapitels. Mehr in die Tiefe des “Datenjudo” führen Wickham und Grolemund (2018); der Autor Hadley Wickham ist die Leitfigur in der R-Community schlechthin. Kap. 5 behandelt (etwas ausführlicher) die Themen dieses Kapitels. Er ist einer der Hauptautoren von den beliebten R-Paketen `dplyr` und `ggplot2`.

Wer sich tiefer in das Datenjudo mit dem Tidyverse einarbeiten möchte, dem sei z.B. dieser [Kurs](#) empfohlen.<sup>24</sup>

## **5.13. Literatur**

<sup>24</sup><https://www.datacamp.com/courses/introduction-to-the-tidyverse>

**Teil III.**

**Modellieren**

# 6. Daten verbildlichen

## 6.1. Lernsteuerung

### 6.1.1. Standort im Lernpfad

Abb. Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 6.1.2. Lernziele

- Sie können erläutern, wann und wozu das Visualisieren statistischer Inhalte sinnvoll ist.
- Sie kennen typische Arten von Datendiagrammen.
- Sie können typische Datendiagramme mit R visualisieren.
- Sie können zentrale Ergebnisse aus Datendiagrammen herauslesen.

### 6.1.3. Benötigte R-Pakete

```
library(tidyverse)
library(easystats)
library(DataExplorer) # nicht vergessen zu installieren
library(ggpubr) # optional
library(ggstatsplot) # optional
```

### 6.1.4. Benötigte Daten

Zuerst definieren wir den Pfad, wo wir die Daten finden, s. Listing 6.1.

---

#### Listing 6.1 Pfad zu den Mariokart-Daten

---

```
mariokart_path <- paste0(
  "https://vincentarelbundock.github.io/Rdatasets",
  "/csv/openintro/mariokart.csv")
```

---

Dann importieren wir die Mariokart-Daten:

```
mariokart <- read.csv(mariokart_path)
```

### 6.1.5. R-Code zum Copy-Pasten

Sie finden den R-Code für jedes Kapitel [hier](#). □

### 6.1.6. Quiz zum Einstieg

Vielleicht fordert Sie die Lehrkraft zu einem Einstiegsquiz auf, etwas mittels der Plattform [antwor-te.jetzt](#).<sup>1</sup> Alternativ überlegen Sie sich selber 10 Quiz-Aufgaben zum Stoff des letzten Kapitels.

### 6.1.7. Wozu das alles?

👽 Wir müssen die Galaxis retten, Kermit.

🐸 Schlock

## 6.2. Ein Dino sagt mehr als 1000 Worte

Es heißt, ein Bild sage mehr als 1000 Worte. Schon richtig, aber ein Dinosaurier sagt auch mehr als 1000 Worte, s. Abbildung 6.1.<sup>2</sup>

In Abbildung 6.1 sieht man verschiedene “Bilder”, also Datensätze: etwa einen Dino und einmal einen Kreis. Obwohl die Bilder grundverschiedene sind, sind die zentralen statistischen Kennwerte (praktisch) identisch.

In die gleiche Bresche schlägt “Anscombes Quartett” (**Anscombe1973?**), s. Abbildung 6.2: Es zeigt vier Datensätze, in denen die zentralen Statistiken fast identisch sind, also Mittelwerte, Streuungen, Korrelationen. Aber die Streudiagramme sind grundverschieden.<sup>3</sup>

Anscombes Beispiel zeigt (zugespitzt): Eine Visualisierung enthüllt, was der Statistik (als Kennzahl) verhüllt bleibt.

#### ! Wichtig

Statistische Diagramme können Einblicke geben, die sich nicht (leicht) in grundlegenden Statistiken (Kennwerten) abbilden. □

<sup>1</sup><https://antwor-te.jetzt/>

<sup>2</sup>Quelle: <https://towardsdatascience.com/how-to-turn-a-dinosaur-dataset-into-a-circle-dataset-with-the-same-statistics-64136c2e2ca0>

<sup>3</sup>Quelle: [https://en.wikipedia.org/wiki/Anscombe%27s\\_quartet](https://en.wikipedia.org/wiki/Anscombe%27s_quartet)

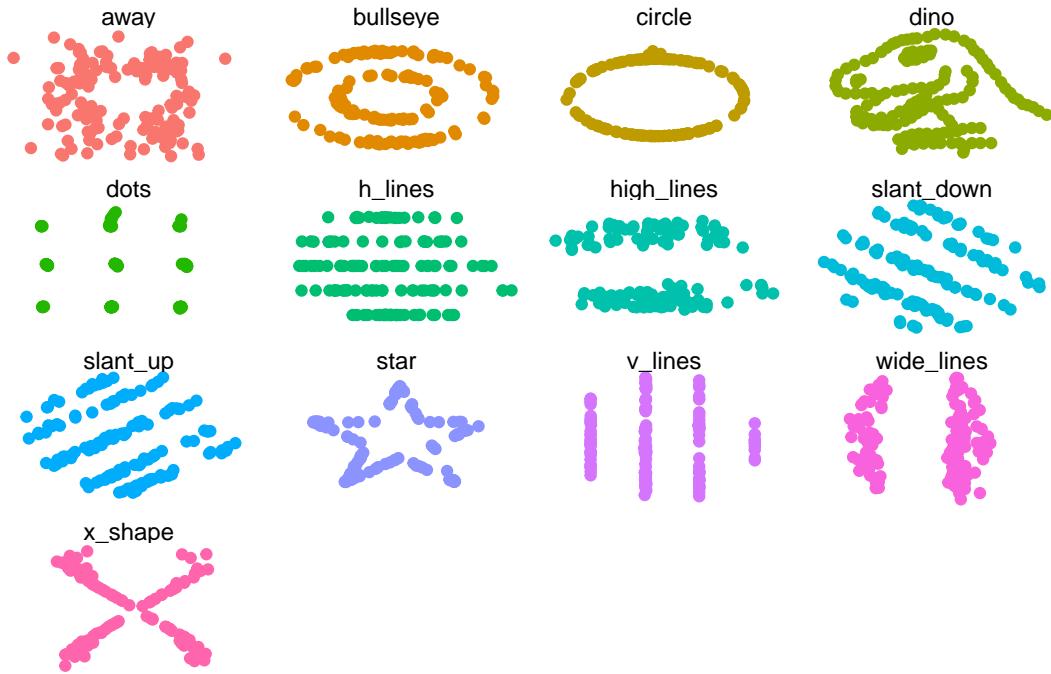


Abbildung 6.1.: Alle Diagramme haben gleiche statistische Koeffizienten, wie Mittelwert und Streuung und Korrelation, aber die Datengrundlage sind komplett verschieden.

Unter visueller Cortex ist sehr leistungsfähig. Wir können ohne Mühe eine große Anzahl an Informationen aufnehmen und parallel verarbeiten. Aus diesem Grund sind Datendiagramme eine effektive und einfache Art, aus Daten Erkenntnisse zu ziehen.

#### Tipp

Nutzen Sie Datendiagramme umfassend; sie sind einfach zu verstehen und doch sehr mächtig.

### 6.2.1. Datendiagramm

Ein *Datendiagramm* (kurz: *Diagramm*) ist ein Diagramm, das Daten und Statistiken zeigt, mit dem Zweck, Erkenntnisse daraus zu ziehen.

**Beispiel 6.1** (Aus der Forschung: Ein aufwändiges (und ansprechendes) Datendiagramm). Hier finden Sie ein Beispiel für ein Datendiagramm, das mit R erzeugt wurde ([scherer\\_seasonal\\_2019?](#)).

### 6.2.2. Ein Bild hat nicht so viele Dimensionen

Abbildung 6.3 zeigt ein Bild mit mehreren (5) Variablen, die jeweils einer “Dimension” entsprechen. Wie man (nicht) sieht, wird es langsam unübersichtlich. Offenbar kann man in einem Bild nicht

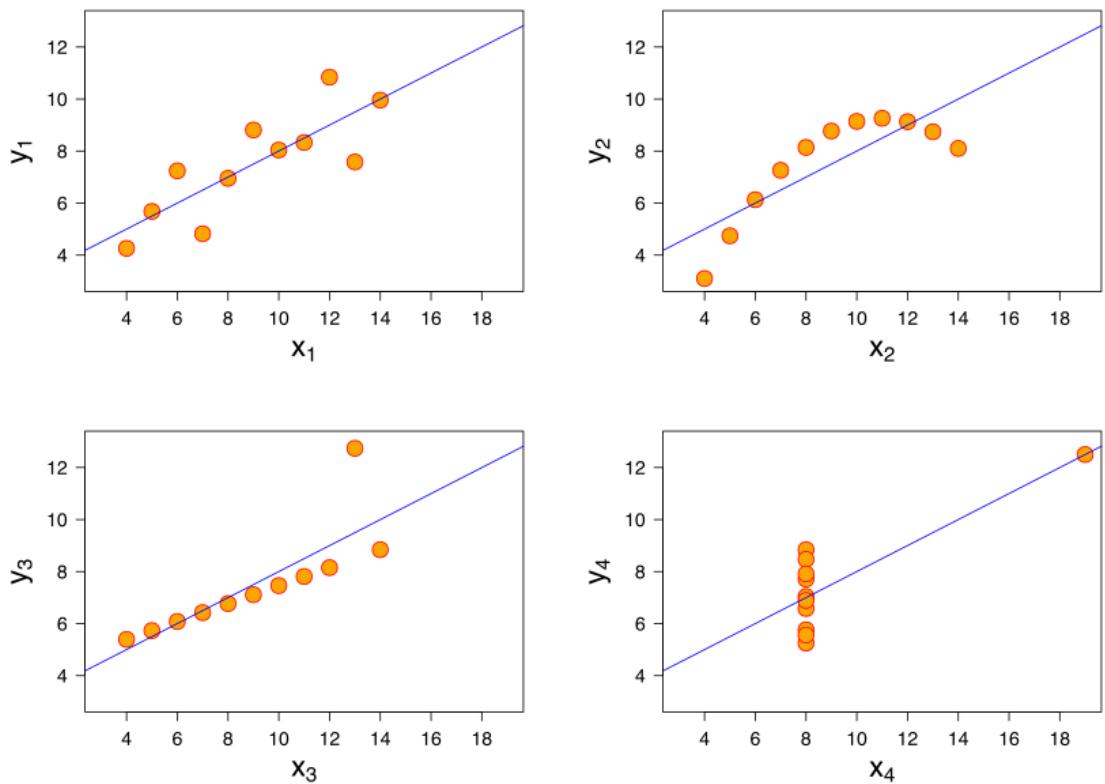


Abbildung 6.2.: Anscombes Quartet: Gleiche statistischen Kennwerte in vier Datensätzen

beliebig viele Variablen sinnvoll reinquetschen. Die “Dimensionalität” eines Diagramms hat ihre Grenzen, vielleicht bei 4-6 Variablen.

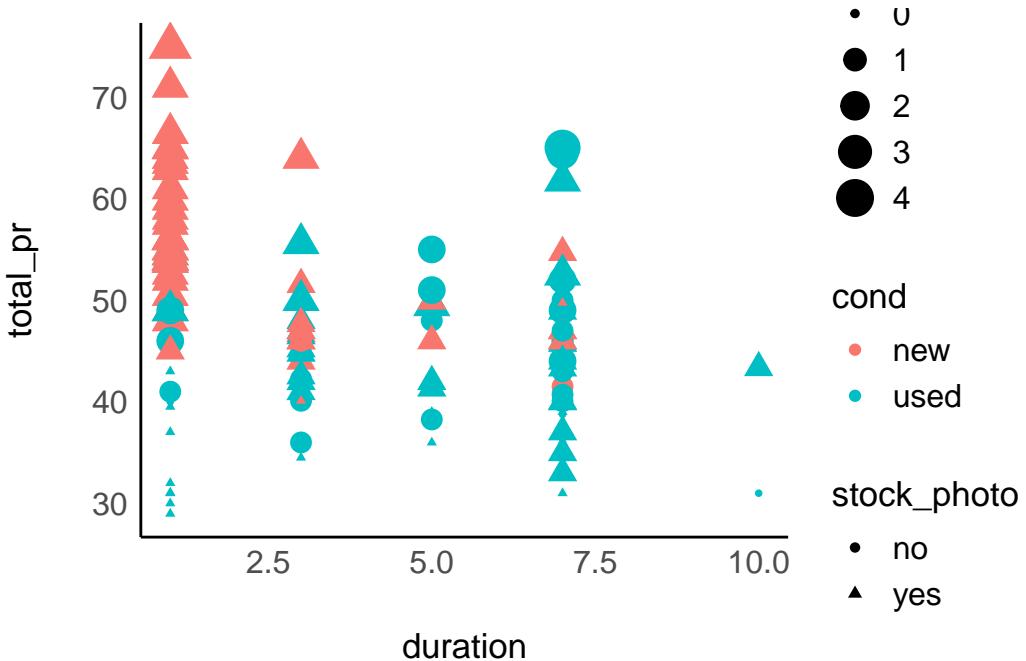


Abbildung 6.3.: Ein Diagramm kann nur eine begrenzte Anzahl von Variablen zeigen. Wenn Sie dieses Bild nicht checken: Prima. Genau das soll das Bild zeigen.

Möchten wir den Zusammenhang von vielen Variablen, z.B. mehr als 5, verstehen, kommen wir mit Bildern nicht weiter. Dann brauchen wir andere Werkzeuge: statistics to the rescue.

#### **i Hinweis**

Bei klaren Zusammenhängen und wenig Variablen braucht man keine (aufwändige) Statistik. Ein Bild (Datendiagramm) ist dann (oft) ausreichend. Man könnte sagen, dass es Statistik nur deshalb gibt, weil unser Auge mit mehr als ca. 4-6 Variablen nicht gleichzeitig umgehen kann.

**Übungsaufgabe 6.1.** Wie viele Variablen sind in Abbildung 6.3 dargestellt?<sup>4</sup>

### 6.3. Nomenklatur von Datendiagrammen

Tabelle 6.1 zeigt eine – sehr kurze Nomenklatur – an Datendiagrammen.<sup>5</sup>

<sup>4</sup>5

<sup>5</sup>Weitere Nomenklaturen sind möglich, aber wir halten hier die Sache einfach. Wer an Vertiefung interessiert ist, findet bei data-to-vis einen Überblick über verschiedene Typen an Diagrammen, sogar in Form einer systematischen Nomenklatur: <https://www.data-to-viz.com/>.

Tabelle 6.1.: Ein (sehr kurze) Nomenklatur von Datendiagrammen

| Erkenntnisziel | qualitativ               | quantitativ                   |
|----------------|--------------------------|-------------------------------|
| Verteilung     | Balkendiagramm           | Histogramm und Dichtediagramm |
| Zusammenhang   | gefülltes Balkendiagramm | Streudiagramm                 |
| Unterschied    | gefülltes Balkendiagramm | Boxplot                       |

**i Hinweis**

Wir arbeiten hier mit dem Datensatz `mariokart`. Hilfe bzw. ein Data-Dictionary (Codebook) finden Sie [hier](#).

## 6.4. Verteilungen verbildlichen

### 6.4.1. Verteilung: nominale Variable

**Definition 6.1** (Verteilung). Eine (Häufigkeits-)Verteilung einer Variablen  $X$  schlüsselt auf, wie häufig jede Ausprägung von  $X$  ist.□

**Beispiel 6.2.** Tabelle 6.2 zeigt die Häufigkeitsverteilung von `cond` (condition, also der Zustand des Artikels, neu oder gebraucht) aus dem Datensatz `mariokart`. Die Variable hat 5 Ausprägungen; z.B. kommt die Ausprägung `new` 59 mal vor.□

Tabelle 6.2.: Häufigkeitsverteilung von `cond` aus dem Datensatz `mariokart`

| cond | n  |
|------|----|
| new  | 59 |
| used | 84 |

Zugegeben, das Datendiagramm von `cond` ist nicht so aufregend, s. Abbildung 6.4. Wie man sieht, besteht so ein Diagramm als *Balken*, daher heißt es *Balkendiagramm*<sup>6</sup>. Man kann so ein Diagramm um 90° drehen; keine Ausrichtung ist unbedingt besser als die andere.

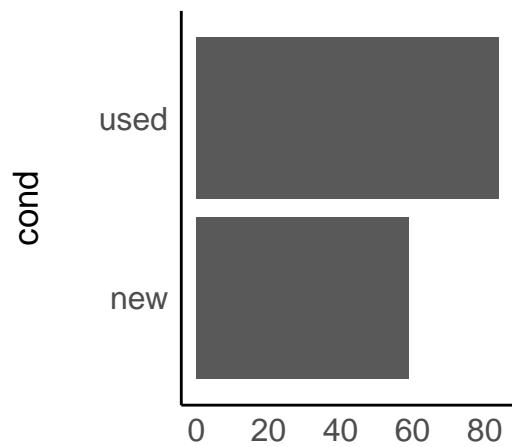
**Definition 6.2** (Balkendiagramm). Ein Balkendiagramm eignet sich, um Häufigkeiten darzustellen

Es gibt viele Methoden, sich mit R ein Balkendiagramm ausgeben zu lassen. Eine einfache, komfortable ist die mit dem Paket `DataExplorer`, s. Abbildung 6.5.

Zuerst importieren wir die Daten, s. Listing 6.2 und Listing 6.1.

<sup>6</sup>synonym: Säulendiagramm

horizontale Balken



vertikale Balken

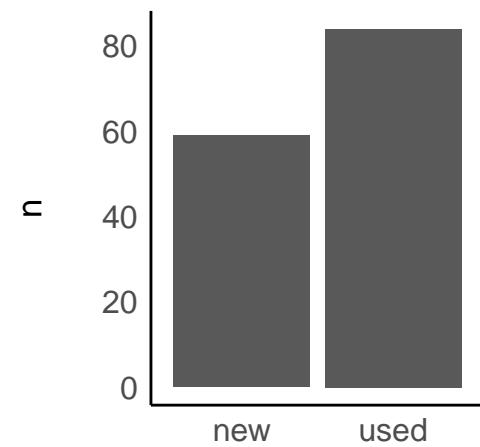


Abbildung 6.4.: Häufigkeitsverteilung der Variable cond

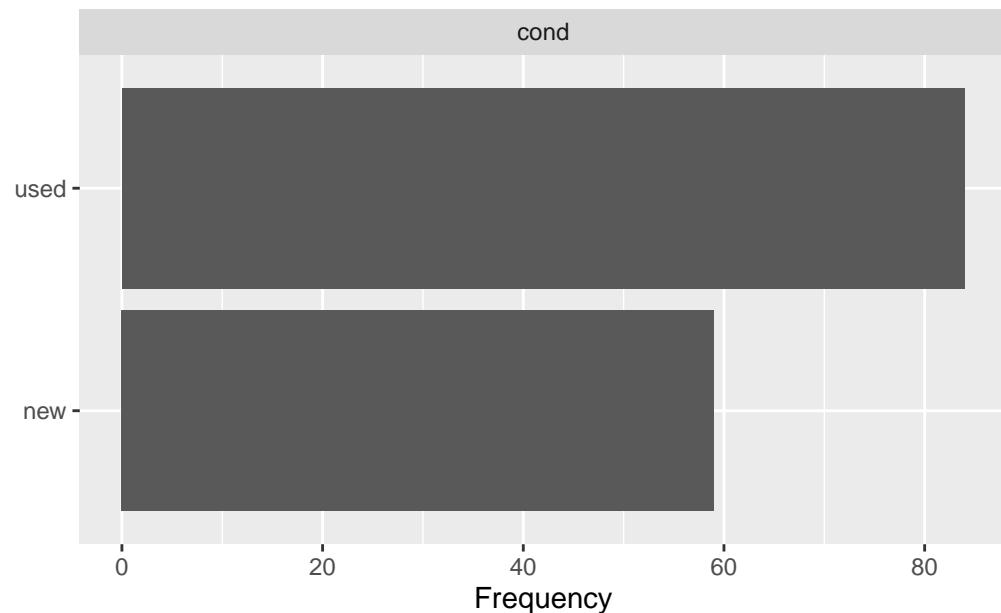


Abbildung 6.5.: Balkendiagramm mit dem R-Paket DataExplorer

---

**Listing 6.2** Mariokart-Daten importieren von einer Webseite

---

```
mariokart <- read.csv(mariokart_path)
```

---

Außerdem nicht vergessen, das Paket `DataExplorer` zu starten, s. Listing 6.3.<sup>7</sup> In diesem Paket “wohnen” die Befehle, die wir zum Erstellen der Datendiagramme nutzen werden. Listing 6.4 zeigt die Syntax, um ein Balkendiagramm zu erstellen.<sup>8</sup>

---

**Listing 6.3** Wir starten das R-Paket DataExplorer

---

```
library(DataExplorer)
```

---

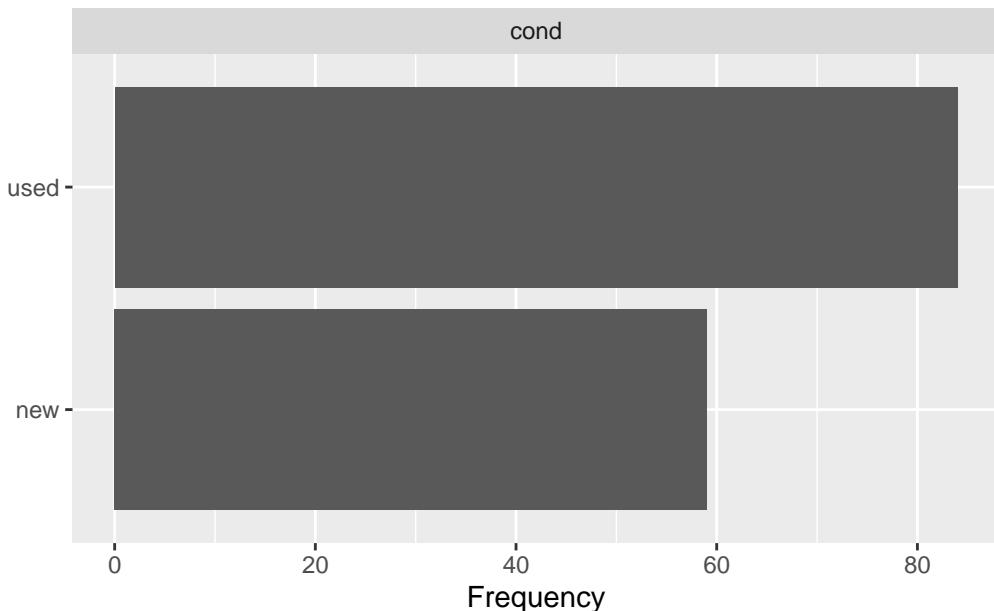
---

**Listing 6.4** Syntax zur Erstellung eines Balkendiagramms

---

```
mariokart %>%
  select(cond) %>%
  plot_bar()
```

---



Die Syntax ist in Listing 6.4 abgedruckt<sup>9</sup>. Übersetzen wir die Syntax ins Deutsche:

Nimm den Datensatz `mariokart` \*und dann\*

<sup>7</sup>Natürlich müssen Sie das Paket einmalig installiert haben, bevor Sie es starten können.

<sup>8</sup>Auf der Hilfeseite der Funktion finden Sie weitere Details zur Funktion.

<sup>9</sup>Zur Erinnerung: `%>%` nennt man die “Pfeife und lässt sich als”und dann” übersetzen, vgl. Kapitel 5.4.

wähle die Spalte cond \*und dann\* zeichne ein Balkendiagramm.

**Übungsaufgabe 6.2** (Spalten wählen für das Balkendiagramm). Hätten wir andere Spalten ausgewählt, so würde das Balkendiagramm die Verteilung jener Variablen zeigen. Ja, Sie können auch mehrere Variablen auf einmal auswählen. Probieren Sie das doch mal aus!

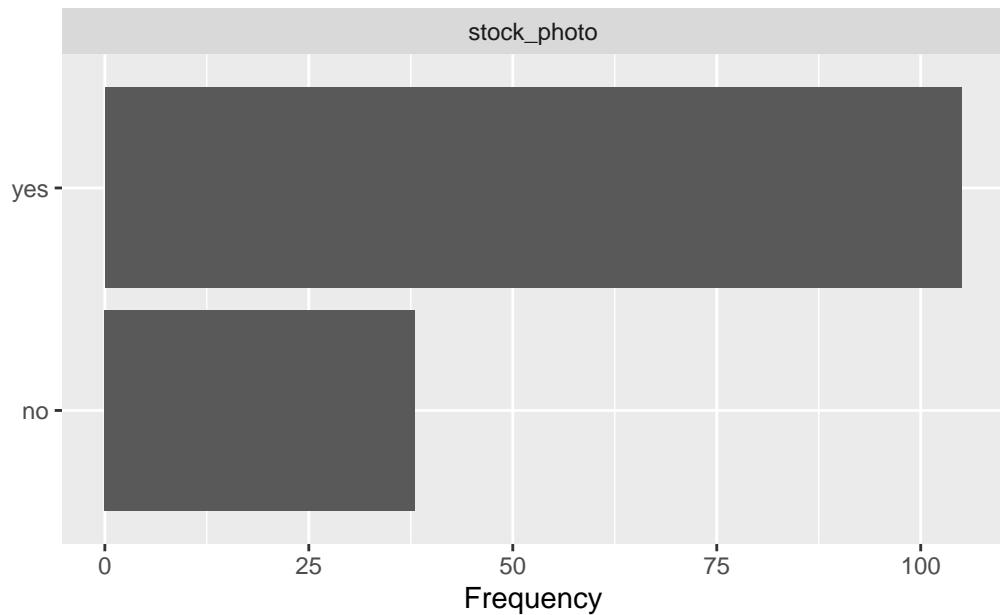
### Übungsaufgabe 6.3.

#### 6.4.2. Aufgabe

Visualisieren Sie die Verteilung von stock\_photo!

#### 6.4.3. Lösung

```
mariokart |>  
  select(stock_photo) |>  
  plot_bar()
```



## 6.4.4. Verteilung: quantitative Variable

### 6.4.4.1. Histogramm

Bei einer quantitativen Variablen mit vielen Ausprägungen wäre ein Balkendiagramm nicht so aussagekräftig, s. Abbildung 6.6 (links). Es gibt einfach zu viele Ausprägungen.

Die Lösung: Wir reduzieren die Anzahl der Ausprägungen, in dem wir auf ganze Dollar runden. Oder, um noch weniger Ausprägungen zu bekommen, können wir einfach Gruppen definieren, z.B.

- Gruppe 1: 0-5 Dollar
- Gruppe 2: 6-10 Dollar
- Gruppe 2: 11-15 Dollar ...

In Abbildung 6.6 (rechts) sind z.B. die Ausprägungen des Verkaufspreis (`total_pr`) in Gruppen der Breite von 5 Dollar aufgeteilt worden. Zusätzlich sind noch die einzelnen Werte als schwarze Punkte gezeigt.

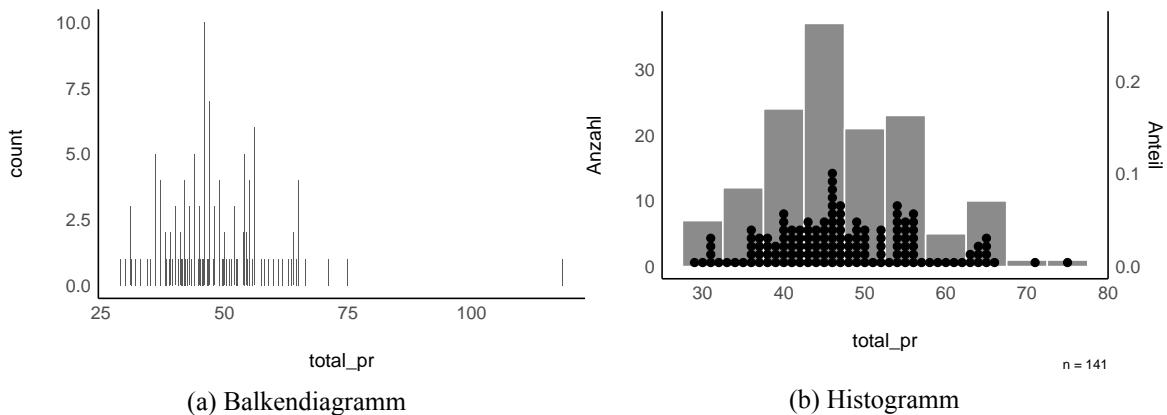


Abbildung 6.6.: Balkendiagramm vs. Histogramm für den Gesamtpreis (`total_pr`)

**Definition 6.3 (Histogramm).** Ein Histogramm ist ein Diagramm zur Darstellung der Häufigkeitsverteilung einer quantitativen Variablen. Die Daten werden in Gruppen (Klassen) eingeteilt, die dann durch einen Balken (pro Klasse) dargestellt sind. Die Höhe der Balken zeigt die Häufigkeit der Daten in dieser Gruppe/in diesem Balken<sup>10</sup>.

Es gibt keine klare Regel, in wie viele Balken ein Histogramm gegliedert sein sollte. Nur: Es sollten nicht sehr viele und nicht sehr wenig sein, s. Abbildung 6.7 links bzw. Abbildung 6.7, rechts.

Zur Erstellung eines Histogramms können Sie die Syntax Listing 6.5 nützen, vgl. Abbildung 6.8, links.

### Übungsaufgabe 6.4.

---

<sup>10</sup>bei konstanter Balkenbreite

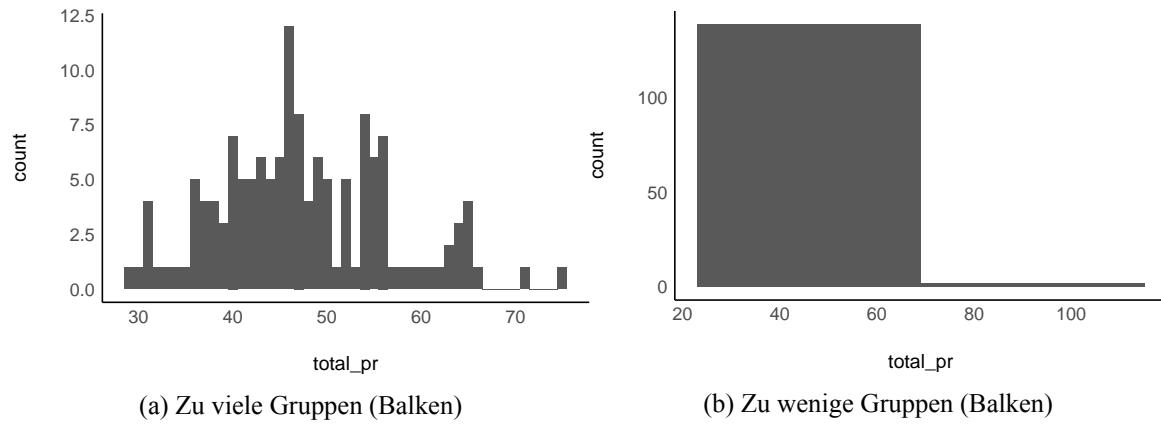


Abbildung 6.7.: Nicht zu wenig und nicht zu viele Balken im Balkendiagramm

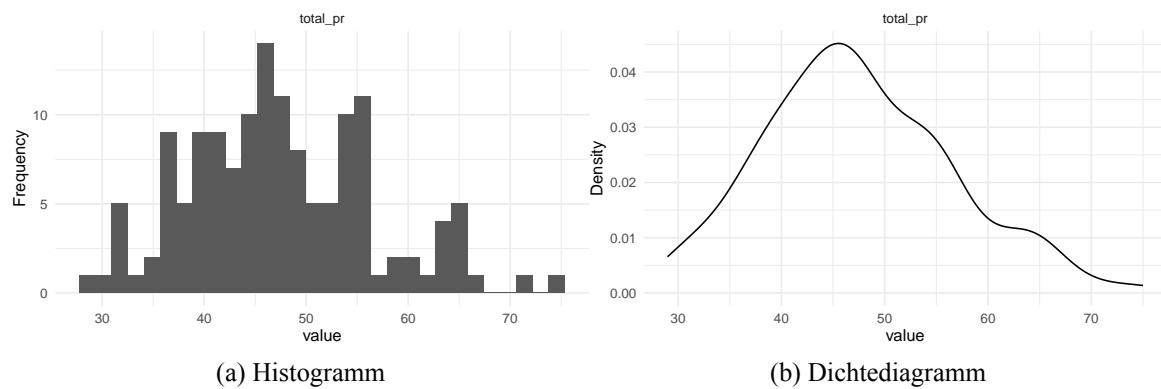


Abbildung 6.8.: Eine stetige Verteilung verbindlichen

---

**Listing 6.5** Syntax zur Erstellung eines Histogramms

---

```
mariokart %>%
  select(total_pr) %>%
  filter(total_pr < 100) %>% # ohne Extremwerte
  plot_histogram()
```

---

**6.4.5. Aufgabe**

Visualisieren Sie die Verteilung von `ship_pr` anhand eines Histogramms!

**6.4.6. Lösung**

```
mariokart |>
  select(ship_pr) |>
  plot_histogram()
```

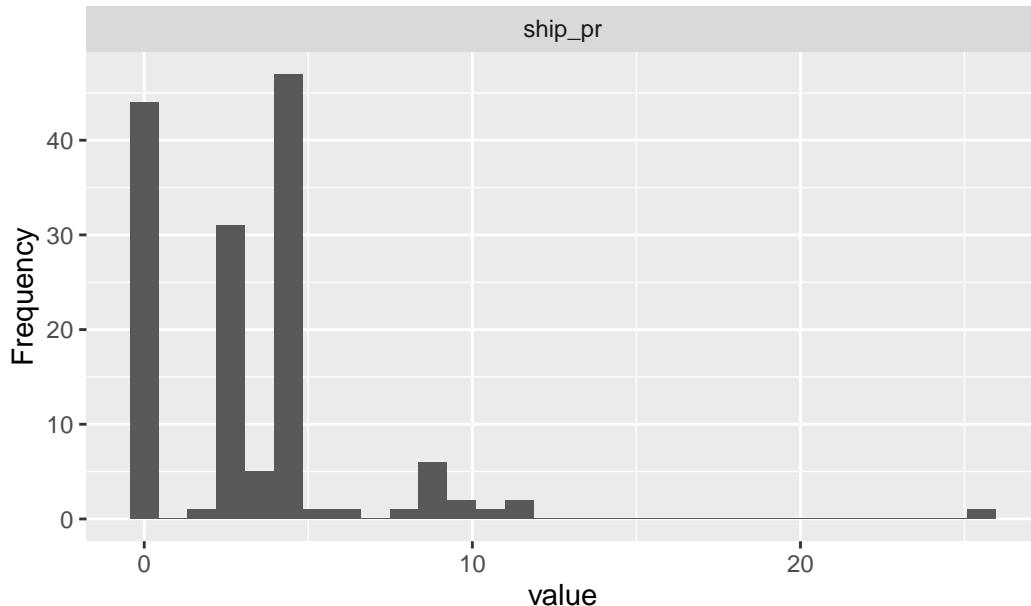
**6.4.6.1. Dichtediagramm**

Abbildung 6.9 fügt zu `?@fig-balken-total-pr-hist` ein *Dichtediagramm* hinzu (rote Linie). Ein Dichtediagramm ähnelt einem “glattgeschmirgeltem” Histogramm.

**Definition 6.4** (Dichtediagramm). Ein Dichtediagramm visualisiert die Verteilung einer stetigen Variablen. Im Gegensatz zum Histogramm wird der Verlauf der Kurve geglättet, so kann Rauschen (Zufallsschwankung) besser ausgeblendet werden.<sup>11</sup>

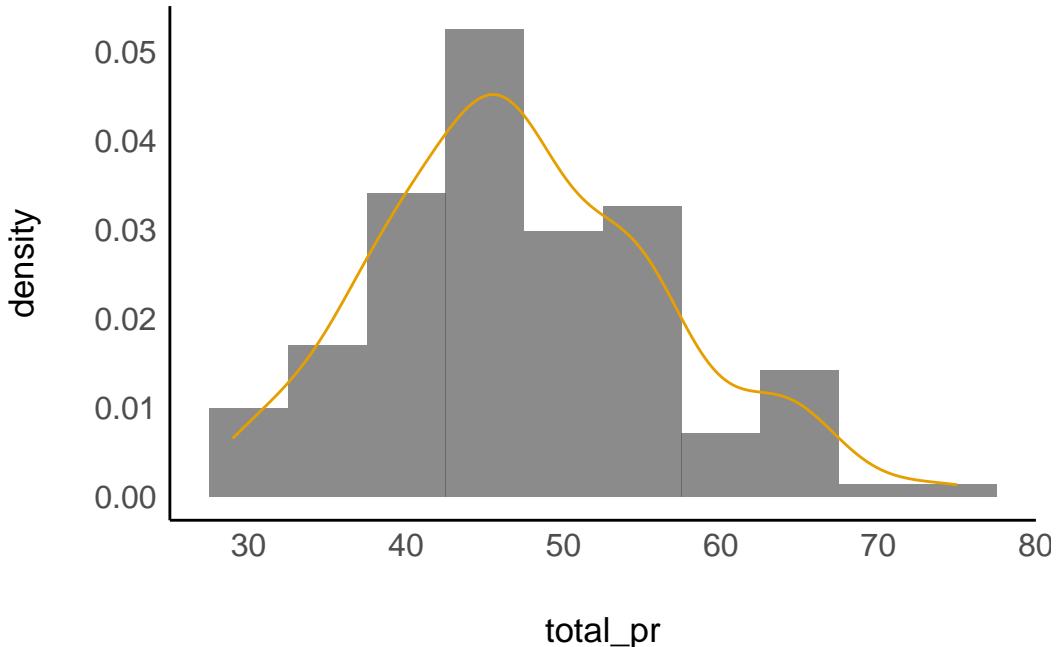


Abbildung 6.9.: Histogramm (graue Balken) und Dichtediagramm (orange Linie) für total\_pr

**Übungsaufgabe 6.5.** Erstellen Sie das Diagramm Abbildung 6.8, rechtes Teildiagramm!<sup>12</sup> □

#### 6.4.6.2. Eigenschaften von Verteilungen

Verteilungen unterscheiden sich z.B. einerseits in ihrem “typischen” oder “mittleren” Wert<sup>13</sup> und andererseits in ihrer Streuung<sup>14</sup>

(Diagramme von) Verteilungen können symmetrisch oder schief (nicht symmetrisch) sein, s. Abbildung 6.10.

Abbildung 6.11 zeigt verschiedene Formen von Verteilungen. “Bimodal” meint “zweigipflig” und “multimodal” entsprechend “mehrgipflig”.<sup>15</sup>

#### Übungsaufgabe 6.6.

<sup>11</sup>Mit *Dichte* ist die Anzahl der Beobachtungen pro Einheit der Variablen auf der X-Achse gemeint.

<sup>12</sup>Grob gesagt: `mariokart %>% plot_density()`.

<sup>13</sup>vgl. Kapitel 7.5

<sup>14</sup>vgl. Kapitel 8.4.

<sup>15</sup>Quelle: ifes/FOM Hochschule, <https://github.com/FOM-ifes/VL-Vorlesungsfolien>

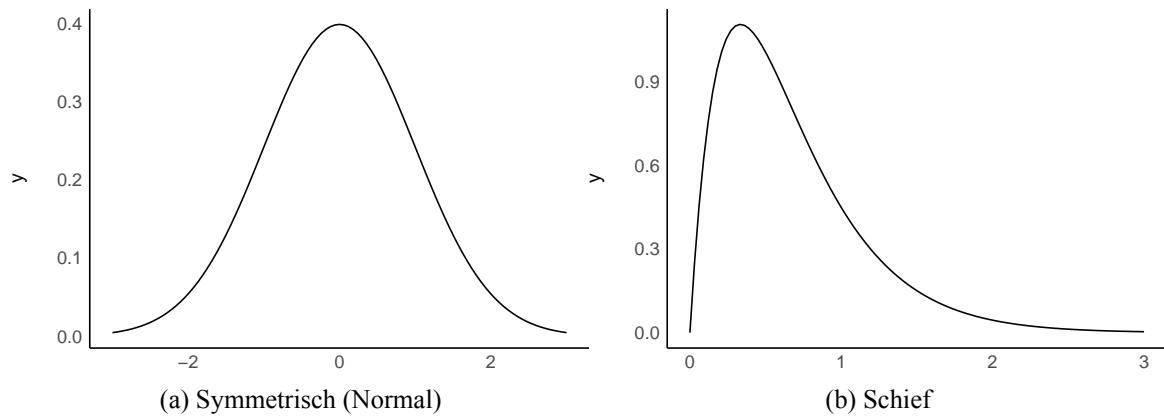


Abbildung 6.10.: Symmetrische vs. schiefe Verteilung, verbildlicht

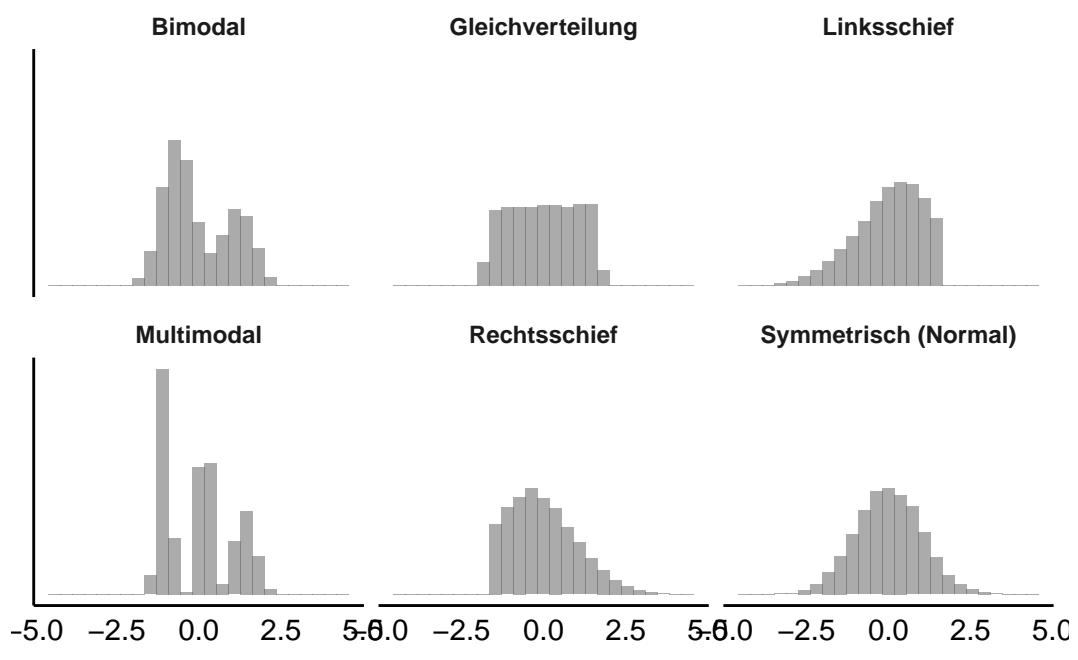


Abbildung 6.11.: Verschiedene Verteilungsformen

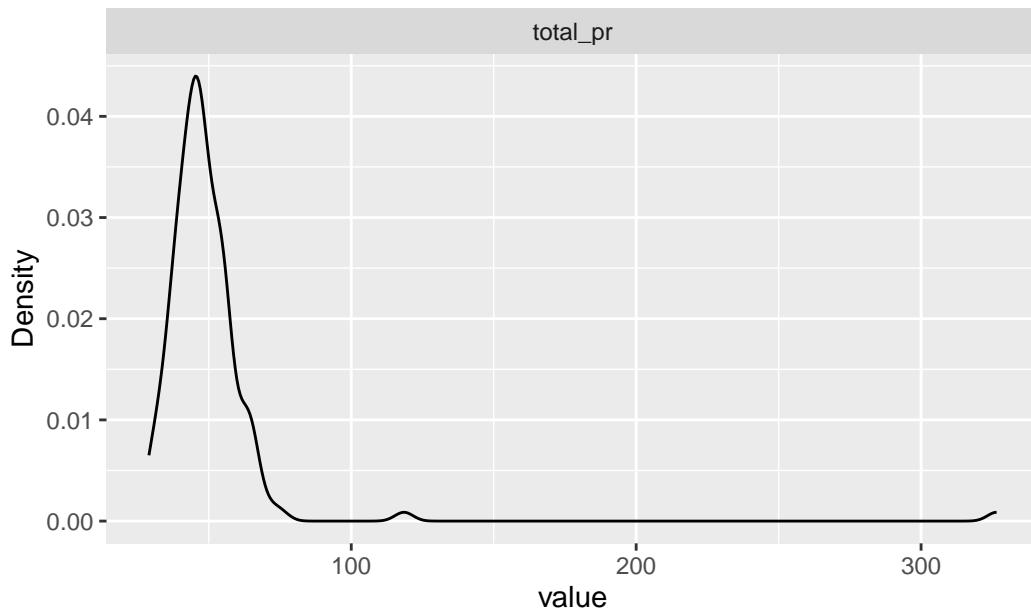
#### 6.4.7. Aufgabe

Bennen Sie die am besten passende Verteilungsform für die Variable `total_pr`.

#### 6.4.8. Lösung

Die Verteilung ist rechtschief.

```
mariokart |>  
  select(total_pr) |>  
  plot_density()
```



#### 6.4.9. Normalverteilung

Eine Normalverteilung ist eine bestimmte Art von Verteilung einer quantitativen Variablen. Aber sie ist besonders wichtig, und ist daher hier herausgestellt.

Eine Normalverteilung sehen Sie in Abbildung 6.10, links. Sie hat u.a. folgende Eigenschaften:

- symmetrisch
- glockenförmig
- stetig
- eingipflig (unimodal)
- Mittelwert, Median und Modus sind identisch

**Beispiel 6.3.** Beispiele für normalverteilte Variablen sind Körpergröße von Männern oder Frauen, IQ-Werte, Prüfungsergebnisse, Messfehler, Lebensdauer von Glühbirnen, Gewichte von Brotlaiben, Milchproduktion von Kühen, Brustumfang schottischer Soldaten (Lyon 2014).  $\square$

Die Normalverteilung ist von hoher Bedeutung, da sich diese Verteilung unter (recht häufigen) Bedingungen zwangsläufig ergeben muss.

**Definition 6.5** (Entstehung einer Normalverteilung). Wenn sich eine Variable  $X$  als Summe mehrerer, unabhängiger, etwa gleich starker Summanden, dann kann man erwarten, dass sich diese Variable  $X$  tendenziell normalverteilt.  $\square$

Dieses Phänomen kann man gut anhand des [Galton-Bretts](#) veranschaulichen.

### ! Parameter der Normalverteilung

Eine Normalverteilung lässt sich exakt beschreiben anhand zweier Parameter: ihres zentralen Werts (Mittelwerts),  $\mu$ , und ihrer Streuung (Standardabweichung),  $\sigma$ .  $\square$

Kennt man diese beiden Parameter, so kann man einfach angeben, welcher Anteil der Fläche sich in einem bestimmten Bereich befindet, s. Abbildung [6.12](#).<sup>16</sup>

Davon leitet sich die “68-95-99-Prozentregel” ab:

- 68 % der Werte im Bereich  $\mu \pm 1 \cdot \sigma$
- 95 % der Werte im Bereich  $\mu \pm 2 \cdot \sigma$
- 99,7 % der Werte im Bereich  $\mu \pm 3 \cdot \sigma$

## 6.5. Zusammenhänge verbildlichen

### 6.5.1. Zusammenhang: nominale Variablen

**Beispiel 6.4** (Beispiele für Zusammenhänge bei nominalen Variablen).

- Hängt Berufserfolg (Führungs kraft ja/nein) mit dem Geschlecht zusammen?
- Hängt der Beruf des Vaters mit dem Schulabschluss des Kindes (Abitur, Realschule, Mittelschule) zusammen?
- Gibt es einen Zusammenhang zwischen Automarke und politische Präferenz einer Partei?  $\square$

Sagen wir, Sie arbeiten immer noch beim Online-Auktionshaus und Sie fragen sich, ob ein Produktfoto wohl primär bei neuwertigen Produkten beiliegt, aber nicht bei gebrauchten? Dazu betrachten Sie wieder die [mariokart](#)-Daten, s. Abbildung [6.13](#).

Tatsächlich: Es findet sich ein Zusammenhang zwischen der Tatsache, ob dem versteigerten Produkt ein Foto bei lag und ob es neuwertig oder gebraucht war (Abbildung [6.13](#), links). Bei neuen Spielen

<sup>16</sup>Quelle: Ainali – Own work, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=3141713>

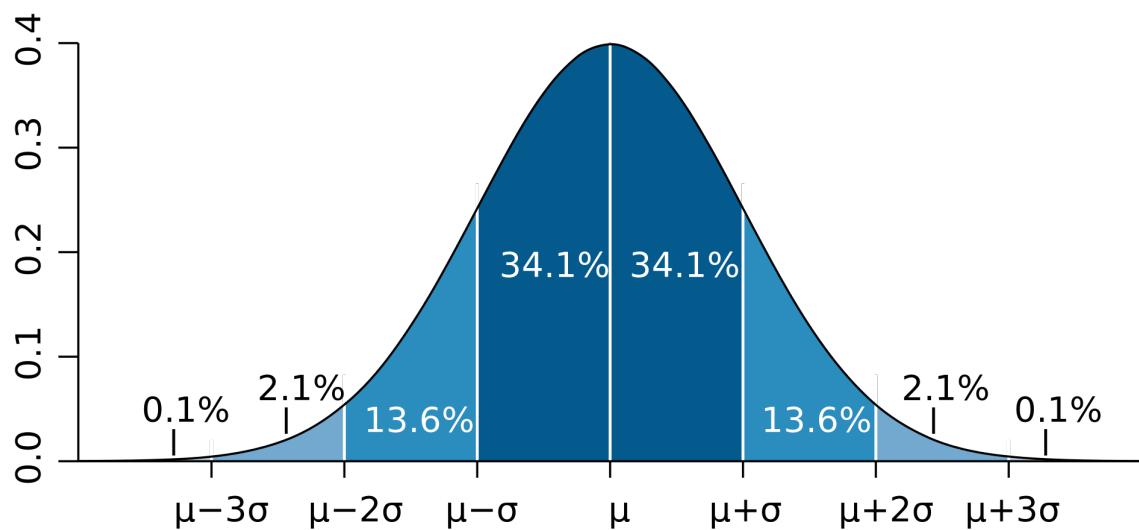
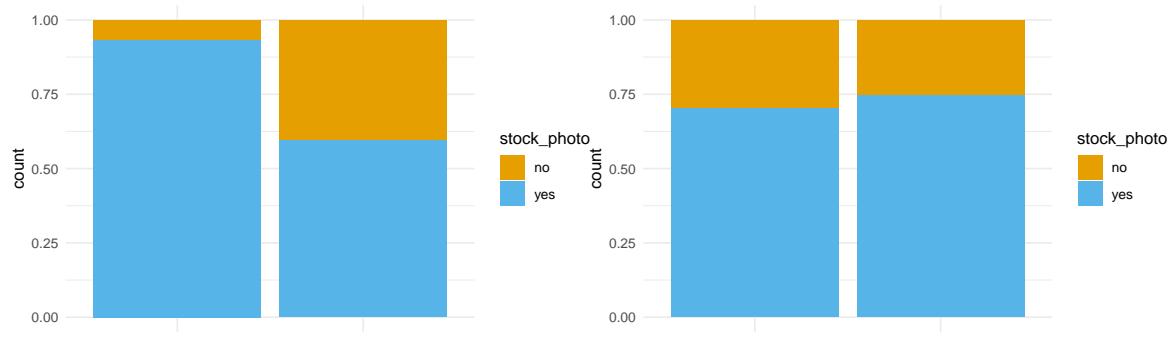


Abbildung 6.12.: Die Flächeninhalte (Wahrscheinlichkeitsmasse) einer Normalverteilung in Abhängigkeit der SD-Einheiten



- (a) Es findet sich ein Zusammenhang von Foto und Zustand in den Daten  
(b) Es findet sich (fast) kein Zusammenhang von wheel und Foto in den Daten

Abbildung 6.13.: Zusammenhang zwischen nominalskalierten Variablen verbildlichen

war fast immer (ca. 90%) ein Foto dabei; bei gebrauchten Spielen immerhin bei gut der Hälfte der Fälle.

Anders sieht es aus für die Frage, ob ein (oder mehrere) Lenkräder dem Spiel beilagen (oder nicht) in Zusammenhang mit der Fotofrage Hier gab es fast keinen Unterschied zwischen neuen und alten Spielen, was die Frage nach “Foto des Produkts dabei” betraf (Abbildung 6.13, rechts), der Anteil betrug jeweils ca. 70%. Das zeigt, dass es keinen Zusammenhang zwischen Foto und Neuwertigkeit des Spiels gibt (laut unseren Daten).

Anders gesagt: Unterscheiden sich die “Füllhöhe” in den Diagrammen, so gibt es einen Unterschied hinsichtlich “Foto ist dabei” zwischen den beiden Gruppen (linker vs. rechter Balken). Unterscheiden sich die Anteile in den Gruppen (neuwertige vs. gebrauchte Spiele), so spielt z.B. die Variable “Foto dabei” offenbar eine Rolle. Dann hängen Neuwertigkeit und “Foto dabei” also zusammen!

So können Sie sich in R ein gefülltes Balkendiagramm ausgeben lassen, s. Abbildung 6.14. Diese Darstellung eignet sich, um Zusammenhänge zwischen zwei zweistufigen nominal skalierten Variablen zu verbildlichen. Die verschiedenen Werte der Füllfarbe werden den Stufen der Variablen cond zugewiesen, s. Listing 6.6.

---

**Listing 6.6** R-Syntax für ein gefülltes Balkendiagramm

---

```
mariokart %>%
  select(cond, stock_photo) %>%
  plot_bar(by = "cond") # aus dem Paket DataExplorer
```

---

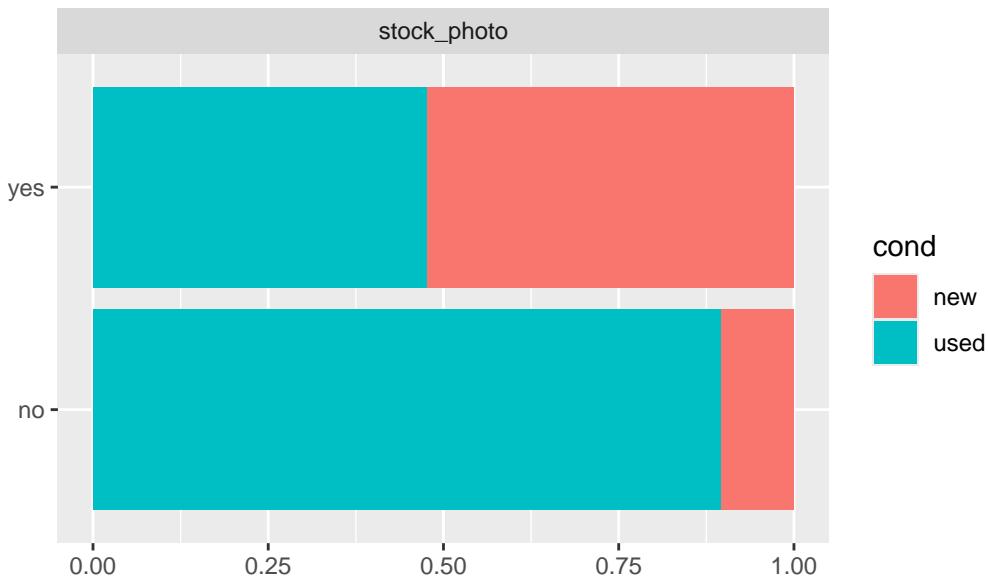


Abbildung 6.14.: Ein gefülltes Balkendiagramm zur Untersuchung eines Zusammenhangs zwischen nominalskalierter Variablen

### **i Hinweis**

Gefüllte Balkendiagramme eignen sich zur Analyse eines Zusammenhangs zwischen nominalskalierten Variablen. Allerdings sollte eine der beiden Variablen nur zwei Ausprägungen aufweisen, sonst sind die Zusammenhänge nicht mehr so gut zu erkennen.□

## Übungsaufgabe 6.7.

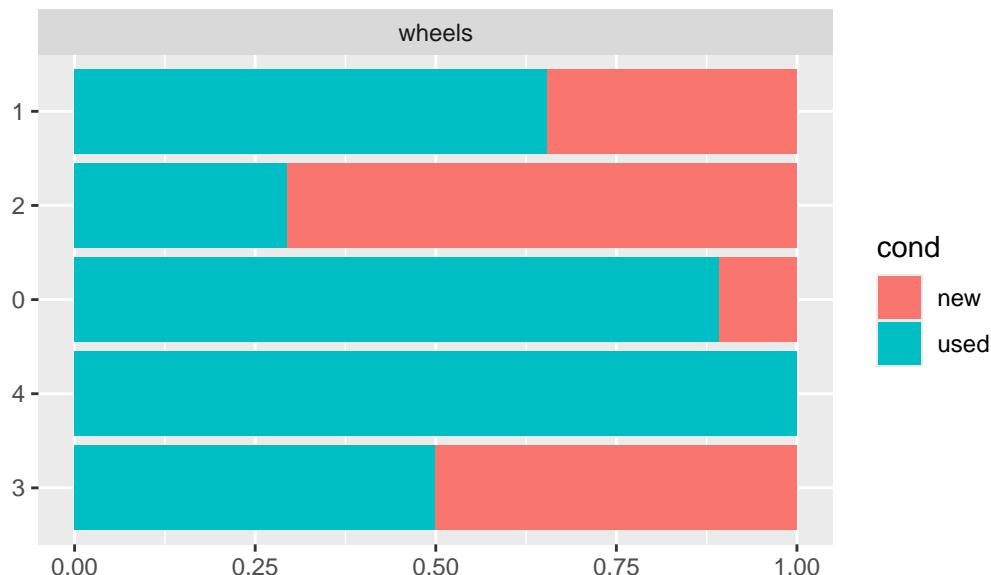
### 6.5.2. Aufgabe

Visualisieren Sie den Zusammenhang der beiden nominalen Variablen cond und wheels!

