Datenvisualisierung 3

Darstellung der zusammenfassenden Statistik

Daniela Palleschi

Inhaltsverzeichnis

Lernziele

Heute werden wir lernen...

- Boxplots zu erstellen und zu interpretieren
- Mittelwerte und Standardabweichungen zu visualisieren

Ressourcen

- Kurswebsite (Datenvisualisierung 3)
- Abschnitt 2.5 (Visualisierung von Beziehungen) in Wickham et al. (2023)
- Kapitel 4 (Darstellung von zusammenfassenden Statistiken) in Nordmann et al. (2022)
- Abschnitte 3.5-3.9 in Winter (2019)

Einrichten

Pakete

```
pacman::p_load(tidyverse,
               here,
               janitor,
               ggthemes,
               patchwork)
```

Daten

```
df_eng <- read_csv(
  here(
    "daten",
    "languageR_english.csv"
  )
) |>
  clean_names() |>
  rename(
    rt_lexdec = r_tlexdec,
    rt_naming = r_tnaming
  )
```

1 Wiederholung

- Betrachten Sie jede Abbildung in Abbildung 1
  - Wie viele Variablen werden in jeder Abbildung dargestellt?
  - welche *Typen* von Variablen sind es?
  - Welche zusammenfassende(n) Statistik(en) wird/werden in jedem Diagramm dargestellt?

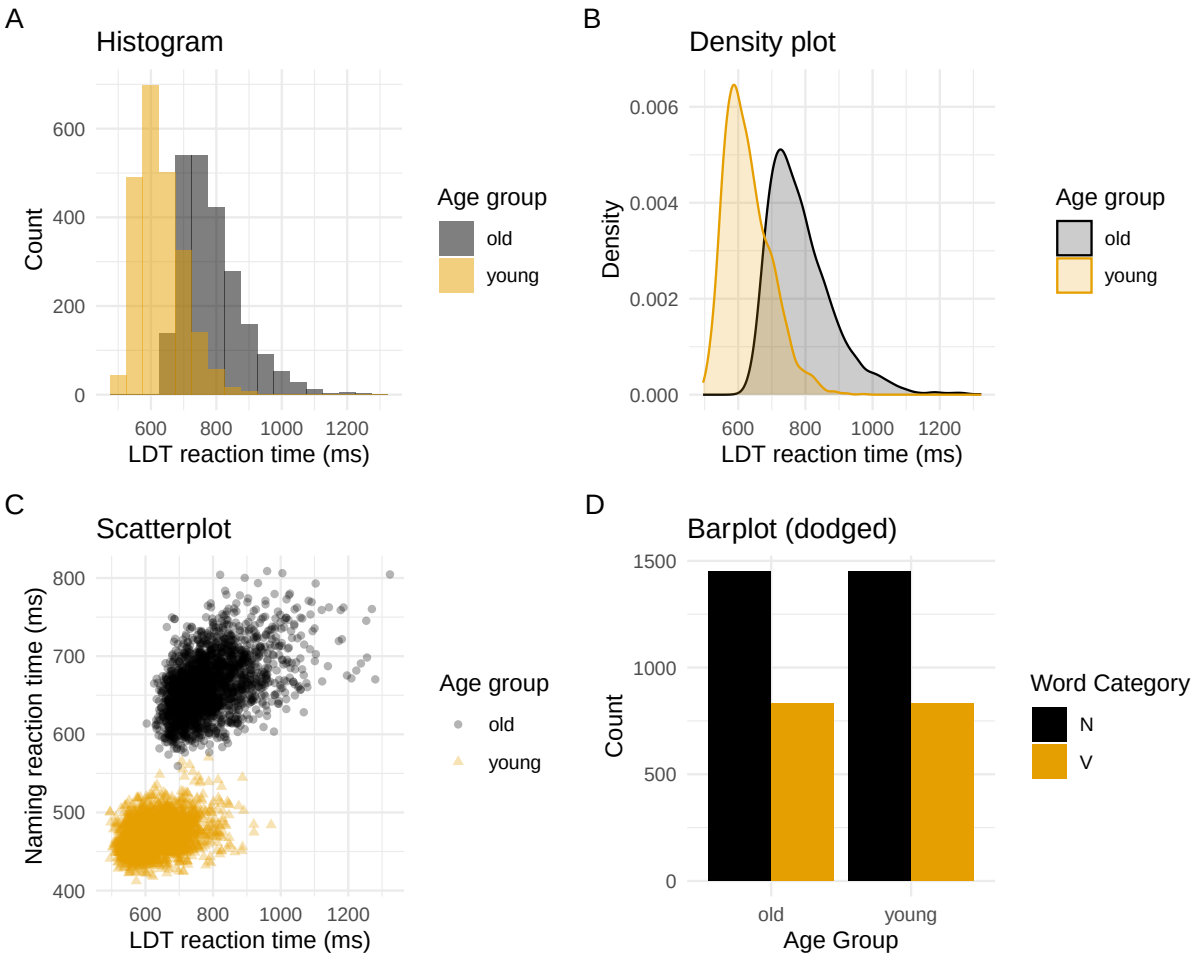


Abbildung 1: Different plots types

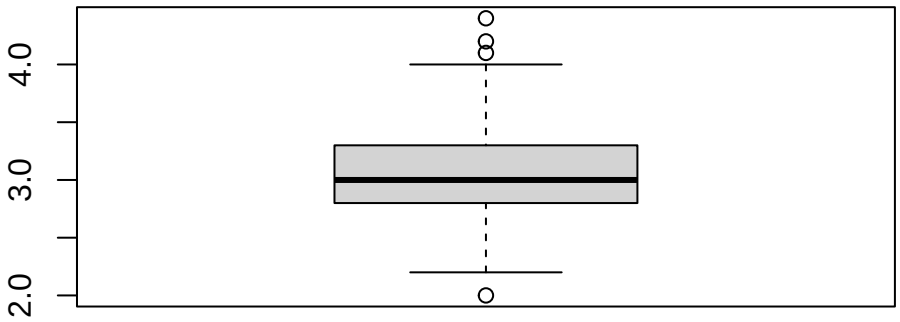
2 Darstellung von zusammenfassenden Statistiken

