

Deskriptive Statistik

Maße der zentralen Tendenz und Streuung

Daniela Palleschi

Humboldt-Universität zu Berlin

Mi. den 06.12.2023

Lernziele

Heute werden wir lernen...

- über Maße der zentralen Tendenz (Mittelwert, Median, Modus)
- über Streuungsmaße (Bereich, Standardabweichung)
- wie man die Funktion `summarise()` von `dplyr` benutzt
- wie man Zusammenfassungen `.by` Gruppe erstellt

Lektüre

Die erforderliche Lektüre für dieses Thema sind:

1. Kap. 3, Abschnitte 3.4-3.9 (*Descriptive statistics, models, and distributions*) in Winter (2019) (online verfügbar für Studierende/Beschäftigte der HU Berlin über das [HU Grimm Zentrum](#)).
2. [Abschnitt 4.5 \(Groups\)](#) in Kap. 4 (*Data Transformation*) in Wickham et al. (2023).

Einrichten

Umgebung löschen

- Starten Sie ein neues Skript *immer* mit einer leeren R-Umgebung
 - keine Objekte in der Umgebung gespeichert
 - keine Pakete geladen
- Klicken Sie auf **Session > Restart R**, um mit einer neuen Umgebung zu beginnen
 - oder das Tastaturkürzel **Cmd/Ctrl+Strg+0**

Pakete

```
1 pacman::p_load(tidyverse,  
2                 here,  
3                 janitor)
```

Daten laden

- zwei Datensätze heute:
 - `groesse_geburtstag_ws2324.csv`: ein leicht veränderter `groesse_geburtstag`-Datensatz von Winter Semester 2023/2024
 - `languageR_english.csv`: komprimierte Version des `english`-Datensatzes aus dem `languageR`-Paket
- wenn Sie diese Daten noch nicht haben, laden Sie sie von Moodle herunter

```
1 df_groesse <- read_csv(here("daten", "groesse_geburtstag_ws2324.csv"))
```

```
1 df_eng <- read_csv(here("daten", "languageR_english.csv")) |>  
2   clean_names() |>  
3   # fix some wonky variable names:  
4   rename(rt_lexdec = r_tlexdec,  
5           rt_naming = r_tnaming)
```

Deskriptive Statistik

- beschreibt quantitativ die zentrale Tendenz, Variabilität und Verteilung von Daten
 - auch zusammenfassende Statistik genannt
- z.B. Wertebereich (Minimum, Maximum), der Mittelwert und die Standardabweichung

Anzahl der Beobachtungen (n)

- ist keine Statistik, aber eine wichtige Information
 - mehr Daten (höher n) = mehr Beweise
 - weniger Daten (niedriger n) = möglicherweise nicht verallgemeinerbar auf die breitere Population
- `nrow()`: liefert die Anzahl der Beobachtungen in einem Datensatz

```
1 nrow(df_groesse)
```

```
[1] 9
```

- `length()`: die Anzahl der Beobachtungen in einem Vektor oder einer Variablen

```
1 length(df_groesse$groesse)
```

```
[1] 9
```

Maße der zentralen Tendenz (Lagemaße)

- beschreiben quantitativ die Mitte unserer Daten
 - der Mittelwert, der Median und der Modus

Mittelwert (μ)

- der Mittelwert oder Durchschnitt: die Summe aller Werte geteilt durch die Anzahl der Werte (wie in Gleichung ???)

$$\mu = \frac{\text{Summe der Werte}}{n}$$

- können wir die Ergebnisse einer Gleichung als Objekt speichern
 - oder mehrere Werte als Vektor (eine Liste von Werten der gleichen Klasse)

```
1 # save heights as a vector
2 heights <- c(171, 168, 182, 190, 170, 163, 164, 167, 189)
```

- könnten wir dann die Funktionen `sum()` und `length()` verwenden, um den Mittelwert zu berechnen

```
1 # divide the sum of heights by the n of heights
2 sum(heights)/length(heights)
```

```
[1] 173.7778
```

