Deskriptive Statistik

Maße der zentralen Tendenz und Streuung

Daniela Palleschi

Mi. den 06.12.2023

Inhaltsverzeichnis

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

1 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).

2 Einrichten

3 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`

3.1 Pakete

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

3.2 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

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

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

4 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

4.1 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

```
nrow(df_groesse)
```

```
[1] 9
```

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

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

```
[1] 9
```

4.2 Maße der zentralen Tendenz (Lagemaße)

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

4.2.1 Mittelwert (μ)

- der Mittelwert oder Durchschnitt: die Summe aller Werte geteilt durch die Anzahl der Werte (wie in Gleichung 1)

$$\mu = \frac{Summe\ der\ Werte}{n} \tag{1}$$

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

```
# save heights as a vector
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

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

```
[1] 173.7778
```

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

```r
# or use the mean() function
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`).

```r
mean(df_groesse$groesse)
```

```
[1] 173.6667
```

4.2.2 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.

4.2.2.1 Median in R

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

```r
sort(df_groesse$groesse)
```

```
[1] 163 164 167 167 170 171 182 189 190
```

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

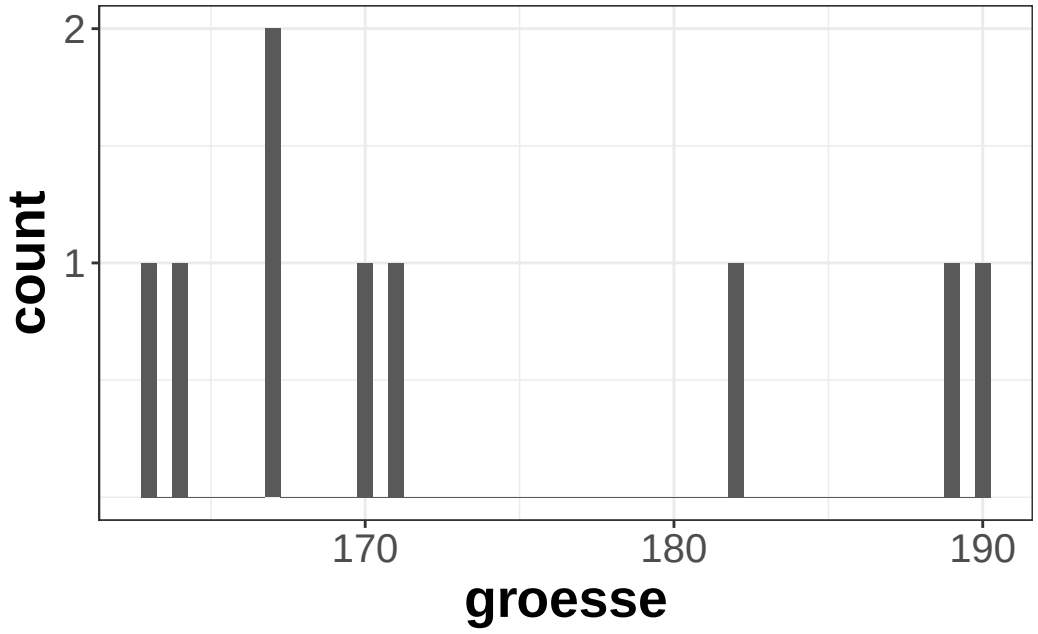
```r
median(df_groesse$groesse)
```

```
[1] 170
```

4.2.3 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

```r
df_groesse |>
  ggplot(aes(x = groesse)) +
  geom_histogram(binwidth = .5) +
  scale_y_continuous(breaks = c(1,2)) +
  theme_bw() +
  theme(axis.text = element_text(size = 15),
        axis.title = element_text(size = 20, face = "bold"))
```



4.3 Streuungsmaße

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

4.3.1 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

```
max(heights)
```

```
[1] 190
```

```
min(heights)
```

```
[1] 163
```