### 6.5.3. Lösung

wheels ist als metrische Variable (int: Integer, d.h. Ganzzahl) formatiert im Datensatz mariokart. Wir müssen Sie zunächst als Faktorvariable umformatieren, damit R sie als nominal skalierte Variable erkennt.

```
mariokart |>
  # Mache aus einer metrischen eine nominale Variable:
  mutate(wheels = factor(wheels)) |>
  select(cond, wheels) |>
  plot_bar(by = "cond")
```



#### 6.5.4. Zusammenhang: metrisch

Den (etwaigen) Zusammenhang zweier metrischer Variablen kann man mit einem *Streudiagramm* visualisieren (engl. scatterplot). Abbildung 6.15 links untersucht den Zusammenhang des Einstiegpreises (X-Achse) und Abschlusspreises (Y-Achse) von Geboten bei Versteigerungen des Computerspiels Mariokart. In dem Diagramm ist eine “Trendgerade” (Regressionsgerade), um die Art des Zusammenhangs besser zu verdeutlichen. Die Trendgerade steigt an (von links nach recht). Daraus kann man schließen: Es handelt sich um einen *gleichsinnigen* (positiven) Zusammenhang: Je höher der Startpreis, desto *höher* der Abschlusspreis, zumindest tendenziell. Diese Gerade liegt “mittig” in den Daten (wir definieren dies später genauer). Diese Trendgerade gibt Aufschluss über “typische” Werte: Welcher Y-Wert ist “typisch” für einen bestimmten X-Wert?

Abbildung 6.15 rechts untersucht den Zusammenhang zwischen Anzahl der Gebote (X-Achse) und Abschlusspreises (Y-Achse). Es handelt sich um einen negativen Zusammenhang: Je mehr Gebote, desto *geringer* der Abschlusspreis. Das erkennt man an der sinkenden Trendgeraden.

Die Ellipse zeigt an, wie eng die Daten um die Trendgerade streuen. Daraus kann man ableiten, wie stark der Absolutwert des Zusammenhangs ist, vgl. Abbildung 6.17.

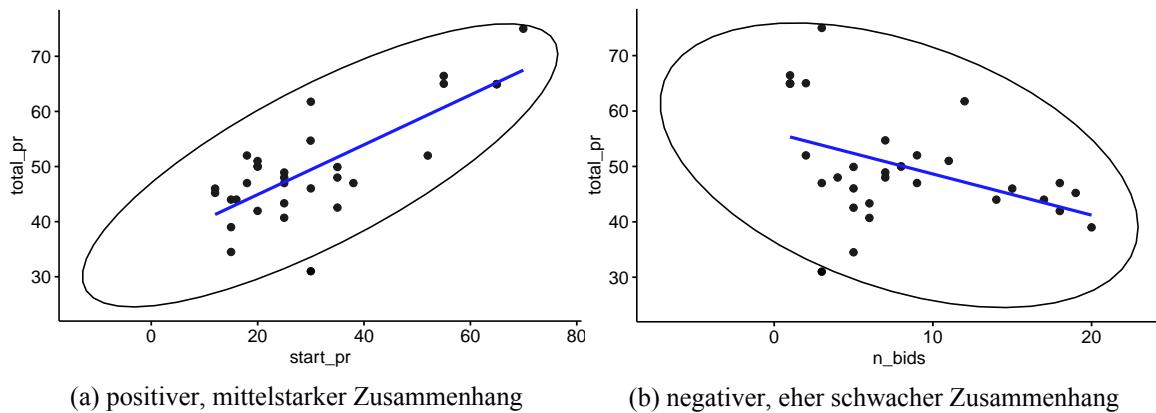


Abbildung 6.15.: Streudiagramm zur Darstellung eines Zusammenhangs zweier metrischer Variablen

**Definition 6.6** (Linearer Zusammenhang). Lässt sich die Beziehung zwischen zwei Variablen mit einer Gerade visualisieren, so spricht man von einem linearen Zusammenhang. Ändert man eine der beiden Variablen um einen bestimmten Wert (z.B. 1), so ändert sich die andere um einen proportionalen Wert (z.B. 0.5). □

Natürlich könnte man auch nicht-lineare Zusammenhänge untersuchen, aber der Einfachheit halber konzentrieren wir uns hier mit linearen; Beispiele für nicht-lineare Zusammenhänge sind in Abbildung 6.16 zu sehen.

**Definition 6.7** (Richtung und Stärke eines Zusammenhang). *Gleichsinnige* (positive) Zusammenhänge erkennt man an *aufsteigenden* Trendgeraden  $\nearrow$ ; *gegensinnige* (negative) Zusammenhänge an *absteigenden* Trendgeraden  $\searrow$ .

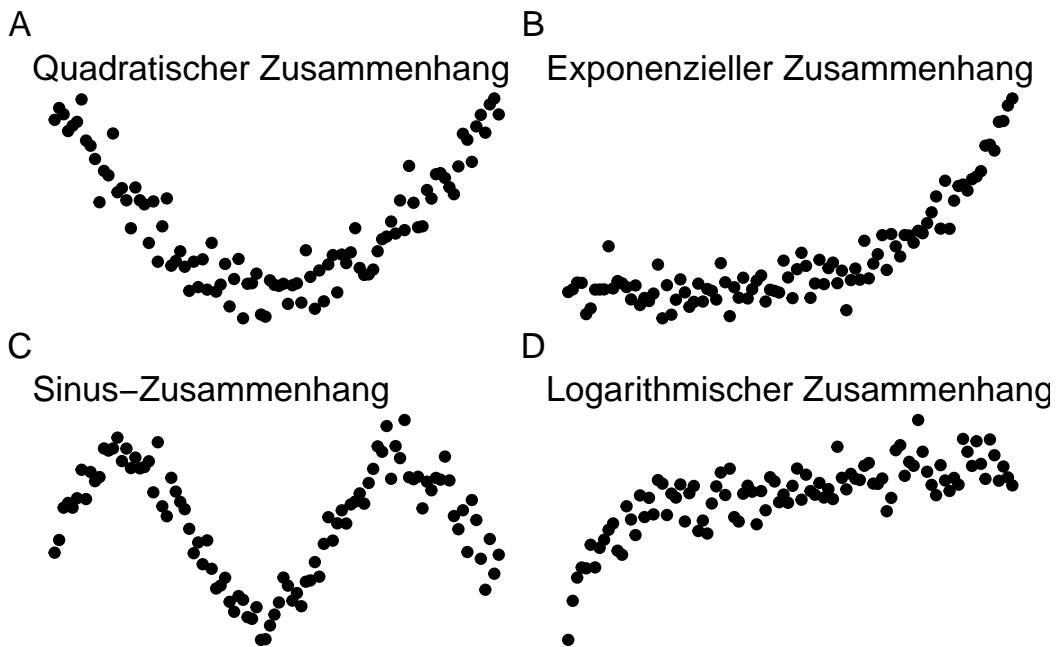


Abbildung 6.16.: Beispiele nichtlinearer Zusammenhänge

Starke Zusammenhänge erkennt man an schmalen Ellipsen (“Baguette”); schwache Zusammenhänge an breiten Ellipsen (“Torte” [□] {.content-visible when-format=“html”})

Abbildung 6.17 bietet einen Überblick über verschiedene Beispiele von Richtung und Stärke von Zusammenhängen.<sup>17</sup>

In Abbildung 6.17 ist für jedes Teildiagramm eine Zahl angegeben: der *Korrelationskoeffizient*. Diese Statistik quantifiziert Richtung und Stärke des Zusammenhangs (mehr dazu in Kap. Kapitel 9). Ein positives Vorzeichen steht für einen positiven Zusammenhang, ein negatives Vorzeichen für einen negativen Zusammenhang. Der (Absolut-)Wert gibt die Stärke des linearen Zusammenhangs an (cohen\_power\_1992?):

- $\pm 0$ : Kein Zusammenhang
- $\pm 0.1$ : schwacher Zusammenhang
- $\pm 0.3$ : mittlerer Zusammenhang
- $\pm 0.5$ : starker Zusammenhang
- $\pm 1$ : perfekter Zusammenhang

Abbildung 6.18 hat die gleiche Aussage, ist aber plakativer, indem *Stärke* (schwach, stark) und *Richtung* (positiv, negativ) gegenübergestellt sind.

Man sieht in Abbildung 6.17 und Abbildung 6.18, dass ein *negativer* Korrelationskoeffizient mit einer *absinkenden* Trendgerade<sup>18</sup> (blaue Linie) einhergeht. Umgekehrt geht ein *positiver* Trend mit einer *ansteigenden* Trendgerade einher. Zweitens erkennt man, dass *starke* Zusammenhänge mit einer *schmaler* Ellipse einhergehen und *schwache* Zusammenhänge mit einer *breiten* Ellipse einhergehen.

<sup>17</sup>Quelle: Aufbauend auf FOM/ifes, Autor: Norman Markgraf

<sup>18</sup>synonym: Regressionsgerade

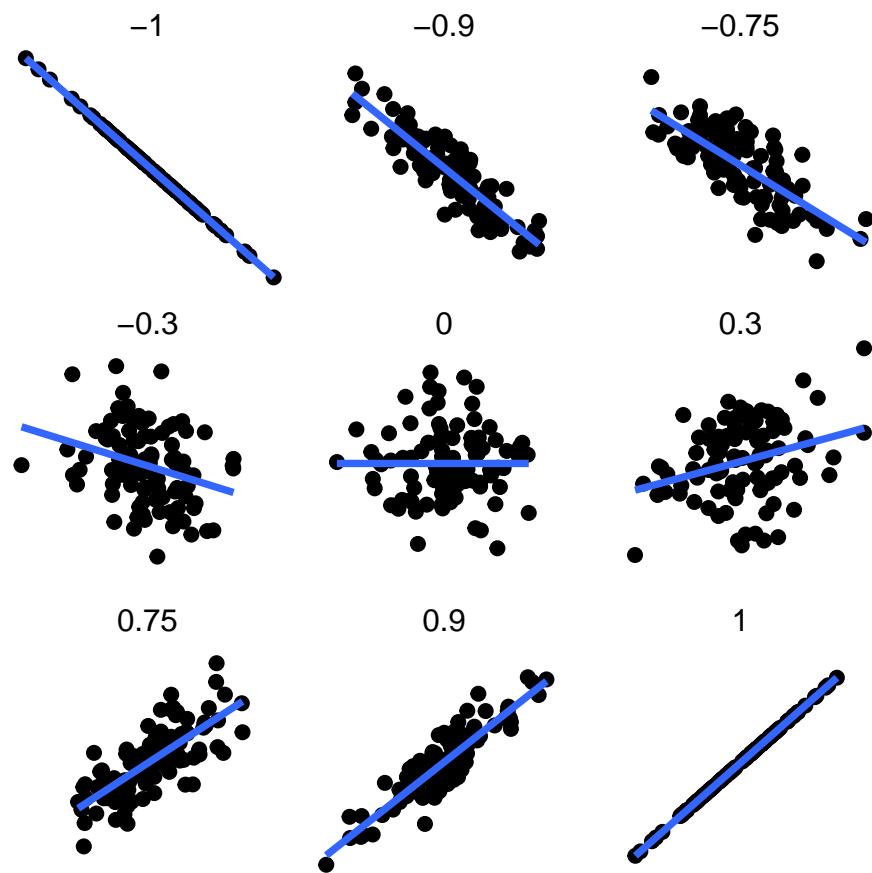


Abbildung 6.17.: Lineare Zusammenhänge verschiedener Stärke und Richtung

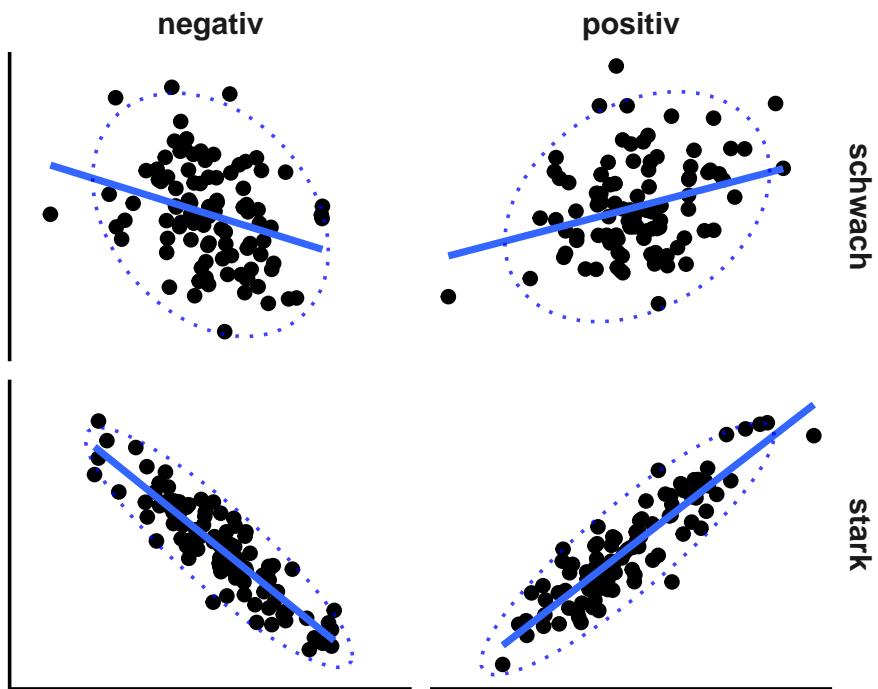


Abbildung 6.18.: Überblick über starke vs. schwache bzw. positive vs. negative Zusammenhänge

**Beispiel 6.5.** Sie arbeiten nach wie vor bei einem Online-Auktionshaus, und manchmal gehört Datenanalyse zu Ihren Aufgaben. Daher interessiert Sie, ob welche Variablen mit dem Abschlusspreis (`total_pr`) im Datensatz `mariokart` zusammenhängen. Sie verbildlichen die Daten mit R, und zwar nutzen Sie das Paket `DataExplorer`. Starten Sie dieses Paket, s. Listing 6.3. Außerdem müssen wir noch die Daten importieren, falls noch nicht getan, s. Listing 6.2.

So, jetzt kann die eigentliche Arbeit losgehen. Da Sie sich nur auf metrische Variablen konzentrieren wollen, wählen Sie (mit `select`) nur diese Variablen aus. Dann weisen Sie R an, einen Scatterplot zu malen (`plot_scatterplot`) und zwar jeweils den Zusammenhang einer der gewählten Variablen mit dem Abschlusspreis (`total_pr`), da das die Variable ist, die Sie primär interessiert. Das Ergebnis sieht man in Abbildung 6.19 bzw. Listing 6.7.

---

**Listing 6.7** Streudiagramm erstellen mit dem R-Paket ‘DataExplorer’

---

```
mariokart %>%
  select(duration, n_bids, start_pr,
         ship_pr, total_pr,
         seller_rate, wheels) %>%
  plot_scatterplot(by = "total_pr")
```

---

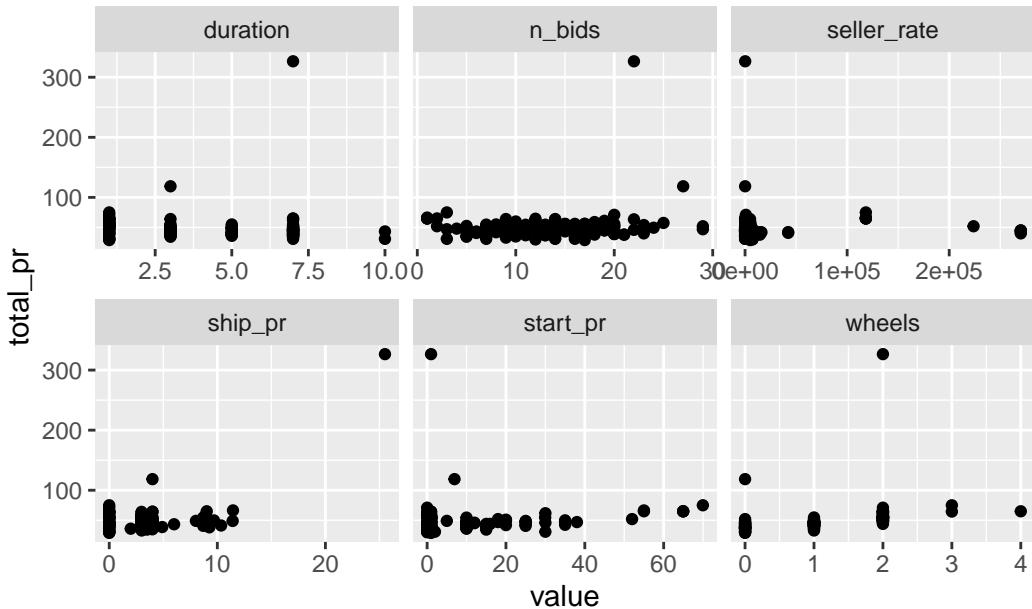


Abbildung 6.19.: Der Zusammenhang metrischer Variablen mit Abschlusspreis

Aha... Was sagt uns das Bild? Hm. Es scheint einige Extremwerte zu geben, die dafür sorgen, dass der Rest der Daten recht zusammengequetscht auf dem Bild erscheint. Vielleicht sollten Sie solche Extremwerte lieber entfernen? Sie entscheiden sich, nur Verkäufe mit einem Abschlusspreis von weniger als 100 Dollar anzuschauen ( $\text{total\_pr} < 100$ ). Das Ergebnis ist in Abbildung 6.20 zu sehen.

```
mariokart_no_extreme <-
  mariokart %>%
  filter(total_pr < 100)

mariokart_no_extreme %>%
  select(duration, n_bids, start_pr,
         ship_pr, total_pr,
         seller_rate, wheels) %>%
  plot_scatterplot(by = "total_pr")
```

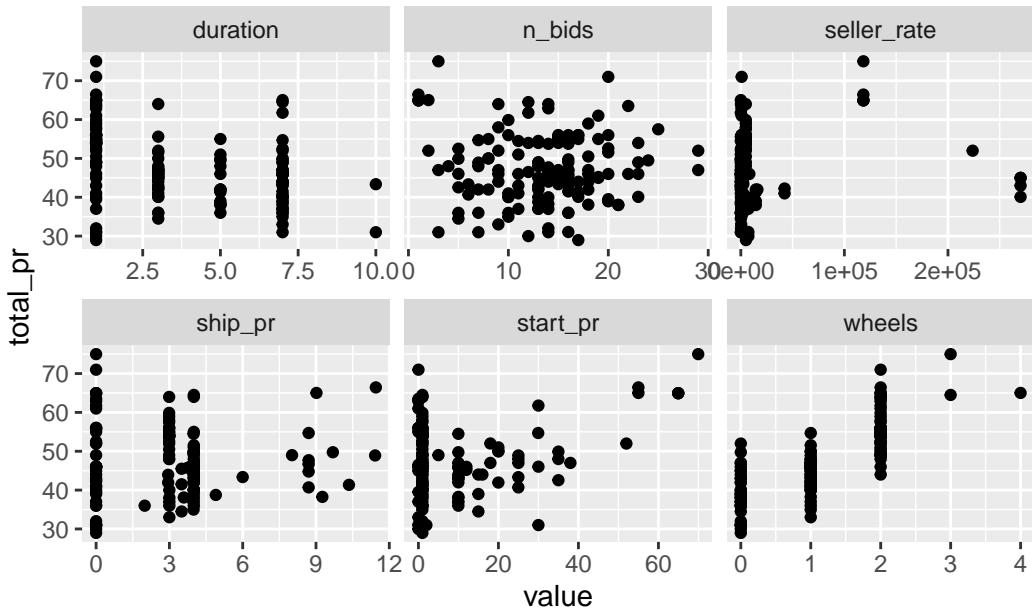


Abbildung 6.20.: Der Zusammenhang metrischer Variablen mit Abschlusspreis

Ohne Extremwerte schält sich ein deutlicheres Bild (Abbildung 6.20) hervor: Startpreis (start\_pr) und Anzahl der Räder (wheels) scheinen am stärksten mit dem Abschlusspreis zusammenzuhängen.

Das Argument `by = "total_pr"` bei `plot_scatterplot` weist R an, als Y-Variable stets `total_pr` zu verwenden. Alle übrigen Variablen kommen jeweils einmal als X-Variable vor. □

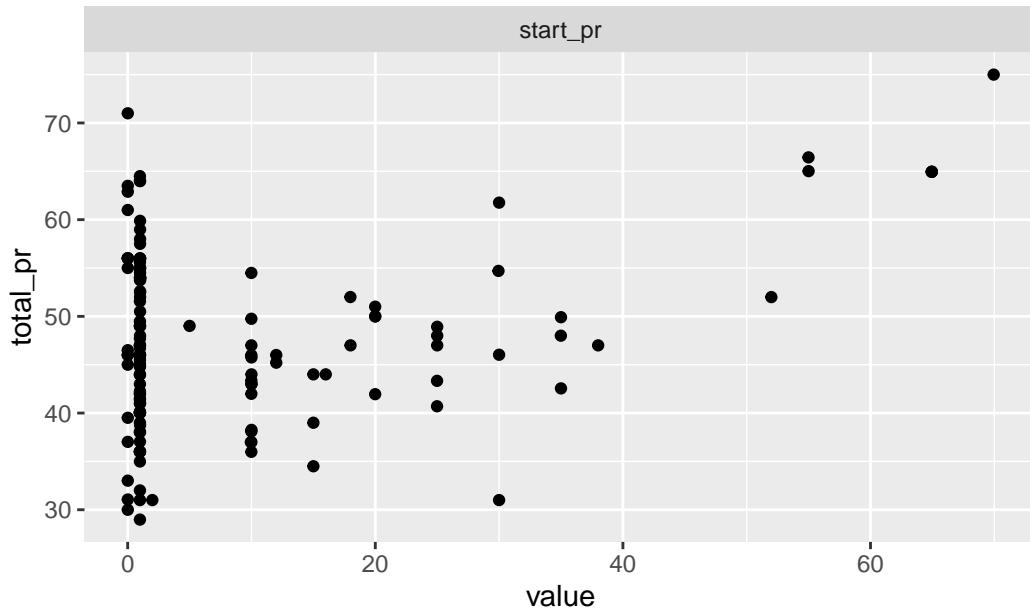
### Übungsaufgabe 6.8.

#### 6.5.5. Aufgabe

Visualisieren Sie den Zusammenhang der beiden metrischen Variablen `start_pr` und `total_pr`. Verwenden Sie den Datensatz ohne Extremwerte wie oben definiert.

#### 6.5.6. Lösung

```
mariokart_no_extreme |>
  select(start_pr, total_pr) |>
  plot_scatterplot(by = "total_pr")
```



## 6.6. Unterschiede verbildlichen

### 6.6.1. Unterschied: nominale Variablen

Gute Nachrichten: Für nominale Variablen bieten sich Balkendiagramme sowohl zur Darstellung von Zusammenhängen als auch von Unterschieden an. Genau genommen zeigt ja Abbildung 6.13 (links) den *Unterschied* zwischen neuen und gebrauchten Spielen hinsichtlich der Frage, ob Photos beiliegen. Und wie man in Abbildung 6.13 sieht, ist der Anteil der Spiele mit Foto bei den neuen Spielen höher als bei gebrauchten Spielen.<sup>19</sup>

### 6.6.2. Unterschied: quantitative Variablen

Eine typische Analysefrage ist, ob sich zwei Gruppen hinsichtlich einer metrischen Zielvariablen deutlich unterscheiden. Genauer gesagt untersucht man z.B. oft, ob sich die Mittelwerte der beiden Gruppen zwischen der Zielvariablen deutlich unterscheiden. Das hört sich abstrakt an? Am besten wir schauen uns einige Beispiele an, s. Abbildung 6.21.

Das linke Teildiagramm von Abbildung 6.21 zeigt das Histogramm von `total_pr`, getrennt für neue und gebrauchte Spiele, vgl. Abbildung 6.8. Das rechte Teildiagramm zeigt die gleichen Verteilungen, aber mit einer vereinfachten, groberen Darstellungsform, den *Boxplot*.<sup>20</sup>

<sup>19</sup>Aber Freunde lassen Freunde keine Tortendiagramme verwenden: <https://github.com/cxli233/FriendsDontLetFriends#10-friends-dont-let-friends-make-pie-chart>.

<sup>20</sup>Übrigens: Freunde lassen Freunde nicht Balkendiagramme verwenden, um Mittelwerte darzustellen: <https://github.com/cxli233/FriendsDontLetFriends#1-friends-dont-let-friends-make-bar-plots-for-means-separation>.

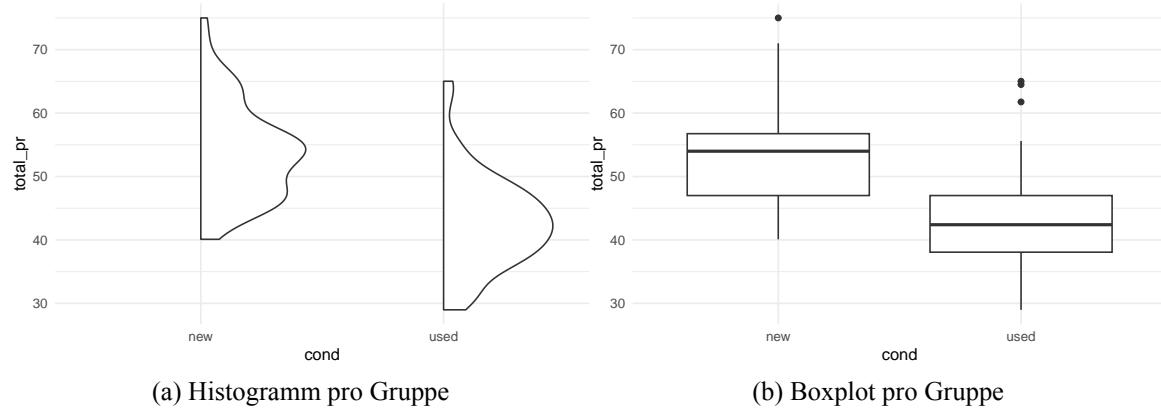


Abbildung 6.21.: Unterschiede zwischen zwei Gruppen: Metrische Y-Variable, nominale X-Variable

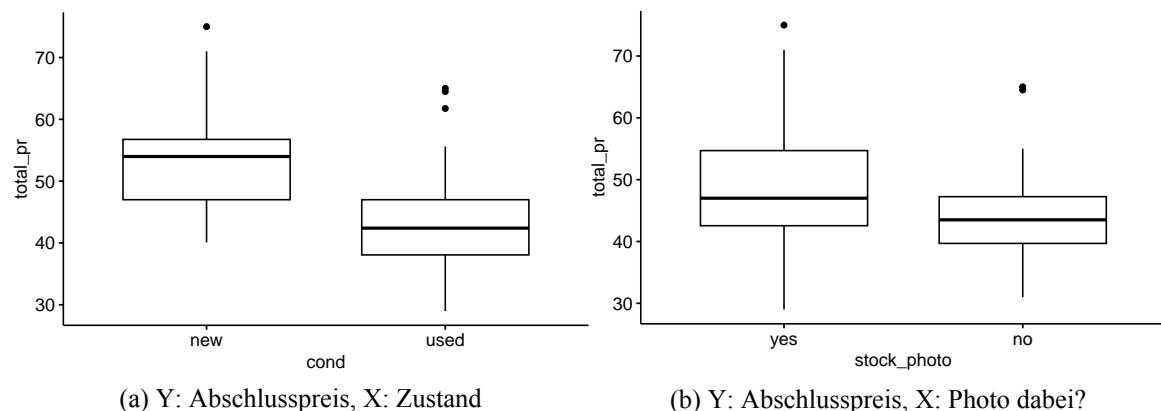


Abbildung 6.22.: Unterschiede zwischen zwei Gruppen: Metrische Y-Variable, nominale X-Variable

Das linke Teildiagramm von Abbildung 6.22 zeigt den Unterschied in den Verteilungen von `total_pr`, einmal für die neuen Computerspiele (`cond == new`) und einmal für gebrauchte Spiele (`cond == used`).

Was ein “deutlicher”<sup>21</sup> Zusammenhang ist, ist keine statistische, sondern inhaltliche Frage, die man mit Sachverstand zum Forschungsgegenstand beantworten muss.

**Definition 6.8** (Boxplot). Der Boxplot ist eine Vereinfachung bzw. eine Zusammenfassung eines Histogramms.<sup>22</sup> Damit stellt der Boxplot auch eine Verteilung (einer metrischen Variablen) dar. □

In Abbildung 6.23 sieht man die “Übersetzung” von Histogramm (oben) zu einem Boxplot (unten).

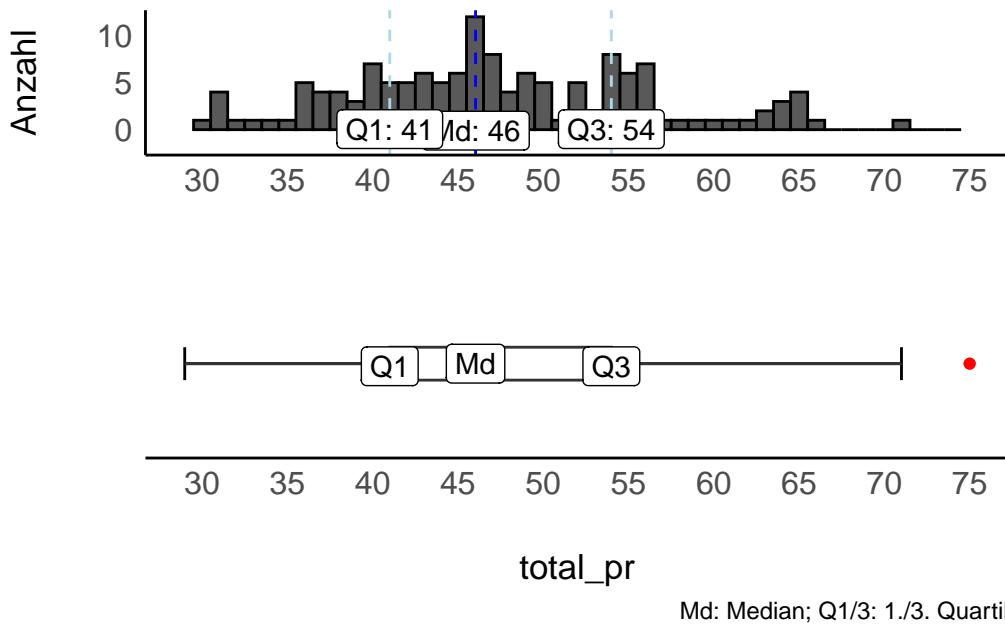


Abbildung 6.23.: Übersetzung eines Histogramms zu einem Boxplot

Schauen wir uns die “Anatomie” des Boxplots näher an:

1. Der *dicke Strich* in der Box zeigt den Median der Verteilung, vgl. Kapitel 7.3.
2. Die *Enden der Box* zeigen das 1. Quartil (41) bzw. das 3. Quartil (54). Damit zeigt die Breite der Box die Streuung der Verteilung an, genauer gesagt die Streuung der inneren 50% der Beobachtungen. Je breiter die Box, desto größer die Streuung. Die Breite der Box nennt man auch den *Interquartilsabstand* (IQR).
3. Die “*Antennen*” des Boxplots zeigen die Streuung in den kleinsten 25% der Werte (linke Antenne) bzw. die Streuung der größten 25% der Werte (rechte Antennen). Je länger die Antenne, desto größer die Streuung.
4. Falls es aber *Extremwerte* gibt, so sollten die lieber einzeln, separat, außerhalb der Antennen gezeigt werden. Daher ist die Antennenlänge auf die 1,5-fache Länge der Box beschränkt.

<sup>21</sup>“substanzieller”, “bedeutsamer”, “relevanter” oder “(inhaltlich) signifikanter”

<sup>22</sup>Ob der Boxplot horizontal oder vertikal steht, ist Ihrem Geschmack überlassen.

Werte die außerhalb dieses Bereichs liegen (also mehr als das 1,5-fache der Boxlänge von Q3 entfernt sind) werden mittels eines Punktes dargestellt.

5. Liegt der Median-Strich in der Mitte der Box, so ist die Verteilung *symmetrisch* (bezogen auf die inneren 50% der Werte), liegt der Median-Strich nicht in der Mitte der Box, so ist die Verteilung nicht symmetrisch (d.h. sie ist *schief*). Gleiches gilt für die Antennenlängen: Sind die Antennen gleich lang, so ist der äußere Teil der Verteilung symmetrisch, andernfalls schief.

**Beispiel 6.6.** In einer vorherigen Analyse haben Sie den Zusammenhang von Abschlusspreis und der Anzahl der Lenkräder untersucht. Jetzt möchten Sie eine sehr ähnliche Fragestellung betrachten: Wie *unterscheiden* sich die Verkaufspreise je nach Anzahl der beigelegten Lenkräder? Flink erstellen Sie dazu folgendes Diagramm, Abbildung 6.24, links. Es zeigt die Verteilung des Abschlusspreises, aufgebrochen nach Anzahl Lenkräder (`by = "wheels"`). □

Aber ganz glücklich sind Sie mit dem Diagramm nicht: R hat die Variable `wheels` komisch aufgeteilt. Es wäre eigentlich ganz einfach, wenn R die Gruppen 0, 1, 2, 3 und 4 aufteilen würde. Aber schaut man sich die Y-Achse (im linken Teildiagramm von Abbildung 6.24) an, so erkennt man, dass R `wheels` als stetige Zahl betrachtet und nicht in ganze Zahlen gruppiert.<sup>23</sup> Aber wir möchten jeden einzelnen Wert von `wheels` (0, 1, 2, 3, 4) als *Gruppe* verstehen. Mit anderen Worten, wir möchten `wheels` als nominale Variable definieren. Das kann man mit dem Befehle `factor(wheels)` erreichen (verpackt in `mutate`), s. Abbildung 6.24 rechts.

```
mariokart_no_extreme %>%
  select(total_pr, wheels) %>%
  plot_boxplot(by = "wheels")

mariokart_no_extreme %>%
  select(total_pr, wheels) %>%
  mutate(wheels = factor(wheels)) %>%
  plot_boxplot(by = "wheels")
```

Sie schließen aus dem Bild, dass Lenkräder und Preis (positiv) zusammenhängen. Allerdings scheint es wenig Daten für `wheels == 4` zu geben. Das prüfen Sie nach:

```
mariokart_no_extreme %>%
  count(wheels)
```

| wheels | n  |
|--------|----|
| 0      | 36 |
| 1      | 52 |
| 2      | 50 |

<sup>23</sup>Vielleicht so, dass in jeder Gruppe gleich viele Wert sind?

| wheels | n |
|--------|---|
| 3      | 2 |
| 4      | 1 |

Tatsächlich gibt es (in `mariokart_no_extreme`) auch für 3 Lenkräder schon wenig Daten, so dass wir die Belastbarkeit dieses Ergebnisses skeptisch betrachten sollten.

Übrigens bezeichnet Sie Ihre Chefin nur noch als "Datengott".

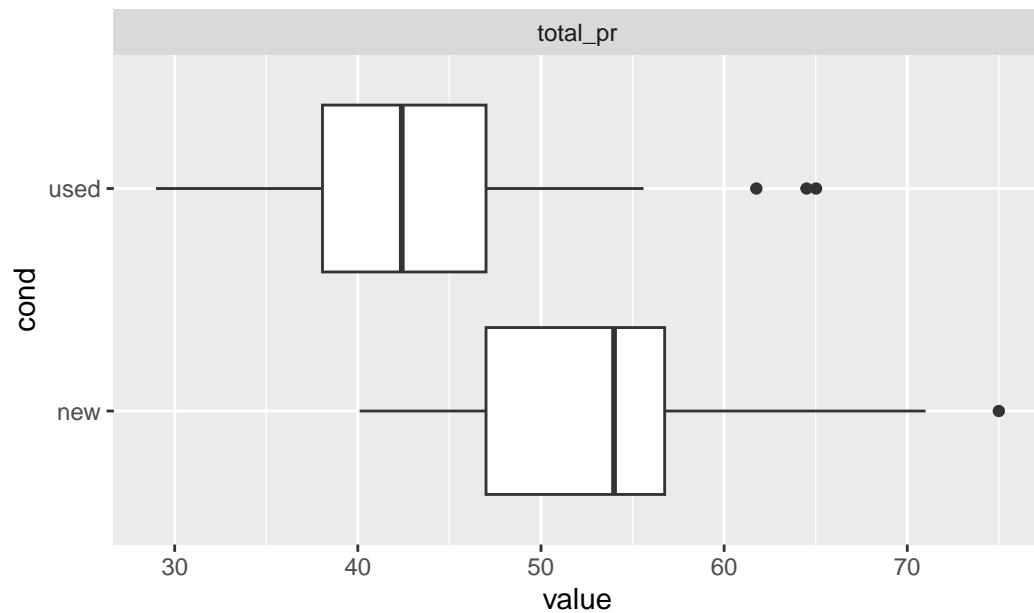
### Beispiel 6.7.

#### 6.6.3. Aufgabe

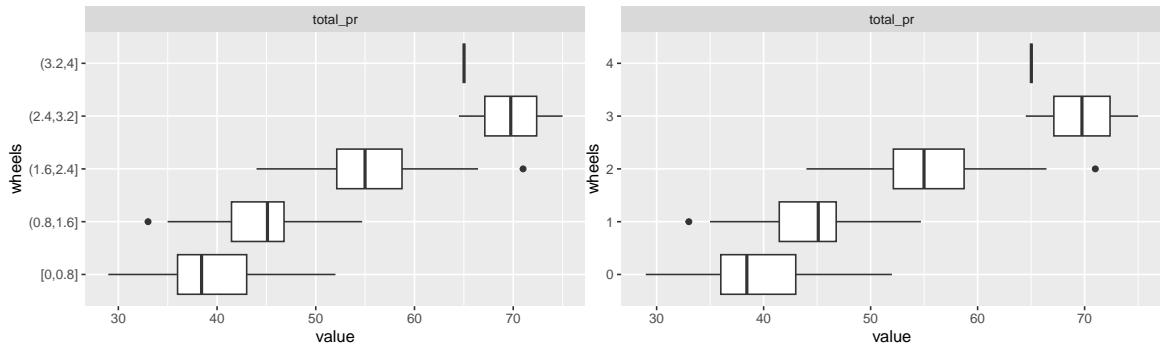
Visualisieren Sie den Unterschied im Verkaufspreis zwischen gebrauchten und neuen Spielen.

#### 6.6.4. Lösung

```
mariokart_no_extreme |>
  select(cond, total_pr) |>
  plot_boxplot(by = "cond")
```



### Beispiel 6.8.



(a) wheels als metrische Variable

(b) wheels als nominale Variable

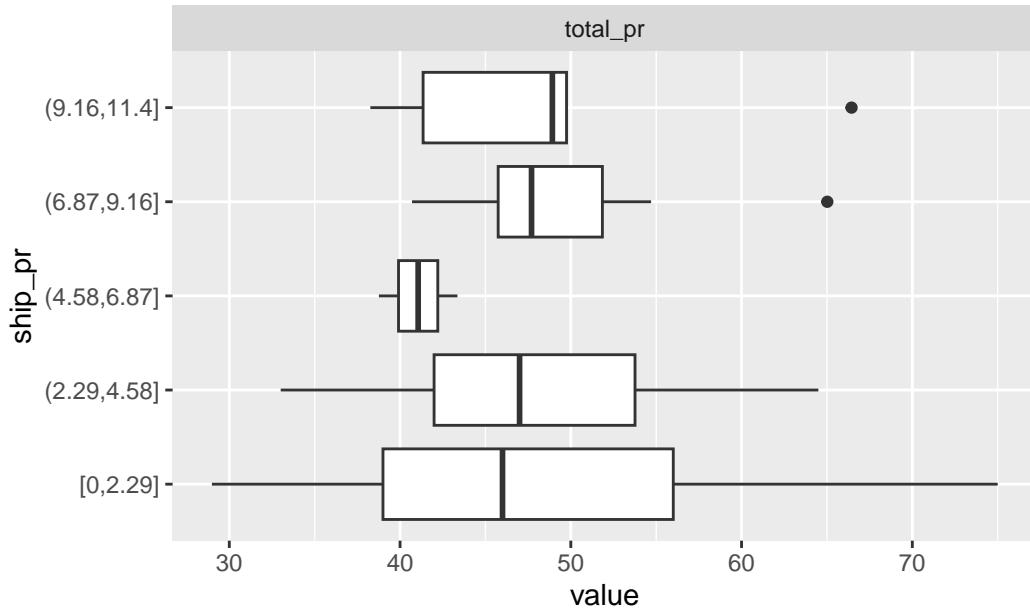
Abbildung 6.24.: Abschlusspreis nach Anzahl von beigelegten Lenkrädern

### 6.6.5. Aufgabe

Visualisieren Sie den Unterschied im Verkaufspreis abhängig von `ship_pr`; betrachten Sie `ship_pr` als eine Gruppierungsvariable. Interpretieren Sie das Ergebnis.

### 6.6.6. Lösung

```
mariokart_no_extreme |>
  select(ship_pr, total_pr) |>
  plot_boxplot(by = "ship_pr")
```



`plot_boxplot` gruppierter *metrische* Variablen, wie `ship_pr` automatisch in fünf Gruppen (mit gleichen Ranges). Wir müssen also nichts tun, um die metrische Variable `ship_pr` in eine Gruppierungsvariable (Faktorvariable) umzuwandeln.

Es sieht so aus, als würde der Median zwischen den Gruppen leicht steigen, mit Ausnahme der mittleren Gruppe.

## 6.7. So lügt man mit Statistik

Diagramme werden häufig eingesetzt, um die Wahrheit “aufzuhübschen”.

### 6.7.1. Achsen manipulieren

Achsen zu stauchen ist ein einfacher Trick, s. Abbildung 6.25.

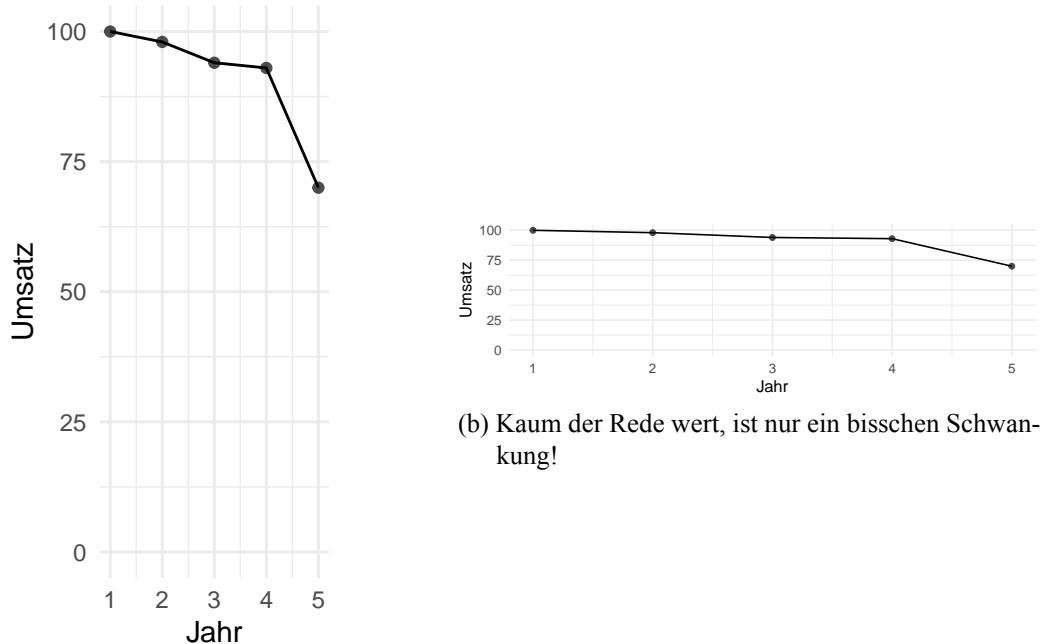


Abbildung 6.25.: Stauchen der Y-Achse, um mit Statistik zu lügen

Natürlich kann man auch durch “Abschneiden” der Y-Achse einen eindrucksvollen Effekt erzielen, s. Abbildung 6.26.

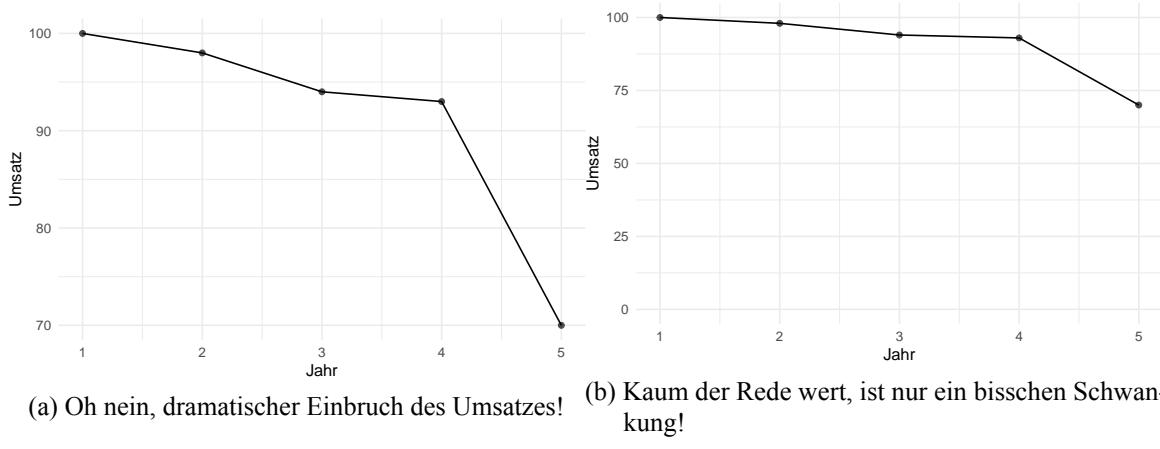


Abbildung 6.26.: Abschneiden der Y-Achse, um mit Statistik zu lügen

### 6.7.2. Scheinkorrelation

Messerli (2012) berichtet von einem Zusammenhang von Schokoladenkonsum und Anzahl von Nobelpreisen (Beobachtungseinheit: Länder), s. Abbildung 6.27. Das ist doch ganz klar: Schoki futtern macht schlau und Nobelpreise! (?)

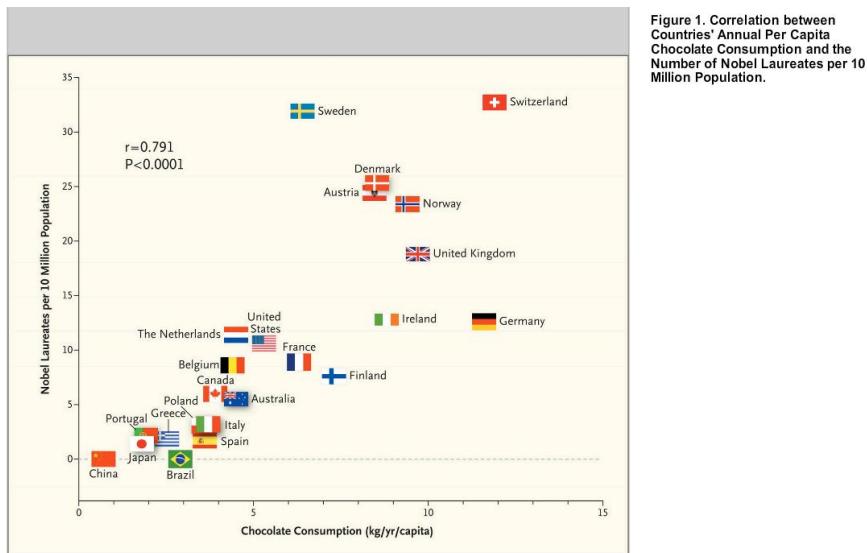


Abbildung 6.27.: Schokoladenkonsum und Nobelpreise

Leider ist hier von einer *Scheinkorrelation* auszugehen: Auch wenn die beiden Variablen *Schokoladenkonsum* und *Nobelpreise* zusammenhängen, heißt das *nicht*, dass die Variable die Ursache und die andere die Wirkung sein muss. So könnte auch eine Drittvariable im Hintergrund die gleichzeitige Ursache von Schokoladenkonsum und Nobelpreise sein, etwa der *allgemeine Entwicklungsstand* des Landes: In höher entwickelten Ländern wird mehr Schokolade konsumiert und es werden mehr Nobelpreise gewonnen im Vergleich zu Ländern mit geringerem Entwicklungsstand.

## 6.8. Praxisbezug

Ein, wie ich finde schlagendes Beispiel zur Stärke von Datendiagrammen ist Abbildung 6.28. Das Diagramm zeigt die Häufigkeit von Masern, vor und nach der Einführung der Impfung. Die Daten und die Idee zur Visualisierung gehen auf ([van\\_panhuis\\_contagious\\_2013?](#)) zurück. Das Diagramm und weitere finden sich in ähnlicher Form im [Wall Street Journal](#).

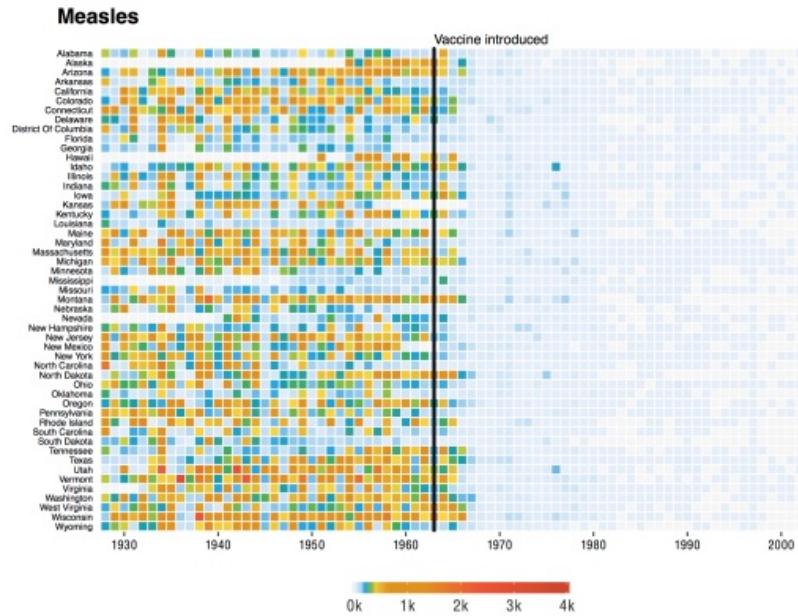


Abbildung 6.28.: Häufigkeit von Masern und Impfung in den USA, Lizenz: MIT

Quellcode<sup>24</sup>

In der “freien Wildbahn” findet man häufig sog. “Tortendiagramme”. Zwar sind sie beliebt, doch ist von ihrer Verwendung zumeist abzuraten; vgl. auch [hier](#).<sup>25</sup>

## 6.9. Vertiefung

Mehr Informationen zu {DataExplorer} finden Sie [hier](#).

### 6.9.1. Schicke Diagramme

Ein Teil der Diagramm dieses Kapitels wurden mit dem R-Paket `ggbpbr` erstellt. Mit diesem Paket lassen sich einfach ansprechende Datendiagramme erstellen, so lautet die etwa die Syntax von

<sup>24</sup>Datenquelle: <https://www.tycho.pitt.edu>

<sup>25</sup><https://www.data-to-viz.com/caveat/pie.html>; <https://github.com/cxli233/FriendsDontLetFriends#10-friends-dont-let-friends-make-pie-chart>

Abbildung 6.22 wie folgt.

```
library(ggpubr) # einmalig installieren nicht vergessen
mariokart %>%
  filter(total_pr < 100) %>%
  ggboxplot(x = "cond", y = "total_pr")
```

Möchte man Mittelwerte vergleichen, so sind Boxplots nicht ideal, da diese ja nicht den Mittelwert, sondern den *Median* heraustellen. Eine Abhilfe (also eine Darstellung des Mittelwerts) schafft man (z.B.) mit ggpublisher, s. Abbildung 6.29.

```
ggviolin(mariokart_no_extreme,
          x = "cond",
          y = "total_pr",
          add = "mean_sd")
```

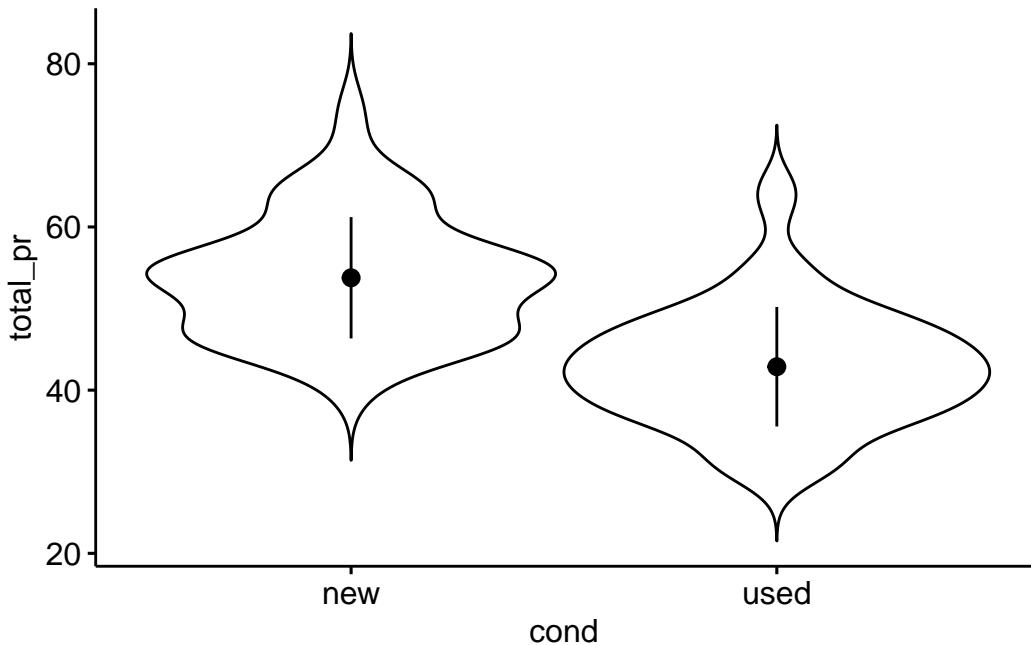


Abbildung 6.29.: Vergleich der Verteilungen zweier Gruppen mit Mittelwert und Standardabweichung pro Gruppe hervorgehoben

Ein ‘‘Violinenplot’’ hat die gleiche Aussage wie ein Dichtediagramm: Je breiter die ‘‘Violine’’, desto mehr Beobachtungen gibt es an dieser Stelle. Weitere Varianten zum Violinenplot mit ggpublisher finden sich [hier](#).<sup>26</sup>

<sup>26</sup><https://rpkgs.datanovia.com/ggpubr/reference/ggviolin.html>

Übrigens sind Modelle – und Diagramme sind Modelle – immer eine Vereinfachung, lassen also Informationen weg. Manchmal auch wichtige Informationen. Dieses Beispiel zeigt, wie etwa Histogramme wichtige Informationen unter den Tisch fallen lassen.<sup>27</sup>

Ein weiteres R-Paket zur Erstellung ansprechender Datenvisualisierung heißt `ggstatsplot`.<sup>28</sup>

Abbildung 6.30 zeigt ein **Histogramm**, das mit `ggstatsplot` erstellt wurde.<sup>29</sup>

```
library(ggstatsplot)

gghistostats(
  data      = mariokart_no_extreme,
  x         = total_pr,
  xlab     = "Verkaufspreis"
  # results.subtitle = FALSE    # unterdrückt statist. Details
)
```

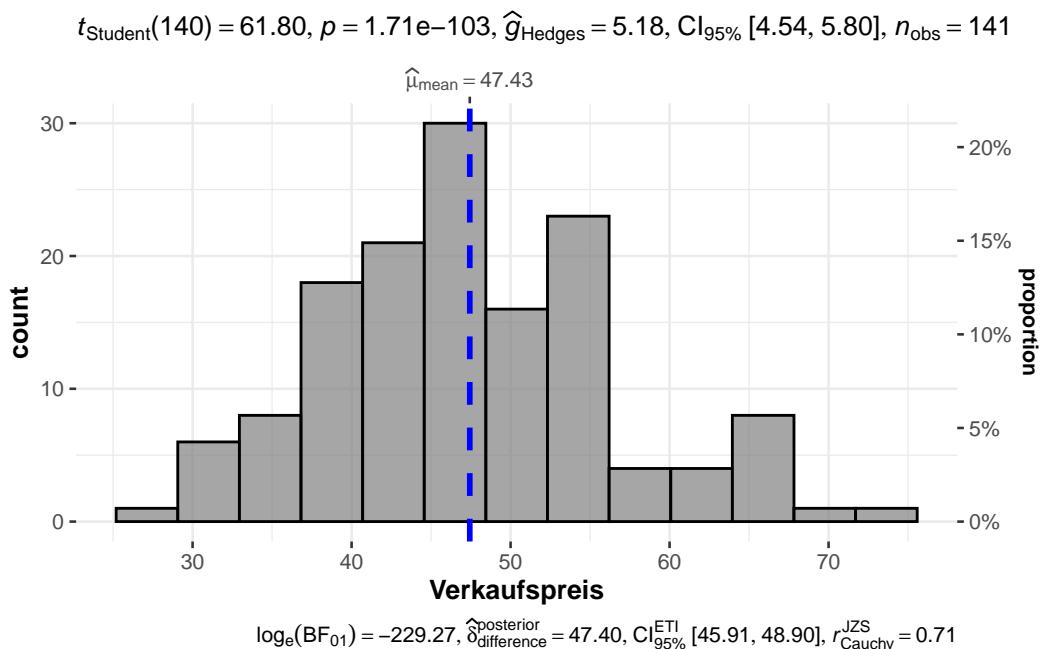


Abbildung 6.30.: Ein Histogramm mit `ggstatsplot`

Die Menge der statistischen Kennzahlen bei `ggstatsplot` schindet ordentlich Eindruck. Möchte man auf die Kennzahlen verzichten, so nutzt man den Schalter `results.subtitle = FALSE`.<sup>27</sup> Weitere Hinweise finden sich [auf der Hilfeseite der Funktion: [https://indrajeetpatil.github.io/ggstatsplot/articles/web\\_only/gghistostats.html](https://indrajeetpatil.github.io/ggstatsplot/articles/web_only/gghistostats.html)].

