



The effect of socio-economic indicators on the fertility rates

An empirical analysis of fertility rates among the regions of Europe

Marcell Granát

February 21, 2021

Contents

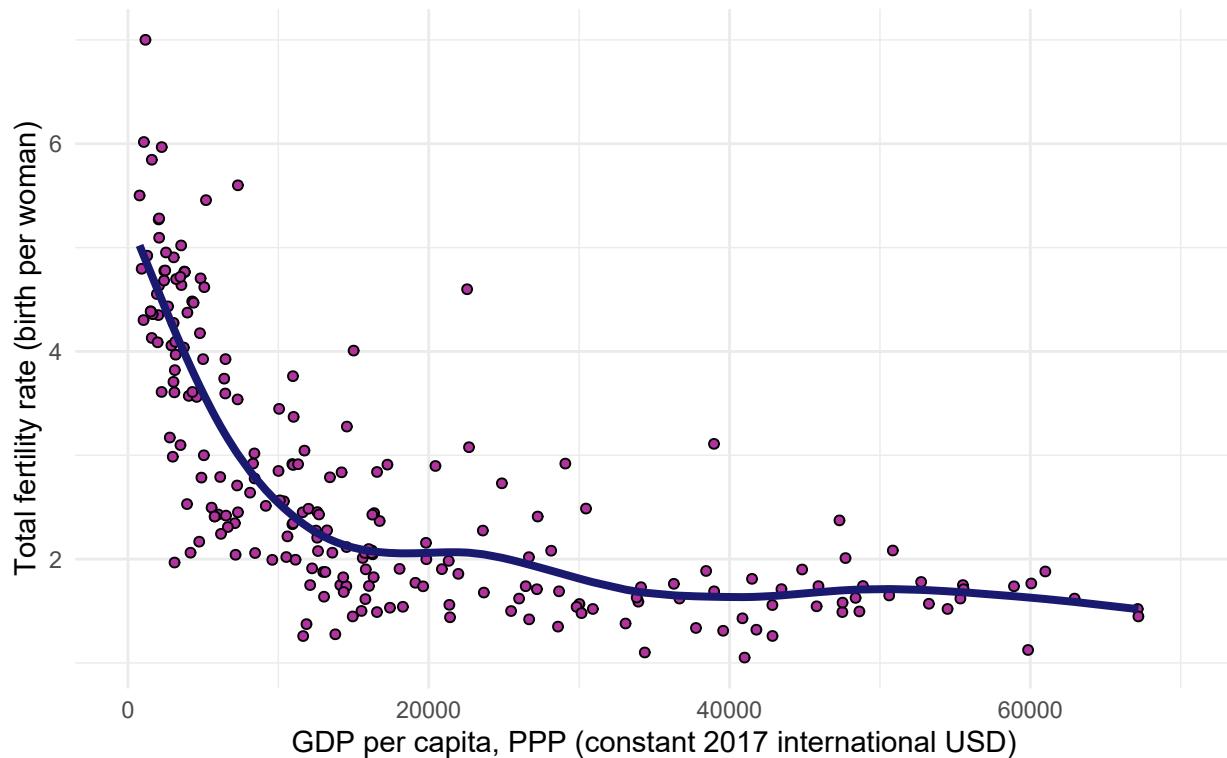
| | |
|--|-----------|
| Introduction | 3 |
| Data | 3 |
| Total fertility rates | 3 |
| Human development | 3 |
| A decent standard of living | 4 |
| Long and healthy life | 4 |
| Knowledge | 4 |
| Family benefits | 4 |
| Youth unemployment | 4 |
| Explore data | 4 |
| Model building | 4 |
| Framework I: with unemployment | 15 |
| Framework II: without unemployment | 17 |
| Appendix: R codes | 22 |

Abstract

Here is the abstract.

Introduction

Ay alábbi rövid kézirat a brit **Family Expenditure Survey** adataiból kinyert fogyasztási jellemzőket tárgyalja. Az empirikus elemzés egyszerű lineáris regresszió alkalmazásával készül. Az adatok összesen 1519 család jövedelmét és kiadását tartalmazzák, továbbá 6 termékcsoport (alkohol, ruházkodás, nem alkoholtartalmú élelmiszer, fűtés, közlekedés, egyéb) fogyasztáson belüli részarányát. Dolgozatom során én kizárolag az egy gyermekek családokra szűkítem az elemzést¹.



Own editing based on the Figure 5-2. from Kreiszné Hudák (2019).
The trend is drawn via splines.
Source of the data: World Bank.

Figure 1: Seemingly negative effect of gross domestic product on fertility rates based on nation level observations (2017)

Data

Total fertility rates

Human development

Table 1: Indicators of similarity between the Human Development Indices provided by UNDP and GDI

| Indicator | Value |
|----------------|--------|
| R^2 | 99.76% |
| Spearman R^2 | 99.71% |

¹Feladatkiírás által előírt megkötés.

| Indicator | Value |
|------------------------------------|-------|
| Mean absolute deviation | 0.007 |
| Mean absolute percentage deviation | 1.20% |

A decent standard of living

Table 2: Indicators of similarity between the income component of the Human Development Indices provided by GDL and the estimation based on regional GDP

| Indicator | Value |
|------------------------------------|--------|
| R^2 | 92.43% |
| Spearman R^2 | 94.54% |
| Mean absolute deviation | 0.0474 |
| Mean absolute percentage deviation | 6.00% |

Long and healthy life

Table 3: Indicators of similarity between the health component of the Human Development Indices provided by GDL and the estimation based on regional life expectancy

| Indicator | Value |
|------------------------------------|--------|
| R^2 | 98.24% |
| Spearman R^2 | 98.38% |
| Mean absolute deviation | 0.0041 |
| Mean absolute percentage deviation | 0.45% |

Knowledge

Table 4: Indicators of similarity between the knowledge component of Human Development Indices provided by UNDP and the calculated principal components using educational attainment level

| Indicator | Comp 1 | Comp 2 | Comp 3 | Comp 4 | Comp 5 | Comp 6 | Comp 7 | Comp 8 | Comp 9 |
|----------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| R^2 | 33.35% | 23.35% | 10.34% | 0.03% | 0.61% | 0.16% | 0.00% | 0.05% | 0.01% |
| Spearman R^2 | 21.15% | 23.77% | 13.54% | 0.03% | 0.62% | 0.51% | 13.29% | 0.47% | 1.74% |

Family benefits

Youth unemployment

Explore data

Model building

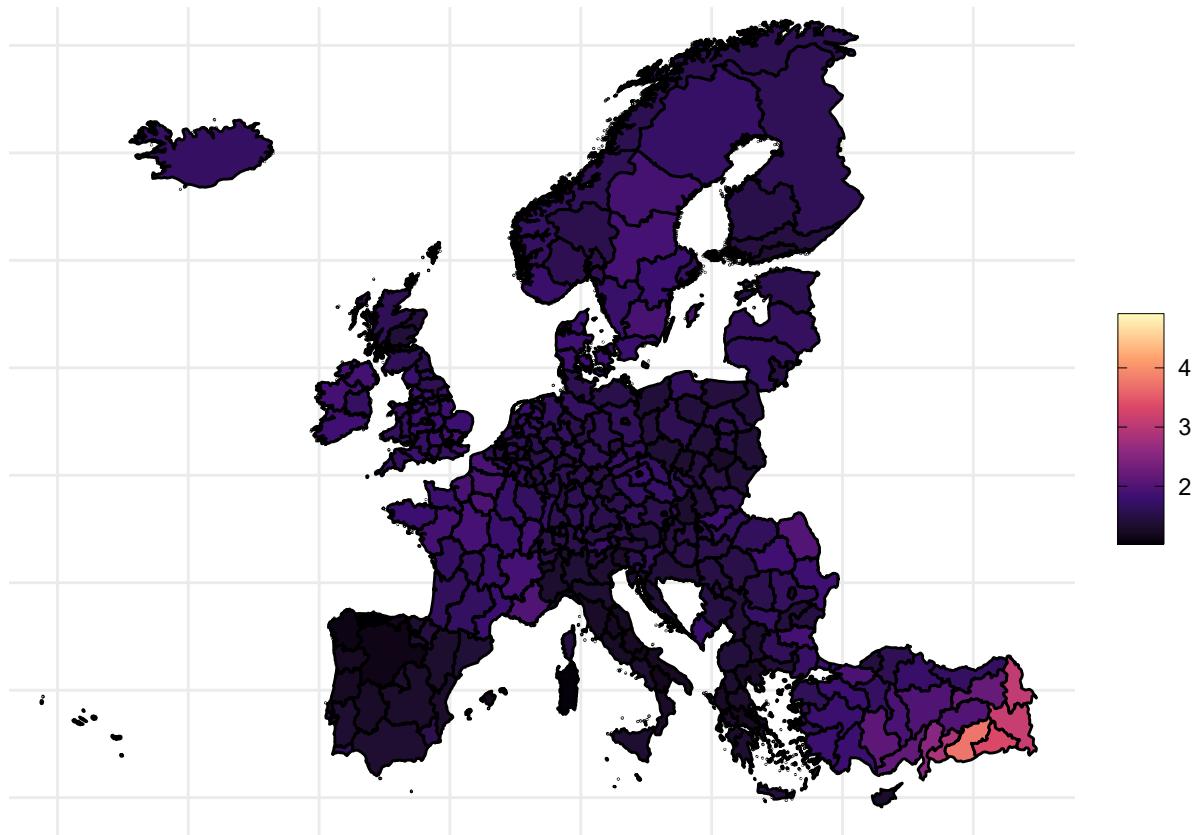


Figure 2: Total fertility rates through Europe in 2017

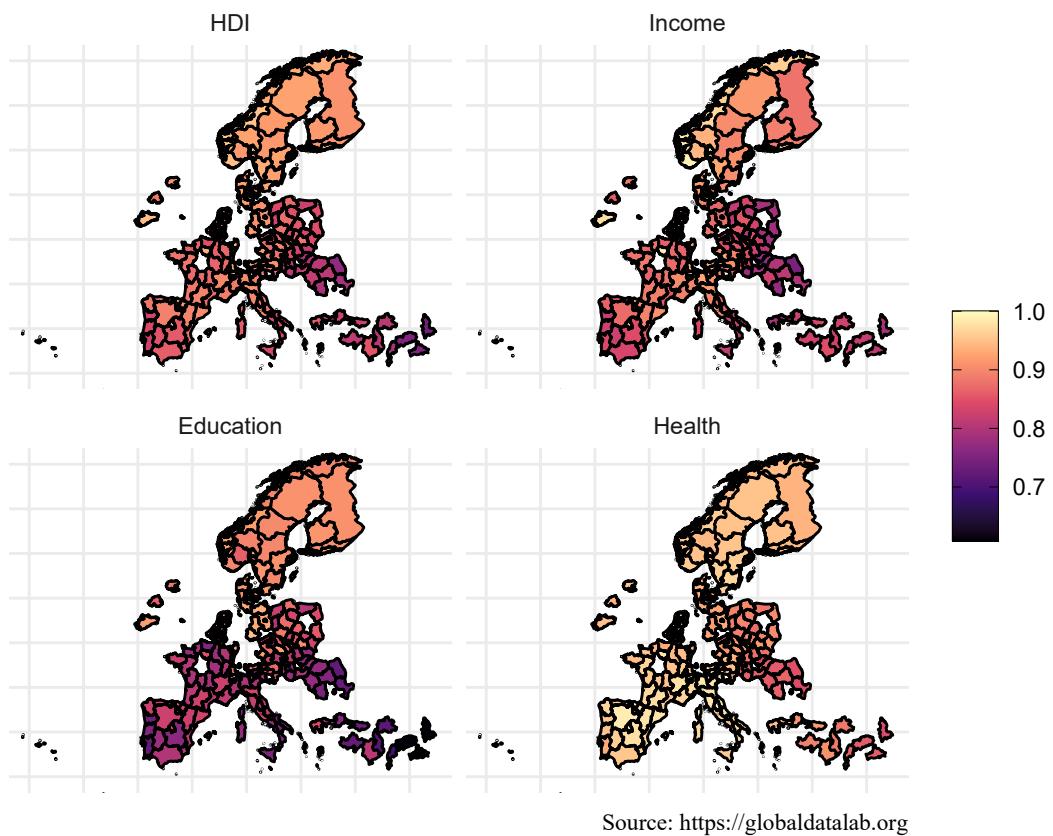


Figure 3: HDI and its components based on the dataset from Global Data Lab (2017)