- Modus und Bereich werden in Histogrammen und Dichteplots visualisiert

- die Anzahl der Beobachtungen wird in Balkendiagrammen visualisiert

2.1 Boxplot

- auch bekannt als Box-and-Whisker-Plots, enthalten
  - eine Box
  - eine Linie in der Mitte der Box
  - Linien, die an beiden Enden der Box herausragen (die 'Whisker')
  - manchmal Punkte



- Betrachten Sie Abbildung 2
  - identifiziere jeden dieser 4 Aspekte des Plots
  - können Sie erraten, was jeder dieser Aspekte darstellen könnte und wie Sie die Darstellung interpretieren sollten?

- Boxplots vermitteln eine Menge Informationen in einer einzigen Visualisierung
  - Die Box selbst stellt den *Interquartilsbereich* (IQR; der Bereich der Werte, der zwischen den mittleren 50% der Daten liegt) dar.
    - Die Grenzen der Box repräsentieren Q1 (1. Quartil, unter dem 25% der Daten liegen) und Q3 (3. Quartil, über dem 25% der Daten liegen)
  - die Linie in der Mitte des Boxplots stellt den *Median* dar
    - auch Q2 genannt (2. Quartil; der mittlere Wert, über/unter dem 50% der Daten liegen)
  - Die Whisker repräsentieren `1,5*IQR` von Q1 (unterer Whisker) oder Q3 (oberer Whisker)

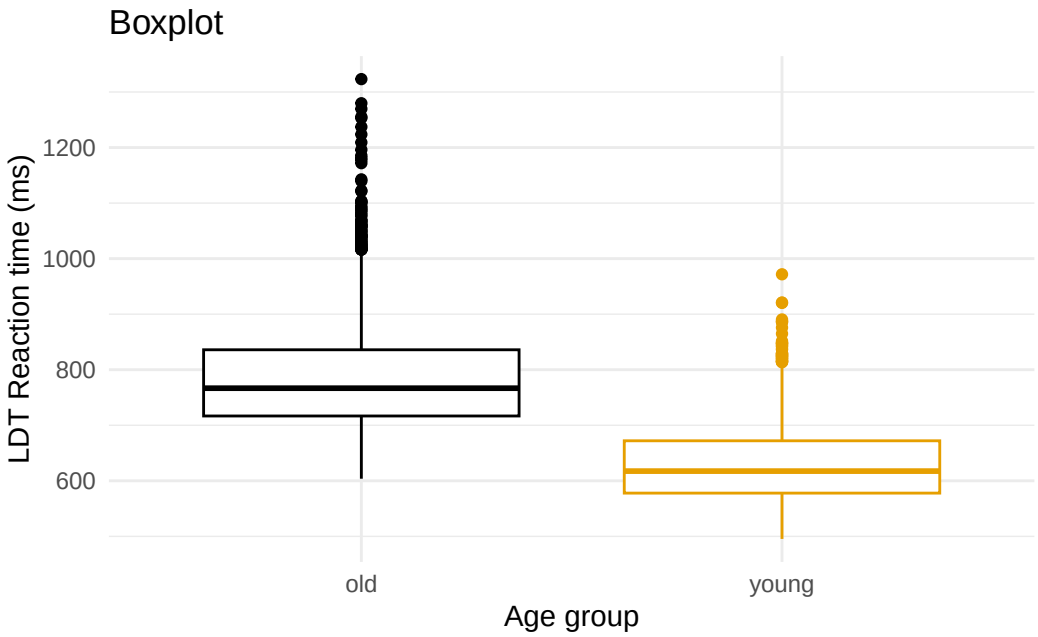


Abbildung 2: Boxplot of `df_eng` (body mass by age_subject)

  - Punkte, die außerhalb der Whisker liegen, stellen Ausreißer dar (d. h. Extremwerte, die außerhalb des IQR liegen).

- **?@fig-winter-boxplot-hist** zeigt die Beziehung zwischen einem Histogramm und einem Boxplot

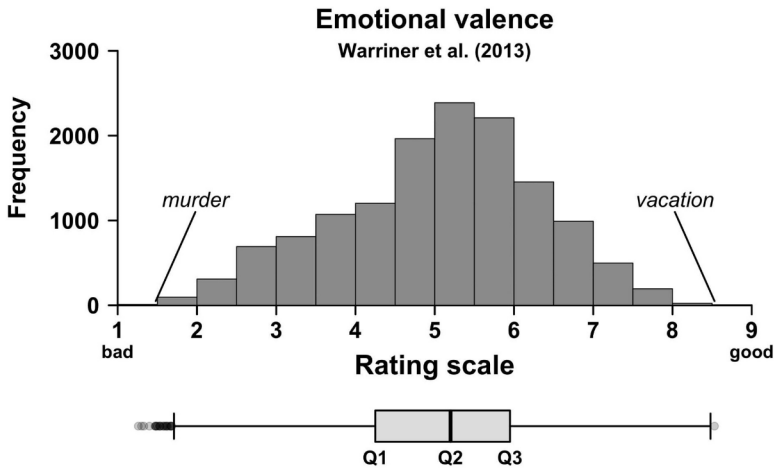


Figure 3.4. A histogram of the emotional valence rating data

Abbildung 3: Image source: Winter (2019) (all rights reserved)

