Package 'gsynth'

August 25, 2018

Type Package

Title Generalized Synthetic Control Method

Version 1.0.9

Date 2018-08-07

Author Yiqing Xu, Licheng Liu

Maintainer Yiqing Xu <yiqingxu@ucsd.edu>

Description Provides causal inference with interactive fixed-effect models. It imputes counterfactuals for each treated unit using control group information based on a linear interactive fixed effects model that incorporates unit-specific intercepts interacted with time-varying coefficients. This method generalizes the synthetic control method to the case of multiple treated units and variable treatment periods, and improves efficiency and interpretability. This version supports unbalanced panels and implements the matrix completion method. Main reference: Yiqing Xu (2017) <doi:10.1017/pan.2016.2>.

URL http://yiqingxu.org/software/gsynth/gsynth_examples.html

NeedsCompilation yes

License GPL-2

Imports Rcpp (>= 0.12.3), ggplot2 (>= 2.1.0), GGally (>= 1.0.1), doParallel (>= 1.0.10), foreach (>= 1.4.3), abind (>= 1.4-0), mvtnorm (>= 1.0-6), MASS (>= 7.3.47)

SystemRequirements A C++11 compiler.

Depends R (>= 2.10)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.0.1

Repository CRAN

Date/Publication 2018-08-24 22:14:01 UTC

R topics documented:

`gsynth-package` *Generalized Synthetic Control Method*

Description

Implements the generalized synthetic control method based on interactive fixed effect models.

Details

Implements the generalized synthetic control method. It imputes counterfactuals for each treated unit using control group information based on a linear interactive fixed effects model that incorporates unit-specific intercepts interacted with time-varying coefficients.

See `gsynth` for details.

Author(s)

Yiqing Xu <yiqingxu@ucsd.edu>, University of California, San Diego

Licheng Liu <liulch.16@sem.tsinghua.edu.cn>, Tsinghua University

References

Yiqing Xu. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." Political Analysis, Vol. 25, Iss. 1, January 2017, pp. 57-76. Available at: https://doi.org/10.1017/pan.2016.2.

For more details, see http://yiqingxu.org/software/gsynth/gsynth_examples.html.

gsynth *Generalized Synthetic Control Method*

Description

Implements the generalized synthetic control method based on interactive fixed effect models.

Usage

```
gsynth(formula=NULL, data, Y, D, X = NULL, na.rm = FALSE,
       index, weight = NULL, force = "unit", r = 0, lambda = NULL,
       nlambda = 10, CV = TRUE, k = 5, EM = FALSE, estimator = "ife",
       se = FALSE, nboots = 200,
       inference = "nonparametric", cov.ar = 1, parallel = TRUE,
       cores = NULL, tol = 0.001, seed = NULL, min.T0 = 5,
       conf.lvl = 0.95, normalize = FALSE)
```

Arguments

formula	an object of class "formula": a symbolic description of the model to be fitted.
data	a data frame (must be with a dichotomous treatment but balanced is not required).
Y	outcome.
D	treatment.
X	time-varying covariates.
na.rm	a logical flag indicating whether to list-wise delete missing data. The algorithm will report an error if missing data exist.
index	a two-element string vector specifying the unit (group) and time indicators. Must be of length 2.
weight	a string specifying the weighting variable(if any) to estimate the weighted average treatment effect. Default is `weight = NULL`.
force	a string indicating whether unit or time fixed effects will be imposed. Must be one of the following, "none", "unit", "time", or "two-way". The default is "unit".
r	an integer specifying the number of factors. If `CV = TRUE`, the cross validation procedure will select the optimal number of factors from `r` to 5.
lambda	a single or sequence of positive numbers specifying the hyper-parameter sequence for matrix completion method. If `lambda` is a sequence and `CV = 1`, cross-validation will be performed.
nlambda	an integer specifying the length of hyper-parameter sequence for matrix completion method. Default is `nlambda = 10`.
CV	a logical flag indicating whether cross-validation will be performed to select the optimal number of factors or hyper-parameter in matrix completion algorithm. If `r` is not specified, the procedure will search through `r = 0` to 5.