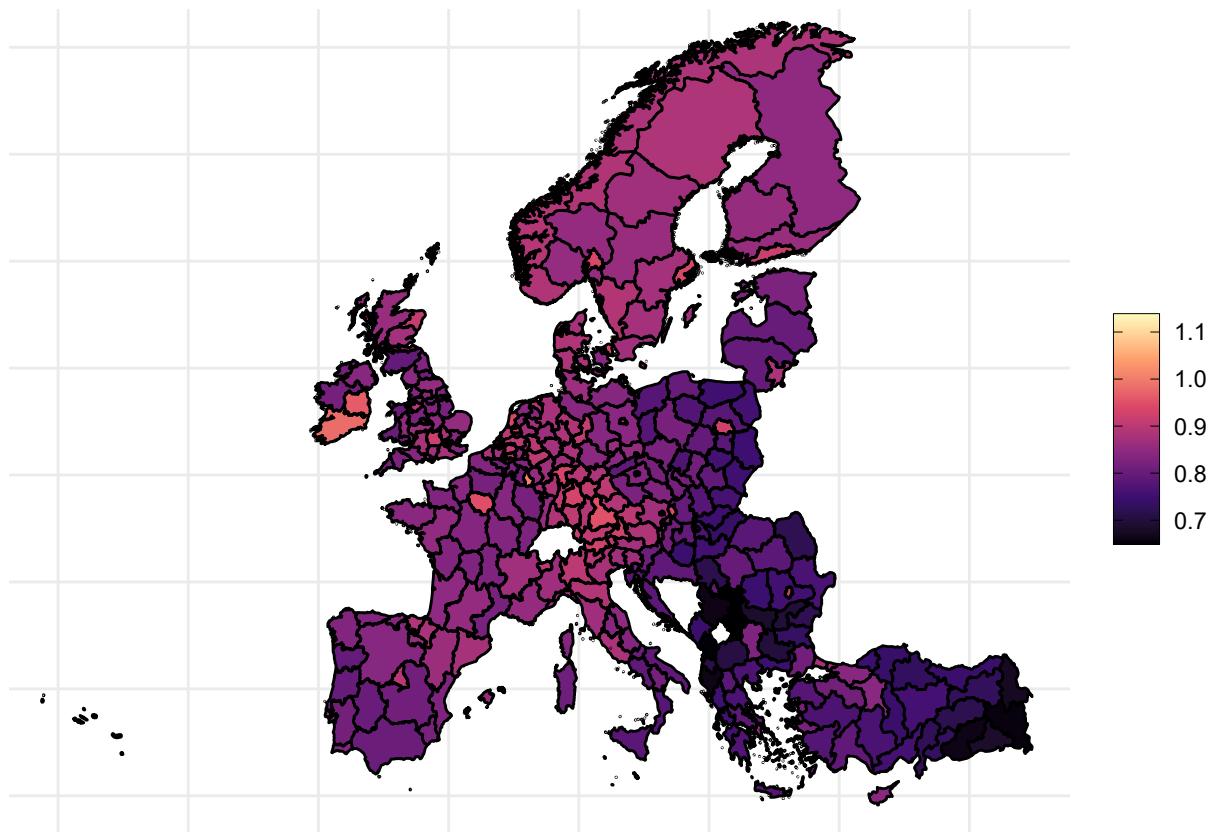


Figure 4: Calculated income index based on regional GDP data

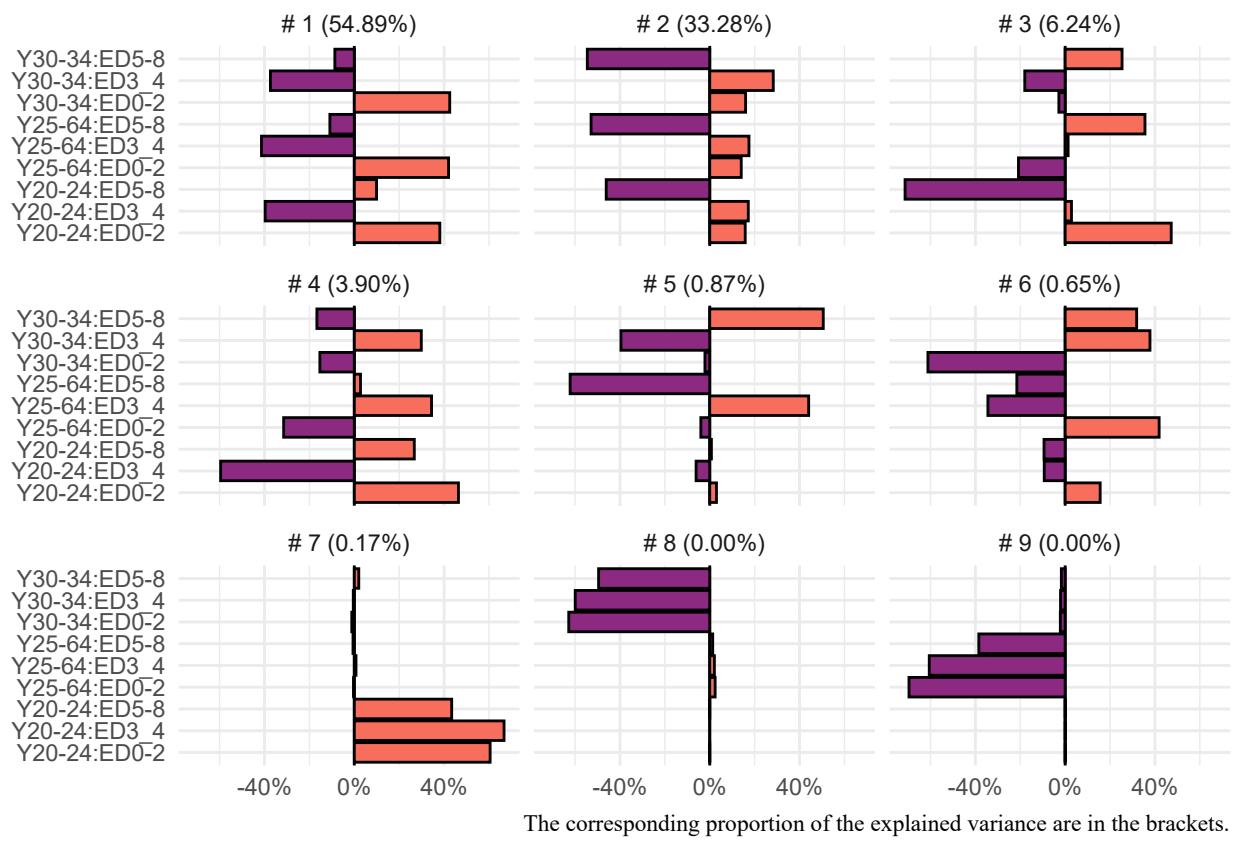


Figure 5: PCAs and the explained variance

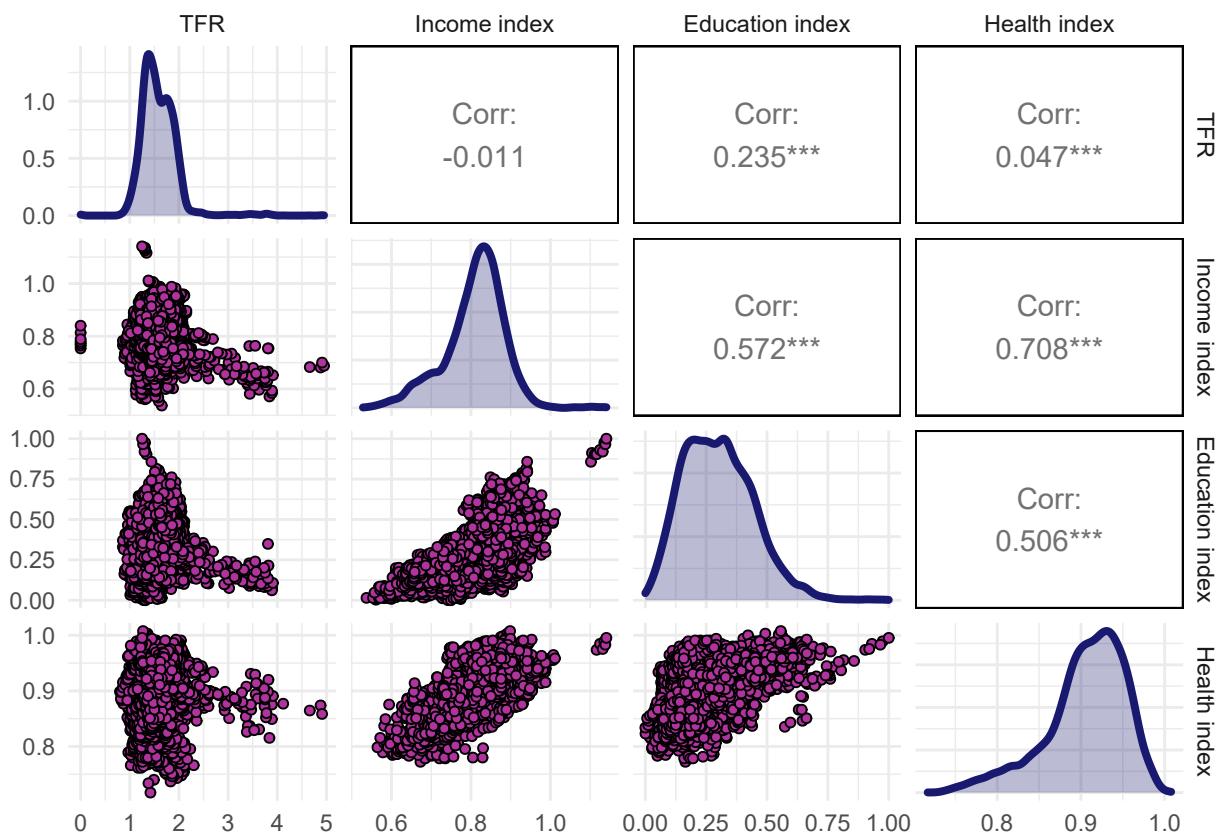


Figure 6: Pairwise correlation among TFR and calculation human development indices