- **?@fig-wickham-boxplot-hist** hat einen ähnlichen Vergleich, einschließlich eines Streudiagramms

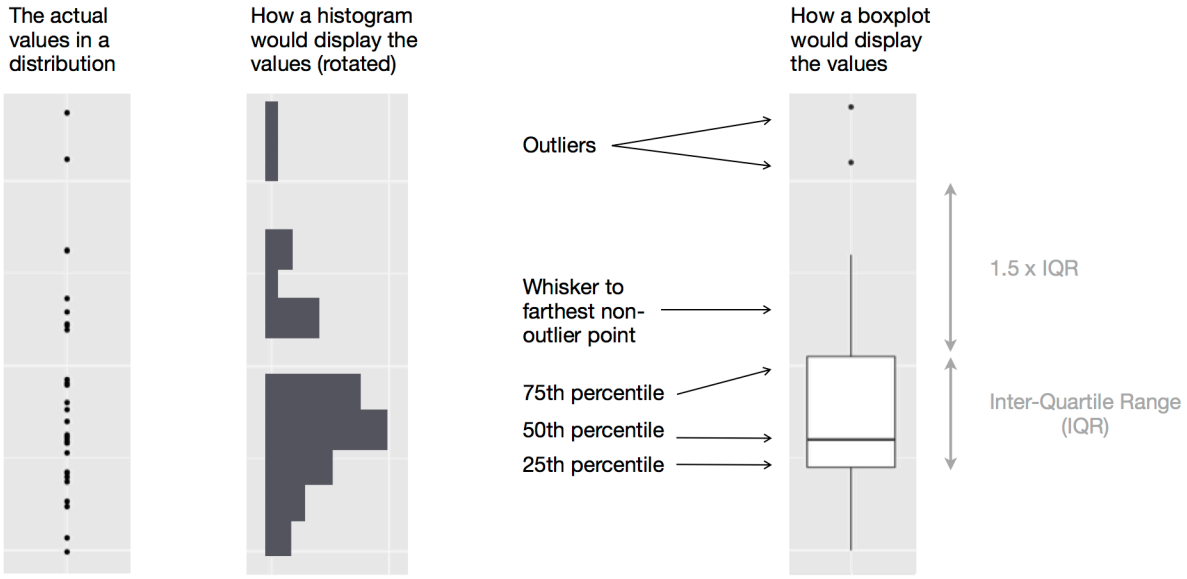


Abbildung 4: Image source: Wickham et al. (2023) (all rights reserved)

2.1.1 `geom_boxplot()`

- Die Funktion `geom_boxplot()` von `ggplot2` erzeugt Boxplots
  - sie benötigt eine numerische Variable als `x` oder `y` Achse (Abbildung 5)

```
df_eng |>
  ggplot(aes(y = rt_lexdec)) +
  geom_boxplot()
```

- für Boxplots verschiedener Gruppen: eine kategorische Variable entlang der anderen Achse (Abbildung 6)

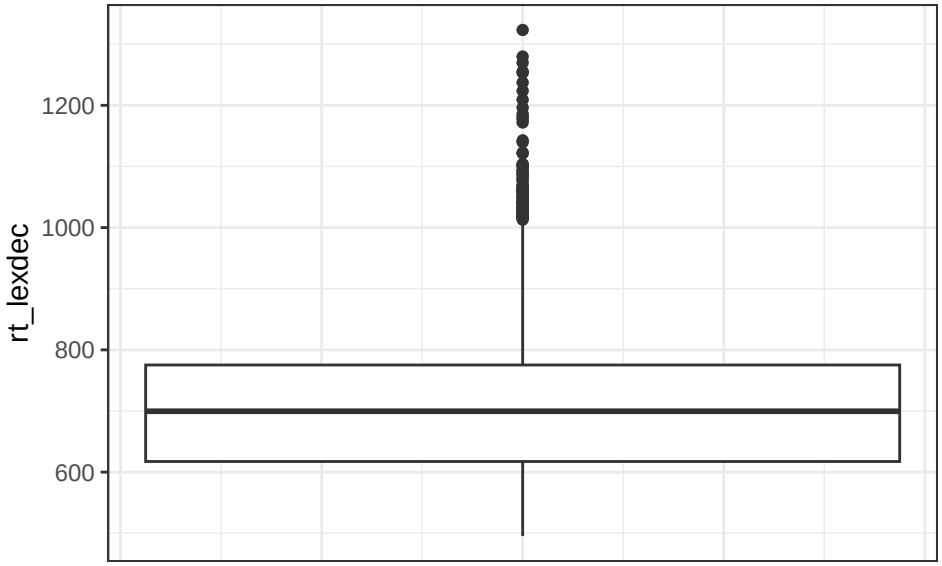


Abbildung 5: A boxplot for all observations of a continuous variable

```
df_eng |>
  ggplot(aes(x = age_subject, y = rt_lexdec)) +
  geom_boxplot() +
  theme_bw()
```