<sup>27</sup><https://www.autodesk.com/research/publications/same-stats-different-graphs>

<sup>28</sup><https://github.com/IndrajeetPatil/ggstatsplot/blob/main/README.md>

<sup>29</sup><https://github.com/IndrajeetPatil/ggstatsplot/blob/main/README.md#gghistostats>

💡 Ich würde gerne mal Beispiele von *schlechten* Datendiagrammen sehen.

💡 Auf der Seite von [Flowingdata](https://flowingdata.com/category/visualization/ugly-visualization/) findet sich eine nette Liste mit schlechten Datendiagrammen.<sup>30</sup>

### 6.9.2. Farbwahl

Einige Überlegungen zur Farbwahl findet sich bei Wilke (2019), s. Kap. 4.<sup>31</sup> Die Farbpalette von Okabe und Ito ist (vgl. [ichihara\\_color\\_2008?](#)) empfehlenswert, da sie auch bei Schwarz-Weiß-Druck und bei Sehschwächen die Farben noch recht gut unterscheiden lässt, s. Abbildung 6.31.

```
mariokart %>%  
  filter(total_pr < 100) %>%  
  ggboxplot(x = "cond", y = "total_pr", fill = "cond") +  
  scale_fill_okabaito()
```

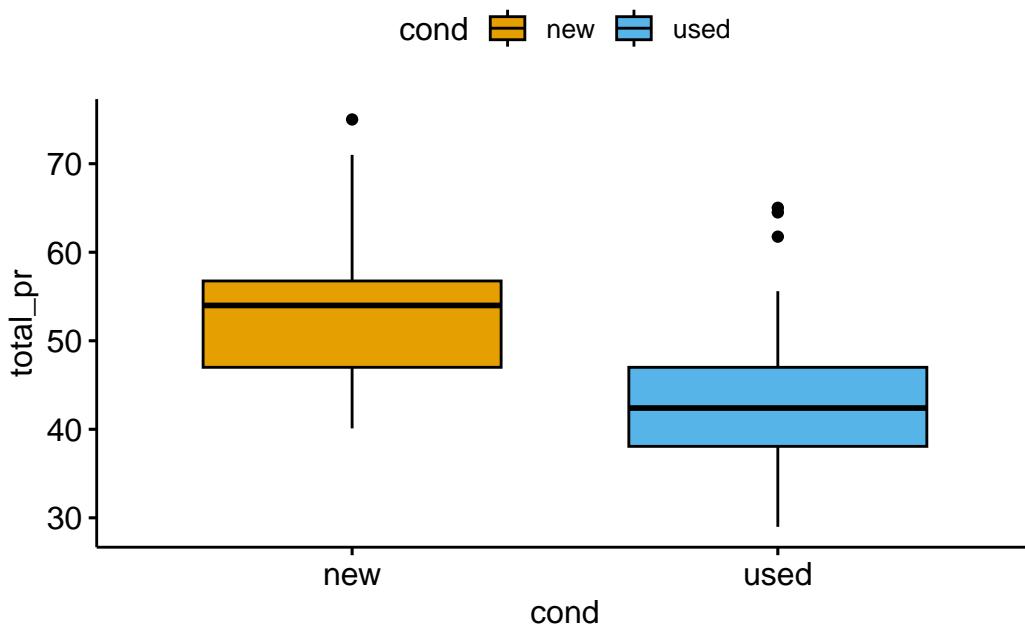


Abbildung 6.31.: Die Farbskala von Okabe und Ito: Geeignet bei Farbseh-Schwächen und für Schwarz-Weiß-Druck. Außerdem nett anzuschauen.

## 6.10. Aufgaben

Die Webseite [datenwerk.netlify.app](https://datenwerk.netlify.app) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

<sup>30</sup><https://flowingdata.com/category/visualization/ugly-visualization/>

<sup>31</sup>Siehe auch: <https://data-se.netlify.app/2023/06/30/farbpallen/>

1. [boxhist](#)
2. [max-corr1](#)
3. [max-corr2](#)
4. [Histogramm-in-Boxplot](#)
5. [Diamonds-Histogramm-Vergleich2](#)
6. [Boxplot-Aussagen](#)
7. [boxplots-de1a](#)
8. [movies-vis1](#)
9. [movies-vis2](#)
10. [vis-gapminder](#)
11. [boxplots-de1a](#)
12. [diamonds-histogramm-vergleich](#)
13. [wozu-balkendiagramm](#)
14. [diamonds-histogram](#)
15. [n-vars-diagram](#)

Noch mehr Aufgaben zum Thema Datenvisualisierung finden Sie im Datenwerk unter dem Tag [vis](#).

## 6.11. Literaturhinweise

Sowohl `ggpubr` als auch `DataExplorer` (und viele andere R-Pakete) bauen auf dem R-Paket `ggplot2` auf. `ggplot2` ist eines der am weitesten ausgearbeiteten Softwarepakete zur Erstellung von Datendiagrammen. Das Buch zur Software (vom Autor von `ggplot2`) ist empfehlenswert ([wickham\\_ggplot2\\_2009?](#)). Eine neue, gute Einführung in Datenvisualisierung findet sich bei Wilke (2019). Beide Bücher sind kostenfrei online lesbar.

Wilke (2019) gibt einen hervorragenden Überblick über praktische Aspekte der Datenvisualisierung; gut geeignet, wenn man mit R arbeitet. In ähnlicher Richtung geht Fisher und Meyer (2018). [Hier](#) ist eine Liste von Büchern zum Thema; dort können Sie bei Interesse tiefer suchen.

## 6.12. Literatur

# 7. Punktmodelle 1

## 7.1. Lernsteuerung

### 7.1.1. Standort im Lernpfad

Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 7.1.2. Lernziele

- Sie können gängige Arten von Lagemaße definieren.
- Sie können erläutern, inwiefern man ein Lagemaß als ein Modell hernehmen kann.
- Sie können Lagemaße mit R berechnen.

### 7.1.3. Benötigte R-Pakete

In diesem Kapitel benötigen Sie folgende R-Pakete.

```
library(tidyverse)
library(easystats)
```

### 7.1.4. Benötigte Daten

```
mariokart_path <- paste0(
  "https://vincentarelbundock.github.io/Rdatasets/",
  "csv/openintro/mariokart.csv")

mariokart <- read.csv(mariokart_path)
```

## 7.2. Mittelwert als Modell

Der klassische Mittelwert (arithmetisches Mittel) ist ein prototypisches Beispiel für ein Modell in der Statistik.

**Übungsaufgabe 7.1.** Welche Vorstellung haben Sie, wenn Sie hören, dass der “typische deutsche Mann” 1,80m groß ist (Roser, Appel, und Ritchie 2013)?<sup>1</sup>

- a) Die Hälfte der Männer ist größer als 1,80m, die andere Hälfte kleiner.
- b) Das arithmetische Mittel der Männer beträgt 1,80m.
- c) Die meisten Männer sind 1,80m groß.
- d) Etwas anderes.
- e) Keine Ahnung! □

**Übungsaufgabe 7.2.** Laut [dieser Quelle](#) beträgt der Wert der mittleren Größe deutscher Frauen etwa 1,66m, also 14 cm weniger als bei Männern.<sup>2</sup> □

### 7.2.1. Frage

Ist das viel?

- a) ja
- b) nein
- c) kommt drauf an
- d) weiß nicht □

### 7.2.2. Antwort

Auf dieser Frage gibt es keine Antwort, zumindest nicht ohne weitere Annahmen. So könnte man z.B. sagen, “mehr als 5 cm sind viel”. So eine Entscheidung ist aber keine statistische Angelegenheit, sondern eine inhaltliche.

**Beispiel 7.1** (Beispiel zum Mittelwert). Ein Statistikkurs besteht aus drei Studentinnen: Anna, Berta und Carla. Sie haben gerade ihre Noten in der Klausur erfahren. Anna hat eine 1, Berta eine 2 und Carla eine 3. Der Durchschnitt (das arithmetische Mittel,  $\bar{x}$ ) beträgt: 2. □

💡 Zu easy!

💡 Schon gut! Chill mal. Wird gleich interessanter.

Die Rechenregel zum Mittelwert lautet:

1. Addiere alle Werte

<sup>1</sup>Ihr Vorstellung updateet sich in Definition 7.1.

<sup>2</sup>[https://en.wikipedia.org/wiki/Average\\_human\\_height\\_by\\_country](https://en.wikipedia.org/wiki/Average_human_height_by_country)

2. Teile durch die Anzahl der Werte
3. Fertig.  $\square$

Etwas abstrakter kann man Beispiel 7.1 in folgendem Schaubild darstellen, s. Gleichung 7.1.

$$\begin{array}{c} \boxed{\square} \\ \boxed{\square} \end{array} + \begin{array}{c} \boxed{\square} \\ \boxed{\square} \end{array} + \begin{array}{c} \boxed{\square} \\ \boxed{\square} \\ \boxed{\square} \end{array} = 3 \cdot \begin{array}{c} \boxed{\square} \\ \boxed{\square} \end{array} \quad (7.1)$$

Der Nutzen des Mittelwerts liegt darin, dass er uns ein Bild gibt (ein Modell ist!) für die “typische Note” im Statistikkurs, s. Gleichung 7.2.

$$\begin{array}{c} \boxed{\square} \\ \boxed{\square} \end{array} + \begin{array}{c} \boxed{\square} \\ \boxed{\square} \end{array} + \begin{array}{c} \boxed{\square} \\ \boxed{\square} \\ \boxed{\square} \end{array} \leftrightarrow \begin{array}{c} \boxed{\square} \\ \boxed{\square} \end{array} \text{ „typischer Vertreter“} \quad (7.2)$$

### ! Wichtig

Der Nutzen des Mittelwerts liegt darin, dass er eine Datenreihe zu einen “typischen Vertreter” zusammenfasst. Er ist typisch in dem Sinne, als dass die Werte aller Merkmalsträger in gleichem Maße einfließen. Er gibt uns eine (mögliche) Vorstellung (ein Modell!), wie wir uns die Werte der Datenreihe vorstellen sollen.

Eine nützliche Anschauung zum Mittelwert ist die Vorstellung des Mittelwerts als eine ausbalancierte Wippe, s. Abbildung 7.1.

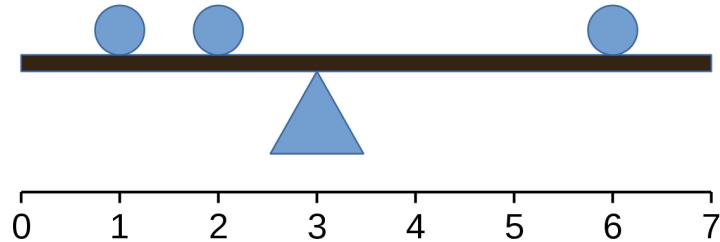


Abbildung 7.1.: Mittelwert als ausbalancierte Wippe mit Mittelwert 3

Quelle: Von Maphry - Eigenes Werk, CC BY-SA 4.0

In "Mathe-Sprech" bezeichnet man den Mittelwert häufig mit  $\bar{x}$  und schreibt die Rechenregel so, s. Gleichung 7.3.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n} \quad (7.3)$$

**Definition 7.1** (Mittelwert). Der Mittelwert (MW, mean) der Variablen  $X$  (präziser: das arithmetische Mittel des Merkmal  $X$ ) ist definiert als die Summe der Elemente von  $X$  geteilt durch deren Anzahl,  $n$ . Den Mittelwert von  $X$  bezeichnet man auch mit  $\bar{x}$ .  $\square$

**Beispiel 7.2.** Angenommen wir haben eine Reihe von Noten: 1,2,3. Der Mittelwert der Noten beträgt dann 2:  $\bar{X} = \frac{1}{3} \sum(1 + 2 + 3) = 6/3 = 2$ .  $\square$

Da der Mittelwert eine zentrale Rolle spielt in der Statistik, sollten wir ihn uns noch etwas genauer anschauen. In s. Abbildung 7.2 sehen wir die Noten von (dieses Mal) vier Studentis. Die gestrichelte horizontale Linie zeigt den Mittelwert der vier Noten. Die schwarzen Punkte sind die Daten, in dem Fall die einzelnen Noten. Die vertikalen Linien zeigen die Abweichungen der Noten zum Mittelwert. Bezeichnen wir die Abweichung – auch als "Fehler", "Rest" oder "Residuum" bezeichnet – der  $i$ -ten Person mit  $e_i$  ( $e$  wie engl. *error*, Fehler) und die  $i$ -te Note mit  $y_i$ , so können wir mit Gleichung 7.4 festhalten:

$$y_i = \bar{x} + e_i \quad (7.4)$$

Anders ausgedrückt (s. Gleichung 7.5):

$$\text{Daten} = \text{Modell} + \text{Rest} \quad (7.5)$$

Der Mittelwert ist hier unser Modell der Daten. Wie gesagt: Ein Modell ist eine vereinfachte (zusammengefasste) Beschreibung einer Datenreihe.

Um Modelle darzustellen, wird in der Datenanalyse häufig folgende Art von Modellgleichung verwendet, s. Gleichung 7.6.

$$\hat{y} \sim x \quad (7.6)$$

Lies: "Der Modellwert  $\hat{y}$  ist eine Funktion der Variable  $x$ ". Der Kringel " $\sim$ "<sup>3</sup> soll also hier heißen "... ist eine Funktion von ...".

Mit  $\hat{y}$  ist die vorhergesagte bzw. die zu erklärende Variable<sup>4</sup> gemeint. Das "Dach" über dem  $y$  bedeutet "vorhergesagter Y-Wert" oder "Y-Wert laut dem Modell". Der tatsächliche, beobachtete Wert  $y$  setzt sich zusammen aus dem Modellwert  $m$  plus einem Fehler  $e$ , s. Gleichung 7.7.

<sup>3</sup>Das "Kringel" oder die "Welle" " $\sim$ " nennt man auch "Tilde".

<sup>4</sup>AV, Output-Variable, Zielvariable

$$y = m + e \quad (7.7)$$

Anstelle von  $m$  schreibt man auch  $\hat{y}$  ("y-Dach"). In diesem Fall ist das Modell einfach gleich dem Mittelwert (und nicht irgendeiner Funktion des Mittelwerts), so dass wir mit Gleichung 7.8 schreiben können:

$$y = \bar{x} + e \quad (7.8)$$

Die Zielvariable  $y$  wird also durch ihren eigenen Mittelwert erklärt, außer gehen wir von einem Fehler  $e$  in unseren Modellvorhersagen aus. Nobody is perfect. In späteren Kapiteln werden wir andere Variablen heranziehen, um die Zielvariable zu erklären. Würden wir z.B. sagen wollen, dass wir  $y$  als Funktion einer Variable  $X$  erklären, so würden wir schreiben (s. Gleichung 7.9):

$$\bar{y} \sim x \quad (7.9)$$

Da wir im Moment aber keine anderen Variablen bemühen, um  $y$  zu erklären, schreibt man mit Gleichung 7.10 auch:

$$\bar{y} \sim 1 \quad (7.10)$$

Diese Schreibweise sieht verwirrend aus. Die 1 soll aber nur zeigen, dass wir keine andere Variable zur Erklärung von  $y$  verwenden, daher steht hier kein Buchstabe, sondern eine einfache 1.<sup>5</sup>

**Beispiel 7.3** (Noten, Mittelwert und Abweichung). Vier Studenten – Anna, Berta, Carl, Dani – haben ihre Statistik-Klausur zurückbekommen (Schluck). Die Noten sehen Sie in Abbildung 7.2, gar nicht so schlecht ausgefallen. Außerdem ist der Mittelwert (gestrichelte horizontale Linie) sowie die Abweichungen der einzelnen Noten vom Mittelwert eingezeichnet. □

Schauen Sie sich die Abweichungsbalken<sup>6</sup> in Abbildung 7.2 einmal genauer an.

Jetzt stellen Sie sich vor, Sie würden die vom Mittelwert nach *oben* ragenden Balkenlängen aneinanderlegen (das sind die gestrichelten). Sehen Sie das vor Ihrem geistigen Auge? Jetzt legen Sie auch noch die Abweichungsbalken, die nach *unten* ragen, aneinander (die mit den durchgezogenen Linien). Wer viel Phantasie hat, erkennt (sieht) jetzt, dass die Gesamtlänge der "Balken nach oben" identisch ist zur Gesamtlänge der nach "unten ragenden Balken", vgl. Abbildung 7.1.

Präziser ausgedrückt und ohne Ihre Phantasie zu strapazieren (Gleichung 7.11):

$$\sum_{i=1}^n (x_i - \bar{x}) = \sum_{i=1}^n x_i - \sum_{i=1}^n \bar{x} = n \cdot \bar{x} - n \cdot \bar{x} = 0 \quad (7.11)$$

---

<sup>5</sup>Der mathematische Hintergrund liegt in der Art, wie man Matrizen multipliziert.

<sup>6</sup>Residuen, Fehler; häufig mit  $e$  wie *error* bezeichnet

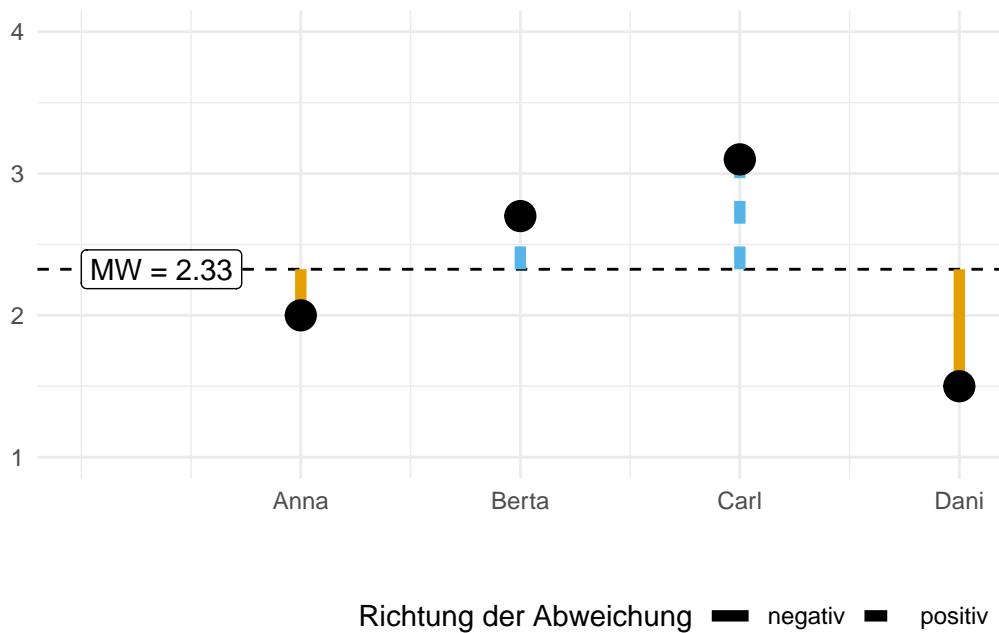


Abbildung 7.2.: Der Mittelwert als horizontale (gestrichelte) Linie. Die vertikalen Linien zeigen die Abweichungen der einzelnen Werte zum Mittelwert. Die Abweichungen summieren sich zu Null auf.

#### i Hinweis

Die Summe der Abweichungen vom Mittelwert ist Null.

**Übungsaufgabe 7.3.** Was schätzen Sie, wie hoch das “mittlere” Vermögen der Haushalte in Deutschland in etwa ist?<sup>7</sup>. □

#### 7.2.3. Auswahl

- a) 50.000 Euro
- b) 100.000 Euro
- c) 150.000 Euro
- d) 200.000 Euro
- e) 250.000 Euro

#### 7.2.4. Antwort

- a) **50.000 Euro**, ca. 60.000 Euro (laut der o.g. Quelle)
- b) 100.000 Euro
- c) 150.000 Euro

<sup>7</sup>Quelle: SI, <https://www.wsi.de/en/how-is-wealth-distributed-in-germany-14401.html>, Abruf 2023-04-19

- d) 200.000 Euro
- e) 250.000 Euro

**Beispiel 7.4** (Der reichste Mensch der Welt in Ihrem Hörsaal). Kommt der wertvollste Fußballspieler der Welt in Ihren Hörsaal, sagen wir, es ist Kylian Mbappé<sup>8</sup>. Sein Jahreseinkommen (2023) liegt bei ca. 120 Millionen Euro<sup>9</sup>.

💡 Hey Leute, wie geht's denn so! Wie viel Kohle verdient ihr eigentlich so?

💡 Äh, wir studieren und verdienen fast nix!

Die 100 Studis im Hörsaal schauen verdattert aus der Wäsche: Was ist das für eine komische Frage!? Aber zumindest verteilt der Fußballspieler Autogramme.

**Übungsaufgabe 7.4** (Mittleres Einkommen im Hörsaal, mit Kylian Mbappé). Schätzen Sie – im Kopf – das mittlere Vermögen im Hörsaal, gehen Sie davon aus, dass alle der 100 Studenten jeweils 1000 Euro im Jahr verdienen. □

In R kann man das mittlere Einkommen (präziser: das arithmetische Mittel des Einkommens) wie folgt berechnen, s. Listing 7.1.<sup>10</sup>

---

**Listing 7.1** Wir simulieren Einkommen von 100 Studis plus Mbappé.

---

```
set.seed(42)  # Zufallszahlen festlegen, hier nicht so wichtig
einkommen_studis <- rep(x = 1000, times = 100)  # "rep" wie
  ↵ "repeat": wiederhole 1000 USD 100 Mal
einkommen <- c(einkommen_studis, 120*1e6)  # 100 Studis mit
  ↵ 1000, 1 Mbappé mit 120 Mio
einkommen_mw <- mean(einkommen)
einkommen_mw
## [1] 1189109
```

---

**i Hinweis**

1 Million hat 6 Nuller hinter der führenden Eins: 1000000. In Taschenrechner- oder Computerschreibweise: 1 Mio = 1e6, das 1e6 ist zu lesen als “1 Mal 10 hoch 6, also mit 6 im Exponenten”.

Der Mittelwert im Hörsaal beträgt also 1,189,109 Euro. Ist das ein gutes Modell für das “typische” Vermögen im Hörsaal?

<sup>8</sup>Quelle: <https://www.transfermarkt.de/spieler-statistik/wertvollstespieler/marktwertetop>, Abruf 2023-03-19

<sup>9</sup>Quelle: <https://www.einkommenmagazin.de/kylian-mbappe-einkommen/>, Abruf 2023-03-19

<sup>10</sup>Die Details der Syntax, z.B. der Befehl `rep()`, sind von geringer Bedeutung.

### 7.2.5. Der Mittelwert als lineares Modell

Man kann den Mittelwert als Gerade einzeichnen, s. Abbildung 7.3, bzw. als Gerade begreifen. Insofern kann man vom Mittelwert auch als *lineares Modell* sprechen.

**Definition 7.2** (Lineares Modell). Ein lineares Modell verwendet eine Gerade als Modell der Daten. Es erklärt die Daten anhand einer Geraden.  $\square$

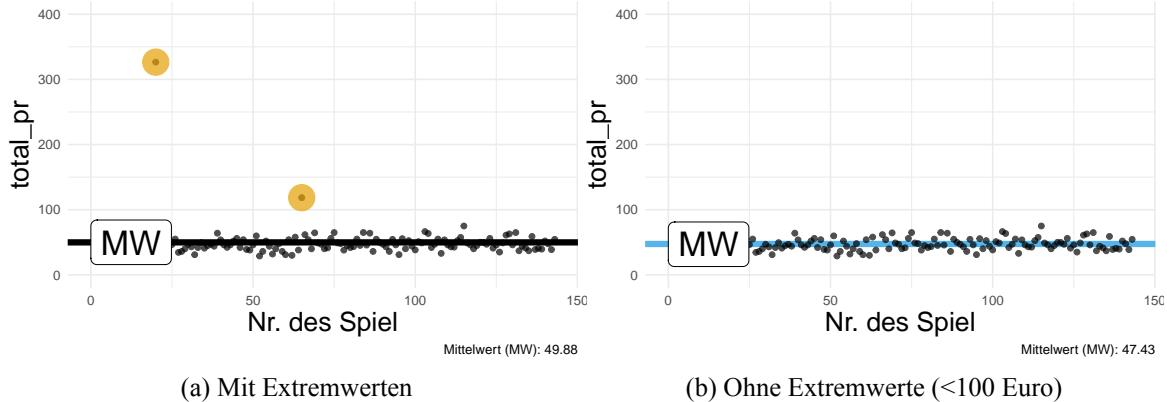


Abbildung 7.3.: Der mittlere Preis von Mariokart-Spielen als horizontale Gerade eingezeichnet

Abbildung 7.3 zeigt den Mittelwert des Verkaufspreises der Mariokart-Spiele (total\_pr), einmal mit Extremwerten (a) bzw. einmal ohne Extremwerte (b).

**Definition 7.3** (Extremwert). Ein Extremwert (engl. *outlier*) ist eine Beobachtung, dessen Wert deutlich vom Großteil der anderen Beobachtungen im Datensatz abweicht, z.B. viel größer ist.  $\square$

Berechnen wir mal den Mittelwert von einkommen mit R (vgl. Listing 7.1):

```
lm(einkommen ~ 1) # lm wie "lineares Modell" oder engl.
#<- "linear modell"
## 
## Call:
## lm(formula = einkommen ~ 1)
## 
## Coefficients:
## (Intercept)
##       1189109
```

Der Befehl gibt als *Koeffizient* einen Wert zurück und zwar den Mittelwert von einkommen, Listing 7.1. Dieser Wert wird als Achsenabschnitt (engl. *intercept*) bezeichnet, das wird verständlich, wenn man z.B. in Abbildung 7.3 sieht, dass die Gerade (des Mittelwerts) genau an diesem Punkt die Y-Achse schneidet.

Die Syntax des Befehls `lm()` sieht etwas merkwürdig aus. Ignorieren Sie das fürs Erste, wir besprechen das später (Kapitel 10) ausführlich. `lm` steht übrigens für “lineares Modell”.

### 7.3. Median als Modell

💡 Hey, der Mittelwert ist doch Quatsch! Das ist gar kein typischer Wert für die Menschen im Hörsaal. Weder für den Mbappé, noch für uns Studis!

💡 Ja, da habt ihr Recht.

⚽ Die Welt ist schon ungerecht!

#### ! Wichtig

Bei (sehr) schießen Verteilungen (s. Abbildung 7.4) ist der Mittelwert (sehr) wenig aussagekräftig, da er nicht mehr “typische” Werte für die Merkmalsträger beschreibt.

Abbildung 7.4 stellt die Verteilung des Einkommens einer mit “normal” skalierten Achse und einmal mit logarithmischer X-Achse.<sup>11</sup> Die logarithmische X-Achse stellt den Unterschied von Mittelwert (MW) und Median deutlicher heraus als die normale (additive) Achse.

Der Mittelwert ist Hörsaal ist nicht typisch für die Menschen im Hörsaal: Weder für Mbappé, noch für die Studis. Genau genommen ist der Mittelwert in diesem Fall ziemlich nutzlos.

#### ! Wichtig

Der Mittelwert ist empfänglich für Extremwerte: Gibt es einen extremen Wert in einer Datenreihe, so spiegelt der Mittelwert stark diesen Wert wieder und weniger die Mehrheit der gemäßigten Werte. Man sagt, der Mittelwert ist nicht *robust* (gegenüber Extremwerten).

**Beispiel 7.5** (Das Median-Einkommen einiger Studentinnen). Fünf Studentinnen tauschen sich über ihr Einkommen aus, s. Abbildung 7.5, links. Es handelt sich um eine schiefe Verteilung.

Wir könnten jetzt behaupten, dass Carla das typische Einkommen (für diese Datenreihe) aufweist, da es genauso viele Studentinnen gibt, die mehr verdienen, wie solche, die weniger verdienen. □

**Definition 7.4** (Median). Merkmalsausprägung, die bei (aufsteigend) sortierten Beobachtungen in der Mitte liegt. □

**Übungsaufgabe 7.5** (Alle mal aufstehen). Auf Geheiß der Lehrkraft stehen jetzt alle Studis bitte auf und sortieren sich der Größe nach im Raum, schön in einer Reihe aufgestellt. Die Körpergröße der Person in der Mitte der Reihe, zu der also gleich viele Personen zu links wie zu rechts stehen, das ist der Medien dieser Datenreihe, vgl. Abbildung 7.6. □

Der Median ist *robust* (gegenüber) Extremwerten: Fügt man Extremwerte zu einer Verteilung hinzu, ändert sich der Median zumeist (deutlich) weniger als der Mittelwert.

Abbildung 7.6 stellt den Median schematisch dar.

<sup>11</sup>Zur Erinnerung:  $4.0+e07$  bedeutet  $4 \cdot 10^{07} = 40000000$ , eine 4 gefolgt von 7 Nullen.

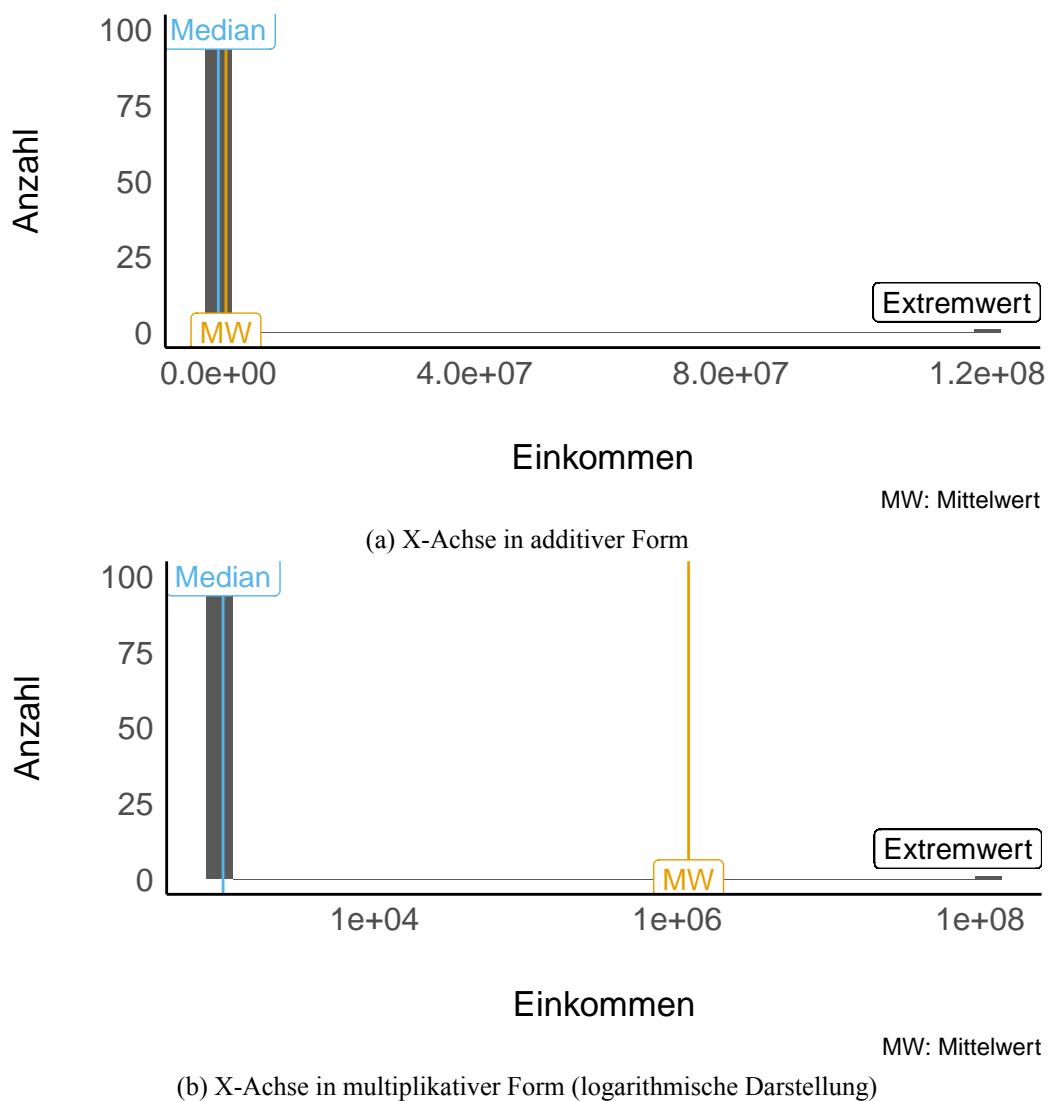
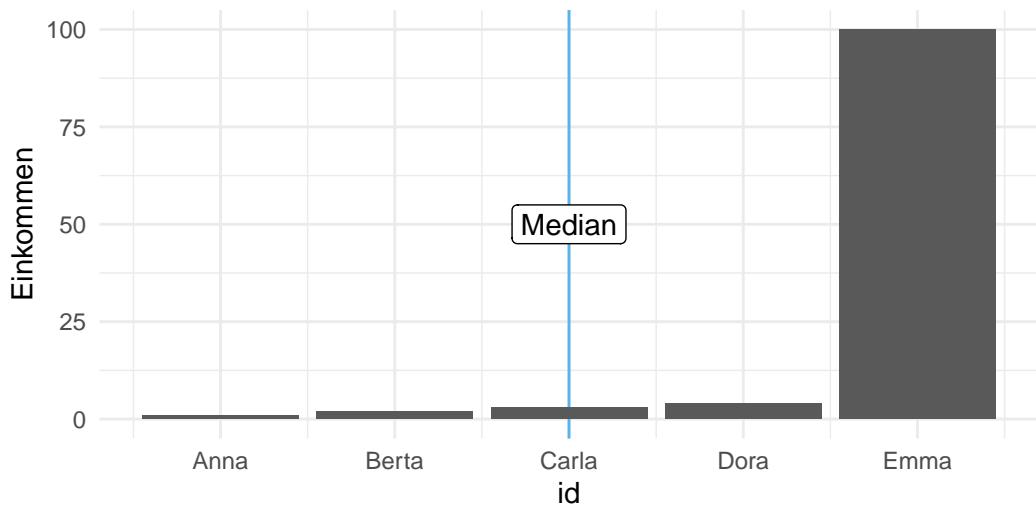
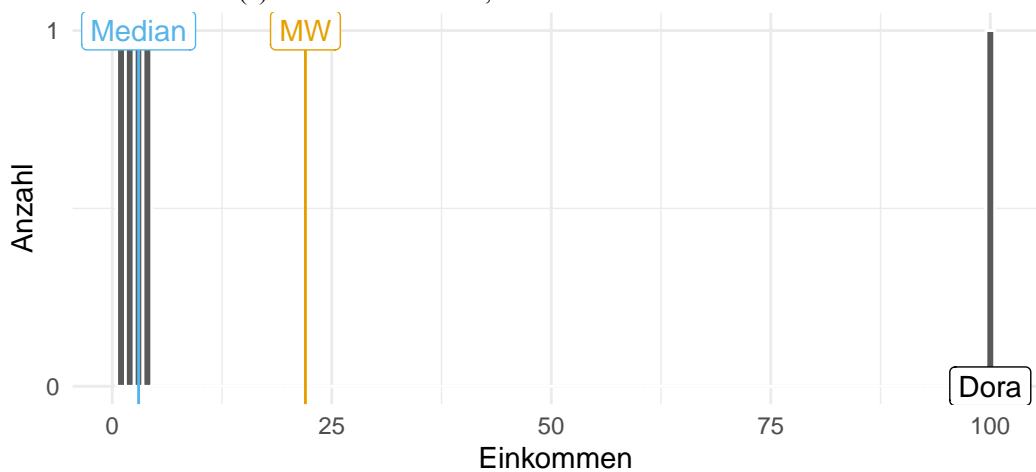


Abbildung 7.4.: Die Einkommensverteilung im Hörsaal



(a) ID auf der X-Achse, Einkommen auf der Y-Achse



(b) Einkommen auf der X-Achse, Häufigkeit auf der Y-Achse

Abbildung 7.5.: Das Median-Einkommen einiger Studentinnen sowie der Mittelwert (MW) ihres Einkommens

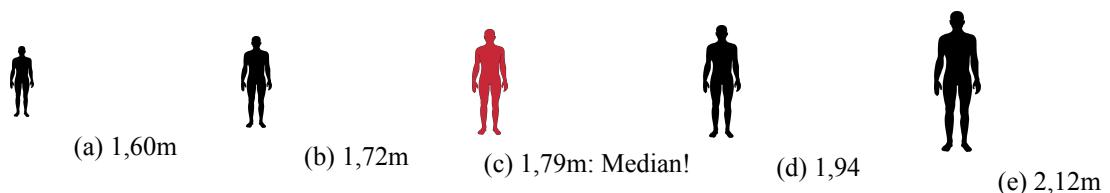


Abbildung 7.6.: Der Median als der Wert des “mittleren” Objekts, wenn die Objekte aufsteigend sortiert sind. Es gibt genauso viele Objekte mit kleinerem Wert als der Median wie Objekte mit größerem Wert als der Median.

Bei geradem  $n$  werden die beiden mittleren Werte betrachtet und das arithmetische Mittel aus diesen beiden Werten gebildet.

**Beispiel 7.6.** Bei der Messreihe 1, 2, 3, 4, 5, 6, 8, 9 beträgt der Median 4.5.  $\square$

**Übungsaufgabe 7.6** (Emma wird reich). Durch ein geniales Patent wird Emma steinreich. Ihr Einkommen erhöht sich um das Hundertfache. Wie verändert sich der Median?<sup>12</sup>  $\square$

**Übungsaufgabe 7.7** (Wer ist mehr “mittel”? Median oder Mittelwert?).

💡 Das arithmetische Mittel sollte Mittelwert heißen, weil es die Mitte von zwei Messwerten widerspiegelt, also z.B. von 1 und 10 ist die Mitte 5,5 – also genau beim Mittelwert!

🤔 Moment! Der Median und nur der Median zeigt den mittleren Messwert! Links und rechts sind gleich viele Messwerte, wenn man die Werte der Größe nach sortiert. Also liegt der Median genau in der Mitte!

Nehmen Sie Stellung zu dieser Diskussion!  $\square$

**Beispiel 7.7** (Ein “mittlerer” Preis für MarioKart). Der Mittelwert (das arithmetische Mittel) und der Median für das Start-Gebot (`start_pr`) von MarioKart-Spielen sind nicht gleich, der Mittelwert ist höher als der Median.

```
mariokart <- read.csv(mariokart_path) # Der Pfad steht zu
  ↵ Beginn des Kapitels

mariokart %>%
  summarise(price_mw = mean(start_pr),
            price_md = median(start_pr))
```

| price_mw | price_md |
|----------|----------|
| 8.777203 | 1        |

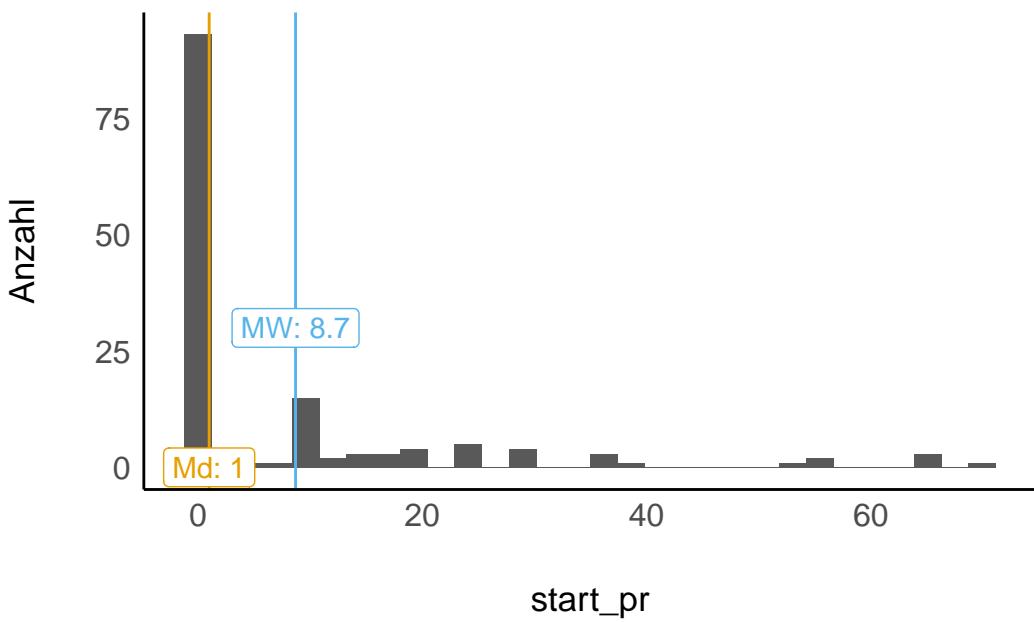
Wie man sieht, ist der Mittelwert größer als der Median, s. Abbildung 7.7.

**i Hinweis**

Kaffen Mittelwert und Median auseinander, so liegt eine schiefe Verteilung vor. Ist der Mittelwert größer als der Median, so nennt man die Verteilung rechtsschief. Bei schießen Verteilungen ist der Median dem Mittelwert als Modell für den “typischen Wert” vorzuziehen.

**Übungsaufgabe 7.8** (MarioKart ohne Extremwerte). Im Datensatz `mariokart` gibt es einige wenige Spiele, die für einen vergleichsweise hohen Preis verkauft wurden. Diese Extremwerte verzerrten den mittleren Verkaufspreis möglicherweise über die Gebühr.  $\square$

<sup>12</sup>Er bleibt gleich, verändert sich also nicht: Der Median ist *robust*, er verändert sich nicht oder kaum, wenn Extremwerte vorliegen.



### 7.3.1. Aufgabe

Entfernen Sie diese Werte und berechnen Sie dann Mittelwert und Median erneut. Vergleichen Sie die Ergebnisse.

### 7.3.2. Lösung

```
mariokart2 <-
mariokart %>%
  filter(total_pr < 100)

# ohne Extremwerte:
mariokart2 |>
  summarise(total_pr_mittelwert = mean(total_pr),
            total_pr_median = median(total_pr))
```

| total_pr_mittelwert | total_pr_median |
|---------------------|-----------------|
| 47.43191            | 46.03           |

```
# mit Extremwerten:
mariokart |>
  summarise(total_pr_mittelwert = mean(total_pr),
            total_pr_median = median(total_pr))
```

| total_pr_mittelwert | total_pr_median |
|---------------------|-----------------|
| 49.88049            | 46.5            |

**Übungsaufgabe 7.9.** Was schätzen Sie, wie hoch das *medianen* Vermögen des Haushalte in Deutschland in etwa ist (Stand 2016)?<sup>13)</sup>

- a) 50.000 Euro
- b) 100.000 Euro
- c) 150.000 Euro
- d) 200.000 Euro
- e) 250.000 Euro

**Übungsaufgabe 7.10.** Was schätzen Sie, wie groß der *Unterschied* zwischen medianem und mittlerem (arithm. Mittel) des Jahreseinkommen deutscher Haushalte ungefähr ist?<sup>14)</sup>

- a) 1.000 Euro
- b) 2.000 Euro
- c) 3.000 Euro
- d) 4.000 Euro
- e) 5.000 Euro

## 7.4. Quantile

Der Median teilt eine Verteilung in eine untere und ein obere Hälfte. Er markiert sozusagen eine “50-Prozent-Marke” (der aufsteigend sortierten Beobachtungen). Betrachten wir einmal nur alle Spiele, die für weniger als 100 Euro verkauft wurden (`total_pr`, finales Verkaufsgebot), s. Abbildung 7.8. 50% aller Spiele wurden für weniger als ca. 46 Euro verkauft; 50% aller Spiele für mehr als 46 Euro. Der Median beträgt als 46 Euro.

Jetzt könnten wir nur die günstigere Hälfte betrachten und wieder nach dem Median fragen (d.h. `total_pr < 46`). Dieser “Median der günstigeren Hälfte” grenzt damit das insgesamt günstigste Viertel vom Rest der Verkaufsgebote ab. In unserem Datensatz liegt dieser Wert bei ca. 41 Euro. Entsprechend kann man nach dem Wert fragen, der das oberste Viertel vom Rest der Verkaufsgebote abtrennt. Dieser Wert liegt bei ca. 54 Euro.

<sup>13</sup>Quelle: WSI, <https://www.wsi.de/en/how-is-wealth-distributed-in-germany-14401.htm>, Abruf 2023-04-19. Die Antwort lautet: ca. 60 Tsd Euro laut der angegebenen Quelle.

<sup>14</sup>Quelle: Wikipedia, Abruf 2023-04-19, der Unterschied beträgt knapp 3000 Euro laut der Quelle.

**Definition 7.5** (Quartile). Sortiert man die Daten aufsteigend, so nennt man den Wert, der das Viertel mit den kleinsten Wert vom Rest der Daten trennt das *erste Quartil* ( $Q_1$ , 25%). Den Median nennt man das *zweite Quartil* ( $Q_2$ , 50%). Entsprechend heißt der Wert, der die drei Viertel kleinsten Werte vom oberen Viertel abtrennt, das *dritte Quartil* ( $Q_3$ , 75%).  $\square$

**Beispiel 7.8** (Quartile des Verkaufsgebot). Abbildung 7.8 zeigt die Quartile für das Verkaufsgebot.  $\square$

Jetzt könnte man sagen, hey, warum nur in 25%-Stücke die Verteilung aufteilen? Warum nicht in 10%-Schritten?

**Definition 7.6** (Dezile). Die neun Quantile  $p = 0.1, 0.2, \dots, 1$ , die die Verteilung in 10 gleiche Teile unterteilen, nennt man Dezile.  $\square$

Oder vielleicht in 1%-Schritten oder in sonstigen Schnitten? Wo die Quartile in 25%-Schritten aufteilen, teilt in *Quantil* in  $p$ -Prozent-Schritten auf.

**Definition 7.7** (Quantile). Ein  $p$ -Quantil ist der Wert, der von  $p$  Prozent der Werte nicht überschritten wird.  $\square$

### i Hinweis

Ein Quantil ist ein Oberbegriff für Quartile, Dezile, etc.  $\square$

Abbildung 7.8 zeigt das 1. ( $Q_1$ ), das 2. (Median) und das 3. Quartil für den Datensatz mariokart2.

*Quantile* kann man in R mit dem Befehl `quantile()` berechnen:

```
mario_quantile <-  
mariokart %>%  
  filter(total_pr < 100) %>%  
  summarise(q25 = quantile(total_pr, .25),  
            q50 = quantile(total_pr, .50),  
            q75 = quantile(total_pr, .75))
```

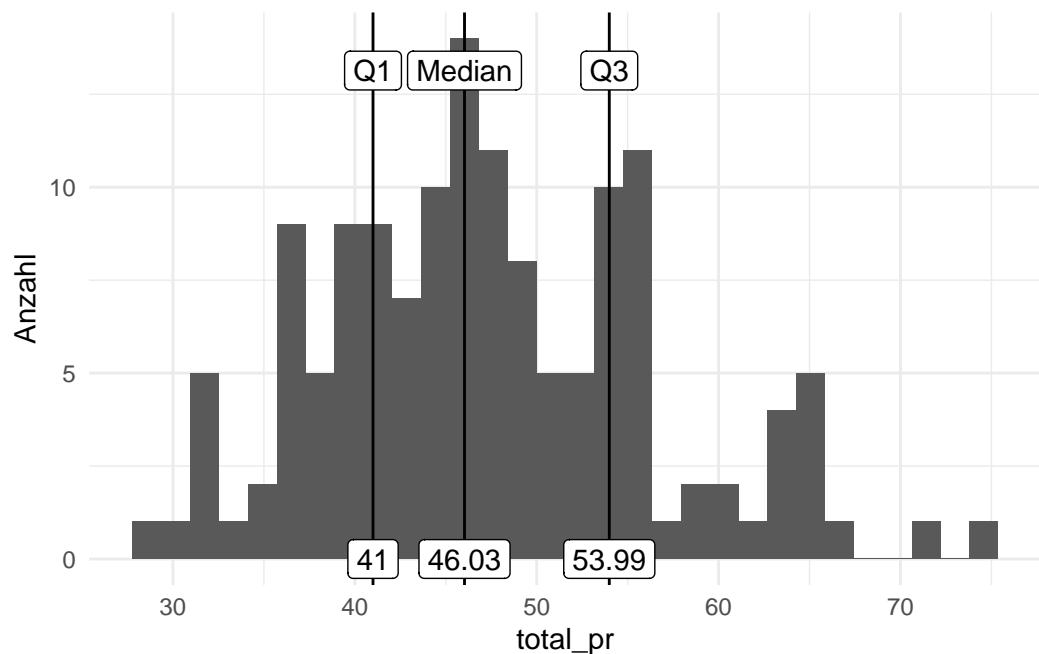
Abbildung 7.9 visualisiert verschiedene Quantile. Man beachte, dass alle Regionen gleichgroße Flächen (d.h. Wahrscheinlichkeitsmassen) aufweisen.

## 7.5. Lagemaße

💡 Was ist der Oberbegriff für Median, Mittelwert und so weiter?

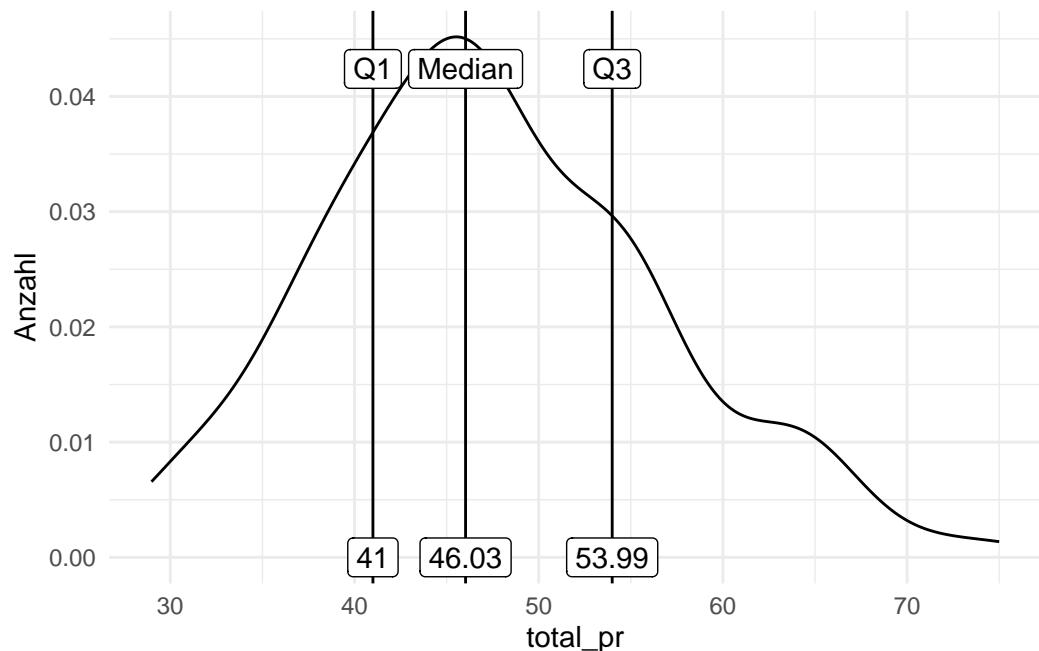
💡 Gute Frage! Wie würden Sie ihn nennen?

#### 7.4.1. Histogramm



(a) Q1, Q2 und Q3 für das Schlussgebot (nur Spiele für weniger als 100 Euro)

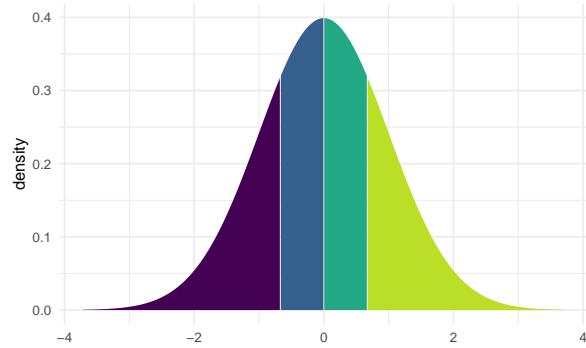
#### 7.4.2. Dichtediagramm



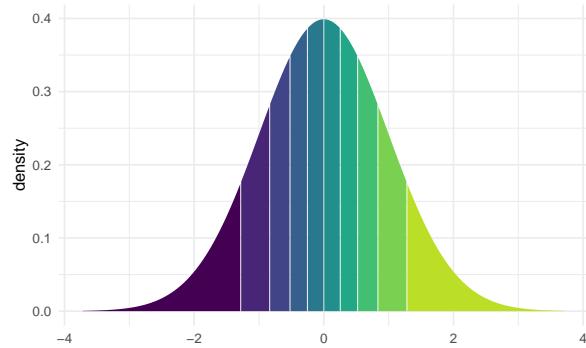
(b) Q1, Q2 und Q3 für das Schlussgebot (nur Spiele für weniger als 100 Euro)

Abbildung 7.8.: Verschiedene Arten von Quantilen.

#### 7.4.3. 25%-Schritte: Quartile



#### 7.4.4. 10%-Schritte: Dezile



#### 7.4.5. 1%-Schritte: Perzentile

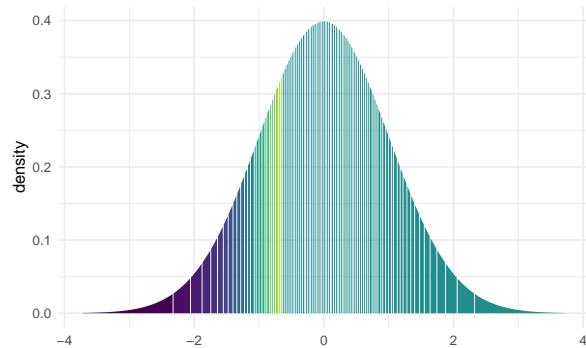


Abbildung 7.9.: Verschiedene Quantile visualisiert. In jedem Diagramm sind die Regionen gleich groß, beinhalten also (ungefähr) die gleiche Anzahl von Beobachtungen.

**Definition 7.8** (Lagemaß). Ein *Lagemaß* (synonym: Maß der zentralen Tendenz) für eine Verteilung gibt einen Vorschlag, welchen Wert der Verteilung wir als typisch, normal, erwartbar, repräsentativ oder “mittel” ansehen sollten.□

**Beispiel 7.9.** Gebräuchliche Lagemaße sind:

- Mittelwert (arithmetisches Mittel)
- Median
- Quartile
- Quantile
- Minimum (kleinster Wert)
- Maximum (größter Wert)
- Modus (häufigster Wert) □

Berechnen wir Lagemaße für den Mariokart-Datensatz, s. Listing 7.2.<sup>15</sup>

---

**Listing 7.2** Syntax zur Berechnung von Lagemaßen

---

```
mariokart_lagemaße_total_pr <-
  mariokart %>%
    summarise(mw = mean(total_pr),
              md = median(total_pr),
              q1 = quantile(total_pr, .25),
              q2 = quantile(total_pr, .5),
              q3 = quantile(total_pr, .75),
              min = min(total_pr),
              max = max(total_pr))
mariokart_lagemaße_total_pr
```

---

| mw       | md   | q1     | q2   | q3    | min   | max    |
|----------|------|--------|------|-------|-------|--------|
| 49.88049 | 46.5 | 41.175 | 46.5 | 53.99 | 28.98 | 326.51 |

---

### 7.5.1. Gruppierte Lagemaße

Häufig möchte man Statistiken wie Lagemaße für mehrere Teilgruppen – z.B. Mittlere Körpergröße von Frauen vs. Mittlere Körpergröße von Männer – berechnen und dann vergleichen. Die zugrundeliegende stehende *Forschungsfrage* könnte lauten:

Unterscheidet sich die mittlere Körpergröße von Frauen und Männern?

