# Vorlesung Statistik A
# Explorative Datenanalyse

Florian Wickelmaier

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN



Wintersemester 2026/2027

# Deskriptive Statistik

## Merkmale

- Bei einer empirischen Erhebung interessiert man sich für ein oder auch mehrere Merkmale
- Die verschiedenen Ergebnisse, die bei der Beobachtung eines Merkmals auftreten, heißen Merkmalsausprägungen
- Beispiele für Mengen von Merkmalsausprägungen
  - Geschlecht: {weiblich, männlich}
  - Farbe: {rot, weiß, grün, blau, ...}
  - Expertise bei Softwarenutzung: {Anfänger, gelegentlicher Nutzer, Experte}
  - Alter (in Jahren): $\{1, 2, 3, \ldots\}$
  - Körpergröße: $\{x \mid x > 0\}$
  - Gewürfelte Augenzahl: $\{1, 2, 3, 4, 5, 6\}$
  - Noten: $\{1, 2, 3, 4, 5, 6\}$

# Merkmalstypen

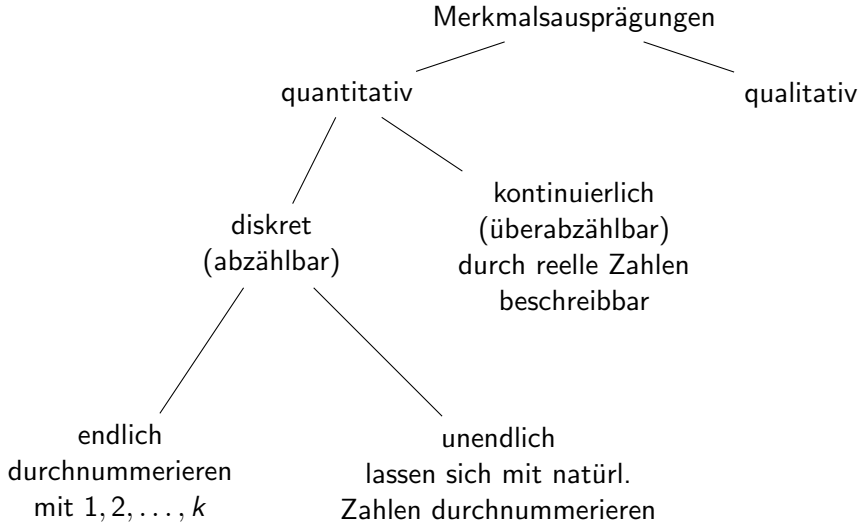
Quantitative Merkmale

- Zahlenmäßige Charakterisierung der Merkmalsausprägungen
- Eigenschaften der Zahlen (z. B. Anordenbarkeit)
- Beispiele: Alter, Körpergröße, gewürfelte Augenzahl, Noten

Qualitative Merkmale

- "Artmäßige" Charakterisierung der Merkmalsausprägungen (Qualia)
- Beispiele: Geschlecht, Farbe, Expertise

# Merkmalstypen



# Skalenniveau

- Einer empirischen Beobachtung (einem Datum) lässt sich häufig auf mehr als eine Weise eine Maßzahl zuordnen (Problem der Eindeutigkeit)
  - Beispielsweise kann bei Längenmessung die Angabe der Maßzahlen in unterschiedlichen Einheiten (z. B. Meter, Fuß) erfolgen
- Dadurch werden verschiedene Repräsentationen definiert, die zusammen eine Skala bilden
- Der Wechsel von einer Repräsentation zu einer anderen wird durch Abbildungen, sogenannte zulässige Transformationen, beschrieben
- Aufgrund der jeweils zulässigen Transformationen werden *Skalenniveaus* unterschieden, ein Begriff der durch Stevens (1946) in die Wissenschaft eingeführt wurde

# Zulässige Transformationen

| Skalenniveau | zulässige Transformationen | Beispiele |
|---|---|---|
| Nominalskala | injektive Abbildungen<br>$f(x) = f(y) \Rightarrow x = y$ | Startnummern, Postleitzahl |
| Ordinalskala | streng monotone Abbildungen<br>$x > y \Rightarrow f(x) > f(y)$ | Härteskala für Mineralien |
| Intervallskala | affine Abbildungen<br>$f(x) = p \cdot x + q,\ p > 0$ | Temperatur in Celsius bzw. Fahrenheit |
| Verhältnisskala | Ähnlichkeitsabbildungen<br>$f(x) = p \cdot x,\ p > 0$ | Länge, Masse, Zeit |
| Absolutskala | identische Abbildung<br>$f(x) = x$ | Wahrscheinlichkeit, Anzahl |

