Replication of ‘Multilevel Models for the Analysis of Comparative Survey Data Common Problems and Some Solutions’ with R

Author: Valentin Velev (Goethe University Frankfurt)
Email: valentin.velev@gmx.de

30 January, 2023

Abstract

Aim: Replicate the analysis conducted by Prof. Dr. Alexander W. Schmidt-Catran (Goethe University Frankfurt), Prof. Dr. Malcolm Fairbrother (Umea University), and Prof. Dr. Hans-Jürgen Andreß (University of Cologne) that was published in a special issue on Cross-National Comparative Research in the German academic journal *Kölner Zeitschrift für Soziologie und Sozialpsychologie* in 2019. *Result:* Almost all calculations, tables and graphs from Schmidt-Catran et al. (2019) could be replicated sufficiently well in R.

Contents

1 Introduction

In this PDF document, which was automatically generated using R Markdown (and some LaTeXcode), I will replicate all calculations, tables, and plots found in Schmidt-Catran et al. (2019) with R. Their paper can be found here.

Note that the colorful sections denote the R code.

The aim of this project is to replicate the statistical analysis conducted in Schmidt-Catran et al. (2019) and thereby determine how closely an analysis conducted in Stata can be replicated in R (i.e., with R Markdown and LaTeX).

In short, the most important results are:

- Most tables and plots can be replicated well in R (i.e., same results as in Stata) using the extensive universe of packages in R (see the Comprehensive R Archive Network; CRAN). Moreover, intermediate knowledge of those packages and of R itself is required.
- Figure 2, Cook's Distance (Cook's D) of the fixed effects from Table 5, and Figure 4 could not be replicated perfectly. The differences are, however, negligible.

1.1 Install and Load CRAN Packages

In this replication file, `tidyverse` (Wickham, 2022b), `haven` (Wickham, 2022a), `sjmisc` (Lüdecke et al., 2021), `ggpubr` (Kassambara, 2022), `lme4` (Bates et al., 2022), `easystats` (Lüdecke et al., 2022), and `influence.ME` (Nieuwenhuis et al., 2017) will primarily be used.

```
library(tidyverse) #data wrangling and visualization
library(haven) #load STATA data
library(sjmisc) #additional data wrangling options
library(ggpubr) #additional visualization options
library(lme4) #compute mixed-effects models
library(easystats) #obtain regression diagnostics
library(influence.ME) #compute outlier statistics
```

Alternatively, all packages can be loaded using the `pacman` package (Rinker & Kurkiewicz, 2019) (which would first need to be installed).

```
pacman::p_load(tidyverse, #data wrangling and visualization
               haven, #load STATA data
               sjmisc, #additional data wrangling options
               ggpubr, #additional visualization options
               lme4, #compute mixed-effects models
               easystats, #obtain regression diagnostics
               influence.ME #compute outlier statistics
               )
```

1.2 Load and Wrangle Data

1.2.1 Load Micro- and Macrodata

Two options for loading the data exist:

Either:

```
#load OECD data
oecd_data <- read_dta("Data/OECDdata.dta") %>%
  sjlabelled::remove_all_labels()

#load and adjust ESS data
ess_data <- read_dta("Data/ESS_example_icsr.dta") %>%
  sjlabelled::remove_all_labels() %>%
  mutate(Year = sjmisc::rec(essround,
                            rec="1=2002; 2=2004; 3=2006; 4=2008;
                            5=2010; 6=2012; 7=2014"),
         Country = as.numeric(as.factor(cntry)))

#merge dataframes
data <- inner_join(ess_data, oecd_data, by=c("Country", "Year")) %>%
  ##adjust vars
  mutate(Year = as.factor(Year),
         male = sjmisc::rec(gndr, rec="2=0; 1=1"),
         attend = (rlgatnd * -1) + 7)
```

Or:

```
data <- inner_join(
  #load and adjust ESS data
  read_dta("Data/ESS_example_icsr.dta") %>%
    sjlabelled::remove_all_labels() %>%
    mutate(Year = sjmisc::rec(essround, rec="1=2002; 2=2004; 3=2006; 4=2008;
                              5=2010; 6=2012; 7=2014"),
           Country = as.numeric(as.factor(cntry))),
  #load OECD data
  read_dta("Data/OECDdata.dta") %>%
    sjlabelled::remove_all_labels(),
  #merge dataframes
  by=c("Country", "Year")) %>%
  #adjust vars
  mutate(male = sjmisc::rec(gndr, rec="2=0; 1=1"),
         attend = (rlgatnd * -1) + 7)
```

1.2.2 Data Wrangling

Reducing data to final sample size (listwise deletion):

```
data <- data[complete.cases(data[,c("attend", "socspend", "gdppc", "domicil",
                                    "eduyrs", "hincfel", "male", "agea",
                                    "Year")]),]
```

Alternatively, listwise deletion can be achieved through first estimating an OLS model to identify rows with missing values (i.e., NAs) and subsequently remove the previously identified unobserved observations:

```
#estimate OLS regression model for identifying NA rows
regmodel_na <- lm(attend ~
                    socspend+gdppc+domicil+eduyrs+hincfel+male+agea+Year,
                  data=data)

#reduce dataset to final sample + remove unnecessary vars
data$used <- T
data$used[na.action(regmodel_na)] <- F

data <- data %>%
  filter(used == T) %>%
  subset(select=-c(cname, cedition, cproddat, cseqno, name,
                   essround, edition, idno, used))

#remove object from environment
rm(regmodel_na)
```

Next, several country-level variables will be created.

Country-year-level mean variables:

```
data <- data %>%
  group_by(Country, Year) %>%
  mutate(across(c(eduyrs, domicil, attend),
              mean, na.rm=T, .names="{col}_C")) %>%
  ungroup()
```

Country-level mean variables:

```
data <- data %>%
  group_by(Country) %>%
  mutate(across(c(socspend, gdppc, eduyrs_C, domicil_C, attend),
              mean, na.rm=T, .names="{col}_m")) %>%
  ungroup()
```

Country-level demeaned (group mean centered) variables:

```
data <- data %>%
  group_by(Country) %>%
  mutate(socspend_dm = socspend - socspend_m,
         gdppc_dm = gdppc - gdppc_m,
         eduyrs_C_dm = eduyrs_C - eduyrs_C_m,
         domicil_C_dm = domicil_C - domicil_C_m,
         attend_dm = attend - attend_m) %>%
  ungroup()
```