---

<sup>15</sup>Es ist übrigens egal, wie sie die Variablen benennen, die Sie berechnen: mw oder mittelwert oder mean oder mein\_krasser\_variablenname – alles okay!

Oder vielleicht:

Hat das Geschlecht einen Einfluss auf die Körpergröße?

Anders ausgedrückt:

Körpergröße  $y$  ist eine Funktion des Geschlechts  $G$ .

Die *Modellformel* könnte also lauten:

$$y \sim G$$

Gruppierte Lagemaße lassen sich in R z.B. so berechnen, s. Listing 7.3, also ähnlich wie in Listing 7.2.

---

#### Listing 7.3 Gruppierte Lagemaße

---

```
mariokart_lagemaße_gruppiert <-
  mariokart %>%
  group_by(wheels) %>% # neue Zeile, der Rest ist gleich!
  summarise(mw = mean(total_pr))

mariokart_lagemaße_gruppiert
```

| wheels | mw       |
|--------|----------|
| 0      | 41.05973 |
| 1      | 44.16885 |
| 2      | 61.02745 |
| 3      | 69.75000 |
| 4      | 65.02000 |

---

Abbildung 7.10 zeigt ein Beispiel für ungruppierte (links) bzw. gruppierte (rechts) Mittelwerte; vgl. Abbildung 7.3. Wie man in dem Diagramm sieht, kann das *Residuum kleiner* werden bei einer Gruppierung (im Vergleich zu einem ungruppierten, “globalen” Mittelwert): Innerhalb der Gruppe ohne Lenkräder und innerhalb der Gruppe mit 2 Lenkräder sind die Abweichungen zu ihrem Gruppen-Mittelwert relativ gering – im Vergleich zu den Abweichungen der Preise zum ungruppierten Mittelwert.

**Definition 7.9** (Punktmodell). Ein Modell, welches für alle Beobachtungen ein und denselben Wert annimmt (vorhersagt), heißt ein *Punktmodell*. Anders gesagt fasst ein Punktmodell eine Wertereihe (häufig ist das eine Tabellenspalte) zu einer einzelnen Zahl zusammen, einem ‘Punkt’ in diesem Sinne, s. Gleichung 7.12.□

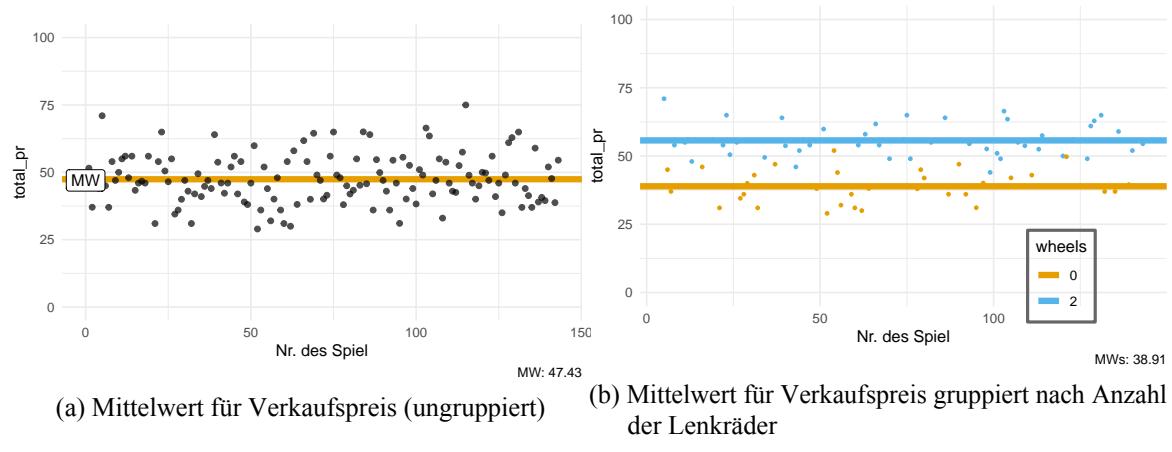


Abbildung 7.10.: Der mittlere Preis von Mariokart-Spielen als horizontale Gerade eingezeichnet



Mittelwert, Median und Quartile sind Beispiele für Punktmodelle: Sie fassen eine Verteilung zu einem einzelnen Wert zusammen und geben uns ein “Bild” der Daten, machen Sie uns verständlich - sie sind uns ein Modell.

## 7.6. Wie man mit Statistik lügt

Mit Statistik kann man vortrefflich lügen, heißt es. Woran liegt das? Der Grund ist, dass die Statistik Freiheitsgrade lässt: Es gibt nicht nur einen richtigen Weg, um eine statistische Analyse durchzuführen. Viele Wege führen nach Rom (aber nicht alle). Um Manipulationsversuche abzuwehren oder einfache Fehler und Unschärfen ohne böse Abwehr aufzudecken, gibt es ein probates Gegenmittel: *Transparenz*.

Stellen Sie hohe Anforderung an die Transparenz einer statistischen Analyse. Nur durch Nachprüfbarkeit können Sie sich von der Stichhaltigkeit der Ergebnisse und deren Interpretation überzeugen.

Hier ist eine (nicht abschließende!) Checkliste, was Sie nachprüfen sollten, um die Belastbarkeit einer Analyse sicherzustellen (**wicherts\_degrees\_2016-1?**):

| Nr | Check   |
|----|---|
| 1  | Wurde die Art und die Zeitdauer der Datenerhebung vorab festgelegt und berichtet?       |
| 2  | Wurden ausreichend Daten gesammelt (z.B. mind. 20 Beobachtungen pro Gruppe)?            |
| 3  | Wurden alle untersuchten Variablen berichtet?   |
| 4  | Wurden alle durchgeföhrten Interventionen berichtet?                                    |
| 5  | Wurden Daten aus der Analyse entfernt? Wenn ja, gibt es eine (stichhaltige) Begründung? |

## 7.7. Vertiefung

**Beispiel 7.10** (Survival-Tipp). Eine Studentin aus dem dem Bachelorstudiengang “Angewandte Medien- und Wirtschaftspsychologie” mit Schwerpunkt *Data Science* berichtet ihre “Survival-Tipps” für Statistik.

1. Wenn man mal nicht weiterkommt, hilft es auch mal ein paar Tage Abstand von R und Statistik zu nehmen.
2. Es hilft, sich während des Semesters neue Begriffe und ihre Erklärung zusammenschreiben.
3. Gut ist auch, sich mit KommilitonInnen auszutauschen oder in höheren Semestern nach Tipps fragen.□

💡 Irgendwie kann ich mir R-Code so schlecht merken.

💻 Frag doch mal ChatGPT, oder einen anderen Chatbot, da bekommt man auch R-Code ausgegeben.

**Übungsaufgabe 7.11** (Übungsfragen vom Chat-Bot). Fragen Sie einen Chat-Bot wie ChatGPT nach Übungsaufgaben.

Sie können sich an folgenden Prompt orientieren. Empfehlenswert ist mit verschiedenen Prompts zu experimentieren.

💡 Ich bin ein Student in einem Bachelor-Studiengang für Psychologie. Gerade bereite ich mich auf die Klausur im Fach “Grundlagen der Statistik” vor. Bitte schreibe mir Aufgaben, die mir helfen, mich auf die Prüfung vorzubereiten. Die Fragen sollten folgende Themen beinhalten: Maße der zentralen Tendenz, Grundlagen von R, Skalenniveau (z.B. Nominalskala vs. Intervallskala), Verteilungsformen, Normalverteilungen, z-Werte. Bitte schreibe die Aufgabe im Stil von Richtig-Falsch-Aufgaben. Schreibe ca. 10 Aufgaben.

□

## 7.8. Aufgaben

Ein Teil der folgenden Aufgaben kann Stoff beinhalten, den Sie noch nicht kennen, aber später kennenlernen. Ignorieren Sie daher Aufgaben(teile) mit (noch) unbekannte Stoff.

Die Webseite [datenwerk.netlify.app](https://datenwerk.netlify.app) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

1. [Kennwert-robust](#)
2. [mw-berechnen](#)
3. [mariokart-max2](#)
4. [nasa01](#)
5. [mariokart-mean1](#)
6. [wrangle10](#)

7. [summarise01](#)
8. [mariokart-max1](#)
9. [Schiefe1](#)
10. [mariokart-mean2](#)
11. [summarise03](#)
12. [mariokart-mean4](#)
13. [mariokart-mean3](#)
14. [summarise02](#)

 Tipp

Schauen Sie sich auch mal auf [datenwerk.netlify.app](#) die Aufgaben zu z.B. dem Tag [EDA](#) an. □

**Übungsaufgabe 7.12.** Mittlerweile verfügen Sie die wesentlichen Werkzeuge des Datenjudo. Hier finden Sie einen Überblick an Datensätze, die Sie nach Herzenslust analysieren können.<sup>16</sup> □

## 7.9. Literaturhinweise

Es gibt viele Lehrbücher zu den Grundlagen der Statistik; die Inhalte dieses Kapitels gehören zu den Grundlagen der Statistik. Vielleicht ist es am einfachsten, wenn Sie einfach in Ihrer Bibliothek des Vertrauens nach einem typischen Lehrbuch schauen. Beispiel für Lehrbücher sind Mittag und Schüller (2020) oder Oestreich und Romberg (2014); ein Klassiker ist Bortz und Schuster (2010). Ein Fokus auf R legt ([sauer\\_moderne\\_2019-1?](#)). Wer vor Englisch nicht zurückschreckt, ist mit ([cetinkaya-rundel\\_introduction\\_2021-2?](#)) oder ([poldrack\\_statistical\\_2022-1?](#)) gut beraten. Beide Bücher sind online verfügbar. Tipp: Mit dem Browser einfach auf Deutsch übersetzen.

## 7.10. Literatur

---

<sup>16</sup><https://data-se.netlify.app/2022/02/23/data-sets-for-teaching/>

# 8. Modellgüte

## 8.1. Lernsteuerung

### 8.1.1. Standort im Lernpfad

Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 8.1.2. Lernziele

- Sie kennen gängige Maße der Streuung einer Stichprobe und können diese definieren und mit Beispielen erläutern.
- Sie können gängige Maße der Streuung einer Stichprobe mit R berechnen.
- Sie können die Bedeutung von Streuung für die Güte eines Modells erläutern.

### 8.1.3. Benötigte R-Pakete

In diesem Kapitel benötigen Sie folgende R-Pakete.

```
library(tidyverse)
library(easystats)
library(DataExplorer)
```

### 8.1.4. Benötigte Daten

Listing 8.1 definiert den Pfad zum Datensatz `mariokart` und importiert die zugehörige CSV-Datei in R, so dass wir einen Tibble mit Namen `mariokart` erhalten.

### 8.1.5. Zum Einstieg

**Übungsaufgabe 8.1** (Freiwillige vor!). Für diese kleine Live-Demonstration brauchen wir einige Freiwillige. Die Lehrkraft teilt die Freiwilligen in zwei Gruppen, Gruppe *Gleich-Groß* und Gruppe *Verschieden-Groß*. Erkennen Sie, dass die *Unterschiedlichkeit* der Größe in Gruppe *Gleich-Groß* gering ist, aber in Gruppe *Verschieden-Groß* hoch? □

---

**Listing 8.1** Pfad zum Datensatz ‘mariokart’

---

```
mariokart_path <- paste0(  
  "https://vincentarelbundock.github.io/Rdatasets/",  
  "csv/openintro/mariokart.csv")  
  
mariokart <- read.csv(mariokart_path)
```

---

## 8.2. Warum Sie die Streuung Ihrer Daten kennen sollten

### 8.2.1. Die Schlankheitspille von Prof. Weiss-Ois

Prof. Weiss-Ois hat eine Erfindung gemacht, eine Schlankheitspille.<sup>1</sup>

#### 8.2.1.1. Was er sagt



(a) “Ich habe eine Schlankheitspille entwickelt, die pro Einnahme das Gewicht im Schnitt um 1kg reduziert!”

#### 8.2.1.2. Was er NICHT sagt



(a) “Allerdings streuen die Werte der Gewichtsveränderung um 10kg um den Mittelwert herum.”

Würden Sie die Pille von Prof. I. Ch. Weiss-Ois nehmen?<sup>2</sup>

- a) ja
- b) nein
- c) Nur wenn ich 100 Euro bekomme
- d) Okay, für 1000 Euro□

! Wichtig

Wie sehr die Werte eines Modells streuen, ist eine wichtige Information.□

### 8.2.2. Wie man seine Kuh über den Fluss bringt

Treffen sich zwei Bauern, Fritz Furchenzieher und Karla Kartoffelsack. Fritz will mit seiner Kuh einen Fluss überqueren, nur kann die Kuh nicht schwimmen<sup>3</sup>.

<sup>1</sup>Bildquelle: Icon unter Flaticon licence, Autor: iconixar, <https://www.flaticon.com/free-icons/professor>

<sup>2</sup>Ich auf keinen Fall.

<sup>3</sup>ob es Fritz kann, ist nicht überliefert.

👉 (Fritz): Sag mal, Karla, ist der Fluss tief?

👉 (Karla): Nö, im Schnitt nur einen Meter.

Also führt Fritz seine Kuh durch den Fluss, leider kam die Kuh nicht am anderen Ufer an, im Floß erstickte sie, s. Abbildung 8.3.

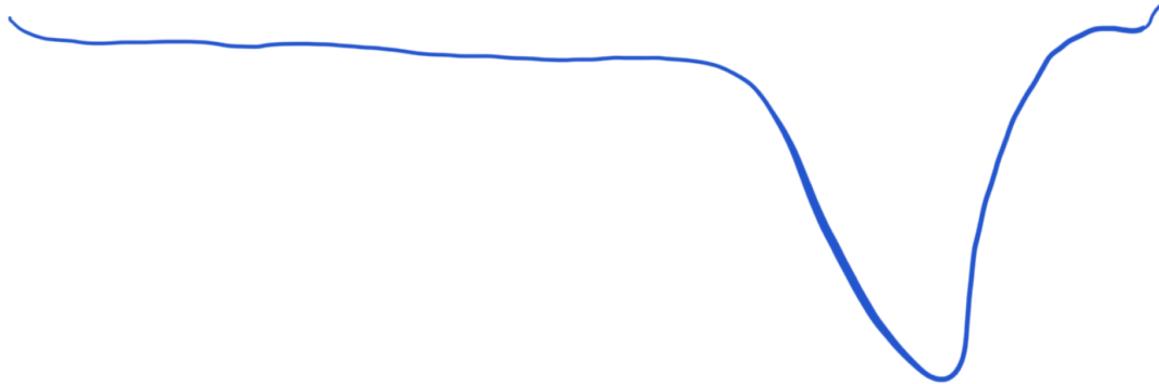


Abbildung 8.3.: Der Fluss ist im Schnitt nur einen Meter tief, trotzdem ist die Kuh erstickt.

👉 (Karla): Übrigens, Lagemaße sagen nicht alles, Fritz.

👉 (Fritz): Läuft die Kuh durch den Fluss, kann sie schwimmen oder 's ist Schluss.

! Wichtig

Die Streuung ihrer Daten zu kennen ist eine wesentliche Information. □

### 8.3. Woran erkennt man ein gutes Modell?

Abbildung 8.4 zeigt ein einfaches Modell (Mittelwert) mit wenig Streuung (links) vs. ein einfaches Modell mit viel Streuung (rechts). Links ist die Streuung der Schlankheitspille *Dicktableitin* und rechts von der Schlankheitspille *Pfundafliptan* abgetragen.

Bei einem Modell mit *wenig* Streuung liegen die tatsächlichen, beobachteten Werte ( $y$ ) nah an den Modellwerten (vorhergesagten Werten,  $\hat{y}$ ); die Abweichungen  $e = y - \hat{y}$  sind also gering (der Modellfehler ist klein). Bei einem Modell mit *viel* Streuung ist der Modellfehler  $e$  (im Vergleich dazu) groß.

**Beispiel 8.1** (Daten zur Schlankheitskur von Prof. Weiss-Ois). In Abbildung 8.4 sind die Daten zu der Gewichtsveränderung nach Einnahme von “Schlankheitspillen” zweier verschiedener Präparate. Wie man sieht unterscheidet sich die typische (vorhergesagte) Gewichtsveränderung zwischen den beiden Präparaten kaum. Die Streuung allerdings schon. Links sieht man die Gewichtsveränderungen nach Einnahme des Präparats “Dickableibtin extra mild” (c) und rechts das Präparat von Prof. Weiss-Ois “Pfundafliptan Forte”. Welches Präparat würden Sie lieber einnehmen? □

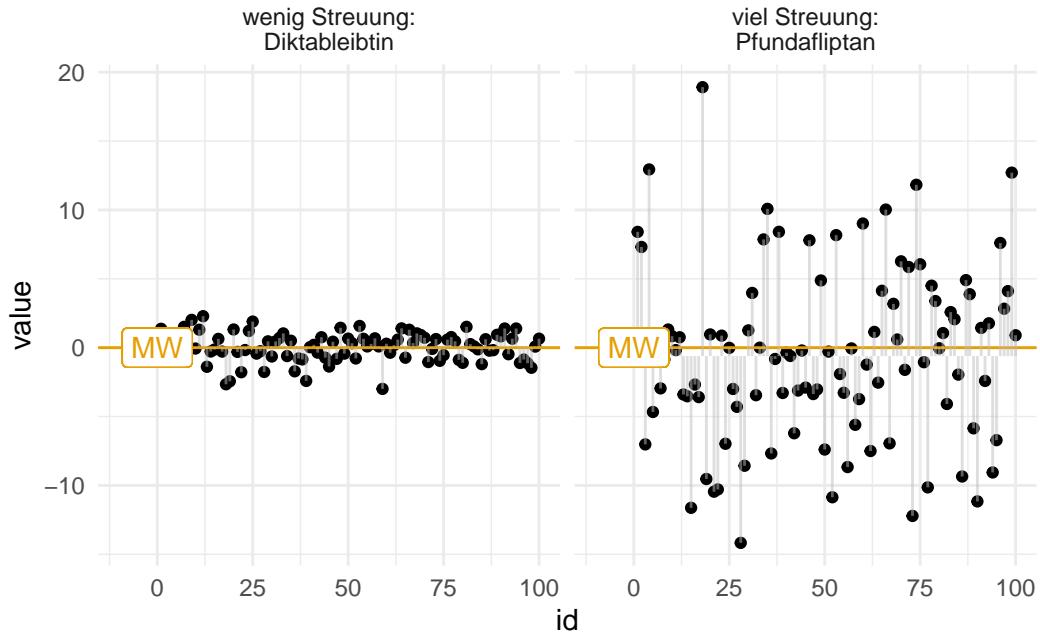


Abbildung 8.4.: Ein Modell mit wenig Streuung (links) vs. ein Modell mit viel Streuung (rechts). Die vertikalen grauen Balken kennzeichnen den (absoluten) Abstand von jeweils einem Datenpunkt zum Mittelwert (horizontale orangefarbene Linie). Je länger die ‘Abstandsbalken’, desto größer die Streuung.

### ! Wichtig

Wir wollen ein präzises Modell, also kurze Fehlerbalken: Das Modell soll die Daten gut erklären, also wenig vom tatsächlichen Wert abweichen. Jedes Modell sollte Informationen über die Präzision des Modellwerts bzw. der Modellwerte (Vorhersagen) angeben. Ein Modell ohne Angaben der Modellgüte, d.h. der Präzision der Schätzung des Modellwerts, ist wenig nütze.□

💡 Ich frage mich, ob man so ein Modell nicht verbessern kann?

💡 Die Frage ist, was wir mit “verbessern” meinen?

💡 Naja, kürzere Fehlerbalken, ist doch klar!

Da die Anzahl der Lenkräder mit dem Verkaufsgebot zusammenhängt, könnte es vielleicht sein, dass wir die Lenkräder-Anzahl da irgendwie nutzen könnten. Das sollten wir ausprobieren.

Abbildung 8.5 zeigt, dass die Fehlerbalken *kürzer* werden, wenn wir ein (sinnvolles) komplexeres Modell finden. Innerhalb jeder der beiden Gruppen (mit 2 Lenkräder vs. mit 0 Lenkräder) sind die Fehlerbalken jeweils im Durchschnitt kürzer (rechtes Teildiagramm) als im Modell ohne Gruppierung (linkes Teildiagramm).<sup>4</sup>

<sup>4</sup>Aus Gründen der Übersichtlichkeit wurden nur Autos mit Verkaufsgebot von weniger als 100 Euros berücksichtigt und nur Spiele mit 0 oder mit 2 Lenkräder.

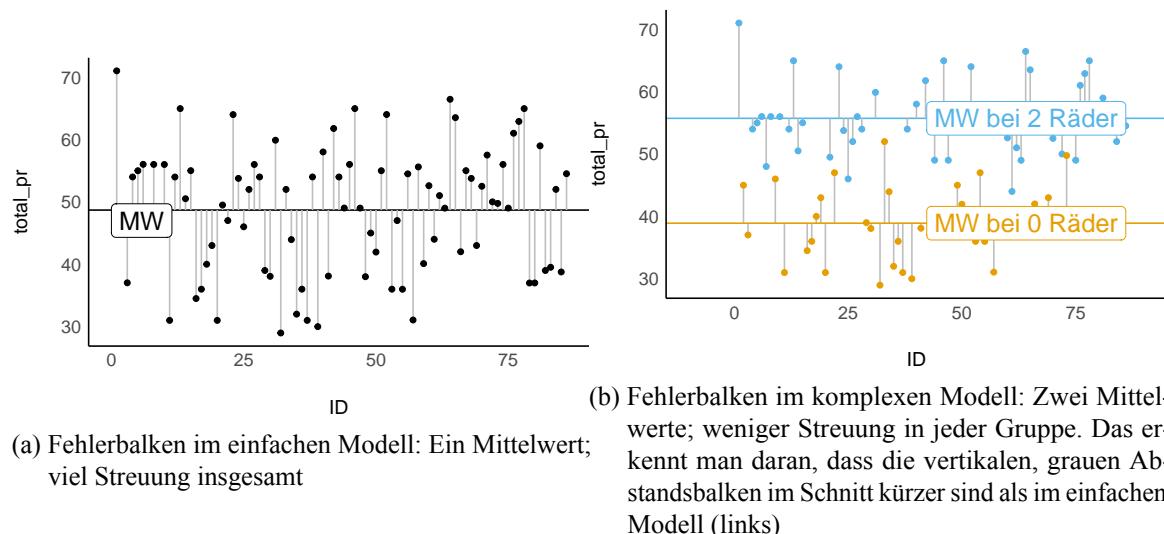


Abbildung 8.5.: Fehlerbalken in einem einfachen und komplexeren Modell

! Wichtig

Durch sinnvolle, komplexere Modelle sinkt die Fehlerstreuung eines Modells. □

## 8.4. Streuungsmaße

**Definition 8.1** (Streuungsmaße). Ein Streuungsmaß quantifiziert die Variabilität (Unterschiedlichkeit, Streuung) eines Merkmals. □

**Definition 8.2.** Ein einfaches Streuungsmaß ist der *Range*  $R$ , definiert als Abstand von größtem und kleinsten Wert eines Merkmals  $X$ :  $R = X_{\max} - X_{\min}$ . □

**Beispiel 8.2.** Angenommen, wir haben einen Datensatz zum Merkmal “Alter” mit den Werten 1, 23, 42, 100. Dann beträgt der Range:  $R = 100 - 1 = 99$ . Das bedeutet, dass die Werte des Merkmals über 99 Einheiten (Jahre in diesem Fall) verteilt sind. □

Dieses Merkmals ist aber nicht robust (gegenüber Extremwerten) und sollte daher nur mit Einschränkung verwendet werden.

### 8.4.1. Der mittlere Abweichungsbalken

💡 Wir müssen jetzt mal präziser werden! Wie können wir die Streuung berechnen?

💡 Gute Frage! Am einfachsten ist es, wenn wir die mittlere Länge eines Abweichungsbalkens ausrechnen.

Legen wir (gedanklich) alle Abweichungsbalken  $e$  aneinander und teilen durch die Anzahl  $n$  der Balken, so erhalten wir den “mittleren Abweichungsbalken”, den wir mit  $\bar{e}$  bezeichnen könnten. Diesen Kennwert bezeichnet man als *Mean Absolute Error* (MAE) bzw. als *Mittlere Absolutabweichung* (MAA). Er ist so definiert, s. Gleichung 8.1.

$$\text{MAE} = \frac{\sum_{i=1}^n |y_i - \bar{y}|}{n} = \frac{\sum_{i=1}^n |e_i|}{n}. \quad (8.1)$$

**Definition 8.3** (Mittlere Absolutabweichung). Die Mittlere Absolutabweichung (MAA, MAE) ist definiert als die Summe der Absolutwerte der Differenzen eines Messwerts zum Mittelwert, geteilt durch die Anzahl der Messwerte.<sup>5</sup> □

**Beispiel 8.3.** Abbildung 8.6 visualisiert ein einfaches Beispiel zum MAE. Rechnen wir den MAE für das Beispiel von Abbildung 8.6 aus:

$$MAE = \frac{1+|-3|+1+1}{4} = 6/4 = 1.5$$

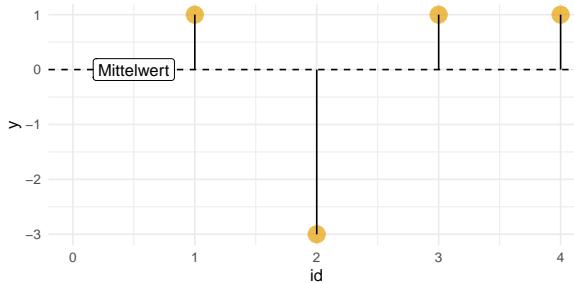


Abbildung 8.6.: Abweichungsbalken und der MAE

Natürlich können wir R auch die Rechenarbeit überlassen.

⌚ Loving it!!

Schauen Sie: Den Mittelwert (s. Abbildung 8.6) kann man doch mit Fug und Recht als ein *lineares Modell*, eine Gerade, betrachten, oder nicht? Schließlich erklären wir  $y$  anhand einer Gerade (die parallel zur X-Achse ist).

In R gibt es einen Befehl für ein *lineares Modell*, er heißt `lm`.

Die Syntax von `lm()` lautet:

`lm(y ~ 1, data = meine_daten)`.

In Worten:

Hey R, berechne mit ein lineares Modell zur Erklärung von Y. Aber verwende keine andere Variable zur Erklärung von Y, sondern nimm den Mittelwert von Y.

---

<sup>5</sup>Wenn man solche Sätze liest, fühlt sich die Formel fast einfacher an.

```
lm1 <- lm(y ~ 1, data = d)
```

Den MAE können wir uns jetzt so ausgeben lassen:

```
mae(lm1)
## [1] 1.5
```

### 8.4.2. Der Interquartilsabstand

Der Interquartilsabstand (IQA; engl. inter quartile range, IQR) ist ein Streuungsmaß, das nicht auf dem Mittelwert aufbaut. Der IQR ist robuster als z.B. der MAA oder die Varianz und die Standardabweichung.

**Definition 8.4** (Interquartilsabstand). Der Interquartilsabstand ist definiert als der die (absolute) Differenz vom 3. Quartil und 1. Quartil.□

**Beispiel 8.4** (IQR im Hörsaal). In einem Statistikkurs betragen die Quartile der Körpergröße: Q1: 1.65m, Q2 (Median): 1,70m, Q3: 1.75m. Der IQR beträgt dann:  $IQR = Q3 - Q1 = 1.75m - 1.65m = 0.10m$ , d.h. 10 cm.□

Abbildung 8.7 stellt den IQR (und einige Quantile) für den Verkaufspreise von MarioKart-Spielen dar.

### 8.4.5. Streuungsmaße für Normalverteilungen

Normalverteilungen sind recht häufig anzutreffen in der Praxis der Datenanalyse. Daher lohnt es sich, zu überlegen, wie man diese Verteilungen gut zusammenfasst. Man kann zeigen, dass eine Normalverteilung sich komplett über ihren *Mittelwert* sowie ihre *Standardabweichung* beschreiben lässt. Außerdem gilt: Sind Ihre Daten normalverteilt, dann sind die Abweichungen vom Mittelwert auch normalverteilt. Denn wenn man eine Konstante zu einer Verteilung addiert (bzw. subtrahiert), “verschiebt man den Berg” ja nur zur Seite, ohne seine Form zu verändern, s. Abbildung 8.12.

#### i Hinweis

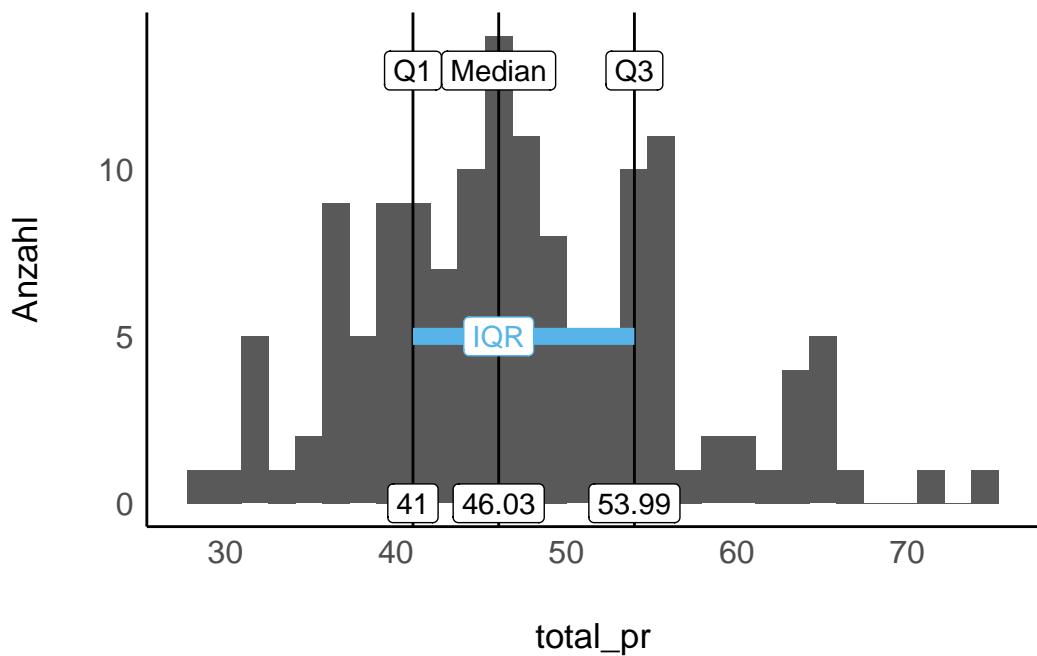
Hat man normalverteilte Variablen/Abweichungen/Residuen, so ist die *Standardabweichung* (engl. standard deviation, SD,  $\sigma$ ,  $s$ ) eine komfortable Maßeinheit der Streuung, denn damit lässt sich die Streuung (Abweichung vom Mittelwert, Residuen) der Normalverteilung gut beschreiben.□

💡 Aber wie berechnet man jetzt diese Standardabweichung?

💡 Moment, noch ein kurzer Exkurs zur Varianz ...

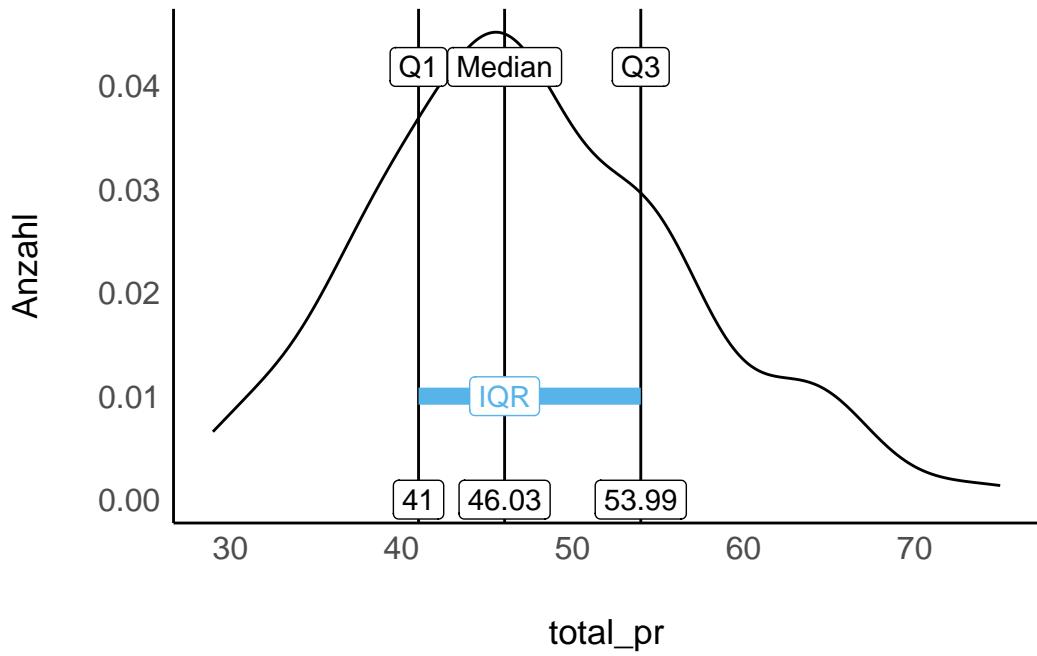
💡 (seufzt)

#### 8.4.3. Histogramm



(a) IQR, Q1, Q2 und Q3 für das Schlussgebot (nur Spiele für weniger als 100 Euro)

#### 8.4.4. Dichtediagramm



(b) IQR, Q1, Q2 und Q3 für das Schlussgebot (nur Spiele für weniger als 100 Euro)

Abbildung 8.7.: Der IQR für den Verkaufspreis von Mariokart-Spielen.

## 8.4.6. Varianz

### 8.4.6.1. Intuition

**i Hinweis**

Die Varianz einer Variable (z.B. Verkaufspreis von MarioKart) ist, grob gesagt, der typische Abstand eines Verkaufspreis vom mittleren Verkaufspreis.□

Abbildung 8.10 illustriert die Varianz:

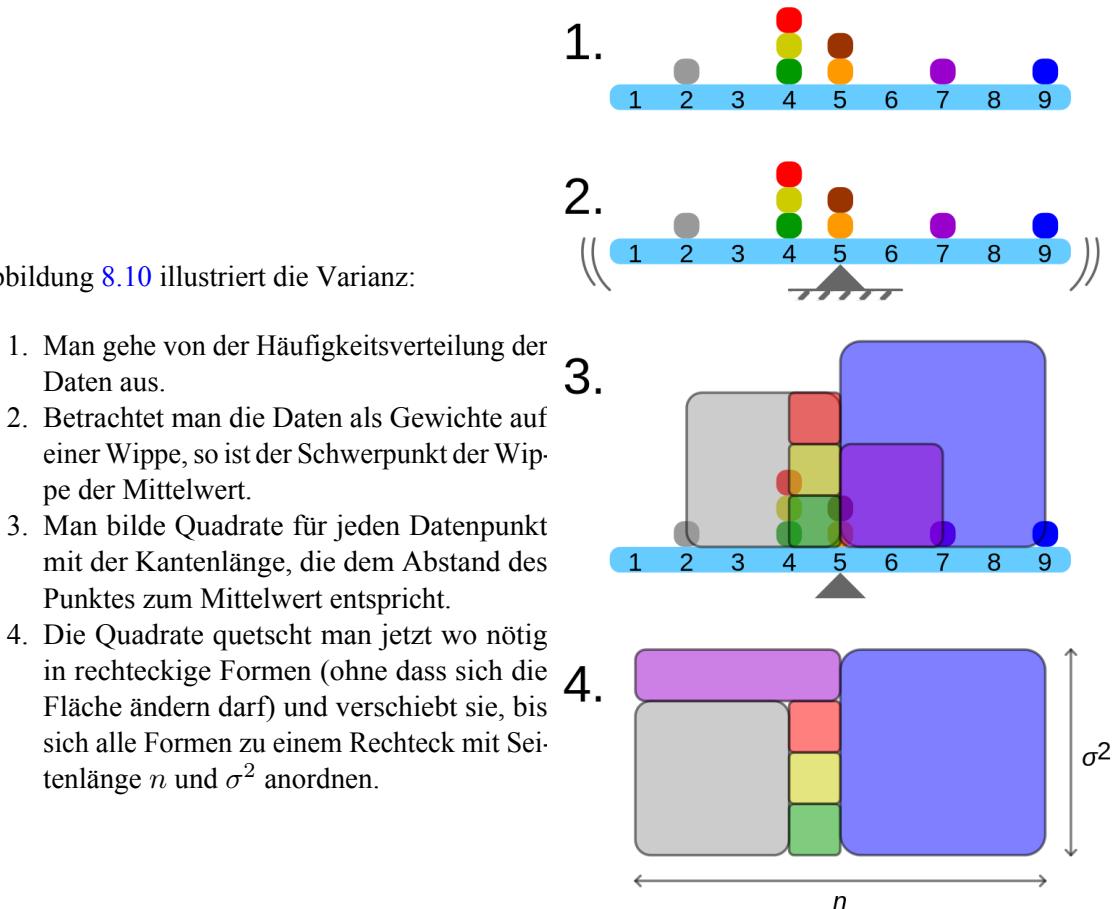


Abbildung 8.8.: Illustration zur Varianz als “mittlerer Quadratfehler”

By Cmglee - Own work, CC BY-SA 3.0

Abbildung 8.9 visualisiert die Varianz für Beispiel 8.3.<sup>6</sup>

<sup>6</sup>Die Abweichungsquadrate wirken optisch nicht quadratisch, da die X-Achse breiter skaliert dargestellt ist als die Y-Achse. Trotzdem sind es Quadrate, nur nicht optisch, wenn Sie wissen, was ich meine...

Links sind die *Abweichungsquadrate* dargestellt, rechts die Varianz als “*typisches Abweichungsquadrat*”.

### **i Hinweis**

Die Varianz ist also ein Maß, das die typische Abweichung der Beobachtungen vom Mittelwert in eine Zahl fasst.□

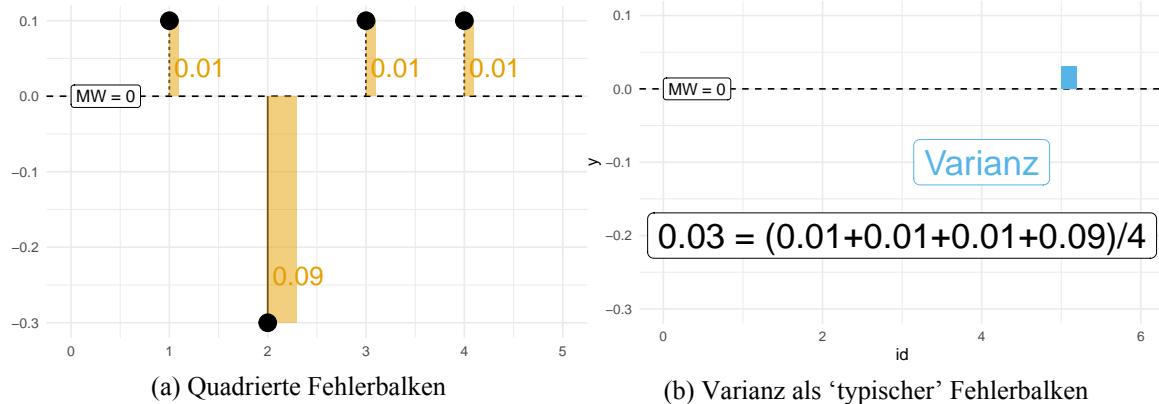


Abbildung 8.9.: Sinnbild zur Varianz als typischer Fehlerbalken

**Beispiel 8.5.** Sie arbeiten immer noch bei einem Online-Auktionshaus und untersuchen den Verkauf von Videospielen. Natürlich mit dem Ziel, dass Ihre Firma mehr von dem Zeug verkaufen kann.

Dazu berechnen Sie die Streuung in den Verkaufspreisen, s. Listing 8.2. □

---

**Listing 8.2** Berechnung der Streuung des Verkaufspreises als Indikatoren für die Modellgüte des Mittelwerts.

---

```
mariokart_no_extreme <-  
  mariokart %>%  
    filter(total_pr < 100)  # ohne Extremwerte  
  
m_summ <-  
  mariokart_no_extreme %>%  
  summarise(  
    pr_mw = mean(total_pr),  
    pr_iqr = IQR(total_pr),  
    pr_maa = mean(abs(total_pr - mean(total_pr))),  
    pr_var = var(total_pr),  
    pr_sd = sd(total_pr))
```

---

| pr_mw | pr_iqr | pr_maa | pr_var | pr_sd |
|-------|--------|--------|--------|-------|
| 47.43 | 12.99  | 7.20   | 83.06  | 9.11  |

Statistiken sind ja schön ... aber Bilder sind auch gut, s. Abbildung 8.10. Datendiagramme eignen sich gut, um (grob) die Streuung einer Variable zu erfassen.

```
mariokart %>%
  mariokart %>%
  select(total_pr) %>%
  filter(total_pr < 100) %>% # ohne Extremwerte
  plot_density()
```

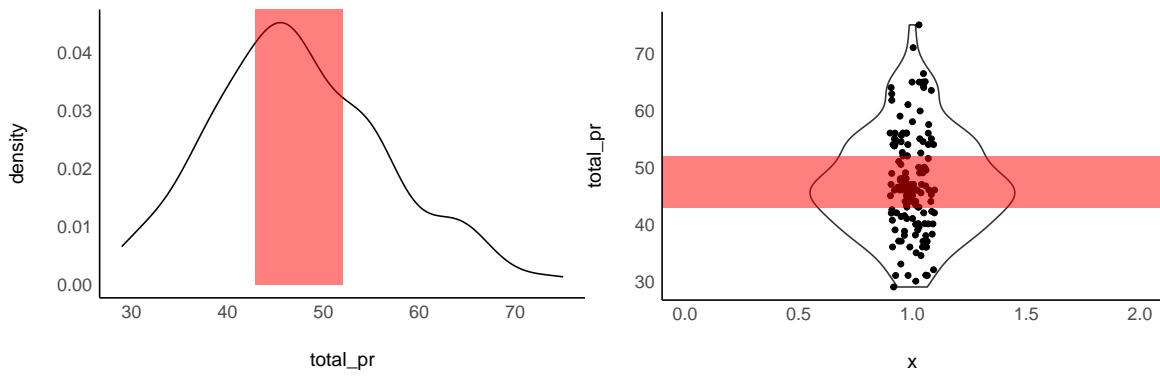


Abbildung 8.10.: Die Verteilung des Verkaufspreises von Mariokart-Spielen

Wer sich die Berechnung von Hand für pr\_maa sparen möchte (s. Listing 8.2), kann die Funktion **MeanAD** aus dem Paket **DescTools** nutzen.

#### 8.4.6.2. Kochrezept für die Varianz

Um die Standardabweichung zu berechnen, berechnet man zunächst die *Varianz*,  $s^2$  abgekürzt. Hier ist ein “Kochrezept”<sup>7</sup> zur Berechnung der Varianz:

1. Für alle Datenpunkte  $x_i$ : Berechne die Abweichungen vom Mittelwert,  $\bar{x}$
2. Quadriere diese Werte
3. Summiere dann auf
4. Teile durch die Anzahl  $N$  der Werte

Als Formel ausgedrückt, lautet die Definition der Varianz<sup>8</sup> einer Stichprobe wie folgt, s. Gleichung 8.2.

<sup>7</sup>Algorithmus

<sup>8</sup>sog. unkorrigierte Stichprobenvarianz; um anhand einer Stichprobe die Varianz der zugehörigen Population zu schätzen, teilt man nicht durch  $N$ , sondern durch  $N - 1$

$$s^2 = \frac{1}{N} \sum_{i=1}^n (y_i - \bar{y})^2 = \frac{1}{N} \sum_{i=1}^n e_i^2. \quad (8.2)$$

**Definition 8.5** (Varianz). Die Varianz ( $s^2, \sigma^2$ ) ist definiert als der Mittelwert der quadrierten Abweichungen,  $e_i^2$ , (vom Mittelwert).  $\square$

Die Varianz steht im engen Verhältnis zur Kovarianz, s. Kapitel 9.3. Die Varianz kann auch verstanden werden als den *mittleren Quadratfehler* (Mean Squared Error, MSE) eines Modells, s. Gleichung 8.3.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2. \quad (8.3)$$

Im Fall eines Punktmodells ist der Mittelwert der vorhergesagte Wert eines Modells.

#### 8.4.7. Die Standardabweichung

Kennt man die Varianz, so lässt sich die Standardabweichung einfach als Quadratwurzel der Varianz berechnen.

**Definition 8.6** (Standardabweichung). Die Standardabweichung (SD,  $s, \sigma$ ) ist definiert als die Quadratwurzel der Varianz, s. Gleichung 8.4.

$$s := \sqrt{s^2} \quad (8.4)$$

Durch das Wurzelziehen besitzt die Standardabweichung wieder *in etwa* die gleiche Größenordnung wie die Daten (im Gegensatz zur Varianz, die durch das Quadrieren sehr groß werden kann).

Aus einem Modellierungsblickwinkel kann man die SD definieren als die Wurzel von MSE. Dann nennt man sie *Root Mean Squared Error* (RMSE):  $RMSE := \sqrt{MSE}$ .

##### i Hinweis

Die SD ist i.d.R. *ungleich* zur MAE, aber (fast) gleich zur RMSE. Entsprechend ist die Varianz (fast) gleich zur MSE.  $\square$

**Beispiel 8.6.** Sie arbeiten weiter an Ihrem Mario Kart-Projekt. Da Sie heute keine Lust auf viel Tippen haben, nutzen Sie das R-Paket `easystats` mit der Funktion `describe_distribution`.

```
library(easystats)

mariokart %>%
  select(total_pr) %>%
  describe_distribution()
```

| Variable | Mean     | SD       | IQR   | Min   | Max    | Skewness | Kurtosis | n   | n_Missing |
|----------|----------|----------|-------|-------|--------|----------|----------|-----|-----------|
| total_pr | 49.88049 | 25.68856 | 12.99 | 28.98 | 326.51 | 9.035897 | 96.14414 | 143 | 0         |

Ah! Das war einfach. Wird auch langsam Zeit für Feierabend.□

**Beispiel 8.7.** Ihr Job als Datenanalyst ist anstrengend, aber auch mitunter interessant. So auch heute. Bevor Sie nach Hause gehen, möchten Sie noch eine Sache anschauen. In einer früheren Analyse (s. Abbildung 8.5) fanden Sie heraus, dass die Fehlerbalken kürzer werden, wenn man ein geschickteres und komplexeres Modell findet. Das wollen Sie natürlich prüfen. Sie überlegen: “Okay, ich will ein einfaches Modell, in dem der Mittelwert das Modell des Verkaufspreis sein soll.”

Das spezifizieren Sie so:

```
lm1 <- lm(total_pr ~ 1, data = mariokart)
mae(lm1)
## [1] 10.01811
```

Im nächsten Schritt spezifizieren Sie ein Modell, in dem der Verkaufspreis eine Funktion der Anzahl der Lenkräder ist (ähnlich wie in Abbildung 8.5):

```
lm2 <- lm(total_pr ~ wheels, data = mariokart)
mae(lm2)
## [1] 7.375873
```

Ah! Sehr schön, Sie haben mit lm2 ein besseres Modell als einfach nur den Mittelwert gefunden. Ab nach hause!□

## 8.5. Streuung als Modellfehler

Wenn wir den Mittelwert als Punktmodell des Verkaufspreises auffassen, so kann man die verschiedenen Kennwerte der Streuung als verschiedene Kennwerte der Modellgüte auffassen.

Definieren wir zunächst als Punktmodell auf Errisch:

```
lm_mario1 <- lm(total_pr ~ 1, data = mariokart)
```

Zur Erinnerung: Wir modellieren total\_pr ohne Prädiktoren, sondern als Punktmodell, und zwar schätzen wir den Mittelwert mit den Daten mariokart.

Das (Meta-)Paket `easystats` bietet komfortable Befehle, um die Modellgüte zu berechnen:

```

mae(lm_mario1) # Mean absolute error
## [1] 10.01811
mse(lm_mario1) # Mean squared error
## [1] 655.2874
rmse(lm_mario1) # Root mean squared error
## [1] 25.59858

```

## 8.6. z-Transformation

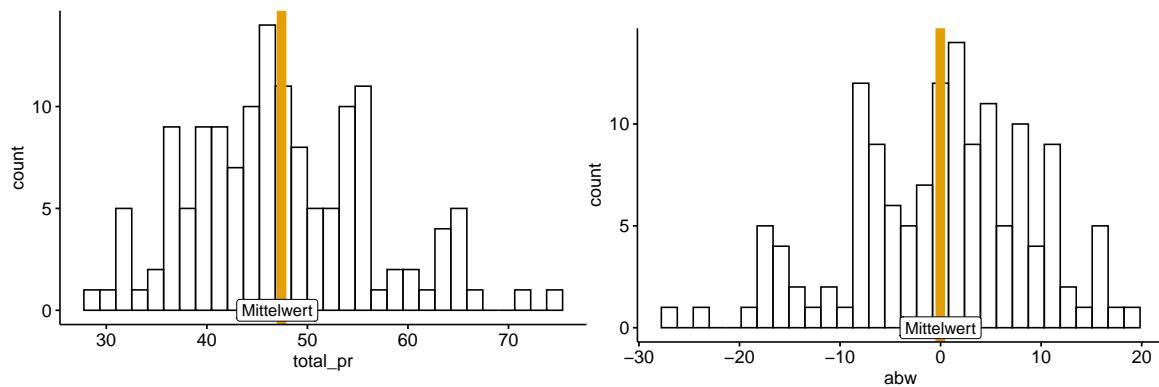
Sie arbeiten immer noch als Datenknecht, Moment, *Datenhecht* bei dem Online-Auktionshaus. Heute untersuchen Sie die Frage, wie gut sich die Verkaufspreise mit einer einzigen Zahl, dem mittleren Verkaufspreis, beschreiben lassen. Einige widerspenstige Werte haben Sie dabei einfach des Datensatzes verwiesen. Schon ist das Leben leichter, s. `mariokart_no_extreme`.

```

mariokart_no_extreme <-
  mariokart %>%
  filter(total_pr < 100)

```

Abbildung 8.11 (links) zeigt, dass es einige Streuung um den Mittelwert herum gibt. Abbildung 8.11 (rechts) zeigt die (um den Mittelwert) *zentrierten* Daten.



- (a) Wie nah drängen sich die Verkaufspreise um ihren Mittelwert? (b) Abweichungen vom Mittelwert: zentrierte Daten

Abbildung 8.11.: Verteilung von `mariokart_no_extreme`

Tja, das ist doch etwas Streuung um den Mittelwert herum.

**!** Wichtig

Je weniger Streuung um den Mittelwert (ca. 47 Euro) herum, desto besser eignet sich der Mittelwert als Modell für die Daten, bzw. desto höher die Modellgüte. □

Ja, es ist *etwas* Streuung, aber wie viel? Kann man das genau angeben? Sie überlegen ... und überlegen. Da! Eine Idee!

Man könnte vielleicht angeben, wie viel Euro jedes Spiel vom Mittelwert entfernt ist. Je größer diese Abweichung, desto schlechter die Modellgüte! Also rechnen Sie diese Abweichung aus.

```
mariokart_no_extreme <-
  mariokart_no_extreme %>%
  mutate(abw = 47.4 - total_pr)
```

Anders gesagt: Wir haben die Verkaufspreise *zentriert*.

**Definition 8.7** (Zentrieren). Zentrieren bedeutet, von jedem Wert einer Verteilung  $X$  den Mittelwert abzuziehen. Daher ist der neue Mittelwert (der zentrierten Verteilung) gleich Null.  $\square$

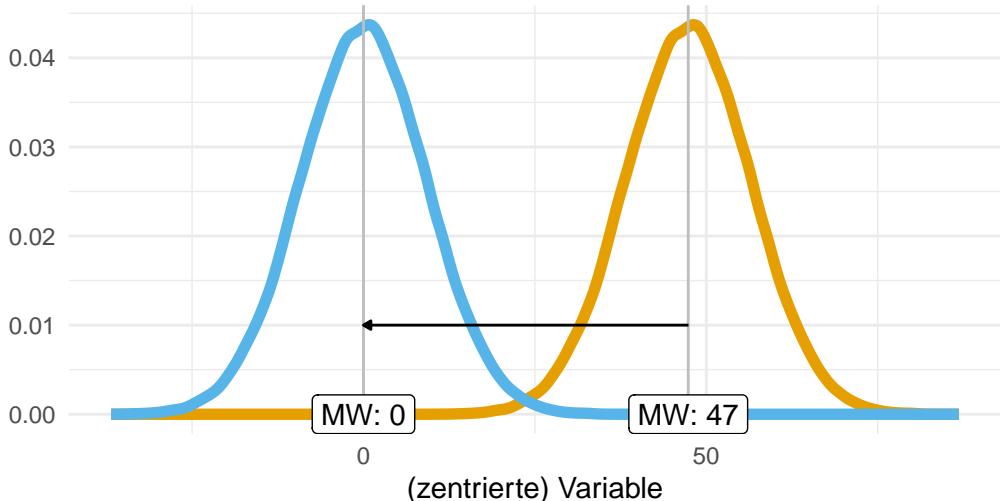


Abbildung 8.12.: Die Abweichungen zum Mittelwert (MW) einer normalverteilten Variable sind selber normalverteilt

Aber irgendwie sind Sie noch nicht am Ziel Ihrer Überlegungen: Woher weiß man, ob 10 Euro oder 20 Euro "viel" Abweichung vom Verkaufspreis ist? Man müsste die Abweichung eines Verkaufspreis zu irgendetwas in Bezug setzen. Wieder! Ein Geistesblitz! Man könnte doch die jeweilige Abweichung in Bezug setzen zur *mittleren (absoluten) Abweichung* (MAA)! Ein alternativer, ähnlicher Kennwert zur mittleren absoluten Abweichung ist die SD. Sie haben gehört, dass die SD gebräuchlicher ist als die MAA. Um sich als Checker zu präsentieren, berechnen Sie also auch die SD; die beiden Koeffizienten sind ja ähnlich.

Also: Wenn ein Spiel 10 Euro vom Mittelwert abweicht und die SD 10 Euro betragen sollte, dann hätten wir eine "standardisierte"<sup>9</sup> Abweichung von 1, weil  $10/10=1$ .

Begeistert über Ihre Schlauheit machen Sie sich ans Werk.

---

<sup>9</sup>abgekürzt manchmal mit std

```

mariokart_no_extreme <-
  mariokart_no_extreme %>%
  mutate(abw_std = abw / sd(abw), # std wie "standardisiert"
        abw_std2 = abw / mean(abs(abw)))

```

Zufrieden betrachten Sie Ihr Werk, s. Abbildung 8.13. In Abbildung 8.13 sieht man oben die Rohwerte und unten die transformierten Werte, die wir hier als *standardisiert* bezeichnen, da wir sie in Bezug zur “typischen Abweichung”, der SD, gesetzt haben.

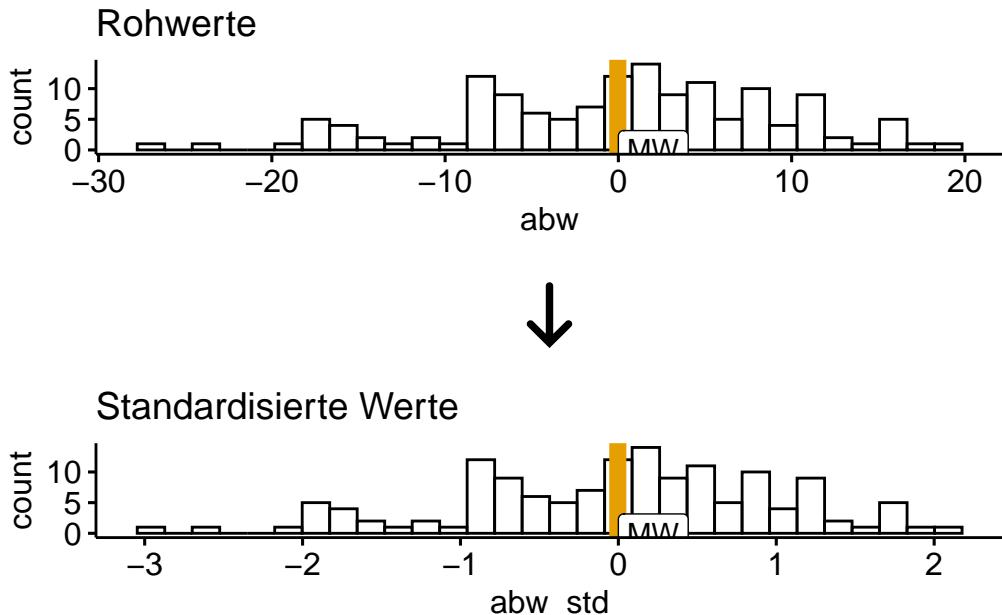


Abbildung 8.13.: Standardisierung von Abweichungswerten bzw. einer Verteilung; der vertikale Balken zeigt den Mittelwert

Wir fassen die Schritte unserer Umrechnung (“Transformation”) zusammen wie in einem Kochrezept:

1. Nimm die Verteilung der Verkaufspreise
2. Berechne die Abweichungen vom mittleren Verkaufspreis (Differenz Mittelwert und jeweiliger Verkaufspreis)
3. Teile die Abweichungen (Schritt 2) durch die SD

Diese Art von Transformation bezeichnet man als *z-Transformation* und die resultierenden Werte als *z-Werte*.