- oder die Funktion `range()` verwenden

```
range(heights)
```

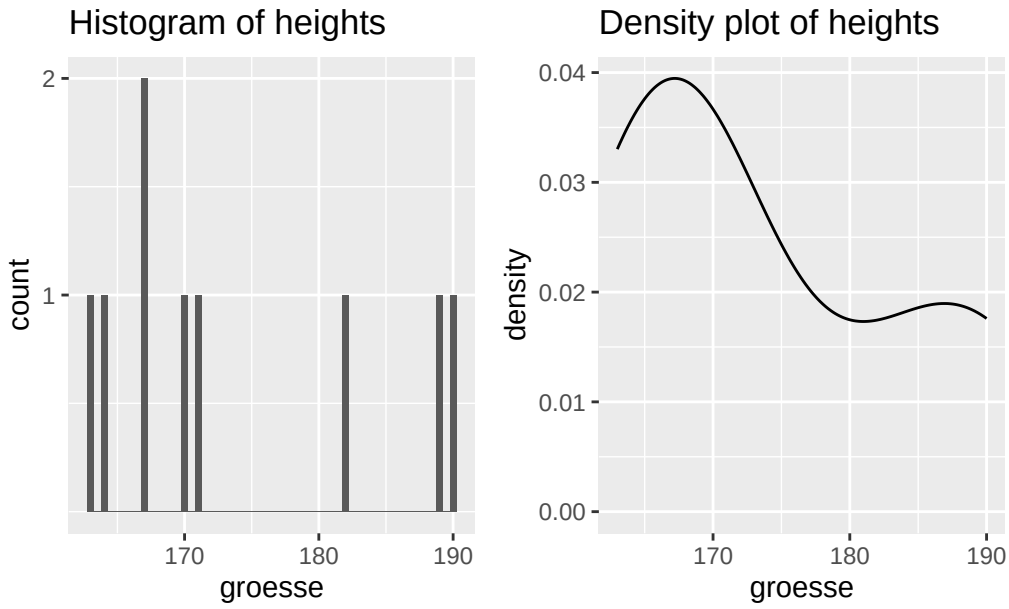
```
[1] 163 190
```

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

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

```
[1] 27
```

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



4.3.2 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 2

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

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

```r
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

$$\sigma_{heights} = \sqrt{\frac{(height_1 - \mu)^2 + (height_2 - \mu)^2 + ...(heights_N - \mu)^2}{N - 1}} \tag{3}$$

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

```r
values <- c(3,5,16)
values
```

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

- Wenn wir diese zu Gleichung 2 hinzufügen, erhalten wir Gleichung 4

$$\sigma_{values} = \sqrt{\frac{(3 - \mu)^2 + (5 - \mu)^2 + (16 - \mu)^2}{N - 1}} \tag{4}$$

- unser Mittelwert (μ) ist:

```r
mean(values)
```

```
[1] 8
```

- Wenn wir diese zu Gleichung 4 hinzufügen, erhalten wir Gleichung 5.

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

- die Anzahl der Beobachtungen (n) ist:

```
length(values)
```

```
[1] 3
```

- Wenn wir diese zu Gleichung 5 hinzufügen, erhalten wir Gleichung 6

$$\sigma_{values} = \sqrt{\frac{(3-8)^2 + (5-8)^2 + (16-8)^2}{3-1}} \tag{6}$$

- Wenn wir die restlichen Operationen durchführen, erhalten wir die Gleichungen 8 bis 2:

$$\begin{aligned}
\sigma_{values} &= \sqrt{\frac{(-5)^2 + (-3)^2 + (8)^2}{3-1}} && (7)\\
& && (8)\\
&= \sqrt{\frac{25 + 9 + 64}{3-1}} && (9)\\
&= \sqrt{\frac{98}{2}} && (10)\\
&= \sqrt{49} && (11)\\
&= 7 && (12)
\end{aligned}$$

- unsere Arbeit überprüfen:

```
sd(values)
```

```
[1] 7
```

5 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

5.1 `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)

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

```
# A tibble: 1 x 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)

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

```
# A tibble: 1 x 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

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

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

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

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

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

6 Variablen gruppieren

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

6.1 `.by =`

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

```
df_eng |>
  drop_na() |>
  summarise(mean_lexdec = mean(rt_lexdec),
```

```
            sd_lexdec = sd(rt_lexdec),
            N = n(),
            .by = age_subject) |>
  arrange(mean_lexdec)
```

```
# A tibble: 2 x 4
  age_subject mean_lexdec sd_lexdec     N
  <chr>             <dbl>     <dbl> <int>
1 young              630.      69.1  2283
2 old                787.      96.2  2284
```

6.2 Gruppieren nach mehreren Variablen

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

```
df_eng |>
  drop_na() |>
  summarise(mean_lexdec = mean(rt_lexdec),
            sd_lexdec = sd(rt_lexdec),
            N = n(),
            .by = c(age_subject, word_category)) |>
  arrange(age_subject)
```

