Package 'SCtools'

June 5, 2018

Type Package

Title Tools for Synthetic Controls Analysis

Version 0.2

Date 2018-05-22

Author Bruno Castanho Silva <bcsilva@wiso.uni-koeln.de>

Maintainer Bruno Castanho Silva <bcsilva@wiso.uni-koeln.de>

Depends Synth, ggplot2, stringr, cvTools

Description A set of functions to extend the synthetic controls analyses performed by the package Synth. Includes generating and plotting placebos, significance tests and plots, and calculating average treatment effects for multiple treated units.

License GPL (>=2)

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 6.0.1

R topics documented:

SCtools-package *Tools for Synthetic Controls Analysis*

Description

A set of functions to extend the synthetic controls analyses performed by the package Synth. Includes generating and plotting placebos, significance tests and plots, and calculating average treatment effects for multiple treated units.

Details

This package was not yet installed at build time.

Index: This package was not yet installed at build time.
Package with functions to extend synthetic control model, including robustness and sensitivity tests, and multiple treated units. Currently under development.

Author(s)

Bruno Castanho Silva <bcsilva@wiso.uni-koeln.de>

Maintainer: Bruno Castanho Silva <bcsilva@wiso.uni-koeln.de>

References

Abadie, A., Diamond, A., Hainmueller, J. (2014). Comparative Politics and the Synthetic Control Method. American Journal of Political Science Forthcoming 2014.

Abadie, A., Diamond, A., Hainmueller, J. (2011). Synth: An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie A, Diamond A, Hainmueller J (2010). Synthetic Control Methods for Comparative Case Studies: Estimating the Effect of California's Tobacco Control Program. Journal of the American Statistical Association 105 (490) 493–505.

generate.placebos *Function to generate placebo synthetic controls*

Description

Constructs a synthetic control group for each unit in the donor pool of an implementation of the synthetic control method for a single treated unit. Used for placebo tests (see plot.placebos, mspe.test, mspe.plot) to assess the strength and significance of a causal inference based on the synthetic control method. On placebo tests, see Abadie and Gardeazabal (2003), and Abadie, Diamond, and Hainmueller (2010, 2011, 2014).

Usage

```
generate.placebos(dataprep.out, synth.out)
```

Arguments

dataprep.out	A data.prep object produced by the dataprep() command
synth.out	A synth.out object produced by the synth() command

Value

df	Data frame with outcome data for each control unit and their respective synthetic control and for the original treated and its control
mspe.placs	Mean squared prediction error for the pretreatment period for each placebo
t0	First time unit in time.optimize.ssr
t1	First time unit after the highest value in time.optimize.ssr
tr	Unit number of the treated unit
names.and.numbers	Dataframe with two columns showing all unit numbers and names from control units
n	Number of control units
treated.name	Unit name of the treated unit
loss.v	Pretreatment MSPE of the treated unit's synthetic control

Author(s)

Bruno Castanho Silva

References

Abadie, A., Diamond, A., Hainmueller, J. (2014). Comparative Politics and the Synthetic Control Method. American Journal of Political Science Forthcoming 2014.

Synthetic : An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie, A., Diamond, A., Hainmueller, J. (2011). Synth: An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie A, Diamond A, Hainmueller J (2010). Synthetic Control Methods for Comparative Case Studies: Estimating the Effect of California's Tobacco Control Program. Journal of the American Statistical Association 105 (490) 493–505.

Abadie, A. and Gardeazabal, J. (2003) Economic Costs of Conflict: A Case Study of the Basque Country American Economic Review 93 (1) 113–132.

Examples

```
## First prepare the required objects

# Load simulated data from Synth
data(synth.data)

# Execute dataprep to produce the necessary matrices for synth
dataprep.out<-
  dataprep(
   foo = synth.data,
   predictors = c("X1", "X2", "X3"),
   predictors.op = "mean",
   dependent = "Y",
   unit.variable = "unit.num",
   time.variable = "year",
   special.predictors = list(
      list("Y", 1991, "mean"),
      list("Y", 1985, "mean"),
      list("Y", 1980, "mean")
                            ),
   treatment.identifier = 7,
   controls.identifier = c(29, 2, 13, 17, 32, 38),
   time.predictors.prior = c(1984:1989),
   time.optimize.ssr = c(1984:1990),
   unit.names.variable = "name",
   time.plot = 1984:1996
   )

# run the synth command to create the synthetic control
synth.out <- synth(dataprep.out)

## run the generate.placebos command to reassign treatment status
## to each unit listed as control, one at a time, and generate their
## synthetic versions.
tdf <- generate.placebos(dataprep.out,synth.out)
```

`mspe.plot` *Plot the post/pre-treatment MSPE ratio*

Description

Plots the post/pre-treatment mean square prediction error ratio for the treated unit and placebos.