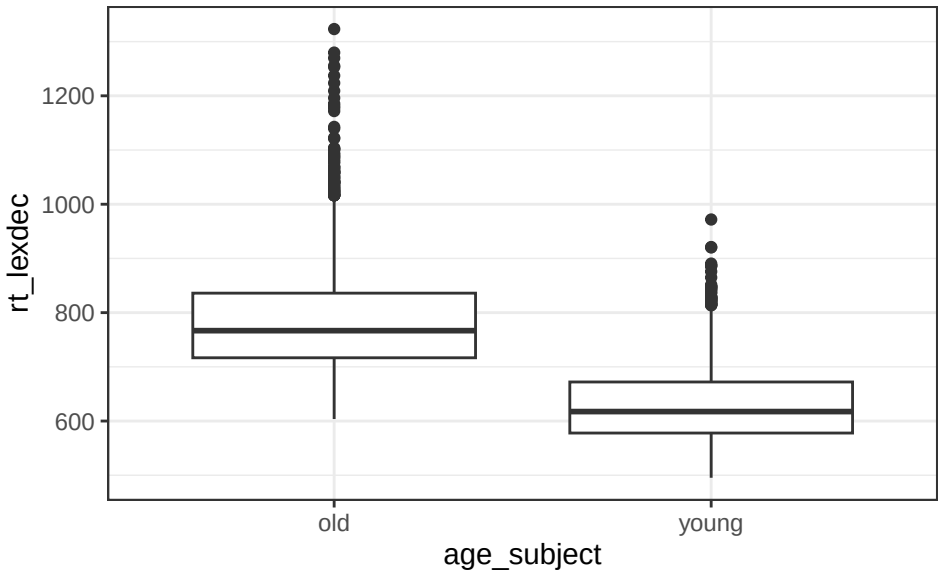


Abbildung 6: A boxplot for two groups

2.1.2 Gruppierter Boxplot

- Wir können gruppierte Boxplots erstellen, um mehr Variablen zu visualisieren
  - einfach eine neue Variable mit `colour` oder `fill` ästhetisch zuordnen

```
df_eng |>
  ggplot(aes(x = age_subject, y = rt_lexdec,
             colour = word_category)) +
  geom_boxplot() +
  labs(
    x = "Age group",
    y = "LDT reaction time (ms)",
    color = "Word type"
  ) +
  scale_colour_colorblind() +
  theme_bw()
```

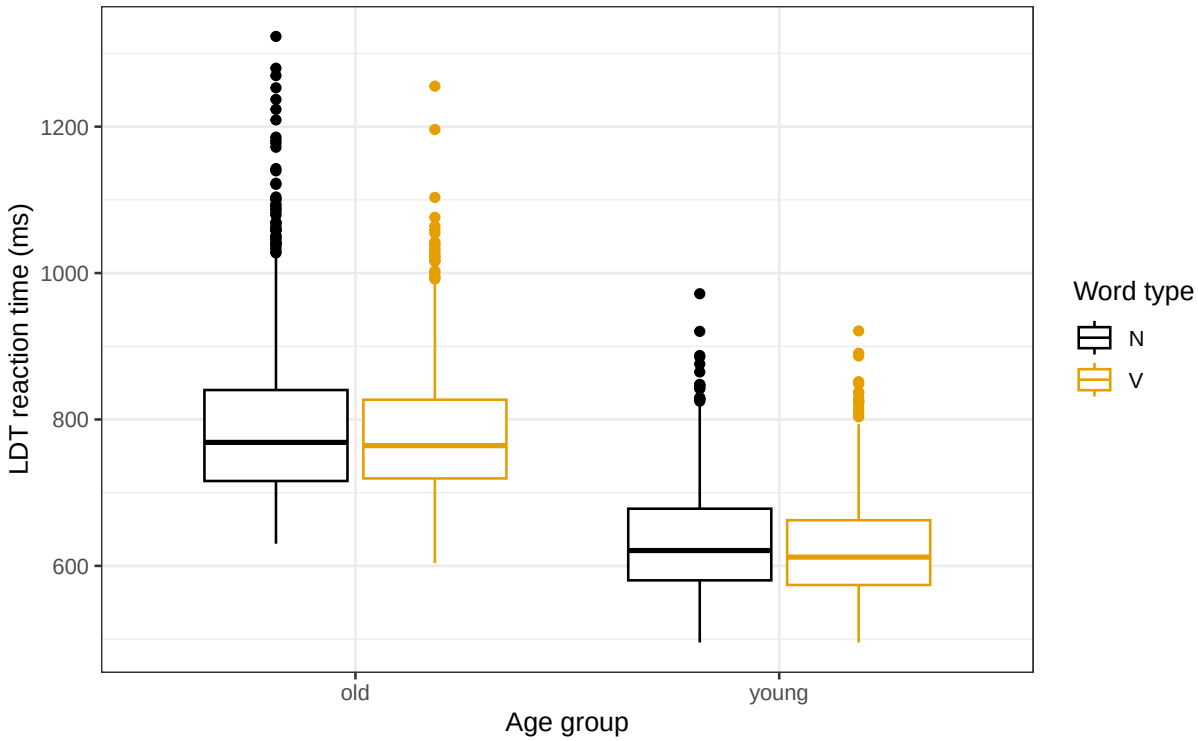


Abbildung 7: A grouped boxplot

3 Visualisierung des Mittelwerts

- In der Regel wollen wir auch den Mittelwert mit der Standardabweichung darstellen.
  - Wie können wir das tun?

