Regression Discontinuity and Synthetic Control

David Carlson

February 25, 2021

Regression Discontinuity Design (RDD)

- Used for quasi-experimental settings
- Supposedly estimates causal effects of some treatment/intervention
- Generally will perform well when done well, but needs a particular set-up
- Basically, if there exists some cutoff or threshold dictating treatment type, you can compare observations that lie close together but on either side of the threshold
- The argument is that in the aggregate, there should be minimal or no difference other than random perturbations between the units in the study and the treatment assignment
- This (supposedly) gets us assignment as if random
- Examples:
- Just above or below a test score for admission
- On either side of a seemingly randomly drawn border (mita in Peru, colonial imposed country or region borders, etc.)
- Elections in which one politician wins by a marginal majority
- Age eligibility
- Can be non-parametric or parametric

Non-parametric RDD estimation (local linear regression)

- Different slopes and intercepts on either side of cutoff
- Triangular kernel is a less biased estimator, but the rectangular kernel (no weighting) is easier to interpret
- Let the cutoff be c
- The bandwidth, or how far from the cutoff the data are that you are willing to analyze (e.g., a kilometer from a border), is denoted h
- X is the value of the variable used for the cutoff (e.g., test score, distance from border, etc.)
- As such, $c - h \leq X \leq c + h$
- D is a binary variable indicating treatment
- Full model:

$$Y = \alpha + \tau D + \beta_1(X - c) + \beta_2 D(X - c) + \epsilon.$$

```
data = read.csv('../day11TSCS/bosnia.csv')
#for simplicity, lets look just at 2006 data, with one observation per municipality
data = data[data$Year == 2006,]
#exercise: analyze Ethnic_Vote_Share as it relates to Log_Casualty
#set the cutoff to the mean, and write a function that takes as an argument the bandwidth, then runs an
#change the bandwidth and see if results vary
#homework: pick one of the models, and plot the results with reliable intervals
```

Parametric Estimation

$$y_i = \alpha + \beta_1 D + \beta_2 x_i + \beta_3 x_i^2 + \ldots + \beta_{k+1} x_i^k + \epsilon.$$

- k can be adjusted to fit the analysis
- We strongly prefer the non-parametric estimation

Miscellaneous

- We do not generally require controls, because we make assumptions
- All potentially relevant variables besides treatment and outcome are continuous at the point where the discontinuities occur
- Assignment is therefore as if random at threshold
- The units considered cannot directly manipulate treatment condition when close to threshold due to noise (e.g., just below or above a test score threshold is assumed due to chance)
- Be careful of selection bias with RDD
- There are some tests for the assumptions, but fairly unprincipled; plot results and analyze, look at other variables (if the data exist), etc.

Synthetic Control Method

- Test the effect of some intervention/treatment by synthesizing a treated unit as if it were not treated
- Compare the synthetic unit to the actual data
- We synthesize the outcome of the treated unit had there been no treatment through weighted linear regression
- The weights of the donor pool are constrained to be between zero and one
- There is generally an (arbitrary) cutoff to not include low weighted units
- Simply, the weights are multiplied by the outcomes of the donor units post-treatment to generate point estimates of the outcome of the treated absent treatment
- Intervention must have no effect during pretreatment period
- Matching on preintervention outcomes helps control for unobserved factors
- Utilize placebo checks when intervention did not occur
- Use placebo checks of untreated units to obtain p-values
- Should exclude from donor pool units affected by treatment, or by events of similar nature

```
#this is the replication material for the original paper
library(Synth)
```

```
## ##
## ## Synth Package: Implements Synthetic Control Methods.
## ## See http://www.mit.edu/~jhainm/software.htm for additional information.
```

```
library(xtable)

# Load Data
load("repgermany.RData")
d=x
rm(x)
dataprep.out <-
  dataprep(
           foo = d,
           predictors    = c("gdp","trade","infrate"),
           dependent     = "gdp",
           unit.variable = 1,
           time.variable = 3,
           special.predictors = list(
```

```
            list("industry", 1971:1980, c("mean")),
            list("schooling",c(1970,1975), c("mean")),
            list("invest70" ,1980, c("mean"))
           ),
           treatment.identifier = 7,
           controls.identifier = unique(d$index)[-7],
           time.predictors.prior = 1971:1980,
           time.optimize.ssr = 1981:1990,
           unit.names.variable = 2,
           time.plot = 1960:2003
         )

# fit training model
synth.out <-
  synth(
        data.prep.obj=dataprep.out,
        Margin.ipop=.005,Sigf.ipop=7,Bound.ipop=6
        )
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 4580.211
##
## solution.v:
##  0.5885675 0.1037433 0.04737192 0.005338085 0.08311456 0.1718646
##
## solution.w:
##  0.1350128 2.598e-07 0.5073743 2.256e-07 1.0906e-06 2.752e-07 2.15e-07 8.495e-07 2.471e-07 0.1659188
```

```
# data prep for main model
dataprep.out <-
 dataprep(
  foo = d,
  predictors    = c("gdp","trade","infrate"),
  dependent     = "gdp",
  unit.variable = 1,
  time.variable = 3,
  special.predictors = list(
    list("industry" ,1981:1990, c("mean")),
    list("schooling",c(1980,1985), c("mean")),
    list("invest80" ,1980, c("mean"))
  ),
  treatment.identifier = 7,
  controls.identifier = unique(d$index)[-7],
```

```
  time.predictors.prior = 1981:1990,
  time.optimize.ssr = 1960:1989,
  unit.names.variable = 2,
  time.plot = 1960:2003
)
```

```
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
```

```
# fit main model with v from training model
synth.out <- synth(
  data.prep.obj=dataprep.out,
  custom.v=as.numeric(synth.out$solution.v)
  )
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## *****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## *****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## *****************
## *****************
## *****************
##
## MSPE (LOSS V): 14178.38
##
## solution.v:
##  0.5885675 0.1037433 0.04737192 0.005338085 0.08311456 0.1718646
##
## solution.w:
##  0.2186415 0.001006986 0.4206785 0.001188088 0.00114549 0.0009857056 0.000634122 0.08734495 0.0006968
```

```
#### Table 2
synth.tables <- synth.tab(
                          dataprep.res = dataprep.out,
                          synth.res = synth.out
                          ); synth.tables
```

```
## $tab.pred
##                               Treated Synthetic Sample Mean
## gdp                          15808.900 15804.636   13669.381
## trade                           56.778    56.915      59.831
## infrate                          2.595     3.507       7.617
## special.industry.1981.1990      34.538    34.385      33.794
## special.schooling.1980.1985     55.500    55.229      38.659
## special.invest80.1980           27.018    27.035      25.895
```

```
##
## $tab.v
##                               v.weights
## gdp                               0.589
## trade                             0.104
## infrate                           0.047
## special.industry.1981.1990        0.005
## special.schooling.1980.1985       0.083
## special.invest80.1980             0.172
##
## $tab.w
##    w.weights  unit.names unit.numbers
## 1      0.219         USA            1
## 2      0.001          UK            2
## 3      0.421     Austria            3
## 4      0.001     Belgium            4
## 5      0.001     Denmark            5
## 6      0.001      France            6
## 8      0.001       Italy            8
## 9      0.087 Netherlands            9
## 10     0.001      Norway           10
## 12     0.112 Switzerland           12
## 14     0.153       Japan           14
## 16     0.000      Greece           16
## 18     0.000    Portugal           18
## 19     0.001       Spain           19
## 20     0.001   Australia           20
## 21     0.001 New Zealand           21
##
## $tab.loss
##           Loss W   Loss V
## [1,] 0.001705482 14178.38
```

```
# Replace means for OECD sample (computed externally using proper pop weighting)
synth.tables$tab.pred[,3]          <- c(8021.1,31.9,7.4,34.2,44.1,25.9)
colnames(synth.tables$tab.pred)[3] <- "Rest of OECD Sample"
rownames(synth.tables$tab.pred) <- c("GDP per-capita","Trade openness",
                                     "Inflation rate","Industry share",
                                     "Schooling","Investment rate")

xtable(round(synth.tables$tab.pred,1),digits=1)
```

```
## % latex table generated in R 3.6.3 by xtable 1.8-3 package
## % Wed Feb 24 17:25:13 2021
## \begin{table}[ht]
## \centering
## \begin{tabular}{rrrr}
##   \hline
##  & Treated & Synthetic & Rest of OECD Sample \\
##   \hline
## GDP per-capita & 15808.9 & 15804.6 & 8021.1 \\
##   Trade openness & 56.8 & 56.9 & 31.9 \\
##   Inflation rate & 2.6 & 3.5 & 7.4 \\
##   Industry share & 34.5 & 34.4 & 34.2 \\
##   Schooling & 55.5 & 55.2 & 44.1 \\
```

```
##   Investment rate & 27.0 & 27.0 & 25.9 \\
##    \hline
## \end{tabular}
## \end{table}
```

