2. Übungsblatt (Paneldaten)

Daria Tisch

1 Organisation

1.1 Arbeitsverzeichnis festsetzen

```
# set working directory
setwd("D:/Seafile/main/teaching/2024_wuppertal/studis/uebungen")
```

1.2 Packages installieren und laden

```
# Packages
pkgs <- c(
  "tidyverse",
  "sjlabelled", # for variable labels
  "plm", # Panelregressionen
  "sjPlot" # um Regressionstabellen schön darzustellen
)

## Install uninstalled packages
lapply(pkgs[!(pkgs %in% installed.packages())], install.packages)

## Load all packages to library
lapply(pkgs, library, character.only = TRUE)
```

1.3 Daten einlesen

Wir arbeiten wieder mit dem Übungsdatensatz des SOEP (DOI:10.5684/soep.practice.v36). Es werden zwei Datensätze eingelesen. Der erste Datensatz enthalt die tatsächlichen Umfragedaten, der zweite Datensatz enthält die Variablenlabels.

```r
# read data
df = read.csv("../daten/soep_uebung.csv")
df_labels = read.csv("../daten/soep_labels.csv")

# Add variable labels
for (i in 1:nrow(df_labels)) {
  variable_name <- df_labels$variable[i]
  variable_label <- df_labels$variable_label[i]

  # Apply the label to the corresponding column in df
  df[[variable_name]] <- set_label(df[[variable_name]], variable_label)
}
# show variable labels
get_label(df)
```

```
                                    id                                 syear
"Personennummer (zufällig generiert)"                    "Erhebungsjahr"
                                   sex                                alter
                          "Geschlecht"          "Alter der Befragungsperson"
                              anz_pers                            anz_kind
         "Anzahl Personen im Haushalt"           "Anzahl Kinder im Haushalt"
                               bildung                              erwerb
            "Anzahl an Bildungsjahren"                    "Erwerbsstatus"
                               branche                         gesund_org
             "Branche aktueller Beruf"                  "subj. Gesundheit"
                           lebensz_org                           einkommenj1
            "Ggw. Lebenszufriedenheit"     "Bruttoeinkommen/Jahr Hauptberuf"
                           einkommenj2                           einkommenm1
     "Bruttoeinkommen/Jahr Nebenberuf"    "Bruttoeinkommen/Monat Hauptberuf"
                           einkommenm2
    "Bruttoeinkommen/Monat Nebenberuf"
```

2 Datenaufbereitung

In dieser Übung möchten wir herausfinden, welche Faktoren mit Lebenszufriedenheit (lebensz_org) zusammenhängen. Denkbar wären zum Beispiel: Geschlecht (sex), Alter(alter), Anzahl an Kindern im Haushalt (anz_kind), Bildung (bildung) und Einkommen (einkommenj1).

Um die Analysen durchzuführen, müssen wir die Variablen erst einmal aufbereiten. Überprüfe bei jeder Variable,

- ob fehlende Werte als solche markiert sind
- um welchen Typ es sich bei den Variablen hanelt und ob etwa eine Umwandlung erforderlich ist.

2.1 Lebenszufriedenheit

```
table(df$lebensz_org, useNA = "always")
```

```
                            [0] ganz und gar unzufrieden
                         611                           62
 [10] ganz und gar zufrieden                            1
                        1350                           83
                           2                            3
                         202                          418
                           4                            5
                         574                         1837
                           6                            7
                        1868                         4645
                           8                            9
                        7963                         3909
                        <NA>
                           0
```

```
df = df %>%
  mutate(lz = as.numeric(sub("\\[([0-9]+)\\].*", "\\1", df$lebensz_org)))

table(df$lz, useNA = "always")
```

```
   0    1    2    3    4    5    6    7    8    9   10 <NA>
  62   83  202  418  574 1837 1868 4645 7963 3909 1350  611
```

2.2 Geschlecht

```
table(df$sex, useNA = "always")
```

```
[0] männlich [1] weiblich         <NA>
       10762        12760            0
```

```
df = df %>%
  mutate(female = as.numeric(sub("\\[([0-9]+)\\].*", "\\1", df$sex)))

table(df$female, useNA = "always")
```

```
    0     1  <NA>
10762 12760     0
```

