Package: DirectEffects (via r-universe)

August 24, 2024

Version 0.2.9000

Title Estimating Controlled Direct Effects for Explaining Causal Findings

Imports stats, Formula, glue, Matching, generics, rlang, broom

Depends R (>= 3.5.0)

Suggests knitr, rmarkdown, dplyr, ggplot2, reshape2, scales, testthat, nnet, glmnet, hdm, ranger

Description A set of functions to estimate the controlled direct effect of treatment fixing a potential mediator to a specific value. Implements the sequential g-estimation estimator described in Vansteelandt (2009) <doi:10.1097/EDE.0b013e3181b6f4c9> and Acharya, Blackwell, and Sen (2016) <doi:10.1017/S0003055416000216> and the telescope matching estimator described in Blackwell and Strezhnev (2020) <doi:10.1111/rssa.12759>.

License GPL (>= 2)

URL `https://mattblackwell.github.io/DirectEffects`

BugReports `https://github.com/mattblackwell/DirectEffects/issues`

VignetteBuilder knitr

LazyData true

Encoding UTF-8

RoxygenNote 7.1.2

Repository https://mattblackwell.r-universe.dev

RemoteUrl https://github.com/mattblackwell/directeffects

RemoteRef HEAD

RemoteSha 16e29e3b66d6cd3277ea51a42c5af7e9e9b3d495

Contents

`balance.tmatch` *Balance diagnostics for Telescope Match objects*

Description

Balance diagnostics for Telescope Match objects

Usage

```
balance.tmatch(object, vars, data, comparison = NULL)
```

Arguments

`object`	an object of class `tmatch` – results from a call to `telescope_match`
`vars`	a formula object containing either the treatment or the mediator as the dependent variable (which denotes whether first-stage or second-stage balance diagnostics are returned) and the covariates for which balance diagnostics are requested as the independent variables. Each covariate or function of covariates (e.g. higher-order polynomials or interactions) should be separated by a `+`.

`data`	the data frame used in the call to `telescope_match`
`comparison`	a binary indicator for if the function should return the balance for the treated group (‘1‘), for the control group (‘0‘), or for overall combined balanced (‘NULL‘, the default).

Details

Provides matching balance diagnostics for `tmatch` objects returned by `telescope_match`

Value

Returns a data frame with the following columns.

- variable: Name of covariate
- before_0: Pre-matching average of the covariate in the mediator == 0 (if first stage balance) or treatment == 0 (if second stage balance) condition
- before_1: Pre-matching average of the covariate in the mediator == 1 (if first stage balance) or treatment == 1 (if second stage balance) condition
- after_0: Post-matching average of the covariate in the mediator == 0 (if first stage balance) or treatment == 0 (if second stage balance) condition
- after_1: Post-matching average of the covariate in the mediator == 1 (if first stage balance) or treatment == 1 (if second stage balance) condition
- before_sd: standard deviation of the outcome (pre-Matching)
- before_diff: Pre-matching covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance).
- before_std_diff: Pre-matching standardized covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance), Equal to Before_Diff/SD.
- after_diff: Post–matching covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance).
- after_std_diff: Post-matching standardized covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance), Equal to Before_Diff/SD.

`balance_table` *Balance diagnostics telescope matching*

Description

Provides matching balance diagnostics for telescope matching CDE estimators

Usage

```
balance_table(object, vars, data, comparison = NULL)
```

Arguments

`object`	output from an estimated `cde_telescope_match` estimator
`vars`	a formula object containing either the treatment or the mediator as the dependent variable (which denotes whether first-stage or second-stage balance diagnostics are returned) and the covariates for which balance diagnostics are requested as the independent variables. Each covariate or function of covariates (e.g. higher-order polynomials or interactions) should be separated by a `+`.
`data`	the data frame used in the call to `estimate` on the `cde_telescope_match` object.
`comparison`	a binary indicator for if the function should return the balance for the treated group (‘1‘), for the control group (‘0‘), or for overall combined balanced (‘NULL‘, the default).

Value

Returns a data frame with the following columns.

- variable: Name of covariate
- before_0: Pre-matching average of the covariate in the mediator == 0 (if first stage balance) or treatment == 0 (if second stage balance) condition
- before_1: Pre-matching average of the covariate in the mediator == 1 (if first stage balance) or treatment == 1 (if second stage balance) condition
- after_0: Post-matching average of the covariate in the mediator == 0 (if first stage balance) or treatment == 0 (if second stage balance) condition
- after_1: Post-matching average of the covariate in the mediator == 1 (if first stage balance) or treatment == 1 (if second stage balance) condition
- before_sd: standard deviation of the outcome (pre-Matching)
- before_diff: Pre-matching covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance).
- before_std_diff: Pre-matching standardized covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance), Equal to Before_Diff/SD.
- after_diff: Post–matching covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance).
- after_std_diff: Post-matching standardized covariate difference between mediator arms (if first stage balance) or treatment arms (if second stage balance), Equal to Before_Diff/SD.