Intervall-, Verhältnis- und Absolutskala bilden *metrische* Skalen

# Skalenniveau

Nominalskala

- Es wird lediglich die Verschiedenheit der Merkmalsausprägungen zum Ausdruck gebracht
- Keine natürliche Rangordnung
- Beispiele: Geschlecht, Farbe

Ordinalskala

- Merkmalsausprägungen können in Rangordnung gebracht werden
- Abstände können nicht sinnvoll interpretiert werden
- Beispiele: Expertise, Noten

# Skalenniveau

Intervallskala

- Abstände zwischen Merkmalsausprägungen lassen sich sinnvoll vergleichen
- Es gibt keinen natürlichen Nullpunkt

Verhältnisskala

- Verhältnisse interpretierbar
- Maßeinheit nicht festgelegt
- Es existiert ein natürlicher Nullpunkt
- Beispiele: Alter, Körpergröße

Absolutskala

- Nullpunkt und Skaleneinheit sind natürlich festgelegt
- Beispiel: Gewürfelte Augenzahl

# Skalenniveau und Bedeutsamkeit

Das Problem der Bedeutsamkeit entsteht als Folge der mangelnden Eindeutigkeit einer Repräsentation

- Der Wahrheitswert einer theoretischen Aussage kann sich ändern, wenn man von einer Repräsentation zu einer anderen möglichen Repräsentation übergeht
- In einem solchen Fall kann die entsprechende Aussage keinen empirischen Gehalt haben, da ihr Wahrheitswert von der willkürlichen Auswahl einer bestimmten Repräsentation abhängt

Als *bedeutsam (meaningful)* sind daher nur solche theoretischen Aussagen ausgezeichnet, deren Wahrheitswert unabhängig davon ist, welche der möglichen Repräsentationen zugrundegelegt wird

Es sind genau diejenigen Aussagen bedeutsam, deren Wahrheitswert invariant ist unter den zulässigen Transformationen

# Bedeutsamkeit: Konsequenzen für Statistik

Wenn die statistische Analyse von Maßzahlen zu empirisch bedeutsamen Ergebnissen führen soll, müssen diese deren Skalenniveau berücksichtigen

- Summen-, Differenzen- und Mittelwertbildung können nur bei metrischen Skalen zu empirisch bedeutsamen Ergebnissen führen, müssen es aber nicht zwangsläufig
- Produkt- und Quotientenbildung können nur bei Verhältnisskalen zu empirisch bedeutsamen Ergebnissen führen, müssen es aber nicht zwangsläufig

Je höher das Skalenniveau, desto mehr Information steckt in den Maßzahlen, die bei statistischen Analysen erhalten bleiben muss

- Beispielsweise erhalten willkürliche Nullpunktverschiebungen nicht die Eigenschaften von Verhältnisskalen

# Einführung in R

R als Taschenrechner

- Einfache Rechenoperationen können direkt über die Kommandozeile eingegeben werden

```
2 + 2   # plus
3 - 7   # minus
2 * 2   # mal
5 / 4   # geteilt
log(3)  # natuerlicher Logarithmus
exp(1)  # Exponentialfunktion
```

(Eingaben ab dem #-Zeichen werden von R ignoriert)

- In längeren Ausdrücken können Klammern verwendet werden

```
sqrt(2) * ((5 - 1/6)^2 - pi/2^(1/3))  # pi = 3.14
```

# Zuweisungen

Ergebnisse von Rechenoperationen oder Funktionsaufrufen können in Variablen gespeichert werden. Der Zuweisungsoperator ist <- (kleiner als + minus)

```
x <- sqrt(2)
y <- x^2
```

Um den Inhalt einer Variable anzuzeigen, muss der Variablenname eingegeben werden

```
y
[1] 2
```

Variablennamen dürfen aus Buchstaben, Ziffern und Punkten bestehen (z. B. `fehler.item6`), aber nicht mit einer Ziffer beginnen; Groß- und Kleinschreibung muss beachtet werden

# Rechnen mit Vektoren

Vektoren werden mit der Funktion `c()` gebildet (combine)

```
gewicht <- c(60, 72, 57, 90, 95, 72)
gewicht
[1] 60 72 57 90 95 72
```

