



Einführung in die Statistik

Praktische Übung – Jürgen Hermes – IDH – SoSe 2023

Programm heute

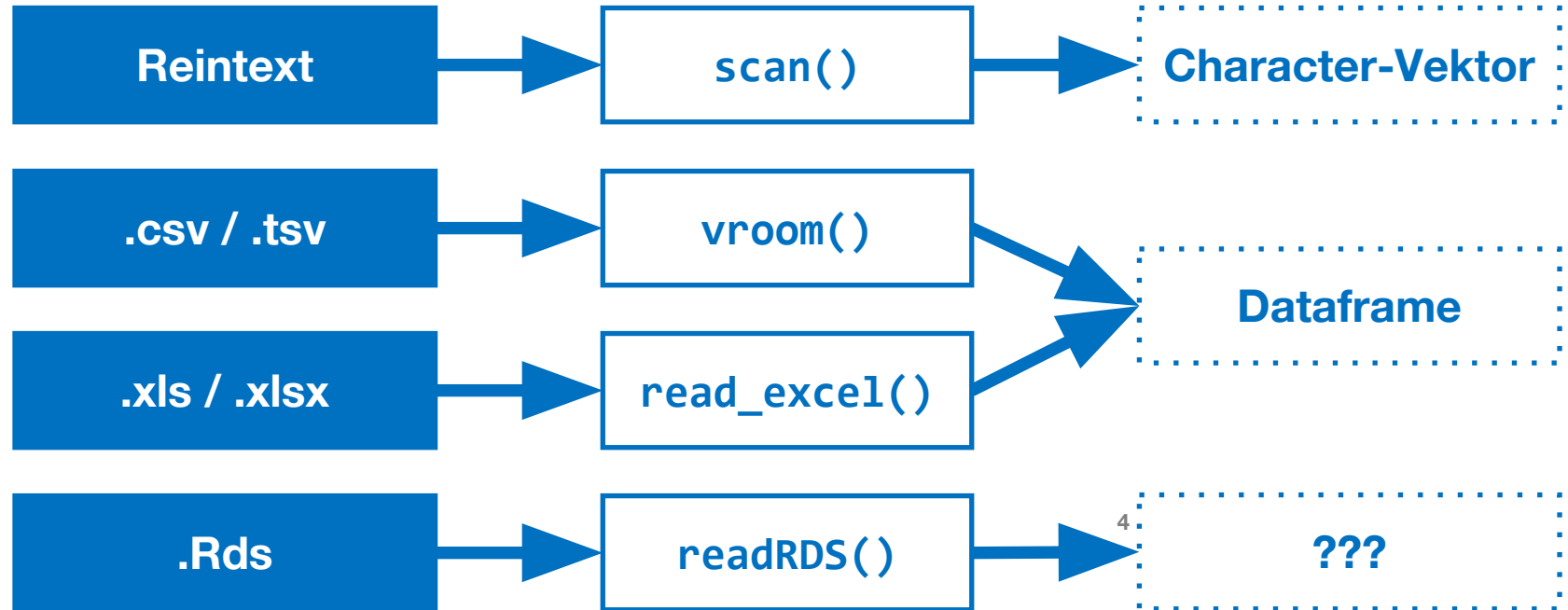
- Daten mit R ausgeben
- Zusammenfassung der R-Einführung
- Besprechung der Hausaufgabe von letzter Woche
- Klärung von Fragen
- Bonus: Visualisierungen

R: Die Grundlagen

Diese und die folgenden Folien sind erstellt worden von Sascha Wolfer für seinen Kurs "Statistik mit R" an der Uni Basel. Ich nutze sie mit seiner freundlichen Genehmigung. DOI für die Materialien ist

[10.5281/zenodo.7431504](https://doi.org/10.5281/zenodo.7431504)

Einlesen: Zusammenfassung



Daten ausgeben



Daten ausgeben: Funktionen

Reintext

`cat()`

.csv / .tsv

`vroom::vroom_write()`

.xls / .xlsx

`WriteXLS::WriteXLS()`

.Rds

`saveRDS()`

6

Reinen Text ausgeben mit `cat()`

- Es kommt nicht sonderlich oft vor, dass Sie aus R laufenden Text ausgeben müssen.
- Wenn, dann am einfachsten mit
`cat(<Vektor>, <Datei>, sep = <Separator>)`
- Ausgeben der Buchstaben a bis z in einzelnen Zeilen der Datei `buchstaben.txt`:

```
cat(letters, file = "buchstaben.txt", sep = "\n")
```

