Package ‘RobinCar2’

April 3, 2025

Type Package

Title ROBust INference for Covariate Adjustment in Randomized Clinical Trials

Version 0.1.1

Date 2025-03-30

Description Performs robust estimation and inference when using covariate adjustment and/or covariate-adaptive randomization in randomized controlled trials. This package is trimmed to reduce the dependencies and validated to be used across industry. See ``FDA's final guidance on covariate adjustment''<`https://www.regulations.gov/docket/FDA-2019-D-0934`>, Tsiatis (2008) <doi:10.1002/sim.3113>, Bugni et al. (2018) <doi:10.1080/01621459.2017.1375934>, Ye, Shao, Yi, and Zhao (2023)<doi:10.1080/01621459.2022.2 Ye, Shao, and Yi (2022)<doi:10.1093/biomet/asab015>, Rosenblum and van der Laan (2010)<doi:10.2202/1557-4679.1138>, Wang et al. (2021)<doi:10.1080/01621459.2021.1981338>, Ye, Bannick, Yi, and Shao (2023)<doi:10.1080/24754269.2023.2205802>, and Bannick, Shao, Liu, Du, Yi, and Ye (2024)<doi:10.48550/arXiv.2306.10213>.

License Apache License 2.0

URL `https://github.com/openpharma/RobinCar2/`

BugReports `https://github.com/openpharma/RobinCar2/issues`

Depends R (>= 3.6)

Imports checkmate, numDeriv, MASS, sandwich, stats, utils

Suggests knitr, rmarkdown, testthat (>= 3.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData true

RoxygenNote 7.3.2

NeedsCompilation no

Author Liming Li [aut, cre] (<`https://orcid.org/0009-0008-6870-0878`>),
Marlena Bannick [aut] (<`https://orcid.org/0000-0001-6279-5978`>),
Dong Xi [aut],
Ting Ye [aut],
Yanyao Yi [aut],
Gregory Chen [ctb],
Gilead Sciences, Inc. [cph, fnd],
F. Hoffmann-La Roche AG [cph, fnd],
Merck Sharp & Dohme, Inc. [cph, fnd],
AstraZeneca plc [cph, fnd],
Eli Lilly and Company [cph, fnd],
The University of Washington [cph, fnd]

Maintainer Liming Li <`liming.li1@astrazeneca.com`>

Repository CRAN

Date/Publication 2025-04-03 16:10:02 UTC

Contents

`RobinCar2-package` `RobinCar2` *Package*

Description

`RobinCar2` implements unbiased prediction and robust inference of variance of a fit in R.

Author(s)

Maintainer: Liming Li <`liming.li1@astrazeneca.com`> (ORCID)

Authors:

- Marlena Bannick <`mnorwood@uw.edu`> (ORCID)
- Dong Xi <`Dong.Xi1@gilead.com`>
- Ting Ye <`tingye1@uw.edu`>

- Yanyao Yi `<yi_yanyao@lilly.com>`

Other contributors:

- Gregory Chen `<xiangyi.gregory.chen@msd.com>` [contributor]
- Gilead Sciences, Inc. [copyright holder, funder]
- F. Hoffmann-La Roche AG [copyright holder, funder]
- Merck Sharp & Dohme, Inc. [copyright holder, funder]
- AstraZeneca plc [copyright holder, funder]
- Eli Lilly and Company [copyright holder, funder]
- The University of Washington [copyright holder, funder]

See Also

Useful links:

- `https://github.com/openpharma/RobinCar2/`
- Report bugs at `https://github.com/openpharma/RobinCar2/issues`

`dummy_data` *Dummy Trial Data with Permute-Block Randomization*

Description

This dataset contains the dummy trial data with permute block randomization.

Usage

```
dummy_data
```

Format

A data frame with 600 rows and 7 columns:

id The ID of the patients.

treatment The treatment assignment, "pbo", "trt1" and "trt2"

s1 The first stratification variable, "a" and "b".

s2 The second stratification variable, "c" and "d".

covar The covariate following normal distribution.

y The continuous response.

y_b The binary response.

Source

The data is generated by the create_dummy.R script.

`find_data` *Find Data in a Fit*

Description

Find Data in a Fit

Usage

```
find_data(fit, ...)
```

Arguments

`fit`	A fit object.
`...`	Additional arguments.

Value

A data frame used in the fit.

`h_diff` *Contrast Functions and Jacobians*

Description

Contrast Functions and Jacobians

Create Contrast of Pairs

Usage

```
h_diff(x, y)

h_jac_diff(x, y)

h_ratio(x, y)

h_jac_ratio(x, y)

h_odds_ratio(x, y)

h_jac_odds_ratio(x, y)

eff_jacob(f)

pairwise(levels, x = levels)
```

```
against_ref(levels, ref = levels[1], x = tail(levels, -1))

custom_contrast(levels, x, y)
```

Arguments

`x`	(`vector`) A vector of treatment levels.
`y`	(`vector`) A vector of treatment levels.
`f`	(`function`) Function with argument x and y to compute treatment effect.
`levels`	(`character`) Levels of the treatment.
`ref`	(`string` or `int`) Reference level.

Value

Vector of contrasts, or matrix of jacobians.

A list of `contrast` object with following elements:

- Index of the treatment group.
- Index of the reference group. Additional attributes include `levels` and `max_levels` indicating the names of the treatment levels and the maximum number of levels.

Examples

```
h_diff(1:3, 4:6)
h_jac_ratio(1:3, 4:6)
```

---

predict_counterfactual

Counterfactual Prediction

Description

Obtain counterfactual prediction of a fit.

