

EINFÜHRUNG IN R - SCHLEIFEN UND FUNKTIONEN

Jan-Philipp Kolb

13 Juni, 2019

MOTIVATION

- Im Prinzip sollte man immer versuchen Aufgaben zu vektorisieren (`apply` Befehlsfamilie).
- Allerdings können Schleifen auch sehr nützlich sein.
- Um dies zu zeigen, möchte ich etwas länger ausholen. . .

ARBEITSSCHRITTE

- Beispieldaten laden
- Geokodierung durchführen.
- Schleifen schreiben und anwenden.

```
load("../data/bauenwohnen_teil.RData")
```

```
head(dat$Stadtteil)
```

```
## [1] "Altstadt"          "Innenstadt"        "Bahnhofsviertel"  "We  
## [5] "Westend-Nord"      "Nordend-West"
```

EINE SPALTE HINZUFÜGEN

```
(dat$stadtteil_1 <- paste("Frankfurt",dat$Stadtteil))
```

```
## [1] "Frankfurt Altstadt" "Frankfurt Innenstadt"
## [3] "Frankfurt Bahnhofsviertel" "Frankfurt Westend-Süd"
## [5] "Frankfurt Westend-Nord" "Frankfurt Nordend-West"
## [7] "Frankfurt Nordend-Ost" "Frankfurt Ostend"
## [9] "Frankfurt Bornheim" "Frankfurt Gutleutviertel"
## [11] "Frankfurt Gallus" "Frankfurt Bockenheim"
## [13] "Frankfurt Sachsenhausen-Nord" "Frankfurt Sachsenhausen-"
## [15] "Frankfurt Oberrad" "Frankfurt Niederrad"
## [17] "Frankfurt Schwanheim" "Frankfurt Griesheim"
## [19] "Frankfurt Rödelheim" "Frankfurt Hausen"
## [21] "Frankfurt Praunheim" "Frankfurt Heddernheim"
## [23] "Frankfurt Niederursel" "Frankfurt Ginnheim"
## [25] "Frankfurt Dornbusch" "Frankfurt Eschersheim"
## [27] "Frankfurt Eckenheim" "Frankfurt Preungesheim"
## [29] "Frankfurt Bonames" "Frankfurt Berkersheim"
## [31] "Frankfurt Riederwald" "Frankfurt Seckbach"
## [33] "Frankfurt Fechenheim" "Frankfurt Höchst"
## [35] "Frankfurt Nied" "Frankfurt Sindlingen"
```

```
library(tmaptools)
```

```
(gc1 <- geocode_OSM("Frankfurt Altstadt"))
```

```
$`query`
```

```
[1] "Frankfurt Altstadt"
```

```
$coords
```

	x	y
	8.682936	50.111045

```
$bbox
```

	xmin	ymin	xmax	ymax
	8.674523	50.106129	8.688171	50.114872

```
gc2 <- geocode_OSM("Frankfurt Altstadt", details = T)
```

```
names(gc2)
```

```
## [1] "query"          "coords"         "bbox"           "place_id"
## [5] "osm_type"       "osm_id"         "place_rank"     "display_name"
## [9] "class"          "type"           "importance"     "icon"
```

Schleifen in R

- Der Befehl `for()` kennzeichnet den Start einer Schleife
- in Klammern, haben wir einen Index und die Anzahl der Läufe (in diesem Fall läuft die Schleife von 1 bis 4).
- in den geschweiften Klammern `{}` ist angegeben, was bei einer Iteration passiert.

```
for (i in 1:4){  
  cat(i, "\n")  
}
```

```
## 1
```

```
## 2
```

```
## 3
```

```
## 4
```

SCHLEIFEN - DIE ERGEBNISSE BEHALTEN

- Wir können die Ergebnisse in einem Objekt speichern
- dieses kann bspw. ein Vektor oder eine Liste sein.

```
erg <- list()

for (i in 1:ncol(dat)){
  erg[[i]] <- summary(dat)
  cat(i, "\n")
}
```


EINE SCHLEIFE UM DIE GEOKODIERTEN ERGEBNISSE ABZUGREIFEN

- Die Geokodierung wird durchgeführt und das Ergebnis an der entsprechenden Stelle in der Liste `erg` gespeichert.

```
erg <- list()
for(i in 1:nrow(dat)){
  erg[[i]] <- geocode_OSM(dat$stadtteil_1[i])
}
```

```
## [1] "erg"
```