`boots_g` *Coefficient Estimates across Bootstrapped Samples*

Description

Performs a simple bootstrap of a fitted DirectEffects model by re-estimating the model with bootstrap samples.

Usage

```
boots_g(seqg, boots = 1000)
```

Arguments

`seqg`	A fitted sequential_g estimate, computed by `sequential_g`.
`boots`	The number of bootstrap replicates. Defaults to 1000.

Value

An object of type `seqgboots` which is a matrix with `boots` rows and columns for each coefficient in the seqg model. Use `summary` to provide summary statistics, such as mean and quantiles.

Examples

```
data(ploughs)
form <- women_politics ~ plow +
 agricultural_suitability + tropical_climate + large_animals + rugged |
 years_civil_conflict + years_interstate_conflict  + oil_pc +
 european_descent + communist_dummy + polity2_2000 |
 centered_ln_inc + centered_ln_incsq
s1 <- sequential_g(form, ploughs)

out.boots <- boots_g(s1)

summary(out.boots)
```

`boots_tm` *Bootstrap Uncertainty Estimates for Telescope Matching*

Description

Performs a weighted bootstrap procedure for the output of `telescope_match`.

Usage

```
boots_tm(obj, boots = 1000, ci_alpha = 0.05)
```

Arguments

`obj`	A `tmatch` object, computed by `telescope_match`.
`boots`	The number of bootstrap replicates. Defaults to 1000.
`ci_alpha`	alpha value for the bootstrapped confidence intervals. Corresponds to a 100 * (1-alpha) confidence interval.

Value

An data.frame with columns ‘ci_low‘ and ‘ci_high‘ which contain the bootstrapped confidence intervals for the estimated ACDEs in `obj$tau`.

Examples

```
data(jobcorps)

## Split male/female
jobcorps_female <- subset(jobcorps, female == 1)

## Telescope matching formula - First stage (X and Z)
tm_form <- exhealth30 ~  schobef + trainyrbef + jobeverbef  |
treat | emplq4 + emplq4full | work2year2q


### Estimate ACDE for women holding employment at 0
tm_out <-  telescope_match(
  tm_form,
  data = jobcorps_female,
  L = 3,
  boot = FALSE,
  verbose = TRUE
)

out.boots <- boots_tm(tm_out)

out.boots
```

`cdesens` *Estimate sensitivity of ACDE estimates under varying levels of unobserved confounding*

Description

Estimate how the Average Controlled Direct Effect varies by various levels of unobserved confounding. For each value of unmeasured confounding, summarized as a correlation between residuals, cdesens computes the ACDE. Standard errors are computed by a simple bootstrap.

Usage

```
cdesens(
  seqg,
  var,
  rho = seq(-0.9, 0.9, by = 0.05),
  bootstrap = c("none", "standard"),
  boots_n = 1000,
  verbose = FALSE,
```

```
  ...
)
```

Arguments

`seqg`	Output from sequential_g. The function only supports specifications with one mediator variable.
`var`	A character indicating the name of the variable for which the estimated ACDE is being evaluated.
`rho`	A numerical vector of correlations between errors to test for. The original model assumes `rho` = `0`
`bootstrap`	character of c("none", "standard"), indicating whether to include bootstrap standard errors. Default is "none".
`boots_n`	Number of bootstrap replicates, defaults to 100.
`verbose`	Whether to show progress and messages, defaults to `FALSE`
`...`	Other parameters to pass on to `lm.fit()` when refitting the model

Examples

```
data(civilwar)


# main formula: Y ~ A + X | Z | M
form_main <- onset ~ ethfrac + lmtnest + ncontig + Oil | warl +
  gdpenl + lpop + polity2l + relfrac | instab

# estimate CDE
direct <- sequential_g(form_main, data = civilwar)

# sensitivity
out_sens <- cdesens(direct, var = "ethfrac")

# plot sensitivity
plot(out_sens)
```

`cde_aipw` *Initialize an AIPW CDE estimator*

Description

Initializes the specification of a CDE estimator based on an augmented inverse probability weighting approach.

Usage

```
cde_aipw(trim = c(0.01, 0.99), aipw_blip = TRUE)
```

Arguments

`trim`	A vector of length 2 indicating what quantiles of the propensity scores should be trimmed. By default this is `c(0.01, 0.99)` meaning that the top and bottom 1% of propensity scores are trunctated to these quantiles. If NULL, no trimming occurs.
`aipw_blip`	If `TRUE` (the default), augmented inverse probability weighting estimators will be used to estimate intermediate outcome regressions (blip functions).

`cde_did_aipw` *Initialize an AIPW DID-CDE estimator*

Description

Initializes the specification of a difference-in-differences estimator for the CDE based on an augmented inverse probability weighting.