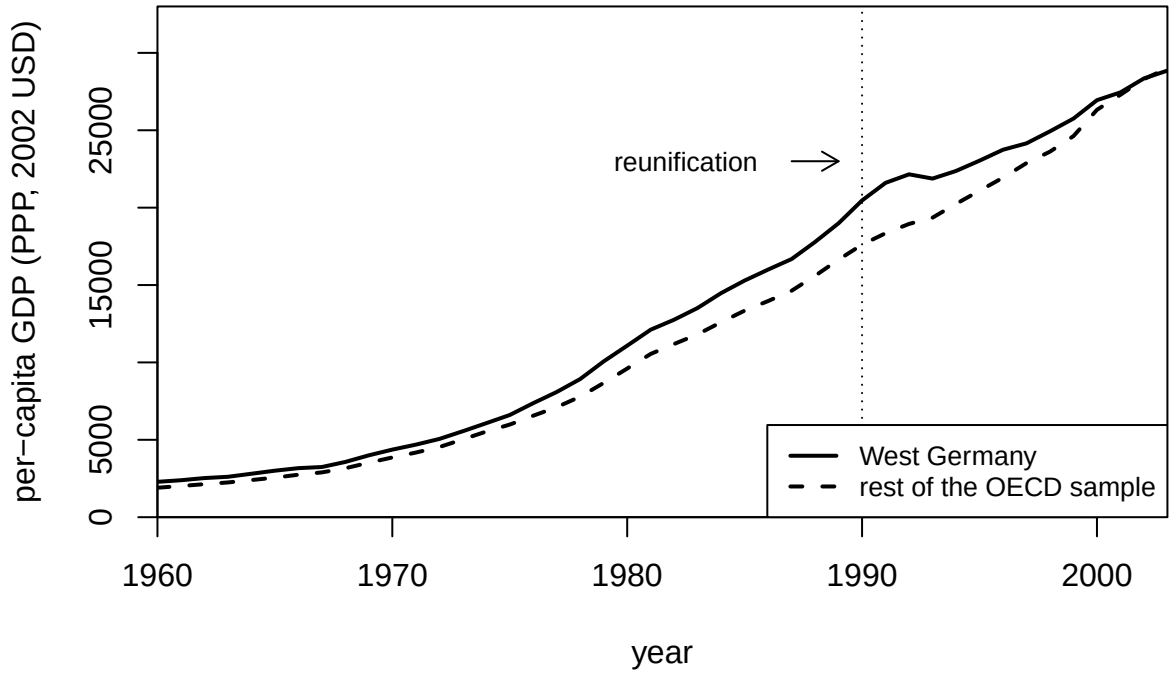
```
#### Table 1
# synth weights
tab1 <- data.frame(synth.tables$tab.w)
tab1[,1] <- round(tab1[,1],2)
# regression weights
X0 <- cbind(1,t(dataprep.out$X0))
X1 <- as.matrix(c(1,dataprep.out$X1))
W     <- X0%*%solve(t(X0)%*%X0)%*%X1
Wdat  <- data.frame(unit.numbers=as.numeric(rownames(X0)),
                    regression.w=round(W,2))
tab1  <- merge(tab1,Wdat,by="unit.numbers")
tab1  <- tab1[order(tab1[,3]),]

xtable(cbind(tab1[1:9,c(3,2,4)],
             tab1[10:18,c(3,2,4)]
             )
       )
```

```
## % latex table generated in R 3.6.3 by xtable 1.8-3 package
## % Wed Feb 24 17:25:13 2021
## \begin{table}[ht]
## \centering
## \begin{tabular}{rlrrlrr}
##   \hline
##  & unit.names & w.weights & regression.w & unit.names & w.weights & regression.w \\
##   \hline
## 15 & Australia & 0.00 & 0.12 & New Zealand & 0.00 & 0.12 \\
##   3 & Austria & 0.42 & 0.26 & Norway & 0.00 & 0.04 \\
##   4 & Belgium & 0.00 & -0.00 & Portugal & 0.00 & -0.08 \\
##   5 & Denmark & 0.00 & 0.08 & Spain & 0.00 & -0.01 \\
##   6 & France & 0.00 & 0.04 & Switzerland & 0.11 & 0.05 \\
##   12 & Greece & 0.00 & -0.09 & UK & 0.00 & 0.06 \\
##   7 & Italy & 0.00 & -0.05 & USA & 0.22 & 0.13 \\
##   11 & Japan & 0.15 & 0.19 &  &  &  \\
##   8 & Netherlands & 0.09 & 0.14 &  &  &  \\
##    \hline
## \end{tabular}
## \end{table}
```

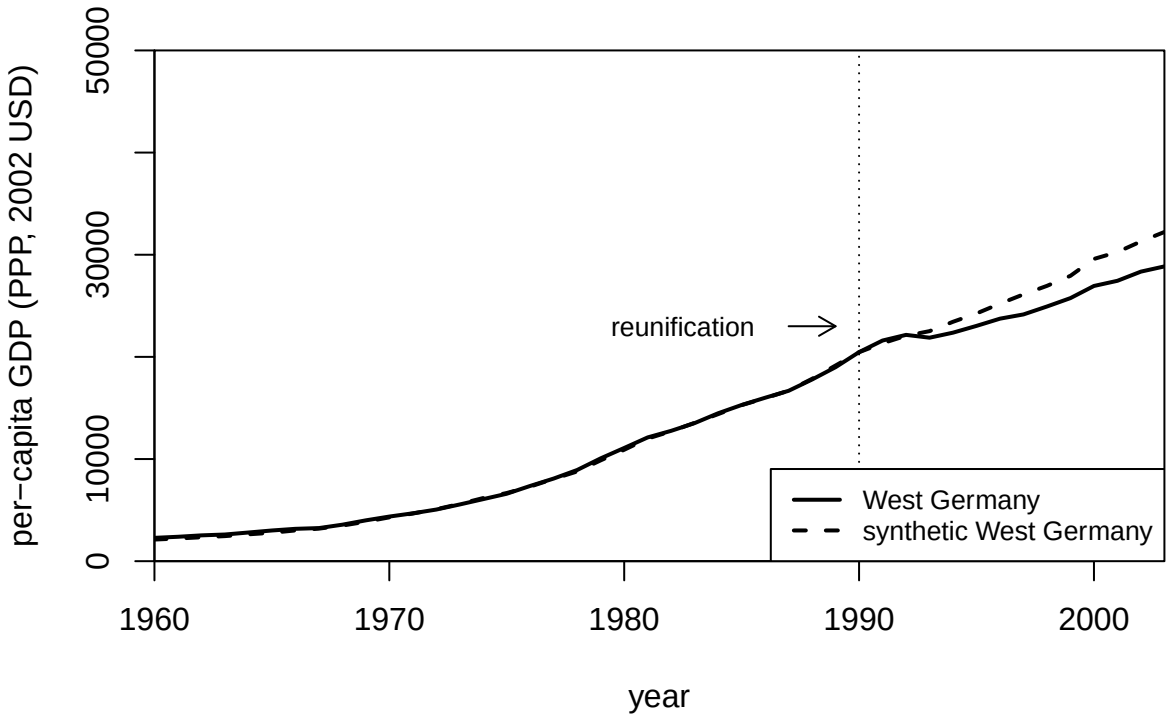
```
#### Figure 1: Trends in Per-Capita GDP: West Germany vs. Rest of the OECD Sample
Text.height <- 23000
Cex.set <- .8
plot(1960:2003,dataprep.out$Y1plot,
     type="l",ylim=c(0,33000),col="black",lty="solid",
     ylab ="per-capita GDP (PPP, 2002 USD)",
     xlab ="year",
     xaxs = "i", yaxs = "i",
     lwd=2
     )
lines(1960:2003,aggregate(d[,c("gdp")],by=list(d$year),mean,na.rm=T)[,2]
```

```
      ,col="black",lty="dashed",lwd=2) # mean 2
abline(v=1990,lty="dotted")
legend(x="bottomright",
       legend=c("West Germany","rest of the OECD sample")
      ,lty=c("solid","dashed"),col=c("black","black")
      ,cex=.8,bg="white",lwd=c(2,2))
arrows(1987,Text.height,1989,Text.height,col="black",length=.1)
text(1982.5,Text.height,"reunification",cex=Cex.set)
```