Rechenoperationen werden für jedes Element des Vektors ausgeführt

```
(gewicht - mean(gewicht))^2  # quadrierte Abweichung
[1] 205.444   5.444 300.444 245.444 427.111   5.444
```

Viele R-Funktionen akzeptieren Vektoren als Argumente

```
log(gewicht)
[1] 4.094 4.277 4.043 4.500 4.554 4.277
```

# Das Hilfesystem in R

R bietet eine Reihe von Hilfemechanismen, die entweder über den Menüpunkt "Hilfe" aufgerufen werden oder über Eingabe der folgenden Aufrufe

- Die textbasierte Dokumentation wird durch ? aufgerufen, so zeigt

  ```
  ?rnorm
  ```

  die Dokumentation der Funktion `rnorm()` an (am Ende der Seite finden sich zumeist Anwendungsbeispiele, die man direkt kopieren kann)
- Alle Funktionen mit dem Ausdruck "norm" im Namen bzw. im Hilfetext werden ausgegeben durch

  ```
  apropos("norm")
  help.search("norm")  # oder: ??norm
  ```

# Dokumentation

Ein HTML-basiertes Hilfesystem erhält man über

```
help.start()
```

R wird mit mehreren Manualen geliefert, die über `help.start()` im Browser zugänglich sind, aber auch im Pdf-Format heruntergeladen werden können

Besonders lesenswert sind

- An Introduction to R (Einführung mit vielen Beispielen)
- R Data Import/Export (Einlesen und Ausgeben von Daten)

Weitere Quellen für Information bieten (ebenfalls über `help.start()` erreichbar)

- Frequently Asked Questions
- FAQ for Windows

# Internet-Ressourcen

Eine Fülle von Informationen bietet auch die R-Website
`https://www.r-project.org/`

- Unter R Project/Search
  R site search, searchable mail archives (insbesondere R-help)
- Unter Documentation
  R-Journal, weitere (kürzere) Dokumente (z. B. "Notes on the use of R for psychology experiments and questionnaires")

Literatur

Dalgaard, P. (2008). *Introductory statistics with R.* New York: Springer.

Verzani, J. (2005). *Using R for introductory statistics.* Boca Raton, FL: Chapman & Hall/CRC.

Venables, W. N. & Ripley, B. D. (2002). *Modern applied statistics with S.* New York: Springer.

# Häufigkeitsverteilungen bei qualitativen Merkmalen

Wir betrachten ein Merkmal $A$ mit seinen Merkmalsausprägungen

$$A = \{a_1, \ldots, a_j, \ldots, a_m\}$$

Die mehrfache (im vorliegenden Fall $n$-fache) empirische Beobachtung des Merkmals $A$ führt zu Beobachtungswerten $x_1, \ldots, x_i, \ldots, x_n$

- Jeder Beobachtungswert $x_i$, für $i = 1, \ldots, n$, stimmt mit einer Merkmalsausprägung überein, also $x_i = a_j$ für genau ein $j \in \{1, \ldots, m\}$

Die Beobachtungswerte werden zu einem $n$-Tupel

$$\mathbf{x} = (x_1, \ldots, x_i, \ldots, x_n)$$

zusammengefasst

# Stichprobe

Das $n$-Tupel $\mathbf{x}$ der Beobachtungswerte heißt

- Beobachtungsreihe,
- Urliste, oder
- Stichprobe

# Beispiel: Blütenfarbe





Merkmal

$$A = \{\text{rot, rosa, weiß}\}$$

Anzahl Beobachtungswerte

$$n = 182 \qquad \text{(Nachkommen einer bestimmten Kreuzung)}$$

Urliste

$$\mathbf{x} = (\text{rosa, rot, weiß, rosa, rosa, } \ldots, \text{ rot}),$$

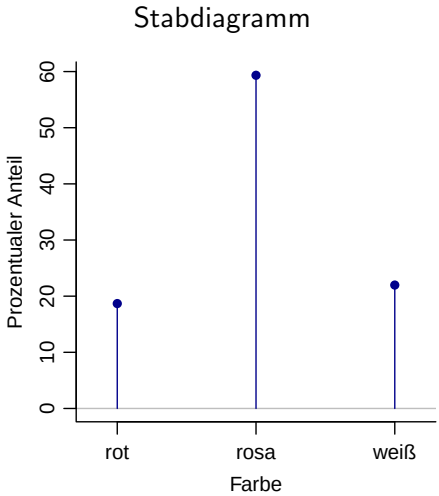
also $x_1 = \text{rosa}$, $x_2 = \text{rot}$, $x_3 = \text{weiß}$, $\ldots$, $x_{182} = \text{rot}$

