

# Eine Einführung in R: Deskriptive Statistiken und Graphiken

Bernd Klaus, Verena Zuber

Institut für Medizinische Informatik, Statistik und Epidemiologie (IMISE),  
Universität Leipzig

28. Oktober 2010

## I. Diskrete Daten

Häufigkeitstabellen  
Darstellung

## II. Stetige Daten

Maße für die Lage  
Maße für die Streuung  
Boxplot  
Stripcharts  
Histogramm

## III. Graphiken in R

## I. Diskrete Daten: Deskriptive Statistiken und Graphiken

## Was sind diskrete Variablen?

Diskrete Variablen nehmen nur eine endliche Anzahl an Werten an:

- ▶ Kategorial: Es besteht keine Rangordnung der Kategorien
- ▶ Ordinal: Kategorien können geordnet werden

Kategoriale oder ordinale Variablen sollten in R als Faktoren definiert sein.

Mit einer Häufigkeitstabelle kann man ein kategoriales Objekt zusammenfassen:

- ▶ `table(object)`: Absolute Häufigkeiten
- ▶ `prop.table(table(object))`: Relative Häufigkeiten

Betrachten wir einen Faktor mit 4 Ausprägungen:

```
DNA <- rep(c("A", "C", "G", "T"), 10)
```

1		"A"
2		"C"
3		"G"
3		"T"
⋮		⋮

► `table(DNA)` ergibt:

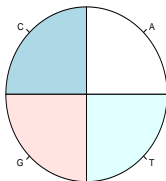
```
  A C G T  
10 10 10 10
```

► `prop.table(table(DNA))` ergibt:

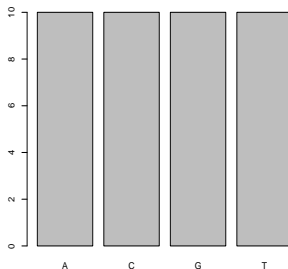
```
  A C G T  
0.25 0.25 0.25 0.25
```

## Kuchendiagramm und Balkendiagramm

Kuchendiagramm



Balkendiagramm



Zu erzeugen mit:

```
pie(table(DNA))
```

```
barplot(table(DNA))
```

## II. Stetige Daten: Deskriptive Statistiken und Graphiken

## Was sind stetige Variablen?

Stetige Variablen können (in der Theorie) eine unendliche Anzahl an Werten annehmen. Beispiele:

- ▶ Gewicht
- ▶ Größe
- ▶ Gehalt

R speichert stetige Variablen als metrische Objekte (`numeric`) ab.

Häufigkeitstabelle sind für stetige Variablen meist nicht geeignet. Wichtiger sind:

- ▶ Maße für die Lage
- ▶ Maße für die Streuung



## Maße für die Lage

Die Lage (*location*) gibt an, in welcher Größenordnung sich Daten bewegen.

- ▶ (Empirische) Mittelwert

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} (x_1 + \dots + x_n).$$

- ▶ In R: `mean()`
- ▶ Modus: Der Wert, der am häufigsten in den Daten vorkommt
- ▶ In R: `which.max()`

## Maße für die Lage II

- ▶  $x\%$ -Quantile, trennen die Daten in zwei Teile.  
So liegen  $x\%$  der Daten unter dem  $x\%$ -Quantile und  $100 - x\%$  darüber.
  - ▶ Median  $x_{0.5}$  entspricht dem 50%-Quantil
  - ▶ In R: `median()`
  - ▶ 25%-Quantil  $x_{0.25}$  (das erste Quartil)
  - ▶ 75%-Quantil  $x_{0.75}$  (das dritte Quartil)
- ▶ Der Median ist robuster gegen Ausreißer als der Erwartungswert
- ▶ Oder gleich in R: `summary()`

## Maße für die Streuung

Die Streuung (*scale*) gibt an, wie stark die verschiedenen Werte voneinander abweichen.