k	a positive integer specifying cross-validation times for matrix completion algorithm. Default is k = 5.
EM	a logical flag indicating whether an Expectation Maximization algorithm will be used (Gobillon and Magnac 2016).
estimator	a string that controls the estimation method, either "ife" (interactive fixed effects) or "mc" (the matrix completion method).
se	a logical flag indicating whether uncertainty estimates will be produced.
nboots	an integer specifying the number of bootstrap runs. Ignored if se = FALSE.
inference	a string specifying which type of inferential method will be used, either "parametric" or "nonparametric". "parametric" is recommended when the number of treated units is small. parametric bootstrap is not valid for matrix completion method. Ignored if estimator = "mc".
cov.ar	an integer specifying order of the auto regression process that the residuals follow. Used for parametric bootstrap procedure when data is in the form of unbalanced panel. The default value is 1.
parallel	a logical flag indicating whether parallel computing will be used in bootstrapping and/or cross-validation. Ignored if se = FALSE.
cores	an integer indicating the number of cores to be used in parallel computing. If not specified, the algorithm will use the maximum number of logical cores of your computer (warning: this could prevent you from multi-tasking on your computer).
tol	a positive number indicating the tolerance level.
seed	an integer that sets the seed in random number generation. Ignored if se = FALSE and r is specified.
min.T0	an integer specifying the minimum value of pre-treatment periods. Treated units with pre-treatment periods less than that will be removed automatically. This item is important for unbalanced panels. If users want to perform cross validation procedure to select the optimal number of factors from (r.min, r.max), they should set min.T0 larger than (r.max+1) if no individual fixed effects or (r.max+2) otherwise. If there are too few pre-treatment periods among all treated units, a smaller value of r.max is recommended.
conf.lvl	a positive number in the range of 0 and 1 specifying confidence levels for uncertainty estimates. The default value is 0.95.
normalize	a logic flag indicating whether to scale outcome and covariates. Useful for accelerating computing speed when magnitude of data is large. The default is normalize=FALSE.

Details

gsynth implements the generalized synthetic control method. It imputes counterfactuals for each treated unit using control group information based on a linear interactive fixed effects model that incorporates unit-specific intercepts interacted with time-varying coefficients. It generalizes the synthetic control method to the case of multiple treated units and variable treatment periods, and improves efficiency and interpretability. It allows the treatment to be correlated with unobserved unit and time heterogeneities under reasonable modeling assumptions. With a built-in cross-validation

procedure, it avoids specification searches and thus is easy to implement. Data must be with a dichotomous treatment.

Value

`Y.dat`	a matrix storing data of the outcome variable.
`Y`	name of the outcome variable.
`D`	name of the treatment variable.
`X`	name of the time-varying control variables.
`index`	name of the unit and time indicators.
`id`	a vector of unit IDs.
`time`	a vector of time periods.
`obs.missing`	a matrix storing status of each unit at each time point. `0` for missing, `1` for control group units, `2` for treat group units at pre-treatment period, `3` for treat group units at post-treatment period, and `4` for removed treated group units. Useful for unbalanced panel data.
`id.tr`	a vector of IDs for the treatment units.
`id.co`	a vector of IDs for the control units.
`D.tr`	a matrix of treatment indicator for the treated unit outcome.
`I.tr`	a matrix of observation indicator for the treated unit outcome.
`Y.tr`	data of the treated unit outcome.
`Y.ct`	predicted counterfactuals for the treated units.
`Y.co`	data of the control unit outcome.
`eff`	difference between actual outcome and predicted Y(0).
`Y.bar`	average values of Y.tr, Y.ct, and Y.co over time.
`att`	average treatment effect on the treated over time (it is averaged based on the timing of the treatment if it is different for each unit).
`att.avg`	average treatment effect on the treated.
`force`	user specified `force` option.
`sameT0`	TRUE if the timing of the treatment is the same.
`T`	the number of time periods.
`N`	the total number of units.
`p`	the number of time-varying observables.
`Ntr`	the number of treated units.
`Nco`	the number of control units.
`T0`	a vector that stores the timing of the treatment for balanced panel data.
`tr`	a vector indicating treatment status for each unit.
`pre`	a matrix indicating the pre-treatment/non-treatment status.
`post`	a matrix indicating the post-treatment status.

r.cv	the number of factors included in the model – either supplied by users or automatically chosen via cross-validation.
lambda.cv	the optimal hyper-parameter in matrix completion method chosen via cross-validation.
res.co	residuals of the control group units.
beta	coefficients of time-varying observables from the interactive fixed effect model.
sigma2	the mean squared error of interactive fixed effect model.
IC	the information criterion.
est.co	result of the interactive fixed effect model based on the control group data. An interFE object.
eff.cnt	difference between actual outcome and predicted Y(0); rearranged based on the timing of the treatment.
Y.tr.cnt	data of the treated unit outcome, rearranged based on the timing of the treatment.
Y.ct.cnt	data of the predicted Y(0), rearranged based on the timing of the treatment.
MSPE	mean squared prediction error of the cross-validated model.
CV.out	result of the cross-validation procedure.
niter	the number of iterations in the estimation of the interactive fixed effect model.
factor	estimated time-varying factors.
lambda.co	estimated loadings for the control group.
lambda.tr	estimated loadings for the treatment group.
wgt.implied	estimated weights of each of the control group unit for each of the treatment group unit.
mu	estimated ground mean.
xi	estimated time fixed effects.
alpha.tr	estimated unit fixed effects for the treated units.
alpha.co	estimated unit fixed effects for the control units.
validX	a logic value indicating if multicollinearity exists.
inference	a string indicating bootstrap procedure.
est.att	inference for att.
est.att.avg	inference for att.avg.
est.beta	inference for beta.
est.ind	inference for att of each treated unit.
att.boot	bootstrap results for att.
beta.boot	bootstrap results for beta.

