Package 'rbw'

April 15, 2020

Type Package

Title Residual Balancing Weights for Marginal Structural Models

Version 0.2.0

Description Residual balancing is a robust method of constructing weights for
marginal structural models, which can be used to estimate marginal effects of
time-varying treatments and controlled direct/mediator effects in causal mediation
analysis (Zhou and Wodtke 2020 <doi:10.1017/pan.2020.2>). This
package provides two main functions, rbwPanel() and rbwMed(), that produce
residual balancing weights for analyzing time-varying treatments and
causal mediation respectively.

Depends R (>= 3.3.0),

Imports stats, rlang (>= 0.4.4)

Suggests ebal, survey

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

URL http://github.com/xiangzhou09/rbw

BugReports http://github.com/xiangzhou09/rbw

NeedsCompilation no

Author Xiang Zhou [aut, cre]

Maintainer Xiang Zhou <xiang_zhou@fas.harvard.edu>

Repository CRAN

Date/Publication 2020-04-15 11:00:02 UTC

R topics documented:

1

campaign_long   *Long-format Data on Negative Campaign Advertising in US Senate and Gubernatorial Elections*

Description

A dataset containing 17 variables and 565 unit-week records on the campaign of 113 Democratic candidates in US Senate and Gubernatorial Elections from 2000 to 2006 (Blackwell 2013).

Usage

```
campaign_long
```

Format

A data frame with 565 rows and 17 columns:

demName name of the Democratic candidate

d.gone.neg whether the candidate went negative in a campaign-week, defined as whether more than 10% of the candidate's political advertising was negative

d.gone.neg.l1 whether the candidate went negative in the previous campaign-week

camp.length length of the candidate's campaign (in weeks)

deminc whether the candidate was an incumbent

base.poll Democratic share in the baseline polls

base.und share of undecided voters in the baseline polls

office type of office in contest. 0: governor; 1: senator

demprcnt Democratic share of the two-party vote in the election

week week in the campaign (in the final five weeks preceding the election)

year year of the election

state state of the election

dem.polls Democratic share in the polls

dem.polls.l1 Democratic share in the polls in the previous campaign-week

undother share of undecided voters in the polls

undother.l1 share of undecided voters in the polls in the previous campaign-week

id candidate id

References

Blackwell, Matthew. 2013. A Framework for Dynamic Causal Inference in Political Science. American Journal of Political Science 57(2): 504-619.

| campaign_wide | *Wide-format Data on Negative Campaign Advertising in US Senate and Gubernatorial Elections* |

Description

A dataset containing 26 variables and 113 unit records from Blackwell (2013).

Usage

```
campaign_wide
```

Format

A data frame with 565 rows and 26 columns:

demName name of the Democratic candidate

camp.length length of the candidate's campaign (in weeks)

deminc whether the candidate was an incumbent.

base.poll Democratic share in the baseline polls

base.und share of undecided voters in the baseline polls

office type of office in contest. 0: governor; 1: senator

demprcnt Democratic share of the two-party vote in the election

year year of the election

state state of the election

id candidate id

d.gone.neg_1 whether the candidate went negative in week 1

d.gone.neg_2 whether the candidate went negative in week 2

d.gone.neg_3 whether the candidate went negative in week 3

d.gone.neg_4 whether the candidate went negative in week 4

d.gone.neg_5 whether the candidate went negative in week 5

dem.polls_1 Democratic share in week 1 polls

dem.polls_2 Democratic share in week 2 polls

dem.polls_3 Democratic share in week 3 polls

dem.polls_4 Democratic share in week 4 polls

dem.polls_5 Democratic share in week 5 polls

undother_1 share of undecided voters in week 1 polls

undother_2 share of undecided voters in week 2 polls

undother_3 share of undecided voters in week 3 polls

undother_4 share of undecided voters in week 4 polls

undother_5 share of undecided voters in week 5 polls

cum_neg the total number of campaign-weeks in which a candidate went negative

References

Blackwell, Matthew. 2013. A Framework for Dynamic Causal Inference in Political Science. American Journal of Political Science 57(2): 504-619.