CSV-Dateien schreiben mit `vroom_write()`

- `vroom_write()` schreibt Dataframes in CSV-Dateien.

```
vroom_write(<Dataframe>, <Dateiname>, delim =  
<Separator>)
```

- Weitere Argumente: `?vroom_write`

R-Objekte schreiben mit `saveRDS()`

- Wenn Sie ein beliebiges R-Objekt in einer .Rds-Datei speichern möchten, benutzen Sie
`saveRDS(<Variable>, <Datei>)`
- Diese Dateien können nur von R wieder eingelesen werden.
 - Erinnerung: Einlesen mit `readRDS()`



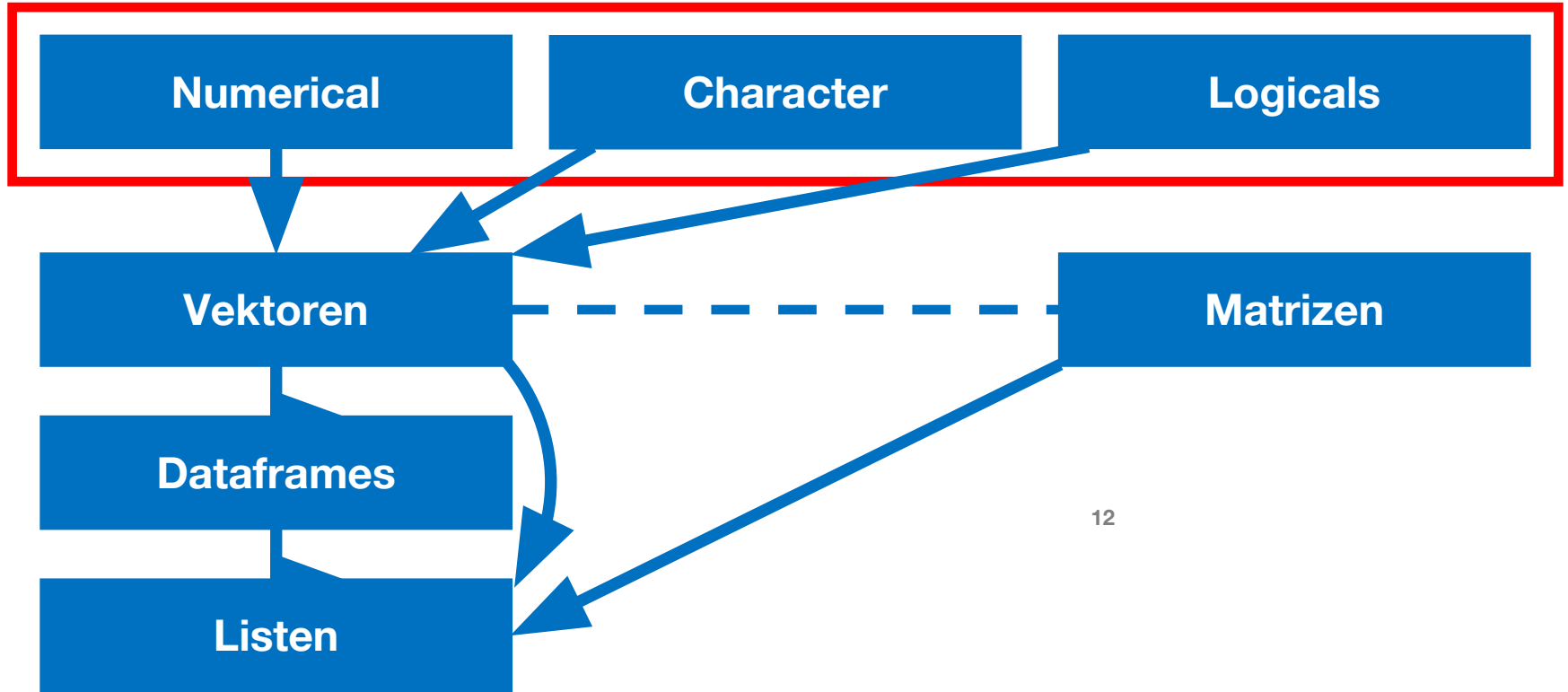
R-Einführung: Zusammenfassung

Oberfläche

- Befehle sammeln im Script-Editor, **jetzt** mit # kommentieren und sich später freuen!
- Pakete installieren mit `install.packages()`
- Hilfe abrufen mit `?<Funktion>` oder Hilfe durchsuchen mit `??<Suchterm>`

Datentypen

Elementare Datentypen



12

Vektoren ...

