## Extension: Synthetic Control without Piauí

```
#-------------------------------------------------------------------------------------------#
# Data Subsetting
#-------------------------------------------------------------------------------------------#

PRIMARY_M_PI <- as.data.frame(filter(DATA_COMPLETE, grade == "P", subject == "math"))
PRIMARY_P_PI <- as.data.frame(filter(DATA_COMPLETE, grade == "P", subject == "port"))
LOWERS_M_PI <- as.data.frame(filter(DATA_COMPLETE, grade == "LS", subject == "math"))
LOWERS_P_PI <- as.data.frame(filter(DATA_COMPLETE, grade == "LS", subject == "port"))

#-------------------------------------------------------------------------------------------#
# Preparing Data for Synth (Excluding Piauí)
#-------------------------------------------------------------------------------------------#

prepare_p_ls_PI <- function(data) {
  library(Synth)

  predictors <- c("homicides", "TWh", "ln_pop", "unemployment", "edu_invest_pc")

  DATA_PI <- dataprep(foo = data,
                      predictors = predictors,
                      dependent = "score",
                      unit.variable = "code_state",
                      time.variable = "year",
                      unit.names.variable = "abbr_state",
                      treatment.identifier = 23, # Ceará Code
                      controls.identifier = c(11:17, 21, 24:29, 31:33, 35, 41:43, 50:53),
                      # Exclude Piauí (22) and Ceará (23)
                      time.predictors.prior = seq(1995, 2007, 2),
                      time.optimize.ssr = seq(1995, 2007, 2),
                      time.plot = seq(1995, 2019, 2))

  return(DATA_PI)
}

#-------------------------------------------------------------------------------------------#
# Execution of Synthetic Control Method
#-------------------------------------------------------------------------------------------#

DATA_PM_PI <- prepare_p_ls_PI(PRIMARY_M_PI)
SCM_PM_PI <- synth(DATA_PM_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
```

```
##
## MSPE (LOSS V): 63.48728
##
## solution.v:
##  0.1934857 0.3127029 2.19e-08 0.2126195 0.2811919
##
## solution.w:
##  0.004474852 0.0009300773 0.0007292257 0.0007054601 0.001420396 0.0005013932 0.002673195 0.1163018 0
```

```
DATA_PP_PI <- prepare_p_ls_PI(PRIMARY_P_PI)
SCM_PP_PI <- synth(DATA_PP_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 46.26716
##
## solution.v:
##  0.3413594 0.4124263 0.01070271 3.33389e-05 0.2354782
##
## solution.w:
##  4.55448e-05 2.50038e-05 7.84002e-05 1.99015e-05 3.23363e-05 2.20135e-05 2.9265e-05 0.0009849393 4.68
```

```
DATA_LSM_PI <- prepare_p_ls_PI(LOWERS_M_PI)
SCM_LSM_PI <- synth(DATA_LSM_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 10.86386
##
## solution.v:
##  0.19474 0.3147814 1e-09 0.2101793 0.2802993
##
## solution.w:
##  0.004648188 0.0009383975 0.0007193599 0.0007152314 0.001422106 0.0005072035 0.002593207 0.1156682 0
```

```r
DATA_LSP_PI <- prepare_p_ls_PI(LOWERS_P_PI)
SCM_LSP_PI <- synth(DATA_LSP_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 13.40889
##
## solution.v:
##  0.1547955 0.2160913 0.0001584944 0.002317496 0.6266372
##
## solution.w:
##  0.004261227 0.001742724 0.002064103 0.001211499 0.002541634 0.001109108 0.003500906 0.1036674 0.0028
```

```r
#-------------------------------------------------------------------------------------------#
# Graphs Generation in ggplot (Excluding Piauí)
#-------------------------------------------------------------------------------------------#

# Generate Plots for Each Category
PM_PI <- plot_scm(PRIMARY_M_PI, synth.tab(dataprep.res = DATA_PM_PI, synth.res = SCM_PM_PI))