2.3 Alter

```
table(df$alter, useNA = "always")
```

```
  17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
 472  470  437  392  347  290  281  263  244  254  283  282  293  304  337  341
  33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
 331  368  387  387  379  397  413  430  447  450  454  434  450  466  466  473
  49   50   51   52   53   54   55   56   57   58   59   60   61   62   63   64
 500  517  526  518  479  447  412  381  381  359  339  336  326  296  298  312
  65   66   67   68   69   70   71   72   73   74   75   76   77   78   79   80
 325  314  309  294  272  239  242  244  246  256  276  271  270  247  223  192
  81   82   83   84   85   86   87   88   89   90   91   92   93   94   95   96
 156  124  105   82   74   64   47   46   36   28   19   18   13   11    8    5
  97   98   99  100  101  102 <NA>
   5    4    3    3    1    1    0
```

```
class(df$alter)
```

```
[1] "integer"
```

2.4 Anzahl an Kindern im Haushalt

```r
table(df$anz_kind, useNA = "always")
```

```
    0     1     2     3     4     5     6     7     8     9    10  <NA>
14108  3874  3557  1360   367   103    52    15     7     1     1    77
```

```r
class(df$anz_kind)
```

```
[1] "integer"
```

```r
sum(is.na(df$anz_kind))
```

```
[1] 77
```

2.5 Bildung

```r
table(df$bildung, useNA = "always")
```

```
   7  8.5    9   10 10.5   11 11.5   12   13 13.5   14 14.5   15   16   17   18
 325   52 2296  839 4044  812 3285 2829 1331  334  556  526 1015  759  113 2795
<NA>
1611
```

```r
class(df$bildung)
```

```
[1] "numeric"
```

```r
sum(is.na(df$bildung))
```

```
[1] 1611
```

2.6 Einkommen

```
summary(df$einkommenj1)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0       0    5786   16776   28525  269425
```

```
class(df$einkommenj1)
```

```
[1] "numeric"
```

```
sum(is.na(df$einkommenj1))
```

```
[1] 0
```

2.7 Analysesample bilden

(a) Selektiere nur die Variablen, die wir brauchen

```
df = df %>%
  select(id,syear, lz,female, alter, anz_kind, bildung, einkommenj1)
```

(b) Nun bilden wir das Analysesample. Wir wenden "listwise deletion".

```
df <- df[complete.cases(df[, c("lz","female", "alter", "anz_kind", "bildung", "einkommenj1")),]
```

(c) Wie viele Beobachtungen gehen uns verloren?

```
# Initialize a data frame to store the results
results <- data.frame(
  Variable = character(),
  Cases_Remaining = integer(),
  Cases_Lost = integer(),
  stringsAsFactors = FALSE
)

# Total number of rows initially
initial_rows <- nrow(df)

# Iterate over each variable
```

```r
for (var in names(df)) {
  # Perform listwise deletion based on the current variable
  remaining_rows <- sum(complete.cases(df[, var, drop = FALSE]))

  # Calculate cases lost
  cases_lost <- initial_rows - remaining_rows

  # Append to the results data frame
  results <- rbind(
    results,
    data.frame(
      Variable = var,
      Cases_Remaining = remaining_rows,
      Cases_Lost = cases_lost
    )
  )
}

results
```

```
     Variable Cases_Remaining Cases_Lost
1          id           21642          0
2       syear           21642          0
3          lz           21642          0
4      female           21642          0
5       alter           21642          0
6    anz_kind           21642          0
7     bildung           21642          0
8 einkommenj1           21642          0
```

3 Datenexploration

3.1 Wieviele Personenjahre sind im Datensatz?

```r
nrow(df)
```

```
[1] 21642
```

3.2 Wie viele Personen sind im Datensatz?

```
length(unique(df$id))
```

```
[1] 5788
```

3.3 Wieviele Personen nehmen pro Jahr teil?

```
table(df$syear)
```

```
2015 2016 2017 2018 2019
5122 4572 4341 3992 3615
```

Wir haben also ein sogenanntes “unbalanced panel”.