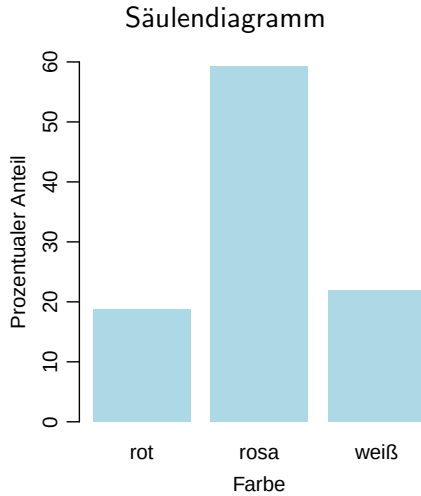
# Beispiel: Häufigkeitstabelle

- Absolute Häufigkeit von $a_j$: $h_j = h(a_j)$
- Relative Häufigkeit von $a_j$: $f_j = f(a_j) = \frac{h_j}{n}$
- Prozentualer Anteil der Merkmalsausprägung $a_j$: $100\% \cdot f_j$

| $A$ | $h_j = h(a_j)$ | $f_j = f(a_j)$ | $100\% \cdot f_j$ |
|---|---|---|---|
| $a_1 =$ rot | 34 | 0.187 | 18.7 |
| $a_2 =$ rosa | 108 | 0.593 | 59.3 |
| $a_3 =$ weiß | 40 | 0.220 | 22.0 |
| $\sum$ | 182 | 1.000 | 100.0 |

# Beispiel: Grafische Darstellung



# Häufigkeitsverteilungen von diskreten Merkmalen

Bemerkungen zur Notation

- Die Summe und das Produkt der Beobachtungswerte $x_1, x_2, \ldots, x_n$ werden in abkürzender Schreibweise notiert durch

$$\sum_{i=1}^{n} x_i = x_1 + x_2 + \ldots + x_n$$

bzw.

$$\prod_{i=1}^{n} x_i = x_1 \cdot x_2 \cdot \ldots \cdot x_n$$

# Häufigkeitsverteilungen von diskreten Merkmalen

Eigenschaften

- Absolute Häufigkeit $h_j$

$$0 \leq h_j \leq n \text{ für alle } j = 1, \ldots, m \quad \text{und} \quad \sum_{j=1}^{m} h_j = h_1 + \ldots + h_m = n$$

- Relative Häufigkeit $f_j$

$$0 \leq f_j \leq 1 \text{ für alle } j = 1, \ldots, m \quad \text{und} \quad \sum_{j=1}^{m} f_j = 1$$

# Datenstrukturen in R

## Funktionen und Argumente

R wird vorwiegend über Funktionen angesprochen,
die bestehen aus

- einem Namen
- einem Klammernpaar
- den Argumenten (keines, eines oder mehrere)
- einem Rückgabewert (sichtbar, unsichtbar, `NULL`)

Beispiel

```
groesse <- c(1.75, 1.80, 1.65, 1.90, 1.74, 1.91)
plot(groesse, gewicht)
```

Argumente können auf voreingestellte Werte gesetzt sein

- Welche dies sind, erfährt man über `?plot.default` (wird von `plot` aufgerufen, wenn für das erste Argument keine spezielle plot-Methode existiert)

# Datenstrukturen in R

## Funktionen und Argumente

Argumente werden

- entweder ohne Namen übergeben (in der definierten Reihenfolge)
  → positional matching
- oder mit Namen (in beliebiger Reihenfolge)
  → keyword matching

Beispiel

```
plot(groesse, gewicht, pch = 16, col = "blue")
```

Auch wenn keine Argumente übergeben werden, müssen Klammern gesetzt werden, z. B. `ls()`, `dir()`, `getwd()`

Eingabe des Namens der Funktion ohne Klammern zeigt den R-Code der Funktion an

# Vektoren

Vektoren haben einen bestimmten Typ

Außer numerischen Vektoren gibt es

- Zeichenvektoren mit Elementen in Anführungszeichen

```
c("Vp01", "Vp02", "Vp03")
[1] "Vp01" "Vp02" "Vp03"
```

