Package ‘pgsc’

October 28, 2018

Type  Package
Title  Computes Powell’s Generalized Synthetic Control Estimator
Version  1.0.0
Date  2018-10-21
Author  Philip Barrett
Maintainer  Philip Barrett <pobarrett@gmail.com>
Description  Computes the generalized synthetic control estimator described in
             Powell (2017) <doi:10.7249/WR1142>. Provides both point estimates, and hypothesis testing.
License  GPL-2
Depends  R (>= 2.10), Rcpp (>= 0.12.18), nloptr, reshape2
LinkingTo  Rcpp, RcppArmadillo
RoxygenNote  6.1.0
Suggests  knitr, rmarkdown, plm
VignetteBuilder  knitr
URL  https://github.com/philipbarrett/pgsc
Encoding  UTF-8
NeedsCompilation  yes
Repository  CRAN
Date/Publication  2018-10-28 22:50:03 UTC

R topics documented:

pgsc ................................................................. 2
pgsc.dta ............................................................ 3
pgsc.wald.test ................................................... 4

Index  5
pgsc

Wrapper function for GSC estimation

Description

Wrapper function for GSC estimation

Usage

pgsc(dtaL depNvarL indepNvarL bNinitL methodL sol.it = NULL, wt.init = NULL, print.level = 0, g.i = NULL, g.i.grad = NULL, ...)

Arguments

dta      A data frame
dep.var   A string defining the dependent variable
indep.var A vector of strings defining the independent (treatment) variables
b.init   An initial value for the treatment variable coefficients. Must have same length
          as 'indep.var'
method   The GSC iteration method to be used. Must be one of:
          • onestep: "Plain" GSC solution, without weights
          • twostep.aggte: Observations weighted by unit MSEs from the one-step
            solution.
          • twostep.indiv: Observations weighted by unit MSEs from individual,
            unit-by-unit unweighted solutions.
sol.it   The first step solution used in the two-step methods. If omitted, a new one-step
          solution is computed.
wt.init  An initial value for the weighting matrix
print.level The level of detail provided in the printed output
g.i     A function defining a restriction on the parameters. Used in hypothesis testing.
g.i.grad The gradient of g.i.
...      Other arguments to be passed to the optimization

Details

See the vignette "Using pgsc" for an extended example.
Value

Returns the point estimate of the model as a gsc object, a list with entries:

- **b** The point estimate of the coefficients on the dependent variables
- **diff** The difference between successive iterations
- **err** The maximum error on the within-iteration optimization problems
- **it** Number of iterations required to solve
- **sig.i** The unit-specific MSEs from the solution
- **W** The "full" weighting matrix for counterfactuals, containing own-unit weights (all zero) and unit-N weights
- **wt** The "minimal" weighting matrix, omitting own-unit weights and weights on unit N (which can be computed as one-minus-rowsum)

Examples

data("pgsc.dta")
sol <- pgsc(pgsc.dta, dep.var = 'y', indep.var = c('D1', 'D2'), b.init = c(0, 0), method='onestep')
summary(sol)
g.i <- function(b) b[1]; g.i.grad <- function(b) c(1, 0)
sol.r <- pgsc(pgsc.dta, dep.var = 'y', indep.var = c('D1', 'D2'), b.init = sol$w, method='onestep', g.i=g.i, g.i.grad=g.i.grad)
summary(sol.r)

Description

A dataset with an outcome given by a treatment and a set of factors.

Usage

pgsc.dta

Format

A data frame with 750 rows and 8 variables:

- **n** The unit, here labeled as a US state
- **t** The time period
- **y** The outcome variable
- **D1** The first treatment variable
- **D2** The second treatment variable
- **X1** The first observed confounding factor
- **X2** The second observed confounding factor
- **X3** The third observed confounding factor
pgsc.wald.test

A wrapper for the wald test of a restricted solution

Description

A wrapper for the wald test of a restricted solution

Usage

pgsc.wald.test(dta, dep.var, indep.var, sol.rest, n.boot = 10000, seed = 42)

Arguments

dta A dataframe
dep.var A vector of strings of names of dependent variables.
indep.var A vector of strings of names of independent (treatment) variables.
sol.rest A restricted solution which is being tested
n.boot The number of bootstrapped samples for the variance calculation. Default is 10000.
seed Randomization seed. Default is 42.

Details

See the vignette "Using pgsc" for an extended example.

Value

Returns the wald test as gsc.wald object, a list with entries:

b The point estimate of the coefficients on the dependent variables
S The Wald statistic
s.boot The bootstrapped Wald statistic
p.value The p-value for the Wald statistic.

Examples

data("pgsc.dta")
g.i <- function(b) b[1] ; g.i.grad <- function(b) c(1,0)
sol.r <- pgsc(pgsc.dta, dep.var = 'y', indep.var = c('D1','D2'), b.init = c(0,1), method='onestep', g.i=g.i, g.i.grad=g.i.grad )
wald <- pgsc.wald.test(pgsc.dta, 'y', indep.var = c('D1','D2'), sol.r )
summary(wald)
plot(wald)
Index

∗Topic datasets
   pgsc.dta, 3

pgsc, 2
pgsc.dta, 3
pgsc.wald.test, 4