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.0
## v readr     2.1.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
## 'data.frame':    5 obs. of  3 variables:
##  $ w.weights   : num  0.116 0.768 0.033 0.012 0.033
##  $ abbr_state  : chr  "MA" "PB" "PE" "SC" ...
##  $ unit.numbers: num  21 25 26 42 51
## 'data.frame':    351 obs. of  14 variables:
##  $ abbr_state   : chr  "AC" "AC" "AC" "AC" ...
##  $ code_state   : num  12 12 12 12 12 27 27 27 27 27 ...
##  $ year         : num  1995 1997 1999 2001 2003 ...
##  $ grade        : chr  "P" "P" "P" "P" ...
##  $ subject      : chr  "math" "math" "math" "math" ...
##  $ score        : num  168 166 162 152 159 ...
##  $ TWh          : num  0.0139 0.0132 0.0137 0.0198 0.0227 ...
##  $ homicides    : num  22.62 19.99 9.66 21.24 22.04 ...
##  $ pop          : num  455242 500185 527937 574355 600595 ...
##  $ ln_pop       : num  13 13.1 13.2 13.3 13.3 ...
```

```
##  $ unemployment : num  9.8 8.89 12.2 8.56 7.45 ...
##  $ gini         : num  0.582 0.574 0.621 0.623 0.578 ...
##  $ edu_invest   : num  3.69e+08 4.34e+08 4.54e+08 6.19e+08 6.26e+08 ...
##  $ edu_invest_pc: num  811 868 860 1077 1042 ...
```

```
PP_PI <- plot_scm(PRIMARY_P_PI, synth.tab(dataprep.res = DATA_PP_PI, synth.res = SCM_PP_PI))
```

```
## 'data.frame':    4 obs. of  3 variables:
##  $ w.weights   : num  0.738 0.028 0.115 0.117
##  $ abbr_state  : chr  "PB" "PE" "BA" "GO"
##  $ unit.numbers: num  25 26 29 52
## 'data.frame':    351 obs. of  14 variables:
##  $ abbr_state   : chr  "AC" "AC" "AC" "AC" ...
##  $ code_state   : num  12 12 12 12 12 27 27 27 27 27 ...
##  $ year         : num  1995 1997 1999 2001 2003 ...
##  $ grade        : chr  "P" "P" "P" "P" ...
##  $ subject      : chr  "port" "port" "port" "port" ...
##  $ score        : num  166 160 154 148 158 ...
##  $ TWh          : num  0.0139 0.0132 0.0137 0.0198 0.0227 ...
##  $ homicides    : num  22.62 19.99 9.66 21.24 22.04 ...
##  $ pop          : num  455242 500185 527937 574355 600595 ...
##  $ ln_pop       : num  13 13.1 13.2 13.3 13.3 ...
##  $ unemployment : num  9.8 8.89 12.2 8.56 7.45 ...
##  $ gini         : num  0.582 0.574 0.621 0.623 0.578 ...
##  $ edu_invest   : num  3.69e+08 4.34e+08 4.54e+08 6.19e+08 6.26e+08 ...
##  $ edu_invest_pc: num  811 868 860 1077 1042 ...
```

```
LSM_PI <- plot_scm(LOWERS_M_PI, synth.tab(dataprep.res = DATA_LSM_PI, synth.res = SCM_LSM_PI))
```