Alternatively, the `de_mean()` command from the `sjmisc` package (Lüdecke et al., 2021) can be used to create the country mean (BE) and country demeaned (WE) variables.

```
data <- data %>%
  sjmisc::de_mean(socspend, gdppc, eduyrs_C, domicil_C, attend, grp="Country",
                  append=T, suffix.dm="_dm", suffix.gm="_m")
```

2 R Replication

2.1 Table 1 (p. 104)

Before estimating Model M0-M2, I will convert the gender variable (`male`) from a numeric to a factor variable and set the seed to ensure reproduceability.

```
#convert to factor
data$male <- as.factor(data$male)

#set seed
set.seed(313)
```

Estimating Model M0 from Table 1 in Schmidt-Catran et al. (2019) using Restricted Maximum Likelihood (REML):

```
m0_reml <- lmer(attend~
                  1+(1|Country),
                data=data %>% filter(Year == "2014"), REML=T)
```

Save variances from previously estimated model:

```
#country-level variance
m0_reml_cv <- as.data.frame(print(VarCorr(m0_reml), comp="Variance"))[1,"vcov"]

#individual-level variance
m0_reml_iv <- as.data.frame(print(VarCorr(m0_reml), comp="Variance"))[2,"vcov"]
```

Calculating intraclass correlation coefficient (ICC) by hand:

```
round((m0_reml_cv)/(m0_reml_cv+m0_reml_iv),3)
```

```
## [1] 0.142
```

Estimating Model M0 using Full Maximum Likelihood (FML) (not included in paper):

```
m0_fml <- lmer(attend~
                 1+(1|Country),
               data=data %>% filter(Year == "2014"), REML=F)
```

Obtaining the ICC of the REML and FML Model M0:

```
data.frame(Model = c("REML", "FML"),
           ICC = c(round(performance::icc(m0_reml)[["ICC_adjusted"]],3),
                   round(performance::icc(m0_fml)[["ICC_adjusted"]],3)))
```

Model	ICC
REML	0.142
FML	0.136

Estimating Model M1 and M2 from Table 1 in Schmidt-Catran et al. (2019):

```
#estimate Model M1
m1 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+(1|Country),
           data=data %>% filter(Year == "2014"), REML=T)

#estimate Model M2
m2 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+socspend+gdppc+
             domicil_C+eduyrs_C+(1|Country),
           data=data %>% filter(Year == "2014"), REML=T)
```

Save variances from Model M1:

```
#country-level variance
m1_cv <- as.data.frame(print(VarCorr(m1), comp="Variance"))[1,"vcov"]

#individual-level variance
m1_iv <- as.data.frame(print(VarCorr(m1), comp="Variance"))[2,"vcov"]
```

Calculate reduction in country-level and individual-level variance from Model M0 to Model M1:

```
#country-level variance
1-(m1_cv/m0_reml_cv)
```

```
## [1] 0.02474368
```

```
#individual-level variance
1-(m1_iv/m0_reml_iv)
```

```
## [1] 0.02810252
```

Save variances from Model M2:

```
#country-level variance
m2_cv <- as.data.frame(print(VarCorr(m2), comp="Variance"))[1,"vcov"]

#individual-level variance
m2_iv <- as.data.frame(print(VarCorr(m2), comp="Variance"))[2,"vcov"]
```

Calculate reduction in country-level and individual-level variance from Model M1 to Model M2:

```
#country-level variance
1-(m2_cv/m1_cv)
```

```
## [1] 0.07967292
```

```
#individual-level variance
1-(m2_iv/m1_iv)
```

```
## [1] 0.0000000428942
```

Regression table (Table 1 in paper):

```
stargazer::stargazer(

  m0_reml, m1, m2,
  digits=4, df=F, model.numbers=F, single.row=F, header=F, no.space=T, title="",
  omit.stat = c("ll","aic", "bic", "n"),
  star.cutoffs=c(.05, .01, .001),
  star.char = c("*", "**", "***"),
  column.labels=c("M0", "M1", "M2"),
  add.lines=list(c("var(country)",
                   round(as.data.frame(VarCorr(m0_reml))[1,"vcov"],4),
                   round(as.data.frame(VarCorr(m1))[1,"vcov"],4),
```

```
                  round(as.data.frame(VarCorr(m2))[1,"vcov"],4)),
                c("var(individual)",
                  round(as.data.frame(VarCorr(m0_reml))[2,"vcov"],4),
                  round(as.data.frame(VarCorr(m1))[2,"vcov"],4),
                  round(as.data.frame(VarCorr(m2))[2,"vcov"],4)),
                c("n(country)",
                  nrow(as.data.frame(ranef(m0_reml))),
                  nrow(as.data.frame(ranef(m1))),
                  nrow(as.data.frame(ranef(m2)))),
                c("n(individuals)",
                  nrow(as.data.frame(residuals(m0_reml))),
                  nrow(as.data.frame(residuals(m1))),
                  nrow(as.data.frame(residuals(m2))))),
  font.size="normalsize",
  report=("vc*"),
  table.layout="=c-!t-!a=n"

)
```

	M0	M1	M2
domicil		0.0669***	0.0669***
eduyrs		−0.0156***	−0.0156***
hincfel		−0.0120	−0.0123
male1		−0.2077***	−0.2076***
agea		0.0091***	0.0091***
socspend			−0.0465
gdppc			−0.00001
domicil_C			0.4394
eduyrs_C			−0.0996
Constant	1.4620***	1.1409***	2.7045
var(country)	0.3348	0.3265	0.3005
var(individual)	2.0231	1.9662	1.9662
n(country)	20	20	20
n(individuals)	37028	37028	37028

Note: *p<0.05; **p<0.01; ***p<0.001

2.2 Bivariate Correlations (p. 107)

First, the dataframe `data` must be aggregated to the country level. The newly created dataframe is called `data_m`.

```r
data_m <- data %>%
  group_by(Country, cntry) %>%
  summarise_all(mean, na.rm=T) %>%
  ungroup()
```

Now, I will create a for loop to obtain jackknifed correlations (Pearson's *r*).

```r
countries <- unique(data_m$cntry)
cor_list_jack = list()
counter = ""
min = 1
max = -1
min_cntry = ""
max_cntry = ""

for (i in countries){

  counter = i
  cor_list_jack[[counter]] <- cor.test(data_m[data_m$cntry!=i,]$socspend_m,
                                       data_m[data_m$cntry!=i,]$attend_m)

  if (cor_list_jack[[counter]]$estimate < min){
    min <- cor_list_jack[[counter]]$estimate
    min_cntry <- i
  }

  if (cor_list_jack[[counter]]$estimate > max){
    max <- cor_list_jack[[counter]]$estimate
    max_cntry <- i
  }

  print(paste("All countries except", i))
  print(cor_list_jack[[counter]])

}
```