- or simply use the `mean()` function.

```
1 # or use the mean() function
2 mean(heights)
```

```
[1] 173.7778
```

- Wir können die Funktion `mean()` auch auf eine Variable in einem Datenrahmen anwenden, indem wir den Operator `$` verwenden (`datenrahmen$variable`).

```
1 mean(df_groesse$groesse)
```

```
[1] 173.6667
```

Median

- der Wert in der Mitte des Datensatzes
- Wenn Sie Ihre Daten in der Reihenfolge ihrer Werte anordnen, liegt die Hälfte der Daten unter dem Median, die andere Hälfte darüber.

Median in R

- können wir die Funktion `sort()` verwenden und zählen, welches der mittlere Wert ist:

```
1 sort(df_groesse$groesse)
[1] 163 164 167 167 170 171 182 189 190
```

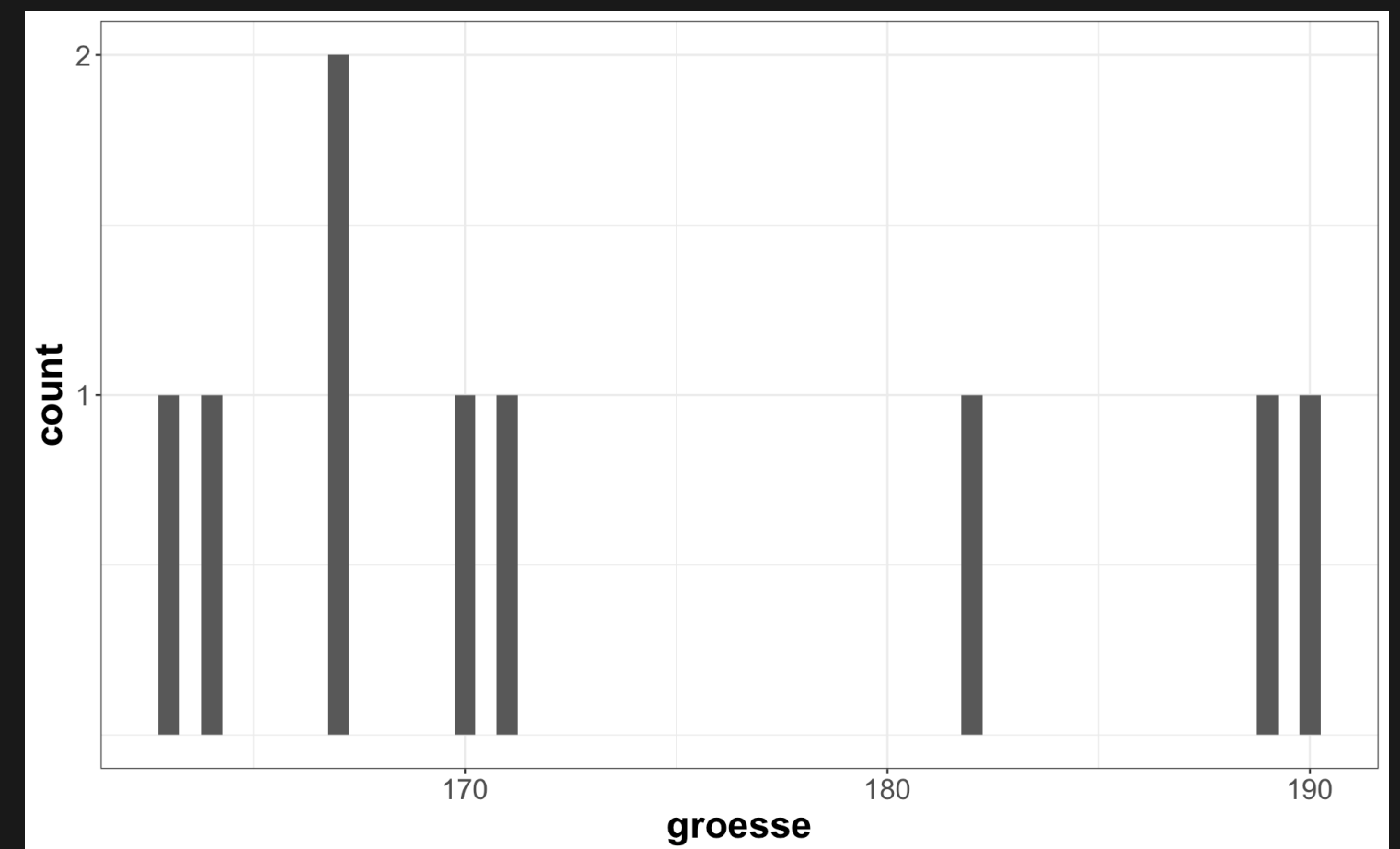
- alternativ könnte man auch einfach die Funktion `median()` verwenden

```
1 median(df_groesse$groesse)
[1] 170
```