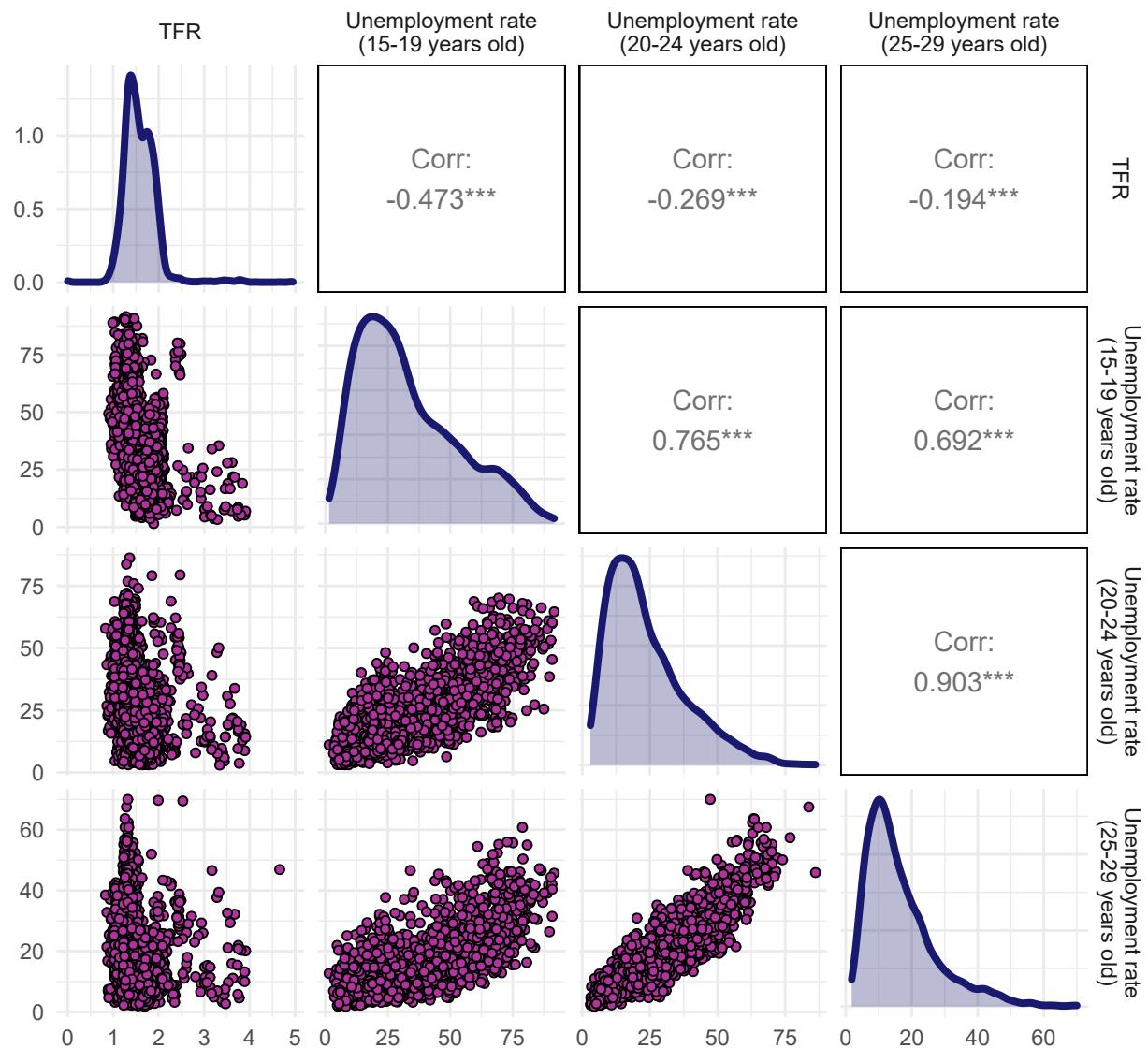


Figure 7: Pairwise correlation among TFR and youth unemployment rates

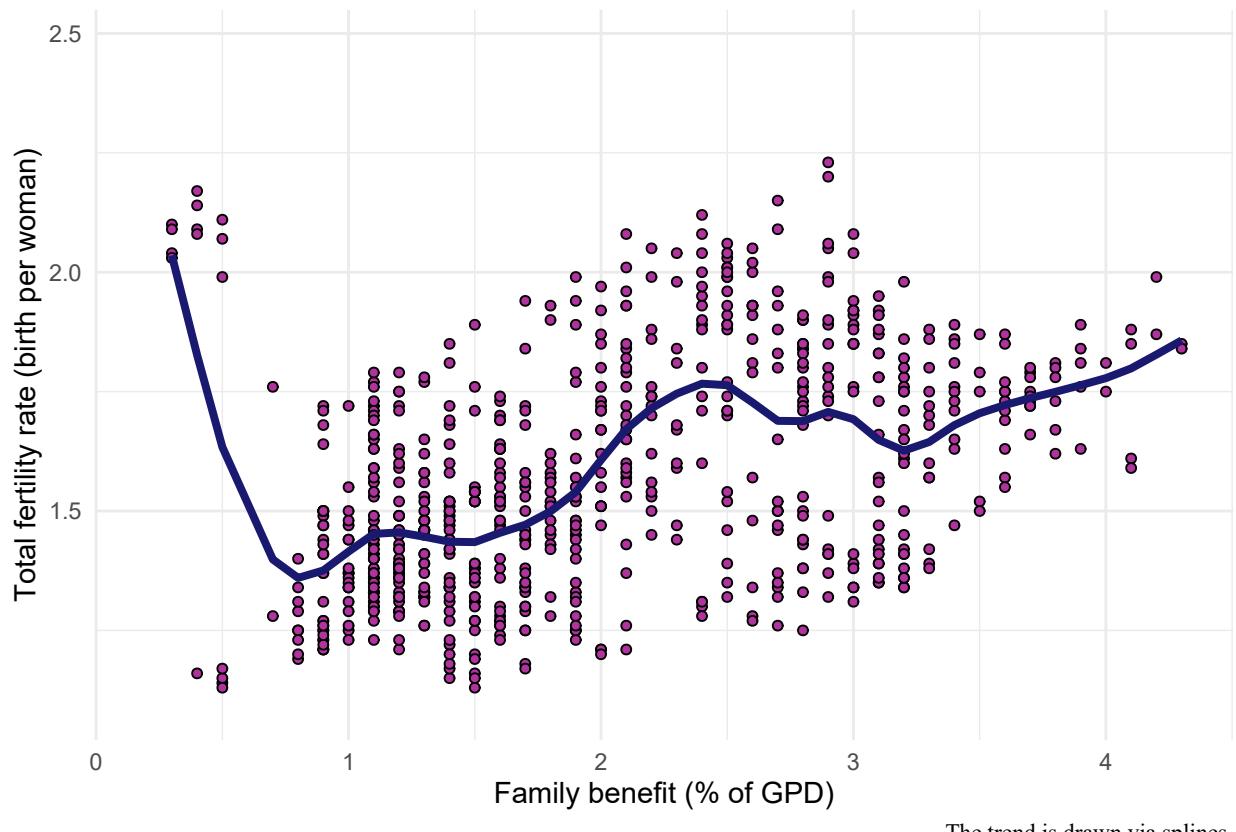


Figure 8: Correlation between TFR and family benefits

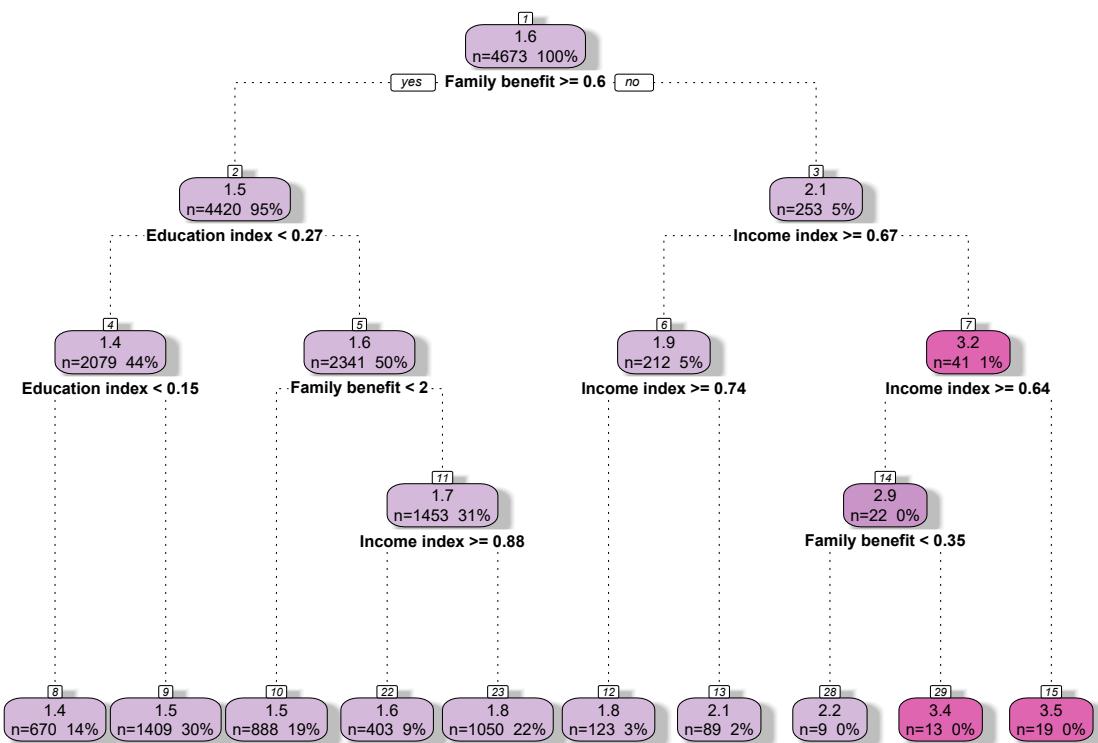


Figure 9: Regression tree explaining the TFR using only the calculated HD indices ($cp = 0.01$)

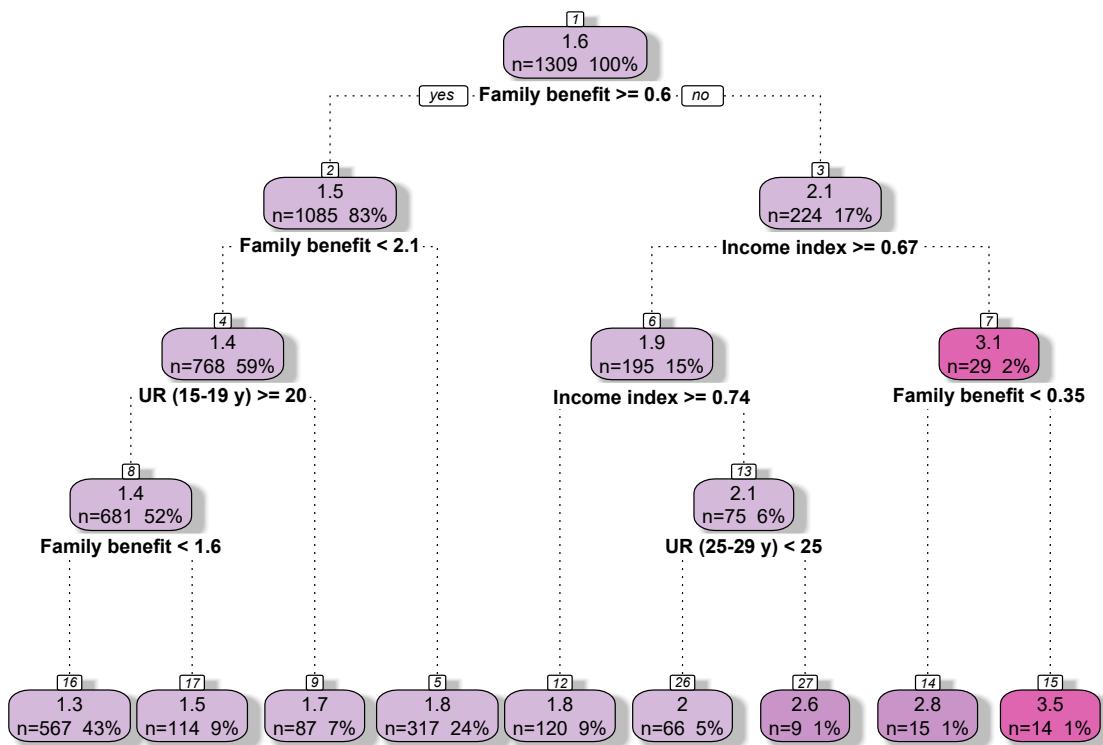


Figure 10: Regression tree explaining the TFR using all the mentioned explanatory variables ($cp = 0.01$)

