ꙅꓕaꓕa
Statistics/Data analysis

Title

   scest ── Estimation for Synthetic Control Methods.

Syntax

   scest , dfname(_string_**)** [**p(**#**) direc(**_string_**) Q(**#**) lb(**#**) name(**_string_**) opt(**_string_**)**]

Description

   scest implements estimation procedures for Synthetic Control (SC) methods using
      least squares, lasso, ridge, or simplex-type constraints according to
      Cattaneo, Feng, and Titiunik (2021). The command is a wrapper of the companion
      Python package. As such, the user needs to have a running version of Python
      with the package installed. A tutorial on how to install Python and link it to
      Stata can be found here.

   Companion R and Python packages are described in Cattaneo, Feng, Palomba and
      Titiunik (2022).

   Companion commands are: scdata for data preparation, scpi for inference
      procedures, and scplot for SC plots.

   Related Stata, R, and Python packages useful for inference in SC designs are
      described in the following website:

   https://nppackages.github.io/scpi/

Options

   dfname(_string_**)** specifies the name of the Python object containing the processed
      data created with scdata.

   ┌─── Constraint ───

 These options let the user specify the type of constraint to be imposed to estimate
    the SC weights. The user controls the norm of the weights to be constrained
    (option **p**), the direction of the constraint on the norm (option **dir**), and the size
    of the constraint on the norm (option **q**). Alternatively, some popular constraints
    can be selected through the option **name**. A detailed description of the popular
    constraints implemented can be found in Cattaneo, Feng, Palomba and Titiunik
    (2022).

   name(_string_**)** specifies the name of the constraint to be used. Options are:
      simplex classic SC estimator as proposed in Abadie (2021). Estimated weights
         are constrained to be non-negative and their L1 norm must be equal to 1.
      lasso weights are estimated using a Lasso-type penalization
      ridge weights are estimated using a Ridge-type penalization.
      ols weights are estimated without constraints using least squares

   p(#**)** sets the type of norm to be constrained. Options are:
      0 no constraint on the norm of the weights is imposed.
      1 a constraint is imposed on the L1 norm of the weights (the default).
      2 a constraint is imposed on the L2 norm of the weights.

   direc(_string_**)** specifies the direction of the constraint on the norm of the
      weights. Options are:
      <= the constraint on the norm of the weights is an inequality constraint.
      == the constraint on the norm of the weights is an equality constraint (the
         default).

   Q(#**)** specifies the size of the constraint on the norm of the weights.

   lb(#**)** specifies the lower bound on the weights. The default is **lb(0)**.

   ┌─── Others ───

opt(*string***)** a string specifying the stopping criteria used by the underling
optimizer (<u>nlopt</u>) for point estimation. The default is a sequential quadratic
programming (SQP) algorithm for nonlinearly constrained gradient-based
optimization ('SLSQP'). The default value is **opt("'maxeval' = 5000,
'xtol_rel' = 1e-8, 'xtol_abs' = 1e-8, 'ftol_rel' = 1e-12, 'ftol_abs' = 1e-12,
'tol_eq' = 1e-8, 'tol_ineq' = 1e-8")**. In case a lasso-type constraint is
implemented, a different optimizer (<u>cvxpy</u>) is used and stopping criteria
cannot be changed.

Example: Cattaneo, Feng and Titiunik (2021) Germany Data

Setup
    . **use scpi_germany.dta**

Prepare data
    . **scdata gdp, dfname("python_scdata") id(country) outcome(gdp) time(year)
    treatment(status) cointegrated**

Estimate Synthetic Control with a simplex constraint
    . **scest, dfname("python_scdata") name(simplex)**

Stored results

scest stores the following in **e()**:

Scalars
  e(M)   number of features
  e(KM)   number of covariates used for adjustment
  e(J)   number of donors
  e(T1)   number of post-treatment periods
  e(q)   size of the constraint on the norm

Macros
  e(features)   name of features
  e(outcomevar)   name of outcome variable
  e(constant)   logical indicating the presence of a common constant
                                across features
  e(cointegrated_data)   logical indicating cointegration
  e(p)   type of norm of the weights used in constrained
                                estimation
  e(dir)   direction of the constraint on the norm of the
                                weights
  e(name)   name of constraint used in estimation

Matrices
  e(T0)   number of pre-treatment periods per feature
  e(A)   pre-treatment features of the treated unit
  e(B)   pre-treatment features of the control units
  e(C)   covariates used for adjustment
  e(pred)   predicted values of the features of the treated unit
  e(res)   residuals **e(A)** - **e(pred)**
  e(w)   weights of the controls
  e(r)   coefficients of the covariates used for adjustment
  e(beta)   stacked version of **e(w)** and **e(r)**
  e(Y_post)   post-treatment outcome of the treated unit
  e(Y_post_fit)   estimated post-treatment outcome of the treated unit
  e(Y_pre)   pre-treatment outcome of the treated unit
  e(Y_pre_fit)   estimate pre-treatment outcome of the treated unit

References

Abadie, A. 2021. <u>Using synthetic controls: Feasibility, data requirements, and
    methodological aspects.</u> *Journal of Economic Literature*, 59(2), 391-425.

Cattaneo, M. D., Feng, Y., and Titiunik, R. 2021. <u>Prediction Intervals for Synthetic Sontrol Methods.</u> *Journal of the American Statistical Association*, 116(536), 1865–1880.

Cattaneo, M. D., Feng, Y., Palomba F., and Titiunik, R. 2022. <u>scpi − Uncertainty Quantification for Synthetic Control Estimators</u>.

Authors

Matias D. Cattaneo, Princeton University, Princeton, NJ. <u>cattaneo@princeton.edu</u>.

Yingjie Feng, Tsinghua University, Beijing, China. <u>fengyj@sem.tsinghua.edu.cn</u>.

Filippo Palomba, Princeton University, Princeton, NJ. <u>fpalomba@princeton.edu</u>.

Rocio Titiunik, Princeton University, Princeton, NJ. <u>titiunik@princeton.edu</u>.