

rstatsZH - Data Science mit R

Mit mehreren Dataframes arbeiten / Funktionen schreiben

Lars Schöbitz

2021-10-12

Rückblick - Woche 6

- Erweiterte Vektoren
 - Faktoren
 - Datumswerte
 - Tibbles
- Daten importieren
 - `read_csv()`
 - `read_excel()`
- Tidy Data Konzept
- Daten drehen (pivoting) mit `{tidyr}`
 - `pivot_longer()`
 - `pivot_wider()`

Hausaufgabe 6



Hausaufgabe 6 - Datenschutz

1. Unterordner für Daten erstellen
2. Daten in diesem Ordner speichern

Hausaufgabe 6 - Datenschutz

1. `.gitignore` Datei öffnen
2. Neue Zeile hinzufügen mit Text `/data`

Hausaufgabe 6 - Lösungen

- **GitHub Organisation:** rstatsZH
 - <https://github.com/rstatsZH/>
- **Repo:** ha-06-treibhausgase
 - <https://github.com/rstatsZH/ha-06-treibhausgase>
- **R Markdown Datei:** ha-06-solutions.Rmd
 - <https://github.com/rstatsZH/ha-06-treibhausgase/blob/main/ha-06-solutions.Rmd>

Ziele für diese Woche

Am Ende dieser Woche könnt ihr:

- Mehrere Dataframes zusammenfügen
- Tabellen mit verschiedenen Packages darstellen
- **Bonusmaterial:** Eine eigene Funktion für ein `{ggplot2}` theme schreiben
- Mit den im Kurs erlernten Fähigkeiten selbstständig weiter arbeiten

Mit mehreren Dataframes arbeiten

Wir...

haben mehrere Dataframes

wollen diese zusammenbringen

Data: Women in science

Informationen zu 10 Frauen in der Wissenschaft welche die Welt verändert haben

name
Ada Lovelace
Marie Curie
Janaki Ammal
Chien-Shiung Wu
Katherine Johnson
Rosalind Franklin
Vera Rubin
Gladys West
Flossie Wong-Staal
Jennifer Doudna

Inputs - Drei Dataframes

professions	dates	works
-------------	-------	-------

```
# A tibble: 10 × 2
  name                profession
  <chr>               <chr>
1 Ada Lovelace        Mathematician
2 Marie Curie         Physicist and Chemist
3 Janaki Ammal        Botanist
4 Chien-Shiung Wu     Physicist
5 Katherine Johnson   Mathematician
6 Rosalind Franklin   Chemist
7 Vera Rubin          Astronomer
8 Gladys West         Mathematician
9 Flossie Wong-Staal  Virologist and Molecular Biologist
10 Jennifer Doudna     Biochemist
```

Gewünschter Output

```
# A tibble: 10 × 5
  name           profession birth_year death_year known_for
<chr>          <chr>          <dbl>    <dbl> <chr>
1 Ada Lovelace   Mathematic...      NA        NA first co...
2 Marie Curie    Physicist ...      NA        NA theory o...
3 Janaki Ammal   Botanist         1897       1984 hybrid s...
4 Chien-Shiung Wu Physicist        1912       1997 confirm a...
5 Katherine Johnson Mathematic...    1918       2020 calculat...
6 Rosalind Franklin Chemist         1920       1958 <NA>
7 Vera Rubin     Astronomer       1928       2016 existenc...
8 Gladys West    Mathematic...    1930        NA mathemat...
9 Flossie Wong-Staal Virologist...    1947        NA first sc...
10 Jennifer Doudna Biochemist       1964        NA one of t...
```

Inputs als Erinnerung

```
names(professions)
```

```
[1] "name"      "profession"
```

```
names(dates)
```

```
[1] "name"      "birth_year" "death_y
```

```
names(works)
```

```
[1] "name"      "known_for"
```

```
nrow(professions)
```

```
[1] 10
```

```
nrow(dates)
```

```
[1] 8
```

```
nrow(works)
```

```
[1] 9
```

Dataframes zusammenfügen

Dataframes zusammenfügen

```
abcd_join(x, y)
```

- `left_join()`: alle Reihen aus x
- `right_join()`: alle Reihen aus y
- `full_join()`: alle Reihen aus x und y
- ...