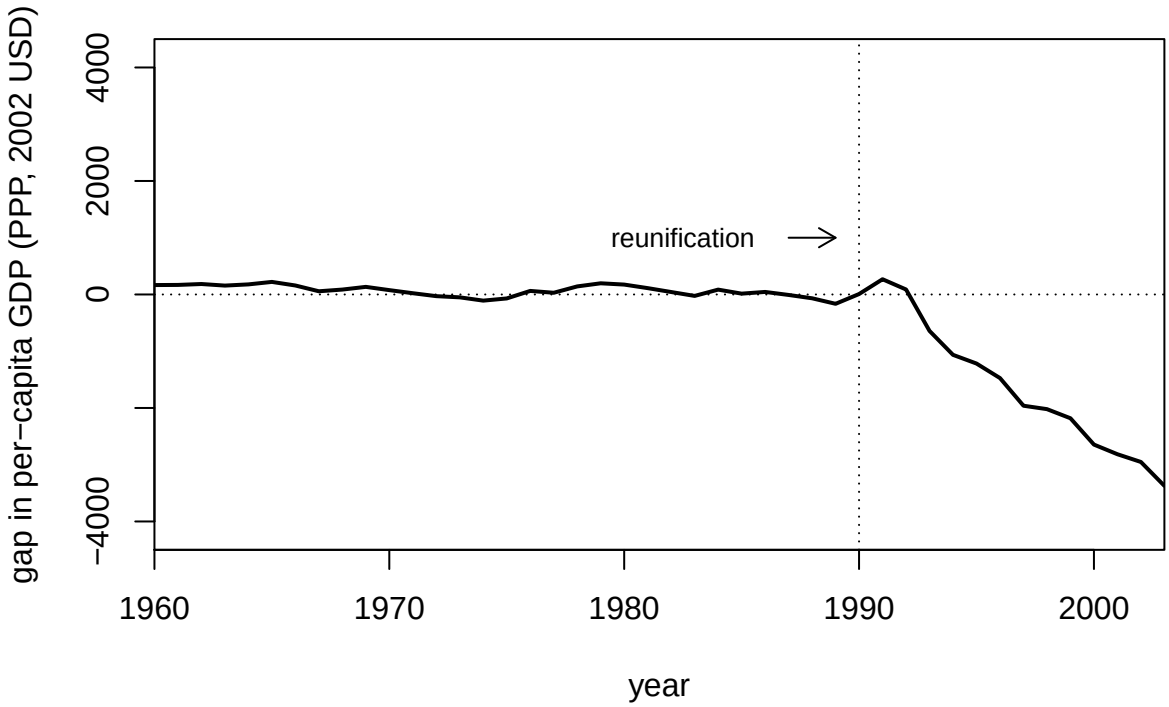



```
synthY0 <- (dataprep.out$Y0%*%synth.out$solution.w)
plot(1960:2003,dataprep.out$Y1plot,
     type="l",ylim=c(0,50000),col="black",lty="solid",
     ylab ="per-capita GDP (PPP, 2002 USD)",
     xlab ="year",
     xaxs = "i", yaxs = "i",
     lwd=2
     )
lines(1960:2003,synthY0,col="black",lty="dashed",lwd=2)
abline(v=1990,lty="dotted")

legend(x="bottomright",
       legend=c("West Germany","synthetic West Germany")
      ,lty=c("solid","dashed"),col=c("black","black")
      ,cex=.8,bg="white",lwd=c(2,2))
arrows(1987,Text.height,1989,Text.height,col="black",length=.1)
text(1982.5,Text.height,"reunification",cex=Cex.set)
```




```
### Figure 3: Per-Capita GDP Gap Between West Germany and Synthetic West Germany
#pdf(file = "ger_vs_synthger_gaps2.pdf", width = 5.5, height = 5.5, family = "Times",pointsize = 12)
gap <- dataprep.out$Y1-(dataprep.out$Y0%*%synth.out$solution.w)
plot(1960:2003,gap,
     type="l",ylim=c(-4500,4500),col="black",lty="solid",
     ylab =c("gap in per-capita GDP (PPP, 2002 USD)"),
     xlab ="year",
     xaxs = "i", yaxs = "i",
     lwd=2
     )
abline(v=1990,lty="dotted")
abline(h=0,lty="dotted")
arrows(1987,1000,1989,1000,col="black",length=.1)
text(1982.5,1000,"reunification",cex=Cex.set)
```




```
#dev.off()

### Figure 4: Placebo Reunification 1975 - Trends in Per-Capita GDP: West Germany vs. Synthetic West Ge

# data prep for training model
dataprep.out <-
  dataprep(
    foo = d,
    predictors    = c("gdp","trade","infrate"),
    dependent     = "gdp",
    unit.variable = 1,
    time.variable = 3,
    special.predictors = list(
      list("industry",1971, c("mean")),
      list("schooling",c(1960,1965), c("mean")),
      list("invest60" ,1980, c("mean"))
    ),
    treatment.identifier = 7,
    controls.identifier = unique(d$index)[-7],
    time.predictors.prior = 1960:1964,
    time.optimize.ssr = 1965:1975,
    unit.names.variable = 2,
    time.plot = 1960:1990
  )
```

```
##
##  Missing data- treated unit; predictor: infrate ; for period: 1960
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
##  Missing data - control unit: 1 ; predictor: infrate ; for period: 1960
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
```

```r
# fit training model
synth.out <- synth(
  data.prep.obj=dataprep.out,
  Margin.ipop=.005,Sigf.ipop=7,Bound.ipop=6
)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1125.569
##
## solution.v:
##  0.6838005 0.00538575 0.3035056 0.001216499 0.005525237 0.0005664583
##
## solution.w:
##  1.7321e-06 9.16e-08 0.1319977 3.65e-08 3.93e-08 4.98e-08 5.49e-08 2.75e-08 2.39e-08 0.1044041 0.115
```