3.1 Fehlerbalkenplots

- Diese Diagramme bestehen aus 2 Teilen:
  - der Mittelwert, visualisiert mit `geom_point()`
  - ein Maß für die Streuung, visualisiert mit "geom_errorbar()".
- für diesen Kurs werden wir die Standardabweichung verwenden
- Abbildung 8 ist das, was wir heute erzeugen werden

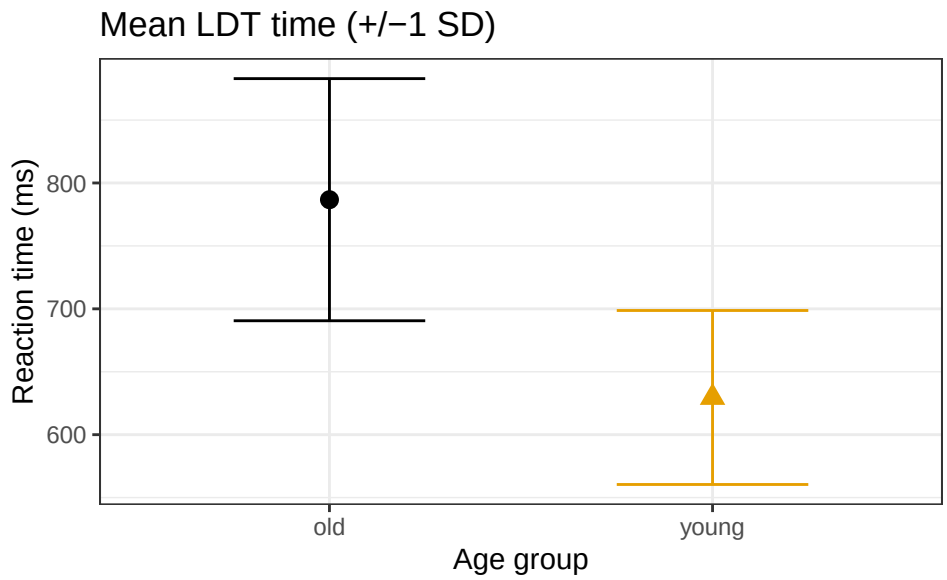


Abbildung 8: Errorbar plot of `df_eng` (body mass by age_subject)

3.1.1 Berechnung der zusammenfassenden Statistik

- müssen wir zunächst den Mittelwert und die Standardabweichung berechnen
  - gruppiert nach den Variablen, die wir visualisieren wollen
- Wie kann man den Mittelwert und die Standardabweichung von `rt_lexdec` nach `age_subject` berechnen?

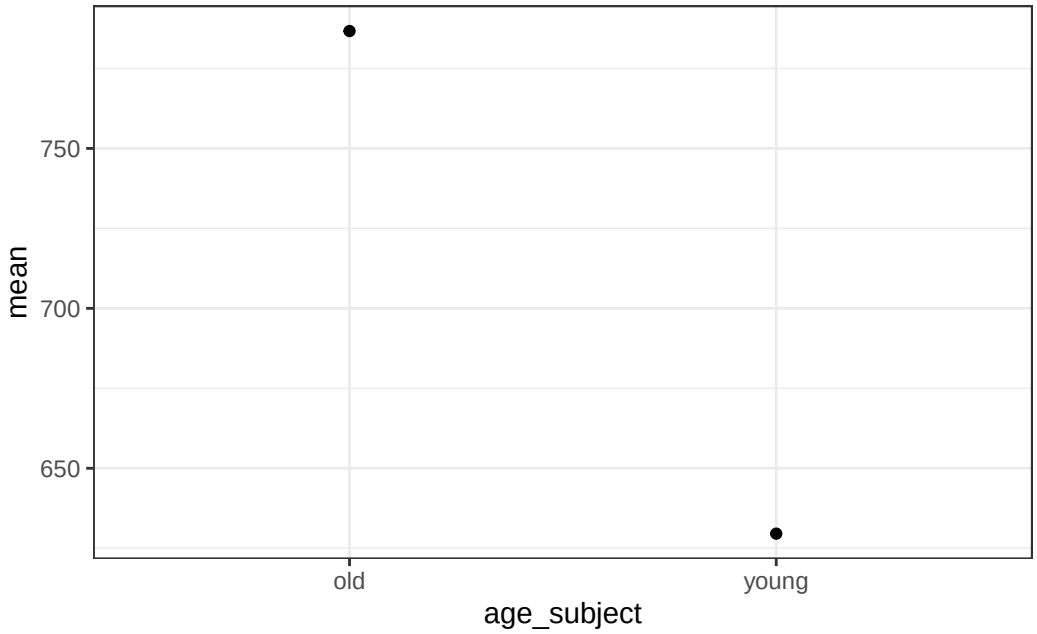
```r
sum_eng <- df_eng |>
  summarise(mean = mean(rt_lexdec),
            sd = sd(rt_lexdec),
            N = n(),
            .by = age_subject) |>
  arrange(age_subject, age_subject)
```

- Diese Zusammenfassung können wir dann in `ggplot()` mit den entsprechenden ästhetischen Zuordnungen und Geomen einfügen

3.1.2 Plotting mean

- Zunächst werden die Mittelwerte mit `geom_point()` dargestellt.