Beispiel

Für die nächsten Folien

```
# A tibble: 3 × 2
  id var_x
<dbl> <chr>
1     1 x1
2     2 x2
3     3 x3
```

```
# A tibble: 3 × 2
  id var_y
<dbl> <chr>
1     1 y1
2     2 y2
3     4 y4
```


left_join()

left_join(x, y)

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

```
left_join(tib_x, tib_y)
```

```
# A tibble: 3 × 3
  id var_x var_y
<dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
3     3 x3    <NA>
```

left_join()

```
professions %>%  
  left_join(dates)
```

```
# A tibble: 10 × 4
```

	name <chr>	profession <chr>	birth_year <dbl>	death_year <dbl>
1	Ada Lovelace	Mathematician	NA	NA
2	Marie Curie	Physicist and Chemist	NA	NA
3	Janaki Ammal	Botanist	1897	1984
4	Chien-Shiung Wu	Physicist	1912	1997
5	Katherine Johnson	Mathematician	1918	2020
6	Rosalind Franklin	Chemist	1920	1958
7	Vera Rubin	Astronomer	1928	2016
8	Gladys West	Mathematician	1930	NA
9	Flossie Wong-Staal	Virologist and Molec...	1947	NA
10	Jennifer Doudna	Biochemist	1964	NA

right_join()

right_join(x, y)

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

```
right_join(tib_x, tib_y)
```

```
# A tibble: 3 × 3  
  id var_x var_y  
<dbl> <chr> <chr>  
1     1 x1     y1  
2     2 x2     y2  
3     4 <NA>   y4
```

right_join()

```
professions %>%  
  right_join(dates)
```

```
# A tibble: 8 × 4  
  name                profession    birth_year death_year  
  <chr>              <chr>          <dbl>      <dbl>  
1 Janaki Ammal       Botanist       1897       1984  
2 Chien-Shiung Wu    Physicist      1912       1997  
3 Katherine Johnson  Mathematician  1918       2020  
4 Rosalind Franklin  Chemist       1920       1958  
5 Vera Rubin         Astronomer    1928       2016  
6 Gladys West        Mathematician  1930        NA  
7 Flossie Wong-Staal Virologist and Molecu... 1947        NA  
8 Jennifer Doudna    Biochemist    1964        NA
```

full_join()

full_join(x, y)

1	x1	1	y1
2	x2	2	y2
3	x3	4	y4

full_join(tib_x, tib_y)

```
# A tibble: 4 × 3
  id var_x var_y
<dbl> <chr> <chr>
1     1 x1    y1
2     2 x2    y2
3     3 x3    <NA>
4     4 <NA> y4
```

full_join()

```
dates %>%  
  full_join(works)
```

```
# A tibble: 10 × 4  
  name          birth_year death_year known_for  
  <chr>         <dbl>     <dbl> <chr>  
1 Janaki Ammal  1897      1984 hybrid species, biod..  
2 Chien-Shiung Wu  1912      1997 confirm and refine th..  
3 Katherine Johnson  1918      2020 calculations of orbi..  
4 Rosalind Franklin  1920      1958 <NA>  
5 Vera Rubin    1928      2016 existence of dark ma..  
6 Gladys West   1930      NA  mathematical modelin..  
7 Flossie Wong-Staal  1947      NA  first scientist to c..  
8 Jennifer Doudna  1964      NA  one of the primary d..  
9 Ada Lovelace   NA        NA  first computer algor..  
10 Marie Curie   NA        NA  theory of radioactiv..
```

Alles in einer Code Sequenz

```
professions %>%  
  left_join(dates) %>%  
  left_join(works)
```

```
# A tibble: 10 × 5  
  name                profession birth_year death_year known_for  
  <chr>              <chr>          <dbl>      <dbl> <chr>  
1 Ada Lovelace       Mathematic...    NA         NA first co...  
2 Marie Curie        Physicist ...    NA         NA theory o...  
3 Janaki Ammal       Botanist        1897       1984 hybrid s...  
4 Chien-Shiung Wu    Physicist       1912       1997 confirm a...  
5 Katherine Johnson  Mathematic...   1918       2020 calculat...  
6 Rosalind Franklin  Chemist        1920       1958 <NA>  
7 Vera Rubin        Astronomer     1928       2016 existenc...  
8 Gladys West        Mathematic...   1930         NA mathemat...  
9 Flossie Wong-Staal Virologist...   1947         NA first sc...  
10 Jennifer Doudna    Biochemist     1964         NA one of t...
```