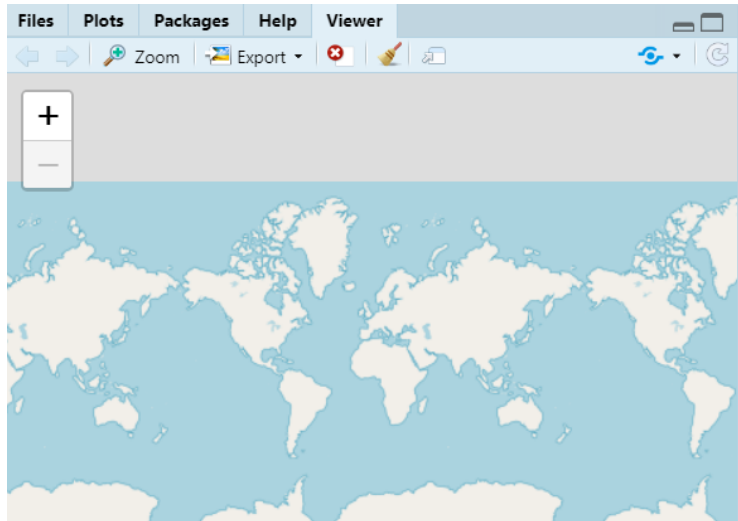
```
erg2 <- lapply(erg,function(x)x$coords)
df_gc1 <- do.call(rbind,erg2)
df_gc <- data.frame(dat$Stadtteil,df_gc1)
```

```
head(df_gc)
```

```
##      dat.Stadtteil      x      y
## 1      Altstadt 8.682936 50.11105
## 2      Innenstadt 8.674922 50.11288
## 3 Bahnhofsviertel 8.668736 50.10774
## 4      Westend-Süd 8.662270 50.11524
## 5      Westend-Nord 8.667921 50.12636
## 6      Nordend-West 8.677950 50.12491
```

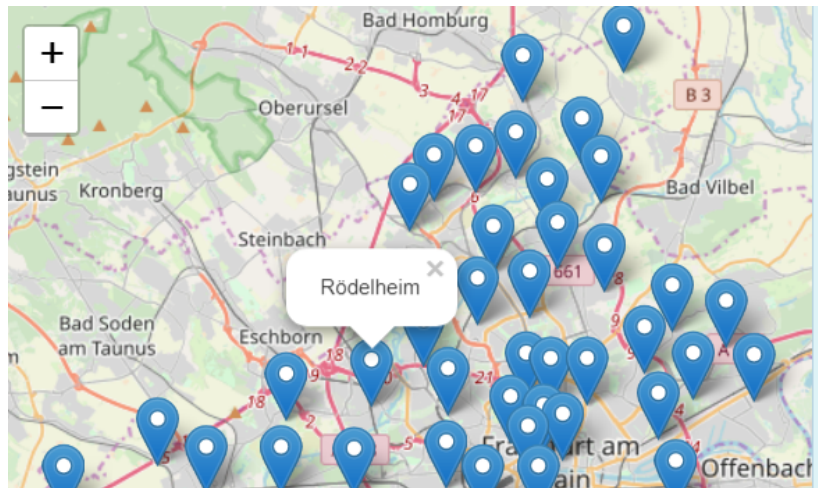
EINE INTERAKTIVE KARTE

```
library(leaflet)
library(magrittr)
leaflet() %>% addTiles()
```



UNSERE PUNKTE AUF DER KARTE

```
leaflet() %>%  
  addTiles() %>%  
  addMarkers(lng=df_gc[,2], lat=df_gc[,3],  
  popup=df_gc[,1])
```



ALLE R-DATEIEN IN EINEM ORDNER

```
dirn <- dir("../data/")  
head(dirn)
```

```
## [1] "bauenwohnen.csv"          "bauenwohnen.xls"  
## [3] "bauenwohnen_teil.csv"     "bauenwohnen_teil.RData"  
## [5] "dat_sn.RData"             "datt_gc_ffm.txt"
```

```
grep(".RData",dirn)
```

```
## [1]  4  5  7  8  9 11 12 13
```

```
head(rdat <- grep(".RData",dirn,value = T))
```

```
## [1] "bauenwohnen_teil.RData"      "dat_sn.RData"  
## [3] "ffm_gc2.RData"              "gc_list_stadtteile_ffm.RD  
## [5] "gp_small.RData"             "list_gc.RData"
```

DIESE DATEIEN EINLESEN

```
for (i in 1:length(rdat)){  
  load(rdat[i])  
  cat(rdat[i],"\n")  
}
```

MOTIVATION

- Funktionen können sehr nützlich sein, bspw. um sich eine Menge Schreibarbeit zu ersparen. - Später kann man sich ganz individuelle Lösungen bauen

ARBEITSSCHRITTE

Im GESIS Panel sind fehlende Werte mit Minuswerten kodierte. Diese sollten durch das Systemmissing von R ersetzt werden um korrekte Berechnungen (bspw. Mittelwert) zu ermöglichen.