Author(s)

Yiqing Xu <yiqingxu@ucsd.edu>

Licheng Liu <liulch.16@sem.tsinghua.edu.cn>

References

Laurent Gobillon and Thierry Magnac, 2016. "Regional Policy Evaluation: Interactive Fixed Effects and Synthetic Controls." The Review of Economics and Statistics, July 2016, Vol. 98, No. 3, pp. 535–551.

Yiqing Xu. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." Political Analysis, Vol. 25, Iss. 1, January 2017, pp. 57-76. Available at: https://doi.org/10.1017/pan.2016.2.

Athey S, Bayati M, Doudchenko N, et al. Matrix completion methods for causal panel data models[J]. arXiv preprint arXiv:1710.10251, 2017.

For more details, see http://yiqingxu.org/software/gsynth/gsynth_examples.html.

For more details about the matrix completion method, see https://github.com/susanathey/MCPanel.

See Also

plot.gsynth and print.gsynth

Examples

```
library(gsynth)
data(gsynth)
out <- gsynth(Y ~ D + X1 + X2, data = simdata, parallel = FALSE,
              index = c("id","time"), force = "two-way",
              CV = TRUE, r = c(0, 5), se = FALSE)
print(out)
```

---

gsynth-internal *Internal Gsynth Functions*

Description

Internal Gsynth functions

Details

These are not to be called by the user (or in some cases are just waiting for proper documentation to be written :).

`interFE` *Interactive Fixed Effects Models*

Description

Estimating interactive fixed effect models.

Usage

```
interFE(formula = NULL, data, Y, X, index, r = 0, force = "none",
         se = TRUE, nboots = 500, seed = NULL, normalize = FALSE)
```

Arguments

`formula`	an object of class "formula": a symbolic description of the model to be fitted.
`data`	a data frame (must be with a dichotomous treatment but balanced is not required).
`Y`	outcome.
`X`	time-varying covariates.
`index`	a two-element string vector specifying the unit (group) and time indicators. Must be of length 2.
`r`	an integer specifying the number of factors.
`force`	a string indicating whether unit or time fixed effects will be imposed. Must be one of the following, "none", "unit", "time", or "two-way". The default is "unit".
`se`	a logical flag indicating whether uncertainty estimates will be produced via bootstrapping.
`nboots`	an integer specifying the number of bootstrap runs. Ignored if `se = FALSE`.
`seed`	an integer that sets the seed in random number generation. Ignored if `se = FALSE` and `r` is specified.
`normalize`	a logic flag indicating whether to scale outcome and covariates. Useful for accelerating computing speed when magnitude of data is large.The default is `normalize=FALSE`.

Details

`interFE` estimates interactive fixed effect models proposed by Bai (2009).

Value

`beta`	estimated coefficients.
`mu`	estimated grand mean.
`factor`	estimated factors.
`lambda`	estimated factor loadings.

VNT	a diagonal matrix that consists of the r eigenvalues.
niter	the number of iteration before convergence.
alpha	estimated unit fixed effect (if force is "unit" or "two-way").
xi	estimated time fixed effect (if force is "time" or "two-way").
residuals	residuals of the estimated interactive fixed effect model.
sigma2	mean squared error of the residuals.
IC	the information criterion.
ValidX	a logical flag specifying whether there are valid covariates.
dat.Y	a matrix storing data of the outcome variable.
dat.X	an array storing data of the independent variables.
Y	name of the outcome variable.
X	name of the time-varying control variables.
index	name of the unit and time indicators.
est.table	a table of the estimation results.
est.boot	a matrix storing results from bootstraps.

Author(s)

Yiqing Xu <yiqingxu@ucsd.edu>

Licheng Liu <liulch.16@sem.tsinghua.edu.cn>

References

Jushan Bai. 2009. "Panel Data Models with Interactive Fixed Effects." Econometrica 77:1229–1279.

See Also

print.interFE and gsynth

Examples

```
library(gsynth)
data(gsynth)
d <- simdata[-(1:150),] # remove the treated units
out <- interFE(Y ~ X1 + X2, data = d, index=c("id","time"),
               r = 2, force = "two-way", nboots = 50)
```

plot.gsynth *Plotting*

Description

Visualizes estimation results of the generalized synthetic control method.