```
## 'data.frame':    5 obs. of  3 variables:
##  $ w.weights   : num  0.116 0.769 0.033 0.012 0.032
##  $ abbr_state  : chr  "MA" "PB" "PE" "SC" ...
##  $ unit.numbers: num  21 25 26 42 51
## 'data.frame':    351 obs. of  14 variables:
##  $ abbr_state   : chr  "AC" "AC" "AC" "AC" ...
##  $ code_state   : num  12 12 12 12 12 27 27 27 27 27 ...
##  $ year         : num  1995 1997 1999 2001 2003 ...
##  $ grade        : chr  "LS" "LS" "LS" "LS" ...
##  $ subject      : chr  "math" "math" "math" "math" ...
##  $ score        : num  223 220 223 219 225 ...
##  $ TWh          : num  0.0139 0.0132 0.0137 0.0198 0.0227 ...
##  $ homicides    : num  22.62 19.99 9.66 21.24 22.04 ...
##  $ pop          : num  455242 500185 527937 574355 600595 ...
##  $ ln_pop       : num  13 13.1 13.2 13.3 13.3 ...
##  $ unemployment : num  9.8 8.89 12.2 8.56 7.45 ...
##  $ gini         : num  0.582 0.574 0.621 0.623 0.578 ...
##  $ edu_invest   : num  3.69e+08 4.34e+08 4.54e+08 6.19e+08 6.26e+08 ...
##  $ edu_invest_pc: num  811 868 860 1077 1042 ...
```

```
LSP_PI <- plot_scm(LOWERS_P_PI, synth.tab(dataprep.res = DATA_LSP_PI, synth.res = SCM_LSP_PI))
```

```
## 'data.frame':    3 obs. of  3 variables:
##  $ w.weights   : num  0.104 0.785 0.039
##  $ abbr_state  : chr  "MA" "PB" "PE"
##  $ unit.numbers: num  21 25 26
## 'data.frame':    351 obs. of  14 variables:
```