Modus

- der Wert, der am häufigsten in einem Datensatz vorkommt
- keine R-Funktion zur Bestimmung des Modus
 - aber wir können ihn visualisieren, z.B. mit einem Histogramm oder einem Dichteplot

```
1 df_groesse |>
2   ggplot(aes(x = groesse)) +
3   geom_histogram(binwidth = .5) +
4   scale_y_continuous(breaks = c(1,2)) +
5   theme_bw() +
6   theme(axis.text = element_text(size = 15),
7         axis.title = element_text(size = 20, f
```



Streuungsmaße

- beschreiben die Streuung von Datenpunkten
 - sagen uns etwas darüber, wie die Daten insgesamt verteilt sind

Bereich

- kann sich auf den höchsten (Maximum) und den niedrigsten (Minimum) Wert beziehen
 - oder die Differenz zwischen höchstem und niedrigstem Wert

- `max()` und `min()`: gibt den höchsten und den niedrigsten Wert aus

```
1 max(heights)
```

```
[1] 190
```

```
1 min(heights)
```

```
[1] 163
```

- oder die Funktion `range()` verwenden

```
1 range(heights)
```

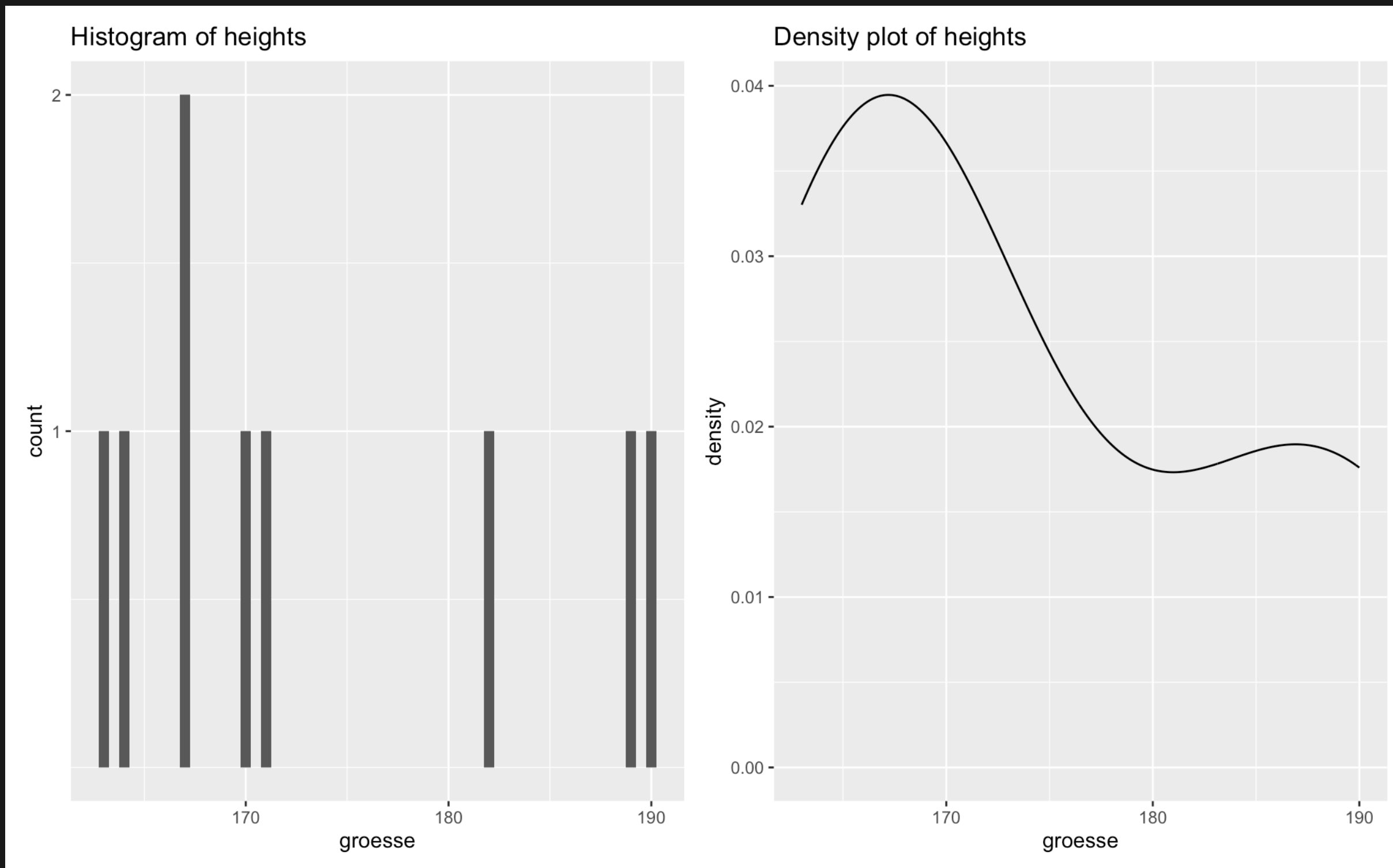
```
[1] 163 190
```

- Die Differenz zwischen diesen Werten erhält man, indem man den Minimalwert vom Maximalwert subtrahiert

```
1 max(heights) - min(heights)
```

```
[1] 27
```

- In einem Histogramm oder Dichteplot: die niedrigsten und höchsten Werte auf der x-Achse