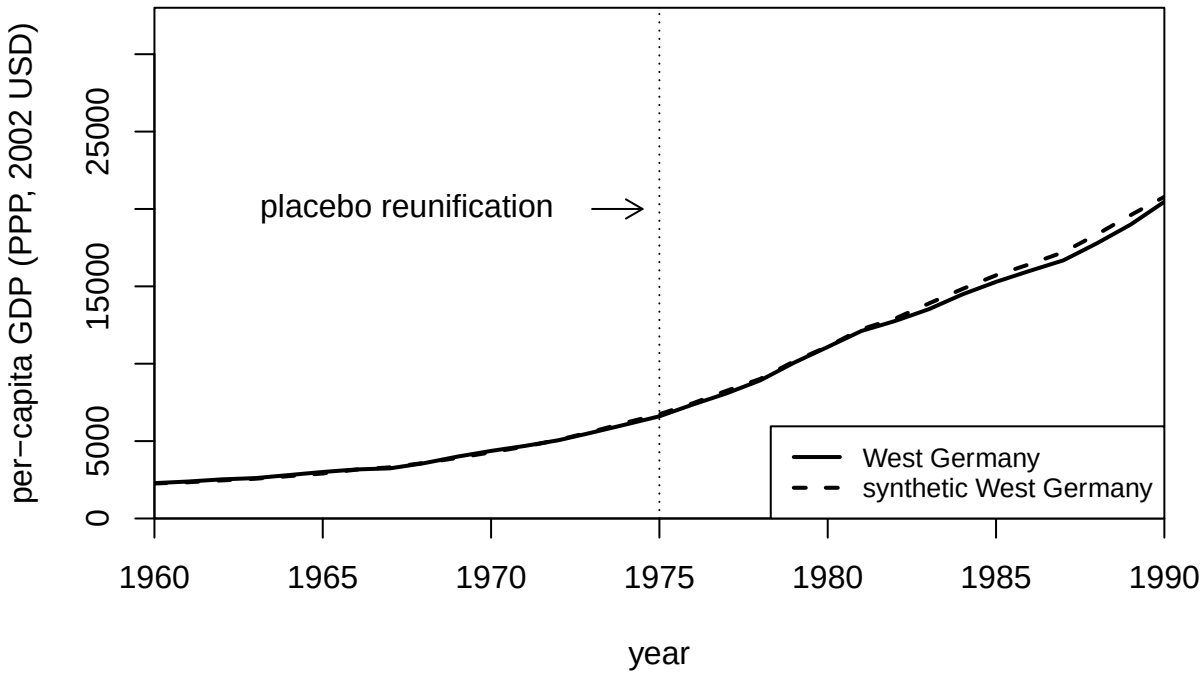
```r
# data prep for main model
dataprep.out <-
  dataprep(
    foo = d,
    predictors    = c("gdp","trade","infrate"),
    dependent     = "gdp",
    unit.variable = 1,
    time.variable = 3,
    special.predictors = list(
      list("industry" ,1971:1975, c("mean")),
      list("schooling",c(1970,1975), c("mean")),
      list("invest70" ,1980, c("mean"))
    ),
    treatment.identifier = 7,
    controls.identifier = unique(d$index)[-7],
    time.predictors.prior = 1965:1975,
    time.optimize.ssr = 1960:1975,
    unit.names.variable = 2,
    time.plot = 1960:1990
  )

# fit main model
synth.out <- synth(
  data.prep.obj=dataprep.out,
  custom.v=as.numeric(synth.out$solution.v)
)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
```

```
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 5082.521
##
## solution.v:
##  0.6838005 0.00538575 0.3035056 0.001216499 0.005525237 0.0005664583
##
## solution.w:
##  0.4294917 2.7e-08 0.5705054 1.17e-08 5.03e-08 1.101e-07 7.96e-08 1.071e-07 8.28e-08 2.211e-06 3.42e
```

```
Cex.set <- 1
#pdf(file = "2intimeplacebo1975.pdf", width = 5.5, height = 5.5, family = "Times",pointsize = 12)
plot(1960:1990,dataprep.out$Y1plot,
     type="l",ylim=c(0,33000),col="black",lty="solid",
     ylab ="per-capita GDP (PPP, 2002 USD)",
     xlab ="year",
     xaxs = "i", yaxs = "i",
     lwd=2
     )
lines(1960:1990,(dataprep.out$Y0%*%synth.out$solution.w),col="black",lty="dashed",lwd=2)
abline(v=1975,lty="dotted")
legend(x="bottomright",
       legend=c("West Germany","synthetic West Germany")
      ,lty=c("solid","dashed"),col=c("black","black")
      ,cex=.8,bg="white",lwd=c(2,2))
arrows(1973,20000,1974.5,20000,col="black",length=.1)
text(1967.5,20000,"placebo reunification",cex=Cex.set)
```




```
#dev.off()


### Figure 5: Ratio of post-reunification RMSPE to pre-reunification RMSPE: West Germany and control cou

# loop across control units
storegaps <-
  matrix(NA,
         length(1960:2003),
         length(unique(d$index))-1
         )
rownames(storegaps) <- 1960:2003
i <- 1
co <- unique(d$index)

for(k in unique(d$index)[-7]){

  # data prep for training model
  dataprep.out <-
    dataprep(
      foo = d,
      predictors    = c("gdp","trade","infrate"),
      dependent     = "gdp",
      unit.variable = 1,
      time.variable = 3,
      special.predictors = list(
        list("industry",1971:1980, c("mean")),
        list("schooling"   ,c(1970,1975), c("mean")),
        list("invest70" ,1980, c("mean"))
      ),
      treatment.identifier = k,
      controls.identifier = co[-which(co==k)],
```

```
      time.predictors.prior = 1971:1980,
      time.optimize.ssr = 1981:1990,
      unit.names.variable = 2,
      time.plot = 1960:2003
    )

  # fit training model
  synth.out <-
   synth(
    data.prep.obj=dataprep.out,
    Margin.ipop=.005,Sigf.ipop=7,Bound.ipop=6
   )

  # data prep for main model
dataprep.out <-
    dataprep(
      foo = d,
      predictors    = c("gdp","trade","infrate"),
      dependent     = "gdp",
      unit.variable = 1,
      time.variable = 3,
      special.predictors = list(
        list("industry" ,1981:1990, c("mean")),
        list("schooling",c(1980,1985), c("mean")),
        list("invest80" ,1980, c("mean"))
      ),
      treatment.identifier = k,
      controls.identifier = co[-which(co==k)],
      time.predictors.prior = 1981:1990,
      time.optimize.ssr = 1960:1989,
      unit.names.variable = 2,
      time.plot = 1960:2003
    )

# fit main model
synth.out <- synth(
   data.prep.obj=dataprep.out,
   custom.v=as.numeric(synth.out$solution.v)
  )

 storegaps[,i] <-
   dataprep.out$Y1-
   (dataprep.out$Y0%*%synth.out$solution.w)
 i <- i + 1
} # close loop over control units
```