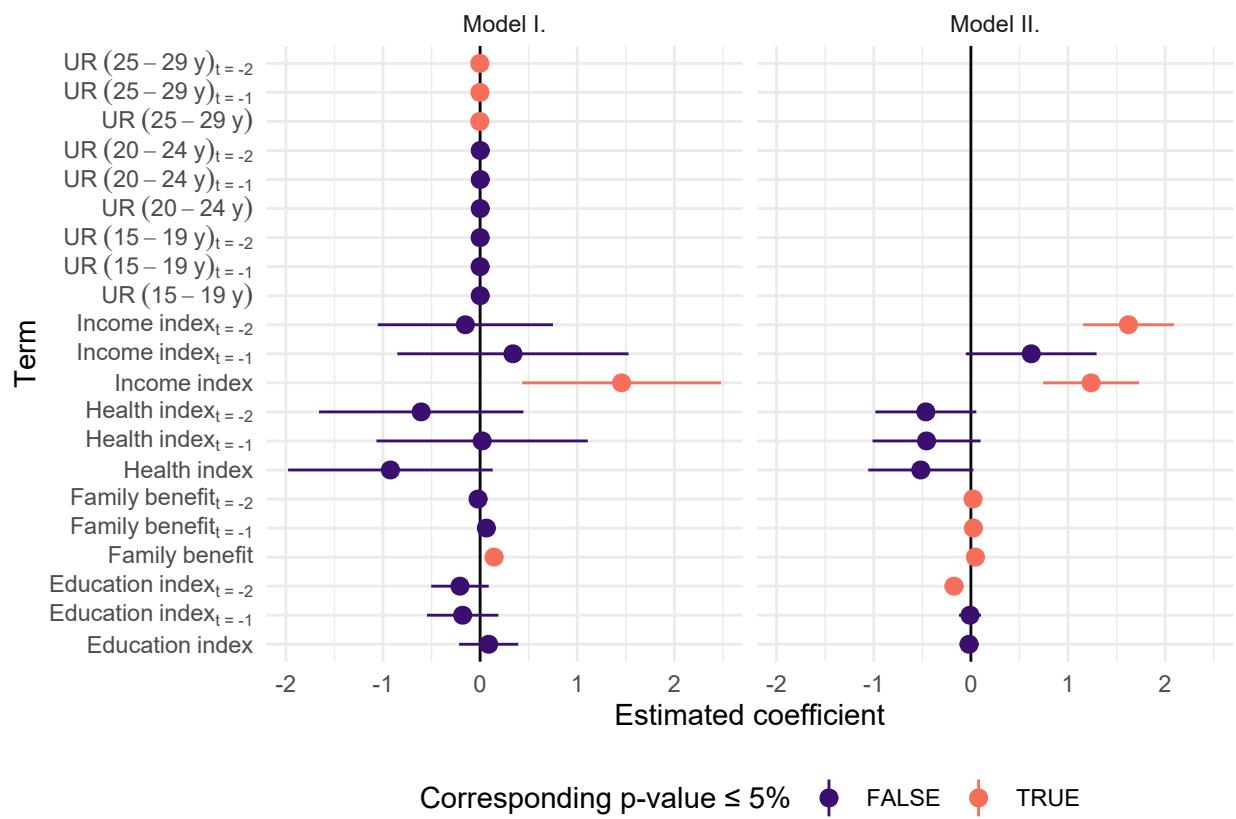


Figure 11: Panel models on the total fertility rates

Table 5: Models

| Indicator | Model I. | Model II. |
|----------------|----------|-----------|
| Pooltest | 0.00% | 0.00% |
| Phtest | 0.00% | 0.00% |
| Adjusted R^2 | 11.07% | 20.06% |
| Observations | 882 | 4020 |

Framework I: with unemployment

Table 6: T-tests

| Variable | Mean in total sample | Mean in used sample | Number of observations in the total sample | T-statistic | P-value |
|-----------------|----------------------|---------------------|--|-------------|---------|
| TFR | 1.5799 | 1.5928 | 7246 | -0.9650 | 33.48% |
| Education index | 0.2939 | 0.3131 | 5375 | -3.6811 | 0.02% |
| Health index | 0.9069 | 0.9294 | 6971 | -16.3615 | 0.00% |
| Income index | 0.8125 | 0.8197 | 4962 | -3.1597 | 0.16% |
| Family benefit | 2.0073 | 1.4374 | 6417 | 18.7511 | 0.00% |
| UR (15-19 y) | 34.1416 | 38.8821 | 1891 | -5.5185 | 0.00% |
| UR (20-24 y) | 24.2588 | 24.9203 | 3088 | -1.2496 | 21.16% |
| UR (25-29 y) | 16.9295 | 16.7099 | 2809 | 0.5654 | 57.19% |

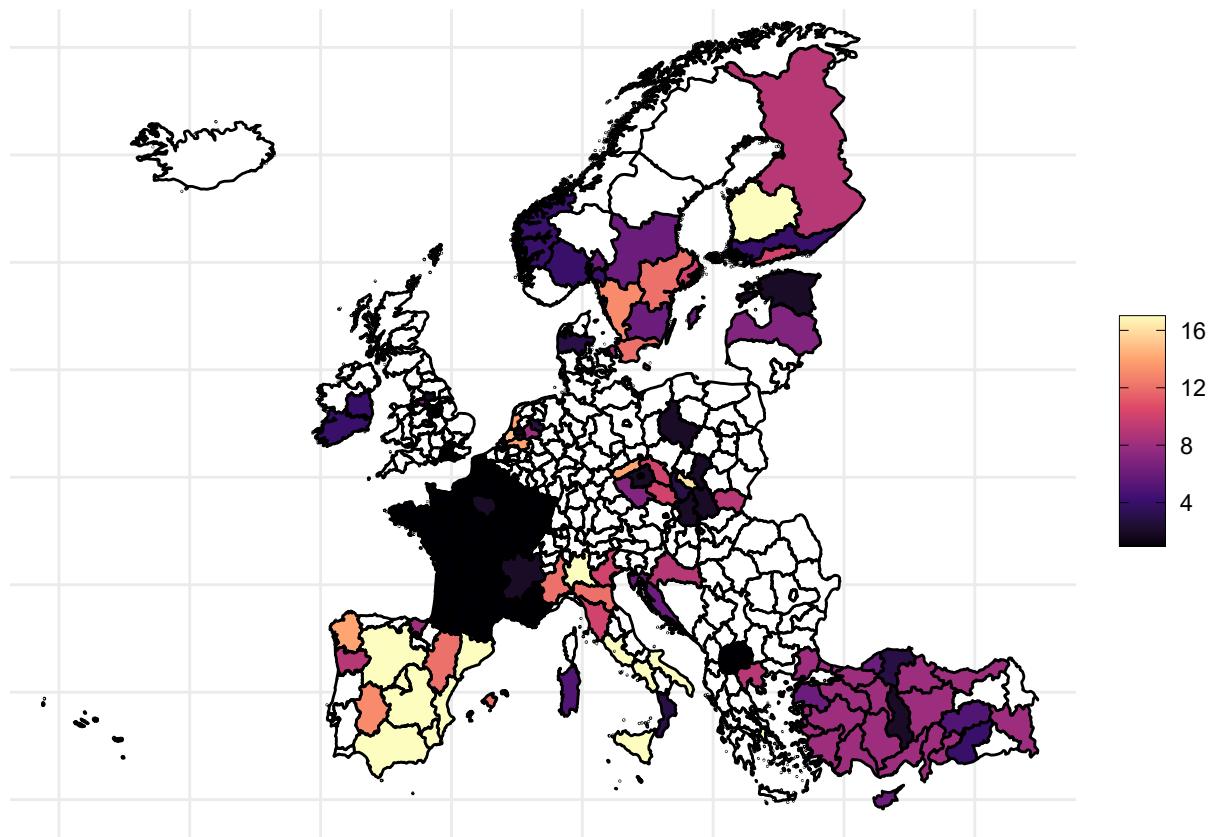
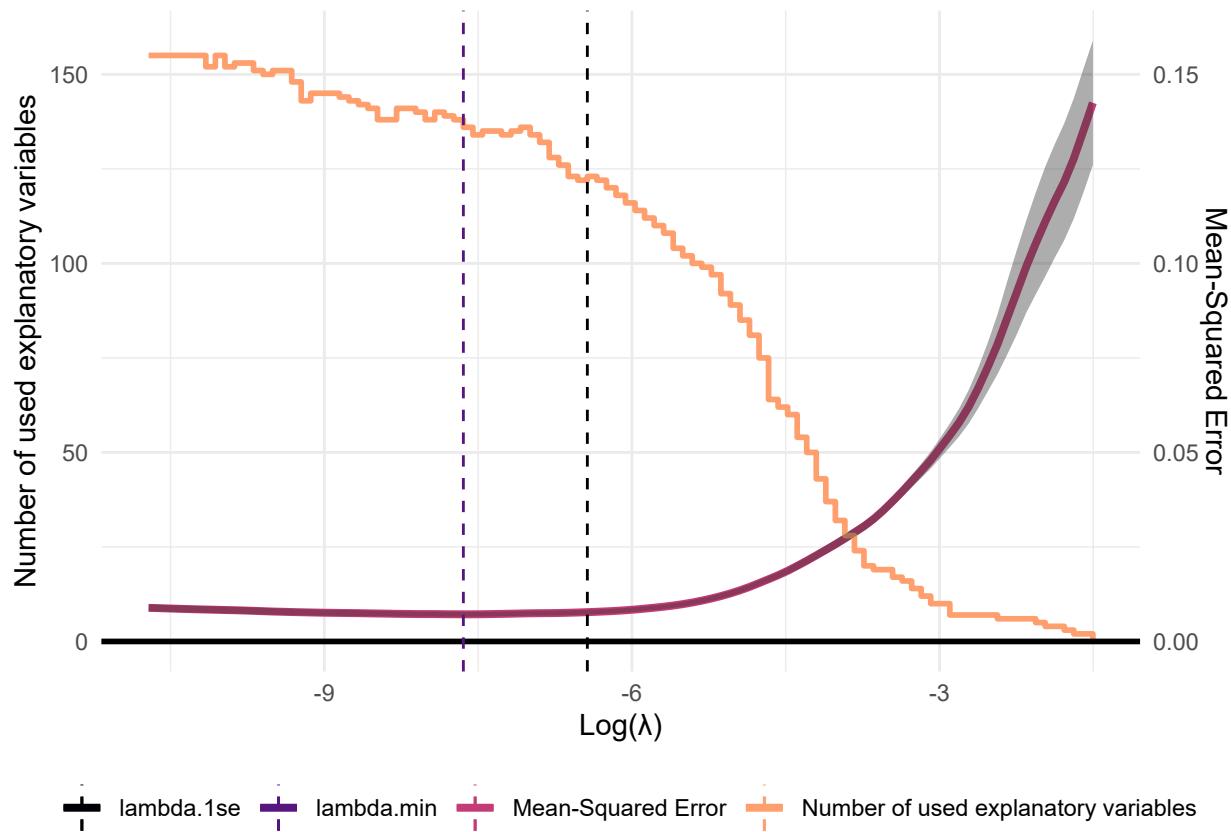


Figure 12: Number of used observations by countries when the model contains youth unemployment