The minimum correlation between church attendance and social spending is:

```r
print(paste(min_cntry, min, sep=":"))
```

```
## [1] "EE:-0.407623977404284"
```

The maximum correlation between church attendance and social spending is:

```
print(paste(max_cntry, max, sep=":"))
```

```
## [1] "TR:-0.267719603401317"
```

The average correlation between church attendance and social spending is:

```
as.data.frame(round(cor(data_m[c("attend_m", "socspend_m")]),4))
```

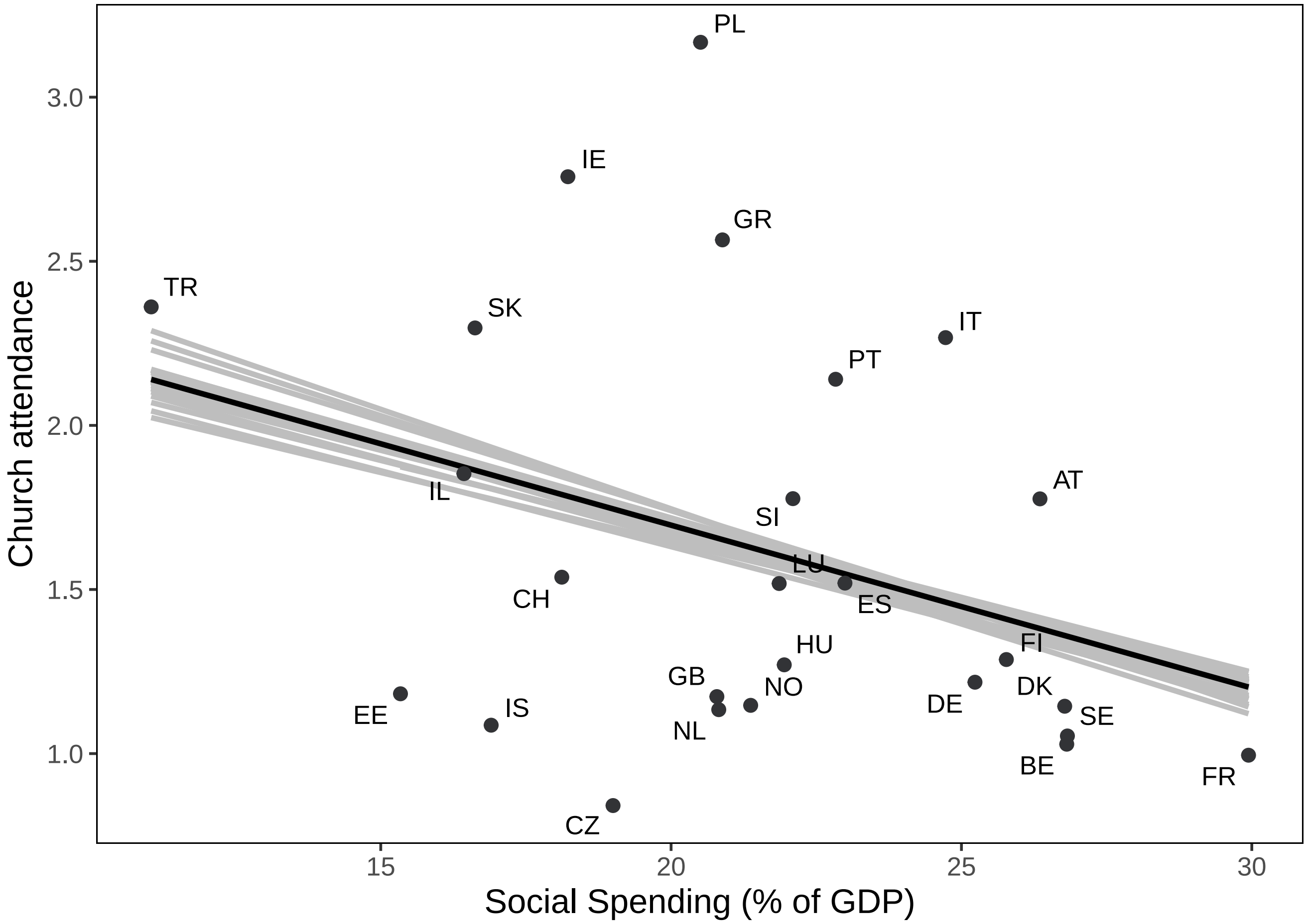
	attend_m	socspend_m
attend_m	1.0000	-0.3448
socspend_m	-0.3448	1.0000

2.3 Figure 2 (p. 108)

This graph will use the aggregated data_m dataframe created earlier. Figure 2 will be created using ggplot2 (Wickham et al., 2022) from the package tidyverse (Wickham, 2022b).

```
ggplot(data=data_m, aes(x=socspend_m, y=attend_m)) +
  geom_smooth(data=data_m[data_m$Country!=1,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=2,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=4,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=6,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=7,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=8,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=9,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=10,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=11,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=12,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=13,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=14,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data_m[data_m$Country!=16,], aes(x=socspend_m, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
```

```r
geom_smooth(data=data_m[data_m$Country!=17,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=18,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=19,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=20,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=22,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=23,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=24,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=25,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=26,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=28,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=29,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=30,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data_m[data_m$Country!=31,], aes(x=socspend_m, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(method="lm", se=F, color="black") +
geom_point(size=2, color="#323336") +
ggrepel::geom_text_repel(data=data_m, aes(label=cntry), size=3.5) +
labs(x="Social Spending (% of GDP)", y="Church attendance") +
theme(plot.background = element_blank(),
      panel.background = element_rect(fill="white", colour="black"),
      panel.grid.minor = element_blank(),
      panel.grid.major = element_blank(),
      axis.text = element_text(size=10),
      axis.title = element_text(size=13))
```



2.4 Table 2 (p. 115-116)

Estimating Model M3-M5:

```
m3 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+socspend+gdppc+
             domicil_C+eduyrs_C+Year+(1|Country)+(1|Country:Year),
           data=data, REML=T)

m4 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+socspend+gdppc+
             domicil_C+eduyrs_C+Year+(1|Country),
           data=data, REML=T)

m5 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+socspend_m+gdppc_m+
             domicil_C_m+eduyrs_C_m+Year+(1|Country),
           data=data, REML=T)
```