Usage

```
cde_did_aipw(
  base_mediator,
  trim = c(0.01, 0.99),
  aipw_blip = TRUE,
  on_treated = FALSE
)
```

Arguments

`base_mediator`	The (unquoted) name of the variable that measures the mediator at baseline.
`trim`	A vector of length 2 indicating what quantiles of the propensity scores should be trimmed. By default this is `c(0.01, 0.99)` meaning that the top and bottom 1% of propensity scores are trunctated to these quantiles. If NULL, no trimming occurs.
`aipw_blip`	If `TRUE` (the default), augmented inverse probability weighting estimators will be used to estimate intermediate outcome regressions (blip functions).
`on_treated`	If `FALSE` (the defafult), the effects are average effects conditional on the levels of the baseline mediator. If `TRUE`, the effects are conditional on the treated path. For difference in identficiation, see Details below.

Details

This function, unlike other CDE estimators in the package, only returns the estimated effects of the first treatment variable. These effects are conditional on the baseline value of the mediator (`base_mediator`) when `on_treated` is `TRUE`. A marginalized CDE estimand is also estimated. When `on_treated` is `FALSE`, these estimates are conditional on the entire "treated" history. Identification requirements are slightly different between these two cases. When `on_treated` is `FALSE`, the confounders for the mediator cannot be affected by treatment. See Blackwell et al (2022) for more information.

`cde_ipw` *Initialize an IPW CDE estimator*

Description

Initializes the specification of a CDE estimator based on an inverse probability weighting approach.

Usage

```
cde_ipw(hajek = TRUE, trim = c(0.01, 0.99))
```

Arguments

`hajek`	If `TRUE`, normalized weights will be used as in the Hajek estimator. If `FALSE`, traditional IPW weights will be used.
`trim`	A vector of length 2 indicating what quantiles of the propensity scores should be trimmed. By default this is `c(0.01, 0.99)` meaning that the top and bottom 1% of propensity scores are tructated to these quantiles. If NULL, no trimming occurs.

`cde_reg_impute` *Initialize an regression imputation CDE estimator*

Description

Initializes the specification of a CDE estimator based on an regression imputation approach

Usage

```
cde_reg_impute(...)
```

Arguments

`...`	Optional arguments to pass to the regression imputation estimator.

`cde_telescope_match` *Initialize an telescope matching CDE estimator*

Description

Initializes the specification of a CDE estimator based on an telescope matching approach

Usage

```
cde_telescope_match(...)
```

Arguments

`...`	Optional arguments to pass to the telescope matching estimator.

`civilwar` *Data on civil wars and internal conflict from 1945-1999.*

Description

A dataset to replicate the analysis in Fearon and Laitin (2003).

Usage

```
data(civilwar)
```

Format

A data frame with 6610 observations and 69 variables.

Details

- ccode. COW country id number
- country. country name
- cname. abbreviated country name
- cmark. 1 for first in each country series
- year. start year of war/conflict
- wars. number wars in progress in country year
- war. 1 if war ongoing in country year
- warl. lagged war, w/ 0 for start of country series
- onset. 1 for civil war onset
- ethonset. 1 if onset = 1 & ethwar ~= 0
- durest. estimated war duration

- aim. 1 = rebels aim at center, 3 = aim at exit or autonomy, 2 = mixed or ambig.
- casename. Id for case, usually name of rebel group(s)
- ended. war ends = 1, 0 = ongoing
- ethwar. 0 = not ethnic, 1 = ambig/mixed, 2 = ethnic
- waryrs. war years for each onset
- pop. population, in 1000s
- lpop. log of pop
- polity2. revised polity score
- gdpen. gdp/pop based on pwt5.6, wdi2001,cow energy data
- gdptype. source/type of gdp/pop estimate
- gdpenl. lagged gdpenl, except for first in country series
- lgdpenl1. log of lagged gdpen
- lpopl1. log population, lagged except for first in country series
- region. country's region, based on MAR project
- western. Dummy for Western Democracies & Japan
- eeurop. Dummy for Eastern Europe
- lamerica. Dummy for Latin America
- ssafrica. Dummy for Sub-Saharan Africa
- asia. Dummy for Asia (not including Japan)
- nafrme. Dummy for North Africa/Middle East
- colbrit. Former British colony
- colfra. former French colony
- mtnest. Estimated percent mountainous terrain
- lmtnest. log of mtnest
- elevdiff. high - low elevation, in meters
- Oil. more than 1/3 export revenues from fuels
- ncontig. noncontiguous state
- ethfrac. ethnic frac. based on Soviet Atlas, plus estimates for missing in 1964
- ef. ethnic fractionalization based on Fearon 2002 APSA paper
- plural. share of largest ethnic group (Fearon 2002 APSA)
- second. share of 2nd largest ethnic group (Fearon 2002 APSA)
- numlang. number languages in Ethnologue > min(1
- relfrac. religious fractionalization
- plurrel. size of largest confession
- minrelpc. size of second largest confession
- muslim. percent muslim
- nwstate. 1 in 1st 2 years of state's existence