Standardabweichung (**sd** oder σ)

- ein Maß für die Streuung der Daten *im Verhältnis zum Mittelwert*
 - eine niedrige Standardabweichung bedeutet, dass die Daten um den Mittelwert herum gruppiert sind (d.h. es gibt eine geringere Streuung)
 - eine hohe Standardabweichung bedeutet, dass die Daten stärker gestreut sind
- Die Standardabweichung wird sehr oft angegeben, wenn der Mittelwert angegeben wird.

- Standardabweichung (**sd**) = die Quadratwurzel ($\sqrt{}$ oder **sqrt()** in R) der Summe der quadrierten Wertabweichungen vom Mittelwert $((x - \mu)^2)$ geteilt durch die Anzahl der Beobachtungen minus 1 ($n - 1$)
 - gegeben in Gleichung ???

$$\sigma = \sqrt{\frac{(x_1 - \mu)^2 + (x_2 - \mu)^2 + \dots + (x_N - \mu)^2}{N - 1}}$$

- das sieht einschüchternd aus, aber wir können die Standardabweichung in R mit der Funktion **sd()** berechnen

```
1 sd(heights)
```

```
[1] 10.46157
```

- wir können die Standardabweichung von Hand berechnen, wenn wir wissen:
 - den Wert der einzelnen Beobachtungen
 - den Mittelwert dieser Werte
 - die Anzahl der Beobachtungen

$$\sigma_{\text{heights}} = \sqrt{\frac{(\text{height}_1 - \mu)^2 + (\text{height}_2 - \mu)^2 + \dots + (\text{height}_N - \mu)^2}{N - 1}}$$