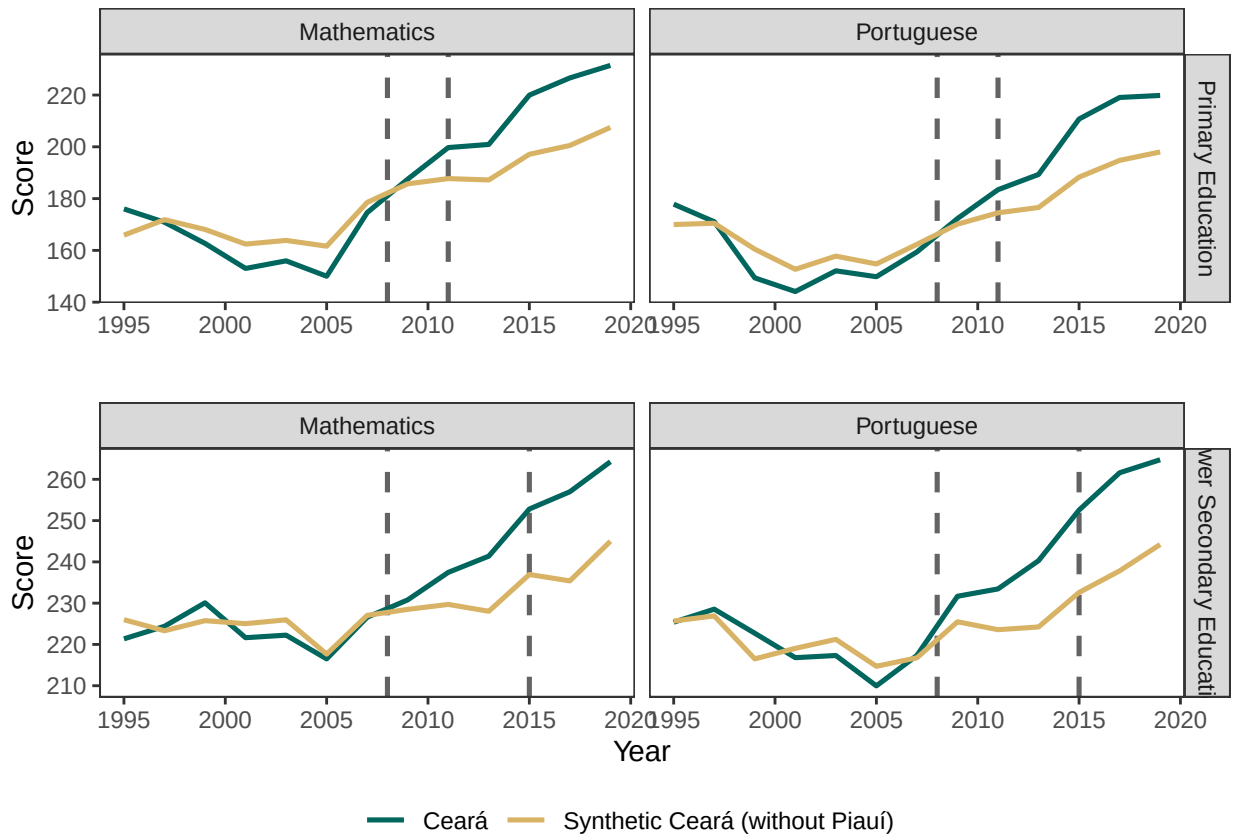
```
##  $ abbr_state   : chr  "AC" "AC" "AC" "AC" ...
##  $ code_state   : num  12 12 12 12 12 27 27 27 27 27 ...
##  $ year         : num  1995 1997 1999 2001 2003 ...
##  $ grade        : chr  "LS" "LS" "LS" "LS" ...
##  $ subject      : chr  "port" "port" "port" "port" ...
##  $ score        : num  227 223 217 218 223 ...
##  $ TWh          : num  0.0139 0.0132 0.0137 0.0198 0.0227 ...
##  $ homicides    : num  22.62 19.99 9.66 21.24 22.04 ...
##  $ pop          : num  455242 500185 527937 574355 600595 ...
##  $ ln_pop       : num  13 13.1 13.2 13.3 13.3 ...
##  $ unemployment : num  9.8 8.89 12.2 8.56 7.45 ...
##  $ gini         : num  0.582 0.574 0.621 0.623 0.578 ...
##  $ edu_invest   : num  3.69e+08 4.34e+08 4.54e+08 6.19e+08 6.26e+08 ...
##  $ edu_invest_pc: num  811 868 860 1077 1042 ...
```

```r
#-----------------------------------------------------------------------------------------#
# Combining Data for Graphs and Plot Adjustments
#-----------------------------------------------------------------------------------------#

# Combine the Synthetic Control Graph Data
DATA_GRAPH_PI <- rbind(PM_PI[[1]], PP_PI[[1]], LSM_PI[[1]], LSP_PI[[1]])

# Adjust the Labels and Factors
DATA_GRAPH_PI$grade[DATA_GRAPH_PI$grade=="P"] <- "Primary Education"
DATA_GRAPH_PI$grade[DATA_GRAPH_PI$grade=="LS"] <- "Lower Secondary Education"
DATA_GRAPH_PI$grade[DATA_GRAPH_PI$grade=="US"] <- "Upper Secondary Education"
DATA_GRAPH_PI$grade <- factor(DATA_GRAPH_PI$grade, levels = c("Primary Education",
                                                              "Lower Secondary Education"))

DATA_GRAPH_PI$subject[DATA_GRAPH_PI$subject=="math"] <- "Mathematics"
DATA_GRAPH_PI$subject[DATA_GRAPH_PI$subject=="port"] <- "Portuguese"
DATA_GRAPH_PI$subject <- factor(DATA_GRAPH_PI$subject, levels = c("Mathematics", "Portuguese"))

#-----------------------------------------------------------------------------------------#
# Final Plotting and Saving
#-----------------------------------------------------------------------------------------#

# Assuming DATA_GRAPH_PI and DATA_GAP_PI are your combined datasets

# Figure for Primary Education
a_without_PI <- ggplot(data = filter(DATA_GRAPH_PI, grade == "Primary Education"),
                       aes(x = year, y = score, color = unit)) +
  geom_vline(xintercept = 2008, color = "#636363", linetype = "dashed", size = 0.9) +
  geom_vline(xintercept = 2011, color = "#636363", linetype = "dashed", size = 0.9) +
  geom_line(size = 0.9) +
  scale_color_manual(values = c("#01665e", "#d8b365"), labels = c("Ceará",
                                                                  "Synthetic Ceará (without Piauí)"), na
  ylab("Score") +
  xlab("") +
  theme_bw() +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        legend.position = "bottom") +
  facet_grid(vars(grade), vars(subject))
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
```

```
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
# Figure for Lower Secondary Education
b_without_PI <- ggplot(data = filter(DATA_GRAPH_PI, grade == "Lower Secondary Education"),
                       aes(x = year, y = score, color = unit)) +
  geom_vline(xintercept = 2008, color = "#636363", linetype = "dashed", size = 0.9) +
  geom_vline(xintercept = 2015, color = "#636363", linetype = "dashed", size = 0.9) +
  geom_line(size = 0.9) +
  scale_color_manual(values = c("#01665e", "#d8b365"),
                     labels = c("Ceará", "Synthetic Ceará (without Piauí)"), name = "") +