- polity2l. lagged polity2, except 1st in country series
- instab. > 2 change in Polity measure in last 3 yrs
- anocl. lagged anocracy (-6 < polity2l < 6)
- deml. lagged democracy (polity2l > 5)
- empethfrac. ethfrac coded for colonial empires
- empwarl. warl coded for data with empires
- emponset. onset coded for data with empires
- empgdpenl. gdpenl coded for empires data
- emplpopl. lpopl coded for empires data
- emplmtnest. lmtnest coded for empires data
- empncontig. ncontig coded for empires
- empolity2l. polity2l adjusted for empires (see fn38 in paper)
- sdwars. number Sambanis/Doyle civ wars in progress
- sdonset. onset of Sambanis/Doyle war
- colwars. number Collier/Hoeffler wars in progress
- colonset. onset of Collier/Hoeffler war
- cowwars. number COW civ wars in progress
- cowonset. onset of COW civ war
- cowwarl. 1 if COW war ongoing in last period
- sdwarl. 1 if S/D war ongoing in last period
- colwarl. 1 if C/H war ongoing in last period

Source

doi:10.1017/S0003055403000534

References

Fearon, James D., and David A. Laitin (2003). Ethnicity, Insurgency, and Civil War. American Political Science Review, 97(1), 75-90. doi:10.1017/S0003055403000534

`estimate` *Fit a specified CDE estimator*

Description

Fit a CDE estimator with the engines specified in the `model_spec` object.

Usage

```
estimate(
  object,
  formula,
  data,
  subset,
  crossfit = TRUE,
  n_folds,
  n_splits = 1L
)
```

Arguments

`object`	A `cde_estimator` object that has already been passed to at least one call to `set_treatment`.
`formula`	A formula object with describing the outcome of interest on the left-hand side and the treatment variables the user wants to estimate effects for (which might be a subset of the treatment variables specified).
`data`	A `data.frame` containing all variables, including treatment variables and covariates specified.
`subset`	Anan optional vector specifying a subset of observations to be used in the fitting process.
`crossfit`	A logical indicator for if cross-fitting should be used in estimating the effects.
`n_folds`	The number of folds to use within a given instance of the cross-fitting algorithm.
`n_splits`	The number of times the cross-fitting procedure should be repeated. Overall estimates use the median value of these repeated estimates.

`jobcorps` *Data on health and employment outcomes measured as part of the U.S. Job Corps employment training experiment.*

Description

A dataset to replicate the analysis in Huber (2014).

Usage

```
data(jobcorps)
```

Format

A data frame with 10025 observations and 62 variables.

Details

- treat. 1 = in program group. 0 = in control group.
- schobef. "in school 1yr before eligibility"
- trainyrbef. "training in year before Job Corps"
- jobeverbef. "ever had a job before Job Corps"
- jobyrbef. "job in year before job corps"
- health012. "good or very good health at assignment"
- health0mis. "general health at assignment missing"
- pe_prb0. "physical/emotional problems at assignment"
- pe_prb0mis. "missing - physical/emotional problems at assignment"
- everalc. "ever abused alcohol before assignment"
- alc12. "alcohol abuse one yr after assignment"
- everilldrugs. "ever took illegal drugs before assignment"
- age_cat. "age at application in years 16-24"
- edumis. "education missing"
- eduhigh. "higher education"
- rwhite. "white"
- everarr. "ever arrested before Job Corps"
- hhsize. "household size at assignment"
- hhsizemis. "household size at assignment missing"
- hhinc12. "low household income at assignment"
- hhinc8. "high household income at assignment"
- fdstamp. "received foodstamps in yr before assignment"
- welf1. "once on welfare while growing up"
- welf2. "twice on welfare while growing up"
- publicass. "public assistance in yr before assignment"
- emplq. "worked some time 9-12 months after assignment"
- emplq4full. "worked all the time in 9-12 months after assignment"
- pemplq4. "proportion of weeks worked 9-12 months after assignment"
- pemplq4mis. "missing - proportion of weeks worked 9-12 months after assignment"
- vocq4. "in vocational training 9-12 months after assignment"
- vocq4mis. "missing - in vocational training 9-12 months after assignment"
- health1212. "very good or good health 1 yr after assignment"
- health123. "fair health 1 yr after assignment"
- pe_prb12. "1=phys/emot probs at 12 mths 0=no prob"
- pe_prb12mis. "missing - physical/emotional problems 1 yr after assignment"
- narry1. "number of arrests in year 1"

- numkidhhf1zero. "no own kids living in household 1 yr after assignment"
- numkidhhf1onetwo. "one or two own kids living in household 1 yr after assignment"
- pubhse12. "1=in public housing 1 yr after assignment, 0=not in"
- h_ins12a. "afdc and other transfers one yr after assignment"
- h_ins12amis. "missing - afdc and other transfers one yr after assignment"
- ... other variables as annotated in the source.