- ... bestehen immer aus *einem* elementaren Datentyp.
- ... werden automatisch **recyclet**.
- ... können **Faktor**vektoren sein, um nominal-/ordinalskalierte Daten zu repräsentieren.

Indizierung

- Zugreifen auf/Selektieren von Elemente(n) in komplexen Datentypen.

[]

[,]

\$

[[]]

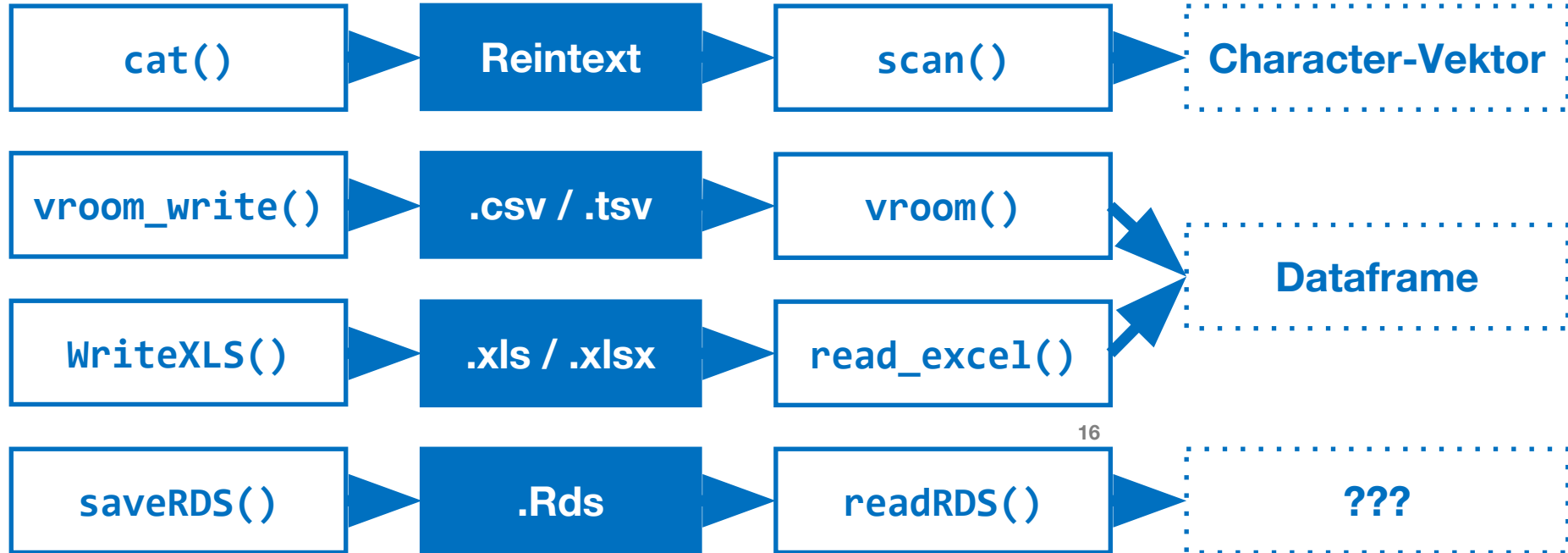
14

Wir können über Zahlen, Namen und Wahrheitswerte indizieren.

Funktionen

- Funktionen haben **Argumente**.
- Funktionen können **verschachtelt** werden.
- **Prädikate** sind spezielle Funktionen, die immer einen Wahrheitswert zurückgeben.

Einlesen & Ausgeben





”

If you're not failing 90% of the time, then you're probably not working on sufficiently challenging problems.