Framework II: without unemployment

Table 7: T-tests

| Variable | Mean in total sample | Mean in used sample | Number of observations in total sample | T-statistic | P-value |
|-----------------|----------------------|---------------------|--|-------------|---------|
| TFR | 1.5799 | 1.5694 | 7246 | 1.6855 | 9.19% |
| Education index | 0.2939 | 0.2982 | 5375 | -1.4818 | 13.84% |
| Health index | 0.9069 | 0.9202 | 6971 | -16.0011 | 0.00% |
| Income index | 0.8125 | 0.8239 | 4962 | -8.2285 | 0.00% |
| Family benefit | 2.0073 | 2.0029 | 6417 | 0.2365 | 81.31% |

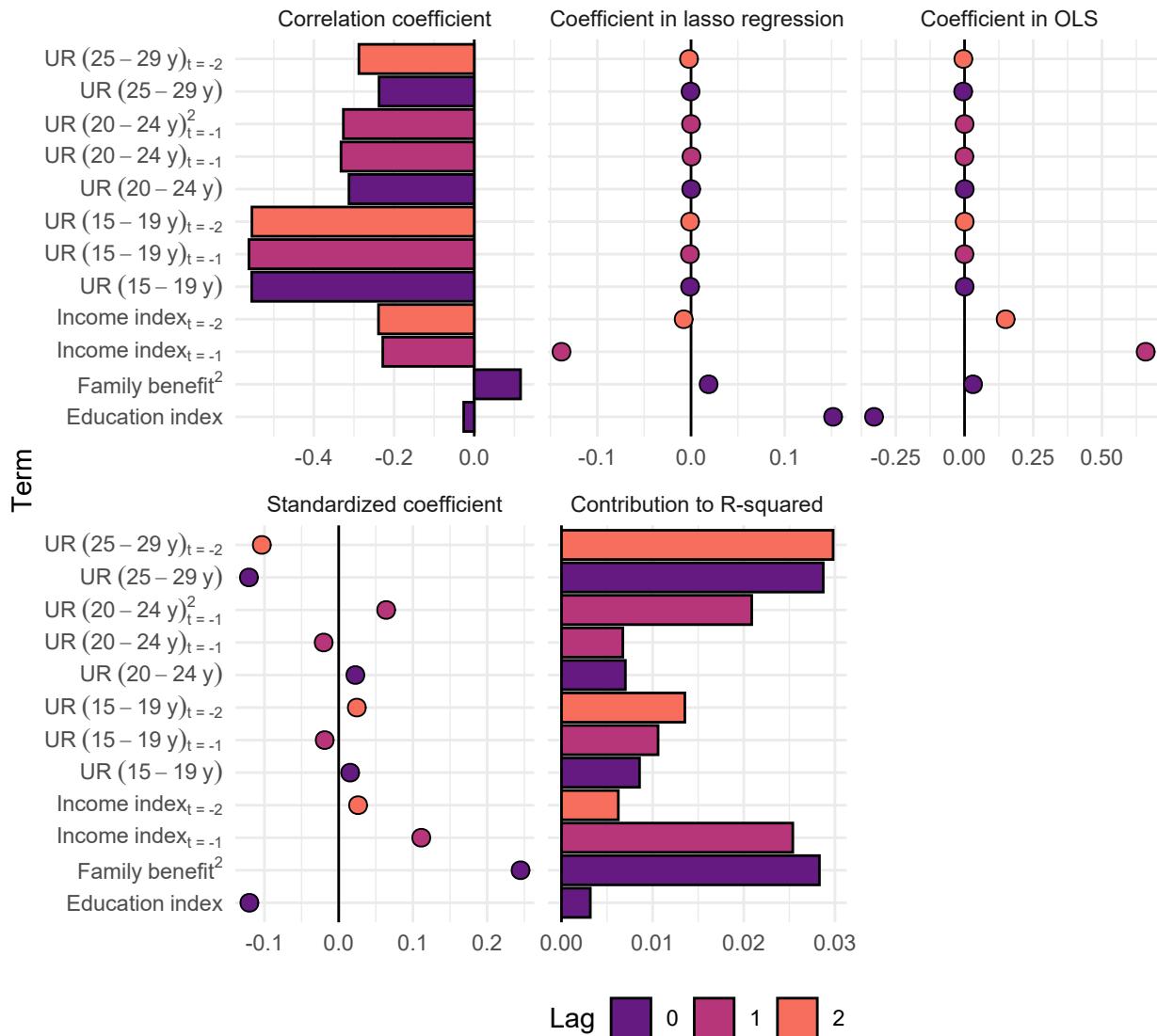


Figure 13: Estimated coefficient of the fixed panel model controlling for youth unemployment indicators

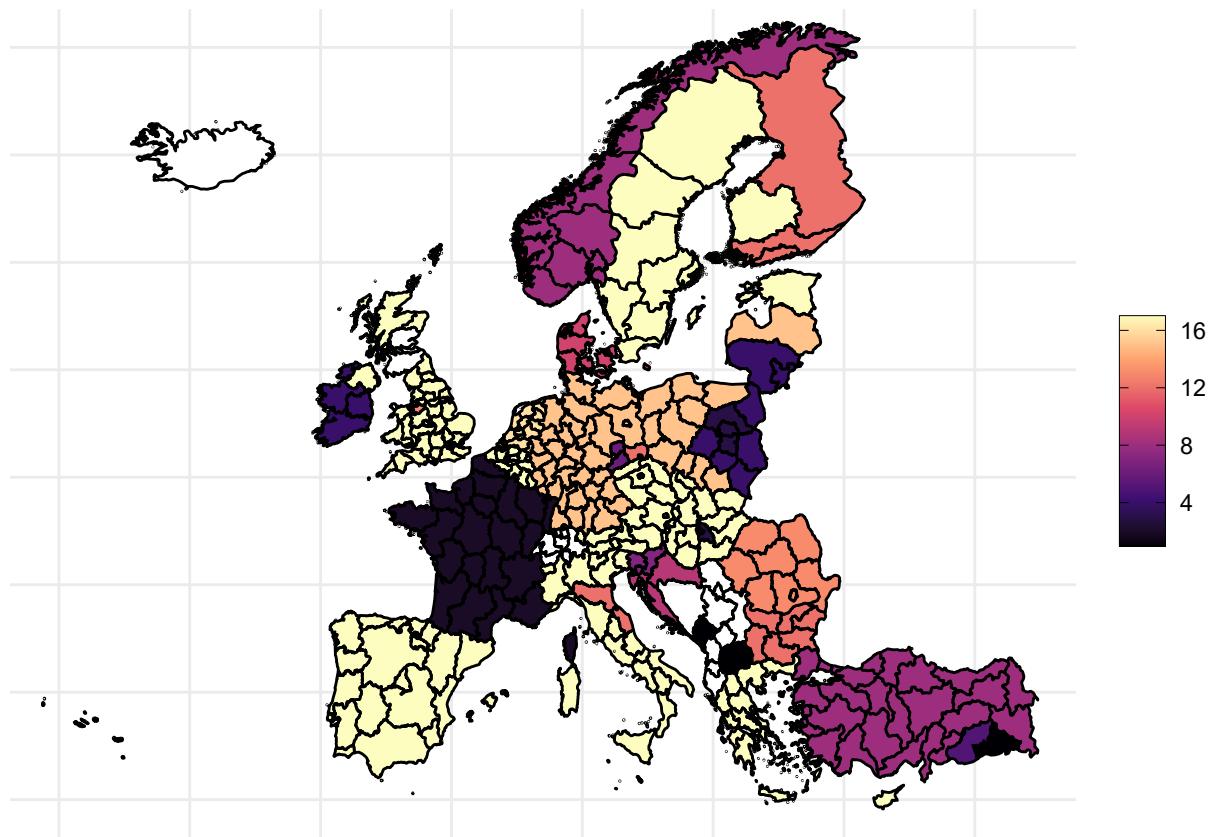


Figure 14: Number of used observations by countries when the model does not contain youth unemployment

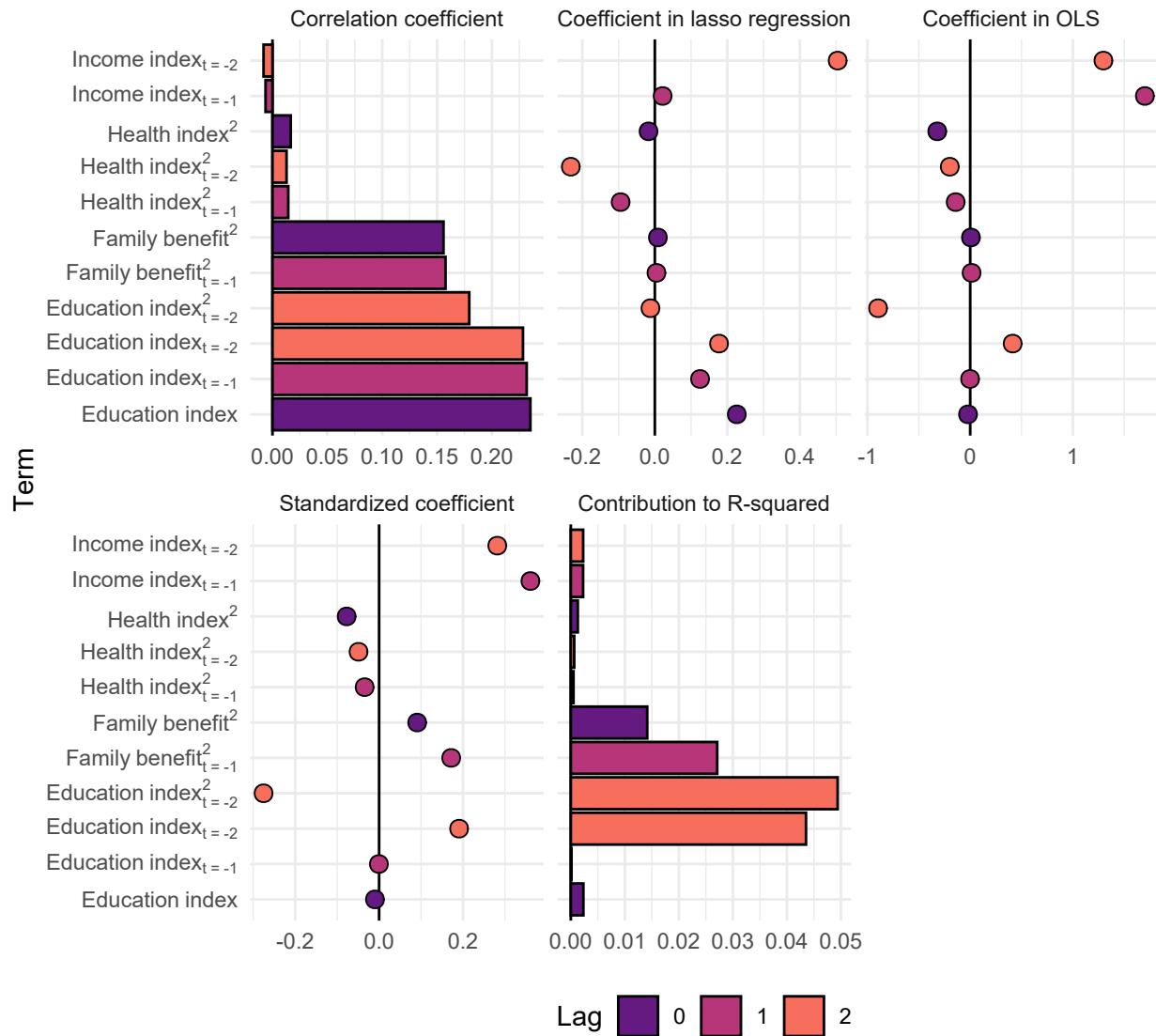


Figure 15: Estimated coefficient of the fixed panel model omitting youth unemployment indicators

References

UNITED NATIONS DEVELOPMENT PROGRAMME (2020), ‘Technical notes: Calculating the human development indices’.