- In einem Vektor mit 3 Beobachtungen (3, 5, 9) sind unsere Werte (x) zum Beispiel folgende:

```
1 values <- c(3,5,16)
2 values
```

```
[1] 3 5 16
```

- Wenn wir diese zu Gleichung ??? hinzufügen, erhalten wir Gleichung ???

$$\sigma_{\text{values}} = \sqrt{\frac{(3 - \mu)^2 + (5 - \mu)^2 + (16 - \mu)^2}{N - 1}}$$

- unser Mittelwert (μ) ist:

```
1 mean(values)
```

```
[1] 8
```

- Wenn wir diese zu Gleichung ??? hinzufügen, erhalten wir Gleichung ???.

$$\sigma_{\text{values}} = \sqrt{\frac{(3 - 8)^2 + (5 - 8)^2 + (16 - 8)^2}{N - 1}}$$

- die Anzahl der Beobachtungen (n) ist:

```
1 length(values)
```

```
[1] 3
```

- Wenn wir diese zu Gleichung ??? hinzufügen, erhalten wir Gleichung ???

$$\sigma_{\text{values}} = \sqrt{\frac{(3 - 8)^2 + (5 - 8)^2 + (16 - 8)^2}{3 - 1}}$$

- Wenn wir die restlichen Operationen durchführen, erhalten wir die Gleichungen ??? bis ???:

$$\begin{aligned}\sigma_{\text{values}} &= \sqrt{\frac{(-5)^2 + (-3)^2 + (8)^2}{3 - 1}} \\ &= \sqrt{\frac{25 + 9 + 64}{3 - 1}} \\ &= \sqrt{\frac{98}{2}} \\ &= \sqrt{49} \\ &= 7\end{aligned}$$

- unsere Arbeit überprüfen:

```
1 sd(values)
```

```
[1] 7
```

Deskriptive Statistiken mit R

- das Paket `dplyr` aus dem `tidyverse` hat einige hilfreiche Funktionen, um zusammenfassende Statistiken zu erstellen
- Lassen Sie uns nun den `df_eng`-Datensatz verwenden, um diese `dplyr`-Verben kennenzulernen

dplyr::summarise

- Die Funktion `summarise()` (`dplyr`) berechnet Zusammenfassungen von Daten
 - aber wir müssen ihr sagen, *was* sie berechnen soll, und für welche Variable(n)
- die Funktion `n()` zum Beispiel liefert die Anzahl der Beobachtungen (nur wenn sie innerhalb von `summarise()` oder `mutate()` verwendet wird)

```
1 df_eng |>  
2   summarise(N = n())
```

```
# A tibble: 1 × 1  
      N  
  <int>  
1  4568
```

- wir können auch mehrere Berechnungen auf einmal durchführen
 - Ermitteln wir auch den Mittelwert und die Standardabweichung der lexikalischen Entscheidungsaufgabe (**rt_lexdec**, in Millisekunden)

```
1 df_eng |>
2   summarise(mean_lexdec = mean(rt_lexdec, na.rm=T),
3             sd_lexdec = sd(rt_lexdec, na.rm = T),
4             N = n())
```

```
# A tibble: 1 × 3
  mean_lexdec sd_lexdec      N
  <dbl>      <dbl> <int>
1    708.      115.  4568
```

Fehlende Werte

- Berechnungen sind bei fehlenden Werten nicht möglich
 - die Variable `rt_naming` hat einen fehlenden Wert
 - die Funktion `mean()` funktioniert nicht mit fehlenden Werten

```
1 df_eng |>  
2   summarise(mean_naming = mean(rt_naming))
```

```
# A tibble: 1 × 1  
  mean_naming  
    <dbl>  
1         NA
```

- können wir sie mit dem Verb `drop_na()` entfernen

```
1 df_eng |>  
2   drop_na() |>  
3   summarise(mean_naming = mean(rt_naming))
```

```
# A tibble: 1 × 1  
  mean_naming  
    <dbl>  
1       566.
```