```
# A tibble: 4 x 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
```

7 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 quratet 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

7.1 DatasaurRus

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

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

Tabelle 2: Zusammenfassende Statistiken der datasauRus-Datensätze

dataset	mean_x	mean_y	std_dev_x	std_dev_y	corr_x
away	54.27	47.83	16.77	26.94	-0
bullseye	54.27	47.83	16.77	26.94	-0
circle	54.27	47.84	16.76	26.93	-0
dino	54.26	47.83	16.77	26.94	-0
dots	54.26	47.84	16.77	26.93	-0

h_lines	54.26	47.83	16.77	26.94	-0
high_lines	54.27	47.84	16.77	26.94	-0
slant_down	54.27	47.84	16.77	26.94	-0
slant_up	54.27	47.83	16.77	26.94	-0
star	54.27	47.84	16.77	26.93	-0
v_lines	54.27	47.84	16.77	26.94	-0
wide_lines	54.27	47.83	16.77	26.94	-0
x_shape	54.26	47.84	16.77	26.93	-0

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

- 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

8 Hausufgaben

Anhang 7: Deskriptive Statistik auf der Website des Kurses.

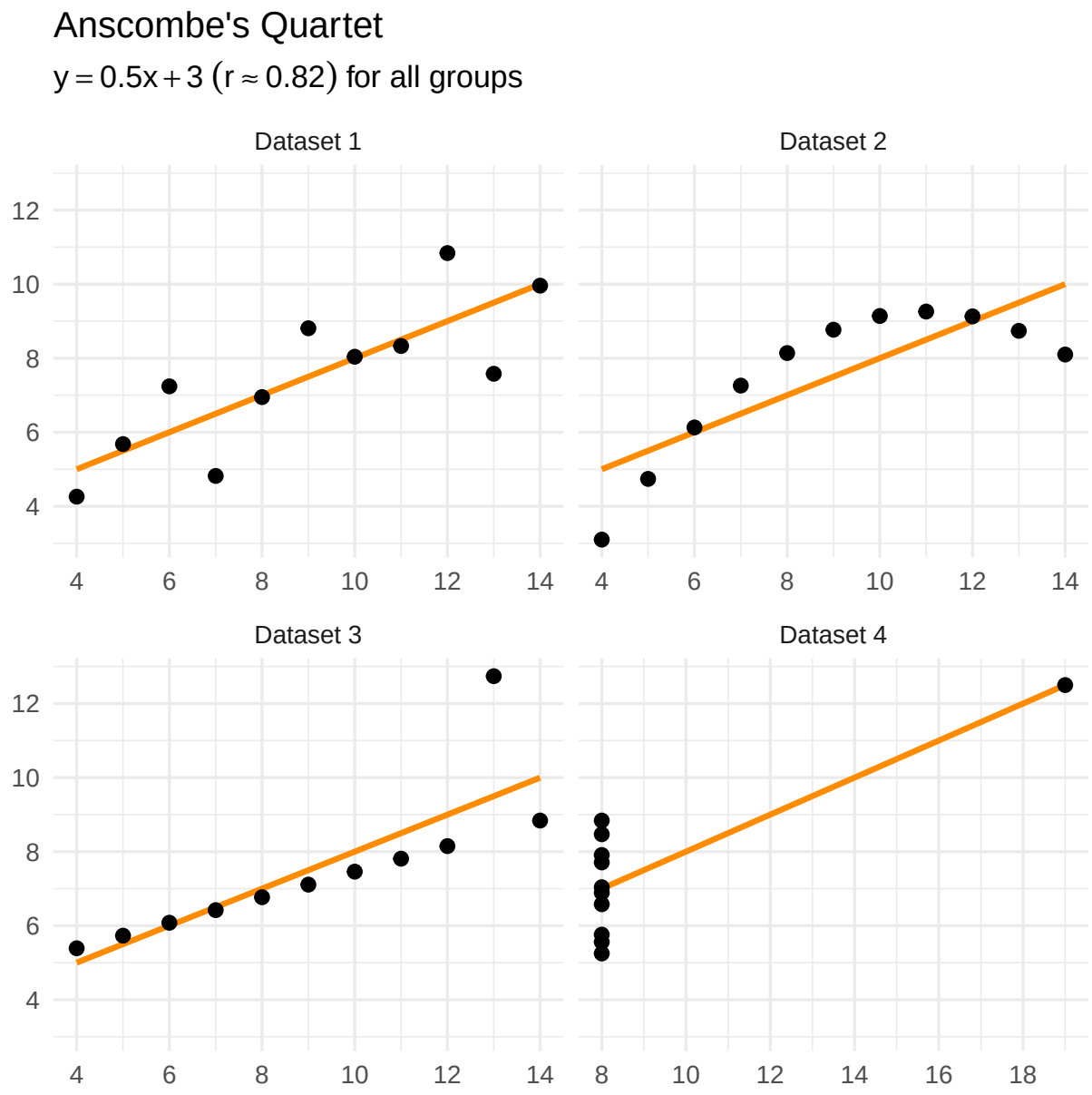


Abbildung 1: Plots of Anscombe's quratet distributions

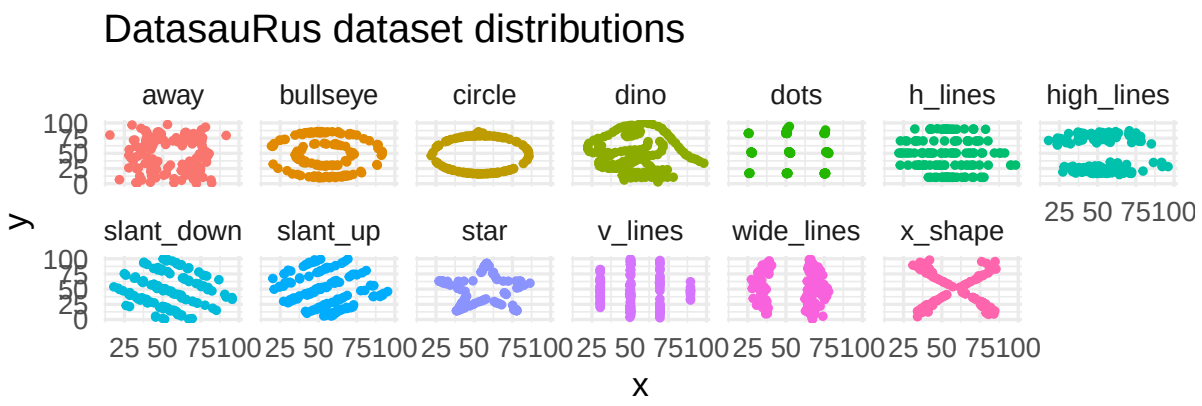


Abbildung 2: Plots of datasauRus dataset distributions

Session Info

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