URL: <http://hdr.undp.org/en/content/calculating-indices>

Appendix: R codes

```

1 # Set up -----
2
3 ## Packages =====
4
5 library(tidyverse)
6 library(patchwork)
7 library(knitr)
8 library(broom)
9 library(eurostat)
10
11 ## Gg theme =====
12
13 update_geom_defaults("point", list(fill = "#B1339E",
14                               shape = 21,
15                               color = "black",
16                               size = 1.4))
17 update_geom_defaults("line",
18                      list(color = "midnightblue", size = 1.4))
19
20 update_geom_defaults("smooth", list(color = "red4", size = 1.4))
21
22 update_geom_defaults("density",
23                      list(color = "midnightblue", fill = "midnightblue", alpha = .3,
24                           size = 1.4))
25
26 extrafont::loadfonts(device="win")
27
28 theme_set(theme_minimal() + theme(
29   legend.direction = "vertical",
30   # text = element_text(family = "Impact"),
31   plot.caption = element_text(family = "serif")
32 ))
33
34 # https://data.worldbank.org/indicator/SP.DYN.TFRT.IN
35
36 WB_fertility <- read_csv("WB_fertility.csv", skip = 4)
37
38 # https://data.worldbank.org/indicator/NY.GDP.PCAP.PP.KD
39
40 WB_GDP <- read_csv("WB_GDP.csv", skip = 4)
41
42 merge(WB_fertility %>%
43       select('Country Name', '2017') %>%
44       rename(tfr = '2017'),
45       WB_GDP %>%
46       select('Country Name', '2017') %>%
47       rename(GDP = '2017')) %>%
48 ggplot(aes(GDP, tfr)) + geom_point() +
49 ggformula::geom_spline() +
50 scale_x_continuous(limits = c(0, 7e+4)) +
51 labs(y = "Total fertility rate (birth per woman)",
52      x = "GDP per capita, PPP (constant 2017 international USD)",
```

```

53     caption = "Own editing based on the Figure 5-2. from Kreiszné Hudák (2019).  

54     The trend is drawn via splines.  

55     Source of the data: World Bank."  

56   )  

57  

58 plot_NUTS2 <- function(df, viridis_c = T, ..., all.x = F) {  

59   p <- df %>%  

60   {merge(eurostat::get_eurostat_geospatial(nuts_level = 2), ., all.x = all.x)} %>%  

61   ggplot(aes(fill = values)) +  

62   geom_sf(color = "black") +  

63   theme(  

64     axis.text = element_blank()  

65   ) +  

66   xlim(c(-30, 44)) +  

67   ylim(c(35, 70)) +  

68   labs(fill = NULL)  

69  

70   if (viridis_c) {  

71     p <- p + scale_fill_viridis_c(option = "magma", ...,  

72                                   guide = guide_colorbar(frame.colour = "black",  

73                                   ticks.colour = "black"),  

74                                   na.value = "white")  

75   }  

76   p  

77 }  

78  

79 # Data import -----  

80  

81 # Fertility: source: Eurostat database =====  

82  

83 f_data <- get_eurostat("demo_r_find2", time_format = "num") %>%  

84   select(geo, time, var = indic_de, values)  

85  

86 ##### Map of TFR #####  

87  

88 f_data %>%  

89   filter(var == "TOTFERRT" & str_length(geo) == 4 & time == 2017) %>%  

90   plot_NUTS2()  

91  

92 # Data from Global Data Labor -----  

93  

94 ##### Sub-national data #####  

95  

96 # source of csv files: https://globaldatalab.org/  

97  

98 GDL_import <- function(x) {  

99   get_eurostat_geospatial(nuts_level = 2) %>%  

100  data.frame() %>%  

101  tibble %>%  

102  mutate(  

103    ISO_Code = countrycode::countrycode(CNTR_CODE, origin = "iso2c", "iso3c"),  

104    ISO_Code = ifelse(CNTR_CODE == "UK", "GBR", ISO_Code),  

105    ISO_Code = ifelse(CNTR_CODE == "EL", "GRC", ISO_Code),

```

```

106 ) %>%
107   select(ISO_Code, NUTS_NAME, geo) %>%
108   merge(read_csv(x), by = "ISO_Code") %>%
109   mutate(
110     z = stringdist::stringsim(NUTS_NAME, Region)
111   ) %>%
112   arrange(desc(z)) %>%
113   filter(!duplicated(Region)) %>%
114   filter(!duplicated(NUTS_NAME)) %>%
115   filter((z > .5 | Country %in% c("Greece", "Turkey", 'Romania',
116                                     'Malta', 'Italy')) & NUTS_NAME != "Dresden") %>%
117   select(geo, '1990':'2018') %>%
118   pivot_longer(-1, names_to = "time", values_to = "values") %>%
119   mutate(time = as.numeric(time))
120 }
121
122 GDL_subnat <- GDL_import("GDL-Sub-national-HDI-data.csv") %>% rename(HDI = values) %>%
123   merge(
124     GDL_import("GDL-Educational-index--data.csv") %>% rename(education = values)
125   ) %>%
126   merge(
127     GDL_import("GDL-Health-index-data.csv") %>% rename(health = values)
128   ) %>%
129   merge(
130     GDL_import("GDL-Income-index-data.csv") %>% rename(income = values)
131   )
132
133 ### National data #####
134
135 GDL_nat <- read_csv("GDL-Sub-national-HDI-data.csv") %>%
136   filter(Level == "National") %>%
137   select(Country, 6:34) %>% pivot_longer(-1, names_to = "time", values_to = "values") %>%
138   mutate(time = as.numeric(time)) %>%
139   na.omit()
140
141 # Data from UNDP =====
142
143 # source: http://hdr.undp.org/
144
145 HDI_UNDP <- read_csv("Human Development Index (HDI).csv",
146                       skip = 5) %>%
147   select(!starts_with("X"), - 'HDI Rank') %>%
148   mutate_at(-1,
149             function(x) {as.numeric(ifelse(x == "...", NA, x))}) %>%
150   pivot_longer(-1, names_to = "time", values_to = "values") %>%
151   mutate(time = as.numeric(time)) %>%
152   na.omit()
153
154 merge(GDL_nat %>% rename(GDL = values),
155       HDI_UNDP %>% rename(UNDP = values)) %>%
156   {
157     c(
158       scales::percent(cor(x = .\$GDL, y = .\$UNDP)^2, accuracy = .01),

```

```

159     scales::percent(cor(x = .\$GDL, y = .\$UNDP, method = "spearman")^2, accuracy = .01),
160     as.character(format(mean(abs(.\$GDL - .\$UNDP)), digits = 1)),
161     scales::percent(mean(abs(.\$GDL - .\$UNDP) / .\$UNDP), accuracy = .01)
162   )
163 } %>%
164 {tibble(
165   Indicator = c("R^2", "Spearman R^2",
166                 "Mean absolute deviation", "Mean absolute percentage deviation"),
167   Value = .
168 )} %>%
169 kable(
170   caption = "Indicators of similarity between the Human Development Indices
171   provided by UNDP and GDL"
172 )
173
174 GDL_subnat %>%
175   filter(time == 2017) %>%
176   select(-time) %>%
177   pivot_longer(-1, names_to = "var", values_to = "values") %>%
178   filter(!is.na(var)) %>%
179   mutate(
180     var = str_to_title(var),
181     var = str_replace(var, "Hdi", "HDI"),
182     var = factor(var,
183                   levels = c("HDI", "Income", "Education", "Health"),
184                   ordered = T)
185   ) %>%
186   plot_NUTS2() + facet_wrap(~ var, ncol = 2) +
187   labs(caption = "Source: https://globaldatalab.org")
188
189 GDP_index <- get_eurostat("nama_10r_2gdp", time_format = "num") %>%
190   filter(unit == "PPS_HAB") %>% # Purchasing power standard (PPS) per inhabitant
191   select(-unit) %>%
192   rename(GDP = values) %>% merge(
193     get_eurostat("ert_bil_eur_a", time_format = "num") %>% # EUR/USD annual avg exc r
194       filter(currency == "USD" & statinfo == "AVG") %>%
195       select(time, e = values)
196   ) %>%
197   {
198     GDP <- .\$GDP / .\$e # mutate to USD
199     mutate(.,
200       GDPindex = (log(GDP) - log(100)) /
201         (log(75000) - log(100)))
202   }
203 }
204
205 merge(GDL_subnat, GDP_index) %>%
206   {
207     c(
208       scales::percent(cor(x = .\$income, y = .\$GDPindex)^2, accuracy = .01),
209       scales::percent(cor(x = .\$income, y = .\$GDPindex, method = "spearman")^2,
210                     accuracy = .01),
211       as.character(format(mean(abs(.\$income - .\$GDPindex)), digits = 1, nsmall = 4)),

```

```

212     scales::percent(mean(abs(.income - .GDPindex) / .GDPindex), accuracy = .01)
213   )
214 } %>%
215 {tibble(
216   Indicator = c("$R^2$",
217                 "Spearman $R^2$",
218                 "Mean absolute deviation",
219                 "Mean absolute percentage deviation"),
220   Value = .
221 )} %>%
222 kable(
223   caption = "Indicators of similiarity between the income component of the
224   Human Development Indices provided by GDL and the estimation based on regional GDP"
225 )
226
227 GDP_index %>%
228   filter(time == 2017) %>%
229   select(geo, values = GDPindex) %>%
230   plot_NUTS2
231
232 health_index <- get_eurostat("demo_r_mlifexp", time_format = "num") %>%
233   filter(age == "Y_LT1" & sex == "T") %>%
234   select(geo, time, le = values) %>%
235   mutate(
236     health_index = (le - 20) / (85 - 20)
237   )
238
239 merge(GDL_subnat, health_index) %>%
240 {
241   c(
242     scales::percent(cor(x = .health, y = .health_index)^2, accuracy = .01),
243     scales::percent(cor(x = .health, y = .health_index, method = "spearman")^2,
244                     accuracy = .01),
245     as.character(format(mean(abs(.health - .health_index)), digits = 1, nsmall = 4)),
246     scales::percent(mean(abs(.health - .health_index)) / .health_index, accuracy = .01)
247   )
248 } %>%
249 {tibble(
250   Indicator = c("$R^2$",
251                 "Spearman $R^2$",
252                 "Mean absolute deviation",
253                 "Mean absolute percentage deviation"),
254   Value = .
255 )} %%%
256 kable(
257   caption = "Indicators of similiarity between the health component of the
258   Human Development Indices provided by GDL and the estimation based on regional life
259   expectancy"
260 )
261
262 edu_wide <- get_eurostat("edat_lfse_04", time_format = "num") %>%
263   filter(sex == "T" & !str_detect(isced11, "GEN") &
264         !str_detect(isced11, "VOC") & isced11 != "ED3-8"
265   ) %>%
266   mutate(
267     var = str_c(age, ":", isced11)
268   ) %>%

```

```

265 select(geo, time, var, values) %>%
266 pivot_wider(names_from = var, values_from = values) %>%
267 {
268   x <- .
269   names(x) <- letters[1:length(x)]
270   x <- cbind(
271     .[, 1:2],
272     mice::complete(mice::mice(select(x, -a,-b), printFlag = F))
273   )
274   names(x) <- names(.)
275   x
276 }
277
278 edu_comps <- edu_wide%>%
279   select(-time, -geo) %>%
280   na.omit() %>%
281   {princomp(scale(.))}
282
283 edu_comp_vars <- edu_comps %>%
284   summary() %>%
285   {$.sdev^2/sum($.sdev^2)} %>%
286   scales::percent(accuracy = .01) %>%
287   {str_c("# ", 1:length(.), " (", ., ")")}
288
289 edu_comps %>%
290   .$loadings %>%
291   unclass() %>%
292   data.frame() %>%
293   rownames_to_column() %>%
294   pivot_longer(-1) %>%
295   mutate(
296     name = as.numeric(str_remove(name, 'Comp.')),
297   ) %>%
298   arrange(name) %>%
299   mutate(
300     name = edu_comps %>%
301       summary() %>%
302       {$.sdev^2/sum($.sdev^2)} %>%
303       scales::percent(accuracy = .01) %>%
304       {str_c("# ", 1:length(.), " (", ., ")")}) %>%
305     .[name]
306   ) %>%
307   ggplot +
308   aes(rowname, value, fill = value < 0) +
309   geom_hline(yintercept = 0) +
310   geom_col(color = 'black') +
311   coord_flip() +
312   scale_fill_viridis_d(guide = F, option = "magma", begin = .4,
313                         end= .7, direction = -1) +
314   scale_y_continuous(labels = scales::percent) +
315   facet_wrap(~name, ncol = 3) +
316   labs(x = NULL, y = NULL, caption =
317     "The corresponding proportion of the explained variance are in the brackets.")