- ▶ Die (empirische) Varianz

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 = \frac{1}{n} ((x_1 - \bar{x})^2 + \dots + (x_n - \bar{x})^2).$$

- ▶ Spannweite:  
Differenz vom größten zum kleinsten Wert
- ▶ Interquartilsabstand:

$$\text{IQR} = x_{0.75} - x_{0.25}$$

## Beispiel: *oecd*-Daten

Betrachten wir das durchschnittliche, frei verfügbare Einkommen einer Familie [ pro Kind, in tausend US-Dollar ].

- ▶ Einen Überblick erhält man durch:

```
summary(Einkommen)
Min. 1st Qu. Median Mean 3rd Qu. Max.
 5.10 16.60 21.10 19.18 22.65 34.20
```

- ▶ Die Varianz bzw. Standardabweichung

```
var(Einkommen)
[1] 50.75937
sd(Einkommen) (alternativ sqrt(var(Einkommen)) )
[1] 7.124561
```

## Beispiel: *oecd*-Daten II

- ▶ Den Interquartilsabstand erhält man durch:

```
IQR(Einkommen)  
[1] 6.05
```

- ▶ Die Spannweite mit

```
max(Einkommen) - min(Einkommen)  
[1] 29.1
```

Bei der Variable *Alkohol* (Prozentsatz der 13-15 jährigen Kinder, die mindestens zweimal betrunken waren) bestehen fehlende Werte.

- ▶ Mittelwertsberechnung über

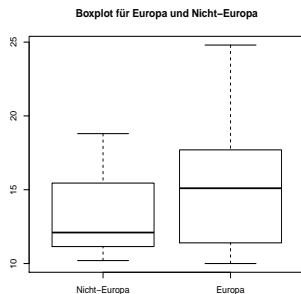
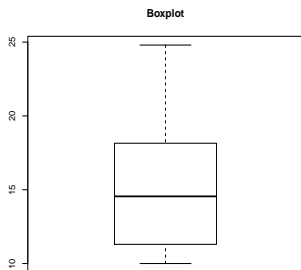
```
mean(Alkohol, na.rm=TRUE)  
[1] 15.225
```

## Was ist ein Boxplot?

Der Boxplot ist eine Graphik zur Darstellung stetiger Variablen.  
Er enthält

- ▶ Minimum und Maximum
- ▶ 25%-Quantil und 75%-Quantil
- ▶ Median
- ▶ In R: `boxplot(variable)`
- ▶ Um Variablen getrennt nach Faktorstufen zu untersuchen, bietet sich an: `boxplot(variable ~ factor)`
- ▶ Einschub: Ein Label für den Faktor `Geo`  
`factor(Geo, levels=c("R", "E"), labels=c("Nicht-Europa", "Europa"))`

## Boxplot: *Alkohol*



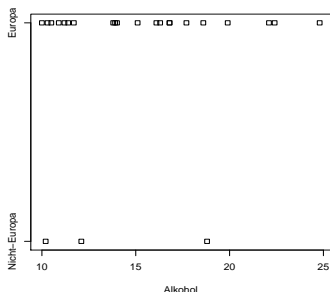
Zu erzeugen mit:

```
boxplot(Alkohol)
```

```
boxplot(Alkohol ~ Geo)
```

## Stripchart: *Alkohol*

Eine Alternative zum Boxplot bei wenigen Beobachtungen ist der Stripchart:



Zu erzeugen mit:

```
stripchart(Alkohol~Geo)
```



## Was ist ein Histogramm?

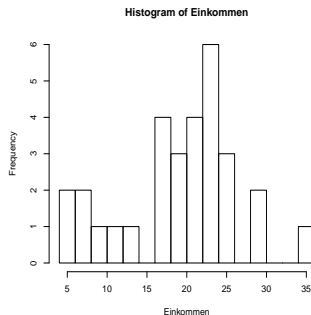
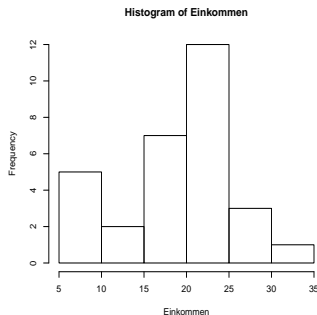
- ▶ Zur Erstellung eines Histogramms teilt man die Daten in homogene Teilintervalle ein und plottet dann die absolute Häufigkeit pro Teilintervall
- ▶ Dieses Verfahren gibt einen ersten Überblick über die Verteilung der Daten  
( => Ermitteln der “empirischen Dichte” möglich )

```
hist(x, breaks = “AnzahlBins”, freq = NULL )
```