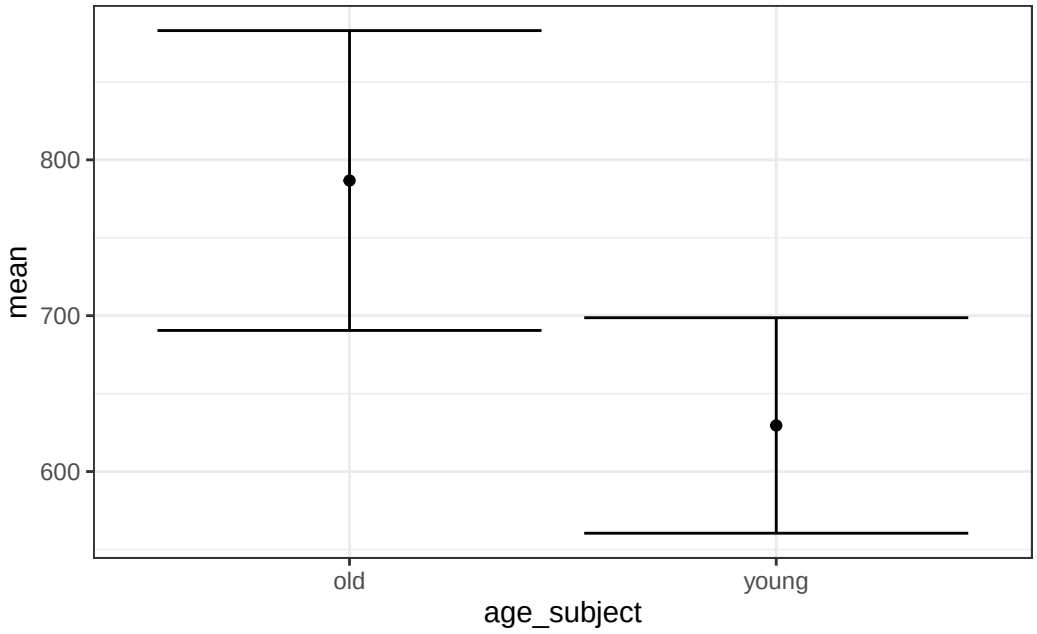
```r
sum_eng |>
  ggplot() +
  aes(x = age_subject, y = mean) +
  geom_point()
```



3.1.3 Hinzufügen von Fehlerbalken

- Fügen wir nun unsere Fehlerbalken hinzu, die 1 Standardabweichung über und unter dem Mittelwert darstellen
- wir tun dies mit `geom_errorbar()`
  - nimmt `ymin` und `ymax` als Argumente
  - In unserem Fall sind dies `mean-/+sd`.

```
sum_eng |>
  ggplot() +
  aes(x = age_subject, y = mean) +
  geom_point() +
  geom_errorbar(aes(ymin = mean-sd,
                    ymax = mean+sd))
```



- Wenn wir weitere Anpassungen hinzufügen, erhalten wir Abbildung 9

```
sum_eng |>
  ggplot() +
  aes(x = age_subject, y = mean,
      colour = age_subject, shape = age_subject,
```

```
    ymin=mean-sd, ymax=mean+sd) +
geom_point(size = 3) +
geom_errorbar(width = .5) +
labs(title = "Mean LDT times (+/-1SD)",
  x = "Age group",
  y = "Reaction time (ms)",
  color = "Age group"
) +
scale_color_colorblind() +
theme_bw() +
theme(
  legend.position = "none"
)
```

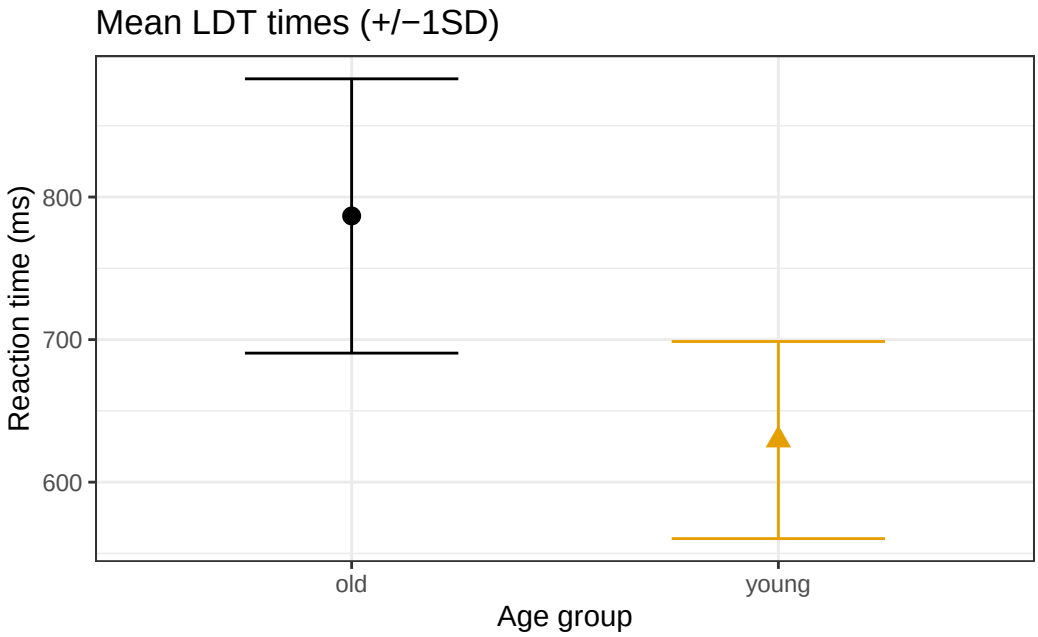


Abbildung 9: Customised errorbar

4 Barplot von Mittelwerten: Finger weg!

- Sie werden sehr oft Balkendiagramme von Mittelwerten sehen
  - aber es gibt viele Gründe, warum dies eine schlechte Idee ist!!
- Der Balkenplot hat ein schlechtes Daten-Tinten-Verhältnis, d.h. die Menge der Datentinte geteilt durch die Gesamttinte, die zur Erstellung der Grafik benötigt wird

- Was ist, wenn es nur sehr wenige oder gar keine Beobachtungen in der Nähe von Null gibt? Wir verbrauchen eine Menge Tinte, wo es keine Beobachtungen gibt!