3.4 Was ist das Durchschnittsalter im Jahr 2015

```
# Berechnung des Durchschnittsalters für das Jahr 2015
mean(df$alter[df$syear == 2015], na.rm = TRUE)
```

```
[1] 48.99863
```

```
# Alternative mit dplyr
durchschnittsalter <- df %>%
  filter(syear == 2015) %>%       # Filtert die Daten für das Jahr 2015
  summarise(mean_alter = mean(alter, na.rm = TRUE)) %>%  # Berechnet den Durchschnitt
  pull(mean_alter)               # Extrahiert den berechneten Wert

cat("Das Durchschnittsalter im Jahr 2015 beträgt:", round(durchschnittsalter,2), "Jahre.")
```

```
Das Durchschnittsalter im Jahr 2015 beträgt: 49 Jahre.
```

3.5 Welche Variablen sind zeitkonstant und welche zeitveränderlich?

```r
# Analyse: Zeitkonstante und zeitveränderliche Variablen

variablen_analyse <- df %>%
  mutate(id2 = id) %>%
  group_by(id) %>%                          # Gruppiert die Daten nach ID
  summarise_all(~ n_distinct(.)) %>%        # Zählt die Anzahl der einzigartigen Werte je ID
  summarise(across(everything(), max))      # Nimmt das Maximum der einzigartigen Werte je Vari


# Ergebnis interpretieren
zeitkonstant <- names(variablen_analyse)[variablen_analyse == 1]  # Variablen mit nur einem W
zeitveraenderlich <- names(variablen_analyse)[variablen_analyse > 1]  # Variablen mit mehrere

# Ausgabe
cat("Zeitkonstante Variablen:\n",zeitkonstant )
```

```
Zeitkonstante Variablen:
 female id2
```

```r
cat("\nZeitveränderliche Variablen:\n", zeitveraenderlich)
```

```
Zeitveränderliche Variablen:
 id syear lz alter anz_kind bildung einkommenj1
```

3.6 Wie viele Personen nahmen an allen fünf Wellen teil? Und wie viele Jahre nahmen Personen durchschnittlich teil?

```r
# Anteil der Personen, die an allen fünf Wellen teilgenommen haben
data_summary <- df %>%
  group_by(id) %>%
  summarize(years_observed = n_distinct(syear))
table(data_summary$years_observed)  # Verteilung der Beobachtungsjahre
```

```
   1    2    3    4    5
 840  649  593  805 2901
```

```r
# Durchschnittliche Beobachtungsjahre pro Person
mean(data_summary$years_observed)
```

```
[1] 3.739115
```

4 Regressionen

4.1 Schätze ein POLS Modell

... und interepretiere die Koeffizienten.

```r
m_pol <- plm(lz ~ female + alter + anz_kind + bildung + einkommenj1, data = df,
             index = c("id", "syear"),
             effect = "individual", model = "pooling")
tab_model(m_pol,
          dv.labels = paste("POLS Model", sep = ""))
```

		POLS Model	
Predictors	Estimates	CI	p
(Intercept)	6.71	6.58 – 6.84	**<0.001**
female	0.00	-0.04 – 0.05	0.871
alter	-0.00	-0.00 – -0.00	**<0.001**
anz kind	0.05	0.03 – 0.07	**<0.001**
bildung	0.06	0.05 – 0.06	**<0.001**
einkommenj1	0.00	0.00 – 0.00	**<0.001**
Observations	21642		
R^2 / R^2 adjusted	0.022 / 0.022		

4.2 Schätze ein BE Modell

... und interepretiere die Koeffizienten.

```r
m_be <- plm(lz ~ female + alter + anz_kind + bildung + einkommenj1, data = df,
             index = c("id", "syear"),
             effect = "individual", model = "between")
tab_model(m_be,
          dv.labels = paste("BE Model", sep = ""))
```

		BE Model	
Predictors	Estimates	CI	p
(Intercept)	6.72	6.51 – 6.93	**<0.001**
female	0.06	-0.01 – 0.14	0.114
alter	-0.00	-0.01 – -0.00	**<0.001**
anz kind	0.05	0.02 – 0.09	**0.005**
bildung	0.06	0.04 – 0.07	**<0.001**
einkommenj1	0.00	0.00 – 0.00	**<0.001**
Observations	5788		
R^2 / R^2 adjusted	0.035 / 0.034		

