Daten Typen & Vektoren & For Loops

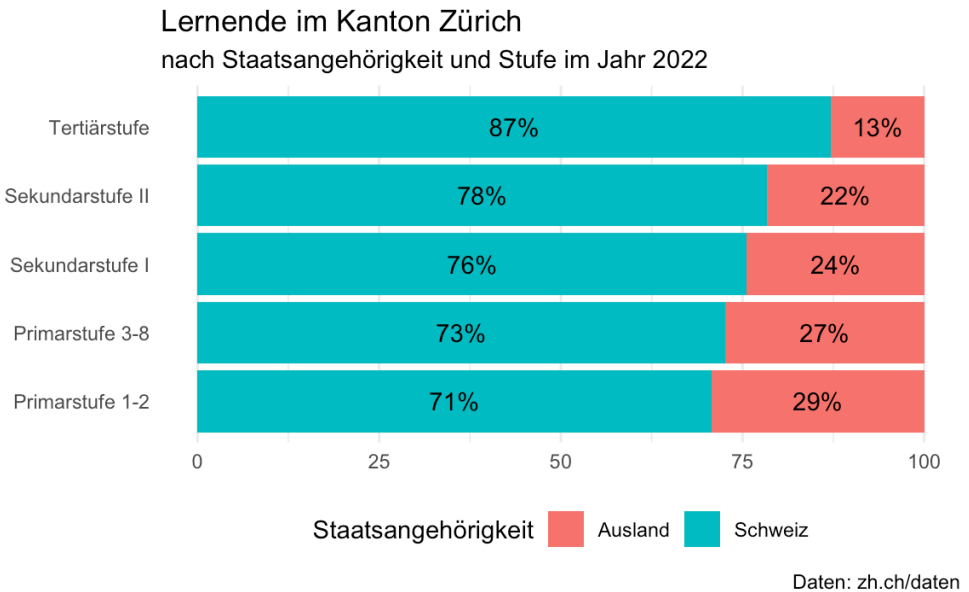
rstatsZH - Data Science mit R

Lars Schöbitz

Oct 29, 2024

Modul 5 - Zusatzaufgabe 3

```r
ggplot(data = lernende2022_stufe_staat_sum,
       mapping = aes(x = Stufe,
                     y = Prozent,
                     fill = Staatsangehoerigkeit)) +
    coord_flip() +
    geom_col() +
    geom_text(aes(label = paste0(round(Prozent, 0), "%")),
              position = position_stack(vjust = 0.5)) +
    labs(title = "Lernende im Kanton Zürich ",
         subtitle = "nach Staatsangehörigkeit und Stufe im Jahr
         fill = "Staatsangehörigkeit",
         caption = "Daten: zh.ch/daten",
         y = NULL,
         x = NULL) +
    theme_minimal() +
    theme(legend.position = "bottom",
          panel.grid.major.y = element_blank())
```



Lernziele (für diese Woche)

1. Die Lernenden können die Bedeutung von Vektoren mit Bezug auf einen Dataframe erläutern.
2. Die Lernenden können drei verschiedene Methoden anwenden um auf einen Vektor in einem dataframe zuzugreifen.
3. Die Lernenden können die vier wichtigsten atomaren Vektortypen in R auflisten.
4. Die Lernenden können einen for loop verwenden, um durch die Elemente eines Vektors in einem Dataframe zu iterieren und spezifische Operationen auf jedes Element anzuwenden.

Daten Typen und Vektoren

Why care about data types? Warum sind Daten Typen wichtig?