Usage

```
mspe.plot(tdf, discard.extreme = TRUE, mspe.limit = 20, plot.hist = FALSE,
title = NULL, xlab = "Pre/Post MSPE ratio", ylab = NULL)
```

Arguments

tdf	An object constructed by generate.placebos.
discard.extreme	Logical. Whether or not placebos with high pretreatement MSPE should be excluded from the plot.
mspe.limit	Numerical. Used if discard.extreme is TRUE. It indicates how many times the pretreatment MSPE of a placebo should be higher than that of the treated unit to be considered extreme and discarded. Default is 20.
plot.hist	Logical. If FALSE, a dotplot with each unit name and its post/pre treatment MSPE ratio is produced. If TRUE, a histogram is produced, with the frequency of each ratio. Should be set to TRUE when there are many controls, to make visualization easier.
title	Character. Optional. Title of the plot.
xlab	Character. Optional. Label of the x axis.
ylab	Character. Optional. Label of the y axis.

Details

Post/pre-treatement mean square prediction error ratio is the difference between the observed outcome of a unit and its synthetic control, before and after treatement. A higher ratio means a small pretreatment prediction error (a good synthetic control), and a high post-treatment MSPE, meaning a large difference between the unit and its synthetic control after the intervention. By calculating this ratio for all placebos, the test can be interpreted as looking at how likely the result obtained for a single treated case with a synthetic control analysis could have occurred by chance given no treatement. For more detailed description, see Abadie, Diamond, and Hainmueller (2011, 2014).

Author(s)

Bruno Castanho Silva

References

Abadie, A., Diamond, A., Hainmueller, J. (2014). Comparative Politics and the Synthetic Control Method. American Journal of Political Science Forthcoming 2014.

Abadie, A., Diamond, A., Hainmueller, J. (2011). Synth: An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie A, Diamond A, Hainmueller J (2010). Synthetic Control Methods for Comparative Case Studies: Estimating the Effect of California's Tobacco Control Program. Journal of the American Statistical Association 105 (490) 493–505.

See Also

See also generate.placebos, mspe.test, plot.placebos, synth

Examples

```
## First prepare the required objects

# Load simulated data from Synth
data(synth.data)

# Execute dataprep to produce the necessary matrices for synth
dataprep.out<-
  dataprep(
   foo = synth.data,
   predictors = c("X1", "X2", "X3"),
   predictors.op = "mean",
   dependent = "Y",
   unit.variable = "unit.num",
   time.variable = "year",
   special.predictors = list(
      list("Y", 1991, "mean"),
      list("Y", 1985, "mean"),
      list("Y", 1980, "mean")
                            ),
   treatment.identifier = 7,
   controls.identifier = c(29, 2, 13, 17, 32, 38),
   time.predictors.prior = c(1984:1989),
   time.optimize.ssr = c(1984:1990),
   unit.names.variable = "name",
   time.plot = 1984:1996
   )

# run the synth command to create the synthetic control
synth.out <- synth(dataprep.out)

## run the generate.placebos command to reassign treatment status
## to each unit listed as control, one at a time, and generate their
## synthetic versions.
tdf<-generate.placebos(dataprep.out,synth.out)

## Test how extreme was the observed treatment effect given the placebos:
ratio<-mspe.test(tdf)
ratio$p.val

## Check visually how extreme is this value in the distribution:
mspe.plot(tdf,discard.extreme=F)
```

`mspe.test` *Function to compute the post/pre treatment MSPE ratio for the treated unit and placebos*

Description

Computes the post/pre treatement mean square prediction error ratio for a treated unit in a synthetic control analysis and all placebos produced with `generate.placebos`. Returns a matrix with ra-

tios and a p-value of how extreme the treated unit's ratio is in comparison with that of placebos. Equivalent to a significance testing of a synthetic controls result.

Usage

```
mspe.test(tdf, discard.extreme = FALSE, mspe.limit = 20)
```

Arguments

`tdf`	An object constructed by `generate.placebos`.
`discard.extreme`	Logical. Whether or not placebos with high pretreatement MSPE should be excluded from the count and significance testing.
`mspe.limit`	Numerical. Used if `discard.extreme` is `TRUE`. It indicates how many times the pretreatment MSPE of a placebo should be higher than that of the treated unit to be considered extreme and discarded. Default is `20`.

