Package ‘DiSCos’

March 14, 2024

Title Distributional Synthetic Controls Estimation

Version 0.0.1

Description The method of synthetic controls is a widely-adopted tool for evaluating causal effects of policy changes in settings with observational data. In many settings where it is applicable, researchers want to identify causal effects of policy changes on a treated unit at an aggregate level while having access to data at a finer granularity. This package implements a simple extension of the synthetic controls estimator, developed in Gunsilius (2023) <doi:10.3982/ECTA18260>, that takes advantage of this additional structure and provides nonparametric estimates of the heterogeneity within the aggregate unit. The idea is to replicate the quantile function associated with the treated unit by a weighted average of quantile functions of the control units. The package contains tools for aggregating and plotting the resulting distributional estimates, as well as for carrying out inference on them.

License MIT + file LICENSE

BugReports `https://github.com/`

URL <link href=``https://github.com/'' rel=``canonical">, <link
href=``http://www.davidvandijcke.com/DiSCos/'' rel=``canonical">

LazyData TRUE

Imports CVXR, pracma, Rdpack, parallel, evmix, utils, extremeStat,
MASS

Depends data.table, R (>= 2.10), ggplot2

RdMacros Rdpack

Suggests haven, latex2exp, knitr, rmarkdown, maps, testthat (>=
3.0.0), quadprog

Encoding UTF-8

RoxygenNote 7.2.2

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author David Van Dijcke [aut, cre] (<https://orcid.org/0000-0001-5825-3447>),
Florian Gunsilius [aut] (<https://orcid.org/0000-0002-1698-6324>),
Siyun He [aut] (<https://orcid.org/0000-0001-5825-3447>)

Maintainer David Van Dijcke <dvdijcke@umich.edu>

Repository CRAN

Date/Publication 2024-03-14 20:00:02 UTC

R topics documented:

`DiSCo` *Distributional Synthetic Controls*

Description

This function implements the distributional synthetic controls (DiSCo) method from Gunsilius (2023). as well as the alternative mixture of distributions approach.

Usage

```
DiSCo(
  df,
  id_col.target,
  t0,
  M = 1000,
  G = 1000,
  num.cores = 1,
  permutation = FALSE,
  q_min = 0,
  q_max = 1,
  CI = FALSE,
  CI_placebo = TRUE,
  boots = 500,
  cl = 0.95,
  graph = FALSE,
  qmethod = NULL,
  seed = NULL,
  simplex = FALSE
)
```

Arguments

`df` Data frame or data table containing the distributional data for the target and control units. The data table should contain the following columns:

- `y_col` A numeric vector containing the outcome variable for each unit. Units can be individuals, states, etc., but they should be nested within a larger unit (e.g. individuals or counties within a state)
- `id_col` A numeric vector containing the aggregate IDs of the units. This could be, for example, the state if the units are counties or individuals
- `time_col` A vector containing the time period of the observation for each unit. This should be a monotonically increasing integer.

`id_col.target` Variable indicating the name of the target unit, as specified in the id_col column of the data table. This variable can be any type, as long as it is the same type as the id_col column of the data table.

`t0` Integer indicating period of treatment.

`M` Integer indicating the number of control quantiles to use in the DiSCo method. Default is 1000.

`G` Integer indicating the number of grid points for the grid on which the estimated functions are evaluated. Default is 1000.

`num.cores` Integer, number of cores to use for parallel computation. Default is 1. If the `permutation` or `CI` arguments are set to TRUE, this can be slow and it is recommended to set this to 4 or more, if possible. If you get an error in "all cores" or similar, try setting num.cores=1 to see the precise error value.

`permutation` Logical, indicating whether to use the permutation method for computing the optimal weights. Default is FALSE.

`q_min` Numeric, minimum quantile to use. Set this together with `q_max` to restrict the range of quantiles used to construct the synthetic control. Default is 0 (all quantiles).

`q_max` Numeric, maximum quantile to use. Set this together with `q_min` to restrict the range of quantiles used to construct the synthetic control. Default is 1 (all quantiles).