Regression table (Table 2 in paper):

```r
stargazer::stargazer(

  m3, m4, m5,
  digits=4, df=F, model.numbers=F, single.row=F, header=F, no.space=T, title="",
  apply.t=abs,
  omit.stat = c("ll","aic", "bic", "n"),
  star.cutoffs=c(.05, .01, .001),
  star.char = c("*", "**", "***"),
  column.labels=c("\\shortstack{M3 \\\\ b/|z|}",
                  "\\shortstack{M4 \\\\ b/|z|}",
                  "\\shortstack{M5 \\\\ b/|z|}"),
  add.lines=list(c("var(country)",
                   round(as.data.frame(VarCorr(m3))[2,"vcov"],4),
                   round(as.data.frame(VarCorr(m4))[1,"vcov"],4),
                   round(as.data.frame(VarCorr(m5))[1,"vcov"],4)),
                 c("var(country-year)",
                   round(as.data.frame(VarCorr(m3))[1,"vcov"],4),
                   NA,
                   NA),
                 c("var(individual)",
                   round(as.data.frame(VarCorr(m3))[3,"vcov"],4),
                   round(as.data.frame(VarCorr(m4))[2,"vcov"],4),
                   round(as.data.frame(VarCorr(m5))[2,"vcov"],4)),
                 c("n(country)",
                   nrow(as.data.frame(ranef(m3)) %>%
                          filter(grpvar=="Country")),
                   nrow(as.data.frame(ranef(m4)) %>%
                          filter(grpvar=="Country")),
                   nrow(as.data.frame(ranef(m5)) %>%
                          filter(grpvar=="Country"))),
                 c("n(country-year)",
                   nrow(as.data.frame(ranef(m3)) %>%
                          filter(grpvar=="Country:Year")),
                   NA,
                   NA),
                 c("n(individuals)",
                   nrow(as.data.frame(residuals(m3))),
                   nrow(as.data.frame(residuals(m4))),
                   nrow(as.data.frame(residuals(m5))))),
  font.size="footnotesize",
  report=("vc*t"),
  table.layout="=c-!t-!a=n"

)
```

	M3	M4	M5
	b/|z|	b/|z|	b/|z|
domicil	0.0940***	0.0939***	0.0937***
	t = 40.8602	t = 40.7989	t = 40.7786
eduyrs	−0.0131***	−0.0131***	−0.0134***
	t = 17.6975	t = 17.7353	t = 18.1736
hincfel	−0.0272***	−0.0292***	−0.0309***
	t = 7.7221	t = 8.3037	t = 8.7872
male1	−0.2493***	−0.2499***	−0.2501***
	t = 46.5050	t = 46.5642	t = 46.6009
agea	0.0116***	0.0117***	0.0117***
	t = 77.0827	t = 77.3547	t = 77.3707
socspend	−0.0137*	−0.0152***	
	t = 2.1967	t = 6.3204	
gdppc	−0.000004	−0.000003	
	t = 1.0054	t = 1.8319	
domicil_C	−0.0542	−0.0407	
	t = 0.6469	t = 1.2676	
eduyrs_C	−0.0277	−0.0198*	
	t = 1.3020	t = 2.3168	
socspend_m			−0.0385
			t = 1.3265
gdppc_m			−0.00002
			t = 1.1746
domicil_C_m			0.2738
			t = 0.6572
eduyrs_C_m			−0.1098
			t = 1.3742
Year2004	0.0013	0.0009	−0.0041
	t = 0.0460	t = 0.0851	t = 0.4091
Year2006	−0.0187	−0.0268	−0.0473***
	t = 0.4947	t = 1.7819	t = 4.4705
Year2008	−0.0158	−0.0269	−0.0649***
	t = 0.3284	t = 1.3830	t = 6.4057
Year2010	−0.0278	−0.0388	−0.1144***
	t = 0.5217	t = 1.7965	t = 11.2168
Year2012	−0.0366	−0.0480	−0.1269***
	t = 0.5952	t = 1.9037	t = 12.4291
Year2014	−0.0109	−0.0271	−0.1171***
	t = 0.1525	t = 0.9168	t = 11.2665
Constant	2.0519***	1.9339***	3.0919*
	t = 5.1533	t = 9.9077	t = 2.5500
var(country)	0.3389	0.3442	0.3267
var(country-year)	0.0063		
var(individual)	1.968	1.9729	1.9732
n(country)	26	26	26
n(country-year)	149		
n(individuals)	277505	277505	277505

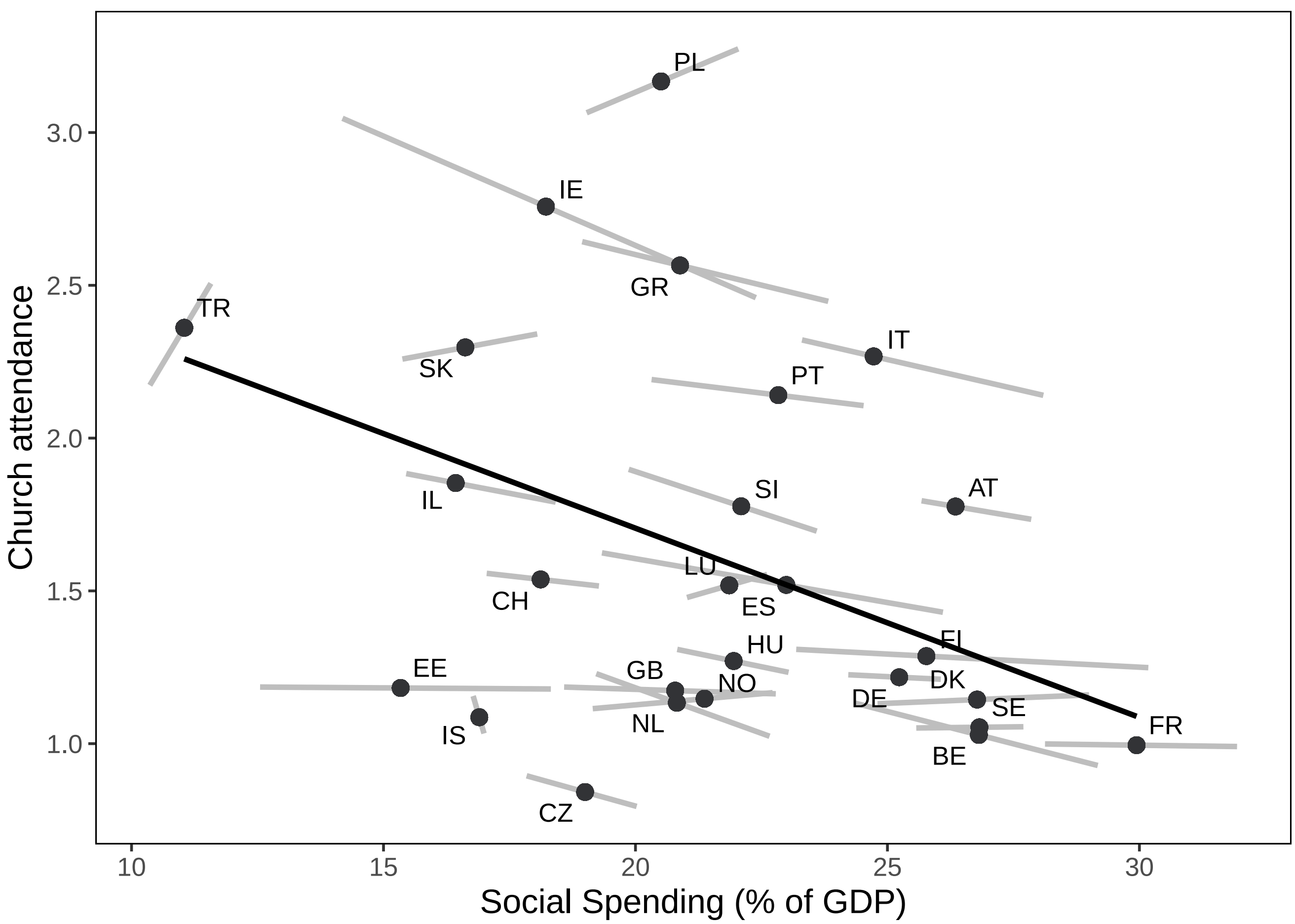
Note: $^{*}p<0.05$; $^{**}p<0.01$; $^{***}p<0.001$