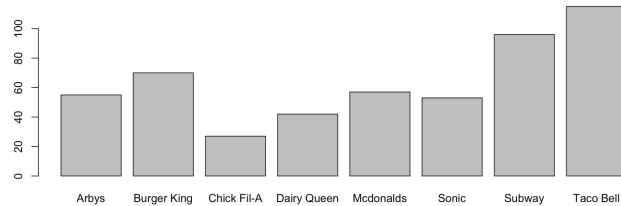
- Alan Kay

Besprechung Hausaufgabe von letzter Woche

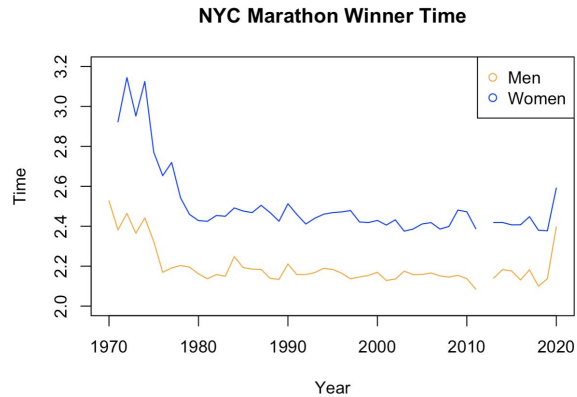
- Abgabemodalitäten finden Sie auf ILIAS
- Ist relativ ähnlich der Übung P06F04 gestaltet
- Sie benötigen das gleiche Paket {openintro}.
- Speichern Sie den data.frame “fastfood” auf einer eigenen Variable.
- Geben Sie alle Produkte (Items) des Restaurants “Burger King” aus, die mehr als 1000 Kalorien enthalten.
- Geben Sie die Items mit den höchsten Werten für Vitamin C aus. Nutzen Sie dafür die Funktionen “order()” und “head()”
- Berechnen Sie für alle Items den Quotienten aus protein und calorien. Speichern Sie diese in einer Spalte \$protPerCal
- Ermitteln Sie für jedes Restaurant, wie viele Items es anbietet. Geben Sie für diese Werte einen Barplot aus. Nutzen Sie dafür die Funktionen table() und barplot()
- Speichern Sie ihr data.frame als R-Objekt und als tsv-Datei.