  ylab("Score") +
  xlab("Year") +
  theme_bw() +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank(),
        legend.position = "bottom") +
  facet_grid(vars(grade), vars(subject))

# Arrange and Save the Plots
ggarrange(a_without_PI, b_without_PI, ncol = 1, nrow = 2, common.legend = TRUE, legend = "bottom")
```




```
ggsave(filename = "figures_without_PI.png", path = "plots", width = 21, height = 15, units = "cm")

# Run Synthetic Control Method for each category (Excluding Piauí)
SCM_PM_PI    <- synth(DATA_PM_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 63.48728
##
## solution.v:
##  0.1934857 0.3127029 2.19e-08 0.2126195 0.2811919
##
## solution.w:
##  0.004474852 0.0009300773 0.0007292257 0.0007054601 0.001420396 0.0005013932 0.002673195 0.1163018 0
```

```
TABLES_PM_PI <- synth.tab(dataprep.res = DATA_PM_PI, synth.res = SCM_PM_PI)

SCM_PP_PI    <- synth(DATA_PP_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 46.26716
##
## solution.v:
##  0.3413594 0.4124263 0.01070271 3.33389e-05 0.2354782
##
## solution.w:
##  4.55448e-05 2.50038e-05 7.84002e-05 1.99015e-05 3.23363e-05 2.20135e-05 2.9265e-05 0.0009849393 4.6
```

```
TABLES_PP_PI <- synth.tab(dataprep.res = DATA_PP_PI, synth.res = SCM_PP_PI)

SCM_LSM_PI   <- synth(DATA_LSM_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
```

```
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 10.86386
##
## solution.v:
##  0.19474 0.3147814 1e-09 0.2101793 0.2802993
##
## solution.w:
##  0.004648188 0.0009383975 0.0007193599 0.0007152314 0.001422106 0.0005072035 0.002593207 0.1156682 0
```

```
TABLES_LSM_PI <- synth.tab(dataprep.res = DATA_LSM_PI, synth.res = SCM_LSM_PI)

SCM_LSP_PI   <- synth(DATA_LSP_PI)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 13.40889
##
## solution.v:
##  0.1547955 0.2160913 0.0001584944 0.002317496 0.6266372
##
## solution.w:
##  0.004261227 0.001742724 0.002064103 0.001211499 0.002541634 0.001109108 0.003500906 0.1036674 0.002
```

```
TABLES_LSP_PI <- synth.tab(dataprep.res = DATA_LSP_PI, synth.res = SCM_LSP_PI)

# Extracting Weights and Renaming
W_PM_PI <- as.data.frame(TABLES_PM_PI$tab.w) %>%
  filter(w.weights > 0.01) %>%
  rename(PM_PI = w.weights, abbreviation = unit.names) %>%
  dplyr::select(-unit.numbers)

W_PP_PI <- as.data.frame(TABLES_PP_PI$tab.w) %>%
  filter(w.weights > 0.01) %>%
  rename(PP_PI = w.weights, abbreviation = unit.names) %>%
  dplyr::select(-unit.numbers)

W_LSM_PI <- as.data.frame(TABLES_LSM_PI$tab.w) %>%
  filter(w.weights > 0.01) %>%
  rename(LSM_PI = w.weights, abbreviation = unit.names) %>%
  dplyr::select(-unit.numbers)
```

```r
W_LSP_PI <- as.data.frame(TABLES_LSP_PI$tab.w) %>%
  filter(w.weights > 0.01) %>%
  rename(LSP_PI = w.weights, abbreviation = unit.names) %>%
  dplyr::select(-unit.numbers)

#-------------------------------------------------------------------------------------------#
# Weights Table
#-------------------------------------------------------------------------------------------#

# Combining Weights into a Single Table
TABLE_W_PI <- left_join(abbr_state, W_PM_PI, by = "abbreviation")
TABLE_W_PI <- left_join(TABLE_W_PI, W_PP_PI, by = "abbreviation")
TABLE_W_PI <- left_join(TABLE_W_PI, W_LSM_PI, by = "abbreviation")
TABLE_W_PI <- left_join(TABLE_W_PI, W_LSP_PI, by = "abbreviation")

# Replace NA values with 0.00
TABLE_W_PI[is.na(TABLE_W_PI)] <- 0.00

# Write the table to a CSV file
write.csv(TABLE_W_PI, file = "table_weights_without_PI.csv")
```