2.5 Figure 3 (p. 117)

The next graph, Figure 3, uses the disaggregated dataframe data.

```
ggplot(data=data, aes(x=socspend_m, y=attend_m)) +
  geom_smooth(data=data[data$Country==1,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==2,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==4,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==6,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==7,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==8,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==9,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==10,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==11,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==12,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==13,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==14,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==16,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==17,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==18,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==19,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==20,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==22,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==23,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==24,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
  geom_smooth(data=data[data$Country==25,], aes(x=socspend, y=attend_C),
              method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
```

```r
geom_smooth(data=data[data$Country==26,], aes(x=socspend, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data[data$Country==28,], aes(x=socspend, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data[data$Country==29,], aes(x=socspend, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data[data$Country==30,], aes(x=socspend, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_smooth(data=data[data$Country==31,], aes(x=socspend, y=attend_C),
            method="lm", se=F, color="grey", alpha=.5, size=1, inherit.aes=F) +
geom_point(size=2.5, color="#323336") +
geom_smooth(method="lm", se=F, color="black") +
ggrepel::geom_text_repel(data=data_m, aes(label=cntry), size=3.5) +
labs(x="Social Spending (% of GDP)", y="Church attendance") +
theme(plot.background = element_blank(),
      panel.background = element_rect(fill="white", colour="black"),
      panel.grid.minor = element_blank(),
      panel.grid.major = element_blank(),
      axis.text = element_text(size=10),
      axis.title = element_text(size=13))
```



2.6 Table 3 (p. 118-119)

Estimating Model M6:

```
m6 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+socspend_m+socspend_dm+
             gdppc_m+gdppc_dm+domicil_C_m+domicil_C_dm+eduyrs_C_m+eduyrs_C_dm+
             Year+(1|Country)+(1|Country:Year),
           data=data, REML=T)
```

Identify outliers or influential observations at the country level using Cook's Distance and DFBETAs:

```
#create "influence object"
influence_obj <-
  influence.ME::influence(m6, "Country")

#table with outlier statistics
outliers_m6 <-
  data.frame(Country = unique(data$cntry),
             #Cook's D
             Cooks_D =
               round(influence.ME::cooks.distance.estex(influence_obj),4),
             #DFBETAs
             DFBETAs =
               round(influence.ME::dfbetas.estex(influence_obj,
                                                 parameters="socspend_dm"),4)
             ) %>%
  #add stars (cut-off values)
  mutate(Cooks_D_cutoff = case_when(Cooks_D >= 0.1538 ~ "*",
                                    TRUE ~ " "), .after=Cooks_D) %>%
  #add stars (cut-off values)
  mutate(DFBETAs_cutoff = case_when(DFBETAs <= -0.3922 | DFBETAs >= 0.3922 ~ "*",
                                    TRUE ~ " "), .after=DFBETAs) %>%
  #paste statistic and stars together
  unite("cooksd", c(Cooks_D, Cooks_D_cutoff), sep="") %>%
  unite("dfbetas", c(DFBETAs, DFBETAs_cutoff), sep="")
```

Estimating Model M7:

```
m7 <- lmer(attend~
             domicil+eduyrs+hincfel+male+agea+socspend_m+socspend_dm+
             gdppc_m+gdppc_dm+domicil_C_m+domicil_C_dm+eduyrs_C_m+eduyrs_C_dm+
             Year+(1|Country)+(1|Country:Year),
           data=data[data$Country!="8" &
                       data$Country!="17" &
                       data$Country!="25",], REML=T)
```

Estimating Model M3 without the outliers identified earlier (results not shown here):

```
m3_out <- lmerTest::lmer(attend~
                           socspend+gdppc+eduyrs_C+domicil_C+domicil+eduyrs+
                           hincfel+male+agea+Year+(1|Country)+(1|Country:Year),
                         data=data[data$Country!="8" &
                                     data$Country!="17" &
                                     data$Country!="25",], REML=T)
```

Regression table (Table 3 in paper):

```
stargazer::stargazer(

  m6, m7,
  digits=4, df=F, model.numbers=F, single.row=F, header=F, no.space=T, title="",
  omit.stat = c("ll","aic", "bic", "n"),
  star.cutoffs=c(.05, .01, .001),
  star.char = c("*", "**", "***"),
  column.labels=c("\\shortstack{M6 \\\\ b/p}",
                  "\\shortstack{M7 \\\\ b/p}"),
  add.lines=list(c("var(country)",
                   round(as.data.frame(VarCorr(m6))[2,"vcov"],4),
                   round(as.data.frame(VarCorr(m7))[2,"vcov"],4)),
                 c("var(country-year)",
                   round(as.data.frame(VarCorr(m6))[1,"vcov"],4),
                   round(as.data.frame(VarCorr(m7))[1,"vcov"],4)),
                 c("var(individual)",
                   round(as.data.frame(VarCorr(m6))[3,"vcov"],4),
                   round(as.data.frame(VarCorr(m7))[3,"vcov"],4)),
                 c("n(country)",
                   nrow(as.data.frame(ranef(m6)) %>%
                          filter(grpvar=="Country")),
                   nrow(as.data.frame(ranef(m7)) %>%
                          filter(grpvar=="Country"))),
                 c("n(country-year)",
                   nrow(as.data.frame(ranef(m6)) %>%
                          filter(grpvar=="Country:Year")),
                   nrow(as.data.frame(ranef(m7)) %>%
                          filter(grpvar=="Country:Year"))),
                 c("n(individuals)",
                   nrow(as.data.frame(residuals(m6))),
                   nrow(as.data.frame(residuals(m7))))),
  font.size="footnotesize",
  report=("vc*p"),
  table.layout="=c-!t-!a=n"

)
```