**Definition 8.8** (*z*-Werte). *z*-Werte sind das Resultat der *z*-Transformation. Für die Variable  $X$  berechnet sich der *z*-Wert der  $i$ -ten Beobachtung so:  $z_i = \frac{x_i - \bar{x}}{sd_x}$ .  $\square$

*z*-Werte sind nützlich, weil sie die “relative” Abweichung einzelner Beobachtungen vom Mittelwert anzeigen.

Nach einer *Faustregel* spricht man von extremen Abweichungen (Extremwerten, Ausreißern), wenn  $z_i > 2$  oder  $z_i > 3$ .

## 8.7. Fazit

Der “gesunde Menschenverstand” würde spontan den mittleren Absolutabstand (MAA oder MAE) der Varianz (oder der Standardabweichung, SD) vorziehen. Das ist vernünftig, denn die MAA ist anschaulicher und damit nützlicher als die Varianz und die SD.

Warum sollte man überhaupt ein unanschauliches Maß wie die Varianz verwenden? Wenn es nur um deskriptive Statistik geht, braucht man die Varianz (oder die SD) nicht unbedingt. Gründe, warum Sie die Varianz (bzw. SD) kennen und nutzen sollten, sind:<sup>10</sup>

- Die SD ist sehr nützlich zur Beschreibung der Normalverteilung
- Die Varianz wird häufig verwendet bzw. in Forschungsarbeiten berichtet, also müssen Sie die Varianz kennen.

Liegen Extremwerte vor, kann es vorteilhafter sein, den IQR vorzuziehen gegenüber Mittelwert basierten Streuungsmaßen (MAA, Varianz, SD).

## 8.8. Aufgaben

### 8.8.1. Datenwerk

Die Webseite [datenwerk.netlify.app](https://datenwerk.netlify.app) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

- [mariokart-sd2](#)
- [mariokart-sd3](#)
- [Kennwert-robust](#)
- [summarise04](#)
- [summarise05](#)
- [vis-mariokart-variab](#)
- [sd-vergleich](#)
- [nasa01](#)
- [Streuung-Histogramm](#)
- [mariokart-sd1](#)
- [summarise06](#)
- [mariokart-desk01](#)

**Übungsaufgabe 8.2** (Analysieren Sie den Datensatz zur Handynutzung).

---

<sup>10</sup>Ich wollte noch hinzufügen, dass die Varianz eng verknüpft mit der linearen Algebra, aber ich war nicht sicher, ob das Argument allgemein überzeugen würde.

### **8.8.2. Aufgabe**

Sind Sie händysüchtig? Das ist die Forschungsfrage [dieser Umfrage](#). Nehmen Sie ggf. an dieser Umfrage teil (sie ist anonym und dauert drei Minuten). Laden Sie den [Datensatz zur Handynutzung](#) von Google-Docs herunter.<sup>11</sup> Berechnen Sie dann gängige deskriptive Statistiken und visualisieren Sie sie. □

### **8.8.3. Lösung: Daten importieren**

Sie können die Daten entweder selber herunterladen oder aber die folgende Version des Datensatzes verwenden. In beiden Fällen ist es nützlich, den (absoluten oder relativen) Pfad anzugeben:

```
data_path <- paste0(  
  "https://raw.githubusercontent.com/sebastiansauer/",  
  "statistik1/main/daten/Smartphone-Nutzung%20",  
  "(Responses)%20-%20Form%20responses%201.csv")
```

Dann können Sie die Daten wie gewohnt importieren:

```
smartphone_raw <- read.csv(data_path)
```

### **8.8.4. Lösung: Daten aufbereiten**

Die Spaltennamen sind sehr unschön. Lassen Sie uns daher die Spaltennamen umbenennen (aber vorab sichern):

```
item_labels <- names(smartphone_raw)  
  
names(smartphone_raw) <- paste0("item", 1:ncol(smartphone_raw))
```

### **8.8.5. Fallstudie zur Lebenszufriedenheit**

Die OECD führt eine [weltweite Studie zur Lebenszufriedenheit](#) durch.<sup>12</sup>

Arbeiten Sie die die [Fallstudie “OECD Wellbeing”](#) durch, um ein tieferes Verständnis für die Lebenszufriedenheit in verschiedenen Ländern der Welt zu bekommen.

<sup>11</sup>[https://docs.google.com/spreadsheets/d/1SWMj4rIIJdAsfsSKQHSg8jHr\\_OuKLpJx\\_0XV4LGnH0/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1SWMj4rIIJdAsfsSKQHSg8jHr_OuKLpJx_0XV4LGnH0/edit?usp=sharing)

<sup>12</sup><https://www.oecd.org/wise/measuring-well-being-and-progress.htm>

## **8.9. Literaturhinweise**

Allen Downey (2023) stellt in seinem vergnlich zu lesenden Buch eine kurzweilige Einfhrung in die Statistik vor; auch Streuungsmae haben dabei einen Auftritt. Wer mehr “Lehrbuch-Feeling” sucht, wird bei ([cetinkaya-rundel\\_introduction\\_2021-1?](#)) fndig (das Buch ist online frei verfgbar). Es ist kein Geheimnis, dass Streuungsmae keine ganz neuen Themen in der Statistik sind. Aber hey, Oldie is Goldie, ohne Streuungsmae geht’s nicht. Jedenfalls werden Sie in jedem Statistik-Lehrbuch, dass Sie in der Bib (oder sonstwo) aus dem Regal ziehen, fndig werden zu diesem Thema. Die Bcher unterscheiden sich meist “nur” in ihrem Anspruch bzw. der didaktischen Aufmachung; fr alle ist da was dabei.

## **8.10. Literatur**

# 9. Punktmodelle 2

## 9.1. Lernsteuerung

### 9.1.1. Standort im Lernpfad

Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 9.1.2. Lernziele

- Sie können die Begriffe Kovarianz und Korrelation definieren und ihren Zusammenhänge erläutern.
- Sie können die Stärke einer Korrelation einschätzen.

### 9.1.3. Benötigte R-Pakete

In diesem Kapitel benötigen Sie folgende R-Pakete.

```
library(tidyverse)
library(easystats)
```

### 9.1.4. Benötigte Daten

Listing 8.1 definiert den Pfad zum Datensatz `mariokart` und importiert die zugehörige CSV-Datei in R, so dass wir einen Tibble mit Namen `mariokart` erhalten.

```
mariokart_path <- paste0(
  "https://vincentarelbundock.github.io/Rdatasets/",
  "csv/openintro/mariokart.csv")

mariokart <- read.csv(mariokart_path)
```

### 9.1.5. Zum Einstieg

#### Beispiel 9.1.

1. Suchen Sie sich eine vertrauenwürdige Partnerin oder einen vertrauenswürdigen Partner. Im Zweifel reicht die Person, die neben Ihnen sitzt.  $\square$
2. Nennen Sie zwei Variablen, die wie folgt zusammenhängen:
  - gleichsinnig (Viel von dem einen, viel von dem anderen)
  - gegensinnig (viel von dem einen, wenig von dem anderen)
  - Scheinzusammenhang (hängt zusammen, ist aber nicht "echt" bzw. kausal)

## 9.2. Zusammenfassen zum Zusammenhang

In Kapitel 7 haben wir gelernt, dass das Wesen eines Punktmodells als Zusammenfassung *einer* Spalte (eines Vektors) zu einer einzelnen Zahl<sup>1</sup>, zu einem "Punkt" sozusagen, zusammengefasst werden kann.

In diesem Kapitel fassen wir *zwei* Spalten zusammen, wieder zu *einer* Zahl, s. Gleichung 9.1.

$$\begin{array}{c|c} \boxed{\phantom{0}} & + \\ \boxed{\phantom{0}} & \end{array} \rightarrow \boxed{\phantom{00}} \quad (9.1)$$

Wo wir in Kapitel 7 eine Variable mit Hilfe eines Lagemaßes beschrieben (bzw. dargestellt, zusammengefasst, modelliert) haben, tun wir hier das Gleiche für zwei Variablen. Beschreibt man aber zwei Variablen, so geht es um die Frage, was die beiden Variablen miteinander zu tun haben: Wie die beiden Variablen von einander *abhängen* bzw. miteinander (irgendwie) *zusammenhängen*. Wir begrenzen auf *metrische* Variablen.

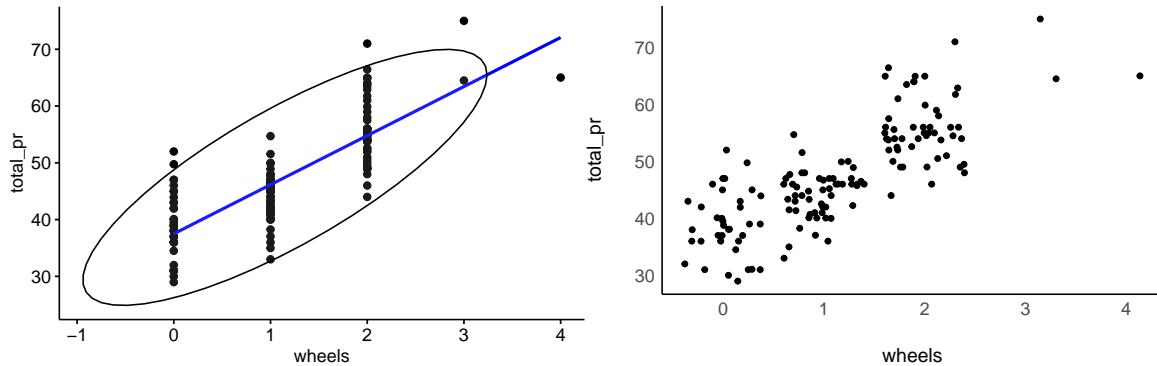
### 9.2.1. Beispiele für Zusammenhänge

- Lernzeit und Klausurerfolg
- Körpergröße und Schuhgröße
- Verbrauchtes Benzin und zurückgelegte Strecke
- Produktionsmenge und Produktionskosten
- Bildschirmzeit und Schlafqualität
- Umweltschutz und Biodiversität  $\square$

Die Verbildlichung<sup>2</sup> zweier metrischer Variablen haben wir bereits in Kapitel 6.5.4 kennengelernt. Zur Verdeutlichung, wie ein Zusammenhang zweier metrischer Variablen aussehen kann, helfe noch einmal Abbildung 9.1.

<sup>1</sup>auch Skalar genannt

<sup>2</sup>Visualisierung



(a) Streudiagramm mit Trendlinie (und Ellipse zur Verdeutlichung)  
(b) ‘Verwackeltes’ Streudiagramm, um die einzelnen Punkte besser zu erkennen

Abbildung 9.1.: Visualisierung des Zusammenhangs von wheels und total\_pr

### 9.3. Abweichungsrechtecke

Die Stärke des linearen Zusammenhangs zweier metrischer Variablen kann man gut mithilfe von Abweichungsrechtecken veranschaulichen. Los geht’s!

#### 9.3.1. Noten und Abweichungsrechtecke

**Beispiel 9.2** (Wieder Statistiknoten). Anton, Bert, Carl und Daniel haben ihre Statistikklausur zurückbekommen. Die Lernzeit  $X$  scheint mit der erreichten Punktzahl  $Y$  (0-100, je mehr desto besser) zusammenzuhängen.<sup>3</sup> Gar nicht so schlecht ausgefallen wie gedacht ..., s. Tabelle 9.1.□

Tabelle 9.1.: Punkte in der Statistikklausur (0-100) und Lernzeit (0-100)

| id | y  | x  |
|----|----|----|
| 1  | 72 | 70 |
| 2  | 44 | 40 |
| 3  | 39 | 35 |
| 4  | 50 | 67 |

Zeichnen wir uns die Daten als Streudiagramm, s. Abbildung 9.2. Dabei zeichnen wir noch *Abweichungsrechtecke* ein.

**Definition 9.1** (Abweichungsrechteck). Im zweidimensionalen Fall spannt sich ein Abweichungsrechteck vom Mittelwert  $\bar{x}$  bis zum Messwert  $x_i$  und genauso für  $Y$ . Wir bezeichnen mit  $dx_i$  die Distanz (Abweichung) vom Mittelwert  $\bar{x}$  bis zum Messwert  $x_i$  (und analog  $dy_i$ ), also  $dx_i = x_i - \bar{x}$ . Die Fläche des Abweichungsrechtecks ist dann das Produkt der Abweichungen:  $dx_i \cdot dy_i$ .□

Stellen Sie sich vor, wir legen alle Rechtecke zusammen aus Abbildung 9.2. Nennen wir das resultierende Rechteck das “Summenrechteck”. Ja, ich weiß, ich strapaziere mal wieder Ihre Phantasie<sup>4</sup>. Jetzt kommt’s: Je größer die Fläche des Summenrechtecks, desto stärker der (lineare) Zusammenhang. Beachten Sie, dass die Flächen Vorzeichen haben, positiv oder negativ (Plus oder Minus), je nach dem, in welchem der vier Quadranten sie stehen. Die Füllfarben der Rechtecke verdeutlichen dies, s. Abbildung 9.2. Das Vorzeichen der Summe zeigt ob der Zusammenhang positiv (gleichsinnig, ansteigende Trendlinie) oder negativ (gegensinnig, absinkende Trendlinie) ist. So zeigt Abbildung 9.3 links eine positive Summe der Abweichungsrechtecke und rechts eine negative Summe. Man sieht im linken Diagramme, dass die Summe der Rechtecke mit positivem Vorzeichen (rot) überwiegt; im rechten Diagramm ist es umgekehrt (blau, negativ überwiegt).

Wir können das Summenrechteck noch durch die Anzahl der Datenpunkte teilen, das ändert nichts an der Aussage, aber der Mittelwert hat gegenüber der Summe den Vorteil, dass er unabhängig ist in

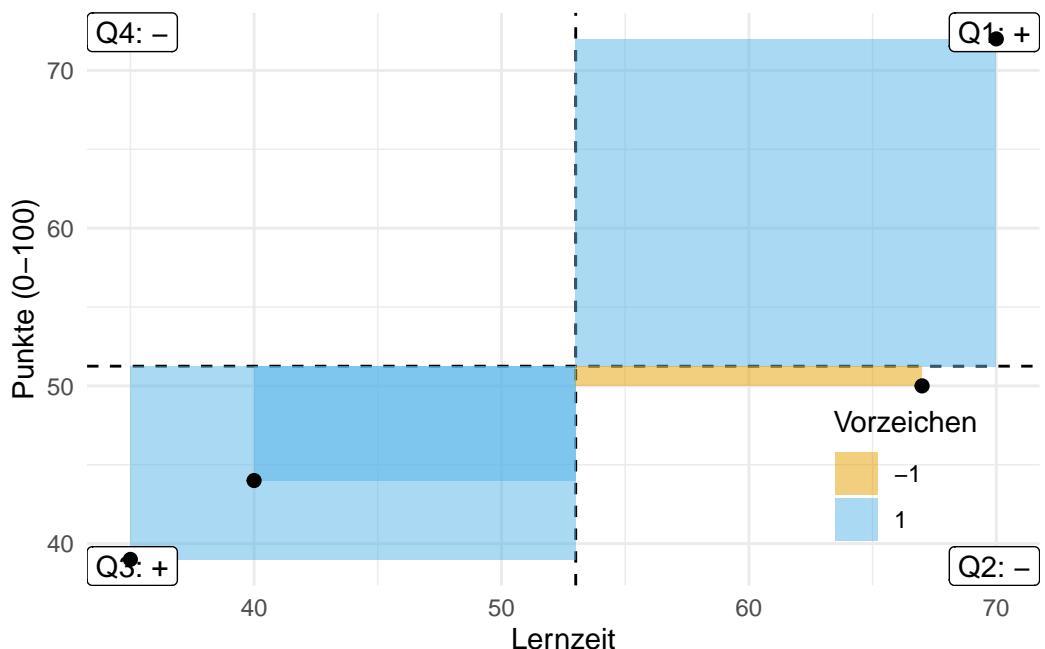
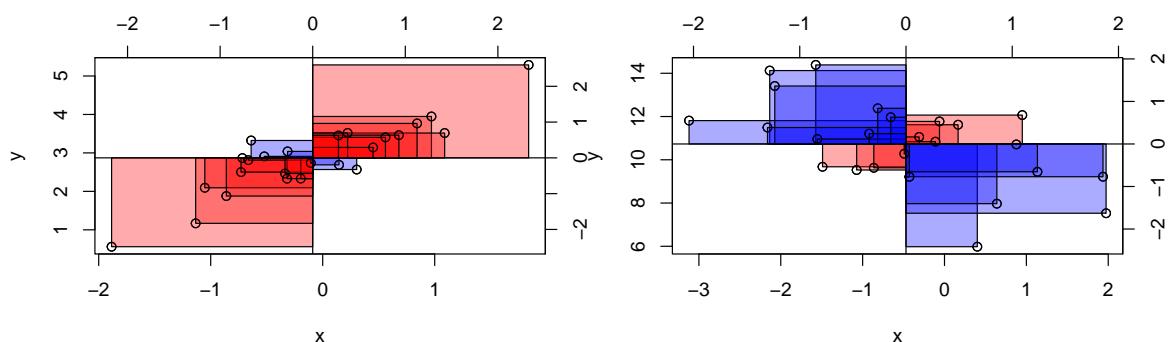


Abbildung 9.2.: Die Kovarianz als mittleres Abweichungsrechteck. In jedem der vier Quadranten (Q1, Q2, Q3, Q4) ist das Vorzeichen der Abweichungsrechtecke dargestellt. Die Farben der Abweichungsrechtecke spiegeln das Vorzeichen wider.



(a) Positive Vorzeichen (Quadranten 1 und 3) überwiegen, was in einer positiven Kovarianz resultiert - Negative Vorzeichen (Quadranten 2 und 4) überwiegen, was in einer negativen Kovarianz resultiert

(b) Positive Vorzeichen (Quadranten 1 und 3) überwiegen, was in einer positiven Kovarianz resultiert - Negative Vorzeichen (Quadranten 2 und 4) überwiegen, was in einer negativen Kovarianz resultiert

Abbildung 9.3.: Positive und negative Kovarianz: Einmal resultiert eine positive Summe, einmal eine negative Summe, wenn man die Flächen der Abweichungsrechtecke addiert.

seiner Aussage von der Anzahl der eingegangenen Datenpunkte. Das resultierende Rechteck nennen wir das *mittlere Abweichungsrechteck*.

Ein Maß für den Zusammenhang von Lernzeit und Klausurpunkte ist also die *Fläche des mittleren Abweichungsrechtecks*, s. Abbildung 9.4.

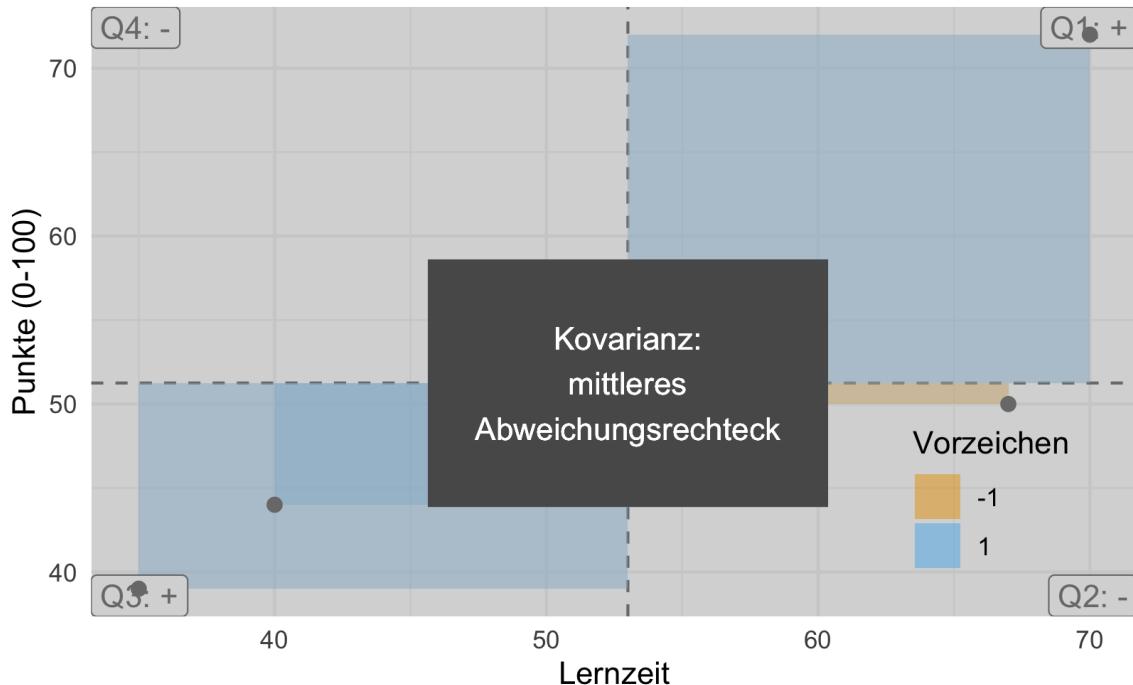


Abbildung 9.4.: Die Kovarianz als mittleres Abweichungsrechteck. Die Fläche der Rechtecks entspricht dem Wert der Kovarianz.

### 9.3.2. Kovarianz

**Definition 9.2** (Kovarianz). Die Kovarianz ist definiert als die Fläche des mittleren Abweichungsrechtecks. Sie ist ein Maß für die Stärke und Richtung des linearen Zusammenhangs zweier metrischer Variablen, s. Abbildung 9.4. □

💡 Zu viele Bilder! Ich brauch Zahlen.

⚠️ Kommen gleich!

Tabelle 9.2 zeigt die Werte für die X- und Y-Abweichung und die resultierenden Flächen der Abweichungsrechtecke. Wenn Sie die Werte selber nachrechnen wollen, finden Sie den Noten-Datensatz in der Datei [noten.csv](#)<sup>5</sup>.

<sup>5</sup><https://raw.githubusercontent.com/sebastiansauer/statistik1/main/daten/noten.csv>

Tabelle 9.2.: Werte der Abweichungsrechtecke. avg: average (Mittelwert), cov\_sign: Vorzeichen der Kovarianz,\_pos: positiver Wert auf der entsprechenden Achse (x/y)?, xy\_area: Produkt von x\_delta und y\_delta

| id | y  | x  | x_avg | y_avg | x_delta | y_delta | cov_sign | xy_area |
|----|----|----|-------|-------|---------|---------|----------|---------|
| 1  | 72 | 70 | 53    | 51.25 | 17      | 20.75   | 1        | 352.75  |
| 2  | 44 | 40 | 53    | 51.25 | -13     | -7.25   | 1        | 94.25   |
| 3  | 39 | 35 | 53    | 51.25 | -18     | -12.25  | 1        | 220.50  |
| 4  | 50 | 67 | 53    | 51.25 | 14      | -1.25   | -1       | -17.50  |

Berechnen wir als nächstes das mittlere Abweichungsrechteck, die Kovarianz:

```
d %>%
  summarise(kovarianz = mean(xy_area))
```

$$\begin{array}{c} \overline{\text{kovarianz}} \\ \hline 162.5 \end{array}$$

Die Formel der Kovarianz lautet, s. Gleichung 9.2:

$$\text{cov}(xy) = s_{xy} := \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = \frac{1}{n} \sum_{i=1}^n dx_i \cdot dy_i \quad (9.2)$$

Gleichung 9.2 in Worten ausgedrückt:

1. Rechne für jedes  $x_i$  die Abweichung vom Mittelwert,  $\bar{x}$ , aus,  $dx_i$ .
2. Rechne für jedes  $y_i$  die Abweichung vom Mittelwert,  $\bar{y}$ , aus,  $dy_i$ .
3. Multipliziere für alle  $i$   $dx_i$  mit  $dy_i$ , um die Abweichungsrechtecke  $dx_i dy_i$  zu erhalten.
4. Addiere die Flächen der Abweichungsrechtecke.
5. Teile durch die Anzahl der Beobachtungen  $n$ .

**Beispiel 9.3** (Variablen mit positiver Kovarianz).

- Größe und Gewicht
- Lernzeit und Klausurerfolg
- Distanz zum Ziel und Reisezeit
- Temperatur und Eisverkauf

**Beispiel 9.4** (Variablen mit negativer Kovarianz).

- Lernzeit und Freizeit
- Alter und Restlebenszeit

- Temperatur und Schneemenge
- Lebenszufriedenheit und Depressivität

Drei Extrembeispiele für Kovarianz-Werte sind in Abbildung 9.5 dargestellt.

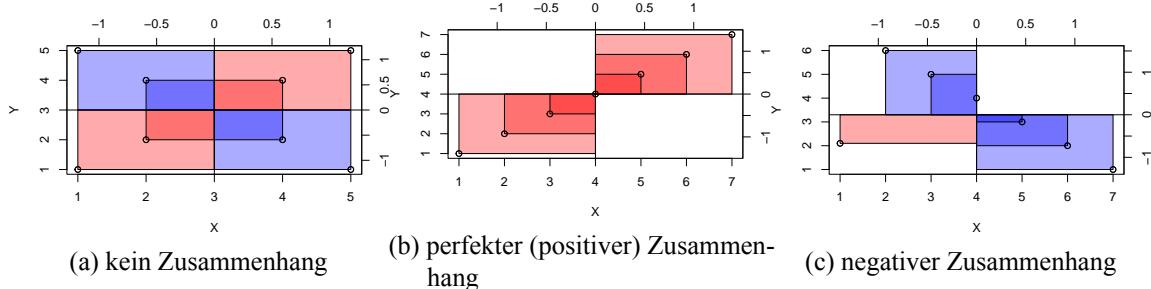


Abbildung 9.5.: Verschiedene Werte der Kovarianz

Bei einer Kovarianz von (ungefähr) 0 ist die Gesamt-Fläche der Abweichungsrechtecke<sup>6</sup>, wenn man sie pro *Quadrant* aufsummiert, ungefähr gleich groß, s. Abbildung 9.6. Addiert man die Abweichungsrechtecke (unter Beachtung der Vorzeichen: rot = positiv; blau = negativ), so beträgt die Summe in etwa (oder genau) Null.

Damit ist die Kovarianz in diesem Fall etwa (bzw. genau) Null, s. Gleichung 9.3: Wenn die Summe der Aweichungsrechtecke Null ist, dann ist auch ihr Mittelwert (MW) Null. Damit ist die Kovarianz Null.

$$\begin{aligned} \sum (dX \cdot dY) &= 0 \\ \Leftrightarrow \text{MW } (dX \cdot dY) &= 0 \\ \Leftrightarrow \text{cov}(X, Y) &= 0 \end{aligned} \tag{9.3}$$

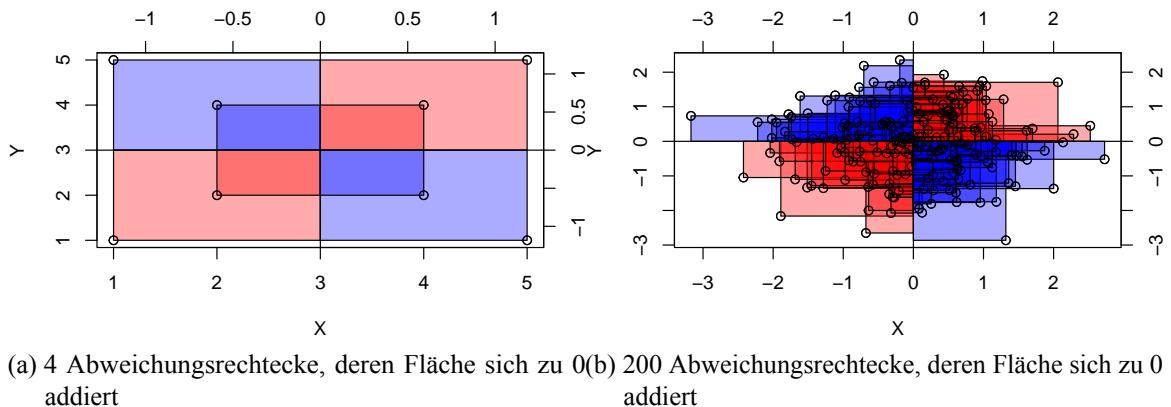


Abbildung 9.6.: Wenn die Kovarianz 0 ist, gleichen sich die Abweichungsrechtecke auf 0 aus

<sup>6</sup>Bei der Varianz waren es Quadrate, bei der Kovarianz sind es Rechtecke.

### 9.3.3. Die Kovarianz ist schwer zu interpretieren

Die Kovarianz hat den Nachteil, dass sie abhängig ist von der Skalierung. So steigt die Kovarianz z.B. um den Faktor 100, wenn man eine Variable (z.B. Einkommen) anstelle von Euro in Cent bemisst. Das ist nicht wünschenswert, denn der Zusammenhang zwischen z.B. Einkommen und Lebenszufriedenheit ist unabhängig davon, ob man Einkommen in Euro, Cent oder Dollar misst. Außerdem hat die Kovarianz keinen Maximalwert, der einen perfekten Zusammenhang anzeigt. Insgesamt ist die Kovarianz schwer zu interpretieren und wird in der praktischen Anwendung nur wenig verwendet.

## 9.4. Korrelation

### 9.4.1. Korrelation als mittleres z-Produkt

Der Korrelationskoeffizient  $r$  nach Karl Pearson löst das Problem, dass die Kovarianz schwer interpretierbar ist. Der Wertebereich von  $r$  reicht von -1 (perfekte negative lineare Korrelation) bis +1 (perfekte positive lineare Korrelation). Eine Korrelation von  $r = 0$  bedeutet *kein linearer Zusammenhang*.

Die Korrelation berechnet sich wie folgt:

1. Teile alle  $x_i$  durch ihre Standardabweichung,  $s_x$
2. Teile alle  $y_i$  durch ihre Standardabweichung,  $s_y$
3. Berechne mit diesen Werten die Kovarianz

Teilt man nämlich alle  $x_i$  bzw.  $y_i$  durch ihre Standardabweichung, so führt man mit  $X$  bzw.  $Y$  eine z-Transformation durch. Daher kann man den Korrelationskoeffizienten  $r$  so definieren:

**Definition 9.3** (Korrelationskoeffizient  $r$ ). Der Korrelationskoeffizient  $r$  (nach Pearson) ist definiert als das mittlere Produkt der z-Wert-Paare, s. Gleichung 9.4, vgl. Cohen u. a. (2003). Er ist ein Maß des linearen Zusammenhangs zweier metrischer Variablen. Der Wertebereich ist  $[-1; 1]$ , wobei 0 keinen Zusammenhang anzeigt und  $|r| = 1$  perfekten Zusammenhang.  $\square$

$$r_{xy} = \frac{1}{n} \sum_{i=1}^n z_{x_i} z_{y_i} \quad (9.4)$$

Man beachte, dass eine Korrelation (genauso wie eine Kovarianz) nur für metrische Variablen definiert ist.

#### **i Hinweis**

Aus dem Korrelationskoeffizienten können Sie zwei Informationen ableiten:

1. *Vorzeichen*: Ein positives Vorzeichen bedeutet positiver (gleichsinniger) linearer Zusammenhang (und umgekehrt: negatives Vorzeichen, negativer, also gegensinniger linearer Zusammenhang).

2. *Absolutwert* der Korrelation: Der Absolutwert<sup>7</sup> des Korrelationskoeffizienten gibt die Stärke des linearen Zusammenhangs an. Je näher der Wert bei 1 liegt desto stärker der Zusammenhang.

- $r = 0$ : kein linearer Zusammenhang
- $r = 1$ : perfekter linearer Zusammenhang  $\square$

Eine Zuordnung des Korrelationskoeffizienten zum Profil des Streudiagramms zeigt Abbildung 9.7.

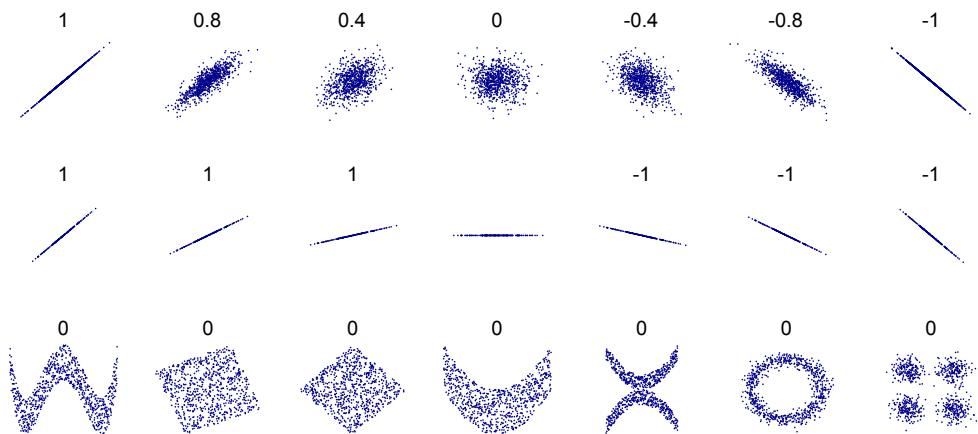


Abbildung 9.7.: Verschiedene Streudiagramme, die sich in ihrem Korrelationskoeffizienten unterscheiden. Quelle: Wikipedia, By DenisBoigelot, original uploader was Imagecreator, CC0, <https://commons.wikimedia.org/w/index.php?curid=15165296> CC0

Die untere Zeile von Abbildung 9.7 zeigt Beispiele für nicht-lineare Zusammenhänge. Wie man sieht, liegt in diesen Beispielen kein linearer Zusammenhang vor ( $r = 0$ ), obwohl ein starker *nicht*-linearer Zusammenhang besteht.

**Übungsaufgabe 9.1** (Korrelationsspiel). Spielen Sie das [Korrelationsspiel](#): Sie Sehen ein Streudiagramm und müssen den richtigen Korrelationskoeffizienten eingeben.  $\square$

**Übungsaufgabe 9.2** (Interaktive Visualisierung der Korrelation). Auf der Seite von [RPsychologist](#) findet sich eine ansprechende dynamische Visualisierung der Korrelation. Nutzen Sie sie, um Ihr Gefühl für die Stärke des Korrelationskoeffizienten zu entwickeln.  $\square$

#### 9.4.2. Korrelation mit R berechnen

Ob der Verkaufspreis (`total_pr`) wohl mit der Dauer der Auktion (`duration`) oder mit der Anzahl der Gebote (`n_bids`) (`linear`) zusammenhängt? Schauen wir nach! Die Funktion `correlation()` (aus dem Paket `{easystats}`) erledigt das Rechnen für uns, s. [?@tbl-mario-corr1](#).

---

<sup>7</sup>Betrag

```
mariokart |>
  select(total_pr, duration, n_bids) |>
  correlation() |> # aus `easystats`  

  summary()
```

Tabelle 9.4.: Korrelation berechnen mittels der Funktion `correlation` aus `easystats`

| Parameter | n_bids | duration |
|-----------|--------|----------|
| total_pr  | 0.13   | -0.04    |
| duration  | -0.12  |          |

p-value adjustment method: Holm (1979)

Sie können auch auf die letzte Zeile, also dem Befehl `summary()` verzichten. Dann ist die Ausgabe ausführlicher.

#### 9.4.3. Korrelation ≠ Kausation

Eine Studie fand eine starke Korrelation, zwischen der (Höhe des) Schokoladenkonsums eines Landes und (Anzahl der) Nobelpreise eines Landes (Messerli 2012), s. Abbildung 9.8.

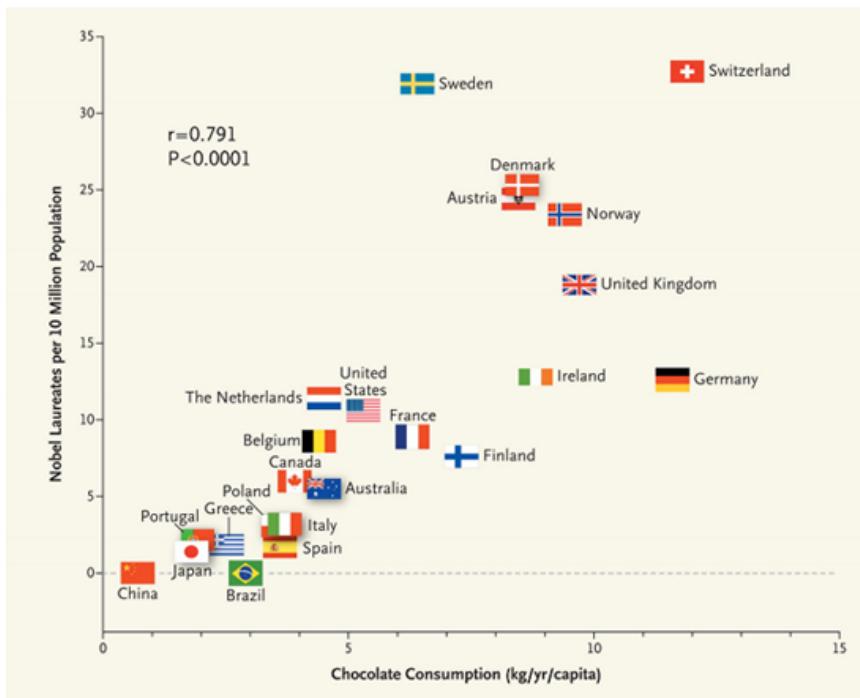


Abbildung 9.8.: Schoki futtern macht schlau?

### Vorsicht

Korrelation (bzw. Zusammenhang) ungleich Kausation! Korrelation kann bedeuten, dass eine Kausation vorliegt, aber es muss auch nicht sein, dass Kausation vorliegt. Liegt Korrelation ohne Kausation vor, so spricht man von einer Scheinkorrelation.

#### **9.4.4. Korrelation misst nur linearen Zusammenhang**

**Beispiel 9.5** (Scheinkorrelation). *Störche und Babies*: Eine Urban Myth besagt: Die Anzahl der Störche pro Landkreis korreliert mit der Anzahl der Babies in diesem Landkreis.

Eine Erklärung für dieses (nur scheinbare) Paradoxon ist, dass die “Naturbelassenheit” des Landkreises die gemeinsame Ursache für Störche ist (Störche lieben Natur) und für Babies ist (die dortige Kultur begünstigt, mehr Kinder pro Frau).

*Corona und Glatze*:

Macht die Glatze krank? Männer mit Glatze bekommen häufiger Corona (Goren u. a. 2020).

Bald men at higher risk of severe case of Covid-19, research finds<sup>8</sup>

Eine Erklärung lautet, dass Alter einen Effekt hat auf Glatze (je älter ein Mann, desto wahrscheinlicher ist es, dass er eine Glatze hat) und auf die Schwere des Corona-Verlaufs (ältere Menschen haben deutlich schwerere Corona-Verläufe). □

### **9.5. Wie man mit Statistik lügt**

#### **9.5.1. Range-Restriktion**

Durch (nicht-randomisierte) Einschränkung (Restriktion) des Ranges einer (oder beider) Variablen sinkt die Stärke (der Absolutwert) einer Korrelation, vgl. Cohen u. a. (2003) und Abbildung 9.9.

Erstellen wir uns dazu zwei Datensätze mit je zwei Variablen,  $X$  und  $Y$  der Größe  $n = 100$ . Ein Datensatz ist ohne Einschränkung des Ranges und einer mit.  $X$  und  $Y$  seien normalverteilt mit  $\mu = 0$  (Mittelwert) und  $\sigma = 1$  (Streuung); s. Datensatz `d` in Listing 9.1. Wir schränken dann den Range von  $X$  ein auf, sagen wir, den Bereich von  $[-0.5, .5]$  (Datensatz `d_filtered`).

**Übungsaufgabe 9.3** (Berechnen Sie die Korrelation). Glauben Sie nicht, prüfen Sie nach! Berechnen Sie die Korrelation von  $X$  und  $Y$  im Datensatz `d` und `d_filtered`! □

<sup>8</sup><https://www.telegraph.co.uk/global-health/science-and-disease/bald-men-higher-risk-severe-case-covid-19-research-finds/>, Abruf 2023-03-24

---

### **Listing 9.1**

---

```
set.seed(42)
n <- 1e2
d <-
  tibble(x = rnorm(n = n, mean = 0, sd = 1),
         e = rnorm(n = n, mean = 0, sd = .5),
         y = x + e)

x_min <- -0.5
x_max <- 0.5

d_filtered <-
d |>
  filter(between(x, x_min, x_max))
```

---

## **9.6. Fallbeispiel**

In Ihrer Arbeit beim Online-Auktionshaus analysieren Sie, welche Variablen mit dem Verkaufspreis von Computerspielen zusammenhängen.

Falls der Datensatz auf Ihrem Computer (am besten in Ihrem Projektverzeichnis in RStudio) abgelegt ist, können Sie die Daten so (in mittlerweile gewohnter Manier) importieren:

```
mariokart <- read.csv("mariokart.csv")
```

Falls der Datensatz im Unterordner mit Namen “Mein\_Unterordner” liegt, so würden Sie folgenden Pfad eingeben:

```
mariokart <- read.csv("Mein_Unterordner/mariokart.csv")
```

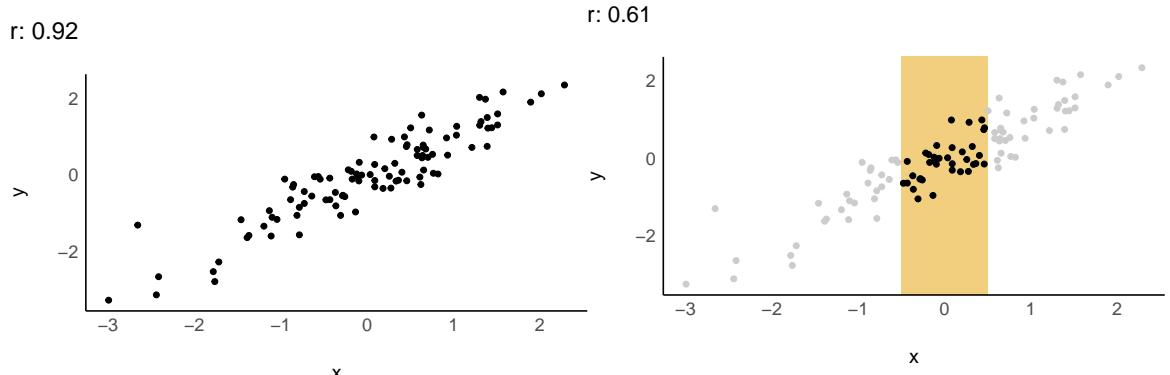
Man beachte, dass solche sog. relativen Pfade (relativ zu Ihrem Arbeitsverzeichnis, d.h. Ihr Projektverzeichnis in R-Studio) *nicht* mit einem Schrägstrich (Slash) beginnen.

Falls Sie die Daten nicht auf Ihrem Computer haben, können Sie sie komfortable von z.B. der Webseite von [Vincent Arel-Bundock](#) herunterladen:

Den Pfad hatten wir in Listing 8.1 definiert.

```
mariokart <- read.csv(mariokart_path)
```

Sie wählen die Variablen von `mariokart`, die Sie in diesem Fall interessieren – natürlich nur die metrischen – und lassen sich mit `cor` die Korrelation aller Variablen untereinander ausgeben:



(a) Ohne Einschränkung des Range: Starke Korrelation  
(b) Mit Einschränkung des Range: Schwächere Korrelation

Abbildung 9.9.: Schränkt man den Range einer (oder beider) Variablen ein, so sinkt die Stärke der Korrelation

```
mariokart %>%
  dplyr::select(start_pr, ship_pr, total_pr) %>%
  cor() %>%
  round(2) # Runden auf zwei Dezimalen
##          start_pr  ship_pr  total_pr
## start_pr      1.00    0.03     0.07
## ship_pr       0.03    1.00     0.54
## total_pr      0.07    0.54     1.00
```

### 🔥 Namensverwechslung (name clash)

Es kann vorkommen, dass Sie zwei R-Pakete geladen haben, in denen es jeweils z.B. eine Funktion mit Namen `select` gibt. R wird in dem Fall diejenige Funktion verwenden, deren Paket Sie als letztes gestartet haben. Das kann dann das falsche `select` sein, wie es mir oben in der Syntax passiert ist. In dem Fall resultiert eine verwirrende Fehlermeldung, die sinngemäß sagt: "Hey Mensch, du hast Argumente in der Funktion verwendet, die du gar nicht verwenden darfst, da es sie nicht gibt." Auf Errisch: `Error in select(.., duration, n_bids, start_pr, ship_pr, total_pr, seller_rate, : unused arguments (duration, n_bids, start_pr, ship_pr, total_pr, seller_rate, wheels)`. Eine einfache Abhilfe ist es, R zu sagen: "Hey R, nimm gefälligst `select` aus dem Paket `dplyr`, dort"wohnt" nämlich `select`". Auf Errisch spricht sich das so: `dplyr::select(...)`. □

Etwas schöner sieht die Ausgabe mit dem Befehl `correlation` aus `{easystats}` aus, s. Tabelle 9.5.

```
mariokart %>%
  dplyr::select(start_pr, ship_pr, total_pr) %>%
  correlation() |>
  summary() |>
  print_md()
```

Tabelle 9.5.: Korrelationstabelle (tidy) im Datensatz mariokart

Tabelle 9.5.: Correlation Matrix (pearson-method)

| Parameter | total_pr | ship_pr |
|-----------|----------|---------|
| start_pr  | 0.07     | 0.03    |
| ship_pr   | 0.54***  |         |

p-value adjustment method: Holm (1979)

Neben einigen Statistiken, die wir einfach geflissentlich ausblenden ( $t$  und  $p$ ) beinhaltet die Tabelle eine interessante Information: den Schätzbereich für die Korrelation, gekennzeichnet als 95% CI. Grob gesagt können wir diese Information so interpretieren: "Mit 95% Wahrscheinlichkeit liegt der echte Wert der Korrelation in folgendem Bereich."<sup>9</sup>

Möchte man nur einzelne Korrelationskoeffizienten ausrechnen, können wir die Idee des Zusammenfassens, s. Gleichung 9.1, nutzen:

```
mariokart %>%
  summarise(cor_super_wichtig = cor(total_pr, wheels))
```

| cor_super_wichtig |
|-------------------|
| 0.3299838         |

### 🔥 Vorsicht

Im Falle von fehlenden Werten müssen Sie den Befehl `cor()` aus seiner schüchternen Vorsicht befreien und ermutigen, trotz fehlender Werte einen Korrelationskoeffizienten auszugeben. Das geht mit dem Argument `use = "complete.obs"` in `cor`.

```
mariokart %>%
  summarise(cor_super_wichtig = cor(total_pr, wheels, use =
    ↴ "complete.obs"))
```

<sup>9</sup>Bayesianische Interpretation

|                          |
|--------------------------|
| <u>cor_super_wichtig</u> |
| 0.3299838                |

💡 Immer so viele Zahlen! Ich brauch Bilder.

Mit dem Befehl `plot_correlation` aus dem R-Paket `{dataExplorer}` bekommt man eine ansehnliche Heatmap zur Verdeutlichung der Korrelationswerte, s. Abbildung 9.10.

```
library(DataExplorer)

mariokart %>%
  dplyr::select(start_pr, ship_pr, total_pr) %>%
  plot_correlation()
```

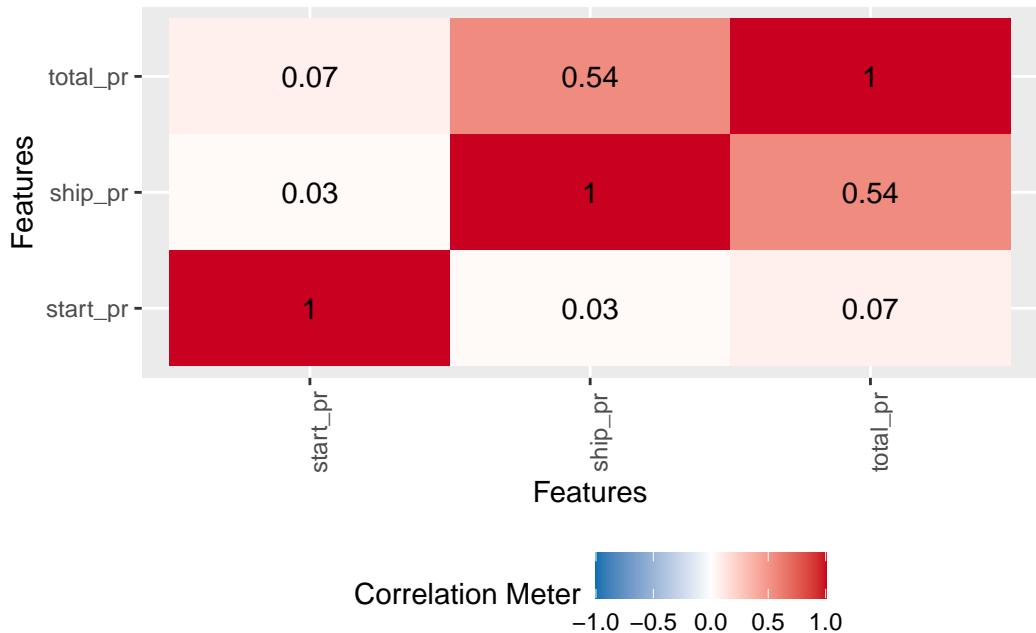


Abbildung 9.10.: Heatmap zu den Korrelationen im Datensatz `mariokart`.

## 9.7. Vertiefung

Dieser [TED-Vortrag](#) informiert zum Thema Scheinkorrelation. [Hier](#) finden Sie weitere Beispiele für Scheinkorrelationen.

## 9.8. Aufgaben

Schauen Sie sich auch mal auf der Webseite *Datenwerk*<sup>10</sup> die Aufgaben zu dem Tag `association` an.

1. [nasa02](#)
2. [mariokart-korr1](#)
3. [mariokart-korr2](#)
4. [mariokart-korr3](#)
5. [mariokart-korr4](#)
6. [korr01](#)
7. [korr02](#)

## 9.9. Halbzeitquiz

Testen Sie Ihr Wissen mit [diesem Quiz](#) zur deskriptiven Statistik (Maße der zentralen Tendenz, Variabilität, Verteilungsformen, Normalverteilung, Korrelation).<sup>11</sup>

## 9.10. Fallstudien

1. [YACSDA: EDA zu Flugverspätungen](#)<sup>12</sup>

**i** Hinweis

Einige der Fallstudien oder Übungsaufgaben können theoretische Inhalte (Konzepte der Statistik) oder praktische Inhalte (R-Befehle) enthalten, die Sie (noch) nicht kennen. In dem Fall: Einfach ignorieren. Oder Sie suchen nach einer Lösung anhand von Konzepten bzw. R-Befehlen, die Sie kennen.□

2. [YACSDA: Topgear](#)<sup>13</sup>
3. [Datensatz flights: Finde den Tag mit den meisten Abflügen](#)<sup>14</sup>
4. [Tidyverse Case Study: Exploring the Billboard Charts](#)<sup>15</sup>

**i** Hinweis

Bitte verstehen Sie die folgende Auswahl an Fallstudien als Auswahl. Es ist nicht nötig, dass Sie alle Fallstudien bearbeiten. Sehen Sie die Fallstudien eher als Angebot zur selektiven Vertiefung und Übung, dort, wo Sie es nötig haben.□

<sup>10</sup><https://datenwerk.netlify.app/>

<sup>11</sup><https://forms.gle/w7eTW3ftKy8Hv3nw8>

<sup>12</sup><https://datenwerk.netlify.app/posts/flights-yacsda-edu>

<sup>13</sup><https://data-se.netlify.app/2021/02/11/yacda-topgear/>

<sup>14</sup><https://data-se.netlify.app/2021/05/27/datensatz-flights-finde-den-tag-mit-den-meisten-abfl%C3%BCgen/>

<sup>15</sup><https://www.njtierney.com/post/2017/11/07/tidyverse-billboard/>

## **9.11. Literaturhinweise**

Auch die Korrelation ist ein Allzeit-Favorit in der Statistik; entsprechend wird Ihnen jedes typische Statistik-Buch die Grundlagen erläutern. Schauen Sie doch mal, was Ihre Bibliothek Ihnen zu bieten hat. Wer eine unorthodoxe (geometrische!) Herangehensweise an die Korrelation (und Regression) sucht, darf sich auf eine Menge Aha-Momente bei Kaplan (2009) freuen. Ein schönes, modernes Statistikbuch bietet der Psychologie-Prof Russel Poldrack von der Princeton University (2023); auch dieses Buch ist frei online verfügbar. Tipp: Nutzen Sie die Übersetzungsfunktion Ihres Browsers, wenn Sie das Buch nicht in Englisch lesen wollen. Ein Klassiker, wenn auch nicht mehr ganz frisch, ist Cohen u. a. (2003); immer noch sehr empfehlenswert, aber etwas höheren Anspruchs.

## **9.12. Literatur**

# 10. Geradenmodelle 1

## 10.1. Lernsteuerung

### 10.1.1. Standort im Lernpfad

Abb. Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 10.1.2. Lernziele

- Sie können ein Punktmodell von einem Geradenmodell begrifflich unterscheiden.
- Sie können die Bestandteile eines Geradenmodells aufzählen und erläutern.
- Sie können die Güte eines Geradenmodells anhand von Kennzahlen bestimmen.
- Sie können Geradenmodelle sowie ihre Modellgüte in R berechnen.

### 10.1.3. Benötigte R-Pakete

```
library(tidyverse)
library(easystats)
```

### 10.1.4. Benötigte Daten

Listing 8.1 definiert den Pfad zum Datensatz `mariokart` und importiert die zugehörige CSV-Datei in R, so dass wir einen Tibble mit Namen `mariokart` erhalten.

```
mariokart_path <- paste0(
  "https://vincentarelbundock.github.io/Rdatasets/",
  "csv/openintro/mariokart.csv")

mariokart <- read.csv(mariokart_path)
```

## 10.2. Vorhersagen

Vorhersagen sind eine nützliche Sache, unter (mindestens) folgenden Voraussetzungen:

1. Sie sind präzise
2. Wir kennen die Präzision
3. Jemand interessiert sich für die Vorhersage

Die Methode des Vorhersagens, die wir hier betrachten, nennt man auch *lineare Regression*.

### 10.2.1. Vorhersagen ohne Prädiktor

**Beispiel 10.1.** Nach intensiver Beschäftigung mit Statistik sind Sie allgemein als Checker bekannt. Viele jüngere Studenten fragen Sie um Rat. eines Tages kommt ein Studenten, Toni, und fragt: "Welche Statistiknote kann ich in der Klausur erwarten?" Sie entgegnen: "Wie viel hast du denn gelernt?". Die Antwort: "Sag ich nicht."

Nach kurzem Überlegen geben sie den Notenschnitt der letzten Klausur als Prognose für diese Person. Dazu rechnen Sie schnell den Notenschnitt (Mittelwert) aus.

Zuerst importieren Sie die Daten der letzten Klausur<sup>1</sup>:

```
noten2 <- read.csv("daten/noten2.csv")
```

Dann rechnen Sie den Mittelwert aus:

```
noten2 %>%
  summarise(mw = mean(y)) # y ist der Punktewert in der
  ↵ Klausur
```

| mw       |
|----------|
| 71.07968 |

Ihre Antwort lautet also: "Im Schnitt haben die Studis bei der letzten Klausur gut 70% der Punkte erzielt. Diesen Wert kannst du erwarten. Solange ich keine genaueren Infos habe, z.B. wieviel du gelernt hast, kann ich dir keine genauere Vorhersage machen, sorry!"□

---

<sup>1</sup>Diese Syntax wird bei Ihnen nur funktionieren, wenn auf *Ihrem Computer* dieser Ordner mit dieser Datei existiert. Andernfalls müssen Sie die Daten erst herunterladen: <https://raw.githubusercontent.com/sebastiansauer/statistik1/main/daten/noten.csv>.

### **i Hinweis**

Ohne Kenntnis eines Prädiktors (UV) (wie z.B. Lernzeit) ist der Mittelwert ein geeigneter Vorhersagewert für jede Beobachtung, s. Abbildung 10.1. Wir nutzen den Mittelwert als Punktmodell für den Klausurerfolg. □

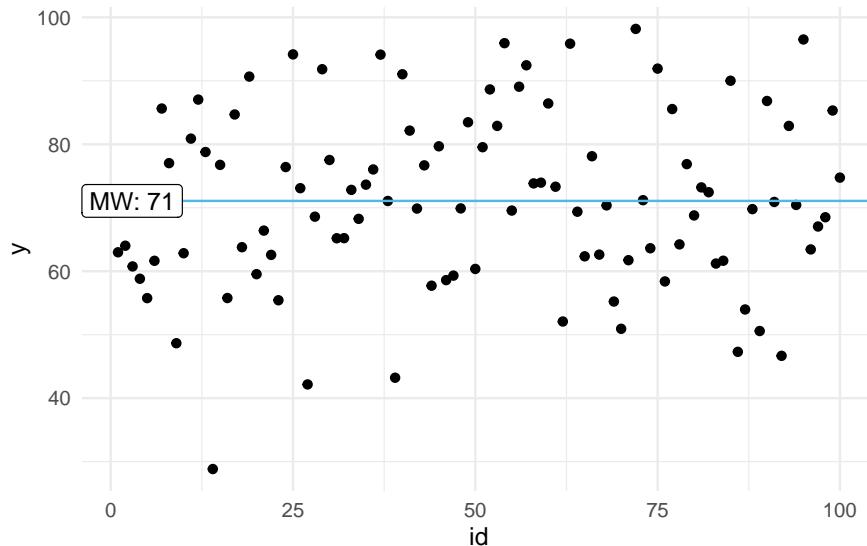


Abbildung 10.1.: Mittelwert als Vorhersagewert, bzw. Mittelwert als Punktmodell

### **10.2.2. Nullmodell (Punktmodell)**

Modelle ohne Prädiktor, Punktmodelle also, kann man so bezeichnen:  $y \sim 1$ . Da das Modell null Prädiktoren hat, nennt man es auch manchmal ‘Nullmodell’.