Source

doi:10.1002/jae.2341

References

Huber, M. (2014). Identifying causal mechanisms (primarily) based on inverse probability weighting. Journal of Applied Econometrics, 29(6), 920-943. doi:10.1002/jae.2341

`outreg_model` *Specify the outcome regression model for a CDE treatment*

Description

Specifies the functional form and estimation engine for an outcome regression of a treatment previously specified by `set_treatment()` and the past history of covariates.

Usage

```
outreg_model(
  object,
  formula,
  engine,
  separate = TRUE,
  include_past = TRUE,
  ...
)
```

Arguments

`object`	A `cde_estimator` object that contains output from a previous call to `set_treatment()`.
`formula`	A formula specifying the design matrix of the covariates. Passed to fitting engine or used with `stats::model.frame()` and `stats::model.matrix()` to create the design matrix for fitting engines that do not take formulas.
`engine`	String indicating the name of the fitting engine.
`separate`	Logical indicating whether the fitting algorithm should be applied separately to each history of the treatment variables up to this point (default) or not.
`include_past`	A logical value where `TRUE` indicates that formulas passed to previous `treat_model` calls should be appended to the formula given.
`...`	Other arguments to be passed to the engine algorithms.

`plot.cdesens` *Plot output from cdesens*

Description

Plot output from cdesens

Usage

```
## S3 method for class 'cdesens'
plot(
  x,
  level = 0.95,
  xlim = NULL,
  ylim = NULL,
  xlab = NULL,
  ylab = "Estimated ACDE",
  bty = "n",
  col = "black",
  lwd = 2,
  ci.col = "grey70",
  ref.lines = TRUE,
  ...
)
```

Arguments

`x`	output from `cdesens`
`level`	level of confidence interval to plot
`xlim`	the x limits (x1, x2) of the plot for the sensitivity analysis parameter, rho. Default is to use the range of `rho`.
`ylim`	the y limits of the plot for the estimated CDEs. Default is to show the all of the confidence intervals.
`xlab`	label for the x axis.
`ylab`	label for the y axis.
`bty`	a character string which determined the type of box which is drawn about plots. Defaults to not drawing a box. See par for more information.
`col`	color for the line indicating the point estimates of the bias-adjusted ACDE.
`lwd`	line width for the line indicating the point estimates of the bias-adjusted ACDE.
`ci.col`	color for the polygon that shows the confidence intervals.
`ref.lines`	a logical indicating whether horizontal and vertical lines at 0 should be plotted.
`...`	Other parameters to pass on to `plot()`

`plotDiag.tmatch` *Histograms of matching weights*

Description

Histograms of matching weights

Usage

```
plotDiag.tmatch(object, stage)
```

Arguments

`object`	an object of class `tmatch` – results from a call to `telescope_match`
`stage`	a character vector equal to the name of one treatment from the ‘object‘.

Details

Provides histograms of the number of times each unit is used as a match given a `tmatch` object returned by `telescope_match`

Value

Outputs a ‘plot()‘ object containing the histogram of match counts

---

`ploughs` *Data on historical plough use and the socioeconomic status of women.*

Description

A dataset to replicate the analysis in Alesina, Giuliano, and Nunn (2013).

Usage

```
data(ploughs)
```

Format

A data frame with 234 observations and 57 variables.

Details

- isocode. 3-letter code for the country.
- flfp2000. Female labor force participation in 2000
- female_ownership. Percent of firms with female ownership (in latest survey year)
- women_politics. Women in Politics in 2000, WDI
- plow. Animal plow cultivation variable (v39): Using Ethnologue - pop weighted
- agricultural_suitability. overall (millets, sorghum, wheat, barley, rye): share defined as suitable
- tropical_climate. Frac land: tropics and subtropics: using Ethnologue - pop weighted
- large_animals. presence of large animals
- political_hierarchies. Jurisdictional hierarchy beyond local community (v33): Using Ethnologue - pop weighted
- economic_complexity. Settlement patterns (v30)
- ln_income. ln (income)
- ln_income_squared. ln (income) ^2
- centered_ln_inc. de-meaned ln_inc
- centered_ln_incsq. de-meaned ln_inc squared
- country. country name
- communist_dummy. Communism indicator variable
- rugged. Ruggedness (Terrain Ruggedness Index, 100 m.)
- years_interstate_conflict. Years of interstate conflict, 1800-2007 - from COW
- serv_va_gdp2000. Value Added in Service/GDP in 2000
- polity2_2000. Polity 2 measure taken from the Polity IV dataset
- oil_pc. oil production/GDP
- ... other variables as annotated in the source.

Source

doi:10.1093/qje/qjt005

References

Alesina, A., Giuliano, P., & Nunn, N. (2013). On the Origins of Gender Roles: Women and the Plough. The Quarterly Journal of Economics, 128(2), 469-530. doi:10.1093/qje/qjt005

`sequential_g`	*Perform linear sequential g-estimation to estimate the controlled direct effect of a treatment net the effect of a mediator.*

Description

Perform linear sequential g-estimation to estimate the controlled direct effect of a treatment net the effect of a mediator.