`CI` Logical, indicating whether to compute confidence intervals for the counterfactual quantiles. Default is FALSE.

`CI_placebo` Logical, indicating whether to compute confidence intervals for the pre-treatment periods. Default is TRUE. If you have a lot of pre-treatment periods, setting this to FALSE can speed up the computation.

`boots` Integer, number of bootstrap samples to use for computing confidence intervals. Default is 500.

`cl` Numeric, confidence level for the (two-sided) confidence intervals.

`graph` Logical, indicating whether to plot the permutation graph as in Figure 3 of the paper. Default is FALSE.

`qmethod` Character, indicating the method to use for computing the quantiles of the target distribution. The default is NULL, which uses the `quantile` function from the stats package. Other options are "qkden" (based on smoothed kernel density function) and "extreme" (based on parametric extreme value distributions). Both are substantially slower than the default method but may be useful for fat-tailed distributions with few data points at the upper quantiles. Alternatively, one could use the q_max option to restrict the range of quantiles used.

`seed` Integer, seed for the random number generator. This needs to be set explicitly in the function call, since it will invoke RNGkind which will set the seed for each core when using parallel processes. Default is NULL, which does not set a seed.

`simplex` Logical, indicating whether to use to constrain the optimal weights to the unit simplex. Default is FALSE, which only constrains the weights to sum up to 1 but allows them to be negative.

Details

This function is called for every time period in the DiSCo function. It implements the DiSCo method for a single time period, as well as the mixture of distributions approach. The corresponding results for each time period can be accessed in the `results.periods` list of the output of the DiSCo function. The DiSCo function returns the average weight for each unit across all periods, calculated as a uniform mean, as well as the counterfactual target distribution produced as the weighted average of the control distributions for each period, using these averaged weights.

Value

A list containing, for each time period, the elements described in the return argument of `DiSCo_iter`, as well as the following additional elements:

- `DiSco`
  - `CI` A list with the confidence intervals and standard errors for the counterfactual quantiles, if `CI` is TRUE and for the periods specified in `CI_periods`. See the output of `DiSCo_CI` for details.
  - `quantile` The counterfactual quantiles for the target unit.
  - `weights` The optimal weights for the target unit.
  - `cdf` The counterfactual CDF for the target unit.
- `perm` A `permut` object containing the results of the permutation method, if `permutation` is TRUE. Call `summary` on this object to print the overall results of the permutation test.

References

Gunsilius FF (2023). “Distributional synthetic controls.” *Econometrica*, **91**(3), 1105–1117.()

DiSCoT *Store aggregated treatment effects*

Description

S3 object holding aggregated treatment effects

Usage

```
DiSCoT(
  agg,
  treats,
  ses,
  grid,
  ci_lower,
  ci_upper,
  t0,
  call,
  cl,
  N,
  J,
  agg_df,
  perm,
  plot
)
```

Arguments

agg	aggregation method
treats	list of treatment effects
ses	list of standard errors
grid	grid
ci_lower	list of lower confidence intervals
ci_upper	list of upper confidence intervals
t0	start time
call	call
cl	confidence level
N	number of observations
J	number of treated units
agg_df	dataframe of aggregated treatment effects and their confidence intervals
perm	list of per mutation results
plot	a ggplot object containing the plot for the aggregated treatment effects using the agg parameter

Value

S3 object of class `DiSCoT` with associated `summary` and `print` methods

`DiSCoTEA` *Aggregate treatment effects from DiSCo function.*

Description

Function to aggregate treatment effects from the output of the DiSCo function, plot the distribution of the aggregation statistic over time, and report summary tables.

Usage

```
DiSCoTEA(
  disco,
  agg = "quantileDiff",
  graph = TRUE,
  t_plot = NULL,
  savePlots = FALSE,
  xlim = NULL,
  ylim = NULL,
  samples = c(0.25, 0.5, 0.75)
)
```

Arguments

`disco` Output of the DiSCo function.