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

```
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1072491
##
## solution.v:
##  0.001226632 0.0003302284 0.9960331 0.001075348 0.0001553649 0.001179333
##
## solution.w:
##  0.3244282 7.59e-08 5.2e-09 6.14411e-05 7.86e-08 2.46e-08 2.6e-09 1.54e-07 2.77e-08 0.6755062 1.04e-
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1309364
##
## solution.v:
##  0.001226632 0.0003302284 0.9960331 0.001075348 0.0001553649 0.001179333
##
## solution.w:
##  0.09743694 0.0002960551 0.0002019896 0.4524152 0.0002934312 0.1574081 6.07364e-05 5.5303e-06 0.0001
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 30704.76
```

```
##
## solution.v:
##  0.2910704 0.02523752 0.5102239 0.001698983 0.09146073 0.08030842
##
## solution.w:
##  0.4524534 2.29e-08 2.265e-07 1.37e-08 8.3e-09 1.12e-08 6.9e-09 4.98e-08 7.6e-09 4.8e-09 2.7e-09 5.4
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## *****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## *****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## *****************
## *****************
## *****************
##
## MSPE (LOSS V): 133979.9
##
## solution.v:
##  0.2910704 0.02523752 0.5102239 0.001698983 0.09146073 0.08030842
##
## solution.w:
##  0.1043831 7.4425e-06 1.77736e-05 0.003570288 1.02275e-05 7.7786e-06 1.32058e-05 0.4686202 3.9948e-0
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## *****************
##  searching for synthetic control unit
##
##
## *****************
## *****************
## *****************
##
## MSPE (LOSS V): 14172.72
##
## solution.v:
##  0.1651224 0.02232987 0.09183286 0.002084313 0.2191919 0.4994386
##
## solution.w:
```

```
##  6.9e-09 1.3e-09 0.2312954 3.15e-08 2.49e-08 0.6278547 1.58e-08 5.618e-07 2.26817e-05 1.8e-09 6.83e-
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 397015.6
##
## solution.v:
##  0.1651224 0.02232987 0.09183286 0.002084313 0.2191919 0.4994386
##
## solution.w:
##  0.01473104 9.03921e-05 3.22475e-05 4.89608e-05 3.5246e-06 0.8907981 3.5705e-06 0.001312142 5.41e-08
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 3070.1
##
## solution.v:
##  0.111522 0.0002241262 0.07408773 0.0207066 0.7916343 0.001825173
##
## solution.w:
##  2.04e-08 1.37e-08 2.557e-07 1.67e-08 0.3879134 0.2579754 5.9e-09 4.29e-08 3.6812e-06 0.1288711 8.28
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 162759.3
##
## solution.v:
##  0.111522 0.0002241262 0.07408773 0.0207066 0.7916343 0.001825173
##
## solution.w:
##  4.8815e-06 3.0789e-06 2.2815e-06 0.0001447455 0.0004003712 3.6288e-06 8.612e-07 0.1273755 2.409e-06
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 22596.94
##
## solution.v:
##  0.3598013 0.005365139 0.1833962 0.3456567 0.1053842 0.0003964741
##
## solution.w:
##  0.4407952 8e-10 1.3e-09 1.8e-09 5.2e-09 4e-10 1.2e-09 6.3e-09 3.805e-07 4e-09 6e-10 0.02147721 1.61
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
```

```
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 131436.3
##
## solution.v:
##  0.3598013 0.005365139 0.1833962 0.3456567 0.1053842 0.0003964741
##
## solution.w:
##  0.6923919 6.72e-08 3.002e-07 1.49623e-05 4.8975e-05 5.724e-07 1.206e-07 0.001086496 5.22e-08 1.149e
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 18271.5
##
## solution.v:
##  0.007906516 0.009637483 0.4108627 0.228974 0.09711018 0.2455091
##
## solution.w:
##  0.06359369 2.037e-07 1.3351e-06 5.006e-07 2.159e-07 0.1288059 1.422e-07 2.929e-07 7.754e-07 0.32879
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
```

```
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 355863.2
##
## solution.v:
##  0.007906516 0.009637483 0.4108627 0.228974 0.09711018 0.2455091
##
## solution.w:
##  0.0006928859 0.0002105257 0.000133043 0.000210058 0.4857387 0.0002191833 0.001282727 0.0002064541 0
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 90302.79
##
## solution.v:
##  0.3598129 0.3654282 7.07594e-05 0.2599986 7.872e-07 0.01468885
##
## solution.w:
##  0.0006345504 0.3873168 0.002748266 0.01408356 0.002346317 0.001837628 0.002128399 0.005448973 0.004
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
```

```
## ****************
## ****************
##
## MSPE (LOSS V): 31074.81
##
## solution.v:
##  0.3598129 0.3654282 7.07594e-05 0.2599986 7.872e-07 0.01468885
##
## solution.w:
##  0.1211999 0.199951 0.02571479 0.01369043 0.02804747 0.03449484 0.03230165 0.019118 0.02407076 0.024
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 11314.57
##
## solution.v:
##  0.03274359 0.585266 0.1900353 8.64e-08 0.113296 0.07865896
##
## solution.w:
##  1.4e-09 2.5e-09 1.034e-07 0.6988393 0.02034418 4.7e-09 0.1889969 3.9e-09 0.0002822088 2.879e-07 1.45
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 109944.6
##
```

```
## solution.v:
##  0.03274359 0.585266 0.1900353 8.64e-08 0.113296 0.07865896
##
## solution.w:
##  0.06299059 0.00162648 0.2269907 0.6170047 2.17415e-05 3.977e-06 0.09134258 2.3332e-06 1.4919e-06 5.5
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 103415.3
##
## solution.v:
##  5.3029e-06 0.003018781 2.86765e-05 0.9848472 0.004021791 0.00807825
##
## solution.w:
##  4.556e-07 3.312e-07 1.55e-07 0.1087954 0.5611106 2.1617e-06 4.72e-08 1.9928e-06 5.0532e-06 2.8487e-0
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 357564
##
## solution.v:
##  5.3029e-06 0.003018781 2.86765e-05 0.9848472 0.004021791 0.00807825
##
## solution.w:
##  0.0002404334 0.114445 0.0009578937 0.1294019 0.0001605345 0.0002268874 0.0004438236 0.0002581625 0.0
```

```
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1762873
##
## solution.v:
##  0.7774543 0.01261047 0.1187674 0.01253333 0.01800161 0.06063288
##
## solution.w:
##  1 2e-10 1.3e-09 9e-10 3.3e-09 9e-10 1.25e-08 4e-10 3.4e-09 9e-10 9e-10 2e-10 1e-10 1e-10 1.1e-09 4e
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1377205
##
## solution.v:
##  0.7774543 0.01261047 0.1187674 0.01253333 0.01800161 0.06063288
##
## solution.w:
##  0.9997446 4.545e-07 8.8965e-06 2.3831e-06 1.3594e-06 4.684e-06 6.44324e-05 2.7553e-06 2.0051e-06 5.3
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
```

```
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 137578.6
##
## solution.v:
##  0.1010769 0.8300357 0.06839542 0.0001059198 0.0001735503 0.0002124574
##
## solution.w:
##  0.4320866 2.77e-08 4.6e-09 3.3e-09 2.08e-08 1.43e-08 0.05327324 3.42e-08 7.5e-09 1.24e-08 1.49e-08 0
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1582150
##
## solution.v:
##  0.1010769 0.8300357 0.06839542 0.0001059198 0.0001735503 0.0002124574
##
## solution.w:
##  0.6299095 2.079e-07 9.69e-08 1.1e-09 5.96e-08 1.721e-07 2.575e-07 3.27e-08 3.48e-08 3.95e-08 4.19e-0
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
```

```
## ****************
## ****************
##
## MSPE (LOSS V): 446470
##
## solution.v:
##  0.03113245 0.03990278 0.002136156 0.7075158 0.1876985 0.0316143
##
## solution.w:
##  7.696e-07 5.928e-07 5.75e-07 5.671e-07 0.1889267 5.0752e-06 2.257e-07 3.1533e-06 7.14e-07 5.0023e-0
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 436704.6
##
## solution.v:
##  0.03113245 0.03990278 0.002136156 0.7075158 0.1876985 0.0316143
##
## solution.w:
##  0.0008246341 0.0001380339 4.79986e-05 2.51858e-05 0.3775699 0.0006116754 9.55349e-05 0.0003024378 5
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 3206068
##
```

```
## solution.v:
##  0.01574527 0.00811563 0.003730266 0.9418119 0.0305029 9.40572e-05
##
## solution.w:
##  9.16e-08 1.157e-06 3.895e-07 1.4613e-06 9e-10 2.26e-07 1.406e-07 7.274e-07 3.54e-07 1.542e-07 1.056
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 1953909
##
## solution.v:
##  0.01574527 0.00811563 0.003730266 0.9418119 0.0305029 9.40572e-05
##
## solution.w:
##  2.5281e-06 4.355e-06 4.5114e-06 0.0003060004 8.6832e-06 4.9272e-06 3.8079e-06 4.3609e-06 2.16908e-05
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 4035.369
##
## solution.v:
##  0.1682138 0.005107489 0.01198158 0.03764541 0.733912 0.04313974
##
## solution.w:
##  1e-10 2e-10 1e-10 3e-10 1e-10 7e-10 0 0.3562438 1e-10 3e-10 3e-10 3e-10 4.10236e-05 0.6437152 1e-10
```

```
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 271197.2
##
## solution.v:
##  0.1682138 0.005107489 0.01198158 0.03764541 0.733912 0.04313974
##
## solution.w:
##  9e-10 0.130913 3.3e-09 1.867e-07 2.94e-08 8.15e-08 1.05e-08 0.1461515 3.78e-08 7.31e-08 7.6e-09 1.25
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 94500.41
##
## solution.v:
##  0.0001689119 1.81979e-05 0.6106232 0.005399943 0.3837889 8.05e-07
##
## solution.w:
##  0.2473217 0.1443421 0.0009493286 0.0004565676 0.000688322 0.001044955 0.1338865 0.005100402 0.00058
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
```