Usage

```
sequential_g(
  formula,
  data,
  subset,
  weights,
  na.action,
  offset,
  contrasts = NULL,
  verbose = TRUE,
  ...
)
```

Arguments

`formula`	formula specification of the first-stage, second-stage, and blip-down models. The right-hand side of the formula should have three components separated by the `|`, with the first component specifying the first-stage model with treatment and any baseline covariates, the second component specifying the intermediate covariates for the first-stage, and the third component specifying the blip-down model. See Details below for more information.
`data`	A dataframe to apply `formula` on.
`subset`	A vector of logicals indicating which rows of `data` to keep.
`weights`	an optional vector of weights to be used in the fitting process. Should be `NULL` or a numeric vector. If non-NULL, weighted least squares is used with weights `weights` (that is, minimizing `sum(w*e^2)`); otherwise ordinary least squares is used. See also 'Details',
`na.action`	a function which indicates what should happen when the data contain `NA`s. The default is set by the `na.action` setting of `options`, and is `na.fail` if that is unset. The 'factory-fresh' default is `na.omit`. Another possible value is `NULL`, no action. Value `na.exclude` can be useful.
`offset`	this can be used to specify an *a priori* known component to be included in the linear predictor during fitting. This should be `NULL` or a numeric vector or matrix of extents matching those of the response. One or more `offset` terms can be included in the formula instead or as well, and if more than one are specified their sum is used. See `model.offset`.

`contrasts`	an optional list. See the `contrasts.arg` of `model.matrix.default`.
`verbose`	logical indicating whether to suppress progress bar. Default is FALSE.
`...`	additional arguments to be passed to the low level regression fitting functions (see below).

Details

The `sequential_g` function implements the linear sequential g-estimator developed by Vansteelandt (2009) with the consistent variance estimator developed by Acharya, Blackwell, and Sen (2016).

The formula specifies specifies the full first-stage model including treatment, baseline confounders, intermediate confounders, and the mediators. The user places `|` bars to separate out these different components of the model. For example, the formula should have the form `y ~ tr + x1 + x2 | z1 + z2 | m1 + m2`. where `tr` is the name of the treatment variable, `x1` and `x2` are baseline covariates, `z1` and `z2` are intermediate covariates, and `m1` and `m2` are the names of the mediator variables. This last set of variables specify the 'blip-down' or 'demediation' function that is used to remove the average effect of the mediator (possibly interacted) from the outcome to create the blipped-down outcome. This blipped-down outcome is the passed to a standard linear model with the covariates as specified for the direct effects model.

See the references below for more details.

Value

Returns an object of `class` A `"seqg"`. Similar to the output of a call to `lm`. Contains the following components:

- coefficients: a vector of named coefficients for the direct effects model.
- residuals: the residuals, that is the blipped-down outcome minus the fitted values.
- rank: the numeric rank of the fitted linear direct effects model.
- fitted.values: the fitted mean values of the direct effects model.
- weights: (only for weighted fits) the specified weights.
- df.residual: the residual degrees of freedom for the direct effects model.
- aliased: logical vector indicating if any of the terms were dropped or aliased due to perfect collinearity.
- terms: the list of `terms` object used. One for the baseline covariates and treatment (`X`) and one for the variables in the blip-down model (`M`).
- formula: the `formula` object used, possibly modified to drop a constant in the blip-down model.
- call: the matched call.
- na.action: (where relevant) information returned by `model.frame` of the special handling of `NA`s.
- xlevels: the levels of the factor variables.
- contrasts: the contrasts used for the factor variables.
- first_mod: the output from the first-stage regression model.

- model: full model frame, including all variables.
- Ytilde: the blipped-down response vector.
- X: the model matrix for the second stage.
- M: the model matrix for demediation/blip-down function.

In addition, non-null fits will have components `assign`, `effects`, and `qr` from the output of `lm.fit` or `lm.wfit`, whichever is used.

References

Vansteelandt, S. (2009). Estimating Direct Effects in Cohort and Case-Control Studies. Epidemiology, 20(6), 851-860.

Acharya, Avidit, Blackwell, Matthew, and Sen, Maya. (2016) "Explaining Causal Effects Without Bias: Detecting and Assessing Direct Effects." American Political Science Review 110:3 pp. 512-529

Examples

```
data(ploughs)

form_main <- women_politics ~ plow +
  agricultural_suitability + tropical_climate + large_animals +
  political_hierarchies + economic_complexity +
  rugged | years_civil_conflict +
  years_interstate_conflict  + oil_pc +
  european_descent + communist_dummy + polity2_2000 +
  serv_va_gdp2000 | centered_ln_inc + centered_ln_incsq

direct <- sequential_g(form_main, ploughs)

summary(direct)
```

`set_treatment` *Specifiy a treatment variable for a controlled direct effect*

Description

This function specifies a treatment variable in the sequence of treatment variables that define the controlled direct effect of interest.