Details

Post/pre-treatement mean square prediction error ratio is the difference between the observed outcome of a unit and its synthetic control, before and after treatement. A higher ratio means a small pretreatment prediction error (a good synthetic control), and a high post-treatment MSPE, meaning a large difference between the unit and its synthetic control after the intervention. By calculating this ratio for all placebos, the test can be interpreted as looking at how likely the result obtained for a single treated case with a synthetic control analysis could have occurred by chance given no treatement. For more detailed description, see Abadie, Diamond, and Hainmueller (2011, 2014).

Value

`p.val`	The p-value of the treated unit post/pre MSPE ratio. It is the proportion of units (placebos and treated) that have a ratio equal or higher that of the treated unit
`test`	Dataframe with two columns. The first is the post/pre MSPE ratio for each unit. The second indicates unit names

Author(s)

Bruno Castanho Silva

References

Abadie, A., Diamond, A., Hainmueller, J. (2014). Comparative Politics and the Synthetic Control Method. American Journal of Political Science Forthcoming 2014.

Synthetic : An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie, A., Diamond, A., Hainmueller, J. (2011). Synth: An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie A, Diamond A, Hainmueller J (2010). Synthetic Control Methods for Comparative Case Studies: Estimating the Effect of California's Tobacco Control Program. Journal of the American Statistical Association 105 (490) 493–505.

See Also

See also generate.placebos, mspe.plot, synth

Examples

```
## First prepare the required objects

# Load simulated data from Synth
data(synth.data)

# Execute dataprep to produce the necessary matrices for synth
dataprep.out<-
  dataprep(
   foo = synth.data,
   predictors = c("X1", "X2", "X3"),
   predictors.op = "mean",
   dependent = "Y",
   unit.variable = "unit.num",
   time.variable = "year",
   special.predictors = list(
      list("Y", 1991, "mean"),
      list("Y", 1985, "mean"),
      list("Y", 1980, "mean")
                          ),
   treatment.identifier = 7,
   controls.identifier = c(29, 2, 13, 17, 32, 38),
   time.predictors.prior = c(1984:1989),
   time.optimize.ssr = c(1984:1990),
   unit.names.variable = "name",
   time.plot = 1984:1996
   )

# run the synth command to create the synthetic control
synth.out <- synth(dataprep.out)

## run the generate.placebos command to reassign treatment status
## to each unit listed as control, one at a time, and generate their
## synthetic versions.
tdf<-generate.placebos(dataprep.out,synth.out)

## Test how extreme was the observed treatment effect given the placebos:
ratio<-mspe.test(tdf)
ratio$p.val

## Check visually how extreme is this value in the distribution:
mspe.plot(tdf,discard.extreme=F)
```

`multiple.synth` *Function to Apply Synthetic Controls to Multiple Treated Units*

Description

Generates one synthetic control for each treated unit and calculates the difference between the treated and the synthetic control for each. Returns a vector with outcome values for the synthetic controls, a plot of average treatment effects, and if required generates placebos out of the donor pool to be used in conjunction with plac.dist. All arguments are the same used for dataprep in the Synth package, except for treated.units, treatment.time, and generate.placebos.

Usage

```
multiple.synth(foo, predictors, predictors.op, dependent, unit.variable, time.variable,
special.predictors, treated.units, control.units, time.predictors.prior,
time.optimize.ssr, unit.names.variable, time.plot, treatment.time, generate.placebos = F)
```

Arguments

`foo`	Dataframe with the panel data.
`predictors`	Vector of column numbers or column-name character strings that identifies the predictors' columns. All predictors have to be numeric.
`predictors.op`	A character string identifying the method (operator) to be used on the predictors. Default is "mean".
`dependent`	The column number or a string with the column name that corresponds to the dependent variable.
`unit.variable`	The column number or a string with the column name that identifies unit numbers. The variable must be numeric.
`time.variable`	The column number or a string with the column name that identifies the period (time) data. The variable must be numeric.
`special.predictors`	A list object identifying additional predictors and their pre-treatment years and operators.
`treated.units`	A vector identifying the "unit.variable" numbers of the treated units.
`control.units`	A vector identifying the "unit.variable" numbers of the control units.
`time.predictors.prior`	A numeric vector identifying the pretreatment periods over which the values for the outcome predictors should be averaged.
`time.optimize.ssr`	A numeric vector identifying the periods of the dependent variable over which the loss function should be minimized between each treated unit and its synthetic control.
`unit.names.variable`	The column number or string with column name identifying the variable with units' names. The variable must be a character.
`time.plot`	A vector identifying the periods over which results are to be plotted with path.plot
`treatment.time`	A numeric value with the value in "time.variable" that marks the intervention.
`generate.placebos`	Logical. Whether a placebo (a synthetic control) for each unit in the donor pool should be constructed. Will increase computation time.

Details

The function runs dataprep and synth for each unit identified in "treated.units". It saves the vector with predicted values for each synthetic control, to be used in estimating average treatment effects in applications of Synthetic Controls for multiple treated units.