Auf Risch kann man dieses Nullmodell so spezifizieren:

```
lm0 <- lm(y ~ 1, data = noten2)
lm0
##
## Call:
## lm(formula = y ~ 1, data = noten2)
##
## Coefficients:
## (Intercept)
##          71.08
```

`lm` steht für ‘lineares Modell’, die `1` sagt, dass es keine Prädiktoren gibt. In dem Fall wird der Mittelwert als Gerade verwendet. Der zurückgemeldete Koeffizient (`Intercept`) ist hier der Modell des Punktmodells. Da es ein Punktmodell ist, sagt es für alle Beobachtungen (hier Studentis) den gleichen Wert vorher.

Die Regressionsgleichung lautet demnach:  $y_{\text{pred}} = 71.08$ . In Worten: "Wir sagen für jede Beobachtung einen Wert von ca. 71 vorher".

### 10.2.3. Vorhersagen mit Prädiktor

**Beispiel 10.2** (Toni verrät die Lernzeit). Toni entschließt sich dann doch noch, die Lernzeit zu verraten: "Okay, also ich hab insgesamt 42 Stunden gelernt, insgesamt." Jetzt müssen Sie erstmal nachdenken: "Wie viele Klausurpunkte sag ich vorher, wenn Toni 42 Stunden gelernt hat?"

Sie visualisieren sich zur Hilfe die vorliegenden Daten, s. Abbildung 10.2, a)<sup>2</sup>

```
library(DataExplorer)
noten2 %>%
  plot_scatterplot(by = "y") # Y-Variable muss angegeben
  ↵ werden
```

Auf dieser Basis antworten Sie Toni: "Bei 42 Stunden Lernzeit solltest du so 46 Punkte bekommen. Könnte mit dem Bestehen eng werden." Toni ist nicht begeistert von Ihrer Prognose und zieht von dannen.□

Der Trend (im Sinne eines linearen Zusammenhangs) von *Lernzeit* und *Klausurpunkte* ist deutlich zu erkennen. Mit einem Lineal könnte man eine entsprechende Gerade in das Streudiagramm einzeichnen, s. Abbildung 10.2, b).

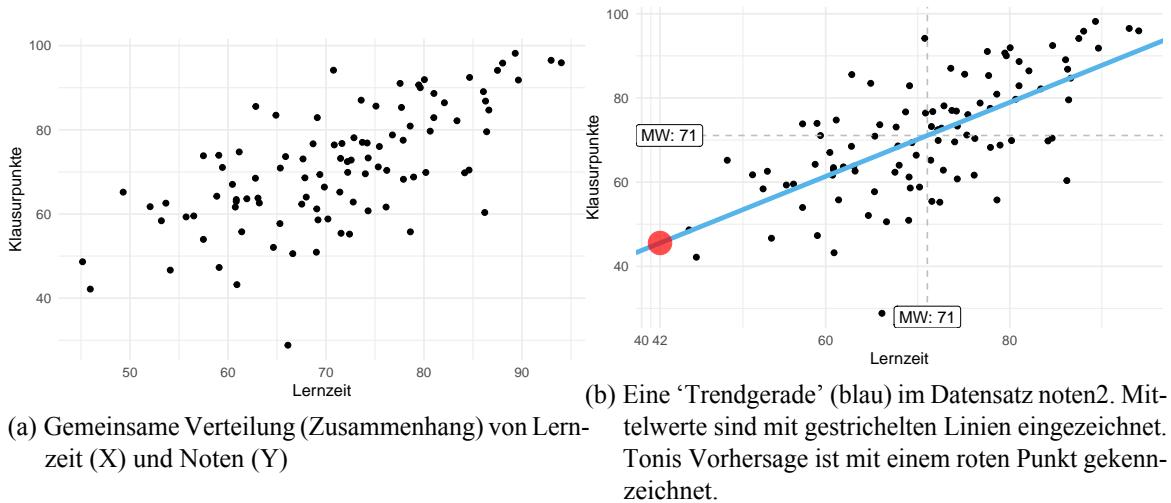


Abbildung 10.2.: Noten und Lernzeit: Rohdaten und Modell

Eine Gerade eignet sich, um einen linearen Trend zusammenzufassen.

<sup>2</sup>Die Daten stehen [hier](#) zum Download bereit.

## 10.3. Geradenmodelle

### 10.3.1. Achsenabschnitt und Steigung definieren eine Gerade

Wir verwenden eine Gerade als Modell für die Daten, s. Abbildung 10.2, rechts. Anders gesagt: Wir modellieren die Daten (bzw. deren Zusammenhang) mit einer Geraden.

Ein *Geradenmodell* ist eine Verallgemeinerung des Punktmodells: Ein Punktmodell sagt für alle Beobachtungen den gleichen Wert vorher. Abbildung 10.1 und Abbildung 10.2 stellen ein Punktmodell einem Geradenmodell gegenüber.

In einem Geradenmodell wird nicht mehr (notwendig) für jede Beobachtung die gleiche Vorhersage  $\hat{y}$  gemacht (wie das bei einem Punktmodell der Fall ist).

**Definition 10.1** (Gerade). Eine Gerade ist das, was man bekommt, wenn man eine lineare Funktion in ein Koordinatensystem einzeichnet. Man kann sie durch zwei *Koeffizienten* festlegen: Achsenabschnitt (engl. *intercept*), und Steigung (engl. *slope*). Häufig wird (z.B. im Schulunterricht) der Achsenabschnitt mit  $t$  und die Steigung mit  $m$  bezeichnet:  $f(x) = y = mx + t$ .

In der Statistik wird folgende Nomenklatur bevorzugt:  $f(x) = \hat{y} = \beta_0 + \beta_1 x$  oder  $f(x) = \hat{y} = b_0 + b_1 x$ .<sup>3</sup>

Das “Dach” über  $y$ ,  $\hat{y}$ , drückt aus, dass es sich den den geschätzten, bzw. vom Modell vorhergesagten (“modellierten”) Wert für  $y$  handelt, nicht das tatsächliche (empirische, beobachtete)  $y$ .  $\square$

Abbildung 10.3 skizziert die Elemente einer Regression.<sup>4</sup>

#### ! Das einfache lineare Modell

Das einfache lineare Modell nimmt den Wert einer abhängigen metrischen Variablen,  $y$  als lineare Funktion von unabhängigen Variablen,  $x$  an, plus einem Fehlerterm,  $e$ .  $\square$

$$y = f(x) + e$$
$$y_i = \beta_0 + \beta_1 \cdot x_i + \epsilon_i \quad \square$$

Mit:

- $\beta_0$ : geschätzter  $y$ -Achsenabschnitt laut Modell
- $\beta_1$ : geschätzte Steigung laut Modell
- $e$ : Fehler des Modells

Je nach Datenlage können sich Regressionsgerade in Steigung oder Achsenabschnitt unterscheiden, s. Abbildung 10.4.

<sup>3</sup>Die Nomenklatur mit  $b_0, b_1$  hat den Vorteil, dass man das Modell einfach erweitern kann:  $b_2, b_3, \dots$ . Anstelle von  $b$  liest man auch oft  $\beta$ . Griechische Buchstaben werden meist verwendet, um zu zeigen, dass man an einer Aussage über eine Population, nicht nur über eine Stichprobe, machen möchte.

<sup>4</sup>Bildquelle: Basierend auf einem Diagramm von Henri Menke, <https://texample.net/tikz/examples/linear-regression/>

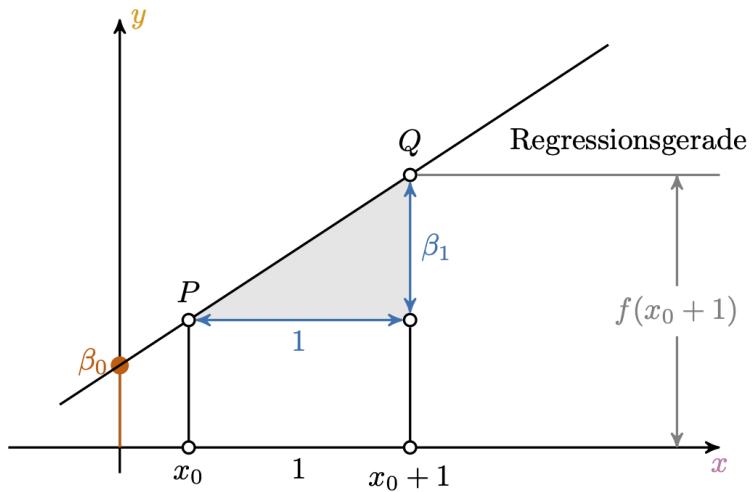


Abbildung 10.3.: Achsenabschnitt und Steigung einer Regressionsgeraden

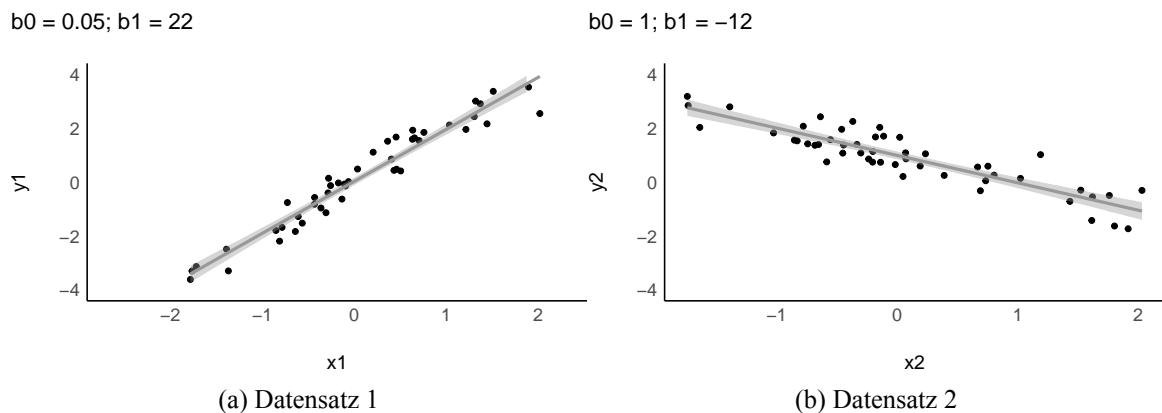


Abbildung 10.4.: Regressionsanalysen mit verschiedenen Koeffizienten, aber gleicher Modellgüte

**Beispiel 10.3** (Toni will es genau wissen). Da Toni Sie als Statistik-Profi abgespeichert hat, werden Sie wieder konsultiert. „Okay, ich hab noch zwei Fragen. Erstens: Wie viele Punkte bekomme ich, wenn ich gar nicht lerne? Zweitens, wie viele Punkte bekomme ich pro gelernte Stunde? Ist immerhin meine Lebenszeit, krieg ich nicht zurück!“

Das sind gute Fragen. Den  $Y$ -Wert (Klausurpunkte) bei  $X = 0$  gibt der Achsenabschnitt zurück. Schnell skizzieren Sie dazu ein Diagramm, s. Abbildung 10.5. Puh, die Antwort wird Toni nicht gefallen ...□

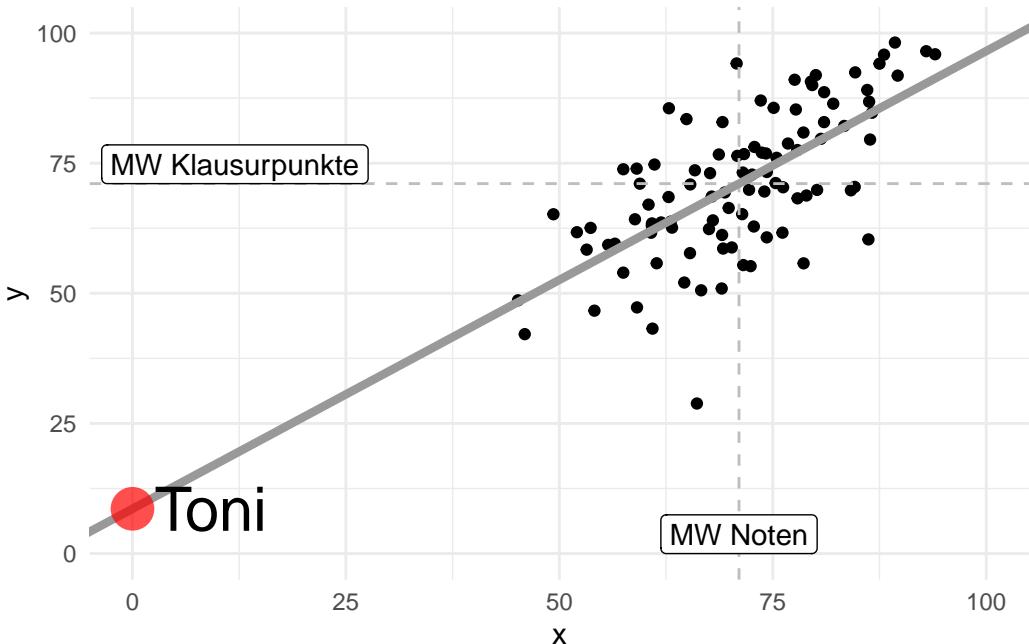


Abbildung 10.5.: Der Achsenabschnitt: Wie viele Punkte kann Toni erwarten bei 0 Lernstunden? (roter Punkt bei  $x=0$ )

Anstelle auf Abbildung 10.5 zu schauen, können Sie sich auch von R Tonis Klausurerfolg vorhersagen (to predict) lassen:

💡 Hey R, predicte mir mal auf Basis vom Modell “lm1” den Lernerfolg für Toni, wenn der x=0 Stunden lernt.

🔮 Okay, ich predicte mit Modell “lm1” und nehme als neue Datentabelle Tonis Lernzeit ( $x=0$ )!

```
tonis_lernzeit <- tibble(x = 0) # `tibble` erstellt eine
                                ↵ Tabelle

predict(lm1, newdata = tonis_lernzeit)
##           1
## 8.603032
```

### 10.3.2. Spezifikation eines Geradenmodells

Ein Geradenmodell kann man im einfachsten Fall so spezifizieren, s. Gleichung 10.2 :

$$\hat{y} \sim x \quad (10.1)$$

Lies: "Laut meinem Modell ist mein (geschätztes)  $\hat{y}$  irgendeine Funktion von  $x$ ".

Wir erinnern uns, dass  $Y$  die  $AV$  und  $X$  die  $UV$  ist:

$$AV \sim UV \quad (10.2)$$

Wir werden als Funktion (erstmal) nur Geraden verwenden. Die genauen Werte der Gerade lassen wir uns (erstmal) vom Computer ausrechnen.

Gleichung 10.2 können Sie so ins Errische übersetzen:

```
lm(y ~ x, data = meine_daten)
```

`lm` steht für "lineares Modell", also eine Gerade als Modell. Die Gerade nennt man auch *Regressionsgerade*<sup>5</sup>.

**Beispiel 10.4** (Zahlen für Toni). Toni ist nicht zufrieden mit Ihren Vorhersagen: "Jetzt hör mal auf mit deinem Lineal hier herum zu malen. Ich will es genau wissen, sag mir präzise Zahlen!".

```
lm1 <- lm(y ~ x, data = noten2)
lm1
##
## Call:
## lm(formula = y ~ x, data = noten2)
##
## Coefficients:
## (Intercept)          x
##       8.6030      0.8794
```

R gibt Ihnen die beiden Koeffizienten für die Gerade aus. Den Namen des Objekts können Sie frei aussuchen, z.B. `mein_erstes_lm`.

Die Regressionsgleichung lautet demnach:  $y_{pred} = 8.6 + 0.88*x$

$8.6$  ist der Achsenabschnitt, d.h. der Wert von  $Y$  wenn  $x = 0$ .  $0.88$  ist das Regressionsgewicht, d.h. die Steigung der Regressionsgeraden: Für jede Stunde Lernzeit steigt der vorhergesagte Klausurerfolg um  $0.88$  Punkte.

<sup>5</sup>an anderer Stelle in diesem Buch unscharf als "Trendgerade" bezeichnet.

Mit Kenntnis der beiden Koeffizienten kann man beliebige  $Y$ -Werte ausrechnen gegeben bestimmte  $X$ -Werte. Hat jemand zum Beispiel 10 Stunden gelernt, würden wir folgendes Klausurergebnis vorhersagen:

```
lernzeit <- 10  
y_pred <- 8.6 + 0.88*lernzeit  
y_pred  
## [1] 17.4
```

**Beispiel 10.5** (Vorhersage für Klausurerfolg, nächster Versuch). Sie versuchen, noch etwas Gutes für Toni zu tun. R hilft Ihnen dabei und rechnet die erwartete Punktzahl aus, wenn Toni 73 Stunden lernt. Sie dürfen es aber auch selber rechnen, wenn Ihnen das lieber ist.

```
tonis_lernzeit2 <- tibble(x = 73) # Der Befehl `tibble`  
#   erstellt eine Tabelle in R.
```

tonis\_lernzeit2 ist eine Tabelle mit einer Zeile und einer Spalte:

```
tonis_lernzeit2
```

| x  |
|----|
| 73 |

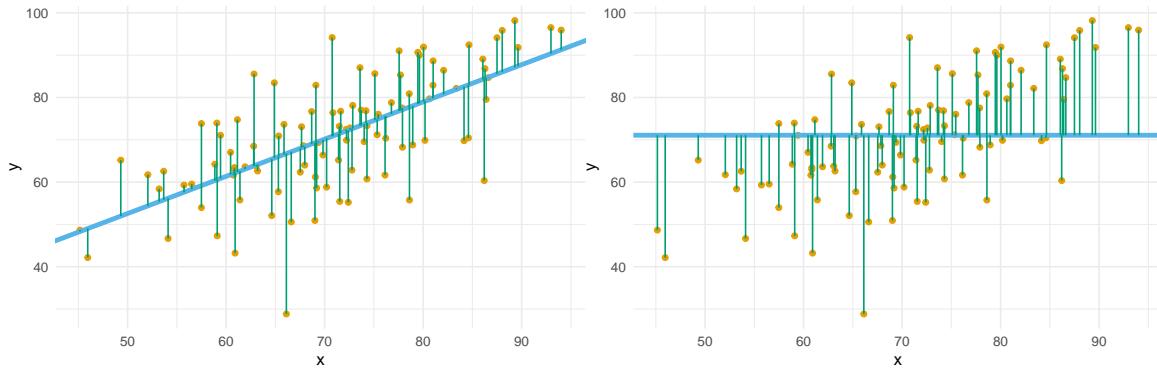
```
predict(lm1, newdata = tonis_lernzeit2)  
##       1  
## 72.7999
```

Die Syntax von `predict` lautet:

```
predict(name_des_objekts, newdata = tabelle_mit_prädiktorwerten)
```

### i Hinweis

Mit `predict` bekommt man eine Vorhersage; im Standard eine “Punkt-Vorhersage”, eine einzelne Zahl.□



(a) Residuen beim Geradenmodell (lm1)

(b) Residuen beim Punktmodell (lm0)

Abbildung 10.6.: Vorhersagefehler als Abweichungsbalken

### 10.3.3. Vorhersagefehler

Die Differenz zwischen vorhergesagten Wert für eine (neue) Beobachtung,  $\hat{y}_0$  und ihrem tatsächlichen Wert nennt man Vorhersagefehler (error,  $e_i$ ) oder *Residuum*:  $e_i = y_i - \hat{y}_i$ .

Wie ist es mit den Vorhersagefehlern von beiden Modellen bestellt?

Lassen wir uns von R die Streuung (Residuen) in Form der mittleren Absolutabweichung (MAE) ausgeben<sup>6</sup>:

```
mae(lm0)
## [1] 11.18385
mae(lm1)
## [1] 7.954085
```

Vergleichen wir MAE im Nullmodell mit MAE in lm1:

```
verhaeltnis_fehler_mae <- mae(lm1) / mae(lm0)
verhaeltnis_fehler_mae
## [1] 0.7112118
```

Ah! Das Geradenmodell ist viel besser: Von lm0 zu lm1 haben die mittlere (Absolut-)Länge des Fehlerbalkens auf 71 Prozent verbessert. Nicht schlecht!

**Definition 10.2** (Fehlerstreuung). Als Fehlerstreuung bezeichnen wir die Gesamtheit der Abweichungen der beobachteten Werte ( $y_i$ ) vom vorhergesagten Wert ( $\hat{y}_i$ ).  $\square$

Zur Berechnung der Fehlerstreuung gibt es mehrere Kenngrößen wie MAE oder MSE.

---

<sup>6</sup>aus dem Paket easystats

### **i Hinweis**

Ein Geradenmodell ist immer besser als ein Punktmodell (im Hinblick auf die Verringerung der Fehlerstreuung), solange X mit Y korreliert ist.<sup>7</sup>

Natürlich können wir - in Analogie zur Varianz - auch den mittleren Quadratfehlerbalken (Mean Squared Error, MSE) berechnen<sup>7</sup>.

```
mse(lm0)
## [1] 192.7863
mse(lm1)
## [1] 106.4519
```

```
verhaeltnis_fehler_mse <- mse(lm1)/mse(lm0)
verhaeltnis_fehler_mse
## [1] 0.5521755
```

#### **10.3.4. Berechnung der Modellkoeffizienten**

Aber wie legt man die Regressionsgerade in das Streudiagramm, bildlich gesprochen?

Die Regressionskoeffizienten<sup>8</sup> b0 und b1 wählt man so, dass die *Residuen minimal* sind, Genauer gesagt wird die Summe der quadrierten Residuen minimiert, s. Gleichung 10.3.

$$\min \sum_i e_i^2 \quad (10.3)$$

Es gibt verschiedene Möglichkeiten, um die Koeffizienten zu berechnen<sup>9</sup>. Eine schöne Darstellung dazu findet sich bei Kaplan (2009).

“Von Hand” können Sie die Optimierung von b0 und b1 in dieser App der FOM-Hochschule<sup>10</sup> ausprobieren.

### **10.4. R-Quadrat als Maß der Modellgüte**

Anders gesagt, wir haben uns um 1 – 0.55 verbessert:

<sup>7</sup>Wer mag, kann den MSE auch von Hand berechnen: `mean( (noten2$y-mean(noten2$y))^2 )`

<sup>8</sup>hier synonym: Modellparameter

<sup>9</sup>die sind aber nicht in diesem Buch zu finden

<sup>10</sup><https://fomshinyapps.shinyapps.io/KleinsteQuadrate/>

```
1 - verhaeltnis_fehler_mse
## [1] 0.4478245
```

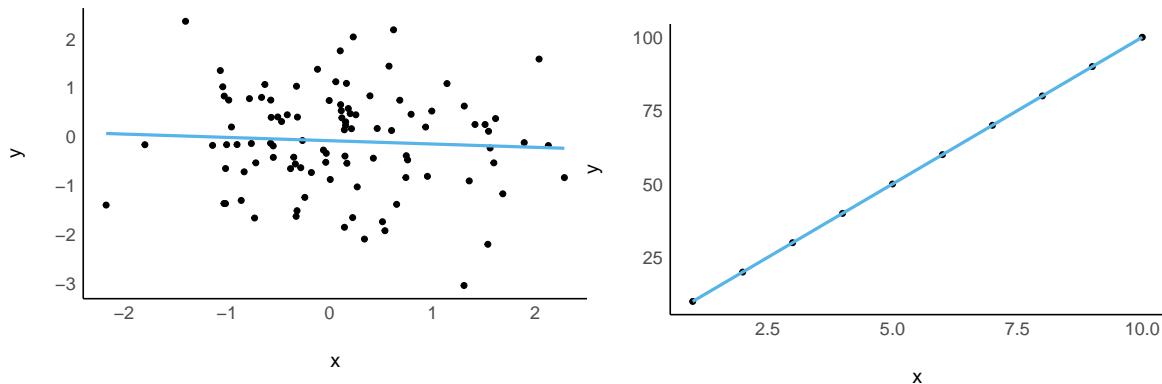
**Definition 10.3** (R-Quadrat). Die Verringerung (als Anteil) der Fehlerstreuung der Zielvariablen von  $m_0$  zum gerade untersuchten Modell nennt man *R-Quadrat* ( $R^2$ ). R-Quadrat ( $R^2$ ) eines Modells  $m$  ist definiert als die Verringerung der Streuung, wenn man das Modell  $m$  mit dem Nullmodell  $m_0$  vergleicht:  $R^2 = 1 - \frac{\text{MSE}_m}{\text{MSE}_{m_0}}$ . R-Quadrat ist ein Maß der *Modellgüte*: Je größer  $R^2$ , desto besser die Vorhersage. Da es ein Anteilsmaß<sup>11</sup> ist, liegt der Wertebereich zwischen 0 und 1. Im Nullmodell liegt R-Quadrat per Definition bei 0. Im Fall von Modellen des Typs  $y \sim x$  gilt:  $R^2 = r_{xy}^2$ .  $\square$

Einfach gesagt:  $R^2$  gibt an, wie gut (zu welchem Anteil) ein Modell die Zielvariable erklärt.

Wir können R-Quadrat ( $R^2$ ) uns von R z.B. so ausgeben lassen:

```
r2(lm1)
## # R2 for Linear Regression
##      R2: 0.448
## adj. R2: 0.442
```

Bei einer perfekten Korrelation ist  $r = 1$ , daher ist dann auch  $R^2 = 1$ <sup>12</sup>, s. Abbildung 10.7.



- (a) Keine Korrelation,  $r \approx 0$  und  $R^2 \approx 0$ . Prognose durch Mittelwert; die Regressionsgerade ist (ungefähr) parallel zur X-Achse
- (b) Perfekte Korrelation,  $r = 1$  und  $R^2 = 1$ . Prognose gleich beobachtetem Wert

Abbildung 10.7.: Extremfälle von R-Quadrat: 0 und 1

Bei einer perfekten Korrelation  $R^2 = 1$  liegen die Punkte auf der Geraden. Im gegenteiligen Extremfall von  $R^2 = 0$  ist die Vorhersage genauso gut, wie wenn man für jedes  $y$  den Mittelwert,  $\bar{y}$ , vorhersagen würde.

<sup>11</sup>Prozentzahl

<sup>12</sup>Bei Modellen mit einem Prädiktor; gibt es mehrere Prädiktoren gilt die Beziehung nur wenn die Prädiktoren alle paarweise unabhängig sind.

### **i Hinweis**

Je größer R-Quadrat, desto besser erklärt das Modell die Daten (desto besser der “Fit”, sagt man).

Diese [App der FOM-Hochschule](#) erlaubt es Ihnen mit der Größe der Residuen eines linearen Modells zu spielen.

## **10.5. Interpretation eines Regressionsmodells**

### **10.5.1. Modellgüte**

Die Residuen (Vorhersagefehler) bestimmen die Modellgüte: Sind die Residuen im Schnitt groß, so ist die Modellgüte gering (schlecht), und umgekehrt. Verschiedenen Koeffizienten stehen zur Verfügung: R-Quadrat,  $r^{13}$ , MSE, RMSE, MAE, ...

### **10.5.2. Koeffizienten**

Die Modellkoeffizienten, also Achsenabschnitt ( $\beta_0^{14}$ ) und Steigung ( $\text{beta}_1$ ) sind nur eingeschränkt zu interpretieren, wenn man die zugrundeliegenden kausalen Abhängigkeiten nicht kennt. Nur aufgrund eines statistischen Zusammenhangs darf man keine kausalen Abhängigkeiten annehmen. Ohne eine guten Grund für eine Kausalbehauptung kann man nur *deskriptiv* argumentieren. Oder sich mit der Modellgüte und den Vorhersagen begnügen. Was auch was wert ist.

#### **10.5.2.1. Achsenabschnitt (b0)**

“Im Modell `lm1` liegt der Achsenabschnitt bei  $y = 8.6$ . Beobachtungen mit  $x = 0$  können also diesen  $Y$ -Wert erwarten.” Leider ist es häufig so, dass Prädiktorwerte von 0 in der Praxis nicht realistisch sind, so dass der Achsenabschnitt dann wenig nützt.

**Beispiel 10.6** (Regression Größe und Gewicht). Nutzt man Körpergröße und das Gewicht von Menschen vorherzusagen, ist der Achsenabschnitt von Körpergröße wenig nützlich, da es keine Menschen gibt der Größe 0.□

#### **10.5.2.2. Geradensteigung (b1)**

“Im Modell `lm1` beträgt der Regressionskoeffizient  $b1$  0.88. Zwei Studenti, deren Lernzeit sich um eine Stunde unterscheidet, unterscheiden sich *laut Modell* um den Wert von  $b1$ .”

<sup>13</sup>als Korrelation von tatsächlichem  $y$  und vorhergesagten  $\hat{y}$

<sup>14</sup>lies: “beta Null”

### 🔥 Vorsicht

Häufig liest man, der “Effekt des Prädiktors” auf die AV betrage z.B. 0.88. “Effekt” ist aber ein Wort, dass man kausal verstehen kann. Ohne weitere Absicherung kann man aber Regressionskoeffizienten nicht kausal verstehen. Daher sollte man das Wort “Effekt” mit Vorsicht genießen. Manche sprechen daher auch von einem “statistischen Effekt”.  $\square$ .

## 10.6. Wie man mit Statistik lügt

Der Unterschied in Modellgüte zwischen, sagen wir,  $r = .1$  und  $r = .2$  ist *viel kleiner* als zwischen  $r = .7$  und  $r = .8$ .  $R^2$  ist ein (lineares) Maß der Modellgüte und da  $r = \sqrt{R^2}$ , darf  $r$  nicht wie  $R^2$  als Maß der Modellgüte interpretiert werden. Abbildung 10.8 zeigt den Zusammenhang von  $r$  und  $R^2$ .

Der Zusammenhang von  $r$  und  $R$ -Quadrat ist nicht linear.

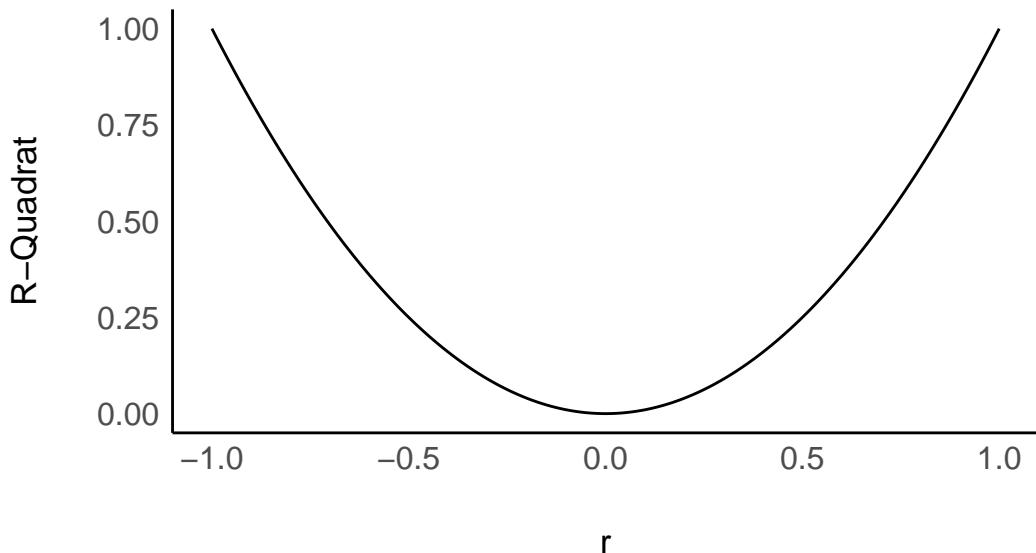


Abbildung 10.8.: Zusammenhang von  $r$  und  $R$ -Quadrat

### 🔥 Vorsicht

Unterschiede zwischen Korrelationsdifferenzen dürfen nicht linear interpretiert werden.  $\square$

## 10.7. Fallbeispiel Mariokart

### 10.7.1. Der Datenwahrsager legt los

Als mittlerweile anerkannter Extrem-Datenanalyst in dem Online-Auktionshaus, in dem Sie arbeiten, haben Sie sich neue Ziele gesetzt. Sie möchten eine genaue Vorhersage von Verkaufspreisen erzielen. Als Sie von diesem Plan berichteten, leuchteten die Augen Ihrer Chefin. Genaue Vorhersagen, das ist etwas von hoher betriebswirtschaftlicher Relevanz. Auf geht's!

Daten laden:<sup>15</sup>

```
mariokart <- read.csv(mariokart_path)
```

```
lm2 <- lm(total_pr ~ start_pr, data = mariokart)
r2(lm2)
## # R2 for Linear Regression
##      R2: 0.005
## adj. R2: -0.002
```

Oh nein! Unterirdisch schlecht. Anstelle von bloßem Rumprobieren überlegen Sie und schauen dann in **?@fig-mario-corr** nach, welche Variable am stärksten korreliert mit `total_pr`; es resultiert `lm3`:

```
lm3 <- lm(total_pr ~ ship_pr, data = mariokart)
parameters(lm3)
```

Tabelle 10.3.: Modellparameter von `lm3`

| Parameter   | Coefficient | SE   | 95% CI         | t(141) | p     |
|-------------|-------------|------|----------------|--------|-------|
| (Intercept) | 36.25       | 2.54 | (31.23, 41.26) | 14.28  | <.001 |
| ship pr     | 4.34        | 0.57 | (3.22, 5.46)   | 7.67   | <.001 |

Der Achsenabschnitt liegt bei ca. 36 Euro, wie man in Tabelle 10.3 sieht: Ein Spiel, das mit Null Euro Preis startet, kann laut `lm3` etwa 36 Euro finaler Verkaufspreis erwarten. *Pro Euro an Versandkosten (ship\_pr)* steigt der zu erwartende finale Verkaufspreis um ca. 4 Euro.<sup>16</sup>.

Die Regressionsgleichung von `lm3` lautet demnach:

`total_pr_pred = 36.25 + 4.34*ship_pr.`

In Worten:

<sup>15</sup>Und die üblichen Pakete starten, nicht vergessen.

<sup>16</sup>Die Spalte 95 CI gibt einen Schätzbereich für den jeweiligen Modellkoeffizienten an, denn es handelt sich bei den Koeffizienten um Schätzwerte; der wahre Wert in der Population ist unbekannt. Wir kennen schließlich nur eine Stichprobe der Größe  $n = 143$ .

Der vorhergesagte Gesamtpreis eines Spiels liegt bei 36.25€ “Sockelbetrag” plus 4.34 mal die Versandkosten.

### 10.7.2. Vertiefung

Man kann sich die erwarteten Werte (“expectations”) des Verkaufspreises in Abhängigkeit vom Wert der UV (`ship_pr`) auch schätzen (“to estimate”) lassen, und zwar so<sup>17</sup>:

```
estimate_expectation(lm3) %>% head() # nur die ersten paar
← vorhergesagten Werte
```

Tabelle 10.4.: Model-based Expectation

| ship_pr | Predicted | SE   | 95% CI         | Residuals |
|---------|-----------|------|----------------|-----------|
| 4.00    | 53.59     | 1.87 | (49.89, 57.30) | -2.04     |
| 3.99    | 53.55     | 1.87 | (49.85, 57.25) | -16.51    |
| 3.50    | 51.43     | 1.82 | (47.82, 55.03) | -5.93     |
| 0.00    | 36.25     | 2.54 | (31.23, 41.26) | 7.75      |
| 0.00    | 36.25     | 2.54 | (31.23, 41.26) | 34.75     |
| 4.00    | 53.59     | 1.87 | (49.89, 57.30) | -8.59     |

Variable predicted: `total_pr`

Ah, bei 4 Euro Versandkosten ist laut dem Modell knapp 54 Euro Verkaufspreis zu erwarten, fassen Sie sich die Ausgabe zusammen.

💡 Das sieht man in der Spalte `Predicted`, dort steht der vorhergesagte Wert für `total_pr` für einen bestimmten Wert von `ship_pr`.

💡 Kann ich auch `predict` benutzen? Ich würde gerne den Verkaufspreis wissen, wenn die Versandkosten bei 1 und bei 4 Euro liegen.

💡 Ja, klar!

```
neue_daten <- tibble(
  ship_pr = c(1, 4)) # zwei Werte zum Vorhersagen

predict(lm3, newdata = neue_daten)
##          1         2
## 40.58276 53.59442
```

<sup>17</sup>Die Funktion stammt aus easystats

Aber nützlich wäre noch, das Modell (bzw. die Schätzung der erwarteten Werte) als Diagramm zu bekommen. Das erreicht man z.B. so, s. Abbildung 11.8.

```
estimate_expectation(lm3) %>% plot()
```

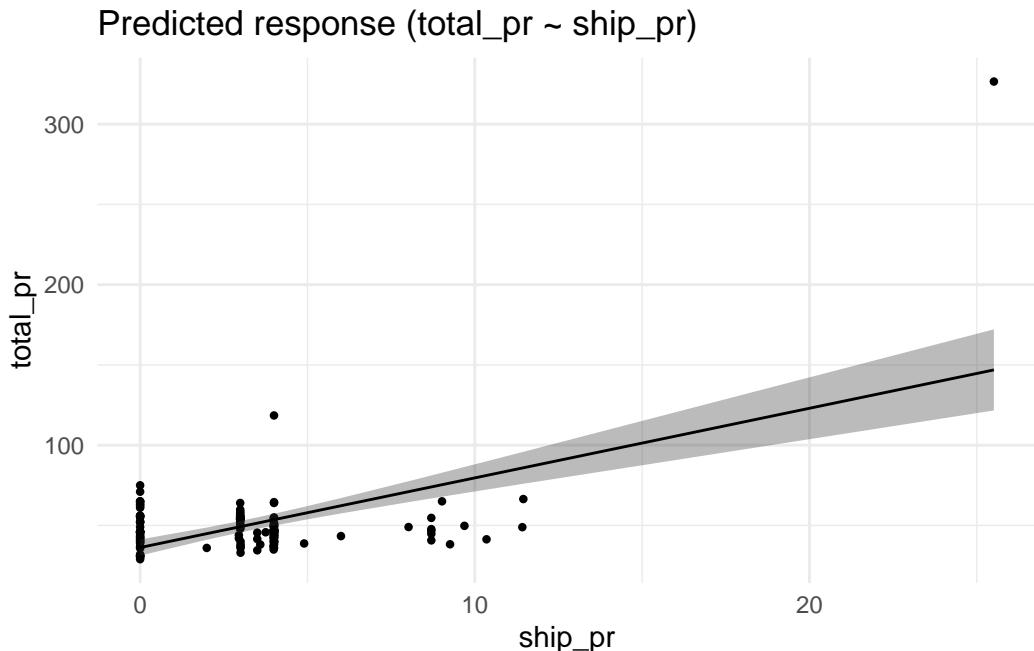


Abbildung 10.9.: Verbildlichung der erwarteten Werte laut lm3

`estimate_expectation` heißt sinngemäß “schätzt den zu erwartenden Wert”. Kurz gesagt: Wir wollen eine Vorhersage von R.

Am wichtigsten ist Ihnen aber im Moment die Frage, wie “gut” das Modell ist, spricht wie lang oder kurz die (absoluten) Vorhersagefehler-Balken sind:

```
mae(lm3)
## [1] 13.0632
```

Das Modell erklärt einen Anteil von ca. 0.29 der Gesamtstreuung.

```
r2(lm3)
## # R2 for Linear Regression
##      R2: 0.294
## adj. R2: 0.289
```

```
mae (lm3)
## [1] 13.0632
```

Im nächsten Meeting erzählen Sie Ihrem Chef “Ich kann den Verkaufspreis von Mariokart-Spielen im Schnitt auf 13 Dollar genau vorhersagen!”. Hört sich gut an. Allerdings hätte ihr Chef es gerne genauer. Kann man da noch was machen?

## 10.8. Fallstudie Immobilienpreise



### Vorsicht

Diese Fallstudie stellt die Prüfungsleistung “Prognosewettbewerb” einführend dar. Es empfiehlt sich für Sie, diese Fallstudie sorgsam zu bearbeiten.□

### 10.8.1. Hintergrund

In dieser Fallstudie geht es darum, die Preise von Immobilien vorherzusagen. Kurz gesagt: Sagen Sie die Hauspreise vorher, und reichen Sie Ihre Vorhersagen als CSV bei [Kaggle](#) ein.

Kaggle ist eine Webseite, die Prognosewettbewerbe veranstaltet.

In dieser Fallstudie nehmen Sie teil an der Kaggle-Competition [Ames House Prices](#).<sup>18</sup>

- [Beschreibung](#)<sup>19</sup>
- [Ziel/Aufgabe](#)<sup>20</sup>
- [Spielregeln](#)<sup>21</sup>

### 10.8.2. Benötigte R-Pakete

```
library(tidyverse)
library(easystats)
```

<sup>18</sup><https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview>

<sup>19</sup><https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview/description>

<sup>20</sup><https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview/evaluation>

<sup>21</sup><https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/rules>

### 10.8.3. Daten

Wenn Sie sich nicht bei Kaggle einloggen möchten, können Sie die Daten von Kaggle herunterladen und zwar [hier](#).

Im Einzelnen müssen Sie folgende Dateien herunterladen:

- *Data\_description.txt*: Code book, d.h. Beschreibung der Variablen im Datensatz
- *train.csv*: Daten von Häusern, die Sie nutzen, um Modelle zu erstellen
- *test.csv*: Daten von Häusern, von denen Sie den Kaufpreis vorhersagen sollen
- *sample\_submission.csv*: Beispielhafte Prognosedatei, die Datei also, mit der Sie Ihre Vorhersagen einreichen

Sie können auch so auf die Daten zugreifen:

```
d_train_path_online <-
  "https://raw.githubusercontent.com/sebastiansauer/Lehre/main/data/ames"
d_test_path_online <-
  "https://raw.githubusercontent.com/sebastiansauer/Lehre/main/data/ames"

d_train <- read.csv(d_train_path_online)
d_test <- read.csv(d_test_path_online)
```

Laden Sie diese Daten am besten herunter und speichern Sie sie in einem passenden Unterverzeichnis (Ihres Projektordners in RStudio) ab.

Das Code Book können Sie [hier einsehen und herunterladen](#).<sup>22</sup>

### 10.8.4. Prognosedatei

Die Prognosedatei ist die Datei, die Ihre Vorhersagen (Prognosen) enthält. Sie soll prinzipiell so aussehen wie in Tabelle 10.5 dargestellt.

Tabelle 10.5.: Beispiel den Aufbau der Prognose-Datei

|  | id   | SalePrice |
|--|------|-----------|
|  | 1461 | 169277.1  |
|  | 1462 | 187758.4  |
|  | 1463 | 183583.7  |

Die Prognosedatei besteht also aus zwei Spalten: der Spalte `id` und der Spalte `Saleprice`. Die Spalte `id` gibt an, welches Haus in einer bestimmten Zeile Ihrer Prognosedatei gemeint ist - für

<sup>22</sup><[https://github.com/sebastiansauer/Lehre/blob/main/data/ames-kaggle/data\\_description.txt](https://github.com/sebastiansauer/Lehre/blob/main/data/ames-kaggle/data_description.txt)>

welches Haus Sie also gerade einen Kaufpreis vorhersagen. die Spalte SalePrice ist Ihre Vorhersage für den Kaufpreis des Hauses mit der Id, die in der betreffenden Zeile steht. Insgesamt soll die Prognosedatei genau so viele Zeilen haben wie der Test-Datensatz, also die Tabelle, die die vorherzusagenden Werte angibt.

Alles klar?

Los geht's!

#### 10.8.5. Daten importieren von der Festplatte

Wir können die Daten auch von der Festplatte importieren; oft müssen wir das auch - wenn die Daten nämlich nicht öffentlich zugreifbar auf einem Server liegen.

```
d_train_path <- "daten/ames-kaggle/train.csv"  
d_test_path <- "daten/ames-kaggle/test.csv"  
d_train <- read.csv(d_train_path)  
d_test <- read.csv(d_test_path)
```

##### Hinweis

In diesem Beispiel gehen wir davon aus, dass die Dateien `train.csv` und `test.csv` in einem Unterordner namens `daten/ames-kaggle` liegen. Sie müssen sie dort abspeichern. Dieser Ordner muss ein Unterordner Ihres aktuellen R-Projekts sein.□

##### Vorsicht

Wenn das Importieren von der Festplatte nicht klappt ... Es ist hilfreich, wenn man Daten von der eigenen Festplatte importieren kann. Aber fürs Erste können Sie die Daten auch von oben angegeben Online-Pfad importieren.□

#### 10.8.6. Ein erster Blick in die Daten

Schauen wir uns einmal die Verteilung der metrischen Variablen an, `?@tbl-ames1`.

```
describe_distribution(d_train)
```

## 10.8.7. Ein erstes Vorhersagemodell

### 10.8.7.1. Welche Variablen eignen sich zur Vorhersage?

Eine einfache Antwort auf die Frage, welche Variablen sich zur Vorhersage eignen, ist, die Korrelation aller Prädiktoren mit der abhängigen Variablen<sup>23</sup> zu berechnen, s. Tabelle ?? und Listing 10.1.

---

**Listing 10.1** Welche Variablen korrelieren stärker als .3?

---

```
d_train %>%
  select(-Id) %>%
  correlation() %>% # berechne Korrelationen
  filter(Parameter2 == "SalePrice") %>% # aber nur, wo die
  ↵ zweite Variable "SalesPrice" ist
  arrange(-abs(r)) %>% # sortiere absteigend nach der Höhe
  ↵ des Korrelationskoeffizienten r
  filter(abs(r) > .3) # nur |r| > .3
```

---

Aha! Ein Menge Information.<sup>24</sup>

Diese Variablen sind einigermaßen stark mit unserer Zielvariablen SalePrice korreliert. Nutzen wir also diese Variablen (oder einige von ihnen) zur Vorhersage.

### 10.8.7.2. Modell 1

Im ersten Modell gehen wir davon aus, dass der Verkaufspreis im Großen und Ganzen durch den Zustand der Immobilie (OverallQual) vorhergesagt werden kann. Diese Variable ist am stärksten mit der Zielvariable korreliert und ist daher ein guter Kandidat für die Vorhersage.

```
m1 <- lm(SalePrice ~ OverallQual, data = d_train)
parameters(m1) # aus easystats
```

| Parameter   | Coefficient | SE      | 95% CI                 | t(1458) | p      |
|-------------|-------------|---------|------------------------|---------|--------|
| (Intercept) | -96206.08   | 5756.41 | (-1.07e+05, -84914.35) | -16.71  | < .001 |
| OverallQual | 45435.80    | 920.43  | (43630.29, 47241.31)   | 49.36   | < .001 |

Wie gut ist das Modell?

<sup>23</sup>die vorherzusagende Variable, auch Ziel- oder Outcome-Variable genannt

<sup>24</sup>Wenn Sie Teile der Ausgabe der Tabelle nicht verstehen: Im Zweifel einfach ignorieren. Wenn Sie die R-Syntax nicht verstehen: Führen Sie die Syntax schrittweise aus. Zuerst d\_train ausführen und das Ergebnis betrachten. Dann d\_train %>% select(-Id) ausführen, wieder die Ausgabe betrachten, usw.

```
rmse(m1) # aus easystats
## [1] 48589.45
```

Im Schnitt liegen wir  $4.54 \times 10^4$  Dollar daneben. Ob das viel oder weniger ist, wird sich im Vergleich mit anderen Modellen zeigen.

R-Quadrat liefert einen anderen Blick auf die Modellgüte:

```
r2(m1) # aus easystats
## # R2 for Linear Regression
##      R2: 0.626
## adj. R2: 0.625
```

#### 10.8.7.3. Model 2

Berechnen wir als nächstes ein Modell mit mehreren UV, m2.

##### **i** Hinweis

Mann kann mehrere UV (Prädiktorvariablen) in ein Regressionsmodell aufnehmen. Dazu trennt man sie mit einem Pluszeichen in `lm()`:

```
mein_modell <- lm(av ~ uv1 + uv2 + ... + uv_n, data =
  ↵ meine_daten)
```

Dabei ist das Pluszeichen kein arithmetischer Operator, sondern sagt nur “als UV nimm UV1 und UV2 und ...”. □

```
m2 <- lm(SalePrice ~ OverallQual + GrLivArea + GarageCars,
  ↵ data = d_train)
parameters(m2)
```

Tabelle 10.7 zeigt die Koeffizienten von m2.

Tabelle 10.7.: Modellparameter von m1

| Parameter   | Coefficient | SE      | 95% CI                 | t(1456) | p      |
|-------------|-------------|---------|------------------------|---------|--------|
| (Intercept) | -98832.49   | 4842.90 | (-1.08e+05, -89332.69) | -20.41  | < .001 |
| OverallQual | 27104.83    | 1072.18 | (25001.64, 29208.01)   | 25.28   | < .001 |
| GrLivArea   | 50.67       | 2.55    | (45.67, 55.68)         | 19.86   | < .001 |
| GarageCars  | 21298.96    | 1807.06 | (17754.23, 24843.69)   | 11.79   | < .001 |

Wie gut sind die Vorhersagen des Modells m2 für die Daten von `d_train`?

```
rmse (m2)
## [1] 40566.42
```

Im Schnitt liegen unsere Vorhersagen  $2.71 \times 10^4$  Dollar daneben. Ist das gut?

Betrachten wir noch  $R^2$ :

```
r2 (m2)
## # R2 for Linear Regression
##      R2: 0.739
## adj. R2: 0.739
```

#### i Hinweis

Ob die Modellgüte (R-Quadrat, RMSE, etc.) “gut” oder “hoch” ist, beantwortet man am besten *relativ*, also im Vergleich zu anderen Modellen. □

#### 10.8.7.4. Nullmodell

Zum Vergleich berechnen wir das maximal einfache Modell: ohne Prädiktoren. Man nennt es das “Nullmodell”. In diesem Modell sagen wir für jedes Haus einfach den mittleren Preis aller Häuser vorher.

```
m0 <- lm(SalePrice ~ 1, data = d_train)
```

Wie gut ist die Vorhersage des Nullmodells?

```
rmse (m0)
## [1] 79415.29
```

Beim Nullmodell liegen wir ca. 80 Tausend Dollar daneben.

Das R-Quadrat der Nullmodells ist per Definition Null:

```
r2 (m0)
## # R2 for Linear Regression
##      R2: 0.000
## adj. R2: 0.000
```

### 10.8.8. Vorhersagen im Test-Datensatz mit m2

Wir haben jetzt unseren Champion, m2. Alle Hoffnung ruht auf diesem Modell. Ob die Vorhersagen im Test-Sample präzise sein werden? Oder himmelweit daneben? Enttäusche uns nicht!

Hier sind die Vorhersagen:

```
m2_pred <- predict(m2, newdata = d_test)          (1)
head(m2_pred)                                     (2)
##           1           2           3           4           5           6
## 103394.7 152441.4 161837.8 187675.8 225467.0 190260.2
```

- ① predice anhand der Regressionsgerade von m1 und zwar anhand der Daten aus d\_test
- ② zeige den “Kopf” der Vorhersagen (m1\_pred), d.h. die ersten paar Vorhersagen

Die Vohersagen fügen wir jetzt dem Test-Sample hinzu:

```
d_test <-
  d_test %>%
  mutate(SalePrice = m2_pred)
```

### 10.8.9. Einreichen!

#### 10.8.9.1. Wir brauchen zwei Spalten: Id und SalePrice

So, wir haben unsere Vorhersagen! Jetzt reichen wir diese Vorhersagen ein.

Für die Prognosedatei (submission file) zum Einreichen brauchen wir nur die Spalten id und SalePrice:

```
m2_subm <-
  d_test %>%
  select(Id, SalePrice)
```

Kaggle möchte keine fehlenden Werten in den Vorhersagen, also prüfen wir das mal:

```
m2_subm %>%
  drop_na() %>%
  nrow()
## [1] 1458
```

- ① Lass alle Zeilen mit NAs (fehlenden Werten in irgendeiner Spalte) fallen, filtere diese Zeilen also raus
- ② zähle die Anzahl der Zeilen (die noch verbleiben)

Die Anzahl der Zeilen, die wir hier erhalten, ist gleich zu den Anzahl der Zeilen von `d_test`. Es gibt also keine fehlenden Werte.

```
nrow(d_test)
## [1] 1459
```

### 10.8.9.2. Hochladen

Diesen Tibble speichern wir als CSV-Datei an geeigneter Stelle ab.<sup>25</sup>

```
write_csv(m1_subm, "daten/ames-kaggle/m1-subm.csv")
```

Und dann laden Sie diese Datei, `m1_subm.csv` bei Kaggle hoch und hoffen auf einen Hauptgewinn.

Das Modell erzielte einen Score von *0.55521*.

### 10.8.10. Fazit

Diese Fallstudie hat ein einfaches Prognosemodell vorgestellt. Sicherlich gibt es viele Ansätze, dieses Modell zu verbessern.

Hier sind einige Fragen, die Sie sich dazu stellen können:

- Welche Prädiktoren sollte ich in das Modell aufnehmen?
- Wie gehe ich mit fehlenden Werten um?
- Wenn ein Prädiktor schief ist, sollte ich ihn dann log-transformieren?
- Vielleicht sollte man manche Prädiktoren quadrieren?
- Wie gehe ich mit nominalskalierten Variablen um, wenn diese viele Stufen haben?
- ...

Viel Spielraum für Ihre Kreativität!

## 10.9. Aufgaben

Eine Aufgabe, die eine Einführung zum [Kaggle-Wettbewerb Ames House Prices](#) bietet<sup>26</sup>, finden Sie im Datenwerk.<sup>27</sup>

Die Webseite [datenwerk.netlify.app](#) stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

<sup>25</sup>Es bietet sich an `write_csv` zu verwenden, da `write.csv` automatisch (ungefragt) noch eine Id-Spalte ohne Namen einfügt (mit den Zeilennummern), das mag aber Kaggle nicht. Kaggle erwartet exakt zwei Spalten und zwar mit den Namen `Id` und `SalePrice`.

<sup>26</sup><https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview>

<sup>27</sup><https://datenwerk.netlify.app/posts/ames-kaggle1/ames-kaggle1.html>

- Aussagen-einfache-Regr
- interpret-koeff-lm
- korrr-als-regr
- Linearitaet1a
- lm1
- mtcars-regr01
- nichtlineare-regr1
- penguins-regr02
- regression1
- regression1b
- Regression3
- Regression4
- Regression5
- Regression6

Schauen Sie sich die Aufgaben beim [Datenwerk](#) an, vor allem die Tags [regression](#) und [lm](#).

*Nicht alle Aufgaben* aus dieser Sammlung passen zum Stoff; vielleicht können Sie einige Aufgaben nicht lösen. Ignorieren Sie einfach diese Aufgaben.

Beachten Sie die [Hinweise zu den Aufgaben](#).<sup>28</sup>

## 10.10. Literaturhinweise

Gelman, Hill, und Vehtari (2021a) liefert eine deutlich umfassendere Einführung in die Regressionsanalyse als dieses Kapitel es tut. Eine moderne, R-orientierte Einführung in Statistik inklusive der Regressionsanalyse findet sich bei ([cetinkaya-rundel\\_introduction\\_2021-2?](#)). Ein Klassiker mit viel Aha-Potenzial ist ([cohen2003?](#)).

## 10.11. Literatur

---

<sup>28</sup><https://datenwerk.netlify.app/hinweise>

# 11. Geradenmodelle 2

## 11.1. Lernsteuerung

### 11.1.1. Standort im Lernpfad

Abb. Abbildung 1.4 zeigt den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 11.1.2. Lernziele

- Sie können Regressionsmodelle für Forschungsfragen mit binärer, nominaler und metrischer UV erläutern und in R anwenden.
- Sie können Interaktionseffekte in Regressionsmodellen erläutern und in R anwenden.
- Sie können den Anwendungszweck von Zentrieren und z-Transformationen zur besseren Interpretation von Regressionsmodellen erläutern und in R anwenden.
- Sie können Modelle nutzen, um Vorhersagen anhand neuer Daten zu erstellen.

### 11.1.3. Benötigte R-Pakete

```
library(tidyverse)
library(yardstick)  # für Modellgüte im Test-Sample
library(easystats)
library(ggpubr)    # Daten visualisieren
library(openintro) # dataset mariokart
```

### 11.1.4. Benötigte Daten

Listing 8.1 definiert den Pfad zum Datensatz `mariokart` und importiert die zugehörige CSV-Datei in R, so dass wir einen Tibble mit Namen `mariokart` erhalten.

```

mariokart_path <- paste0(
  "https://vincentarelbundock.github.io/Rdatasets/",
  "csv/openintro/mariokart.csv")
mariokart <- read.csv(mariokart_path)

wetter_path <- paste0(
  "https://raw.githubusercontent.com/sebastiansauer/",
  "Lehre/main/data/wetter-dwd/precip_temp_DWD.csv")
wetter <- read.csv(wetter_path)

```

Die Wetterdaten stammen vom [DWD](#).<sup>1</sup>

## 11.2. Forschungsbezug: Gläserne Kunden

Lineare Modelle<sup>2</sup> sind ein altes, aber mächtiges Werkzeug. Sie gehören immernoch zum Standard-Repertoire moderner Analystis.