Usage

```
set_treatment(
  object,
  treat,
  formula = NULL,
  treat_type = "categorical",
  eval_vals = NULL
)
```

Arguments

`object`	A `cde_estimator` object that may or may have previous treatment variables specified/
`treat`	Name of the treatment variable (not quoted).
`formula`	One-sided formula giving the covariates that are pre-treatment to this treatment, but post-treatment to any previous treatment. Unless overridden by the arguments to `treat_model()` or `outreg_model()`, this formula will be the specification used in the modeling of the propensity scores or outcome regressions.
`treat_type`	A string indicating the type of variable this is. Takes either the values `"categorical"` or `"regression"` (the latter is not yet implemented). of
`eval_vals`	A numeric vector of values of this variable to evaluate the controlled direct effecct. If `NULL` (the default), this will be set to all observed values of the variable.

Value

An updated `cde_estimator` with this information about the treatment specified.

Author(s)

Matthew Blackwell

`summary.seqg` *Computes standard errors and p-values of DirectEffects estimates*

Description

Computes standard errors and p-values of DirectEffects estimates

Usage

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

Arguments

`object`	An object of class seqg, computed by `sequential_g`.
`...`	additional arguments affecting the summary produced.

`summary.seqgboots` *Summary of DirectEffect Bootstrap Estimates*

Description

Summary of DirectEffect Bootstrap Estimates

Usage

```
## S3 method for class 'seqgboots'
summary(object, level = 0.95, ...)
```

Arguments

`object`	An output of class seqg estimated by `boots_g`.
`level`	level of intervals to estimate. Defaults to 0.95
`...`	additional arguments affecting the summary produced.

`summary.tmatch` *Summarize telescope match objects*

Description

Summarize telescope match objects

Usage

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

Arguments

`object`	an object of class `tmatch` – results from a call to `telescope_match`
`...`	additional arguments affecting the summary produced.

Details

`summary` method for `tmatch` objects returned by `telescope_match`

Returns a summary data frame containing the estimate and standard errors from the ‘telescope_match‘ object.

Value

Returns an object of `class summary.tmatch`. Contains the following components

- `call`: matched call.
- `m_summary`: data.frame summarizes the matching ratios (`{ratio}`), number of units `n_1, n_0`, and number of matched units (`matched_1, matched_0`) for each treatment/mediator (`term`).
- `K`: `K` data frame from the `object` telescope matching output.
- `L`: `L` vector from the `object` telescope matching output.
- `a_names`: character vector of the names of the treatment/mediator variables used in matching.
- `estimates`: matrix of estimated ACDEs with and without bias correction and the estimated standard errors.

`telescope_match` *Perform telescope matching to estimate the controlled direct effect of a binary treatment net the effect of binary mediators*

Description

Perform telescope matching to estimate the controlled direct effect of a binary treatment net the effect of binary mediators

Usage

```
telescope_match(
  formula,
  data,
  caliper = NULL,
  L = 5,
  verbose = TRUE,
  subset,
  contrasts = NULL,
  separate_bc = TRUE,
  ...
)
```

Arguments

`formula`	A formula object that specifies the covariates and treatment variables (or mediators) in causal ordering from oldest to newest with each group separated by `|`. See below for more details.
`data`	A dataframe containing variables referenced by `formula`.
`caliper`	A scalar denoting the caliper to be used in matching in the treatment stage (calipers cannot be used for matching on the mediator). Observations outside of the caliper are dropped. Calipers are specified in standard deviations of the covariates. NULL by default (no caliper).

`L`	Number of matches to use for each unit. Must be a numeric vector of either length 1 or 2. If length 1, L sets the number of matches used in both the first stage (matching on mediator) and in the second stage (matching on treatment). If length 2, the first element sets the number of matches used in the first stage (matching on mediator) and the second element sets the number of matches used in the second stage (matching on treatment) Default is 5.
`verbose`	logical indicating whether to display progress information. Default is `TRUE`.
`subset`	A vector of logicals indicating which rows of `data` to keep.
`contrasts`	a list to be passed to the `contrasts.arg` argument of `model.matrix()` when generating the data matrix.
`separate_bc`	logical indicating whether or not bias correction regressions should be run separately within levels of the treatment and mediator. Defaults to `TRUE`. If `TRUE`, any interactions between treatment/mediator and covariates in the specification should be omitted.
`...`	additional arguments to be passed to the low level regression fitting functions (see below).

Details

The `telescope_match` function implements the two-stage "telescope matching" procedure developed by Blackwell and Strezhnev (2021).