Beispiel: Recycling Umfrage in Zürich

Eine Umfrage zum Recycling-Verhalten in der Stadt Zürich:

- `job`: Was ist dein Beruf?
- `price_glass`: Welchen monatlichen Betrag wärst du bereit für eine Metall/Glas-Tonne vor deinem Haus zu zahlen?

id	job	price_glass
1	Student	0
2	Retired	0
3	Other	0
4	Employed	10
5	Employed	See comment
6	Student	5-10
7	Student	0
8	Retired	0
9	Student	10

id	job	price_glass
10	Employed	0
11	Employed	20 (2chf per person with 10 people in the WG)
12	Student	10
13	Student	10
14	Employed	0
15	Student	10
16	Student	0
17	Employed	5-10
18	Other	0
19	Student	0
20	Employed	10
21	Employed	0
22	Employed	5

Oh warum klappt das nicht?!

```
1 survey_data_small |>
2   summarise(mean_price_glass = mean(price_glass))
```

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

Oh warum klappt das immernoch nicht!!??

```
1 survey_data_small |>
2   summarise(mean_price_glass = mean(price_glass, na.rm = TRUE))
```

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

Atme tief durch und schau dir deine Daten an

id	job	price_glass
1	Student	0
2	Retired	0
3	Other	0
4	Employed	10
5	Employed	See comment
6	Student	5-10
7	Student	0
8	Retired	0
9	Student	10
10	Employed	0
11	Employed	20 (2chf per person with 10 people in the WG)
12	Student	10
13	Student	10
14	Employed	0

rstatszh-k009.github.io/website/

id	job	price_glass
15	Student	10
16	Student	0
17	Employed	5-10
18	Other	0
19	Student	0
20	Employed	10
21	Employed	0
22	Employed	5

Atme tief durch und schau dir deine Daten an

```
# A tibble: 22 × 3
      id job      price_glass
   <int> <chr>    <chr>
 1     1 Student  0
 2     2 Retired  0
 3     3 Other    0
 4     4 Employed 10
 5     5 Employed See comment
 6     6 Student  5-10
 7     7 Student  0
 8     8 Retired  0
 9     9 Student  10
10    10 Employed 0
# ℹ 12
```

Ein sehr typischer Schritt in der Datenbereinigung!

```
survey_data_small |>
  mutate(price_glass_new = case_when(
    price_glass == "5-10" ~ "7.5",
    price_glass == "05-Oct" ~ "7.5",
    str_detect(price_glass, pattern = "2chf") == TRUE ~ "20",
    str_detect(price_glass, pattern = "See comment") == TRUE ~ NA_character_,
    TRUE ~ price_glass
  ))
```

Ein sehr typischer Schritt in der Datenbereinigung!

id	job	price_glass_new	price_glass
1	Student	0	0
2	Retired	0	0
3	Other	0	0
4	Employed	10	10
5	Employed	NA	See comment
6	Student	7.5	5-10
7	Student	0	0
8	Retired	0	0
9	Student	10	10
10	Employed	0	0
11	Employed	20	20 (2chf per person with 10 people in the WG)
12	Student	10	10
13	Student	10	10

id	job	price_glass_new	price_glass
14	Employed	0	0
15	Student	10	10
16	Student	0	0
17	Employed	7.5	5-10
18	Other	0	0
19	Student	0	0
20	Employed	10	10
21	Employed	0	0
22	Employed	5	5

Summarise? Argh!!!!

```r
survey_data_small |>
  mutate(price_glass_new = case_when(
    price_glass == "5-10" ~ "7.5",
    price_glass == "05-Oct" ~ "7.5",
    str_detect(price_glass, pattern = "20") == TRUE ~ "20",
    str_detect(price_glass, pattern = "See comment") == TRUE ~ NA_character_,
    TRUE ~ price_glass
  )) |>
  summarise(mean_price_glass = mean(price_glass_new, na.rm = TRUE))
```

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

Respektiere deine Daten Typen!

> ⚠ Den Durchschnitt von einem Vektor mit Typ **“character”** zu berechnen ist nicht möglich.

```
# A tibble: 22 × 4
      id job      price_glass price_glass_new
   <int> <chr>    <chr>       <chr>
 1     1 Student  0           0
 2     2 Retired  0           0
 3     3 Other    0           0
 4     4 Employed 10          10
 5     5 Employed See comment <NA>
 6     6 Student  5-10        7.5
 7     7 Student  0           0
 8     8 Retired  0           0
 9     9 Student  10          10
10    10 Employed 0           0
# ℹ 12 more rows
```