Visualisierungen

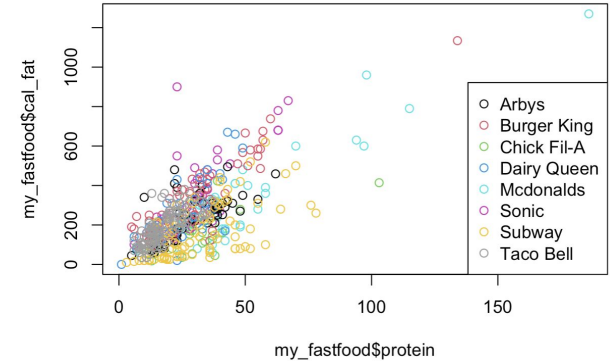
Barplots



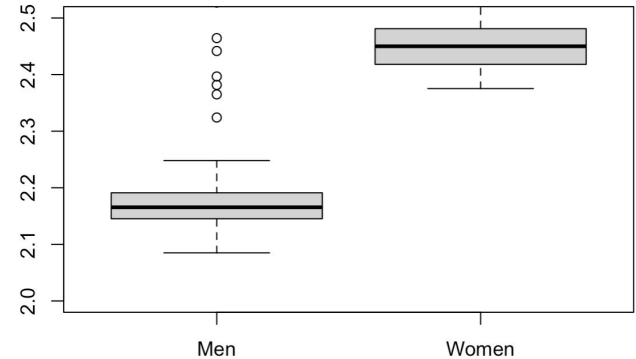
Linienplots



Punktplots

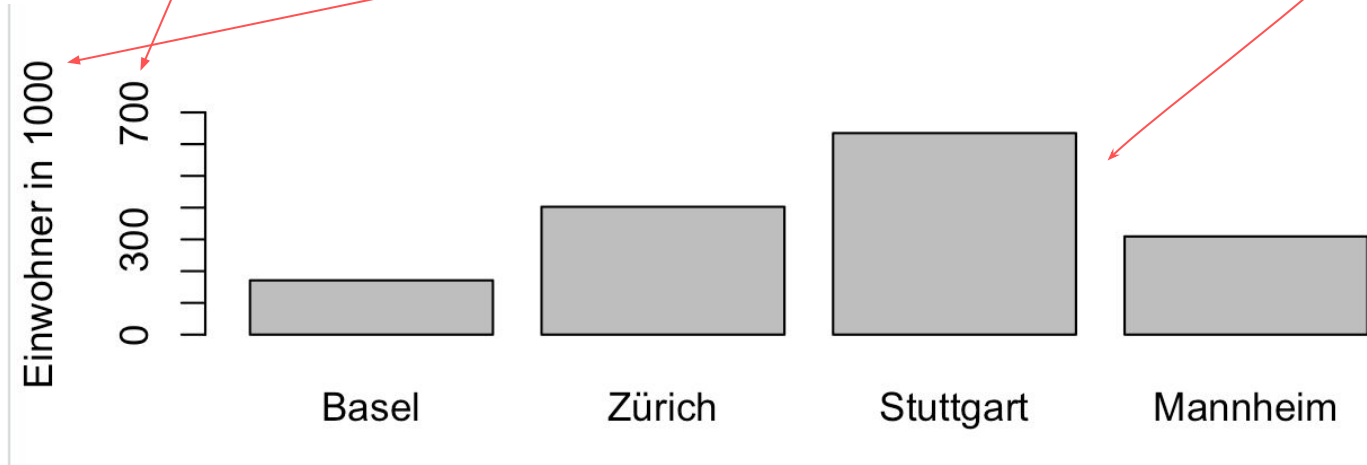


Boxplots



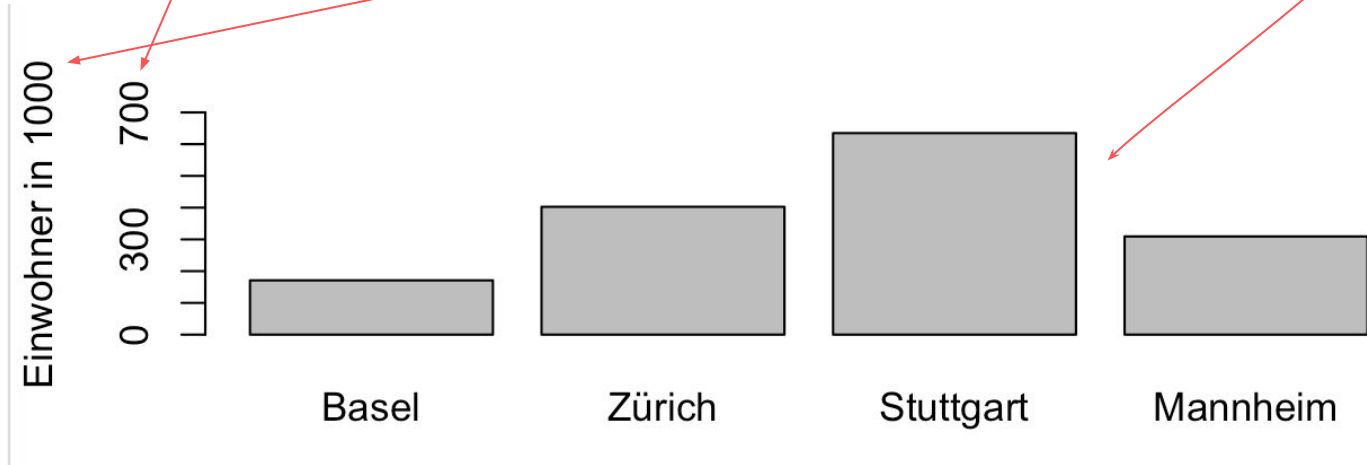
Visualisierung: Barplots

- `barplot(height = cit$einw/1000, names.arg = cit.name, ylim = c(0, 700), ylab= "Einwohner in 1000")`



Visualisierung: Barplots

- `barplot(height = cit$einw/1000, names.arg = cit.name, ylim = c(0, 700), ylab= "Einwohner in 1000")`

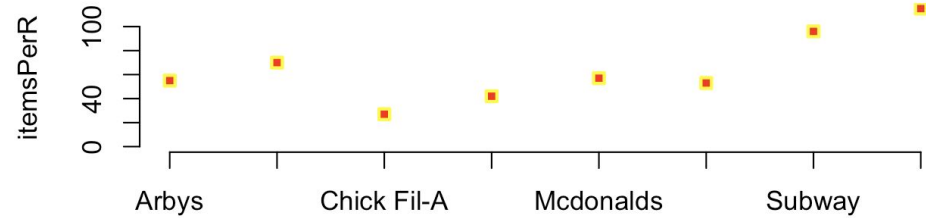


Visualisierung: Punktdiagramme

```
plot(itemsPerR, pch = 22,  
      bg = "red", color = "yellow")
```

plot (x, y, pch = _)

0 □	1 ○	2 △	3 +	4 ×	5 ◇	6 ▽	
7 ⊠	8 ✱	9 ⊞	10 ⊕	11 ⊗	12 ⊞	13 ⊗	14 ⊞
15 ■	16 ●	17 ▲	18 ◆	19 ●	20 ●		
21 ●	22 ■	23 ◆	24 ▲	25 ▼	21:25 mit bg füllbar		