```

```

318
319 edu_comps %>% .$scores %>%
320   cbind(edu_wide) %>% merge(GDL_subnat) %>%
321   select(3:11, education) %>%
322   {
323     x <- .$education
324     apply(select(., -education), 2, function(y) {
325       c(
326         scales::percent(cor(x = x, y = y)^2, accuracy = .01),
327         scales::percent(cor(x = x, y = y, method = "spearman")^2, accuracy = .01)
328       )
329     })
330   } %>%
331   data.frame() %>%
332   mutate(Indicator = c("R^2", "Spearman R^2")) %>%
333   rename_all(function(x) str_replace(x, "p.", "p ")) %>%
334   select(Indicator, 1:9) %>%
335   kable(
336     caption = "Indicators of similarity between the knowledge component of Human
337     Development Indices provided by UNDP and the calculated principal components using
338     educational attainment level"
339   )
340
341 edu_index <- edu_comps %>%
342   .$scores %>%
343   data.frame() %>%
344   select(2) %>%
345   cbind(edu_wide) %>%
346   select(geo, time, edu_index = Comp.2) %>%
347   mutate(
348     edu_index = -edu_index,
349     edu_index = edu_index + abs(min(edu_index)),
350     edu_index = edu_index/max(edu_index)
351   )
352
353 dat <- f_data %>%
354   pivot_wider(names_from = var, values_from = values) %>%
355   merge(edu_index, all = T) %>%
356   merge(health_index, all = T) %>%
357   merge(GDP_index, all = T)
358
359 FAM_df <- get_eurostat("spr_exp_sum", time_format = "num") %>%
360   filter(spdeps == "FAM" & unit == "PC_GDP") %>%
361   rename(FAM = values, country = geo) %>%
362   select(-(spdeps:unit))
363
364 yth_empl_byage <- get_eurostat("yth_empl_110", time_format = "num") %>%
365   filter(unit == "PC" & sex == "F") %>%
366   filter(age %in% c("Y15-19", "Y20-24", "Y25-29")) %>%
367   select(-unit, -sex) %>%
368   pivot_wider(names_from = age, values_from = values) %>%
369   rename(
370     "uY15" = "Y15-19",
371     "uY20" = "Y20-24",

```

```

372     "uY25" = "Y25-29"
373   )
374
375 dat <- dat %>%
376   mutate(country = str_sub(geo, end = 2)) %>%
377   merge(FAM_df, all.x = T, all.y = F) %>%
378   merge(yth_empl_byage, all.x = T, all.y = F)
379
380 f.clean_names <- function(v, Tosparse = F) {
381   v <- str_replace_all(v, "GDPindex", "Income index") %>%
382     str_replace_all("health_index", "Health index") %>%
383     str_replace_all("edu_index", "Education index") %>%
384     str_replace_all("TOTFERRT", "TFR") %>%
385     str_replace_all("GDPindex", "Income index") %>%
386     str_replace_all("FAM", "Family benefit") %>%
387     str_replace_all("uY15", "UR (15-19 y)") %>%
388     str_replace_all("uY20", "UR (20-24 y)") %>%
389     str_replace_all("uY25", "UR (25-29 y)")
390   if(Tosparse) v <- str_replace_all(v, " ", "~")
391   v
392 }
393
394 # Explore the data -----
395
396 # Pairwise correlations =====
397
398 dat %>%
399   filter(str_length(geo) == 4) %>%
400   select(TOTFERRT, GDPindex, edu_index, health_index) %>%
401   {set_names(., f.clean_names(names(.)))} %>%
402   GGally::ggpairs()
403
404 dat %>%
405   filter(str_length(geo) == 4) %>%
406   select(TOTFERRT, uY15, uY20, uY25) %>%
407   set_names("TFR", "Unemployment rate\n(15-19 years old)",
408             "Unemployment rate\n(20-24 years old)",
409             "Unemployment rate\n(25-29 years old)") %>%
410   GGally::ggpairs()
411
412 dat %>%
413   filter(str_length(geo) == 2) %>%
414   ggplot(aes(FAM, TOTFERRT)) + geom_point() +
415   ggformula::geom_spline() +
416   labs(y = "Total fertility rate (birth per woman)",
417         x = "Family benefit (% of GPD)",
418         caption = "The trend is drawn via splines."
419       )
420
421 # Regression trees =====
422
423 m_part <- dat %>%
424   filter(str_length(geo) == 4) %>%

```

```

425   select(TOTFERRT, GDPindex, edu_index, health_index, FAM) %>%
426   {set_names(., f.clean_names(names(.)))} %>%
427   na.omit() %>%
428   rpart::rpart(formula = TFR ~ ., cp = .01)
429
430 m_part %>% rattle::fancyRpartPlot(palettes = 'PuRd', sub = NULL)
431
432 m_part %>% summary()
433
434 m_part2 <- dat %>%
435   filter(str_length(geo) == 4) %>%
436   select(TOTFERRT, GDPindex, edu_index, health_index, uY15, uY20, uY25, FAM) %>%
437   {set_names(., f.clean_names(names(.)))} %>%
438   na.omit() %>%
439   rpart::rpart(formula = TFR ~ ., cp = .01)
440
441 m_part2 %>% rattle::fancyRpartPlot(palettes = 'PuRd', sub = NULL)
442
443 # Model building -----
444
445 # Transform the data for panel modeling =====
446
447 dat_plm <- dat %>%
448   select(
449     geo, time, TOTFERRT, edu_index, health_index, GDPindex, FAM, uY15, uY20, uY25
450   ) %>%
451   filter(str_length(geo) == 4 & !is.na(TOTFERRT))
452
453 dat_plm <- dat_plm %>%
454   select(-TOTFERRT) %>%
455   mutate(time = time + 1) %>%
456   {
457     set_names(., ifelse(names(.) == 'geo' | names(.) == 'time', names(.),
458                           paste0(names(.), '_1')))
459   } %>%
460   merge(dat_plm, all.x = F, all.y = T)
461
462 dat_plm <- dat_plm %>%
463   select(!ends_with("_1")) %>%
464   select(-TOTFERRT) %>%
465   mutate(time = time + 2) %>%
466   {
467     set_names(., ifelse(names(.) == 'geo' | names(.) == 'time', names(.),
468                           paste0(names(.), '_1_1')))
469   } %>%
470   merge(dat_plm, all.x = F, all.y = T)
471
472 dat_plm <- dat_plm %>%
473   select(-TOTFERRT) %>%
474   mutate_at(-(1:2), function(x) x^2) %>%
475   {
476     set_names(., ifelse(names(.) == 'geo' | names(.) == 'time', names(.),
477                           paste0(names(.), '_2')))
```

```

478 } %>%
479 merge(dat_plm, all.x = F, all.y = T)
480
481 ## Initial models =====
482
483 library(plm)
484
485 m_panels <- c(
486   'TOTFERRT ~ edu_index_l_1 + health_index_l_1 + GDPindex_l_1 + FAM_l_1 +
487   uY15_l_1 + uY20_l_1 + uY25_l_1 + edu_index_l + health_index_l + GDPindex_l + FAM_l +
488   uY15_l + uY20_l + uY25_l + edu_index + health_index + GDPindex + FAM + uY15 + uY20 +
489   uY25',
490   'TOTFERRT ~ edu_index_l_1 + health_index_l_1 + GDPindex_l_1 + FAM_l_1 + edu_index_l +
491   health_index_l + GDPindex_l + FAM_l + edu_index + health_index + GDPindex + FAM'
492 ) %>%
493 lapply(function(formula) {
494   pooling <- plm(eval(formula), data = dat_plm, model = "pooling")
495   within <- plm(eval(formula), data = dat_plm, model = "within")
496   random <- plm(eval(formula), data = dat_plm, model = "random")
497
498   list(
499     tests = c(
500       pooltest(pooling, within)$p.value,
501       phptest(within, random)$p.value,
502       plm::r.squared(within, dfcor = T)),
503     model = within,
504     OLS = lm(data = dat_plm, formula = eval(paste(formula, "+ geo")))
505   )
506 })
507 #### Plot coefficients #####
508
509 m_panels %>%
510 lapply(function(output) {
511   output$model %>% broom::tidy(conf.int = T) %>%
512     rownames_to_column()
513 }) %>%
514 reduce(rbind) %>%
515 mutate(
516   rowname = paste0("Model ", as.roman(cumsum(rowname == 1)), "."),
517   term = f.clean_names(term, Tosparse = T),
518   term = gsub("_2", "^2", term),
519   term = gsub("_l_1", '[t = -2]', term),
520   term = gsub("_l", '[t = -1]', term),
521 ) %>%
522 ggplot() +
523   aes(estimate, term, color = p.value <= .05) +
524   geom_vline(xintercept = 0, color = "gray4") +
525   geom_point() +
526   geom_pointrange(aes(xmin = conf.low, xmax = conf.high)) +
527   facet_wrap(~rowname, nrow = 1) +
528   labs(x = "Estimated coefficient", y = "Term", color = "Corresponding p-value \u2264 5%") +
529   scale_color_viridis_d(option = "magma", begin = .2, end = .7) +
530   scale_y_discrete(labels=scales::parse_format())

```

```

531 theme(
532   legend.position = "bottom",
533   legend.direction = "horizontal"
534 )
535
536 ### Model descriptions #####
537
538 m_panels %>%
539   lapply(function(output) {
540     c(output$tests, nrow(augment(output$OLS)))
541   }) %>%
542   reduce(rbind) %>%
543   t() %>%
544   data.frame() %>%
545   mutate_all(function(x) c(scales::percent(x[1:3], accuracy = .01),
546                           as.character(x[4]))) %>%
547   {set_names(., paste0("Model ", as.roman(1:ncol(.)), "."))} %>%
548   mutate(
549     Indicator = c("Pooltest", "Phtest", "Adjusted $R^2$", "Observations")
550   ) %>%
551   select(Indicator, everything()) %>%
552   knitr::kable(caption = "Models", align = c("l", "c", "c", "c"))
553
554 # Framework I. =====
555
556 ## Bias of framework #####
557
558 names(dat_plm) %>%
559   { ifelse(
560     . %in% c("geo", "time") | str_detect(., "_1") | str_detect(., "_2"), NA, .
561   )} %>%
562   na.omit() %>%
563   lapply(function(variable){
564     x <- pull(dat_plm, variable) %>% na.omit()
565     y <- pull(na.omit(dat_plm), variable)
566     t <- t.test(x, y)
567     tibble(
568       Variable = f.clean_names(variable),
569       'Mean in total sample' = mean(x),
570       'Mean in used sample' = mean(y),
571       'Number of observations in the total sample' = length(x),
572       'T-statistic' = t$statistic,
573       'P-value' = scales::percent(t$p.value, accuracy = .01)
574     )
575   }
576 ) %>% reduce(rbind) %>%
577   knitr::kable(caption = 'T-tests', digits = 4,
578               align = c('l', 'c', 'c', 'c', 'c', 'c'))
579
580 dat_plm %>%
581   na.omit() %>%
582   group_by(geo) %>%
583   summarise(