Usage

```
predict_counterfactual(fit, treatment, data, vcov, vcov_args, ...)
```

Arguments

`fit`	fitted object.
`treatment`	(`formula`) formula of form treatment ~ strata(s).
`data`	(`data.frame`) raw dataset.
`vcov`	(`function` or `character`) variance function or name.
`vcov_args`	(`list`) additional arguments for variance function.
`...`	Additional arguments for methods.

Value

List of class `prediction_cf` containing following elements:

- `estimate`: predicted marginal mean.
- `residual`: residual of the bias-corrected prediction.
- `predictions`: all predicted values.
- `predictions_liner`: linear predictions.
- `schema`: randomization schema.
- `response`: response value.
- `fit`: fitted model.
- `model_matrix`: model matrix.
- `treatment_formula`: treatment assignment and randomization formula.
- `treatment`: treatment value.
- `group_idx`: group index based on the stratification.
- `variance`: estimated variance of the marginal mean.
- `variance_name`: name of the variance.

`robin_glm` *Covariate adjusted glm model*

Description

Covariate adjusted glm model

Usage

```
robin_glm(
  formula,
  data,
  treatment,
  contrast = "difference",
  contrast_jac = NULL,
  vcov = "vcovG",
  family = gaussian(),
  vcov_args = list(),
  pair,
  ...
)
```

Arguments

`formula`	(`formula`) A formula of analysis.
`data`	(`data.frame`) Input data frame.
`treatment`	(`formula` or `character(1)`) A formula of treatment assignment or assignment by stratification, or a string name of treatment assignment.
`contrast`	(`function` or `character(1)`) A function to calculate the treatment effect, or character of "difference", "risk_ratio", "odds_ratio" for default contrasts.
`contrast_jac`	(`function`) A function to calculate the Jacobian of the contrast function. Ignored if using default contrasts.
`vcov`	(`function`) A function to calculate the variance-covariance matrix of the treatment effect, including `vcovHC` and `vcovG`.
`family`	(`family`) A family object of the glm model.
`vcov_args`	(`list`) Additional arguments passed to `vcov`.
`pair`	Pairwise treatment comparison.
`...`	Additional arguments passed to `glm` or `glm.nb`.

Details

If family is `MASS::negative.binomial(NA)`, the function will use `MASS::glm.nb` instead of `glm`.

Value

A treatment_effect object.

Examples

```
robin_glm(
  y ~ treatment * s1,
  data = dummy_data,
  treatment = treatment ~ s1, contrast = "difference"
)
```

`robin_lm` *Covariate adjusted lm model*

Description

Covariate adjusted lm model

Usage

```
robin_lm(
  formula,
  data,
  treatment,
  vcov = "vcovG",
  vcov_args = list(),
  pair,
  ...
)
```

Arguments

`formula`	(`formula`) A formula of analysis.
`data`	(`data.frame`) Input data frame.
`treatment`	(`formula` or `character(1)`) A formula of treatment assignment or assignment by stratification, or a string name of treatment assignment.
`vcov`	(`function`) A function to calculate the variance-covariance matrix of the treatment effect, including `vcovHC` and `vcovG`. The default is 'vcovG'.
`vcov_args`	(`list`) Additional arguments passed to `vcov`.
`pair`	Pairwise treatment comparison.
`...`	Additional arguments passed to `lm`.

Value

A treatment_effect object.

Examples

```
robin_lm(
  y ~ treatment * s1,
  data = dummy_data,
  treatment = treatment ~ s1
)
```

`treatment_effect` *Treatment Effect*

Description

Obtain treatment effect and variance from counter-factual prediction

Usage

```
treatment_effect(
  object,
  pair = pairwise(names(object$estimate)),
  eff_measure,
  eff_jacobian = eff_jacob(eff_measure),
  ...
)

difference(object, ...)

risk_ratio(object, ...)

odds_ratio(object, ...)
```

Arguments

`object`	Object from which to obtain treatment effect.
`pair`	(`contrast`) Contrast choices.
`eff_measure`	(`function`) Treatment effect measurement function.
`eff_jacobian`	(`function`) Treatment effect jacobian function.
`...`	Additional arguments for variance.

Value

A list of `treatment_effect` object with following elements:

- `estimate`: estimate of the treatment effect.
- `pair`: `contrast` object indicating the pairwise treatment effect.
- `contrast`: name of the contrast function.
- `euqal_val`: the value for no treatment effect given the contrast.
- `marginal_mean`: the `prediction_cf` object.
- `fit`: the fitted model.
- `treatment`: the treatment assignment.
- `variance`: the variance of the treatment effect.
- `jacobian`: the Jacobian matrix.

`vcovG` *ANHECOVA Covariance*

Description

ANHECOVA Covariance

Usage

```
vcovG(x, decompose = TRUE, ...)
```

Arguments

`x`	(`prediction_cf`) Counter-factual prediction.
`decompose`	(`flag`) whether to use decompose method to calculate the variance.
`...`	Not used.

Value

Named covariance matrix.

`vcovHC` *Heteroskedasticity-consistent covariance matrix for predictions*

Description

The heteroskedasticity-consistent covariance matrix for predictions is obtained with `sandwich::vocvHC` using sandwich method.

Usage

```
vcovHC(x, type = "HC3", ...)
```

Arguments

`x`	(`prediction_cf`) Counter-factual prediction.
`type`	(`character`) Type of HC covariance matrix.
`...`	Additional arguments for `sandwich::vcovHC`.

Value

Matrix of the heteroskedasticity-consistent covariance for the predictions.

Index