- logische Vektoren mit den Werten `TRUE` und `FALSE` (fehlende Werte werden mit `NA` bezeichnet)

```
c(TRUE, TRUE, FALSE, TRUE)
[1]  TRUE  TRUE FALSE  TRUE
```

Logische Vektoren resultieren oft aus einer Vergleichsoperation

```
gewicht > 60
[1] FALSE  TRUE FALSE  TRUE  TRUE  TRUE
```

# Vektoren

Vektoren werden erzeugt

- indem man die Elemente mit `c()` zusammenfasst
- durch Sequenzen

```
0:10                  # 11 Zahlen von 0 bis 10
seq(1.65, 1.90, .05)  # Folge in Schritten von 0.05
```

- durch Wiederholung von Elementen

```
rep(1:5, 3)
[1] 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5

rep(c("high", "low"), each = 3)
[1] "high" "high" "high" "low"  "low"  "low"

rep(c("w", "m"), c(5, 2))
[1] "w" "w" "w" "w" "w" "m" "m"

paste("reiz", 1:5, sep = "")
[1] "reiz1" "reiz2" "reiz3" "reiz4" "reiz5"
```

# Faktoren

Faktoren sind Datenstrukturen für kategoriale Variablen, z. B. Diagnose, sozialer Status, Geschlecht

```
status <- factor(c("niedrig", "mittel", "hoch"))
[1] niedrig mittel  hoch
Levels: hoch mittel niedrig
```

Werden Faktoren aus Zeichenvektoren gebildet, dann sind die Faktorstufen alphabetisch geordnet ("default")

Die Ordnung der Stufen wird über das Argument `levels` beeinflusst

```
stat2 <- factor(status,
                levels = c("niedrig", "mittel", "hoch"))
[1] niedrig mittel  hoch
Levels: niedrig mittel hoch
```

# Häufigkeitstabellen in R

Urliste als Faktor

```
x <- rep(c("rot", "rosa", "weiss"),
         c(34, 108, 40)) |>
  factor(levels = c("rot", "rosa", "weiss"))
```

Damit ist sichergestellt, dass die Kategorien in der gewünschten Reihenfolge angegeben werden

Berechnung der absoluten Häufigkeiten über die Funktion `table()`

```
h <- table(x)
```

Berechnung der relativen Häufigkeiten

```
f <- h / sum(h)
```

# Stabdiagramm in R

Stabdiagramm mit Funktion `plot()`

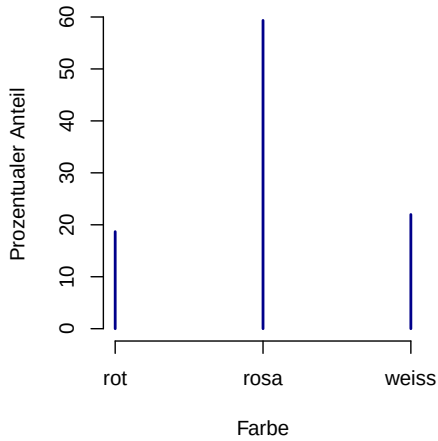
```
plot(100 * f,
     type = "h",
     col = "darkblue",
     xlab = "Farbe", ylab = "Prozentualer Anteil")
```

Argumente

- `100 * f` gibt den prozentualen Anteil an
- `type` spezifiziert den Typ des Diagramms, wobei der Wert `h` den Typ "Histogramm" kennzeichnet
- Die Farbe der Histogramm-Linien wird durch `col` definiert
- Mit `xlab` und `ylab` kann man die Beschriftung der Achsen festlegen

# Stabdiagramm in R

```
plot(100 * f,
     type = "h", col = "darkblue",
     xlab = "Farbe",
     ylab = "Prozentualer Anteil")
```



# Säulendiagramm in R

Säulendiagramm mit Funktion `barplot()`

```
barplot(100 * f,
        col = "lightblue",
        xlab = "Farbe", ylab = "Prozentualer Anteil",
        ylim = c(0, 60)
)
```

Argumente

- `100 * f` gibt den prozentualen Anteil an
- Die Farbe der Säulen wird durch `col` definiert
- Mit `ylim` wird der Bereich der Ordinate ($y$-Achse) festgelegt

# Säulendiagramm in R

```
barplot(100 * f, col = "lightblue",
        xlab = "Farbe",
        ylab = "Prozentualer Anteil",
        ylim = c(0, 60))
```