Variablen gruppieren

- Wir wollen normalerweise bestimmte Gruppen *vergleichen*.
 - z. B. den Vergleich von “Groesse” zwischen L1-Sprechergruppen

.by =

- das Argument `.by =` in `summarise()` berechnet unsere Berechnungen für Gruppen innerhalb einer kategorialen Variable

```
1 df_eng |>
2   drop_na() |>
3   summarise(mean_lexdec = mean(rt_lexdec),
4             sd_lexdec = sd(rt_lexdec),
5             N = n(),
6             .by = age_subject) |>
7   arrange(mean_lexdec)
```

A tibble: 2 × 4

	age_subject	mean_lexdec	sd_lexdec	N
	<chr>	<dbl>	<dbl>	<int>
1	young	630.	69.1	2283
2	old	787.	96.2	2284

Gruppieren nach mehreren Variablen

- wir können auch nach mehreren Variablen gruppieren
 - dafür brauchen wir **Verkettung** (`c()`)

```

1 df_eng |>
2   drop_na() |>
3   summarise(mean_lexdec = mean(rt_lexdec),
4             sd_lexdec = sd(rt_lexdec),
5             N = n(),
6             .by = c(age_subject, word_category)
7   arrange(age_subject)

```

```

# A tibble: 4 × 5
  age_subject word_category mean_lexdec sd_lexdec      N
  <chr>        <chr>          <dbl>    <dbl> <int>
1 old         N             790.    101.  1452
2 old         V             780.     86.5   832
3 young       N             633.     70.8  1451
4 young       V             623.     65.7   832