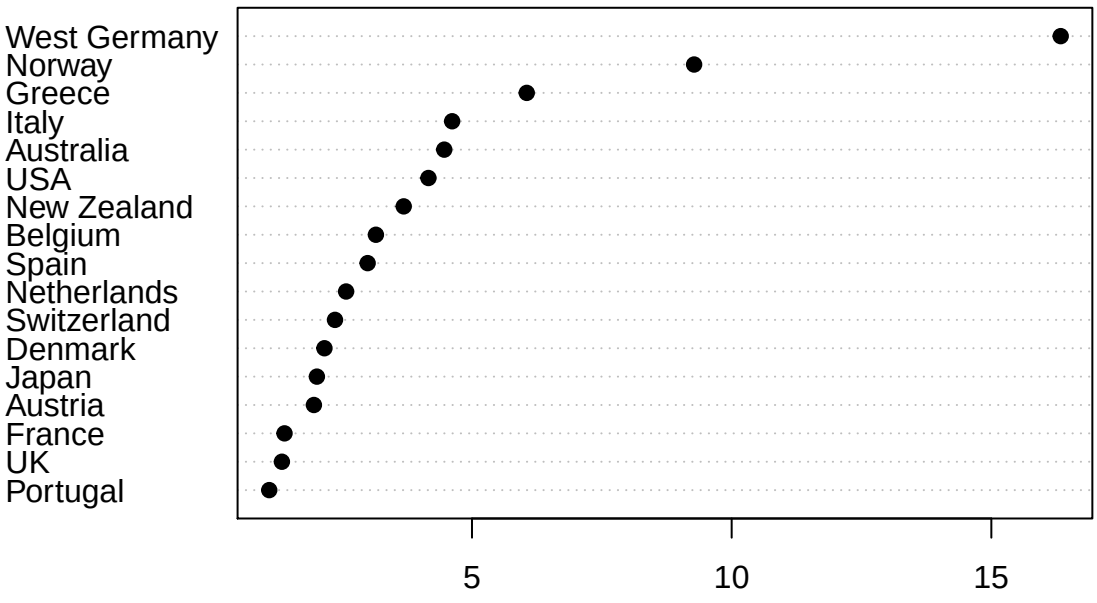
```
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 210254.2
##
## solution.v:
##  0.0001689119 1.81979e-05 0.6106232 0.005399943 0.3837889 8.05e-07
##
## solution.w:
##  0.1125022 0.01755314 0.1701483 0.01447702 0.0182226 0.01658635 0.02556025 0.01846707 0.01611229 0.0
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 497090
##
## solution.v:
##  4.6e-08 1.98255e-05 0.6727525 0.3272085 1.90842e-05 9.3e-09
##
## solution.w:
##  2.22652e-05 9.59796e-05 0.0007065426 0.001111175 0.3403507 0.000735125 0.0001271146 0.0005876782 0.
##
##
##  Missing data - control unit: 7 ; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
```

```
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 412006.1
##
## solution.v:
##  4.6e-08 1.98255e-05 0.6727525 0.3272085 1.90842e-05 9.3e-09
##
## solution.w:
##  0.0627752 0.01843601 0.02949969 0.03934955 0.1241466 0.05079127 0.03121416 0.03925164 0.03888327 0.0
```

```r
d <- d[order(d$index,d$year),]
colnames(storegaps) <- unique(d$country)[-7]
storegaps <- cbind(gap,storegaps)
colnames(storegaps)[1] <- c("West Germany")

# compute ratio of post-reunification RMSPE
# to pre-reunification RMSPE
rmse <- function(x){sqrt(mean(x^2))}
preloss <- apply(storegaps[1:30,],2,rmse)
postloss <- apply(storegaps[31:44,],2,rmse)

#pdf("2ratio_post_to_preperiod_rmse2a.pdf")
dotchart(sort(postloss/preloss),
         xlab="Post-Period RMSE / Pre-Period RMSE",
         pch=19)
```




```
#dev.off()

### Figure 6: Leave-one-out distribution of the synthetic control for West Germany

# loop over leave one outs
storegaps <-
  matrix(NA,
         length(1960:2003),
         5)
colnames(storegaps) <- c(1,3,9,12,14)
co <- unique(d$index)[-7]

for(k in 1:5){

# data prep for training model
omit <- c(1,3,9,12,14)[k]
  dataprep.out <-
    dataprep(
      foo = d,
      predictors    = c("gdp","trade","infrate"),
      dependent     = "gdp",
      unit.variable = 1,
      time.variable = 3,
      special.predictors = list(
        list("industry",1971:1980, c("mean")),
        list("schooling"   ,c(1970,1975), c("mean")),
        list("invest70" ,1980, c("mean"))
      ),
      treatment.identifier = 7,
      controls.identifier = co[-which(co==omit)],
      time.predictors.prior = 1971:1980,
      time.optimize.ssr = 1981:1990,
```

```
      unit.names.variable = 2,
      time.plot = 1960:2003
    )

  # fit training model
  synth.out <- synth(
    data.prep.obj=dataprep.out,
    Margin.ipop=.005,Sigf.ipop=7,Bound.ipop=6
  )

# data prep for main model
dataprep.out <-
  dataprep(
    foo = d,
    predictors    = c("gdp","trade","infrate"),
    dependent     = "gdp",
    unit.variable = 1,
    time.variable = 3,
    special.predictors = list(
      list("industry" ,1981:1990, c("mean")),
      list("schooling",c(1980,1985), c("mean")),
      list("invest80" ,1980, c("mean"))
    ),
    treatment.identifier = 7,
    controls.identifier = co[-which(co==omit)],
    time.predictors.prior = 1981:1990,
    time.optimize.ssr = 1960:1989,
    unit.names.variable = 2,
    time.plot = 1960:2003
  )

  # fit main model
  synth.out <- synth(
    data.prep.obj=dataprep.out,
    custom.v=as.numeric(synth.out$solution.v)
  )
  storegaps[,k] <- (dataprep.out$Y0%*%synth.out$solution.w)
} # close loop over leave one outs
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 7178.77
##
## solution.v:
```

```
##  0.2285184 0.4851792 0.22103 1.7349e-06 0.0009094366 0.06436127
##
## solution.w:
##  1.901e-07 0.3606603 6.77e-08 1.344e-07 7.593e-07 1.219e-07 8.95e-08 7.81e-08 0.2926267 0.2401705 1.3
##
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 46022.33
##
## solution.v:
##  0.2285184 0.4851792 0.22103 1.7349e-06 0.0009094366 0.06436127
##
## solution.w:
##  0.220306 0.0003977769 9.22888e-05 0.0002399268 0.0001557244 8.64815e-05 0.1189366 5.58233e-05 0.376
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 3735.194
##
## solution.v:
##  0.307833 0.04295867 0.3766373 0.0002337457 0.2377632 0.03457411
##
## solution.w:
##  0.1886962 5.5e-09 1.34e-08 2.1e-08 1.54e-08 3.6e-09 0.5599955 3.52e-08 0.1546483 0.09665981 3.9e-09
##
```

```
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 67986.59
##
## solution.v:
##  0.307833 0.04295867 0.3766373 0.0002337457 0.2377632 0.03457411
##
## solution.w:
##  0.3949745 1.4724e-06 2.6917e-06 4.0437e-06 2.4983e-06 1.9982e-06 0.2998465 1.11946e-05 0.08900743 0
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 4927.651
##
## solution.v:
##  0.4845651 0.1268982 0.1112233 0.01162803 0.1582388 0.1074467
##
## solution.w:
##  0.1595804 1.26e-08 0.5841454 4.2e-09 5e-08 1.69e-08 1.15e-08 1.51e-08 0.1369613 0.119307 8.6e-09 6e
##
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
```