- Außerdem deckt der Balken nur den Bereich ab, in dem die untere *Hälfte* der Beobachtungen liegt; ebenso viele Beobachtungen liegen über dem Mittelwert!

4.1

- Wie groß ist die Bandbreite der beobachteten Werte?

```
range(df_eng$rt_lexdec)
```

```
[1]  495.38 1323.20
```

- beachten Sie, dass der tatsächliche Bereich der Datenpunkte und das Balkendiagramm viel "Tinte" für datenfreie (d. h. unbeobachtete) Reaktionszeitwerte verwenden

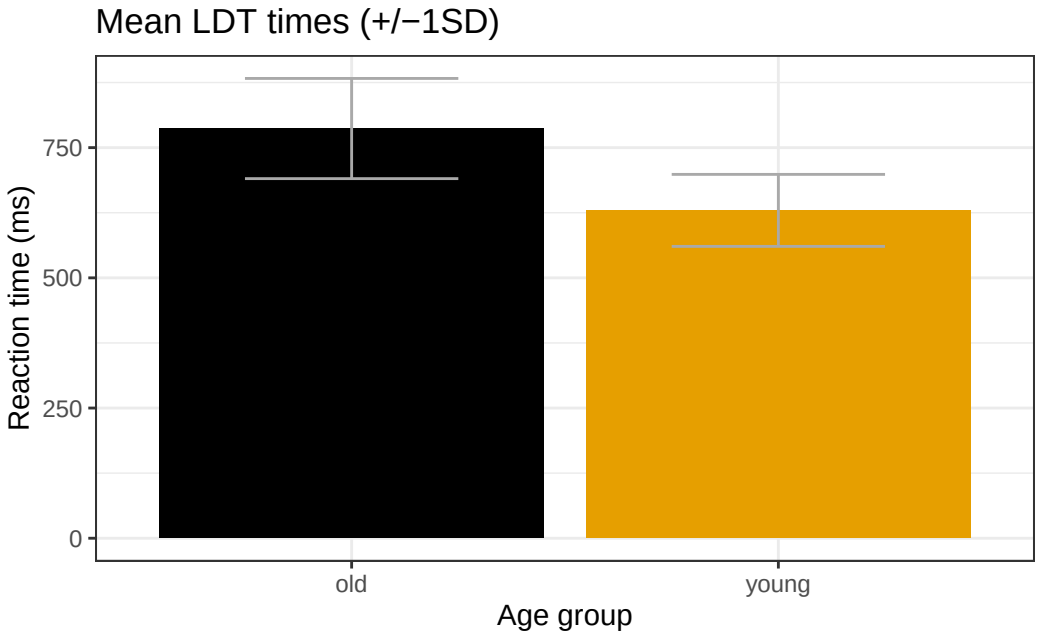


Abbildung 10: Balkendiagramm des Mittelwerts mit +/- 1 Standardabweichung (ich empfehle, von solchen Diagrammen abzusehen!)

4.2 Fehlerbalkendiagramm vs. Balkendiagramm für Mittelwerte

- Abbildung 11 A und B stellen dieselbe Information dar

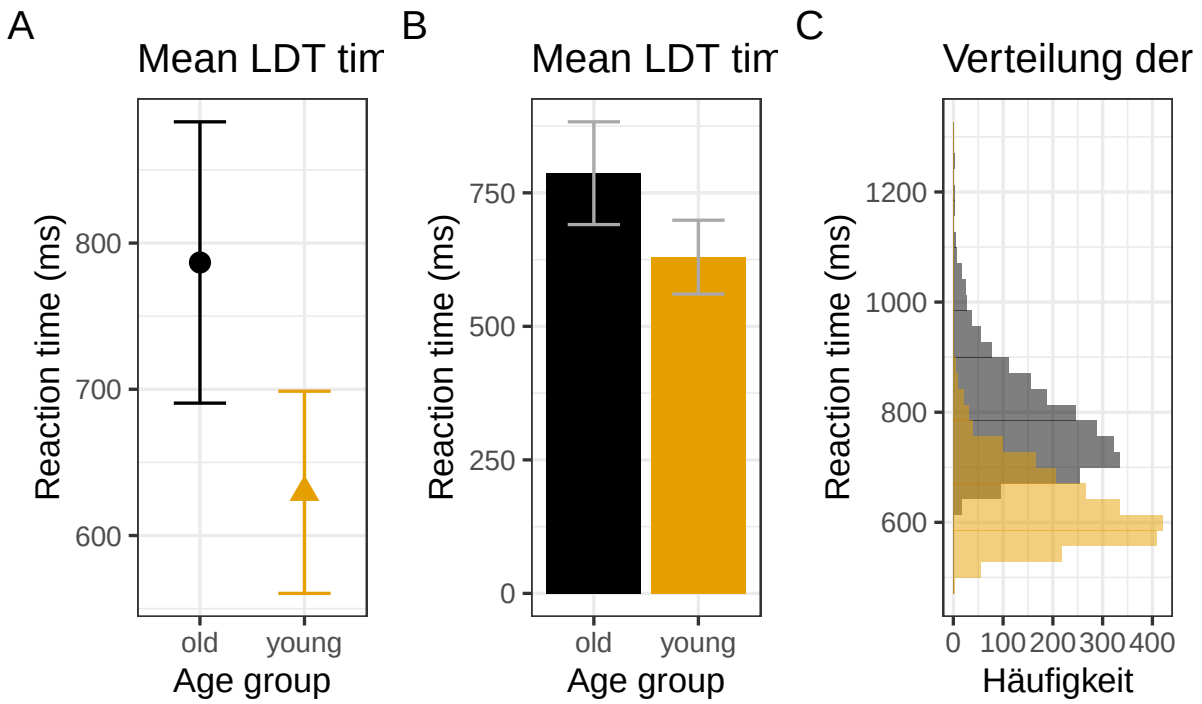


Abbildung 11: Fehlerbalken-Diagramm und Balkendiagramme, die Mittelwerte (+/-1SD) darstellen, sowie ein Histogramm der gleichen Daten