```

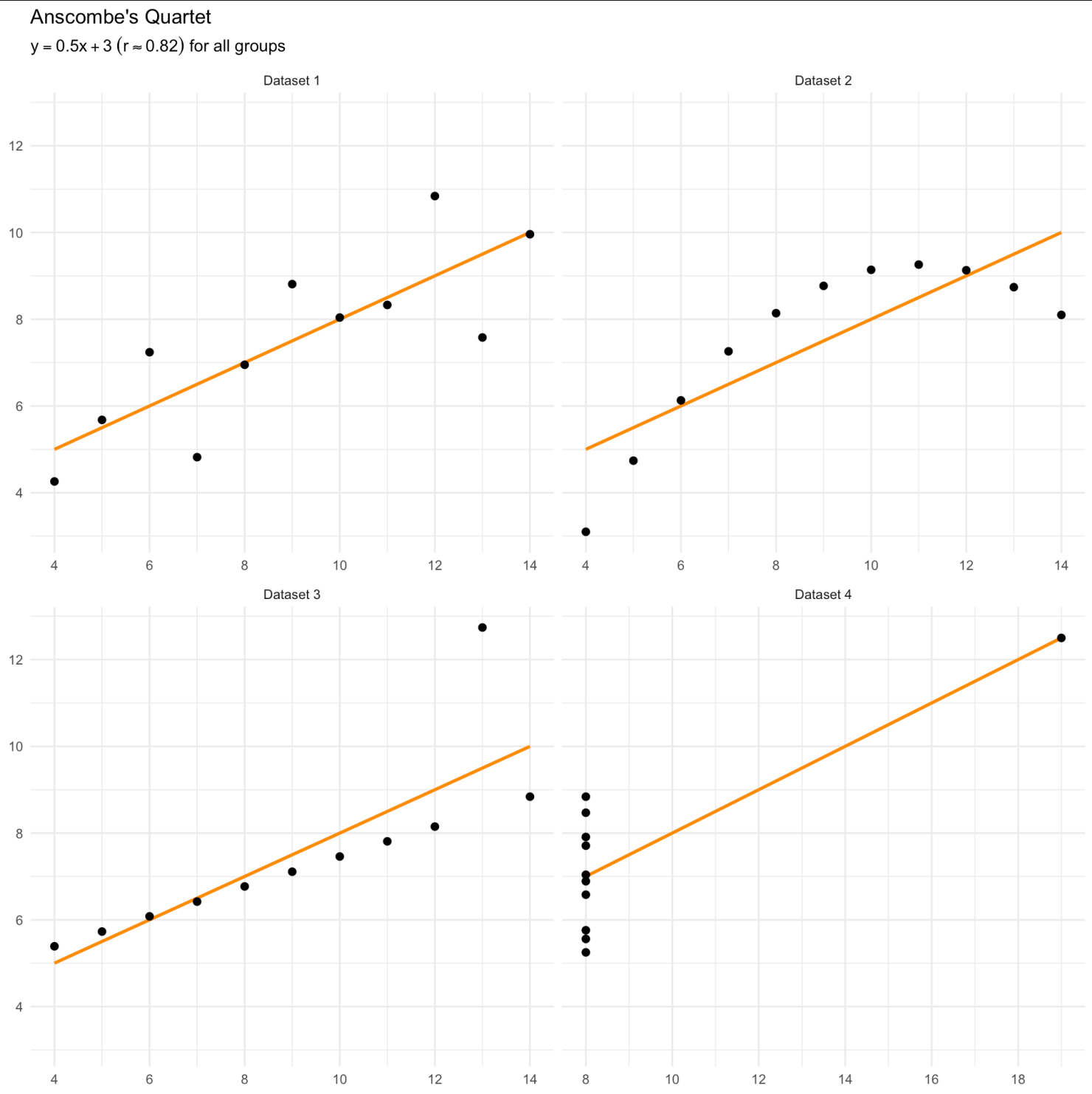
Das Quartett von Anscombe

- Francis Anscombe konstruierte 1973 4 Datensätze, um zu veranschaulichen, wie wichtig es ist, Daten zu visualisieren, bevor man sie analysiert und ein Modell erstellt
- Diese vier Diagramme stellen 4 Datensätze dar, die alle einen nahezu identischen Mittelwert und eine Standardabweichung, aber sehr unterschiedliche Verteilungen aufweisen

Tabelle 1: Summary stats of Anscombe's quartet datasets

dataset	mean_x	mean_y
Dataset 1	9	7.5
Dataset 2	9	7.5
Dataset 3	9	7.5
Dataset 4	9	7.5

Abbildung 1: Plots of Anscombe's quartet distributions



DatasaurRus

- datasauRus-Paket ([Davies et al., 2022](#)) enthält einige weitere Datensätze, die ähnliche Mittelwerte und Standardabweichung, aber unterschiedliche Verteilungen haben
 - angegeben in [Tabelle 2](#)

```
1 pacman::p_load("datasauRus")
```

Tabelle 2: Zusammenfassende Statistiken der datasauRus-Datensätze

dataset	mean_x	mean_y	std_dev_x	std_dev_y	corr_x_y
away	54.27	47.83	16.77	26.94	-0.06
bullseye	54.27	47.83	16.77	26.94	-0.07
circle	54.27	47.84	16.76	26.93	-0.07
dino	54.26	47.83	16.77	26.94	-0.06
dots	54.26	47.84	16.77	26.93	-0.06
h_lines	54.26	47.83	16.77	26.94	-0.06
high_lines	54.27	47.84	16.77	26.94	-0.07
slant_down	54.27	47.84	16.77	26.94	-0.07
slant_up	54.27	47.83	16.77	26.94	-0.07
star	54.27	47.84	16.77	26.93	-0.06
v_lines	54.27	47.84	16.77	26.94	-0.07
wide_lines	54.27	47.83	16.77	26.94	-0.07
x_shape	54.26	47.84	16.77	26.93	-0.07

- aber wenn wir sie aufzeichnen, sehen sie alle sehr unterschiedlich aus ([Abbildung 2](#))!