Usage

```
## S3 method for class 'gsynth'
plot(x, type = "gap", xlim = NULL, ylim = NULL,
            xlab = NULL, ylab = NULL, legendOff = FALSE, raw = "none",
            main = NULL, nfactors = NULL, id = NULL, axis.adjust = FALSE,
            theme.bw = FALSE, shade.post = NULL, ...)
```

Arguments

x	a gsynth object.
type	a string that specifies the type of the plot. Must be one of the following: "gap" (plotting the average treatment effect on the treated; "raw" (plotting the raw data); "counterfactual", or "ct" for short, (plotting predicted Y(0)'s); "factors" (plotting estimated factors); "loadings" (plotting the distribution of estimated factor loadings); "missing" (plotting status of each unit at each time point).
xlim	a two-element numeric vector specifying the range of x-axis. When class of time variable is string, must specify not original value but a counting number e.g. xlim=c(1,30).
ylim	a two-element numeric vector specifying the range of y-axis.
xlab	a string indicating the label of the x-axis.
ylab	a string indicating the label of the y-axis.
legendOff	a logical flag controlling whether to show the legend.
raw	a string indicating whether or how raw data for the outcome variable will be shown in the "counterfactual" plot. Ignored if type is not "counterfactual". Must be one of the following: "none" (not showing the raw data); "band" (showing the middle 90 percentiles of the raw data); and "all" (showing the raw data as they are).
main	a string that controls the title of the plot. If not supplied, no title will be shown.
nfactors	a positive integer that specifies the number of factors to be shown. The maximum number if 4. Ignored if type is not "factors"
id	a unit identifier of which the predicted counterfactual or the difference between actual and predicted counterfactual is to be shown. It can also be a vector specifying units to be plotted if type=="missing" when data magnitude is large. Ignored if type is none of "missing", "counterfactual", "gap".
axis.adjust	a logical flag indicating whether to adjust labels on x-axis. Useful when class of time variable is string and data magnitude is large.

theme.bw	a logical flag indicating whether to use a black/white theme.
shade.post	a logical flag controlling whether to shade the post-treatment periods.
...	other argv.

Details

plot.gsynth visualizes the raw data used by, or estimation results obtained from, the generalized synthetic control method.

Author(s)

Yiqing Xu <yiqingxu@ucsd.edu>

Licheng Liu <liulch.16@sem.tsinghua.edu.cn>

References

Yiqing Xu. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." Political Analysis, Vol. 25, Iss. 1, January 2017, pp. 57-76. Available at: https://doi.org/10.1017/pan.2016.2.

See http://yiqingxu.org/software/gsynth/gsynth_examples.html for more detailed information.

See Also

gsynth and print.gsynth

---

print.gsynth *Print Results*

Description

Print results of the generalized synthetic control method.

Usage

```
## S3 method for class 'gsynth'
print(x, ...)
```

Arguments

x	a gsynth object.
...	other argv.

Author(s)

Yiqing Xu <yiqingxu@ucsd.edu>

Licheng Liu <liulch.16@sem.tsinghua.edu.cn>

References

Yiqing Xu. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." Political Analysis, Vol. 25, Iss. 1, January 2017, pp. 57-76. Available at: https://doi.org/10.1017/pan.2016.2.

For more details, see http://yiqingxu.org/software/gsynth/gsynth_examples.html.

See Also

gsynth and plot.gsynth

`print.interFE` *Print Results*

Description

Print results of interactive fixed effects estimation.

Usage

```
## S3 method for class 'interFE'
print(x, ...)
```

Arguments

x	an interFE object.
...	other argv.

Author(s)

Yiqing Xu <yiqingxu@ucsd.edu>

Licheng Liu <liulch.16@sem.tsinghua.edu.cn>

References

Jushan Bai. 2009. "Panel Data Models with Interactive Fixed Effects." Econometrica 77:1229–1279.

See Also

interFE and gsynth

simdata *simdata*

Description

A simulated dataset.

Format

dataframe

References

Yiqing Xu. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." Political Analysis, Vol. 25, Iss. 1, January 2017, pp. 57-76. Available at: https://doi.org/10.1017/pan.2016.2.

For more details, see http://yiqingxu.org/software/gsynth/gsynth_examples.html.

turnout *turnout*

Description

State-level voter turnout data.

Format

dataframe

References

Melanie Jean Springer. 2014. How the States Shaped the Nation: American Electoral Institutions and Voter Turnout, 1920-2000. University of Chicago Press.

Yiqing Xu. 2017. "Generalized Synthetic Control Method: Causal Inference with Interactive Fixed Effects Models." Political Analysis, Vol. 25, Iss. 1, January 2017, pp. 57-76. Available at: https://doi.org/10.1017/pan.2016.2.

For more details, see http://yiqingxu.org/software/gsynth/gsynth_examples.html.

Index