---

eb2	*Function for Generating Minimum Entropy Weights Subject to a Set of Balancing Constraints*

Description

eb2 is an adaptation of eb that generates minimum entropy weights subject to a set of balancing constraints. Using the method of Lagrange multipliers, the dual problem is an unconstrained optimization problem that can be solved using Newton's method. When a full Newton step is excessive, an exact line search is used to find the best step size.

Usage

```
eb2(C, M, Q, Z = rep(0, ncol(C)), max_iter = 200, tol = 1e-04, print_level = 2)
```

Arguments

C	A constraint matrix.
M	A vector of moment conditions to be met in the reweighted sample.
Q	A vector of base weights.
Z	A vector of Lagrange multipliers to be initialized.
max_iter	Maximum number of iterations for Newton's method.
tol	Tolerance parameter used to determine convergence.
print_level	The level of printing:

    1 normal: print whether the algorithm converges or not

    2 detailed: print also the maximum absolute value of the deviation between the moments of the reweighted data and the target moments in each iteration

    3 very detailed: print also the step length of the line searcher in iterations where a full Newton step is excessive.

Value

A list containing the results from the algorithm.

W	A vector of normalized minimum entropy weights.
Z	A vector of Lagrange multipliers.
converged	A logical indicator for convergence.
maxdiff	A scalar indicating the maximum deviation between the moments of the reweighted data and the target moments.

---

education	*Data on Education Attainment and Mental Health in a Sample of US adults.*

Description

A dataset containing 14 variables on education, socioeconomic status, mental health, gender, race, and family background for 2,835 respondents from the National Longitudinal Survey of Youth, 1979 (NLSY79)

Usage

```
education
```

Format

A data frame with 565 rows and 17 columns:

id respondent id

weights NLSY79 sampling weight

college whether the respondent obtained a Bachelor's degree by age 25

ses socioeconomic status, measured by the percentile rank of the respondent's family income at age 40.

cesd40 depression score at age 40, measured by the Center for Epidemiologic Studies Depression Scale

female a dummy variable indicating whether the respondent is female.

race the race of the respondent, 1: Hispanic; 2: black; 3: other.

momedu mother's education, measured by highest grade completed

parinc the percentile rank of parent income in 1979-1983

afqt3 cognitive ability in 1981, measured by the Armed Forces Qualification Test

educexp educational expectation in 1979

cesd92 depression score in 1992, measured by the Center for Epidemiologic Studies Depression Scale

prmarr98 proportion of time married between 1990 and 1998

transitions98 number of family transitions between 1990 and 1998

---

rbwMed   *Residual Balancing Weights for Causal Mediation Analysis*

Description

rbwMed is a function that produces residual balancing weights for causal mediation analysis. The weights can be used to fit marginal structural models for the joint effects of the treatment and a mediator.

Usage

```
rbwMed(
  treatment,
  mediator,
  zmodels,
  data,
  baseline_x,
  interact = FALSE,
  base_weights,
  max_iter = 200,
  print_level = 1,
  tol = 1e-04
)
```

Arguments

treatment	A symbol or character string for the treatment variable.
mediator	A symbol or character string for the mediator variable.
zmodels	A list of fitted lm or glm objects for post-treatment confounders of the mediator-outcome relationship. If there's no post-treatment confounder, set it to be NULL.
data	A data frame containing all variables in the model.
baseline_x	(Optional) An expression for a set of baseline confounders stored in data.
interact	A logical variable indicating whether baseline and post-treatment covariates should be balanced against the treatment-mediator interaction term(s).
base_weights	(Optional) A vector of base weights (or its name).
max_iter	Maximum number of iterations for Newton's method.
print_level	The level of printing:
	1 normal: print whether the algorithm converges or not
	2 detailed: print also the maximum absolute value of the deviation between the moments of the reweighted data and the target moments in each iteration
	3 very detailed: print also the step length of the line searcher in iterations where a full Newton step is excessive.
tol	Tolerance parameter used to determine convergence.