4.3 Schätze ein FE Modell

... und interepretiere die Koeffizienten.

```
m_fe <- plm(lz ~ female + alter + anz_kind + bildung + einkommenj1 , data = df,
            index = c("id", "syear"),
            effect = "individual", model = "within")
# Retrieve the number of groups
num_groups <- length(unique(index(m_fe)[, "id"]))
tab_model(m_fe,
          dv.labels = paste("FE Model (", num_groups, " Groups)", sep = ""))
```

		FE Model (5788 Groups)	
Predictors	Estimates	CI	p
alter	0.01	-0.01 – 0.02	0.273
anz kind	0.02	-0.04 – 0.07	0.560
bildung	0.03	-0.03 – 0.09	0.312
einkommenj1	0.00	-0.00 – 0.00	0.574
Observations	21642		
R^2 / R^2 adjusted	0.000 / -0.365		

4.4 Schätze ein RE Modell

... und interepretiere die Koeffizienten.

```r
m_re <- plm(lz ~ female + alter + anz_kind + bildung + einkommenj1, data = df,
            index = c("id", "syear"),
            effect = "individual", model = "random")
# Retrieve the number of groups
num_groups <- length(unique(index(m_re)[, "id"]))
tab_model(m_re,
          dv.labels = paste("RE Model (", num_groups, " Groups)", sep = ""))
```

	RE Model (5788 Groups)		
Predictors	Estimates	CI	p
(Intercept)	6.74	6.54 – 6.94	**<0.001**
female	0.02	-0.05 – 0.10	0.540
alter	-0.00	-0.01 – -0.00	**<0.001**
anz kind	0.04	0.01 – 0.08	**0.004**
bildung	0.06	0.05 – 0.07	**<0.001**
einkommenj1	0.00	0.00 – 0.00	**<0.001**
Observations	21642		
R^2 / R^2 adjusted	0.164 / 0.164		

4.5 Vergleich

Vergleiche die vier Modelle.

```r
tab_model(m_pol, m_be, m_fe , m_re, show.ci = FALSE,
          dv.labels = c("POLS", "BE", "FE", "RE"))
```

	POLS		BE		FE		RE	
Predictors	Estimates	p	Estimates	p	Estimates	p	Estimates	p
(Intercept)	6.71	**<0.001**	6.72	**<0.001**			6.74	**<0.001**
female	0.00	0.871	0.06	0.114			0.02	0.540
alter	-0.00	**<0.001**	-0.00	**<0.001**	0.01	0.273	-0.00	**<0.001**
anz kind	0.05	**<0.001**	0.05	**0.005**	0.02	0.560	0.04	**0.004**
bildung	0.06	**<0.001**	0.06	**<0.001**	0.03	0.312	0.06	**<0.001**
einkommenj1	0.00	**<0.001**	0.00	**<0.001**	0.00	0.574	0.00	**<0.001**
Observations	21642		5788		21642		21642	
R^2 / R^2 adjusted	0.022 / 0.022		0.035 / 0.034		0.000 / -0.365		0.164 / 0.164	

a) Warum wird im FE kein Koeffizient für *female* geschätzt?

b) Warum ist die Anzahl an Beobachtungen im BE Modell kleiner als die Anzahl an Beobachtungen in den anderen Modellen?

4.6 Hausman Test

Sollen wir das RE oder das FE Modell nutzen? Führe einen Hausman Test durch. Wie entscheidest Du Dich?

```
phtest(m_fe, m_re)
```

```
    Hausman Test

data:  lz ~ female + alter + anz_kind + bildung + einkommenj1
chisq = 19.898, df = 4, p-value = 0.000523
alternative hypothesis: one model is inconsistent
```

4.7 Kausalanalyse

Wir sind am kausalen Effekt von Einkommen auf Lebenszufriedenheit interessiert. Auf welche Variablen sollten wir kontrollieren und auf welche nicht? Nenne Beispiele für mögliche confounder, collider und Variablen, die zu overcontrol führen. Begründe die Auswahl der Variablen jeweils in einem Satz. Es können durchaus Variablen genannt werden, die **nicht** im Datensatz enthalten sind.

5 Render

Wandle dieses Dokument in ein PDF und ein HTML Dokument um.

6 Weiterführende Literatur

- R for Data Science
- https://ruettenauer.github.io/Panel-Data-Analysis/