	M6 b/p	M7 b/p
domicil	0.0940***	0.0827***
	p = 0.0000	p = 0.0000
eduyrs	−0.0131***	−0.0131***
	p = 0.0000	p = 0.0000
hincfel	−0.0273***	−0.0130***
	p = 0.0000	p = 0.0006
male1	−0.2494***	−0.2334***
	p = 0.0000	p = 0.0000
agea	0.0116***	0.0108***
	p = 0.0000	p = 0.0000
socspend_m	−0.0382	−0.0139
	p = 0.1888	p = 0.5393
socspend_dm	−0.0112	−0.0081
	p = 0.0826	p = 0.2338
gdppc_m	−0.00002	−0.000004
	p = 0.2512	p = 0.6884
gdppc_dm	−0.000001	−0.000000
	p = 0.8811	p = 0.9079
domicil_C_m	0.2796	−0.1784
	p = 0.5031	p = 0.5785
domicil_C_dm	−0.0485	−0.0509
	p = 0.5726	p = 0.5111
eduyrs_C_m	−0.1109	−0.1532**
	p = 0.1668	p = 0.0086
eduyrs_C_dm	−0.0139	−0.0021
	p = 0.5379	p = 0.9231
Year2004	−0.0056	−0.0226
	p = 0.8401	p = 0.3807
Year2006	−0.0428	−0.0414
	p = 0.2819	p = 0.2641
Year2008	−0.0537	−0.0514
	p = 0.2999	p = 0.2794
Year2010	−0.0725	−0.0759
	p = 0.2109	p = 0.1514
Year2012	−0.0899	−0.0919
	p = 0.1802	p = 0.1348
Year2014	−0.0746	−0.0815
	p = 0.3420	p = 0.2566
Constant	3.0431*	3.8998***
	p = 0.0125	p = 0.00002
var(country)	0.327	0.1662
var(country-year)	0.0062	0.0044
var(individual)	1.968	1.9903
n(country)	26	23
n(country-year)	149	128
n(individuals)	277505	239881

Note: *p<0.05; **p<0.01; ***p<0.001

2.7 Figure 4 (p. 120)

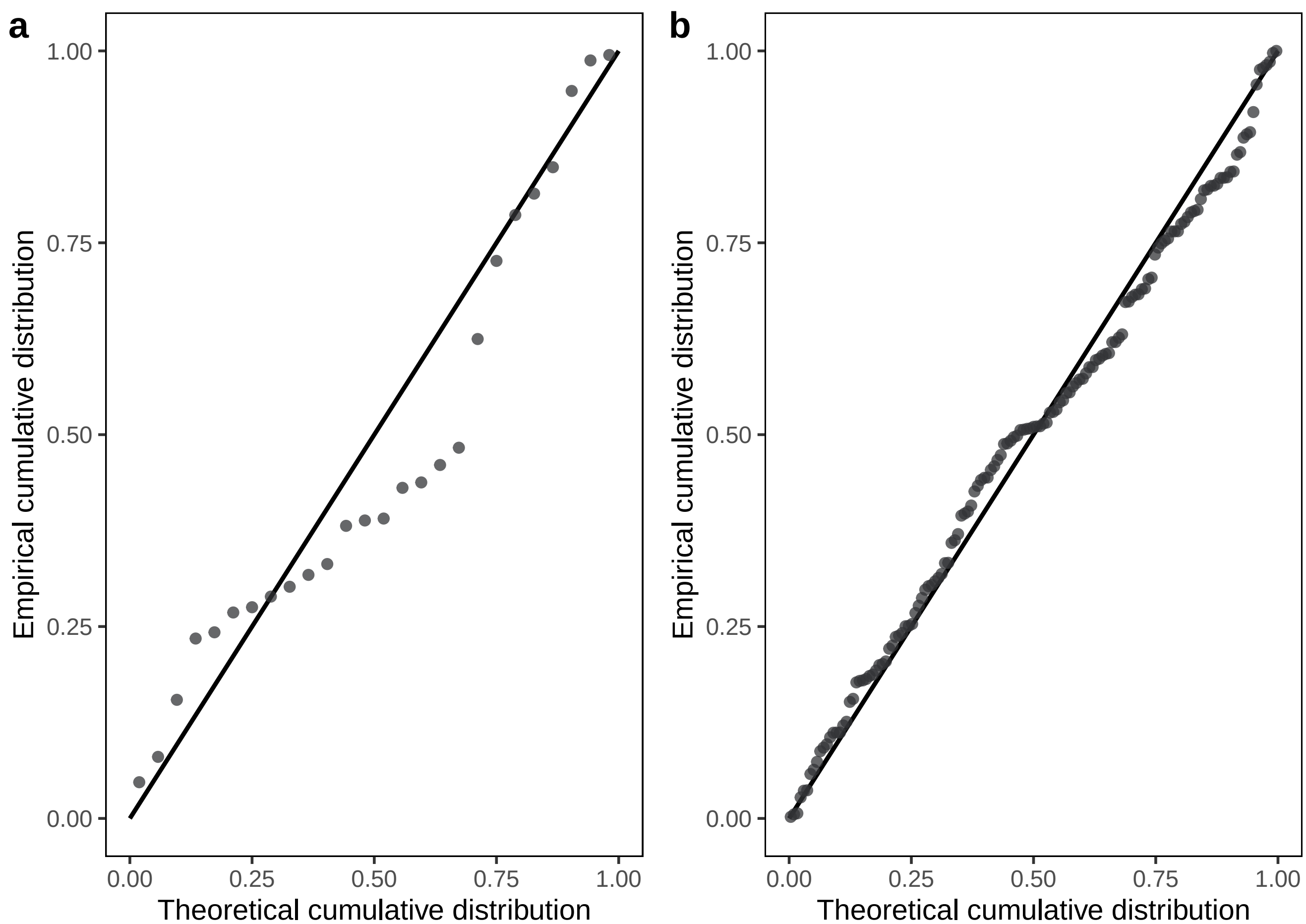
Creating the P-P plot (Figure 4) from Schmidt-Catran et al. (2019):

```
#obtain upper-level residuals from Model M6
resid_m6_country <- ranef(m6)[["Country"]]
resid_m6_country_year <- ranef(m6)[["Country:Year"]]

#create P-P plot of country-level residuals
pp_plot1 <- ggplot(data=resid_m6_country,
                   aes(sample=resid_m6_country$`(Intercept)`)) +
  qqplotr::stat_pp_line(colour="black") +
  qqplotr::stat_pp_point(colour="#323336", alpha=.75) +
  labs(x="Theoretical cumulative distribution",
       y="Empirical cumulative distribution") +
  theme(plot.background = element_blank(),
        panel.background = element_rect(fill="white", colour="black"),
        panel.grid.minor = element_blank(),
        panel.grid.major = element_blank(),
        axis.text = element_text(size=9),
        axis.title = element_text(size=11))

#create P-P plot of country-year-level residuals
pp_plot2 <- ggplot(data=resid_m6_country_year,
                   aes(sample=resid_m6_country_year$`(Intercept)`)) +
  qqplotr::stat_pp_line(colour="black") +
  qqplotr::stat_pp_point(colour="#323336", alpha=.75) +
  labs(x="Theoretical cumulative distribution",
       y="Empirical cumulative distribution") +
  theme(plot.background = element_blank(),
        panel.background = element_rect(fill="white", colour="black"),
        panel.grid.minor = element_blank(),
        panel.grid.major = element_blank(),
        axis.text = element_text(size=9),
        axis.title = element_text(size=11))

#plot both P-P plots
ggpubr::ggarrange(pp_plot1, pp_plot2, ncol=2, labels=c("a", "b"))
```