The procedure first estimates a demediated outcome using a combination of matching and a regression bias-correction. The `data.frame` passed to `data` should be in the wide format so that each row corresponds to a single unit and treatments and covariates from different time periods appear as different columns. The `formula` argument specifies both the causal ordering of the variables and the regression specifications for the bias correction. It should be of the form `Y ~ X1 | A1 | X2 | A2`, where `Y` is the outcome, `X1` is a formula of baseline covariates, `A1` is a single variable name indicating the binary treatment in the first period, `X2` is a formula of covariates in period 2, and `A2` is a single variable name indicating treatment in period 2 (which is also sometimes called the mediator). Note that it is possible to add more covariate/treatment pairs for additional time periods.

Under the default `separate_bc == TRUE`, the function will match for each treatment/mediator based on the the covariates up to that point within levels of past treatments (so for `A2` this matching finds units with similar values of `X1` and `X2` and the same value of `A1`). Once this matching is complete, the function moves backward through treatments and imputes potential outcomes using matches and bias-correction regressions, which regress the current imputed potential outcome on the past covariates, within levels of the treatment history up to the current period. The functional form comes from the specification in `formula`. Controlled direct effects of `A1` are estimated for every possible combination of future treatments.

When `separate_bc` is `FALSE`, the bias correction regressions are not broken out by the treatments/mediators and those variables are simply included as separate regressors as specified in `formula`. In this setting, interactions between the treatment/mediator and covariates can be added on a selective basis to the covariate block (`X1` or `X2` and so on) specifications.

Matching is performed using the `Match()` routine from the `Matching` package. By default, matching is L-to-1 nearest neighbor with replacement using Mahalanobis distance.

See the references below for more details.

Value

Returns an object of `class tmatch`. Contains the following components

- `call`: the matched call.
- `formula`: formula used to fit the model.
- `m_out`: list of matching solutions at each time point. Each member of the list has a ‘matches‘ list giving the units matched to that unit, a ‘donors‘ list with the units to which the unit is matched, and a ‘tr‘ vector which is just the treatment vector being matched.
- `K`: data.frame of indicating how many times a unit has been used as a match, directly in each period and indirectly across periods.
- `L`: vector of matching ratios used in each period.
- `r_out`: nested list of regression imputations used in the bias correction. The first level of the list varies across different controlled direct effects (different sequences of future treatments/mediators). Each of these is a list of time periods and each of these time periods is a list of ‘yhat_r_0‘ and ‘yhat_r_1‘ that give the regression predictions for the potential outcomes at that time point when the treatment at that time point is 0 or 1, respectively, along with ‘n_coefs‘ giving the number of coefficients estimated in those models.
- `tau`: vector of bias-corrected estimates of average controlled direct effects for different vectors of future treatments/mediators.
- `tau_raw`: vector of standard matching estimates of average controlled direct effects for different vectors of future treatments/mediators without using bias correction.
- `tau_se`: vector of estimated standard errors for the average controlled direct effects estimates for different vectors of future treatments/mediators.
- `tau_i`: matrix of individuals contributions to the ACDE estimates (units on rows, different ACDEs on columns). Used for weighted bootstrap.
- `included`: logical vector indicating if each row of `data` was included in estimating `tau`.
- `effects`: data frame where each row describes the different ACDEs in `tau`. The `active` column describes the which variable’s direct effect is being assessed and the rest of the columns describe the fixed values of the future treatments/mediators for that ACDE.
- `a_names`: character vector with the names of the treatment/mediator variables used in estimation.
- `caliper`: caliper (if any) used in matching to drop distant observations.

References

Blackwell, Matthew, and Strezhnev, Anton (2020) "Telescope Matching: Reducing Model Dependence in the Estimation of Direct Effects." Journal of the Royal Statistical Society (Series A). doi:10.1111/rssa.12759

Examples

```
data(jobcorps)

## Split male/female
jobcorps_female <- subset(jobcorps, female == 1)
```

```
## Telescope matching formula - First stage (X and Z)
tm_form <- exhealth30 ~  schobef + trainyrbef + jobeverbef  |
treat | emplq4 + emplq4full | work2year2q


### Estimate ACDE for women holding employment at 0
tm_out <-  telescope_match(
  tm_form,
  data = jobcorps_female,
  L = 3,
  boot = FALSE,
  verbose = TRUE
)
```

`treat_model` *Specify the propensity score model for a CDE treatment*

Description

Specifies the functional form and estimation engine for a treatment previously specified by `set_treatment()`.

Usage

```
treat_model(object, formula, engine, separate = TRUE, include_past = TRUE, ...)
```

Arguments

`object`	A `cde_estimator` object that contains output from a previous call to `set_treatment()`.
`formula`	A formula specifying the design matrix of the covariates. Passed to fitting engine or used with `stats::model.frame()` and `stats::model.matrix()` to create the design matrix for fitting engines that do not take formulas.
`engine`	String indicating the name of the fitting engine.
`separate`	Logical indicating whether the fitting algorithm should be applied separately to each history of the treatment variables up to this point (default) or not.
`include_past`	A logical value where `TRUE` indicates that formulas passed to previous `treat_model` calls should be appended to the formula given.
`...`	Other arguments to be passed to the engine algorithms.

Author(s)

Matthew Blackwell

Index