Farben Übersicht (mehr in bF Anhang)



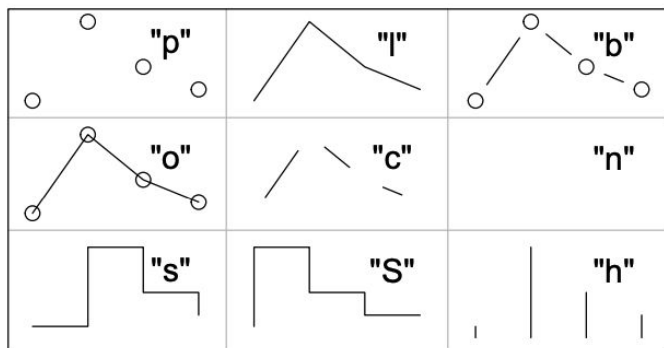
Visualisierung: Liniendiagramme

```
plot(y=nycm_women$time_hrs, x=nycm_women$year, col="blue", type = "l", ylim=c(2,3.2),  
     xlab = "Year", ylab="Time", main = "NYC Marathon Winner Time")
```

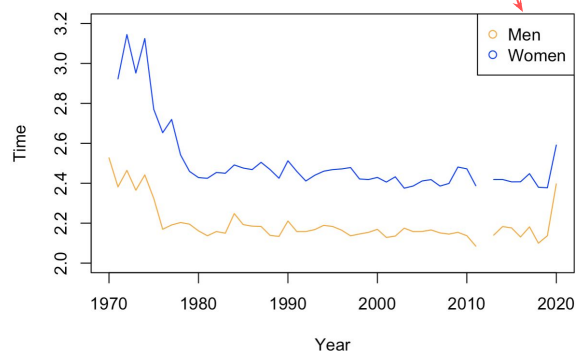
```
lines(y=nycm_men$time_hrs, x=nycm_men$year, col="orange")
```

```
legend("topright", levels(as.factor(nyc_marathon$division)),  
      col =c("orange","blue"), pch = 1)
```

plot (x, y, type = _)

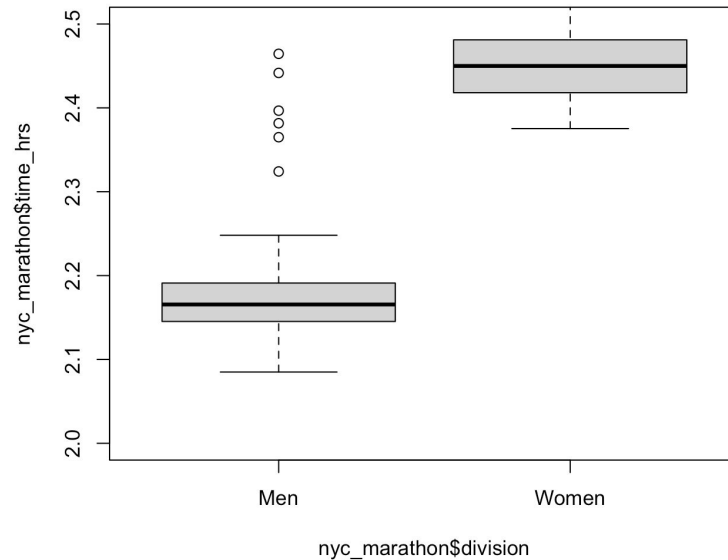


NYC Marathon Winner Time



Visualisierung: Boxplots

```
boxplot(nyc_marathon$time_hrs ~ nyc_marathon$division, ylim=c(2,3.2))
```



Hausaufgabe

- Wiederholen Sie noch einmal alle bisherigen Folien, Skripte und Hausaufgaben!
- Das Zwischentestat wird in ILIAS gestellt werden, Sie müssen es an den in unserem Kursraum zur Verfügung stehenden Rechnern bearbeiten.
- Hilfsmittel sind erlaubt, soweit sie keine Chat-Komponente enthalten (egal, ob am anderen eine Bekannte, ein Fremder oder eine LLM-KI sitzt).
- Bearbeitungszeit wird so gegen 45 Minuten tendieren, es bleibt am Anfang der nächsten Stunde ausreichend Zeit zur Klärung eventueller Fragen.