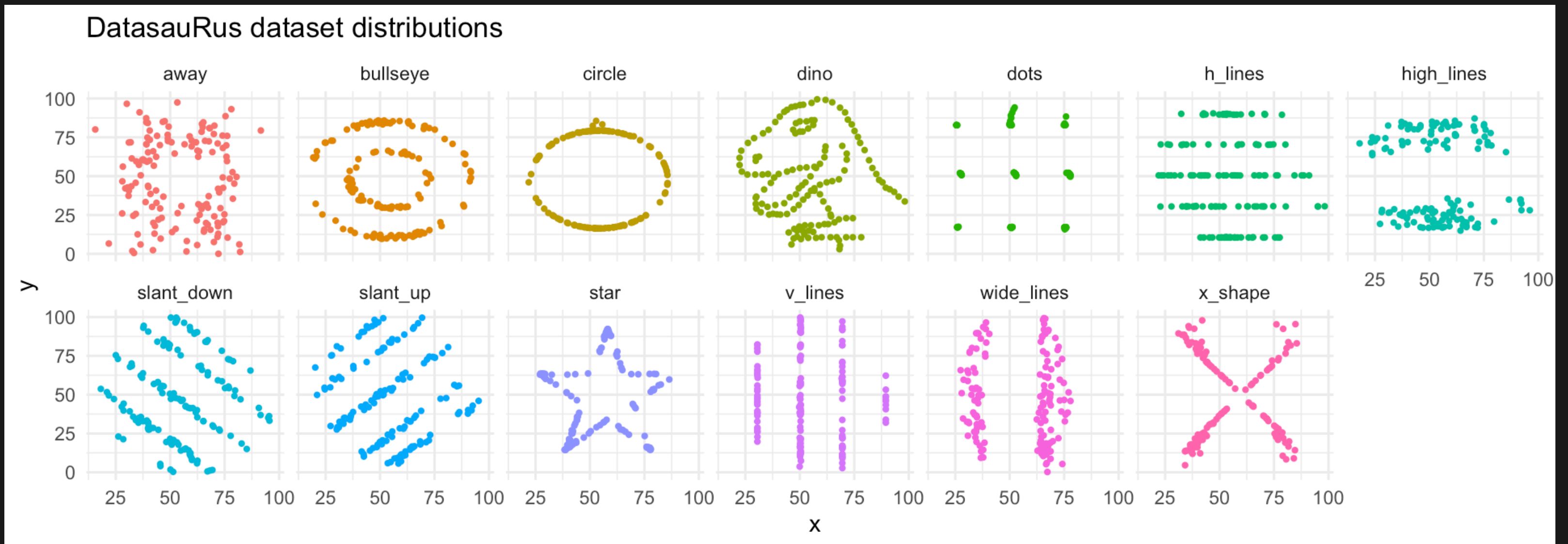


Abbildung 2: Plots of datasauRus dataset distributions

- Also, *immer die Daten aufzeichnen*
 - Schauen Sie sich nicht nur die deskriptiven Statistiken an!
- Beides ist sehr wichtig für das Verständnis Ihrer Daten.
- Nächste Woche sehen wir uns an, wie wir unsere zusammenfassenden Statistiken darstellen

Learning objectives

Heute haben wir gelernt...

- über Maße der zentralen Tendenz ✓
- über Streuungsmaße ✓
- wie man die Funktion `summarise()` von `dplyr` benutzt ✓
- wie man Zusammenfassungen `.by` Gruppe erstellt ✓

Hausaufgaben

Anhang 7: Deskriptive Statistik auf der Website des Kurses.

Session Info

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

```
1 sessionInfo()
```

```
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; LAPACK version 3.12.0
```

```
locale:
```

```
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
```

```
time zone: Europe/Berlin
```

```
tzcode source: internal
```

```
attached base packages:
```

```
stats      graphics  grDevices  utils      datasets  MASS      Matrix
```

Literaturverzeichnis

Davies, R., Locke, S., & D'Agostino McGowan, L. (2022). *datasauRus: Datasets from the Datasaurus Dozen*. <https://CRAN.R-project.org/package=datasauRus>

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>