**Beispiel 11.1** (Wie gut kann man Ihre Persönlichkeit auf Basis des Facebook-Profil vorhersagen?). In einer Studie mit viel Medienresonanz untersuchten ([Kosinski2013?](#)), wie gut Persönlichkeitszüge durch Facebook-Daten (Likes etc.) vorhergesagt werden können. Die Autoren resümieren:

We show that easily accessible digital records of behavior, Facebook Likes, can be used to automatically and accurately predict a range of highly sensitive personal attributes including: sexual orientation, ethnicity, religious and political views, personality traits, intelligence, happiness, use of addictive substances, parental separation, age, and gender.

Die Autoren berichten über hohe Modellgüte ( $r$ ) zwischen den tatsächlichen persönlichen Attributen und den vorhergesagten Werten Ihres Modells, s. Abbildung 11.1. Das eingesetzte statistische Modell beruht auf einem linearen Modell, also ähnlich zu dem in diesem Kapitel vorgestellten Methoden.

Neben der analytischen Stärke der Regressionsanalyse zeigt das Beispiel auch, wie gläsern Konsument:innen im Internet sind.□

## 11.3. Wetter in Deutschland

**Beispiel 11.2** (Wetterdaten). Nachdem Sie einige Zeit als Datenanalyst bei dem Online-Auktionshaus gearbeitet haben, stand Ihnen der Sinn nach etwas Abwechslung. Viel Geld verdienen und Ruhm und Anerkennung sind ja schon ganz nett, aber dann fiel Ihnen ein, dass Sie ja zu Generation Z gehören, und daher den schnöden Mammon nicht so hoch schätzen sollten. Sie entschließen sich,

<sup>1</sup>Lizenzhinweis: Datenbasis: Deutscher Wetterdienst, eigene Elemente ergänzt.

<sup>2</sup>synonym: Regressionsanalysen

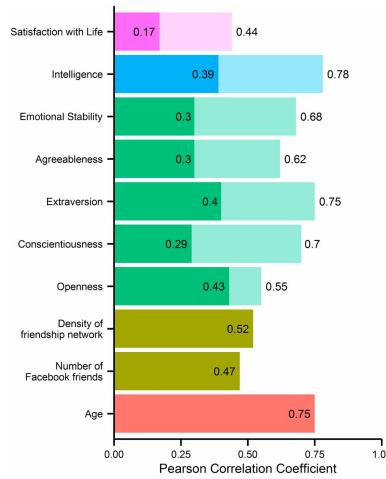


Abbildung 11.1.: Prediction accuracy of regression for numeric attributes and traits expressed by the Pearson correlation coefficient between predicted and actual attribute values

Ihre hochgeschätzten Analyse-Skills für etwas einzusetzen, das Ihnen sinnvoll erscheint: Die Analyse des Klimawandels.

Beim [Deutschen Wetterdienst, DWD](#) haben Sie sich Wetterdaten von Deutschland heruntergeladen. Nach etwas [Datenjudo, auf das wir hier nicht eingehen wollen](#) resultiert ein schöner Datensatz, den Sie jetzt analysieren wollen<sup>3</sup>:

```
wetter_path <- paste0(
  "https://raw.githubusercontent.com/sebastiansauer/",
  "Lehre/main/data/wetter-dwd/precip_temp_DWD.csv")

wetter <- read.csv(wetter_path)
```

Ein *Data-Dictionary* für den Datensatz können Sie [hier](#) herunterladen.<sup>4</sup>

#### **i** Hinweis

Ein *Data-Dictionary* (Codebook) erklärt einen Datensatz. Oft bedeutet das, dass für jede Spalte der Datentabelle erklärt wird, was die Spalte bedeutet.□

Hervorragend! An die Arbeit!

<sup>3</sup>Temperatur: Grad Celcius, Niederschlag (`precip`) mm Niederschlag pro Quadratmeter

<sup>4</sup><https://raw.githubusercontent.com/sebastiansauer/Lehre/main/data/wetter-dwd/wetter-dwd-data-dict.md>

### 11.3.1. metrische UV

#### 11.3.1.1. Modell Wetter1

Sie stellen sich nun folgende Forschungsfrage:

- 💡 Um wieviel ist die Temperatur in Deutschland pro Jahr gestiegen, wenn man die letzten ca. 100 Jahre betrachtet?

Die Modellparameter von `lm_wetter1` sind in Tabelle 11.1 zu sehen.

```
lm_wetter1 <- lm(temp ~ year, data = wetter)
parameters(lm_wetter1)
```

Tabelle 11.1.: Modellparameter von `lm_wetter1`

| Parameter   | Coefficient | SE       | 95% CI           | t(28864) | p      |
|-------------|-------------|----------|------------------|----------|--------|
| (Intercept) | -14.25      | 1.85     | (-17.87, -10.63) | -7.71    | < .001 |
| year        | 0.01        | 9.47e-04 | (9.80e-03, 0.01) | 12.30    | < .001 |

Laut Ihrem Modell wurde es pro Jahr um 0.01 Grad wärmer, pro Jahrzehnt also 0.1 und pro Jahrhundert 1 Grad.

- 💡 Das ist sicherlich nicht linear! Vermutlich ist die Temperatur bis 1950 konstant geblieben und jetzt knallt sie durch die Decke!

- 💡 Mit der Ruhe, das schauen Sie sich später an.

#### 11.3.1.2. Punkt- vs. Bereichsschätzung

In `tbl-lm-wetter1` finden sich zwei Arten von Information für den Wert des Achsenabschnitts (`b0`) und des Regressionsgewichts von `year(b1)`:

1. *Punktschätzungen* In der Spalte `Coefficient` sehen Sie den “Best-Guess” für den entsprechenden Koeffizienten in der Population. Das ist sozusagen der Wert für den sich das Modell festlegen würde, wenn es sonst nichts sagen dürfte.
2. *Bereichsschätzungen* Cleverer als Punktschätzungen sind Bereichsschätzungen (Intervallschätzungen): Hier wird ein Bereich plausibler Werte für den entsprechenden Wert angegeben. Der “Bereich plausibler Werte” wird auch als Konfidenzintervall (engl. confidence interval, CI) bezeichnet. Entsprechend gibt `CI_low` die Untergrenze des Bereichs plausibler Werte und `CI_high` die Obergrenze aus. So können wir ablesen, dass das Regressionsgewicht von `year` irgendwo zwischen praktisch Null (0.009) und ca. 0.01 Grad geschätzt wird.

- Merke: Je schmäler das Konfidenzintervall, desto genauer wird der Effekt geschätzt.

### 11.3.1.3. Modell Wetter1a

Das Modell `lm_wetter1`, bzw. die Schätzungen zu den erwarteten Werten, kann mich sich so ausgeben lassen, s. Abbildung 11.2, links. Allerdings sind das zu viele Datenpunkte. Wir sollten es vielleicht anders visualisieren, s. Abbildung 11.2, rechts. Dazu aggregieren wir die Messwerte eines Jahres zu jeweils einem Mittelwert.

```
wetter_summ <-
  wider %>%
  group_by(year) %>%
  summarise(temp = mean(temp),
            precip = mean(precip)) # precipitation: engl. für
                           ↴ Niederschlag
```

Auf dieser Basis erstellen wir ein neues lineares Modell, s. Tabelle 11.2.

```
lm_wetter1a <- lm(temp ~ year, data = wider)
parameters(lm_wetter1a)
```

Tabelle 11.2.: Modellparameter von `lm_wetter1a`

| Parameter   | Coefficient | SE       | 95% CI           | t(140) | p      |
|-------------|-------------|----------|------------------|--------|--------|
| (Intercept) | -14.14      | 2.70     | (-19.48, -8.79)  | -5.23  | < .001 |
| year        | 0.01        | 1.38e-03 | (8.86e-03, 0.01) | 8.38   | < .001 |

```
plot(estimate_relation(lm_wetter1))
plot(estimate_relation(lm_wetter1a))
```

💡 Moment mal, der Achsenabschnitt liegt bei -15 Grad! Was soll das bitte bedeuten?

### 11.3.2. UV zentrieren

Zur Erinnerung: Der Achsenabschnitt ( $\beta_0$ ; engl. *intercept*) ist definiert als der Y-Wert an der Stelle X=0, s. Kapitel 10.5.

In den Wetterdaten wäre Jahr=0 Christi Geburt. Da unsere Wetteraufzeichnung gerade mal ca. 150 Jahre in die Vergangenheit reicht, ist es vollkommen vermessen, dass Modell 2000 Jahre in die Vergangenheit zu extraplieren, ganz ohne dass wir dafür Daten haben, s. Abbildung 11.3.

Sinnvoller ist es da, z.B. einen *Referenzwert* festzulegen, etwa 1950. Wenn wir dann von allen Jahren 1950 abziehen, wird das Jahr 1950 zum neuen Jahr Null. Damit bezöge sich der Achsenabschnitt auf das Jahr 1950, was Sinn macht, denn für dieses Jahr haben wir Daten.

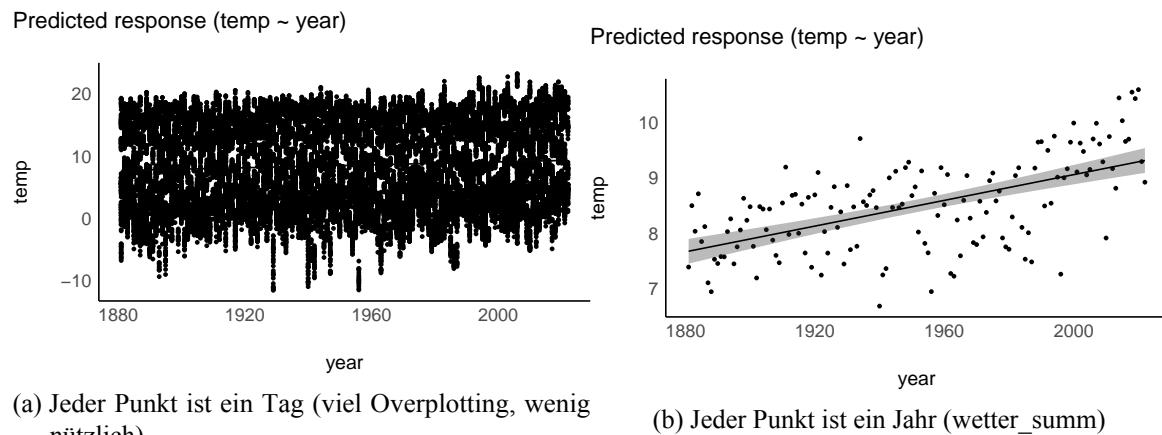


Abbildung 11.2.: Die Veränderung der mittleren Temperatur in Deutschland im Zeitverlauf (Datenquelle: DWD)

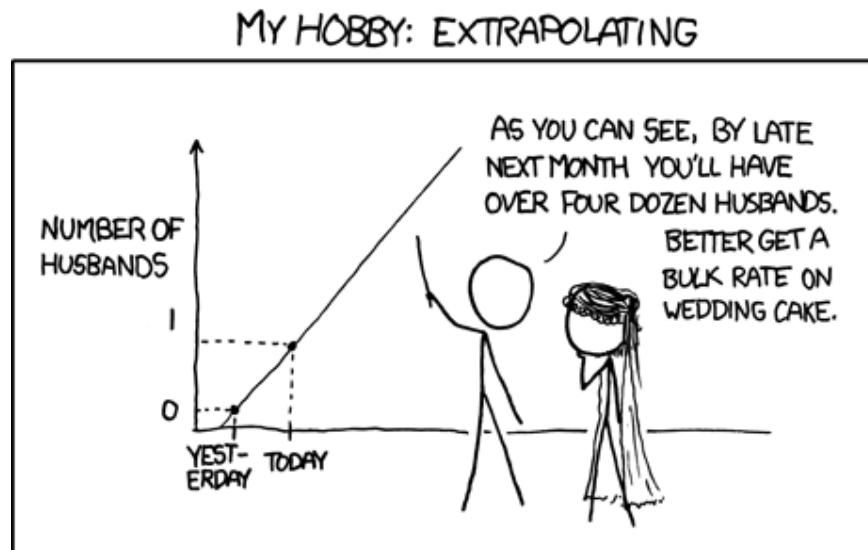


Abbildung 11.3.: Du sollst nicht ein Modell weit außerhalb seines Datenbereichs extrapoliieren

Hat man nicht einen bestimmten Wert, der sich als Referenzwert anbietet, so ist es üblich, z.B. den Mittelwert (der UV) als Referenzwert zu nehmen. Diese Transformation bezeichnet man als *Zentrierung* (engl. centering) der Daten.

So zentriert man eine Verteilung:

```
wetter <-
  wetter %>%
  mutate(year_c = year - mean(year)) # "c" wie centered
```

Das mittlere Jahr in unserer Messwertreihe ist übrigens 1951:

```
wetter %>%
  summarise(mean(year))
```

$$\begin{array}{c} \hline \text{mean(year)} \\ \hline 1951.251 \\ \hline \end{array}$$

Die Steigung (d.h. der Regressionskoeffizient für `year_c`) bleibt unverändert, nur der Achsenabschnitt ändert sich, s. Tabelle 11.4.

```
lm_wetter1_zentriert <- lm(temp ~ year_c, data = wetter)
parameters(lm_wetter1_zentriert)
```

Tabelle 11.4.: Modellparameter von `lm_wetter1_zentriert`

| Parameter   | Coefficient | SE       | 95% CI           | t(28864) | p      |
|-------------|-------------|----------|------------------|----------|--------|
| (Intercept) | 8.49        | 0.04     | (8.42, 8.57)     | 219.43   | < .001 |
| year c      | 0.01        | 9.47e-04 | (9.80e-03, 0.01) | 12.30    | < .001 |

Jetzt ist die Interpretation des Achsenabschnitts komfortabel: Im Jahr 1951 ( $x=0$ ) lag die mittlere Temperatur in Deutschland (laut DWD) bei ca. 8.5 Grad Celcius. Die Regressionsgleichung lautet:  $\text{temp\_pred} = 8.49 + 0.01 * \text{year\_c}$ . In Worten: Wir sagen eine Temperatur vorher, die sich als Summe von 8.49 Grad plus 0.01 mal das Jahr (in zentrierter Form) berechnet.

! Referenzwert entspricht Null

Der Referenzwert bzw. der Wert der Referenzgruppe entspricht dem Y-Wert bei  $x=0$  im Regressionsmodell.□

Wie gut erklärt unser Modell die Daten?

```
r2(lm_wetter1_zentriert) # aus `easystats`  
## # R2 for Linear Regression  
##      R2: 0.005  
## adj. R2: 0.005
```

Viel Varianz des Wetters erklärt das Modell mit `year_c`<sup>5</sup> aber nicht. Macht auch Sinn: Abgesehen von der Jahreszahl spielt z.B. die Jahreszeit eine große Rolle für die Temperatur. Das haben wir nicht berücksichtigt.

💡 Wie warm ist es laut unserem Modell dann im Jahr 2051?

```
predict(lm_wetter1_zentriert, newdata = tibble(year_c = 100))  
##      1  
## 9.65775
```

💡 Moment! Die Vorhersage ist doch Quatsch! Schon im Jahr 2022 lag die Durchschnittstemperatur bei 10,5° Celcius.<sup>6</sup>

💡 Wir brauchen ein besseres Modell! Zum Glück haben wir ambitionierte Nachwuchs-Wissenschaftler:innen.

### 11.3.3. Binäre UV

**Definition 11.1** (Binäre Variable). Eine *binäre* UV, auch *Indikatorvariable* oder *Dummyvariable* genannt, hat nur zwei Ausprägungen: 0 und 1.□

**Beispiel 11.3** (Binäre Variablen). Das sind zum Beispiel *weiblich* mit den Ausprägungen 0 (nein) und 1 (ja) oder *before\_1950* mit 1 für Jahre früher als 1950 und 0 ansonsten.□

**Beispiel 11.4.** Hier interessiert Sie folgende Forschungsfrage:

💡 Ob es in der zweiten Hälfte des 20. Jahrhunderts wohl wärmer warm, im Durchschnitt, als vorher?□

Aber wie erstellen Sie eine Variable `after_1950`, um die zweite Hälfte des 20. Jahrhunderts (und danach) zu fassen? Nach einem Überlegen kommen Sie auf die Idee, das vektorisierte Rechnen von R (s. Kapitel 4.7.4) auszunutzen:

---

<sup>5</sup>`year` und `year_c` sind gleich stark mit `temp` korreliert, daher wird sich die Modellgüte nicht unterscheiden.

<sup>6</sup>Quelle: Umweltbundesamt

```

year <- c(1940, 1950, 1960)
after_1950 <- year > 1950 # prüfe ob es Jahr größer als 1950
  ↳ ist
after_1950
## [1] FALSE FALSE TRUE

```

Die ersten zwei Jahre von `year` sind nicht größer als 1950, das dritte schon.

Ja, so könnte das klappen! Diese Syntax übertragen Sie auf Ihre `wetter`-Daten:

```

wetter <-
  wetter %>%
  mutate(after_1950 = year > 1950) %>%
  filter(region != "Deutschland") # ohne Daten für
  ↳ Gesamt-Deutschland

```

Scheint zu klappen!

Jetzt ein lineares Modell dazu berechnen:

```
lm_wetter_bin_uv <- lm(temp ~ after_1950, data = wetter)
```

Die Parameter des Modells lassen darauf schließen, dass es tatsächlich wärmer war nach 1950, und zwar im Schnitt offenbar ein gutes halbes Grad, s. Abbildung 11.4.

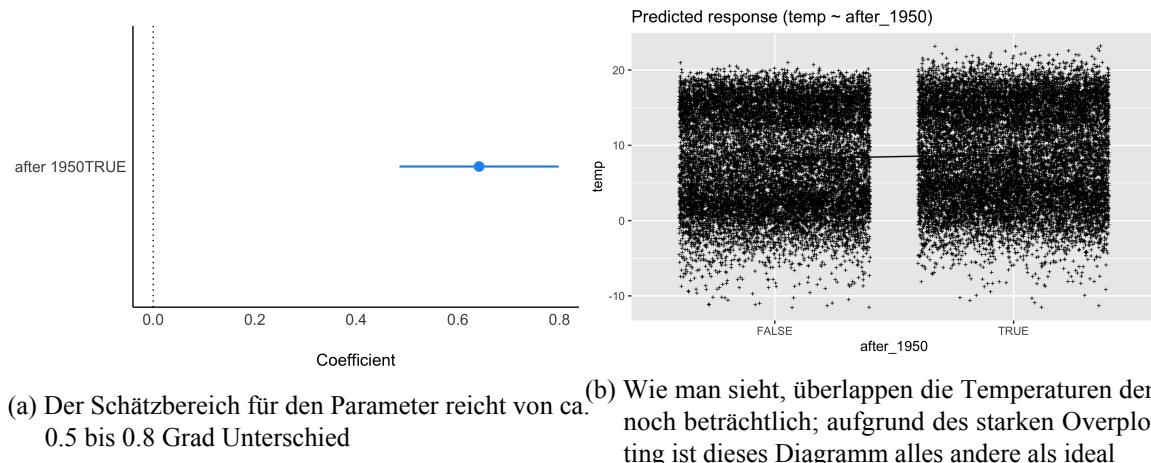


Abbildung 11.4.: Modell `temp ~ after_1950`

Leider zeigt ein Blick zum `r2`, dass die Vorhersagegüte des Modells zu wünschen übrig lässt<sup>7</sup>. □

---

<sup>7</sup>`r2(lm_wetter_bin_uv)`

### ! Lineare Modelle verkraften nur metrische Variablen

Um die Koeffizienten eines linearen Modells auszurechnen, benötigt man eine metrische X- und eine metrische Y-Variable. Hier haben wir aber keine richtige metrische X-Variable<sup>8</sup>, sondern eine *logische* Variable mit den Werten TRUE und FALSE.□

Um die X-Variable in eine metrische Variable umzuwandeln, gibt es einen einfachen Trick, den R für uns ohne viel Ankündigung durchführt: Umwandlung in mehrere binäre Variablen.

Hat ein nominaler Prädiktor zwei Stufen, so überführt<sup>9</sup> `lm()` diese Variable in eine binäre Variable. Da eine binäre Variable metrisch ist, kann die Regression in gewohnter Weise durchgeführt werden. Wenn Sie die Ausgabe der Parameter betrachten, so sehen Sie die neu erstellte binäre Variable. Man beachte, dass der ursprüngliche Datensatz nicht geändert wird, nur während der Analyse von `lm` wird die Umwandlung der Variable<sup>10</sup> durchgeführt.

⌚ Eine 1 kannst du als “Ja! Richtig!” verstehen und eine 0 als “Nein! Falsch!”

`after_1950` wird in eine Indikatorvariable umgewandelt:

| id | after_1950 |
|----|------------|
| 1  | TRUE       |
| 2  | FALSE      |

→

| id | after_1950TRUE |
|----|----------------|
| 1  | 1              |
| 2  | 0              |

**Beispiel 11.5** (Beispiel: ‘Geschlecht’ in eine binäre Variable umwandeln.). Angenommen wir haben eine Variable `geschlecht` mit den zwei Stufen Frau und Mann und wollen diese in eine Indikatorvariable umwandeln. Da “Frau” alphabetisch vor “Mann” kommt, nimmt R “Frau” als *erste* Stufe bzw. als *Referenzgruppe*. “Mann” ist dann die zweite Stufe, die in der Regression dann in Bezug zur Referenzgruppe gesetzt wird. `lm` wandelt uns diese Variable in `geschlechtMann` um mit den zwei Stufen 0 (kein Mann, also Frau) und 1 (Mann).□

| id | geschlecht |
|----|------------|
| 1  | Mann       |
| 2  | Frau       |

→

| id | geschlechtMann |
|----|----------------|
| 1  | 1              |
| 2  | 0              |

Ein lineares Modell mit binärer UV ist nichts anderes die Differenz der Gruppenmittelwerte zu berechnen:

```
wetter %>%
  group_by(after_1950) %>%
  summarise(temp_mean = mean(temp))
```

<sup>8</sup>UV

<sup>9</sup>synonym: transformiert

<sup>10</sup>Transformation

|       | after_1950 | temp_mean |
|-------|------------|-----------|
| FALSE |            | 8.175287  |
| TRUE  |            | 8.816761  |

Die Interpretation eines linearen Modells mit binärer UV veranschaulicht Abbildung 11.5: Der Achsenabschnitt ( $b_0$ ) entspricht dem Mittelwert der 1. Gruppe. Der Mittelwert der 2. Gruppe entspricht der *Summe* aus Achsenabschnitt und dem Koeffizienten der zweiten Gruppe. (Abbildung 11.5 zeigt nur die Daten für den Monat Juli im Bundesland Bayern, der Einfachheit und Übersichtlichkeit halber.)

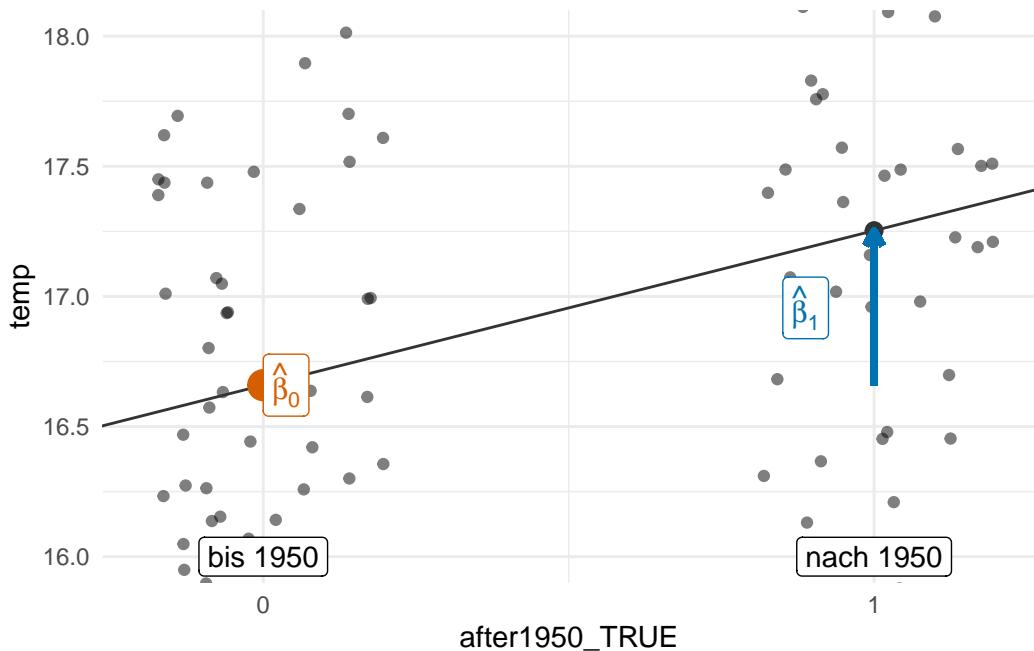


Abbildung 11.5.: Sinnbild zur Interpretation eines linearen Modells mit binärer UV (reingezoomt, um den Mittelwertsunterschied hervorzuheben)

Fassen wir die Interpretation der Koeffizienten für das Modell mit binärer UV zusammen:

1. Mittelwert der 1. Gruppe (bis 1950): Achsenabschnitt ( $b_0$ )
2. Mittelwert der 2. Gruppe (nach 1950): Achsenabschnitt ( $b_0$ ) + Steigung der Regressionsgeraden ( $b_1$ )

Für die Modellwerte  $\hat{y}$  gilt also:

- Temperatur laut Modell bis 1950:  $\hat{y} = \beta_0 = 17.7$
- Temperatur laut Modell bis 1950:  $\hat{y} = \beta_0 + \beta_1 = 17.7 + 0.6 = 18.3$

### **i Hinweis**

Bei *nominalen* (und auch bei *binären*) Variablen ist  $\beta_1$  ein *Schalter*; bei *metrischen* Variablen ein *Dimmer*.<sup>11</sup> □

#### 11.3.4. Nominale UV

In diesem Abschnitt betrachten wir ein lineare Modell<sup>12</sup> mit einer mehrstufigen<sup>13</sup> (nominalskalierten) UV.<sup>14</sup>

**Beispiel 11.6.** Ob es wohl substanzielle<sup>15</sup> Temperaturunterschiede zwischen den Bundesländern gibt?

Befragen wir dazu ein lineares Modell, s. Tabelle 11.10.

```
lm_wetter_region <- lm(temp ~ region, data = wetter)
parameters(lm_wetter_region)
```

Tabelle 11.10.: Modellparameter für lm\_wetter\_region

| Parameter                       | Coefficient | SE   | 95% CI         | t(27152) | p      |
|---------------------------------|-------------|------|----------------|----------|--------|
| (Intercept)                     | 8.25        | 0.16 | (7.93, 8.56)   | 51.62    | < .001 |
| region (Bayern)                 | -0.63       | 0.23 | (-1.07, -0.19) | -2.79    | 0.005  |
| region (Brandenburg)            | 0.57        | 0.23 | (0.13, 1.02)   | 2.53     | 0.011  |
| region (Brandenburg/Berlin)     | 0.58        | 0.23 | (0.14, 1.03)   | 2.59     | 0.010  |
| region (Hessen)                 | 0.11        | 0.23 | (-0.33, 0.56)  | 0.51     | 0.612  |
| region (Mecklenburg-Vorpommern) | 0.08        | 0.23 | (-0.37, 0.52)  | 0.34     | 0.732  |
| region (Niedersachsen)          | 0.52        | 0.23 | (0.07, 0.96)   | 2.29     | 0.022  |
| region                          | 0.52        | 0.23 | (0.08, 0.96)   | 2.31     | 0.021  |
| (Niedersachsen/Hamburg/Bremen)  |             |      |                |          |        |
| region (Nordrhein-Westfalen)    | 0.80        | 0.23 | (0.35, 1.24)   | 3.53     | < .001 |
| region (Rheinland-Pfalz)        | 0.46        | 0.23 | (0.02, 0.90)   | 2.03     | 0.042  |
| region (Saarland)               | 0.71        | 0.23 | (0.27, 1.16)   | 3.16     | 0.002  |
| region (Sachsen)                | -0.04       | 0.23 | (-0.48, 0.40)  | -0.18    | 0.853  |
| region (Sachsen-Anhalt)         | 0.55        | 0.23 | (0.11, 1.00)   | 2.45     | 0.014  |
| region (Schleswig-Holstein)     | 0.17        | 0.23 | (-0.27, 0.62)  | 0.76     | 0.446  |
| region (Thueringen)             | -0.48       | 0.23 | (-0.92, -0.03) | -2.11    | 0.035  |

<sup>11</sup> Ich danke Karsten Lübke für diese Idee.

<sup>12</sup> für uns synonym: Regressionsmodell

<sup>13</sup> drei oder mehr Stufen bzw. Ausprägungen

<sup>14</sup> So ein Modell ist von den Ergebnissen her praktisch identisch zu einer einfachen *Varianzanalyse*.

<sup>15</sup> wie könnte man dieses Wort eigentlich definieren?

Tabelle 11.10.: Modellparameter für lm\_wetter\_region

| Parameter                             | Coefficient | SE   | 95% CI        | t(27152) | p     |
|---------------------------------------|-------------|------|---------------|----------|-------|
| region<br>(Thueringen/Sachsen-Anhalt) | 0.10        | 0.23 | (-0.34, 0.54) | 0.43     | 0.664 |

Hat die nominalskalierte UV mehr als zwei Stufen, so transformiert lm sie in mehr als eine Indikatorvariablen um. Genauer gesagt ist es immer eine Indikatorvariablen weniger als es Stufen in der nominalskalierten Variablen gibt.

Betrachten wir ein einfaches Beispiel, eine Tabelle mit der Spalte Bundesland – aus Gründen der Einfachheit hier nur mit *drei* Bundesländern. Damit lm arbeiten kann, wird Bundesland in *zwei* Indikatorvariablen umgewandelt:

| id | Bundesland  |   | id | BL_Bayern | BL_Bra |
|----|-------------|---|----|-----------|--------|
| 1  | BaWü        | → | 1  | 0         | 0      |
| 2  | Bayern      |   | 2  | 1         | 0      |
| 3  | Brandenburg |   | 3  | 0         | 1      |

Auch im Fall mehrerer Ausprägungen einer nominalen Variablen gilt die gleiche Logik der Interpretation wie bei binären Variablen:

1. Mittelwert der 1. Gruppe: Achsenabschnitt (b0)
2. Mittelwert der 2. Gruppe: Achsenabschnitt (b0) + Steigung der 1. Regressionsgeraden (b1)
3. Mittelwert der 2. Gruppe: Achsenabschnitt (b0) + Steigung der 2. Regressionsgeraden (b2)
4. usw.

Es kann nervig sein, dass das Bundesland, welches als *Referenzgruppe* (sprich als Gruppe des Achsenabschnitts ausgewählt wurde) nicht explizit in der Ausgabe angegeben ist. Der Wert der Referenzgruppe findet seinen Niederschlag im Achsenabschnitt.

**i Hinweis**

Bei einer Variable vom Typ character wählt R den alphabetisch ersten Wert als Referenzgruppe für ein lineares Modell aus. Bei einer Variable vom Typ factor ist die Reihenfolge bereits festgelegt, vgl. Kapitel 11.3.5. Der Mittelwert dieser Gruppe entspricht dem Achsenabschnitt. □

**Beispiel 11.7** (Achsenabschnitt in wetter\_lm2). Da Baden-Württemberg das alphabetisch erste Bundesland ist, wird es von R als Referenzgruppe ausgewählt, dessen Mittelwert als Achsenabschnitt im linearen Modell hergenommen wird. □

Am einfachsten verdeutlicht sich lm\_wetter\_region vielleicht mit einem Diagramm, s. Abbildung 11.6.

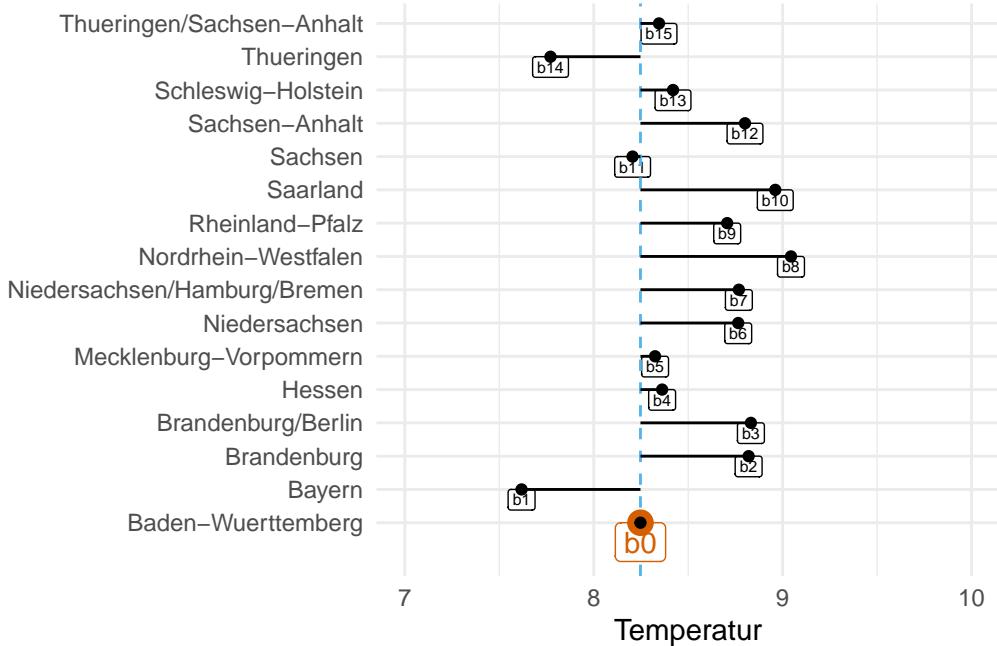


Abbildung 11.6.: Sinnbild zur Interpretation eines linearen Modells mit nominaler UV (reingezoomt, um den Mittelwertsunterschied hervorzuheben). Die Achsen wurden um  $90^\circ$  gedreht, damit man die Namen der Bundesländer besser lesen kann.

**Beispiel 11.8** (Niederschlagsmenge im Vergleich der Monate). Eine weitere Forschungsfrage, die Sie nicht außer acht lassen wollen, ist die Frage nach den jahreszeitlichen Unterschieden im Niederschlag (engl. precipitation). Los R, rechnen!

💡 Endlich geht's weiter! Ergebnisse in Tabelle 11.13! □

```
lm_wetter_month <- lm(precip ~ month, data = wetter)
parameters(lm_wetter_month)
```

Tabelle 11.13.: Modellparameter für lm\_wetter\_month

| Parameter   | Coefficient | SE   | 95% CI         | t(27166) | p      |
|-------------|-------------|------|----------------|----------|--------|
| (Intercept) | 53.27       | 0.41 | (52.46, 54.08) | 128.76   | < .001 |
| month       | 1.14        | 0.06 | (1.03, 1.25)   | 20.29    | < .001 |

Ja, da scheint es deutliche Unterschiede im Niederschlag zu geben. Wir brauchen ein Diagramm zur Verdeutlichung, s. Abbildung 11.7, links.<sup>16</sup> Oh nein: R betrachtet month als numerische Variable! Aber "Monat" bzw. "Jahreszeit" sollte nominal sein.

<sup>16</sup>plot(estimate\_expectation(lm\_wetter\_month))

👉 Aber month ist als Zahl in der Tabelle hinterlegt. Jede ehrliche Maschine verarbeitet eine Zahl als Zahl, ist doch klar!

😺 Okay, R, wir müssen month in eine nominale Zahl transformieren. Wie geht das?

👉 Dazu kannst du den Befehl `factor` nehmen. Damit wandelst du eine numerische Variable in eine nominalskalierte Variable (Faktorvariable) um. Faktisch heißt das, dass dann eine Zahl als Text gesehen wird.

**Beispiel 11.9.** Transformiert man 42 mit `factor`, so wird aus 42 "42". Aus der Zahl wird ein Text. Alle metrischen Eigenschaften gehen verloren; die Variable ist jetzt auf nominalen Niveau. □

```
wetter <-  
  weather %>%  
  mutate(month_factor = factor(month))
```

Jetzt berechnen wir mit der faktorisierten Variablen ein lineares Modell, s. Tabelle 11.14.

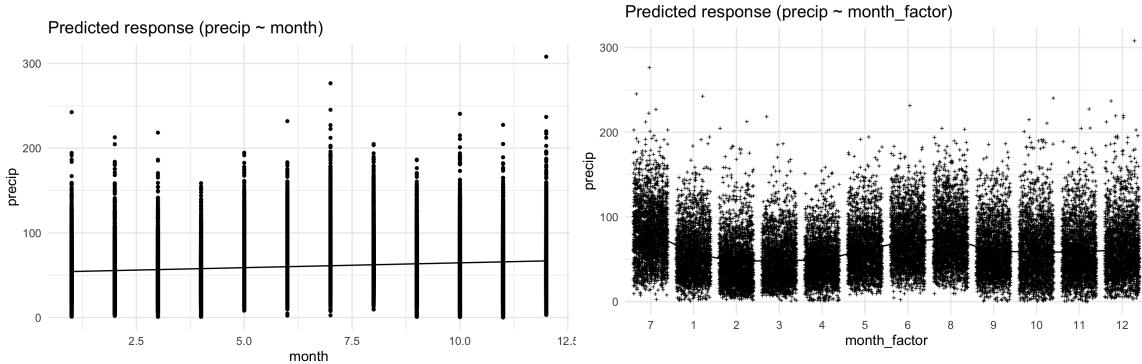
```
lm_wetter_month_factor <- lm(precip ~ month_factor, data =  
  ↪ weather)  
parameters(lm_wetter_month_factor)
```

Tabelle 11.14.: Modellparameter von lm\_wetter\_month\_factor

| Parameter         | Coefficient | SE   | 95% CI          | t(27156) | p      |
|-------------------|-------------|------|-----------------|----------|--------|
| (Intercept)       | 56.95       | 0.64 | (55.68, 58.21)  | 88.56    | < .001 |
| month factor (2)  | -9.95       | 0.91 | (-11.73, -8.17) | -10.94   | < .001 |
| month factor (3)  | -7.78       | 0.91 | (-9.56, -6.00)  | -8.56    | < .001 |
| month factor (4)  | -8.49       | 0.91 | (-10.27, -6.71) | -9.34    | < .001 |
| month factor (5)  | 4.74        | 0.91 | (2.96, 6.53)    | 5.22     | < .001 |
| month factor (6)  | 14.34       | 0.91 | (12.56, 16.12)  | 15.77    | < .001 |
| month factor (7)  | 24.36       | 0.91 | (22.57, 26.14)  | 26.74    | < .001 |
| month factor (8)  | 17.52       | 0.91 | (15.74, 19.31)  | 19.24    | < .001 |
| month factor (9)  | 1.93        | 0.91 | (0.15, 3.72)    | 2.12     | 0.034  |
| month factor (10) | 2.29        | 0.91 | (0.51, 4.08)    | 2.52     | 0.012  |
| month factor (11) | 0.89        | 0.91 | (-0.89, 2.68)   | 0.98     | 0.327  |
| month factor (12) | 5.20        | 0.91 | (3.42, 6.99)    | 5.71     | < .001 |

Sehr schön! Jetzt haben wir eine Referenzgruppe (Monat 1, d.h. Januar) und 11 Unterschiede zum Januar, s. Abbildung 11.7, rechts.

Möchte man die Referenzgruppe eines Faktors ändern, kann man dies mit `relevel` tun:



(a) `lm_wetter_month`, Monat fälschlich als metrische Variable  
(b) `lm_wetter_month_text`, Monat korrekt als nominale Variable (aber mit viel Overplotting, das müsste man besser machen)

Abbildung 11.7.: Niederschlagsunterschiede pro Monat (ein Punkt ist ein Jahr); aufgrund der vielen Datenpunkte ist das Diagramm wenig übersichtlich (Overplotting).

```
wetter <-
  wetter %>%
  mutate(month_factor = relevel(month_factor, ref = "7"))
```

So sieht dann die geänderte Reihenfolge aus:<sup>17</sup>

```
levels(wetter$month_factor)
## [1] "7"  "1"  "2"  "3"  "4"  "5"  "6"  "8"  "9"  "10" "11"
## [12] "12"
```

### 11.3.5. Binäre plus metrische UV

In diesem Abschnitt untersuchen wir ein lineares Modell mit zwei UV: einer *zweistufigen* (binären) UV plus einer *metrischen* UV.<sup>18</sup>

**Beispiel 11.10.** Ob sich die Niederschlagsmenge wohl unterschiedlich zwischen den Monaten entwickelt hat in den letzten gut 100 Jahren? Der Einfachheit halber greifen Sie sich nur zwei Monate heraus (Januar und Juli).

```
wetter_month_1_7 <-
  wetter %>%
  filter(month == 1 | month == 7)
```

💡 Ich muss mal kurz auf eine Sache hinweisen...

<sup>17</sup>Zum Dollar-Operator s. Kapitel 4.12.3

<sup>18</sup>So ein Modell kann auch als *Kovarianzanalyse* (engl. analysis of covariance, ancova) bezeichnet werden.

### **i** Faktorvariable

Eine Faktorvariable ist einer der beiden Datentypen in R, die sich für nominalskalierte Variablen anbieten: Textvariablen (`character`) und Faktor-Variablen (`factor`). Ein wichtiger Unterschied ist, dass die erlaubten Ausprägungen (“Faktorstufen”) bei einer Faktor-Variable mitgespeichert werden, bei der Text-Variable nicht.

Das kann praktisch sein, denn bei einer Faktorvariable ist immer klar, welche Ausprägungen in Ihrer Variable möglich sind.□

**Beispiel 11.11** (Beispiel für eine Faktorvariable).

```
geschlecht <- c("f", "f", "m")
geschlecht_factor <- factor(geschlecht)
geschlecht_factor
## [1] f f m
## Levels: f m
```

**Beispiel 11.12** (Filtern verändert die Faktorstufen nicht). Wenn Sie von der Faktorvariablen<sup>19</sup> `geschlecht` das 3. Element ("m") herausfiltern, so dass z.B. nur die ersten beiden Elemente übrig bleiben mit allein der Ausprägung "f", merkt sich R trotzdem, dass es *zwei* Faktorstufen gibt ("f" und "m").

Genaus so ist es, wenn Sie aus `wetter` nur die Monate "1" und "7" herausfiltern: R merkt sich, dass es 12 Faktorstufen gibt. Möchten Sie die herausgefilterten Faktorstufen “löschen”, so können Sie einfach die Faktorvariable neu berechnen (mit `factor`).□

```
wetter_month_1_7 <-
wetter %>%
filter(month == 1 | month == 7) %>%
# Faktor (und damit die Faktorstufen) neu berechnen:
mutate(month_factor = factor(month))
```

Okay. Wie spezifiziert man jetzt das lineare Modell?□

Hat man mehrere (“multiple”) X-Variablen<sup>20</sup>, so trennt man sich mit einem Plus-Zeichen in der Regressionsformel, z.B. `temp ~ year_c + month`.

### **!** Multiple Regression

Eine multiple Regression beinhaltet mehr als eine X-Variable. Die Modellformel spezifiziert man so:

<sup>19</sup>synonym: nominalskalierte Variable

<sup>20</sup>Prädiktoren, unabhängige Variablen, X-Variablen

$$y x_1 + x_2 + \dots + x_n \quad \square$$

### **i** Modellgleichung

Das Pluszeichen hat in der Modellgleichung<sup>21</sup> *keine* arithmetische Funktion. Es wird nichts addiert. In der Modellgleichung sagt das Pluszeichen nur “und noch folgende UV...”. $\square$

Die obige Modellgleichung liest sich also so:

Temperatur ist eine Funktion von der (zentrierten) Jahreszahl und des Monats

```
lm_year_month <- lm(precip ~ year_c + month_factor, data =
  ↴ wetter_month_1_7)
```

Die Modellparameter sind in Tabelle 11.15 zu sehen.

Tabelle 11.15.: Modellparameter von lm\_year\_month

| Parameter        | Coefficient | SE   | 95% CI           | t(4525) | p      |
|------------------|-------------|------|------------------|---------|--------|
| (Intercept)      | 56.94       | 0.68 | (55.60, 58.27)   | 83.57   | < .001 |
| year c           | 0.03        | 0.01 | (5.59e-03, 0.05) | 2.43    | 0.015  |
| month factor (7) | 24.37       | 0.97 | (22.48, 26.27)   | 25.25   | < .001 |

Die Modellkoeffizienten sind so zu interpretieren:

1. Achsenabschnitt (b0, (Intercept)): Im Referenzjahr (1951) im *Referenzmonat Januar* lag die Niederschlagsmenge bei 57 mm pro Quadratmeter.
2. Regressionskoeffizient für Jahr (b1, year\_c): Pro Jahr ist die Niederschlagsmenge im Schnitt um 0.02 mm an (im Referenzmonat).
3. Regressionskoeffizient für Monat (b2, month [7]) Im Monat 7 (Juli) lag die mittlere Niederschlagsmenge (im Referenzjahr) knapp 25 mm über dem mittleren Wert des Referenzmonats (Januar).

Die Regressiongleichung von lm\_year\_month lautet: precip\_pred = 56.94 + 0.03\*year\_c + 24.37\*month\_factor\_7.

Im Monat Juli ist month\_factor\_7 = 1, ansonsten (Januar) ist month\_factor = 0.

**💡** Puh, kompliziert!

**💡** Es gibt einen Trick, man kann sich von R einfach einen beliebigen Y-Wert berechnen lassen, s. Beispiel 11.13.

---

<sup>21</sup>synonym: Regressionsformel

**Beispiel 11.13** (Niederschlag laut Modell Im Juli 2020?). Hey R, berechne uns anhand neuer Daten den laut Modell zu erwartenden Niederschlag für Januar im Jahr 2020!

```
neue_daten <- tibble(year_c = 2020-1951,
                      month_factor = factor("1"))
predict(lm_year_month, newdata = neue_daten)
##      1
## 58.92171
```

**i Hinweis**

Alle Regressionskoeffizienten beziehen sich auf den Y-Wert *unter der Annahme, dass alle übrigen Prädiktoren den Wert Null (bzw. Referenzwert) aufweisen.* □

Visualisieren wir uns die geschätzten Erwartungswert pro Prädiktorwert, s. Abbildung 11.8:  
`plot(estimate_expectation(lm_year_month))`

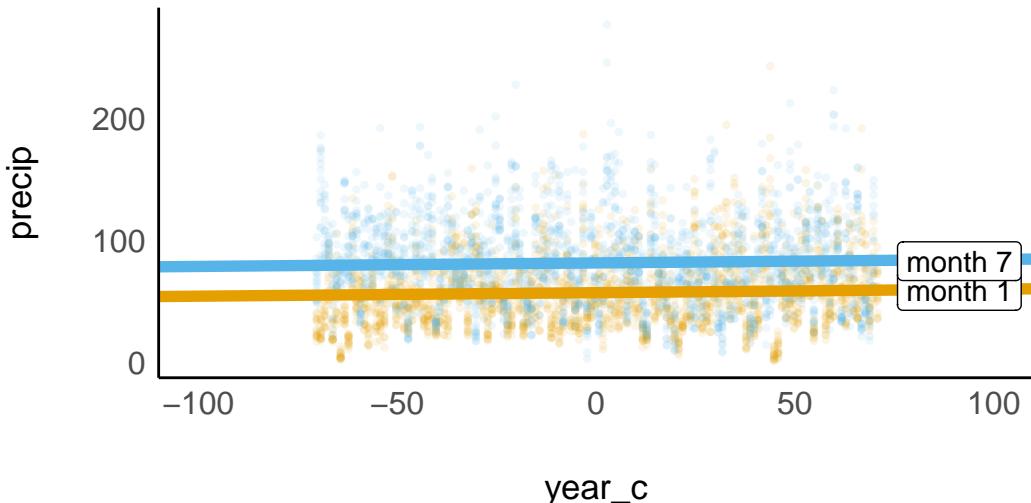


Abbildung 11.8.: Temperaturverlauf über die Jahre für zwei Monate. Man beachte, dass die Regressionsgeraden *parallel* sind.

Mit `scale_color_okabeito` haben wir die Standard-Farbpalette durch die von (Okabe und Ito 2023) ersetzt<sup>22</sup>. Das ist nicht unbedingt nötig, aber robuster bei Schwarz-Weiß-Druck und bei Sehschwächen, vgl. Kapitel 6.9.2.

Die erklärte Varianz von `lm_year_month` liegt bei:

```
r2(lm_year_month)
## # R2 for Linear Regression
##      R2: 0.124
## adj. R2: 0.124
```

<sup>22</sup>s. Hinweise hier: [https://malcolmbarrett.github.io/ggokabeito/reference/palette\\_okabe\\_ito.html](https://malcolmbarrett.github.io/ggokabeito/reference/palette_okabe_ito.html)

### 11.3.6. Interaktion

Eine Modellgleichung der Form  $\text{temp} \sim \text{year} + \text{month}$  zwingt die Regressionsgeraden dazu, parallel zu verlaufen. Aber vielleicht würden sie besser in die Punktwolken passen, wenn wir ihnen erlauben, auch *nicht* parallel verlaufen zu dürfen?

Nicht-parallele Regressionsgeraden erlauben wir, indem wir das Regressionsmodell wie folgt spezifizieren und visualisieren, s. Listing 11.1.

---

**Listing 11.1** Ein Interaktionsmodell spezifiziert man in dieser Art:  $y \sim x_1 + x_2 + x_1:x_2$

---

```
lm_year_month_interaktion <- lm(  
  precip ~ year_c + month_factor + year_c:month_factor,  
  data = wetter_month_1_7)
```

---

Visualisiert ist das Modell in Abbildung 11.9.

```
plot(estimate_expectation(lm_year_month_interaktion)) +  
  scale_color_okabeito() # schönes Farbschema
```

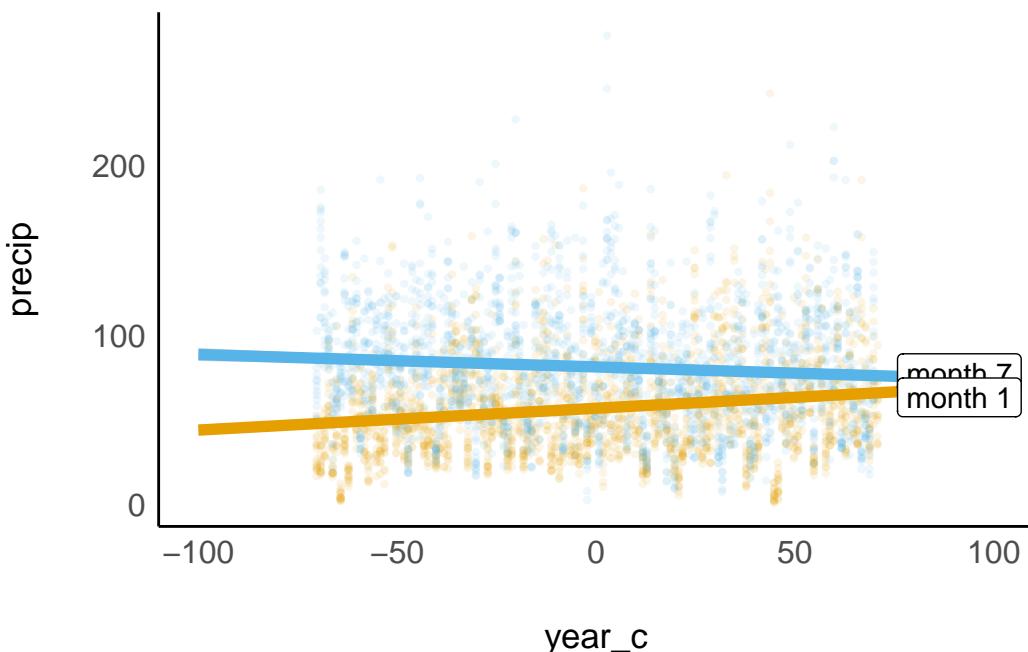


Abbildung 11.9.: Niederschlag im Jahresverlauf und Monatsvergleich mit Interaktionseffekt: Die Veränderung im Verlauf der Jahre ist unterschiedlich für die Monate (Janur vs. Juli). Die beiden Regressionsgeraden sind *nicht* parallel.

Der *Doppelpunkt-Operator* (`:`) fügt der Regressionsgleichung einen *Interaktionseffekt* hinzu, in diesem Fall die Interaktion von Jahr (`year_c`) und Monat (`month_factor`):

```
precip ~ year_c + month_factor + year_c:month_factor
```

! Wichtig

Einen Interaktionseffekt von x1 und x2 kennzeichnet man in R mit dem Doppelpunkt-Operator,  
 $x1:x2:$

$$y \sim x1 + x2 + x1:x2 \square$$

In Worten:

y wird modelliert als eine Funktion von x1 und x2 und dem Interaktionseffekt von x1 mit x2.

Wie man in Abbildung 11.9 sieht, sind die beiden Regressionsgeraden *nicht parallel*.

i Hinweis

Sind die Regressionsgeraden von zwei (oder mehr) Gruppen nicht parallel, so liegt ein Interaktionseffekt vor.□

**Beispiel 11.14** (Interaktionseffekt von Niederschlag und Monat). Wie ist die Veränderung der Niederschlagsmenge (Y-Achse) im Verlauf der Jahre (X-Achse)? *Das kommt darauf an, welchen Monat man betrachtet*. Der Effekt der Zeit ist *unterschiedlich* für die Monate: Im Juli nahm der Niederschlag ab, im Januar zu.□

Liegt ein Interaktionseffekt vor, kann man nicht mehr von “dem” (statistischen) Effekt eines Prädiktors (afu die Y-Variable) sprechen. Vielmehr muss man unterscheiden: Je nach Gruppe (z.B. Monat) unterscheidet der Effekt.<sup>23</sup>

Betrachten wir die Parameterwerte des Interaktionsmodells, s. Tabelle 11.16.

Tabelle 11.16.: Modellparameter von lm\_year\_month\_interaktion

| Parameter                 | Coefficient | SE   | 95% CI         | t(4524) | p      |
|---------------------------|-------------|------|----------------|---------|--------|
| (Intercept)               | 56.91       | 0.68 | (55.59, 58.24) | 84.21   | < .001 |
| year c                    | 0.13        | 0.02 | (0.10, 0.16)   | 7.80    | < .001 |
| month factor (7)          | 24.37       | 0.96 | (22.50, 26.25) | 25.45   | < .001 |
| year c × month factor (7) | -0.20       | 0.02 | (-0.25, -0.16) | -8.62   | < .001 |

Neu bei der Ausgabe zu diesem Modell ist die Zeile `year c × month factor [7]`. Sie gibt die Stärke des Interaktionseffekts an. Die Zeile zeigt, wie unterschiedlich sich die die Niederschlagsmenge zwischen den beiden Monaten im Verlauf der Jahre ändert: Im Monat "7" ist der Effekt von `year_c` um 0.20 mm geringer: Die Regressionsgerade neigt sich mehr nach "unten" im Monat Juli, da der Koeffizient kleiner als Null ist.

<sup>23</sup>Effekt ist hier immer statistisch, nie kausal gemeint.

Die Regressionsgleichung lautet: `precip_pred = 56.91 + 0.13*year_c + 24.37*month_factor_7 - 0.20*year_c:month_factor_7`.

### ! Wichtig

Der Achsenabschnitt gibt den Wert für Y an unter der Annahme, dass alle Prädiktoren den Wert Null aufweisen. In diesem Fall gibt der Achsenabschnitt also den Niederschlag für den Janur des Jahres 1951 an. Die Regressionskoeffizienten geben die Zunahme in Y an, wenn der jeweilige Prädiktorwert um 1 steigt, die übrigen Prädiktoren aber den Wert 0 aufweisen. □

Das R-Quadrat von `lm_year_month_interaktion` beträgt übrigens nur geringfügig mehr als im Modell ohne Interaktion:

```
r2(lm_year_month_interaktion) # aus `easystats`  
## # R2 for Linear Regression  
##      R2: 0.139  
## adj. R2: 0.138
```

## 11.4. Modelle mit vielen UV

### 11.4.1. Zwei metrische UV

Ein Modell mit zwei metrischen UV kann man sich im 3D-Raum visualisieren, s. Abbildung 11.12, oder im 2D-Raum, s. Abbildung 11.11. Im 3D-Raum wird die Regressionsgerade zu einer *Regressionsfläche*.