Respektiere deine Daten Typen!

```r
survey_data_small |>
  mutate(price_glass_new = case_when(
    price_glass == "5-10" ~ "7.5",
    price_glass == "05-Oct" ~ "7.5",
    str_detect(price_glass, pattern = "20") == TRUE ~ "20",
    str_detect(price_glass, pattern = "See comment") == TRUE ~ NA_character_,
    TRUE ~ price_glass
  )) |>
  mutate(price_glass_new = as.numeric(price_glass_new)) |>
  summarise(mean_price_glass = mean(price_glass_new, na.rm = TRUE))
```

```
# A tibble: 1 × 1
  mean_price_glass
             <dbl>
1             4.76
```

Ich bin dran: Vektoren und Iteration mit for-Schleifen

Zurücklehnen und Fragen stellen!

rstatszh-k009.github.io/website/

30:00

Pause machen

Bitte steh auf und beweg dich. Lasst eure E-Mails in Frieden ruhen.

Ihr seid dran: `02-vektor-typen-ihr.qmd`

1. Öffne posit.cloud in deinem Browser (verwende dein Lesezeichen).
2. Öffne den rstatszh-k009 Arbeitsbereich (Workspace) für den Kurs.
3. Klicke auf `Start` neben `md-06-uebungen`.
4. Suche im Dateimanager im Fenster unten rechts die Datei `02-vektor-typen-ihr.qmd` und klicke darauf, um sie im Fenster oben links zu öffnen.
5. Folge den Anweisungen in der Datei.

Zeitpuffer: Modul 6 Uebungen

Kann ich noch etwas zu den Übungen in `02-vektor-typen-ihr.qmd` sagen?

Pause machen

Bitte steh auf und beweg dich. Lasst eure E-Mails in Frieden ruhen.

Sensitive Daten und GitHub

schützenswerte Daten dürfen nicht auf GitHub

schützenswerte Daten:

- verletzen die Privatsphäre (z.B. Einzeldaten)
- sind sicherheitskritisch (z.B. Passwörter)
- unterliegen Drittrechten (z.B. Copyrights)

Lösung: `.gitignore`

- Dateien und Verzeichnisse in `.gitignore` eintragen
- werden nicht auf GitHub hochgeladen

Daten teilen

Damit eine Analyse reproduzierbar ist, müssen die Daten für andere zugänglich sein. Die Dateien können auf anderen Wegen geteilt werden, z.B. per E-Mail, USB-Stick, Cloud-Dienst, etc.

Informationssicherheit

Folgender Dateipfad enthält Informationen zum Dateisystem und sollte nicht auf GitHub hochgeladen werden:

```
1 read_csv("C:/Users/Lars/Documents/projekt-umfrage/daten/umfrage_daten.csv")
```

Ein guter Weg dies zu vermeiden ist die Verwendung von relativen Pfaden in Kombination mit der `here()` Funktion aus dem gleichnamigen R-Paket `here`. Im RStudio Project / GitHub Repository mit dem Namen `projekt-umfrage`:

```
1 read_csv(here::here("daten/umfrage_daten.csv"))
```

Wir sind dran: `03-gitignore-wir.qmd` & `docs/04-dateipfade.qmd`

1. Öffne posit.cloud in deinem Browser (verwende dein Lesezeichen).
2. Öffne den rstatszh-k009 Arbeitsbereich (Workspace) für den Kurs.
3. Klicke auf `Continue` neben `md-06-uebungen`.
4. Suche im Dateimanager im Fenster unten rechts die Datei `03-gitignore-wir.qmd` und klicke darauf, um sie im Fenster oben links zu öffnen.

Zeitpuffer: Modul 6 Uebungen

Kann ich noch etwas zum heutigen Modul erklären?

Zusatzaufgaben Modul 6

Modul 6 Dokumentation

rstatszh-k009.github.io/website/module/md-06.html

Zusatzaufgaben Abgabedatum

- Abgabedatum: Montag, 04. November
- Korrektur- und Feedbackphase bis zu: Donnerstag, 07. November

Danke

rstatszh-k009.github.io/website/

Danke! 🌻

Folien erstellt mit revealjs und Quarto: https://quarto.org/docs/presentations/revealjs/ Access slides als PDF auf GitHub