- Beispieldaten laden
- Fehlende Werte ersetzen
- Funktion schreiben und anwenden

- Wahrscheinlichkeitsbasiertes Access Panel für Individuen: - Allgemeine Bevölkerung in Deutschland, Deutschsprachige Bevölkerung, 18-70 Jahre
- Panelisten wurden aus den Melderegistern rekrutiert - (270 Sampling Points) 7599 face-to-face Interviews (CAPI)
- Ungefähr 5000 Panelisten (Basis Stichprobe / erste Kohorte 2014)



A Probability-Based Mixed-Mode Access Panel
for the Social Sciences

Search

[GESIS Panel Home](#) ▾ [Submission](#) [Data](#) ▾ [Documentation](#) ▾ [FAQ](#)

You are here: [GESIS Panel](#) ▸ [Data](#) ▸ [GESIS Panel Campus File](#)

GESIS Panel Campus File

The [GESIS Panel Campus File](#) is intended for teaching purposes only. It provides interested parties (e.g., students or lecturers) with an opportunity to work with an easily accessible, high quality panel dataset that should satisfy many requirements set forth in the interested parties' curriculum. In exchange for easy accessibility, the GESIS Panel Campus File contains only selected portions of the original GESIS Panel scientific use file. In order to ensure the anonymity of our panelists, the GESIS Panel Campus File contains only a **random 25% sample** of all panel members who were still active at the start of the respective year. For the current Campus File, the final sample size is N=1222. An exact documentation of studies and variables included in this year's GESIS Panel Campus File can be found in the [data description](#) and [codebook](#).

Access to the GESIS Panel Campus File (ZA5666 / doi:10.4232/1.12749) can be acquired at the [GESIS Data Archive](#).

Researchers interested in using GESIS Panel data for scientific publication purposes should use the full dataset (either the GESIS Panel [Standard Edition](#) or the GESIS Panel [Extended Edition](#)). Data users are strongly advised not to use the GESIS Panel Campus File for scientific publications other than for teaching purposes.

- Übersichtsseite: **GESIS Panel Campus File**
- Registrierung notwendig

LINKS FÜR DEN DOWNLOAD:

- **Download .csv**
- **Download .sav**
- **Download **14.dta**



ZA5666_v1-0-0.csv



ZA5666_v1-0-0.sav



ZA5666_v1-0-0_Stata14.dta

```
load("../data/gp_small.RData")  
head(dats)
```

##	a11d056z	a11d094a	bczd001a	a11c019a	bdao067a	a11d096b	a11c
## 9	10	2	3	2	97	11	
## 14	11	-88	1	1	97	11	
## 56	7	1	4	1	97	2	
## 41	2	2	4	1	97	2	
## 61	7	-88	3	3	2	5	
## 54	5	1	1	3	97	13	

DIE MISSING CODES IM GESIS PANEL

```
kable(df,row.names = F)
```

	Value	Value.label	Remark
-11	Not invited		only in recruitment waves - when profile sur
-22	Not in panel		not willing to join the panel after recruitme
-33	Unit nonresponse		invited but not participating in correspondin
-44	Missing by m.o.p.		mode of participation (m.o.p.): online or of
-55	Missing by technical error		e.g. questionnaire programming error
-66	Missing by design		experimental variation
-77	Not reached		only in online mode: panelist has not seen t
-88	Missing by filter		filtered item
-99	Item nonresponse		due to nonresponse by the respondent
-111	Ambiguous answer		ambiguous answers in questionnaire

```
df %>% regulartable() %>% autofit() %>%  
width(j=~Value,width=1) %>% width(j=~Value.label,width=1)%>% wid
```

FEHLENDE WERTE ERSETZEN

```
mean(dats$a11d056z)
```

```
## [1] 6.011
```

```
dats$a11d056z[dats$a11d056z== -99] <- NA  
mean(dats$a11d056z)
```

```
## [1] NA
```

```
mean(dats$a11d056z, na.rm=T)
```

```
## [1] 7.502028
```

EINE FUNKTION UM DIE FEHLENDEN WERTE ZU ERSETZEN

```
code_miss <- function(var){  
  var[var=="-99"] <- NA  
  return(var)  
}
```

- Laden Sie den synthetischen Datensatz des GESIS Panels von Github herunter und importieren Sie ihn in R.
- Erweitern Sie die Funktion auf der vorhergehenden Folie so, dass auch die anderen fehlenden Werte durch NA ersetzt werden.
- Wenden Sie die Funktion auf den ganzen Datensatz an und ersetzen Sie alle fehlenden Werte mit NA