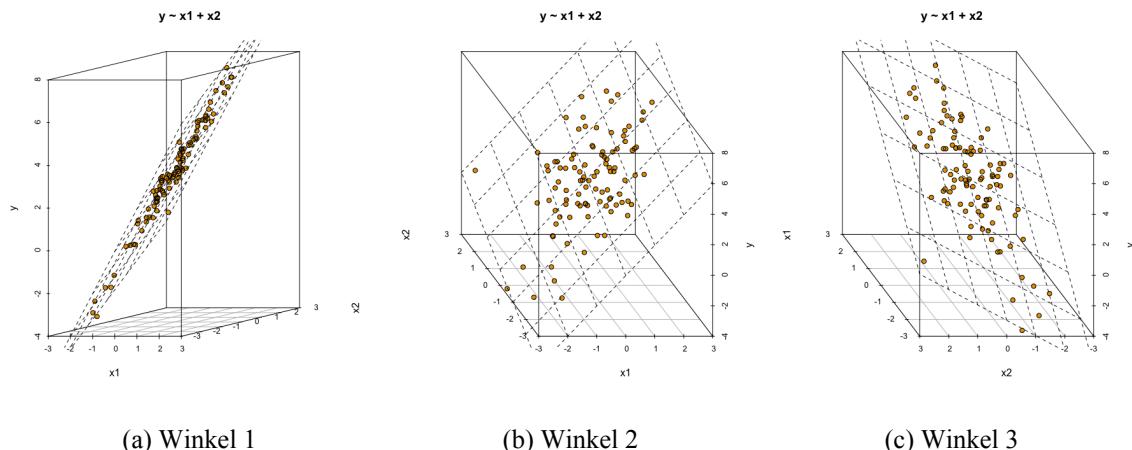


Abbildung 11.10.: Ein lineares Modell,  $y \sim x_1 + x_2$  mit zwei Prädiktoren im 3D-Raum.

Grundsätzlich kann man viele Prädiktoren in ein (lineares) Modell aufnehmen. Betrachten wir z. B. folgendes lineares Modell mit zwei metrischen UV.

## Predicted response ( $y \sim x_1 + x_2$ )

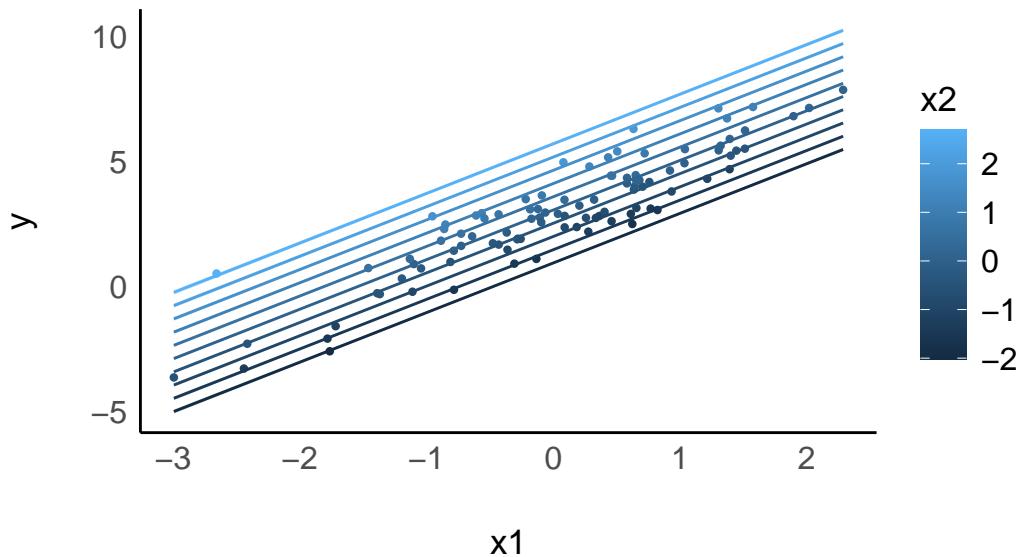


Abbildung 11.11.: 2D-Diagramm für 3D-Modell

```
lm_mario_2uv <- lm(total_pr ~ start_pr + ship_pr, data =  
  ↵ mariokart %>% filter(total_pr < 100))
```

?@fig-mario-2uv visualisiert das Modell lm\_mario2v in einem 3D-Diagramm (betrachtet aus verschiedenen Winkeln).

### 11.4.2. Viele UV ins Modell?

Wir könnten im Prinzip alle Variablen unserer Datentabelle als Prädiktoren in das Regressionsmodell aufnehmen. Die Frage ist nur: Macht das Sinn?

Hier sind einige Richtlinien, die helfen, welche Prädiktoren (und wie viele) man in ein Modell aufnehmen sollte (Gelman, Hill, und Vehtari 2021b), s. S. 199:

1. Man sollte alle Prädiktoren aufnehmen, von denen anzunehmen ist, dass Sie Ursachen für die Zielvariablen sind
2. Bei Prädiktoren mit starken (absoluten) Effekten kann es Sinn machen, ihre Interaktionseffekte auch mit in das Modell aufzunehmen
3. Prädiktoren mit kleinem Schäzbereich (95% CI) sollten tendenziell im Modell belassen werden, da sie die Modellgüte verbessern

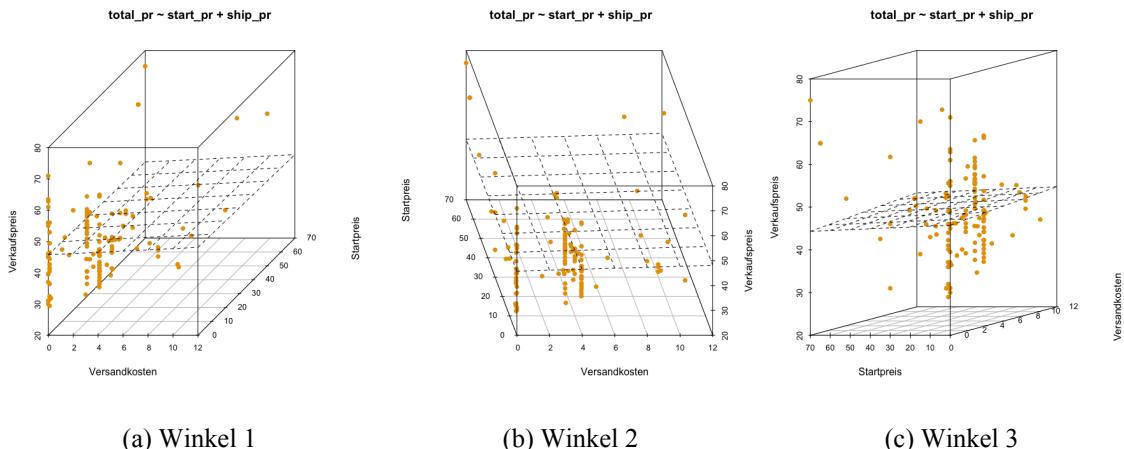


Abbildung 11.12.: Das Modell lm\_mario2v mit 2 metrischen UV (und 1 metrische AV) als 3D-Diagramm

## 11.5. Fallbeispiel zur Prognose

**Beispiel 11.15** (Prognose des Verkaufspreis). Ganz können Sie von Business-Welt und ihren Gratifikationen nicht lassen, trotz Ihrer wissenschaftlichen Ambitionen. Sie haben den Auftrag bekommen, den Verkaufspreis von Mariokart-Spielen möglichst exakt vorherzusagen. Also gut, das Honorar ist phantastisch, Sie sind jung und brauchen das Geld.□

### 11.5.1. Modell "all-in"

Um die Güte Ihrer Vorhersagen zu prüfen, teilt Ihr Chef den Datensatz in zwei zufällige Teile.

□♂ Ich teile den Datensatz mariokart zufällig in zwei Teile. Den ersten Teil kannst du nutzeln, um Modelle zu berechnen ("trainieren") und ihre Güte zu prüfen. Den Teil nenne ich "Trainingssample", hört sich cool an, oder? Im Train-Sample ist ein Anteil (`fraction`) von 70% der Daten, okay? Die restlichen Daten behalte ich. Wenn du ein gutes Modell hast, kommst du und wir berechnen die Güte deiner Vorhersagen in dem verbleibenden Teil, die übrigen 30% der Daten. Diesen Teil nennen wir Test-Sample, alles klar?

Wenn die Daten auf Ihrer Festplatte liegen, z.B. im Unterordner `daten`, dann können Sie sie von dort importieren:

```
mariokart_train <- read.csv("daten/mariokart_train.csv")
```

Alternativ können Sie sie auch von diesem Pfad von einem Rechner in der Cloud herunterladen:

```
mariokart_train_path <- paste0(
  "https://raw.githubusercontent.com/sebastiansauer/",
  "statistik1/main/daten/mariokart_train.csv")

mariokart_train <- read.csv(mariokart_train_path)
```

Dann importieren wir auf gleiche Weise Test-Sample in R:

```
mariokart_test_path <- paste0(
  "https://raw.githubusercontent.com/sebastiansauer/",
  "statistik1/main/daten/mariokart_test.csv")

mariokart_test <- read.csv(mariokart_test_path)
```

Also los. Sie probieren mal die “All-in-Strategie”: Alle Variablen rein in das Modell. Viel hilft viel, oder nicht?

```
lm_allin <- lm(total_pr ~ ., data = mariokart_train)
r2(lm_allin) # aus easystats
## # R2 for Linear Regression
##          R2: 0.994
## adj. R2: 0.979
```

Der Punkt in `total_pr ~ .` heißt “alle Variablen in der Tabelle (außer `total_pr`)”.

👉 Hey! Das ist ja fast perfekte Modellgüte!

⚠️ Vorsicht: Wenn ein Angebot aussieht wie “too good to be true”, dann ist es meist auch too good to be true.

### 💡 Overfitting

Der Grund für den fast perfekten Modellfit ist die Spalte `Title`. Unser Modell hat einfach den Titel jeder Auktion auswendig gelernt. Weiß man, welcher Titel zu welcher Auktion gehört, kann man perfekt die Auktion aufsagen bzw. das Verkaufsgebot perfekt vorhersagen. Leider nützen die Titel der Auktionen im Train-Sample *nichts* für andere Auktionen. Im Test-Sample werden unsere Vorhersagen also grottenschlecht sein, wenn wir uns auf die Titel der Auktionen im Test-Sample stützen. Merke: Höchst idiografische Informationen wie Namen, Titel etc. sind nicht nützlich, um allgemeine Muster zu erkennen und damit exakte Prognosen zu erstellen.□

Probieren wir also die Vorhersage im Test-Sample:

```
predict(lm_allin, newdata = mariokart_test)
## Error in eval(predvars, data, env): object 'V1' not found
```

Oh nein! Was ist los!? Eine Fehlermeldung!

### 🔥 Vorsicht

Nominalskalierte Prädiktorvariablen mit vielen Ausprägungen, wie `title` sind problematisch. Kommt eine Ausprägung von `title` im Test-Sample vor, die es *nicht* im Train-Sample gab, so resultiert ein Fehler beim `predict`. Häufig ist es sinnvoll, auf diese Variable zu verzichten, da diese Variablen oft zu Overfitting führen.□

### 11.5.2. Modell “all-in”, ohne Titelspalte

Okay, also auf die Titelspalte sollten wir vielleicht besser verzichten. Nächster Versuch.

```
mariokart_train2 <-
  mariokart_train %>%
  select(-c(title, V1, id))
```

Wir entfernen auch die Spalte `V1` und `id`, da sie ebenfalls keine Informationen bergen.

```
lm_allin_no_title <- lm(total_pr ~ ., data = mariokart_train2)
r2(lm_allin_no_title)
## # R2 for Linear Regression
##          R2: 0.521
## adj. R2: 0.441
```

Das R-Quadrat ist ja durchaus ordentlich. Schauen wir uns noch den `rmse` (die SD der Vorhersagefehler) an<sup>24</sup>:

🏆 Gut gemacht!

```
performance::rmse(lm_allin_no_title)
## [1] 20.22998
```

### 🔥 Name Clash

Im Paket `yardstick` gibt es eine Funktion namens `rmse` und im Paket `performance`, Teil des Meta-Pakets `easystats` ebenfalls. Da sind Probleme vorprogrammiert. Das ist so

<sup>24</sup>der Befehl wohnt im Paket `performance`, Teil des Metapakets `easystats`

als würde die Lehrerin rufen: “Schorsch, komm her!”. Dabei gibt es zwei Schorsche in der Klasse: Den Müllers Schorsch und den Meiers Schorsch. Sonst kommen beide, was die Lehrerin nicht will. Die Lehrerin müsste also rufen: “Müller Schorsch, komm her!”. Genau dasselbe machen wir, wenn wir das R-Paket eines Befehls mitschreiben, sozusagen den “Nachnamen” des Befehls: `paketname::funktion` ist wie `Müller::Schorsch`. In unserem Fall also: `performance::rmse` Endlich weiß R wieder, was zu tun ist!□

Sie rennen zu Ihrem Chef, der jetzt die Güte Ihrer Vorhersagen in den *restlichen* Daten bestimmen soll.

👉 Da wir dein Modell in diesem Teil des Komplett-Datensatzes *testen*, nennen wir diesen Teil das “Test-Sample”.

Ihr Chef schaut sich die Verkaufspreise im Test-Sample an:

```
mariokart_test %>%
  select(id, total_pr) %>%
  head()
```

|   | id           | total_pr |
|---|--------------|----------|
| 1 | 120477729093 | 37.02    |
| 2 | 290355805215 | 54.99    |
| 3 | 180415462166 | 56.01    |
| 4 | 180415244903 | 56.00    |
| 5 | 350261958546 | 64.95    |
| 6 | 110443013258 | 46.50    |

👉 Okay, hier sind die ersten paar echten Verkaufspreise. Jetzt mach mal deine Vorhersagen auf Basis deines besten Modells!

Hier sind Ihre Vorhersagen<sup>25</sup>:

```
lm_allin_predictions <- predict(lm_allin_no_title, newdata =
  mariokart_test)
```

Hier sind Ihre ersten paar Vorhersagen:

```
head(lm_allin_predictions)
##      1      2      3      4      5      6
## 28.62826 53.85885 53.28035 54.03619 41.75512 46.57713
```

Dies Vorhersagen fügen wir noch der Ordnung halber in die Tabelle mit den Test-Daten:

<sup>25</sup>engl. predictions; to predict: vorhersagen

```
mariokart_test <-
  mariokart_test %>%
  mutate(lm_allin_predictions = predict(lm_allin_no_title,
    ↴ newdata = mariokart_test))
```

Okay, was ist jetzt der mittlere Vorhersagefehler?

Um die Vorhersagegüte im Test-Sample auszurechnen<sup>26</sup>, nutzen wir die Funktionen des R-Paketes `yardstick`<sup>27</sup>:

```
library(yardstick)

yardstick::mae(data = mariokart_test,
               truth = total_pr, # echter Verkaufspreis
               estimate = lm_allin_predictions) # Ihre
               ↴ Vorhersage
yardstick::rmse(data = mariokart_test,
                 truth = total_pr, # echter Verkaufspreis
                 estimate = lm_allin_predictions) # Ihre
                 ↴ Vorhersage
```

| .metric | .estimator | .estimate |
|---------|------------|-----------|
| mae     | standard   | 10.01509  |

| .metric | .estimator | .estimate |
|---------|------------|-----------|
| rmse    | standard   | 13.45659  |

Ihr mittlerer Vorhersagefehler (RMSE) liegt bei ca. 13 Euro.<sup>28</sup>[Wir haben hier `yardstick::rmse` geschrieben und nicht nur `rmse`, da es sowohl im Paket `performance` (Teil des Metapakets `easystats`) als auch im Paket `yardstick` (Teil des Metapakets `tidymodels`) einen Befehl des Namens `rmse` gibt. Name-Clash-Alarm! R könnte daher den anderen ‘rmse’ meinen als Sie, was garantiert zu Verwirrung führt.<sup>28</sup>

😊 Ganz okay.

Wie ist es um das R-Quadrat Ihrer Vorhersagen bestellt?

<sup>26</sup>wir verwenden dazu die Funktionen `mae` und `rsq`

<sup>27</sup>welches Sie vielleicht noch installieren müssen.

<sup>28</sup>Entweder bei R oder bei Ihnen.

```
# `rsq` ist auch aus dem Paket yardstick:  
rsq(data = mariokart_test,  
     truth = total_pr, # echter Verkaufspreis  
     estimate = lm_allin_predictions) # Ihre Vorhersage
```

| .metric | .estimator | .estimate |
|---------|------------|-----------|
| rsq     | standard   | 0.1741705 |

👉 17%, nicht berauschend, aber immerhin!

### i Modellgüte im Test-Sample meist geringer als im Train-Sample

Wie das Beispiel zeigt, ist die Modellgüte im Test-Sample (leider) oft *geringer* als im Train-Sample. Die Modellgüte im Train-Sample ist mitunter übermäßig optimistisch. Dieses Phänomen bezeichnet man als *Overfitting*.□

### 💡 Tipp

Bevor man Vorhersagen eines Modells einreicht, bietet es sich, die Modellgüte in einem neuen Datensatz, als einem Test-Sample, zu überprüfen.□

## 11.6. Vertiefung: Das Aufteilen Ihrer Daten

### 11.6.1. Analyse- und Assessment-Sample

Wenn Sie eine robuste Schätzung der Güte Ihres Modells erfahren möchten, bietet sich folgendes Vorgehen an (vgl. Abbildung 11.13):

1. Teilen Sie Ihren Datensatz (das Train-Sample) in zwei Teile: Das sog. Validation-Sample und das sog. Assessment-Sample.
2. Berechnen Sie Ihr Modell im ersten Teil Ihres Datensatzes (dem *Validation-Sample*).
3. Prüfen Sie die Modellgüte im zweiten Teil Ihres Datensatzes (dem *Assessment-Sample*)

Diese Aufteilung Ihres Datensatzes in diese zwei Teile nennt man auch *Validierungsaufteilung* (validation split); Sie können sie z.B. so bewerkstelligen:

```
library(rsample)  
mariokart <- read_csv("daten/mariokart.csv") # Wenn die  
# CSV-Datei in einem Unterordner mit Namen "daten" liegt  
  
meine_aufteilung <- initial_split(mariokart, strata =  
# total_pr)
```

`initial_split` bestimmt für jede Zeile (Beobachtung) zufällig aus, ob diese Zeile in das Analyse- oder in das Assessment-Sample kommen soll. Im Standard werden 75% der Daten in das Analyse- und 25% in das Assessment-Sample eingeteilt<sup>29</sup>; das ist eine sinnvolle Aufteilung. Das Argument `strata` sorgt dafür, dass die Verteilung der AV in beiden Stichproben gleich ist. Es wäre nämlich blöd für Ihr Modell, wenn im Train-Sample z.B. nur die teuren, und im Test-Sample nur die günstigen Spiele landen würde.<sup>30</sup> In so einem Fall würde sich Ihr Modell unnötig schwer tun.

Im nächsten Schritt können Sie anhand anhand der von `initial_split` bestimmten Aufteilung die Daten tatsächlich aufteilen.<sup>31</sup>

```
mariokart_train <- training(meine_aufteilung)    #
  ↳ Analyse-Sample
mariokart_test <- testing(meine_aufteilung)    #
  ↳ Assessment-Sample
```

Ich persönliche nenne die Tabelle mit den Daten gerne `d_analysis` bzw. `d_assess`, das ist kürzer zu tippen und einheitlich. Sie können aber auch ein eigenes Namens-Schema nutzen; was aber hilfreich ist, ist Konsistenz in der Benamung, außerdem Kürze und aussagekräftige Namen.

### 11.6.2. Train- vs. Test-Sample

**Definition 11.2** (Train-Sample). Den Datensatz, für die Sie sowohl UV als auch AV vorliegen haben, nennt man Train-Sample. □

Das Train-Sample stellt die bekannten Daten dar; aus denen können wir lernen, d.h. unser Modell berechnen.

**Definition 11.3** (Test-Sample). Den Datensatz, für den Sie nur Daten der UV, aber nicht zu der AV vorliegen haben, nennt man Test-Sample. □

Das Test-Sample stellt das Problem der wirklichen Welt dar: Neue Beobachtungen, von denen man (noch) nicht weiß, was der Wert der AV ist.

Der Zusammenhang dieser verschiedenen, aber zusammengehörigen Arten von Stichproben ist in Abbildung 11.13 dargestellt.

---

<sup>29</sup>vgl. `help(initial_split)`

<sup>30</sup>Anderes Beispiel: In den ersten Zeilen stehen nur Kunden aus Land A und in den unteren Zeilen nur aus Land B.

<sup>31</sup>`initial_split` sagt nur, welche Zeile in welche der beiden Stichproben kommen soll. Die eigentliche Aufteilung wird aber noch nicht durchgeführt.

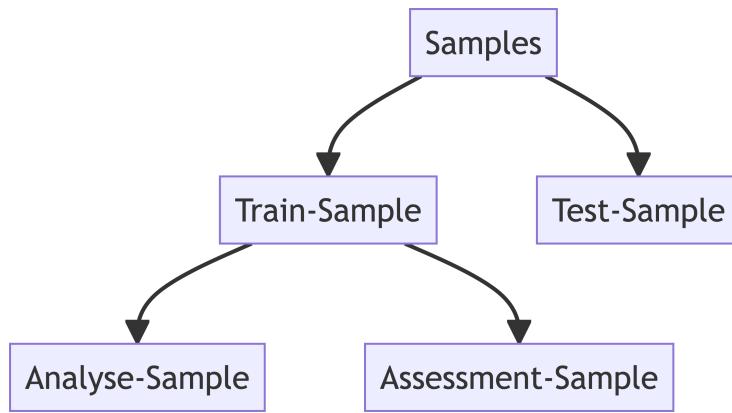


Abbildung 11.13.: Verschiedene Arten von zusammengehörigen Stichprobenarten im Rahmen einer Prognosemodellierung

## 11.7. Praxisbezug

Ein Anwendungsbezug von moderner Datenanalyse ist es vorherzusagen, welche Kunden “abwanderungsgefährdet” sind, also vielleicht in Zukunft bald nicht mehr unsere Kunden sind (“customer churn”). Es gibt eine ganze Reihe von Untersuchungen dazu, z.B. die von ([lalwani\\_customer\\_2022](#)). Die Forschis versuchen anhand von Daten und u.a. auch der linearen Regression vorherzusagen, welche Kunden abgewandert sein werden. Die Autoren berichten von einer Genauigkeit von über 80% in Ihrem (besten) Vorhersagemodell.

## 11.8. Wie man mit Statistik lügt

### 11.8.1. Pinguine drehen durch

Ein Forscher-Team untersucht Pinguine von der [Palmer Station, Antarktis](#). Das Team ist am Zusammenhang von Schnabellänge (*bill length*) und Schnabeltiefe (*bill depth*) interessiert, s. Abbildung [11.14](#).

Das Team hat in schweißtreibender eiszapfentreibender Arbeit  $n = 344$  Tiere vermessen bei antarktischen Temperaturen. Hier sind die Daten:

```

penguins_path <- paste0(
  "https://vincentarelbundock.github.io/",
  "Rdatasets/csv/palmerpenguins/penguins.csv")

penguins <- read.csv(penguins_path)
  
```

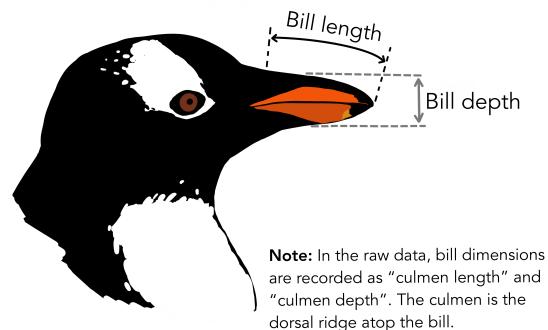
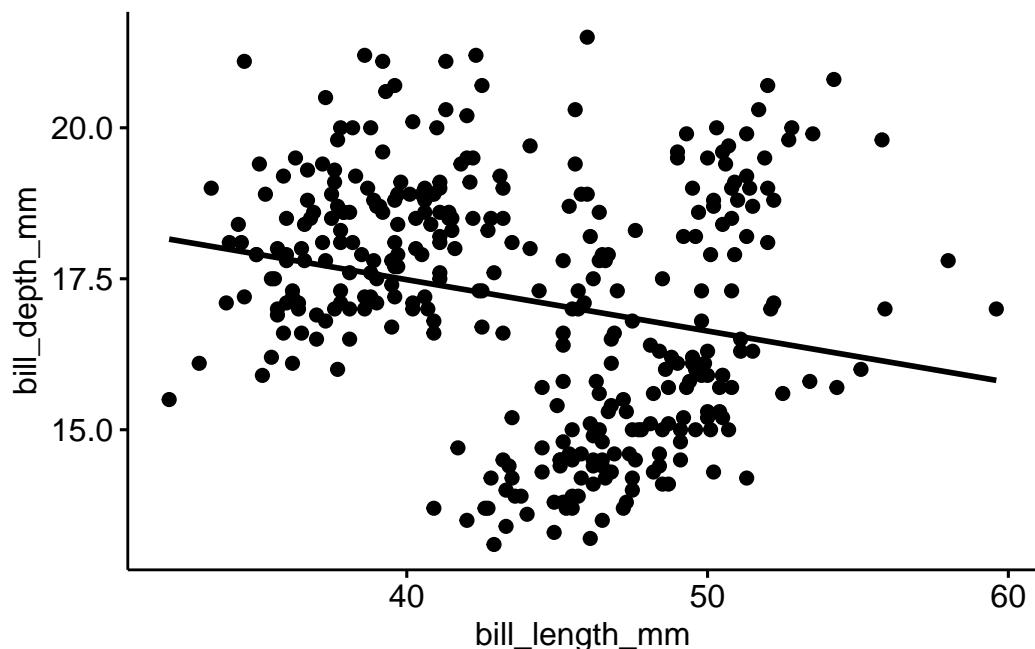


Abbildung 11.14.: Schnabellänge und Schnabeltiefe

### 11.8.2. Analyse 1: Gesamtdaten

Man untersucht, rechnet und überlegt. Ah! Jetzt haben wir es! Klarer Fall: Ein *negativer* Zusammenhang von Schnabellänge und Schnabeltiefe. Das könnte einen Nobelpreis wert sein. Schnell publizieren!

```
ggscatter(penguins, x = "bill_length_mm", y = "bill_depth_mm",
          add = "reg.line") # aus `ggpubr`
```



Hier sind die statistischen Details, s. Tabelle 11.21.

```
lm1 <- lm(bill_depth_mm ~ bill_length_mm, data = penguins)
```

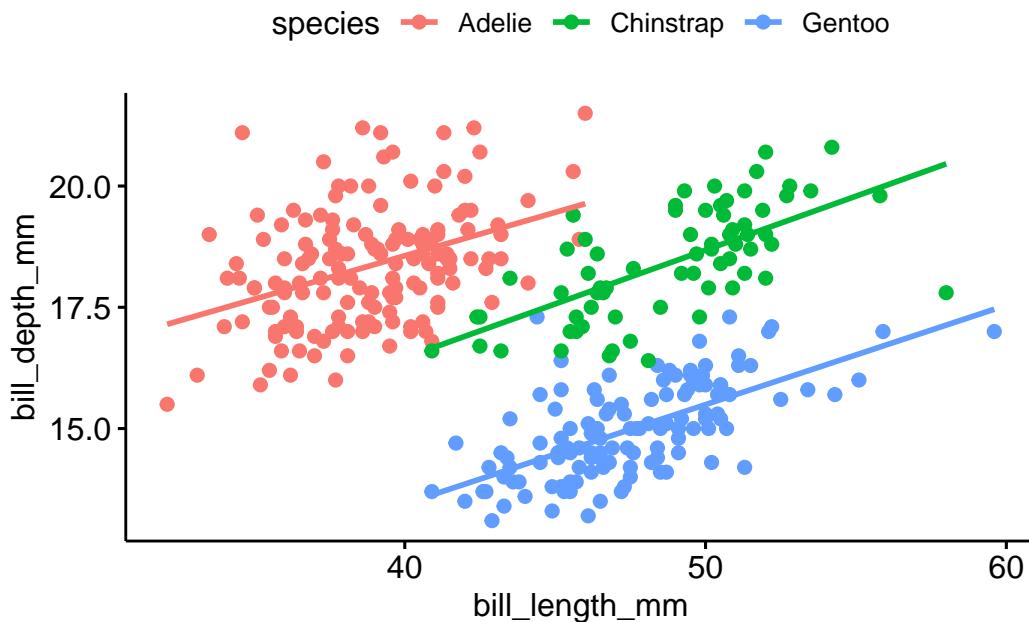
Tabelle 11.21.: Koeffizienten des Modells 1: Negativer Effekt von bill\_length\_mm

| Parameter      | Coefficient | SE   | 95% CI         | t(340) | p      |
|----------------|-------------|------|----------------|--------|--------|
| (Intercept)    | 20.89       | 0.84 | (19.23, 22.55) | 24.75  | < .001 |
| bill length mm | -0.09       | 0.02 | (-0.12, -0.05) | -4.46  | < .001 |

### 11.8.3. Analyse 2: Aufteilung in Arten (Gruppen)

Kurz darauf veröffentlicht eine verfeindete Forscherin auch einen Aufsatz zum gleichen Thema. Gleiche Daten. Aber mit *gegenteiligem* Ergebnis: Bei *jeder Rasse* von (untersuchten) Pinguinen gilt: Es gibt einen *positiven* Zusammenhang von Schnabellänge und Schnabeltiefe.

```
ggscatter(penguins, x = "bill_length_mm", y = "bill_depth_mm",
          add = "reg.line", color = "species")
```



Oh nein! Was ist hier nur los? Daten lügen nicht, oder doch?

Hier sind die statistischen Details der zweiten Analyse, s. Tabelle 11.22. Im zweiten Modell kam `species` als zweiter Prädiktor neu ins Modell (zusätzlich zur Schnabellänge).

```
lm2 <- lm(bill_depth_mm ~ bill_length_mm + species, data =
  ↵ penguins)
```

Tabelle 11.22.: Koeffizienten des Modells 2: Positiver Effekt von bill\_length\_mm

| Parameter           | Coefficient | SE   | 95% CI         | t(338) | p      |
|---------------------|-------------|------|----------------|--------|--------|
| (Intercept)         | 10.59       | 0.68 | (9.25, 11.94)  | 15.51  | < .001 |
| bill length mm      | 0.20        | 0.02 | (0.17, 0.23)   | 11.43  | < .001 |
| species (Chinstrap) | -1.93       | 0.22 | (-2.37, -1.49) | -8.62  | < .001 |
| species (Gentoo)    | -5.11       | 0.19 | (-5.48, -4.73) | -26.67 | < .001 |

🔥 Daten alleine reichen nicht

Ohne Hintergrundwissen oder ohne weitere Analysen kann *nicht* entschieden werden, welche Analyse – Gesamtdaten oder Subgruppen – die richtige ist. Nicht-experimentelle Studien können zu grundverschiedenen Ergebnissen führen, je nachdem ob Prädiktoren dem Modell hinzugefügt oder weggelassen werden. □

#### 11.8.4. Vorsicht bei der Interpretation von Regressionskoeffizienten

❗ Wichtig

Interpretiere nie Modellkoeffizienten kausal ohne ein Kausalmodell.□

Nur wenn man die Ursache-Wirkungs-Beziehungen in einem System kennt, macht es Sinn, die Modellkoeffizienten kausal zu interpretieren. Andernfalls lässt man besser die Finger von der Interpretation der Modellkoeffizienten und begnügt sich mit der Beschreibung der Modellgüte und mit Vorhersage<sup>32</sup>. Wer das nicht glaubt, der betrachte Abbildung 11.15, links.<sup>33</sup> Ei Forschi stellt das Modell m1:  $y \sim x$  auf und interpretiert dann b1: "Ist ja klar, X hat einen starken positiven Effekt auf Y!".

In der nächsten Studie nimmt das Forschi dann eine zweite Variable, group (z.B. Geschlecht) in das Modell auf: m2:  $y \sim x + g$ . Oh Schreck! Jetzt ist b1 auf einmal nicht mehr stark positiv, sondern praktisch Null, und zwar in jeder Gruppe, s. Abbildung 11.15, rechts!

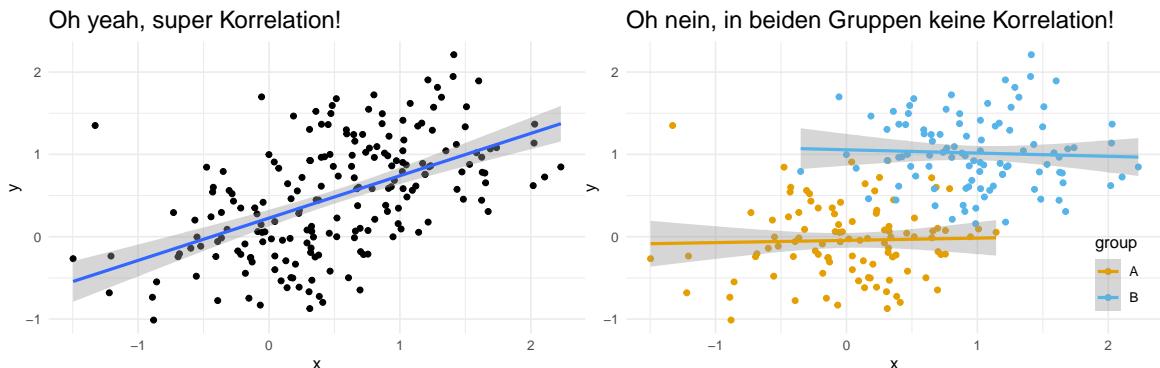
Dieses Umschwenken der Regressionskoeffizienten kann *nicht* passieren, wenn der Effekt "echt", also kausal, ist. Handelt es sich aber um "nicht echte", also nicht-kausale Zusammenhänge, um Scheinzusammenhänge also, so können sich die Modellkoeffizienten dramatisch verändern (sogar das Vorzeichen kann wechseln<sup>34</sup>), wenn man das Modell verändert, also Variablen hinzufügt oder aus dem Modell entfernt.

<sup>32</sup>synonym: Prognose

<sup>33</sup>Quelle

<sup>34</sup>das nennt man dann *Simpsons Paradox*

Wenn man die kausalen Abhängigkeiten nicht kennt, weiß man also nicht, ob die Zusammenhänge kausal oder nicht-kausal sind. Man weiß also nicht, ob die Modellkoeffizienten belastbar, robust, stichhaltig sind oder nicht.



- (a) Modell:  $y \sim x$ , starker Zusammenhang; b1 ist stark positiv  
(b) Modell:  $y \sim x + g$ , in jeder der beiden Gruppen ist der Zusammenhang praktisch Null, b1 = 0

Abbildung 11.15.: Fügt man in ein Modell eine Variable hinzu, können sich die Koeffizienten massiv ändern. In beiden Diagrammen wurden die gleichen Daten verwendet.

Man könnte höchstens sagen, dass man (wenn man die Kausalstruktur nicht kennt) die Modellkoeffizienten nur *deskriptiv* interpretiert, z.B. ‘Dort wo es viele Störche gibt, gibt es auch viele Babies’.<sup>35</sup> Leider ist unser Gehirn auf kausale Zusammenhänge geprägt: Es fällt uns schwer, Zusammenhänge nicht kausal zu interpretieren. Daher werden deskriptive Befunde immer wieder unzulässig kausal interpretiert – von Laien und Wissenschaftlern auch.

## 11.9. Fazit

In diesem Kapitel haben Sie lineare Modelle gelernt, die über einfache Modelle der Art  $y \sim x$  hinausgehen. Dazu gehören multiple Modelle, das sind Modelle mit mehr als einer UV (Prädiktor) und auch Interaktionsmodelle. Außerdem haben Sie sich mit einem Datensatz von gesamtgesellschaftlichen Nutzen beschäftigt – sehr schön. Das Fallbeispiel zum Schluss war vielleicht erhellend insofern, als dass ein gutes Modell im Train-Sample nicht (notwendig) zu guten Vorhersagen im Test-Sample führt.

### ! Wichtig

Wenn Sie dran bleiben an der Statistik, wird der Erfolg sich einstellen, s. Abbildung 11.16. □

<sup>35</sup>Das Störche-Babies-Beispiel passt auch zu Abbildung 11.15.

<sup>36</sup>Quelle: imgflip, <https://imgflip.com/memegenerator/Distracted-Boyfriend>



(a) So ging es Ihnen gestern

(b) So wird es Ihnen morgen ergehen, wenn Sie dran bleiben

Abbildung 11.16.: Statistik, Sie und Party: Gestern und (vielleicht) morgen.<sup>36</sup>

## 11.10. Fallstudien

Die folgenden Fallstudien zeigen auf recht anspruchsvollem Niveau (bezogen auf diesen Kurs) beispielhaft zwei ausführlichere Entwicklungen eines Prognosemodells.

Nutzen Sie diese Fallstudien, um sich intensiver mit der Entwicklung eines Prognosemodells auseinander zu setzen.

### 11.10.1. New Yorker Flugverspätungen 2023

[Vorhersage von Flugverspätungen](#)

### 11.10.2. Filmerlöse

[Vorhersagen von Filmerlösen](#)

## 11.11. Vertiefung

[Allison Horst](#) erklärt die lineare Regression mit Hilfe von Drachen. Sehenswert.

## 11.12. Aufgaben

Die Webseite [datenwerk.netlify.app](https://datenwerk.netlify.app)<sup>37</sup> stellt eine Reihe von einschlägigen Übungsaufgaben bereit. Sie können die Suchfunktion der Webseite nutzen, um die Aufgaben mit den folgenden Namen zu suchen:

- [interpret-koeff-lm](#)
- [Aussagen-einfache-Regr](#)
- [interpret-koeff](#)
- [regression1b](#)
- [mtcars-regr01](#)
- [regression1a](#)
- [lm1](#)
- [Regression5](#)
- [Regression6](#)
- [lm-mario1](#)
- [lm-mario2](#)
- [lm-mario3](#)
- [ausreisser1](#)
- [mario-compare-models](#)

## 11.13. Literaturhinweise

Wenn es ein Standardwerk für Regressionsanalyse geben könnte, dann vielleicht das neueste Buch von Andrew Gelman, ein bekannter Statistiker (Gelman, Hill, und Vehtari 2021b). Sein Buch ist für Sozialwissenschaftler geschrieben, also nicht für typische Nerds, hat aber deutlich mehr Anspruch als dieses Kapitel.

## 11.14. Literatur

---

<sup>37</sup><https://datenwerk.netlify.app>

**Teil IV.**

## **Abschluss**

# 12. Abschluss

## 12.1. Lernsteuerung

### 12.1.1. Standort im Lernpfad

Abb. Abbildung 1.4 den Standort dieses Kapitels im Lernpfad und gibt damit einen Überblick über das Thema dieses Kapitels im Kontext aller Kapitel.

### 12.1.2. Lernziele

*kein neuer Stoff*

Ziel dieses Kapitels ist es, den Stoff des Moduls zu wiederholen und zu konsolidieren.

### 12.1.3. Benötigte R-Pakete

```
library(tidyverse)
library(easystats)
```

### 12.1.4. Benötigte Daten

```
data(mtcars)
```

## 12.2. Herzlichen Glückwunsch!



Herzlichen Glückwunsch - Sie haben diesen Kurs abgeschlossen! Es sei denn, Sie haben nur ein bisschen durchgeschaut. Dann war es hoffentlich zumindest interessant.

## 12.3. Wie geht's weiter?

Es gibt viele weiterführende Bücher und Kurse. Ein logischer nächster Schritt ist es, sich mit Inferenzstatistik zu beschäftigen. Dazu bietet sich z.B. der Kurs [Start:Bayes!](#) an, zufälligerweise aus der Feder des gleichen Autors...

Wenn Sie sich breiter (nicht tiefer) mit Data Literacy beschäftigen wollen, bietet sich der Online-Kurs des [KI-Campus](#) an.<sup>1</sup> Es gibt viele Online-Kurse, die sich anbieten, wenn Sie im Thema moderne Datenanalyse fit werden wollen. Schauen Sie doch mal z.B. bei Anbietern wie [Coursera](#) oder ähnlichen Anbietern vorbei.<sup>2</sup>

## 12.4. Aufgabensammlungen

Auf dem [Datenwerk](#) finden Sie reichlich Aufgaben zur Prüfungsvorbereitung.<sup>3</sup>

U.a. folgende Tags sind für diesen Kurs relevant:

- [R](#)
- [association](#)
- [datawrangling](#)
- [dplyr](#)
- [lagemaße](#)
- [streuungsmaß](#)
- [variablelevles](#)
- [yacsda](#)

## 12.5. Quizze

[Hier](#) geht's zu einem Quiz zur deskriptiven Statistik (Maße der zentralen Tendenz, Variabilität, Verteilungsformen, Normalverteilung, Korrelation).<sup>4</sup>

[Hier](#) geht's zu einem Quiz zum Thema Verteilungen.<sup>5</sup>

---

<sup>1</sup><https://learn.ki-campus.org/courses/dali-basis-THK2021>

<sup>2</sup><https://www.coursera.org/specializations/data-science>

<sup>3</sup><https://datenwerk.netlify.app/>

<sup>4</sup><https://forms.gle/w7eTW3ftKy8Hv3nw8>

<sup>5</sup>Datenwerk: Verteilungen-Quiz

## 12.6. Fallstudien

### ! Wichtig

Wenn Sie mit Aufgaben “in der Wildnis” der freien Datenanalyse zu tun haben, wird es immer wieder passieren, dass Sie auf bisher unbekannte Probleme, Methoden und Lösungen stoßen. Das kann weh tun, weil man das Gefühl hat, man kennt sich nicht gut aus. Aber lassen Sie sich nicht ins Boxhorn jagen! Auf etwas Unbekanntes zu stoßen, bedeutet nichts anderes als der Beweis, dass man lernt! Es ist also eine gute Nachricht. Denn vergessen Sie nicht: Sie machen die Fallstudien nur aus einem Grund – um zu lernen, um ihre Grenzen zu erweitern, zu wachsen, schlauer zu werden, ein Handwerk zu lernen, ihre Persönlichkeit zu entfalten. Let’s grow! □

### 12.6.1. Datenvisualisierung

#### Fallstudien – NUR Datenvisualisierung

- [vis-gapminder](#)
- [vis-penguins](#)
- [vis-mtcars](#)
- [Aufgabe zur Datenvisualisierung des Diamantenpreises](#)

### 12.6.2. Explorative Datenanalyse

In diesem Abschnitt sind Fallstudien, die Methoden der deskriptiven Statistik verwenden, aufgeführt. Datenvisualisierung und Datenjudo spielt dabei auch eine (zum Teil wichtige) Rolle.

#### FALLSTUDIEN - NUR EXPLORATIVE DATENANALYSE

- [Louise E. Sinks: TidyTuesday Week 18: Portal Project](#)
- [Louise E. Sinks: TidyTuesday Week 17: London Marathon](#)
- [Louise E. Sinks: TidyTuesday Week 16: Neolithic Founder Crops](#)
- [Datenjudo mit Pinguinen](#)
- [Data-Wrangling-Aufgaben zur Lebenserwartung](#)
- [Case study: data vizualization on flight delays using tidyverse tools](#)
- [Fallstudie Flugverspätungen - EDA](#)
- [Fallstudie zur EDA: Top-Gear](#)
- [Fallstudie zur EDA: OECD-Wellbeing-Studie](#)
- [Fallstudie zur EDA: Movie Rating](#)

- Fallstudie zur EDA: Women in Parliament
- Finde den Tag mit den meisten Flugverspätungen, Datensatz ‘nycflights13’
- Cleaning and visualizing genomic data: a case study in tidy analysis
- Tidyverse Case Study: Exploring the Billboard Charts
- Analyse einiger RKI-Coronadaten: Eine reproduzierbare Fallstudie
- OpenCaseStudies - Health Expenditure
- Open Case Studies: School Shootings in the United States - includes dashboards
- Open Case Studies: Disparities in Youth Disconnection
- YACSDA Seitensprünge
- The Open Case Study Search provides a nice collection of helpful case studies.
- ifes@FOM Fallstudienseite

### **12.6.3. Lineare Modelle**

#### **FALLSTUDIEN - NUR LINEARE MODELLE**

- Beispiel für Prognosemodellierung 1, grundlegender Anspruch, Video
- Beispiel für Ihre Prognosemodellierung 2, mittlerer Anspruch
- Beispiel für Ihre Prognosemodellierung 3, hoher Anspruch
- Fallstudie: Modellierung von Flugverspätungen 2023 (mittlerer Anspruch)
- Fallstudie: Modellierung von Flugverspätungen 2023 (höherer Anspruch)
- Fallstudie: Modellierung von Flugverspätungen 2013
- Modelling movie successes: linear regression
- Movies
- Fallstudie Einfache lineare Regression in Base-R, Anfängerniveau, Kaggle-Competition TMDB
- Fallstudie Sprit sparen
- Fallstudie zum Beitrag verschiedener Werbeformate zum Umsatz; eine Fallstudie in Python, aber mit etwas Erfahrung wird man den Code einfach in R umsetzen können (wenn man nicht in Python schreiben will)
- Practical Linear Regression with R: A case study on diamond prices
- Case Study: Italian restaurants in NYC
- Vorhersage-Modellierung des Preises von Diamanten
- Modellierung Diamantenpreis 2

- YACSDA Seitensprünge

## 12.7. FAQ

Werfen Sie auch einen Blick in typische R-Fragen.

### 12.7.1. SD berechnen

FRAGE: Macht es einen Unterschied, ob man dafür den Befehl `summary()` oder den Befehl `sd()` verwendet? Bei mir kommen da nämlich unterschiedliche Zahlen raus.

ANTWORT: `summary()` gibt nicht SD aus, sondern nur den IQR ( $IQR = Q3 - Q1$ ).

```
data(mtcars)
sd(mtcars$mpg)
## [1] 6.026948
summary(mtcars$mpg)
##      Min. 1st Qu. Median      Mean 3rd Qu.      Max.
##     10.40    15.43   19.20    20.09   22.80    33.90
```

### 12.7.2. count vs. filter

FRAGE: Wann benutzt man `count()` und wann `filter()`?

ANTWORT: Mit `filter` plus dem Zählen der übrig gebliebenen Zeilen erreicht man etwas Ähnliches wie mit `count`:

```
mtcars |>
  filter(am == 0) |>
  nrow()
## [1] 19
```

```
mtcars |>
  count(am)
```

| am | n  |
|----|----|
| 0  | 19 |
| 1  | 13 |

### 12.7.3. 1000

FRAGE: gibt es einen Unterschied zwischen  $10^3$  und 1e3? Es kommen nämlich unterschiedliche Ergebnisse raus.

ANTWORT: Nein, beide Schreibweisen meinen das Gleiche, nämlich die Zahl 1000.

```
10^3 == 1000
## [1] TRUE
1e3 == 1000
## [1] TRUE
```

## 12.8. Literaturhinweise

Diese [Literaturliste](#) empfiehlt Ihnen Lehrbücher zu grundlegenden Themen der Datenanalyse (mit R).<sup>6</sup>

---

<sup>6</sup><https://www.zotero.org/groups/4583286/intro-stats/library>

## 13. Definitionen

- **Abweichungsrechteck:** Definition [9.1](#)
- **Argumente einer Funktion:** Definition [4.4](#)
- **Ausprägung:** Definition [3.8](#)
- **Balkendiagramm:** Definition [6.2](#)
- **Beobachtungseinheit:** Definition [3.6](#)
- **Binäre Variable:** Definition [11.1](#)
- **Boxplot:** Definition [6.8](#)
- **Data-Dictionary:** Definition [3.4](#)
- **Dataframe:** Definition [4.6](#)
- **Hello, Daten:** Definition [3.3](#)
- **Datenjudo:** Definition [5.1](#)
- **Dezile:** Definition [7.6](#)
- **Dichtediagramm:** Definition [6.4](#)
- **Extremwert:** Definition [7.3](#)
- **Fehlerstreuung:** Definition [10.2](#)
- **Funktion:** Definition [4.2](#)
- **Gerade:** Definition [10.1](#)
- **# Histogramm:** Definition [6.3](#)
- **Interquartilsabstand:** Definition [8.4](#)
- **Kovarianz:** Definition [9.2](#)
- **Lagemaß:** Definition [7.8](#)
- **Linearer Zusammenhang:** Definition [6.6](#)
- **Lineares Modell:** Definition [7.2](#)
- **Mittlere Absolutabweichung:** Definition [8.3](#)
- **Median:** Definition [7.4](#)

- **## Modelle:** Definition [3.11](#)
- **Mittelwert:** Definition [7.1](#)
- **Entstehung einer Normalverteilung:** Definition [6.5](#)
- **Pfeife:** Definition [5.2](#)
- **Punktmodell:** Definition [7.9](#)
- **Quantile:** Definition [7.7](#)
- **Quartile:** Definition [7.5](#)
- **Korrelationskoeffizient r:** Definition [9.3](#)
- **R-Quadrat:** Definition [10.3](#)
- **Reproduzierbarkeit:** Definition [4.1](#)
- **Residuum:** Definition [3.2](#)
- **Standardabweichung:** Definition [8.6](#)
- **Skalenniveau:** Definition [3.10](#)
- **Statistik:** Definition [3.1](#)
- **Streuungsmaße:** Definition [8.1](#)
- **Test-Sample:** Definition [11.3](#)
- **Tidy Data:** Definition [3.9](#)
- **Train-Sample:** Definition [11.2](#)
- **Variable:** Definition [8.5](#)
- **Varianz:** Definition [8.5](#)
- **Vektorielles Rechnen:** Definition [4.5](#)
- **Vektor:** Definition [4.3](#)
- **Verteilung:** Definition [6.1](#)
- **Wert:** Definition [3.7](#)
- **z-Werte:** Definition [8.8](#)
- **Zentrieren :**Definition [8.7](#)
- **Richtig und Stärke eines Zusammenhang:** Definition [6.7](#)

# Literatur

- Bortz, Jürgen, und Christof Schuster. 2010. *Statistik Für Human- Und Sozialwissenschaftler*. Springer-Lehrbuch. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-12770-0>.
- Bowne-Anderson, Hugo. 2018. „What Data Scientists Really Do, According to 35 Data Scientists“. *Harvard Business Review*, 15. August 2018. <https://hbr.org/2018/08/what-data-scientists-really-do-according-to-35-data-scientists>.
- Broman, Karl W., und Kara H. Woo. 2018. „Data Organization in Spreadsheets“. *The American Statistician* 72 (1): 2–10. <https://doi.org/10.1080/00031305.2017.1375989>.
- Cetinkaya-Rundel, Mine, und Johanna Hardin. 2021. *Introduction to Modern Statistics*. <https://openintro-ims.netlify.app/>.
- Cohen, Jacob, Patricia Cohen, Stephen G. West, und Leona S. Aiken. 2003. *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences, 3rd Ed.* Applied Multiple Regression-/Correlation Analysis for the Behavioral Sciences, 3rd Ed. Mahwah, NJ, US: Lawrence Erlbaum Associates Publishers.
- Downey, Allen. 2023. *Probably Overthinking It: How to Use Data to Answer Questions, Avoid Statistical Traps, and Make Better Decisions*. Chicago ; London: The University of Chicago Press.
- Fisher, Danyel, und Miriah Meyer. 2018. *Making Data Visual: A Practical Guide to Using Visualization for Insight*. First edition. Beijing, [China]: O'Reilly.
- Forum, World Economic. 2020. „The Future of Jobs Report 2020“. CH-1223 Cologny/Geneva Switzerland: World Economic Forum. [https://www3.weforum.org/docs/WEF\\_Future\\_of\\_Jobs\\_2020.pdf](https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf).
- Gelman, Andrew, Jennifer Hill, und Aki Vehtari. 2021a. *Regression and Other Stories*. Analytical Methods for Social Research. Cambridge: Cambridge University Press.
- . 2021b. *Regression and Other Stories*. Analytical Methods for Social Research. Cambridge: Cambridge University Press.
- Goren, Andy, Sergio Vaño-Galván, Carlos Gustavo Wambier, John McCoy, Alba Gomez-Zubiaur, Oscar M. Moreno-Arrones, Jerry Shapiro, u. a. 2020. „A Preliminary Observation: Male Pattern Hair Loss Among Hospitalized COVID-19 Patients in Spain – A Potential Clue to the Role of Androgens in COVID-19 Severity“. *Journal of Cosmetic Dermatology* 19 (7): 1545–47. <https://doi.org/10.1111/jocd.13443>.
- Haug, Severin, Raquel Paz Castro, Min Kwon, Andreas Filler, Tobias Kowatsch, und Michael P. Schaub. 2015. „Smartphone Use and Smartphone Addiction among Young People in Switzerland“. *Journal of Behavioral Addictions* 4 (4): 299–307. <https://doi.org/10.1556/2006.4.2015.037>.
- Ismay, Chester, und Albert Young-Sun Kim. 2020. *Statistical Inference via Data Science: A Modern-Dive into R and the Tidyverse*. Chapman & Hall/CRC the R Series. Boca Raton: CRC Press / Taylor & Francis Group. <https://moderndive.com/>.

- Kaplan, Daniel T. 2009. *Statistical Modeling: A Fresh Approach*. Scotts Valley, Calif.: CreateSpace. <https://dtkaplan.github.io/SM2-bookdown/>.
- Kwon, Min, Dai-Jin Kim, Hyun Cho, und Soo Yang. 2013. „The Smartphone Addiction Scale: Development and Validation of a Short Version for Adolescents“. *PloS One* 8 (12): e83558. <https://doi.org/10.1371/journal.pone.0083558>.
- Lyon, Aidan. 2014. „Why Are Normal Distributions Normal?“ *The British Journal for the Philosophy of Science* 65 (3): 621–49. <https://doi.org/10.1093/bjps/axs046>.
- MacKay, R. J., und R. W. Oldford. 2000. „Scientific Method, Statistical Method and the Speed of Light“. *Statistical Science* 15 (3): 254–78. <https://doi.org/10.1214/ss/1009212817>.
- Messerli, Franz H. 2012. „Chocolate Consumption, Cognitive Function, and Nobel Laureates“. *New England Journal of Medicine* 367 (16): 1562–64. <https://doi.org/10.1056/NEJMoa1211064>.
- Mittag, Hans-Joachim, und Katharina Schüller. 2020. *Statistik: Eine Einführung mit interaktiven Elementen*. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-61912-4>.
- Mulukom, Valerie van, Barbara Muzzolini, Bastiaan Rutjens, Caspar J. van Lissa, und Miguel Farias. 2020. „Psychological Impact of COVID-19 Pandemic“, März. <https://doi.org/10.17605/OSF.IO/TSJNB>.
- Obels, Pepijn, Daniël Lakens, Nicholas A. Coles, Jaroslav Gottfried, und Seth A. Green. 2020. „Analysis of Open Data and Computational Reproducibility in Registered Reports in Psychology“. *Advances in Methods and Practices in Psychological Science* 3 (2): 229–37. <https://doi.org/10.1177/2515245920918872>.
- Oestreich, Markus, und Oliver Romberg. 2014. *Keine Panik vor Statistik!: Erfolg und Spaß im Horrorfach nichttechnischer Studiengänge*. Wiesbaden: Springer Fachmedien Wiesbaden. <https://doi.org/10.1007/978-3-658-04605-7>.
- Okabe, Masataka, und Kei Ito. 2023. „Color Universal Design (CUD) / Colorblind Barrier Free“. 2023. <https://jfly.uni-koeln.de/color/>.
- Plesser, Hans E. 2018. „Reproducibility vs. Replicability: A Brief History of a Confused Terminology“. *Frontiers in Neuroinformatics* 11 (Januar): 76. <https://doi.org/10.3389/fninf.2017.00076>.
- Poldrack, Russell A. 2023. *Statistical Thinking: Analyzing Data in an Uncertain World*. Princeton: Princeton University Press. <https://statsthinking21.github.io/statsthinking21-core-site/>.
- Roser, Max, Cameron Appel, und Hannah Ritchie. 2013. „Human Height“. *Our World in Data*. <https://ourworldindata.org/human-height>.
- Rothstein, Hannah R. 2014. „Publication Bias“. In *Wiley StatsRef: Statistics Reference Online*. John Wiley & Sons, Ltd. <https://doi.org/10.1002/9781118445112.stat07071>.
- Sauer, Sebastian. 2019. *Moderne Datenanalyse mit R: Daten einlesen, aufbereiten, visualisieren und modellieren*. 1. Auflage 2019. FOM-Edition. Wiesbaden: Springer. <https://www.springer.com/de/book/9783658215866>.
- Schwaiger, Elizabeth, und Rameen Tahir. 2022. „The Impact of Nomophobia and Smartphone Presence on Fluid Intelligence and Attention“. *Cyberpsychology: Journal of Psychosocial Research on Cyberspace* 16 (1). <https://doi.org/10.5817/CP2022-1-5>.
- Simmons, Joseph P., Leif D. Nelson, und Uri Simonsohn. 2011. „False-Positive Psychology: Undisclosed Flexibility in Data Collection and Analysis Allows Presenting Anything as Significant“. *Psychological Science* 22 (11): 1359–66. <https://doi.org/10.1177/0956797611417632>.
- Stigler, Stephen M. 2016. *The Seven Pillars of Statistical Wisdom*. Cambridge, Massachusetts: Harvard University Press.
- Wickham, Hadley, und Garrett Grolemund. 2018. *R für Data Science: Daten importieren, bereinigen,*

- umformen, modellieren und visualisieren.* Übersetzt von Frank Langenau. 1. Auflage. Heidelberg: O'Reilly. <https://r4ds.had.co.nz/index.html>.
- Wilke, C. 2019. *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures*. First edition. Sebastopol, CA: O'Reilly Media. <https://clauswilke.com/dataviz/>.