2.8 Appendix

2.8.1 Table 4 (p. 124):

```
addmargins(table(data$cntry, data$Year)) %>%
  kbl(booktabs=T, align="llllll", linesep="") %>%
  kable_styling(font_size=12, full_width=FALSE)
```

2.8.2 Table 5 (p. 125):

```
outliers_m6 %>%
  kbl(booktabs=T, align="lcc", linesep="",
      col.names=c("Country", "Cook's D", "DFBETAs")) %>%
  kable_styling(font_size=12, full_width=FALSE)
```

	2002	2004	2006	2008	2010	2012	2014	Sum
AT	2138	2154	2288	0	0	0	1778	8358
BE	1725	1742	1781	1747	1662	1863	1759	12279
CH	2002	2111	1787	1785	1473	1477	1508	12143
CZ	1224	2525	0	1934	2279	1737	1943	11642
DE	2849	2723	2803	2688	2986	2920	3001	19970
DK	1461	1451	1443	1584	1557	1621	1487	10604
EE	0	1970	1471	1596	1784	2358	2016	11195
ES	1532	1589	1743	2481	1834	1838	1849	12866
FI	1972	1997	1880	2181	1852	2169	2064	14115
FR	0	0	1965	2039	1714	1952	1895	9565
GB	2007	1862	2342	2300	2352	2222	2221	15306
GR	2515	2388	0	2039	2667	0	0	9609
HU	1672	1471	1481	1511	1548	1937	1645	11265
IE	1930	2195	1586	1743	2514	2582	2316	14866
IL	2289	0	0	2264	2020	2353	2460	11386
IS	0	551	0	0	0	720	0	1271
IT	1160	1484	0	0	0	856	0	3500
LU	1403	1567	0	0	0	0	0	2970
NL	2309	1847	1870	1748	1784	1825	1881	13264
NO	2027	1754	1744	1541	1540	1614	1432	11652
PL	2071	1695	1687	1595	1703	1840	1563	12154
PT	1456	1989	2072	2237	2003	2066	1239	13062
SE	1979	1924	1903	1811	1488	1812	1761	12678
SI	1487	1369	1433	1234	1352	1233	1210	9318
SK	0	1373	1649	1725	1790	1784	0	8321
TR	0	1805	0	2341	0	0	0	4146
Sum	39208	43536	34928	42124	39902	40779	37028	277505

	Country	Cook's D	DFBETAs
1	AT	0.3132*	0.0025
2	BE	0.4202*	-0.1199
4	CH	0.1	-0.0782
6	CZ	0.0825	0.0474
7	DE	0.617*	-0.2021
8	DK	0.3026*	0.5436*
9	EE	0.7283*	0.1051
10	ES	1.8804*	-0.0312
11	FI	0.6731*	0.2576
12	FR	0.3586*	0.1015
13	GB	1.3143*	0.0654
14	GR	0.6553*	-0.0859
16	HU	0.1531	0.1163
17	IE	5.3724*	-1.8462*
18	IL	5.0814*	-0.3309
19	IS	0.0065	-0.0238
20	IT	0.0551	0.0137
22	LU	0.0331	0.113
23	NL	0.1592*	-0.0785
24	NO	0.3044*	0.0436
25	PL	0.4881*	0.7625*
26	PT	2.7471*	0.0936
28	SE	0.4498*	-0.2784
29	SI	0.7143*	0.192
30	SK	0.5514*	0.0055
31	TR	2.2641*	0.2088

2.9 Online Appendix

To create Table OA1, I will create a dataframe (`descr`), which contains the descriptive statistics of all variables.