```
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;

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:
[1] stats     graphics  grDevices datasets  utils     methods   base

other attached packages:
 [1] datasauRus_0.1.8 patchwork_1.2.0  janitor_2.2.0    here_1.0.1
 [5] lubridate_1.9.3  forcats_1.0.0    stringr_1.5.1    dplyr_1.1.4
 [9] purrr_1.0.2      readr_2.1.5      tidyr_1.3.1      tibble_3.2.1
```

```
[13] 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        xml2_1.3.6
 [5] lattice_0.22-6    stringi_1.8.3     hms_1.1.3         digest_0.6.35
 [9] magrittr_2.0.3    evaluate_0.23     grid_4.4.0        timechange_0.3.0
[13] fastmap_1.1.1     Matrix_1.7-0      rprojroot_2.0.4   jsonlite_1.8.8
[17] tinytex_0.50      mgcv_1.9-1        fansi_1.0.6       viridisLite_0.4.2
[21] scales_1.3.0      cli_3.6.2         rlang_1.1.3       crayon_1.5.2
[25] splines_4.4.0     bit64_4.0.5       munsell_0.5.1     withr_3.0.0
[29] yaml_2.3.8        parallel_4.4.0    tools_4.4.0       tzdb_0.4.0
[33] colorspace_2.1-0  pacman_0.5.1      kableExtra_1.4.0  vctrs_0.6.5
[37] R6_2.5.1          lifecycle_1.0.4   snakecase_0.11.1  bit_4.0.5
[41] vroom_1.6.5       pkgconfig_2.0.3   pillar_1.9.0      gtable_0.3.5
[45] glue_1.7.0        systemfonts_1.0.6 xfun_0.43         tidyselect_1.2.1
[49] rstudioapi_0.16.0 knitr_1.46        farver_2.1.1      nlme_3.1-164
[53] htmltools_0.5.8.1 svglite_2.1.3     labeling_0.4.3    rmarkdown_2.26
[57] compiler_4.4.0
```

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