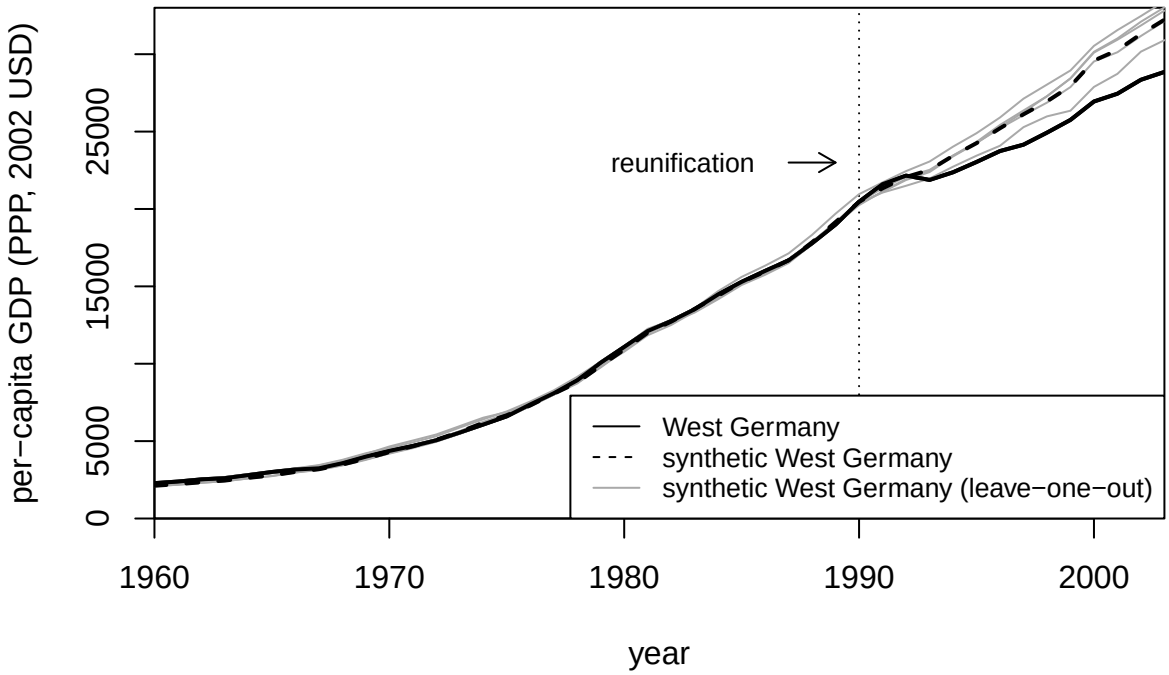
```
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 22481.3
##
## solution.v:
##  0.4845651 0.1268982 0.1112233 0.01162803 0.1582388 0.1074467
##
## solution.w:
##  0.2247631 8.12697e-05 0.4721386 0.04815105 0.0002917513 0.0001351167 6.2233e-05 5.42047e-05 0.10104
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 9464.114
##
## solution.v:
##  0.3522977 0.3785275 0.1242839 0.04639133 0.09849836 1.2153e-06
##
## solution.w:
##  0.31027 2.2e-09 0.6794631 0 2e-10 2.4e-09 1.9e-09 0 4e-10 0.01026683 1e-09 6e-10 2.1e-09 6.4e-09 7e
##
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
```

```
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 29453.09
##
## solution.v:
##  0.3522977 0.3785275 0.1242839 0.04639133 0.09849836 1.2153e-06
##
## solution.w:
##  0.3200134 1.28074e-05 0.3902975 9.18241e-05 4.6507e-06 7.9132e-06 8.6774e-06 0.1741099 2.87242e-05 (
##
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 7854.055
##
## solution.v:
##  0.6279978 0.02947095 0.08808591 0.008168683 0.2449758 0.001300786
##
## solution.w:
##  0.2322202 1.55e-08 0.711551 1.75e-08 3.84e-08 1.47e-08 1.11e-08 1.81e-08 1.56e-08 0.05622816 8.4e-09
##
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
```

```
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 5520.717
##
## solution.v:
##  0.6279978 0.02947095 0.08808591 0.008168683 0.2449758 0.001300786
##
## solution.w:
##  0.2918854 0.05658427 0.416006 1.34009e-05 1.50928e-05 0.001309335 2.60145e-05 0.1783121 2.57615e-05
```

```
Text.height <- 23000
Cex.set <- .8
#pdf(file = "1jackknife2.pdf", width = 5.5, height = 5.5, family = "Times",pointsize = 12)
plot(1960:2003,dataprep.out$Y1plot,
     type="l",ylim=c(0,33000),col="black",lty="solid",
     ylab ="per-capita GDP (PPP, 2002 USD)",
     xlab ="year",
     xaxs = "i", yaxs = "i",lwd=2
     )

abline(v=1990,lty="dotted")
arrows(1987,23000,1989,23000,col="black",length=.1)
 for(i in 1:5){
  lines(1960:2003,storegaps[,i],col="darkgrey",lty="solid")
  }
lines(1960:2003,synthY0,col="black",lty="dashed",lwd=2)
lines(1960:2003,dataprep.out$Y1plot,col="black",lty="solid",lwd=2)
text(1982.5,23000,"reunification",cex=.8)
legend(x="bottomright",
       legend=c("West Germany",
                "synthetic West Germany",
                "synthetic West Germany (leave-one-out)")
      ,lty=c("solid","dashed","solid"),
      col=c("black","black","darkgrey")
      ,cex=.8,bg="white",lwdc(2,2,1))
```




```
#dev.off()


### Table 3: Synthetic Weights from Combinations of Control Countries
library(gtools)
library(kernlab)

# data prep for training model
dataprep.out <-
  dataprep(
    foo = d,
    predictors    = c("gdp","trade","infrate"),
    dependent     = "gdp",
    unit.variable = 1,
    time.variable = 3,
    special.predictors = list(
      list("industry", 1971:1980, c("mean")),
      list("schooling",c(1970,1975), c("mean")),
      list("invest70" ,1980, c("mean"))
    ),
    treatment.identifier = 7,
    controls.identifier = unique(d$index)[-7],
    time.predictors.prior = 1971:1980,
    time.optimize.ssr = 1981:1990,
    unit.names.variable = 2,
    time.plot = 1960:2003
  )

# fit training model
synth.out <-
  synth(
    data.prep.obj=dataprep.out,
```

```
    Margin.ipop=.005,Sigf.ipop=7,Bound.ipop=6
  )
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
##
##
## ****************
##  searching for synthetic control unit
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 4580.211
##
## solution.v:
##  0.5885675 0.1037433 0.04737192 0.005338085 0.08311456 0.1718646
##
## solution.w:
##  0.1350128 2.598e-07 0.5073743 2.256e-07 1.0906e-06 2.752e-07 2.15e-07 8.495e-07 2.471e-07 0.1659188
```

```
# data prep for main model
dataprep.out <-
  dataprep(
    foo = d,
    predictors    = c("gdp","trade","infrate"),
    dependent     = "gdp",
    unit.variable = 1,
    time.variable = 3,
    special.predictors = list(
      list("industry" ,1981:1990, c("mean")),
      list("schooling",c(1980,1985), c("mean")),
      list("invest80" ,1980, c("mean"))
    ),
    treatment.identifier = 7,
    controls.identifier = unique(d$index)[-7],
    time.predictors.prior = 1981:1990,
    time.optimize.ssr = 1960:1989,
    unit.names.variable = 2,
    time.plot = 1960:2003
  )
```

```
##
##  Missing data: treated unit; special predictor: special.industry.1981.1990 ; for period: 1990
##  We ignore (na.rm = TRUE) all missing values for predictors.op.
```

```
# fit main model with v from training model
synth.out <- synth(
  data.prep.obj=dataprep.out,
  custom.v=as.numeric(synth.out$solution.v)
)
```

```
##
## X1, X0, Z1, Z0 all come directly from dataprep object.
```

```
##
##
## ****************
##  optimization over w weights: computing synthtic control unit
##
##
##
## ****************
##  v weights supplied manually: computing synthtic control unit
##
##
##
## ****************
## ****************
## ****************
##
## MSPE (LOSS V): 14178.38
##
## solution.v:
##  0.5885675 0.1037433 0.04737192 0.005338085 0.08311456 0.1718646
##
## solution.w:
##  0.2186415 0.001006986 0.4206785 0.001188088 0.00114549 0.0009857056 0.000634122 0.08734495 0.0006968
```

```
synth.tables <- synth.tab(
  dataprep.res = dataprep.out,
  synth.res = synth.out
)

table3 <- list()
synth.tables$tab.w[,1] <- round(synth.tables$tab.w[,1],2)
table3[[5]] <-synth.tables$tab.w[order(-1*synth.tables$tab.w[,1]),2:1][1:5,]

# compute loss for all combinations
# of 4, 3, 2, 1 sized donor pools

# get W and v
solution.w <- round(synth.out$solution.w,3)
V <- diag(as.numeric(synth.out$solution.v))

# compute scaled Xs
nvarsV <- dim(dataprep.out$X0)[1]
big.dataframe <- cbind(dataprep.out$X0, dataprep.out$X1)
divisor <- sqrt(apply(big.dataframe, 1, var))
scaled.matrix <-
  t(t(big.dataframe) %*% ( 1/(divisor) *
  diag(rep(dim(big.dataframe)[1], 1)) ))
X0.scaled <- scaled.matrix[,c(1:(dim(dataprep.out$X0)[2]))]
X1.scaled <- as.matrix(scaled.matrix[,dim(scaled.matrix)[2]])

dn <- d[d$year==1970,c("country","index")]
dn <- dn[order(dn$index),]
dn <- dn[-7,]
```

```
table2store <- matrix(NA,nrow(dataprep.out$X1),4)
fig7store   <- matrix(NA,length(1960:2003),4)

# loop through number of controls
for(pp in 4:1){
  store       <- combinations(length(unique(d$index)[-7]),
                              r=pp, v=unique(d$index)[-7])
  store.loss  <- matrix(NA,nrow=nrow(store),1)
  store.w     <- matrix(NA,nrow=nrow(store),pp)
  store.c     <- store.w

# loop through combinations
  for(k in 1:nrow(store)){
    # index positions of control units
    posvector <- c()
    for(i in 1:pp){
      posvector <- c(posvector,which(dn$index==store[k,i]))
    }

  # run quad optimization
    X0temp <- X0.scaled[ , posvector ]
    H <- t(X0temp) %*% V %*% (X0temp)
    c <- -1*c(t(X1.scaled) %*% V %*% (X0temp) )

    if(pp == 1){
      solution.w <- matrix(1)
    } else {
        res <- ipop(c = c, H = H, A = t(rep(1, length(c))),
                    b = 1, l = rep(0, length(c)),
                    u = rep(1, length(c)), r = 0,
                    margin = 0.005,sigf = 7, bound = 6)
        solution.w <- as.matrix(primal(res))
    }
    loss.w <- t(X1.scaled - X0temp %*% solution.w) %*% V %*% (X1.scaled - X0temp %*% solution.w)

    store.loss[k] <- loss.w
    store.w[k,]   <- t(solution.w)
    store.c[k,]   <- dn$country[posvector]
  } # close loop over combinations

# get best fitting combination
  dat <- data.frame(store.loss,
                    store,
                    store.c,
                    store.w
                    )
  colnames(dat) <- c("loss",
                     paste("CNo",1:pp,sep=""),
                     paste("CNa",1:pp,sep=""),
                     paste("W",1:pp,sep="")
                     )
  dat <- dat[order(dat$loss),]
  Countries <- dat[1,paste("CNo",1:pp,sep="")]
```

```
  Cweights  <- as.numeric(dat[1,paste("W",1:pp,sep="")])

  outdat <-  data.frame(unit.names=as.vector(
   (t(as.vector(dat[1,paste("CNa",1:pp,sep="")])))),
                    w.weights=round(Cweights,2))

table3[[pp]]<- outdat[order(-1*outdat$w.weights),]

 # get posvector for fitting
  posvector <- c()
  if(pp == 1 ){
    posvector <- c(posvector,which(dn$index==Countries))
  } else {
    for(i in 1:pp){
      posvector <- c(posvector,which(dn$index==Countries[1,i]))
    }
  }

  X0t <- as.matrix(dataprep.out$X0[,posvector])%*% as.matrix(Cweights)
  table2store[,(4:1)[pp]] <- X0t

  fig7store[,(4:1)[pp]] <-
   dataprep.out$Y0[,posvector]%*%as.matrix(Cweights)

} # close loop over number of countries

# Table 3
table3
```

```
## [[1]]
##   unit.names w.weights
## 1    Austria         1
##
## [[2]]
##   unit.names w.weights
## 2    Austria      0.74
## 1        USA      0.26
##
## [[3]]
##   unit.names w.weights
## 2    Austria      0.57
## 1        USA      0.29
## 3      Japan      0.14
##
## [[4]]
##    unit.names w.weights
## 2     Austria      0.56
## 1         USA      0.22
## 4       Japan      0.13
## 3 Switzerland      0.10
##
## [[5]]
##     unit.names w.weights
## 3      Austria      0.42
```

```
## 1          USA      0.22
## 14       Japan      0.15
## 12 Switzerland      0.11
## 9  Netherlands      0.09
```