For further details on the arguments, see the documentation of Synth.

Value

df	Data frame. Each column contains the outcome values for every time-point for one unit or its synthetic control. The last column contains the time-points.

Author(s)

Bruno Castanho Silva

Examples

```
When I have the time.
```

---

plac.dist *Plot the distribution of placebo samples for synthetic control analysis with multiple treated units.*

Description

Takes the output object of \linKmultiple.synth creates a distribution of placebo average treatment effects, to test the significance of the observed ATE. Does so by sampling k placebos (where k = the number of treated units) nboots times, and calculating the average treatment effect of the k placebos each time.

Usage

```
plac.dist(multiple.synth, nboots)
```

Arguments

multiple.synth	An object returned by the function multiple.synth.
nboots	Number of bootstrapped samples of placebos to take.

Value

p	The plot.
att.t	The observed average treatment effect.
df	Dataframe where each row is the ATT for one bootstrapped placebo sample, used to build the distribution plot.

Author(s)

Bruno Castanho Silva

Examples

```
When I have the time.
```

`plot.placebos` *Function to plot placebos of a synthetic control analysis*

Description

Creates plots with the difference between observed units and synthetic controls for the treated and control units. See Abadie, Diamond, and Hainmueller (2011).

Usage

```
plot.placebos(tdf = tdf, discard.extreme = FALSE, mspe.limit = 20, xlab = NULL,
ylab = NULL, title = NULL)
```

Arguments

`tdf`	An object with a list of outcome values for placebos, constructed by generate.placebos
`discard.extreme`	Logical. Whether or not units with high pretreatement MSPE should be excluded from the plot.
`mspe.limit`	Numerical. Used if `discard.extreme` is `TRUE`. It indicates how many times the pretreatment MSPE of a placebo should be higher than that of the treated unit to be considered extreme and discarded. Default is `20`.
`xlab`	Character. Optional. Label of the x axis.
`ylab`	Character. Optional. Label of the y axis.
`title`	Character. Optional. Title of the plot.

Author(s)

Bruno Castanho Silva

References

Abadie, A., Diamond, A., Hainmueller, J. (2014). Comparative Politics and the Synthetic Control Method. American Journal of Political Science Forthcoming 2014.

Synthetic : An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie, A., Diamond, A., Hainmueller, J. (2011). Synth: An R Package for Synthetic Control Methods in Comparative Case Studies. Journal of Statistical Software 42 (13) 1–17.

Abadie A, Diamond A, Hainmueller J (2010). Synthetic Control Methods for Comparative Case Studies: Estimating the Effect of California's Tobacco Control Program. Journal of the American Statistical Association 105 (490) 493–505.

Abadie, A. and Gardeazabal, J. (2003) Economic Costs of Conflict: A Case Study of the Basque Country American Economic Review 93 (1) 113–132.

See Also

generate.placebos, gaps.plot, synth, dataprep

Examples

```
## First prepare the required objects

# Load simulated data from Synth
data(synth.data)

# Execute dataprep to produce the necessary matrices for synth
dataprep.out<-
  dataprep(
   foo = synth.data,
   predictors = c("X1", "X2", "X3"),
   predictors.op = "mean",
   dependent = "Y",
   unit.variable = "unit.num",
   time.variable = "year",
   special.predictors = list(
      list("Y", 1991, "mean"),
      list("Y", 1985, "mean"),
      list("Y", 1980, "mean")
                           ),
   treatment.identifier = 7,
   controls.identifier = c(29, 2, 13, 17, 32, 38),
   time.predictors.prior = c(1984:1989),
   time.optimize.ssr = c(1984:1990),
   unit.names.variable = "name",
   time.plot = 1984:1996
   )

# run the synth command to create the synthetic control
synth.out <- synth(dataprep.out)

## run the generate.placebos command to reassign treatment status
## to each unit listed as control, one at a time, and generate their
## synthetic versions.
tdf<-generate.placebos(dataprep.out,synth.out)

## Plot the gaps in outcome values over time of each unit --
## treated and placebos -- to their synthetic controls
```

```
p <- plot.placebos(tdf,discard.extreme=T,mspe.limit=10,xlab='Year')
p
```

---

synth.data *Simulated data from synth*

Usage

```
data("synth.data")
```

Format

A data frame with 168 observations on the following 7 variables.

`unit.num` a numeric vector

`year` a numeric vector

`name` a character vector

`Y` a numeric vector

`X1` a numeric vector

`X2` a numeric vector

`X3` a numeric vector

Examples

```
data(synth.data)
## maybe str(synth.data) ; plot(synth.data) ...
```

Index