```

```

584     values = n()
585   ) %>%
586   plot_NUTS2(all.x = T)
587
588 ### Run lasso regression #####
589
590 library(glmnet)
591
592 y <- na.omit(dat_plm)$TOTFERRT
593 X <- model.matrix(TOTFERRT ~ ., data = select(na.omit(dat_plm), -time))
594 LASSO <- cv.glmnet(X, y)
595
596 tidy(LASSO) %>%
597   ggplot() +
598   aes(log(lambda), ymin = conf.low*1000, ymax = conf.high*1000) +
599   geom_line(aes(log(lambda), estimate*1000, color = "Mean-Squared Error")) +
600   geom_step(aes(y = nonzero, color = "Number of used explanatory variables"),
601             size = 1) +
602   geom_ribbon(alpha = .4) +
603   geom_hline(yintercept = 0, size = 1) +
604   geom_vline(aes(xintercept = log(LASSO$lambda.1se), color = "lambda.1se"),
605              # TODO name the line
606              linetype = 2) +
607   geom_vline(aes(xintercept = log(LASSO$lambda.min), color = "lambda.min"),
608              linetype = 2) +
609   scale_y_continuous(
610     name = "Number of used explanatory variables",
611     sec.axis = sec_axis(trans=~./1000, name = "Mean-Squared Error")
612   ) +
613   labs(y = "Mean-Squared Error", x = "Log(\u03bb)", color = NULL) +
614   scale_color_viridis_d(option = "magma", end = .8) +
615   theme(legend.position = "bottom", legend.direction = "horizontal")
616
617 lasso_coefs <- capture.output(
618   coef(LASSO, LASSO$lambda.1se)
619 ) %>%
620   .[-(1:2)] %>%
621   {tibble(x = .)} %>%
622   mutate(term = gsub(" .*", "", x), coef = gsub(".* ", "", x)) %>%
623   select(-x) %>%
624   filter(!str_detect(term, "geo") & coef != "" & term != "(Intercept)")
625
626 m_panels2 <- paste("TOTFERRT ~", paste(lasso_coefs$term, collapse = " + ")) %>%
627   lapply(function(formula) {
628     pooling <- plm(eval(formula), data = dat_plm, model = "pooling")
629     within <- plm(eval(formula), data = dat_plm, model = "within")
630     random <- plm(eval(formula), data = dat_plm, model = "random")
631     list(
632       tests = c(
633         pooltest(pooling, within)$p.value,
634         phtest(within, random)$p.value,
635         plm::r.squared(within, dfcor = T)),
636       model = within,

```

```

637     OLS = lm(data = dat_plm, formula = eval(paste(formula, "+ geo")))
638   )
639 }
640 )
641
642 m_panels2 %>%
643   lapply(function(output) {
644     output$tests
645   })
646
647 standard_beta <- m_panels2[[1]]$OLS %>%
648   QuantPsyc::lm.beta() %>%
649   {tibble(term = names(.), beta = .)} %>%
650   filter(!str_detect(term, "geo"))
651
652 standard_beta <- augment(m_panels2[[1]]$OLS) %>%
653   select(TOTFERRT:uY25) %>%
654   cor() %>%
655   data.frame() %>%
656   select(1) %>%
657   rownames_to_column() %>%
658   rename(term = rowname) %>%
659   merge(standard_beta) %>%
660   mutate(explain = abs(TOTFERRT*beta)) %>%
661   select(term, cor = TOTFERRT, standard_beta = beta, explain)
662
663 lasso_coefs %>%
664   rename(lasso = coef) %>%
665   merge(tidy(m_panels2[[1]]$OLS, conf.int = T)) %>%
666   merge(standard_beta) %>%
667   mutate_at(-1, function(x) as.numeric(x)) %>%
668   pivot_longer(c(lasso:estimate, cor:explain)) %>%
669   mutate(
670     conf.low = ifelse(name != "estimate", NA, conf.low),
671     conf.high = ifelse(name != "estimate", NA, conf.high),
672     lag = ifelse(str_detect(term, "_l"),
673                  ifelse(str_detect(term, "_l_l"), 2, 1), 0),
674     bar = ifelse(name %in% c("cor", "explain"), value, NA),
675     value = ifelse(!(name %in% c("cor", "explain")), value, NA),
676     term = f.clean_names(term, Tosparse = T),
677     term = gsub("_2", "^2", term),
678     term = gsub("_l_l", '[t = -2]', term),
679     term = gsub("_l", '[t = -1]', term),
680     name = case_when(
681       name == "cor" ~ 'Correlation coefficient',
682       name == "estimate" ~ 'Coefficient in OLS',
683       name == "lasso" ~ 'Coefficient in lasso regression',
684       name == "standard_beta" ~ 'Standardized coefficient',
685       name == "explain" ~ 'Contribution to R-squared'
686     ),
687     name = factor(
688       name, levels = c(
689         'Correlation coefficient',

```

```

690     'Coefficient in lasso regression',
691     'Coefficient in OLS',
692     'Standardized coefficient',
693     'Contribution to R-squared'
694   )
695   ),
696 ) %>%
697 {
698 ggplot(.) +
699   geom_vline(xintercept = 0) +
700   geom_point(aes(value, term, fill = factor(lag)), size = 3) +
701   geom_col(aes(bar, term, fill = factor(lag)), color = "black") +
702   scale_fill_viridis_d(option = "magma", begin = .3, end = .7) +
703   scale_y_discrete(labels=scales::parse_format()) +
704   facet_wrap(~ name, scales = "free_x") +
705   labs(x = NULL, y = "Term", fill = "Lag") +
706   theme(legend.position = "bottom", legend.direction = "horizontal")
707 }
708
709 dat_plm <- dat_plm %>%
710   select(!starts_with("uY"))
711
712 LASSO <- cv.glmnet(model.matrix(TOTFERRT ~ ., data = select(na.omit(dat_plm), -time)),
713                      na.omit(dat_plm)$TOTFERRT)
714
715 names(dat_plm) %>%
716   {ifelse(
717     . %in% c("geo", "time") | str_detect(., "_1") | str_detect(., "_2"), NA, .
718   )} %>%
719   na.omit() %>%
720   lapply(function(variable){
721     x <- pull(dat_plm, variable) %>% na.omit()
722     y <- pull(na.omit(dat_plm), variable)
723     t <- t.test(x, y)
724     tibble(
725       Variable = f.clean_names(variable),
726       'Mean in total sample' = mean(x),
727       'Mean in used sample' = mean(y),
728       'Number of observations in total sample' = length(x),
729       'T-statistic' = t$statistic,
730       'P-value' = scales::percent(t$p.value, accuracy = .01)
731     )
732   })
733 } %>% reduce(rbind) %>%
734 knitr::kable(caption = 'T-tests', digits = 4,
735               align = c('l', 'c', 'c', 'c', 'c', 'c', 'c'))
736
737 dat_plm %>%
738   na.omit() %>%
739   group_by(geo) %>%
740   summarise(
741     values = n()
742   ) %>%

```

```

743 plot_NUTS2(all.x = T)
744
745 lasso_coefs <- capture.output(
746   coef(LASSO, LASSO$lambda.1se)
747 ) %>%
748   .[-(1:2)] %>%
749   {tibble(x = .)} %>%
750   mutate(term = gsub(" .*", "", x), coef = gsub(".* ", "", x)) %>%
751   select(-x) %>%
752   filter(!str_detect(term, "geo") & coef != "" & term != "(Intercept)")
753
754 m_panels2 <- paste("TOTFERRT ~", paste(lasso_coefs$term, collapse = " + ")) %>%
755   lapply(function(formula) {
756     pooling <- plm(eval(formula), data = dat_plm, model = "pooling")
757     within <- plm(eval(formula), data = dat_plm, model = "within")
758     random <- plm(eval(formula), data = dat_plm, model = "random")
759     list(
760       tests = c(
761         pooltest(pooling, within)$p.value,
762         phtest(within, random)$p.value,
763         plm::r.squared(within, dfcor = T)),
764       model = within,
765       OLS = lm(data = dat_plm, formula = eval(paste(formula, "+ geo")))
766     )
767   }
768 )
769
770 m_panels2 %>%
771   lapply(function(output) {
772     output$tests
773   })
774
775 standard_beta <- m_panels2[[1]]$OLS %>%
776   QuantPsyc::lm.beta() %>%
777   {tibble(term = names(.), beta = .)} %>%
778   filter(!str_detect(term, "geo"))
779
780 standard_beta <- augment(m_panels2[[1]]$OLS) %>%
781   select(TOTFERRT:edu_index) %>%
782   cor() %>%
783   data.frame() %>%
784   select(1) %>%
785   rownames_to_column() %>%
786   rename(term = rowname) %>%
787   merge(standard_beta) %>%
788   mutate(explain = abs(TOTFERRT*beta)) %>%
789   select(term, cor = TOTFERRT, standard_beta = beta, explain)
790
791 lasso_coefs %>%
792   rename(lasso = coef) %>%
793   merge(tidy(m_panels2[[1]]$OLS, conf.int = T)) %>%
794   merge(standard_beta) %>%
795   mutate_at(-1, function(x) as.numeric(x)) %>%

```

```

796 pivot_longer(c(lasso$estimate, cor$explain)) %>%
797   mutate(
798     conf.low = ifelse(name != "estimate", NA, conf.low),
799     conf.high = ifelse(name != "estimate", NA, conf.high),
800     lag = ifelse(str_detect(term, "_l"),
801                  ifelse(str_detect(term, "_l_l"), 2, 1), 0),
802     bar = ifelse(name %in% c("cor", "explain"), value, NA),
803     value = ifelse(!(name %in% c("cor", "explain")), value, NA),
804     term = f.clean_names(term, Tosparse = T),
805     term = gsub("_2", "^2", term),
806     term = gsub("_l_l", '[t = -2]', term),
807     term = gsub("_l", '[t = -1]', term),
808     name = case_when(
809       name == "cor" ~ 'Correlation coefficient',
810       name == "estimate" ~ 'Coefficient in OLS',
811       name == "lasso" ~ 'Coefficient in lasso regression',
812       name == "standard_beta" ~ 'Standardized coefficient',
813       name == "explain" ~ 'Contribution to R-squared'
814     ),
815     name = factor(
816       name, levels = c(
817         'Correlation coefficient',
818         'Coefficient in lasso regression',
819         'Coefficient in OLS',
820         'Standardized coefficient',
821         'Contribution to R-squared'
822       )
823     )
824   ) %>%
825 {
826   ggplot(.) +
827     geom_vline(xintercept = 0) +
828     geom_point(aes(value, term, fill = factor(lag)), size = 3) +
829     geom_col(aes(bar, term, fill = factor(lag)), color = "black") +
830     scale_fill_viridis_d(option = "magma", begin = .3, end = .7) +
831     scale_y_discrete(labels = scales::parse_format()) +
832     facet_wrap(~ name, scales = "free_x") +
833     labs(x = NULL, y = "Term", fill = "Lag") +
834     theme(legend.position = "bottom", legend.direction = "horizontal")
835 }

```