`agg` String indicating the aggregation statistic to be used. Options include

- `quantileDiff` Difference in quantiles between the target and the weighted average of the controls.
- `quantile` Plots both the observed and the counterfactual quantile functions. No summary statistics will be produced.
- `cdfDiff` Difference in CDFs between the target and the weighted average of the controls.
- `cdf` Plots both the observed and the counterfactual CDFs. No summary statistics will be produced.

`graph` Boolean indicating whether to plot graphs (default is TRUE).

`t_plot` Optional vector of time periods (`t_col` values in the original dataframe) to be plotted (default is NULL, which plots all time periods).

`savePlots` Boolean indicating whether to save the plots to the current working directory (default is FALSE). The plot names will be `[agg]_[start_year]_[end_year].pdf`.

`xlim` Optional vector of length 2 indicating the x-axis limits of the plot. Useful for zooming in on relevant parts of the distribution for fat-tailed distributions.

`ylim` Optional vector of length 2 indicating the y-axis limits of the plot.

samples	Numeric vector indicating the range of quantiles of the aggregation statistic (agg) to be summarized in the summary property of the S3 class returned by the function (default is c(0.25, 0.5, 0.75)). For example, if samples = c(0.25, 0.5, 0.75), the summary table will include the average effect for the 0-25th, 25-50th, 50-75th and 75-100th quantiles of the distribution of the aggregation statistic over time.

Details

This function takes in the output of the DiSCo_per function and computes aggregate treatment effect using a user-specified aggregation statistic. The default is the differences between the counterfactual and the observed quantile functions (quantileDiff). If graph is set to TRUE, the function will plot the distribution of the aggregation statistic over time. The S3 class returned by the function has a summary property that will print a selection of aggregated effects (specified by the samples parameter) for the chosen agg method, by post-treatment year (see examples below). This summary call will only print effects if the agg parameter requested a distribution difference (quantileDiff or cdfDiff). The other aggregations are meant to be inspected visually. If the permutation parameter was set to TRUE in the original DiSCo call, the summary table will include the results of the permutation test. If the original DiSCo call was restricted to a range of quantiles smaller than [0,1] (i.e. q_min > 0 or q_max < 1), the samples parameter is ignored and only the aggregated differences for the quantile range specified in the original call are returned.

Value

A DiSCoT object, which is an S3 class that stores a list of treatment effects, their standard errors, the corresponding confidence intervals (if specified), and a dataframe with treatment effects aggregated according to the agg input. The S3 class also has a summary property that will print a selection of aggregated effects (specified by the samples parameter) for the chosen agg method, by post-treatment year, as well as the permutation test results, if specified.

DiSCo_iter *Estimate DiSCo in a single period*

Description

This function implements the DiSCo method for a single time period, as well as the mixture of distributions approach. Its return values contain valuable period-specific estimation outputs.

Usage

```
DiSCo_iter(
  yy,
  df,
  evgrid,
  id_col.target,
  M,
  G,
  T0,
  qmethod = NULL,
  q_min = 0,
  q_max = 1,
  simplex = FALSE,
  controls.id
)
```

Arguments

`yy` Integer indicating the current year being processed.

`df` Data frame or data table containing the distributional data for the target and control units. The data table should contain the following columns:

- `y_col` A numeric vector containing the outcome variable for each unit. Units can be individuals, states, etc., but they should be nested within a larger unit (e.g. individuals or counties within a state)
- `id_col` A numeric vector containing the aggregate IDs of the units. This could be, for example, the state if the units are counties or individuals
- `time_col` A vector containing the time period of the observation for each unit. This should be a monotonically increasing integer.

`evgrid` A vector of grid points on which to evaluate the quantile functions.

`id_col.target` Variable indicating the name of the target unit, as specified in the id_col column of the data table. This variable can be any type, as long as it is the same type as the id_col column of the data table.

`M` Integer indicating the number of control quantiles to use in the DiSCo method. Default is 1000.

`G` Integer indicating the number of grid points for the grid on which the estimated functions are evaluated. Default is 1000.

`T0` Integer indicating the last pre-treatment period starting from 1.