Value

A list containing the results.

weights	A vector of residual balancing weights.
constraints	A matrix of (linearly independent) residual balancing constraints
eb_out	Results from calling eb2 function
call	The matched call.

Examples

```
# models for post-treatment confounders
m1 <- lm(cesd92 ~ female + race + momedu + parinc + afqt3 +
  educexp + college, data = education)
m2 <- lm(prmarr98 ~ female + race + momedu + parinc + afqt3 +
  educexp + college, data = education)
m3 <- lm(transitions98 ~ female + race + momedu + parinc + afqt3 +
  educexp + college, data = education)

# residual balancing weights
rbwMed_fit <- rbwMed(treatment = college, mediator = ses,
  zmodels = list(m1, m2, m3), baseline_x = female:educexp,
  interact = TRUE, base_weights = weights, data = education)

# attach residual balancing weights to data
education$rbw <- rbwMed_fit$weights

# fit marginal structural model
if(require(survey)){
  rbw_design <- svydesign(ids = ~ 1, weights = ~ rbw, data = education)
  msm_rbw <- svyglm(cesd40 ~ college * ses, design = rbw_design)
  summary(msm_rbw)
}
```

---

rbwPanel   *Residual Balancing Weights for Analyzing Time-varying Treatments*

Description

rbwPanel is a function that produces residual balancing weights (rbw) for estimating the marginal effects of time-varying treatments. The user supplies a long format data frame (each row being a unit-period) and a list of fitted model objects for time-varying confounders. In the current implementation, the residuals of a time-varying covariate X_t are balanced across both current treatment D_t and the regressors of X_t.

Usage

```
rbwPanel(
  exposure,
  xmodels,
  id,
  time,
  data,
  base_weights,
  max_iter = 200,
  print_level = 1,
  tol = 1e-04
)
```

Arguments

exposure	A symbol or character string for the exposure/treatment variable.
xmodels	A list of fitted lm or glm objects for time-varying confounders.
id	A symbol or character string for the unit id variable.
time	A symbol or character string for the time variable.
data	A data frame containing all variables in the model.
base_weights	(Optional) A vector of base weights (or its name).
max_iter	Maximum number of iterations for Newton's method.
print_level	The level of printing:
	1 normal: print whether the algorithm converges or not
	2 detailed: print also the maximum absolute value of the deviation between the moments of the reweighted data and the target moments in each iteration
	3 very detailed: print also the step length of the line searcher in iterations where a full Newton step is excessive.
tol	Tolerance parameter used to determine convergence.

Value

A list containing the results.

weights	A data frame containing id and residual balancing weights.
constraints	A matrix of (linearly independent) residual balancing constraints
eb_out	Results from calling [eb2](#) function
call	The matched call.

Examples

```
# models for time-varying confounders
m1 <- lm(dem.polls ~ (d.gone.neg.l1 + dem.polls.l1 + undother.l1) * factor(week),
data = campaign_long)
m2 <- lm(undother ~ (d.gone.neg.l1 + dem.polls.l1 + undother.l1) * factor(week),
```

```
  data = campaign_long)

xmodels <- list(m1, m2)

# residual balancing weights
rbwPanel_fit <- rbwPanel(exposure = d.gone.neg, xmodels = xmodels, id = id,
time = week, data = campaign_long)

summary(rbwPanel_fit$weights)

# merge weights into wide-format data
campaign_wide2 <- merge(campaign_wide, rbwPanel_fit$weights, by = "id")

# fit a marginal structural model (adjusting for baseline confounders)
if(require(survey)){
  rbw_design <- svydesign(ids = ~ 1, weights = ~ rbw, data = campaign_wide2)
  msm_rbw <- svyglm(demprcnt ~ cum_neg * deminc + camp.length + factor(year) + office,
  design = rbw_design)
  summary(msm_rbw)
}
```

Index