Praktikum 10 - Daten zusammenfügen

2er Teams

1. **E-Mail:** Öffne deine Email und klicke auf den Link zu deinem persönlichen GitHub repo
2. **GitHub:** Klicke auf den grünen Button "Code" und kopiere den Link für das Repo in deine Zwischenablage
3. **RStudio Cloud:** Öffne deinen Arbeitsbereich für den Kurs in der RStudio Cloud
4. **RStudio Cloud / Projects:** Klicke auf "New Project from GitHub Repository"

Inputs - Drei Dataframes

einkaeufe

preise

kundenprofile

```
# A tibble: 9 × 4
  kunden_id produkt_name einkauf einheit
  <chr>      <chr>          <dbl> <chr>
1 k1        Chips            2  anzahl
2 k1        Milch            3  anzahl
3 k1        Avocado            1  anzahl
4 k2        Pfirsich           2.5 kg
5 k2        Birne            0.5 kg
6 k2        Apfel            2   kg
7 k2        Tomate            1.5 kg
8 k2        Pfirsich            1   kg
9 k2        Milch            4   anzahl
```

Gewünschter Output

vorname	nachname	summe	email
Edwin	Dumont	9.2	edwin.dumont@example.com
Leonora	Garcia	24.6	leonora.garcia@example.com

Schritt 1 - Daten zusammenfügen

```
einkaeufe_preise <- einkaeufe %>%  
  left_join(preise)
```

```
einkaeufe_preise
```

```
# A tibble: 9 × 5  
  kunden_id produkt_name einkauf einheit preis  
  <chr>      <chr>         <dbl> <chr>   <dbl>  
1 k1        Chips           2  anzahl  3.8  
2 k1        Milch           3  anzahl  2.2  
3 k1        Avocado          1  anzahl  3.2  
4 k2        Pfirsich        2.5 kg      6.5  
5 k2        Birne           0.5 kg      2.6  
6 k2        Apfel           2  kg      4.1  
7 k2        Tomate          1.5 kg      2.7  
8 k2        Pfirsich         1  kg      6.5  
9 k2        Milch           4  anzahl  2.2
```

Schritt 2 - Neue Variable erstellen

```
einkaeufe_kosten <- einkaeufe_preise %>%  
  mutate(kosten = einkauf * preis)
```

```
einkaeufe_kosten
```

```
# A tibble: 9 × 6  
  kunden_id produkt_name einkauf einheit preis kosten  
  <chr>      <chr>          <dbl> <chr>    <dbl> <dbl>  
1 k1        Chips            2  anzahl  3.8    7.6  
2 k1        Milch             3  anzahl  2.2    6.6  
3 k1        Avocado            1  anzahl  3.2    3.2  
4 k2        Pfirsich          2.5  kg      6.5   16.2  
5 k2        Birne             0.5  kg      2.6    1.3  
6 k2        Apfel             2    kg      4.1    8.2  
7 k2        Tomate            1.5  kg      2.7    4.05  
8 k2        Pfirsich           1    kg      6.5    6.5  
9 k2        Milch             4    anzahl  2.2    8.8
```

Schritt 3 - Daten zusammenfassen

```
einkaeufe_kosten_sum <- einkaeufe_kosten %>%  
  group_by(kunden_id) %>%  
  summarise(  
    summe = sum(kosten)  
  )  
  
einkaeufe_kosten_sum
```

```
# A tibble: 2 × 2  
  kunden_id summe  
  <chr>      <dbl>  
1 k1         9.2  
2 k2        24.6
```

Schritt 4 - Daten zusammenfügen + eingrenzen

```
kunden_tab <- einkaeufe_kosten_sum %>%  
  left_join(kundenprofile) %>%  
  select(ends_with("name"), summe, email)
```

```
kunden_tab
```

```
# A tibble: 2 × 4  
  vorname nachname summe email  
  <chr>    <chr>    <dbl> <chr>  
1 Edwin   Dumont      9.2 edwin.dumont@example.com  
2 Leonora Garcia    24.6 leonora.garcia@example.com
```