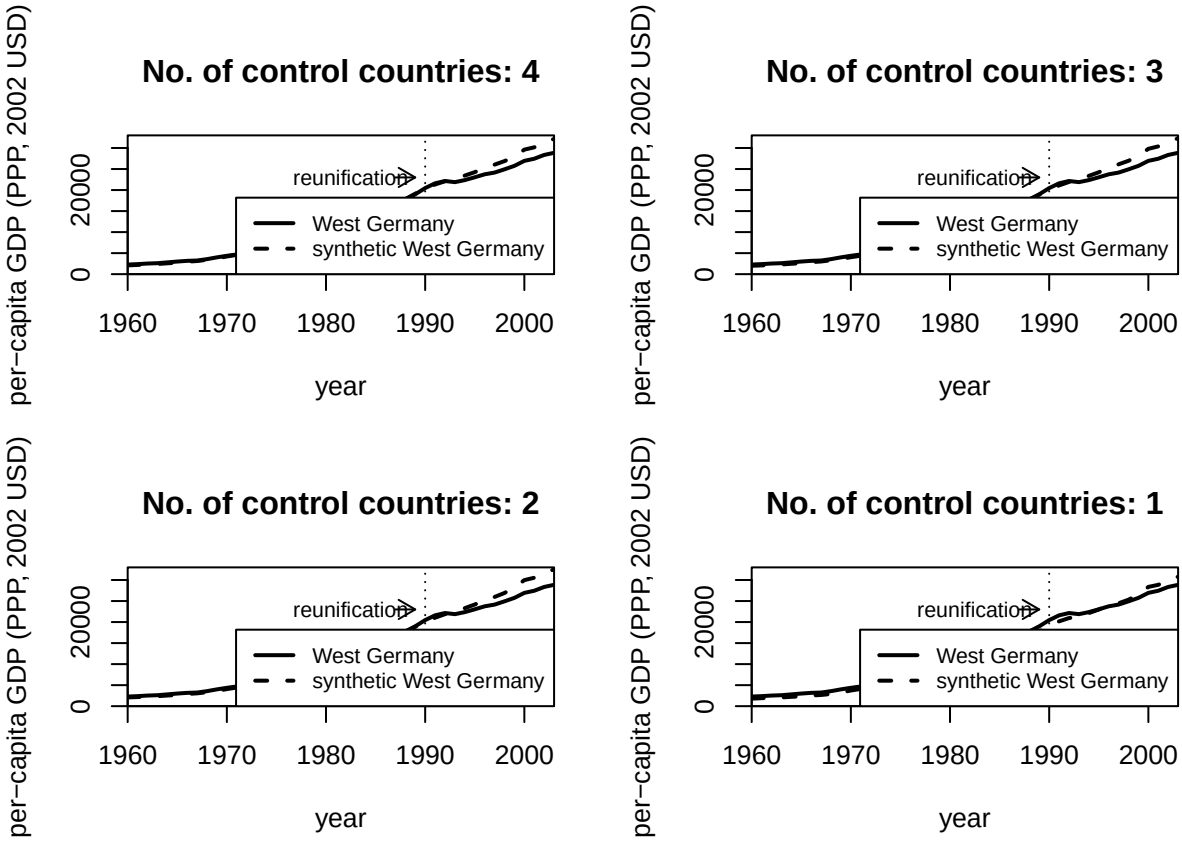
```
# Table 4
synth.tables$tab.pred[,3] <- c(8021.1,31.9,7.4,34.2,44.1,25.9)
table4 <- round(
  cbind(synth.tables$tab.pred[,1:2],
        table2store,
        synth.tables$tab.pred[,3]),1)
rownames(table4) <- c("GDP per-capita","Trade openness",
                      "Inflation rate","Industry share",
                      "Schooling","Investment rate")
colnames(table4)[2:7] <- c(5:1,"OECD Sample")
table4
```

```
##                 Treated       5       4       3       2       1
## GDP per-capita  15808.9 15804.6 15805.2 15576.3 15649.3 14817.0
## Trade openness     56.8    56.9    55.9    51.5    60.3    74.6
## Inflation rate      2.6     3.5     3.6     3.7     3.8     3.5
## Industry share     34.5    34.4    34.6    34.7    34.2    35.5
## Schooling          55.5    55.2    57.6    57.8    60.7    60.9
## Investment rate    27.0    27.0    27.2    26.6    25.5    26.6
##                 OECD Sample
## GDP per-capita       8021.1
## Trade openness         31.9
## Inflation rate          7.4
## Industry share         34.2
## Schooling              44.1
## Investment rate        25.9
```

```
## Figure 7: Per-Capita GDP Gaps Between West Germany and Sparse Synthetic Controls
Text.height <- 23000
Cex.set <- .8

par(mfrow=c(2,2))
for(pp in 4:1){
#pdf(file = paste("2ger_vs_synth","CValt",pp,".pdf",sep=""), width = 5.5, height = 5.5, family = "Times
  plot(1960:2003,dataprep.out$Y1,
       type="l",ylim=c(0,33000),col="black",lty="solid",
       ylab ="per-capita GDP (PPP, 2002 USD)",
       xlab ="year",
       xaxs = "i", yaxs = "i",
       lwd=2,
       main=paste("No. of control countries: ",pp,sep="")
       )
  lines(1960:2003,fig7store[,c(4:1)[pp]],col="black",lty="dashed",lwd=2)
  abline(v=1990,lty="dotted")
  legend(x="bottomright",
         legend=c("West Germany","synthetic West Germany")
         ,lty=c("solid","dashed"),col=c("black","black")
         ,cex=.8,bg="white",lwd=c(2,2))
  arrows(1987,Text.height,1989,Text.height,col="black",length=.1)
  text(1982.5,Text.height,"reunification",cex=Cex.set)
```

```
  #dev.off()
}
```


No. of control countries: 4
No. of control countries: 3
No. of control countries: 2
No. of control countries: 1
per−capita GDP (PPP, 2002 USD)
0
20000
1960
1970
1980
1990
2000
year
reunification
West Germany
synthetic West Germany