4.3 Gleiche Grenzen auf der y-Achse

- Abbildung 12 zeigt die gleichen Daten, aber mit dem gleichen y-Achsenbereich

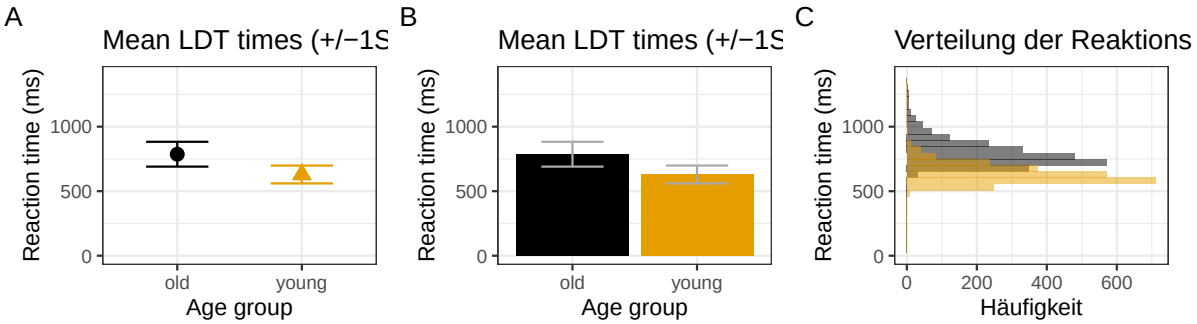


Abbildung 12: Fehlerbargraphik und Balkenplots, die Mittelwerte (+/-1SD) darstellen, sowie ein Histogramm derselben Daten mit demselben y-Achsenbereich

4.4 Zusammenfassende Statistiken und Verteilung

- Fehlerbalken allein sind keine Lösung: auch hier wird eine Menge Information verborgen
  - ein guter Grund, die Rohdatenpunkte *immer* zu visualisieren, unabhängig davon, welche zusammenfassende Darstellung Sie erstellen

Lernziele

In diesem Abschnitt haben wir gelernt, wie man…

- Boxplots erstellen und interpretieren
- Fehlerbalkendiagramme erstellen und interpretieren

Hausaufgabe

Anhang 8: Datenvisualisierung 3 auf der Website des Kurses.

Session Info

Hergestellt mit R version 4.4.0 (2024-04-24) (Puppy Cup) und RStudioversion 2023.9.0.463 (Desert Sunflower).

```
print(sessionInfo(),locale = F)
```

```
R version 4.4.0 (2024-04-24)
Platform: aarch64-apple-darwin20
Running under: macOS Ventura 13.2.1

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRlapack.dylib;  l

attached base packages:
[1] stats     graphics  grDevices datasets  utils     methods   base

other attached packages:
 [1] magick_2.8.3    patchwork_1.2.0 ggthemes_5.1.0  janitor_2.2.0
 [5] here_1.0.1      lubridate_1.9.3 forcats_1.0.0   stringr_1.5.1
 [9] dplyr_1.1.4     purrr_1.0.2     readr_2.1.5     tidyr_1.3.1
[13] tibble_3.2.1    ggplot2_3.5.1   tidyverse_2.0.0

loaded via a namespace (and not attached):
 [1] utf8_1.2.4        generics_0.1.3    renv_1.0.7        stringi_1.8.3
 [5] hms_1.1.3         digest_0.6.35     magrittr_2.0.3    evaluate_0.23
 [9] grid_4.4.0        timechange_0.3.0  fastmap_1.1.1     rprojroot_2.0.4
[13] jsonlite_1.8.8    tinytex_0.50      fansi_1.0.6       scales_1.3.0
[17] cli_3.6.2         crayon_1.5.2      rlang_1.1.3       bit64_4.0.5
[21] munsell_0.5.1     withr_3.0.0       yaml_2.3.8        parallel_4.4.0
[25] tools_4.4.0       tzdb_0.4.0        colorspace_2.1-0  pacman_0.5.1
[29] png_0.1-8         vctrs_0.6.5       R6_2.5.1          lifecycle_1.0.4
[33] snakecase_0.11.1  bit_4.0.5         vroom_1.6.5       pkgconfig_2.0.3
[37] pillar_1.9.0      gtable_0.3.5      glue_1.7.0        Rcpp_1.0.12
[41] xfun_0.43         tidyselect_1.2.1  rstudioapi_0.16.0 knitr_1.46
[45] farver_2.1.1      htmltools_0.5.8.1 labeling_0.4.3    rmarkdown_2.26
[49] compiler_4.4.0
```

Literaturverzeichnis

Nordmann, E., McAleer, P., Toivo, W., Paterson, H., & DeBruine, L. M. (2022). Data Visualization Using R for Researchers Who Do Not Use R. *Advances in Methods and Practices in Psychological Science*, *5*(2), 251524592210746. https://doi.org/10.1177/25152459221074654

Wickham, H., Çetinkaya-Rundel, M., & Grolemund, G. (2023). *R for Data Science* (2. Aufl.).

Winter, B. (2019). Statistics for Linguists: An Introduction Using R. In *Statistics for Linguists: An Introduction Using R*. Routledge. https://doi.org/10.4324/9781315165547