- ▶ `x`: Daten
- ▶ `breaks = “AnzahlBins”`: Steuerung der Teilintervalle
- ▶ `freq=TRUE`: absolute Häufigkeiten
- ▶ `freq=FALSE`: relative Häufigkeiten (“empirische Dichte”)

## Histogramm: *Einkommen*

Histogramme des Einkommens mit verschiedenen Binstärken



Zu erzeugen mit:

```
hist(Einkommen)
```

```
hist(Einkommen, breaks=15)
```

## III. Graphiken in R: Grundaufbau und Parameter

## Graphiken in R

R kennt einen Standardbefehl für einfache Graphiken (`plot()`), aber auch viele spezielle Befehle, wie `hist()` oder `pie()`.

```
plot(x, y, type, main, par (...))
```

- ▶ `x`: Daten der  $x$ -Achse
- ▶ `y`: Daten der  $y$ -Achse
- ▶ `type="l"`: Darstellung durch eine Linie
- ▶ `type="p"`: Darstellung durch Punkte
- ▶ `main`: Überschrift der Graphik
- ▶ `par (...)`: Zusätzlich können sehr viele Parametereinstellungen geändert werden

## Parameter für Graphiken in R

```
par(cex, col, lty, mfrow, pch, x/yaxs)
```

- ▶ `cex`: Skalierung von Graphikelementen
- ▶ `col`: Farbe (`colors()` zeigt die vordefinierten Farben an)
- ▶ `lty`: Linienart
- ▶ `mfrow`: Anordnen von mehreren Graphiken nebeneinander
- ▶ `pch`: Andere Punkte oder Symbole
- ▶ `x/yaxs`: Stil der  $x$ - bzw.  $y$ -Achse

Einen Überblick über die Parameter erhält man mit `?par`.  
`par()` kann entweder im `plot()` -Befehl gesetzt werden oder als eigene Funktion vor einem oder mehreren `plot()`-Befehlen.

## Aufbau von Graphiken in R

1. `plot()`: Bildet den Grundstein einer Graphik
2. Zusätzlich können weitere Elemente eingefügt werden wie:
  - ▶ `lines()`: Linien
  - ▶ `points()`: Punkte
  - ▶ `legend()`: Legende
  - ▶ `text()`: Text
3. `dev.off()`: schließt die Graphik

Einen Überblick erhält man mit der betreffenden Hilfsfunktion, z.B. `?legend`.

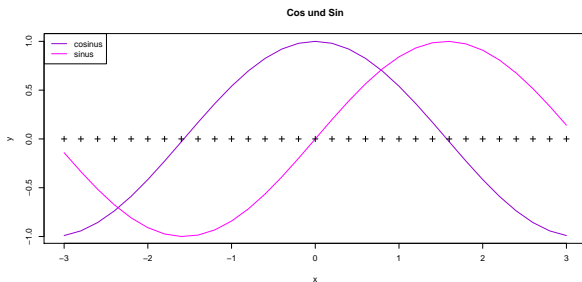
## Abspeichern von Graphiken

Folgende Graphikformate können in R erzeugt werden:

- ▶ `pdf()`
- ▶ `ps()`
- ▶ `jpg()`

Beispiel:

```
pdf(file="boxplot.pdf", width=13, height=6)
par(mfrow=c(1,2))
boxplot(Alkohol, main="Boxplot")
boxplot(Alkohol~Geo, main="Boxplot für ...")
par(mfrow=c(1,1))
dev.off()
```



```
pdf(file="RGraphiken/beispiel.pdf", width=12, height=6)
plot(x,y, type="l", col="darkviolet", main="Cos und Sin")
lines(x,z, col="magenta")
points(x,null, pch=3)
legend("topleft", c("cosinus","sinus"),
col=c("darkviolet", "magenta"), lty=1)
dev.off()
```