Please note that a function could be used instead of fully manually creating the table (i.e., inserting each value individually).

```
descr <-
  data.frame(

    Variables = c("Church attendance", "Urban vs. rural", "Education (in years)",
                  "Sujective income", "Male", "Age", "Social Spending",
                  "GDP/capita", "Average education", "Average urban vs. rural",
                  "Social Spending", "GDP/capita", "Average education",
                  "Average urban vs. rural", "Social Spending", "GDP/capita",
                  "Average education", "Average urban vs. rural"),
    N = nrow(data),
    Mean = c(mean(data$attend), mean(data$domicil), mean(data$eduyrs),
             mean(data$hincfel), mean(as.numeric(data$male)), mean(data$agea),
             mean(data$socspend), mean(data$gdppc), mean(data$eduyrs_C),
             mean(data$domicil_C), mean(data$socspend_m), mean(data$gdppc_m),
             mean(data$eduyrs_C_m), mean(data$domicil_C_m), mean(data$socspend_dm),
             mean(data$gdppc_dm), mean(data$eduyrs_C_dm), mean(data$domicil_C_dm)),
    "Std. Dev." = c(sd(data$attend), sd(data$domicil), sd(data$eduyrs),
                    sd(data$hincfel), sd(as.numeric(data$male)), sd(data$agea),
                    sd(data$socspend), sd(data$gdppc), sd(data$eduyrs_C),
                    sd(data$domicil_C), sd(data$socspend_m), sd(data$gdppc_m),
                    sd(data$eduyrs_C_m), sd(data$domicil_C_m), sd(data$socspend_dm),
                    sd(data$gdppc_dm), sd(data$eduyrs_C_dm), sd(data$domicil_C_dm)),
    Min = c(min(data$attend), min(data$domicil), min(data$eduyrs), min(data$hincfel),
            min(as.numeric(data$male)), min(data$agea), min(data$socspend),
            min(data$gdppc), min(data$eduyrs_C), min(data$domicil_C),
            min(data$socspend_m), min(data$gdppc_m), min(data$eduyrs_C_m),
            min(data$domicil_C_m), min(data$socspend_dm), min(data$gdppc_dm),
            min(data$eduyrs_C_dm), min(data$domicil_C_dm)),
    Max = c(max(data$attend), max(data$domicil), max(data$eduyrs),
            max(data$hincfel), max(as.numeric(data$male)), max(data$agea),
            max(data$socspend), max(data$gdppc), max(data$eduyrs_C),
            max(data$domicil_C), max(data$socspend_m), max(data$gdppc_m),
            max(data$eduyrs_C_m), max(data$domicil_C_m), max(data$socspend_dm),
            max(data$gdppc_dm), max(data$eduyrs_C_dm), max(data$domicil_C_dm)),
    check.names=F

  )
```

Now Table OA1 can be created.

Variables	N	Mean	Std. Dev.	Min	Max
Individual-level variables					
Church attendance	277505	1.583	1.566	0.000	6.000
Urban vs. rural	277505	2.903	1.220	1.000	5.000
Education (in years)	277505	12.237	4.159	0.000	56.000
Sujective income	277505	1.995	0.852	1.000	4.000
Male	277505	1.468	0.499	1.000	2.000
Age	277505	47.672	18.479	13.000	123.000
Original country-level variables					
Social Spending	277505	21.968	4.238	10.363	31.938
GDP/capita	277505	33938.624	10922.754	10753.466	65716.476
Average education	277505	12.237	1.506	6.591	14.606
Average urban vs. rural	277505	2.903	0.325	1.810	3.483
Between-country variables					
Social Spending	277505	21.968	3.912	11.047	29.944
GDP/capita	277505	33938.624	9646.664	13660.016	60830.930
Average education	277505	12.237	1.430	6.711	14.036
Average urban vs. rural	277505	2.903	0.313	2.009	3.364
Within-country variables					
Social Spending	277505	0.000	1.631	-4.037	4.404
GDP/capita	277505	0.000	5123.321	-16095.558	13103.317
Average education	277505	0.000	0.472	-1.489	1.893
Average urban vs. rural	277505	0.000	0.086	-0.297	0.251

```
descr %>%
  kbl(booktabs=T, align="lccccc", digits=3, linesep="") %>%
  kable_styling(font_size=12, full_width=FALSE) %>%
  pack_rows(group_label="Individual-level variables",
            start_row=1, end_row=6, bold=F, italic=T) %>%
  pack_rows(group_label="Original country-level variables",
            start_row=7, end_row=10, bold=F, italic=T) %>%
  pack_rows(group_label="Between-country variables",
            start_row=11, end_row=14, bold=F, italic=T) %>%
  pack_rows(group_label="Within-country variables",
            start_row=15, end_row=18, bold=F, italic=T)
```

Table OA2 - Correlation Matrix[1]:

```
cortab <- round(cor(data[c("attend", "domicil", "eduyrs", "hincfel", "male",
                           "agea", "socspend", "gdppc", "eduyrs_C",
                           "domicil_C", "socspend_m", "gdppc_m", "eduyrs_C_m",
                           "domicil_C_m", "socspend_dm", "gdppc_dm",
                           "eduyrs_C_dm", "domicil_C_dm")]),2)
```

[1]Output will not be printed due to space constraints.

```
upper <- cortab
upper[upper.tri(cortab)] <- ""
upper <- as.data.frame(upper)
upper
```

References

Bates, D., Maechler, M., Bolker, B., & Walker, S. (2022). *Lme4: Linear Mixed-Effects Models using 'Eigen' and S4* (Version 1.1-31) [Computer software]. https://cran.r-project.org/web/packages/lme4

Kassambara, A. (2022). *Ggpubr: 'ggplot2' Based Publication Ready Plots* (Version 0.5.0) [Computer software]. https://cran.r-project.org/web/packages/ggpubr

Lüdecke, D., Giné-Vázquez, I., & Bartel, A. (2021). *Sjmisc: Data and Variable Transformation Functions* (Version 2.8.9) [Computer software]. https://cran.r-project.org/web/packages/sjmisc

Lüdecke, D., Makowski, D., Ben-Shachar, M. S., Patil, I., & Wiernik, B. M. (2022). *Easystats: Framework for Easy Statistical Modeling, Visualization, and Reporting* (Version 0.6.0) [Computer software]. https://cran.r-project.org/web/packages/easystats

Nieuwenhuis, R., Pelzer, B., & Te Grotenhuis, M. (2017). *Influence.ME: Tools for Detecting Influential Data in Mixed Effects Models* (Version 0.9-9) [Computer software].

Rinker, T., & Kurkiewicz, D. (2019). *Pacman: Package Management Tool* (Version 0.5.1) [Computer software]. https://cran.r-project.org/web/packages/pacman

Schmidt-Catran, A. W., Fairbrother, M., & Andreß, H.-J. (2019). Multilevel Models for the Analysis of Comparative Survey Data: Common Problems and Some Solutions. *Kölner Zeitschrift Für Soziologie Und Sozialpsychologie*, *71*(S1), 99–128. https://doi.org/10.1007/s11577-019-00607-9

Wickham, H. (2022a). *Haven: Import and Export 'SPSS', 'Stata' and 'SAS' Files* (Version 2.5.1) [Computer software]. https://cran.r-project.org/web/packages/haven

Wickham, H. (2022b). *Tidyverse: Easily Install and Load the "Tidyverse"* (Version 1.3.2) [Computer software]. https://cran.r-project.org/web/packages/tidyverse

Wickham, H., Chang, W., Henry, L., Pedersen, T. L., Takahashi, K., Wilke, C., Woo, K., Yutani, H., & Dunnington, D. (2022). *Ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics* (Version 3.4.0) [Computer software]. https://cran.r-project.org/web/packages/ggplot2