`qmethod` Character, indicating the method to use for computing the quantiles of the target distribution. The default is NULL, which uses the `quantile` function from the stats package. Other options are "`qkden`" (based on smoothed kernel density function) and "`extreme`" (based on parametric extreme value distributions). Both are substantially slower than the default method but may be useful for fat-tailed distributions with few data points at the upper quantiles. Alternatively, one could use the q_max option to restrict the range of quantiles used.

`q_min` Numeric, minimum quantile to use. Set this together with `q_max` to restrict the range of quantiles used to construct the synthetic control. Default is 0 (all quantiles).

`q_max` Numeric, maximum quantile to use. Set this together with `q_min` to restrict the range of quantiles used to construct the synthetic control. Default is 1 (all quantiles).

`simplex` Logical, indicating whether to use to constrain the optimal weights to the unit simplex. Default is FALSE, which only constrains the weights to sum up to 1 but allows them to be negative.

`controls.id` List of strings specifying the column names for the control units' identifiers.

Details

This function is part of the DiSCo method, called for each time period. It calculates the optimal weights for the DiSCo method and the mixture of distributions approach for a single time period. The function processes data f or both the target and control units, computes the quantile functions, and evaluates these on a specified grid. The function is designed to be used within the broader context of the DiSCo function, which aggregates results across multiple time periods.

Value

A list with the following elements:

- `DiSCo_weights` Weights calculated using the DiSCo method.
- `mixture`
  - `weights` Optimal weights for the mixture approach.
  - `distance` Value of the objective function for the mixture approach.
  - `mean` Weighted mixture of the controls' CDFs.
- `target`
  - `cdf` Empirical CDF of the target.
  - `grid` Grid on which the quantile and CDF functions were evaluated.
  - `data` Original data for the target unit.
  - `quantiles` Quantiles for the target unit, evaluated on the specified grid.
- `controls`
  - `data` Original data for the control units.
  - `cdf` Empirical CDFs of the control units.
  - `quantiles` Quantiles for the control units, evaluated on the specified grid. .
- `controls.q` Quantiles for the control units, evaluated on the specified grid.

`dube` *Data from (Dube 2019)*

Description

As used in the empirical application of Gunsilius (2023).

Usage

```
dube
```

Format

`dube`**:**

A data frame with 652,870 rows and 3 columns:

id_col State FIPS

time_col Year

y_col `adj0contpov` variable in Dube (2019). Captures the distribution of equalized family income from wages and salary, defined as multiples of the federal poverty threshold. ...

ex_gmm *ex_gmm*

Description

Example data for `DiSCo` command. Returns simulated target and control that are mixtures of Gaussian distributions.

Usage

```
ex_gmm(Ts = 2, num.con = 30, numdraws = 1000)
```

Arguments

`Ts`	an integer indicating the number of time periods
`num.con`	an integer indicating the number of control units
`numdraws`	an integer indicating the number of draws

Value

`target`	a vector.
`control`	a matrix.

permut *permut*

Description

Object to hold results of permutation test

Usage

```
permut(distp, distt, p_overall, J_1, q_min, q_max, plot)
```

Arguments

`distp`	List of squared Wasserstein distances between the control units
`distt`	List of squared Wasserstein distances between the target unit and the control units
`p_overall`	Overall p-value
`J_1`	Number of control units
`q_min`	Minimum quantile
`q_max`	Maximum quantile
`plot`	ggplot object containing plot of squared Wasserstein distances over time for all permutations.

Value

A list of class permut, with the same elements as the input arguments.

`summary.DiSCoT` *summary.DiSCoT*

Description

Summary of DiSCoT object

Usage

```
## S3 method for class 'DiSCoT'
summary(object, ...)
```

Arguments

`object`	DiSCoT object
`...`	Additional arguments

Value

summary of DiSCoT object

`summary.permut` *summary.permut*

Description

Summarize permutation test results

Usage

```
## S3 method for class 'permut'
summary(object, ...)
```

Arguments

`object`	Object of class permut
`...`	Additional arguments

Value

Prints permutation test results

Index