Schritt 5 - Daten als Tabelle darstellen

```
kunden_tab %>%  
  gt()
```

vorname nachname summe email			
Edwin	Dumont	9.2	edwin.dumont@example.com
Leonora	Garcia	24.6	leonora.garcia@example.com

Als eine Code Sequenz

```
einkaeufe %>%  
  left_join(preise) %>%  
  mutate(kosten = einkauf * preis) %>%  
  group_by(kunden_id) %>%  
  summarise(  
    summe = sum(kosten)  
  ) %>%  
  left_join(kundenprofile) %>%  
  select(ends_with("name"), summe, email) %>%  
  gt()
```

vorname nachname		summe	email
Edwin	Dumont	17.4	edwin.dumont@example.com
Leonora	Garcia	45.1	leonora.garcia@example.com

Funktionen

Funktionen - Bonusmaterial

Praktikum 11 - Funktionen

Live Coding

1. **E-Mail:** Öffne deine Email und klicke auf den Link zu deinem persönlichen GitHub repo
2. **GitHub:** Klicke auf den grünen Button "Code" und kopiere den Link für das Repo in deine Zwischenablage
3. **RStudio Cloud:** Öffne deinen Arbeitsbereich für den Kurs in der RStudio Cloud
4. **RStudio Cloud / Projects:** Klicke auf "New Project from GitHub Repository"

The End

The End - Noch nicht ganz

Was habt ihr gelernt?

- Anwendung von Tidyverse Packages zum
 - Importieren,
 - Aufräumen (Tidying),
 - Transformieren,
 - Visualisieren, und
 - Kommunizieren von Daten.
- Kollaboration und Versionsverwaltung mit Git/GitHub
- Datenprojekte reproduzierbar publizieren mit GitHub
- Das Konzept von Tidy Data

Wie geht's weiter?

Raus aus der RStudio Cloud

1. Installationen: <https://github.com/rstatsZH/kochbuch/tree/main/01-Installation>
2. Einmalig: Tidyverse Packages installieren
3. Danach: Tidyverse Packages laden
4. Packages ausserhalb des Tidyverse installieren und laden (e.g. `janitor`)

```
# Einmalig in Konsole ausführen  
install.packages("tidyverse")
```

```
# In jedem Skript  
library(tidyverse)
```


Weiterführende Ressourcen - Üben + Vertiefen

<https://rstatszh.github.io/website/posts/2021-04-30-woche07/>

Projektarbeit - Unterstützung bis Mitte August

Hausaufgabe 6

1. GitHub Repository erstellen und RStudio Projekt aufgleisen (Hausaufgabe 6)
2. Daten für das Projekt identifizieren

Wie es weiter geht: Bericht mit R Markdown schreiben

1. Daten importieren
2. Daten (visuell) erkunden
3. Daten ggf. transformieren und dann erneut (visuell) erkunden
4. Fragen an den Datensatz formulieren
5. Versuchen zu Antworten zu kommen und dokumentieren
6. Immer wieder, git add, commit, push

Reflexion

Reflexion

5 min Nachdenken + Notizen

1. Was sind die drei nützlichsten Dinge die du gelernt hast?
2. Welches Thema war besonders schwer zu folgen?
3. Was hat dir gefehlt?

Magst du mir ein Kommentar hinterlassen?

Long answer text

Feedback

Ziele erreicht?

Bitte ausfüllen: kutt.it/rstatszh-eval

Photo by: Virgil Cayasa



Wie es für mich weiter geht

1. **Beratung:** Projektbezogener Support, Code Review, Coaching
2. **rstatsZH Kursleitung:** Info über den Kurs verbreiten
3. **Kurse zu vertiefenden Themen:** Entwicklung von 4-Stunden Workshops

Contact: Lars@Lse.de



Für die Aufmerksamkeit!

Für die R packages `{xaringan}` und `{xaringanthemer}` mit welchen die Folien geschrieben wurden.

Eine PDF Version der Folien kann hier heruntergeladen werden:

https://github.com/rstatsZH/website/raw/master/slides/e1_d07-data-join/e1_d07-data-join.pdf

Für [Data Science in a Box](#) und [Remaster the Tidyverse](#), von welchen ich Materialien für diesen Kurs nutze und welche genau wie diese Folien mit [Creative Commons Attribution Share Alike 4.0 International](#) lizenziert sind.