Package: RobinCar2 (via r-universe)

October 19, 2026

Type Package

Title ROBust INference for Covariate Adjustment in Randomized Clinical Trials

Version 0.2.4.9000

Date 2026-10-12

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:10.1080/01621459.2022.2049278>, 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, survival, utils

Suggests knitr, rmarkdown, speff2trial, testthat (>= 3.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

Config/roxygen2/version 8.1.0

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

Date/Publication 2026-10-19 12:53:36 UTC

RemoteUrl https://github.com/openpharma/RobinCar2

RemoteRef HEAD

RemoteSha da87b40f1bbbb54b360a52cf586187323668c626

Contents

RobinCar2-package . . . 2
confint . . . 3
find_data . . . 4
glm_data . . . 4
h_diff . . . 5
predict_counterfactual . . . 6
robin_glm . . . 7
robin_lm . . . 9
robin_surv . . . 10
surv_control . . . 11
surv_data . . . 12
surv_effect_methods . . . 13
treatment_effect . . . 14
vcovG . . . 15
vcovHC . . . 15

Index **16**

`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:

- Liming Li <liming.li1@astrazeneca.com> (ORCID)
- Marlena Bannick <mnorwood@uw.edu> (ORCID)
- Daniel Sabanes Bove <daniel@rconis.com> (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]
- Novartis Pharma AG [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

---

`confint` *Confidence Interval*

Description

Obtain the confidence interval for the marginal mean or the contrast.

Usage

```
## S3 method for class 'prediction_cf'
confint(object, parm, level = 0.95, include_se = FALSE, ...)

## S3 method for class 'surv_effect'
confint(object, parm, level = 0.95, transform, ...)

## S3 method for class 'treatment_effect'
confint(object, parm, level = 0.95, transform, ...)
```

Arguments

`object`	Object to construct confidence interval.
`parm`	(`character` or `integer`) Names of the parameters to construct confidence interval.
`level`	(`numeric`) Confidence level.
`include_se`	(`flag`) Whether to include the standard error as a column in the result matrix.
`...`	Not used.
`transform`	(`function`) Transform function.

Value

A matrix of the confidence interval.

Examples

```
robin_res <- robin_glm(
  y_b ~ treatment * s1,
  data = glm_data, treatment = treatment ~ s1, contrast = "log_risk_ratio"
)
confint(robin_res$marginal_mean, level = 0.7)
confint(robin_res$contrast, parm = 1:3, level = 0.9)
```

---

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.

---

glm_data *Example Trial Data for GLMs with Permute-Block Randomization*

Description

This dataset contains the trial example data for GLMs with permute block randomization.

Usage

```
glm_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_glm_data.R` script.

`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_risk_ratio(x, y)

h_jac_risk_ratio(x, y)

h_odds_ratio(x, y)

h_jac_odds_ratio(x, y)

h_log_risk_ratio(x, y)

h_jac_log_risk_ratio(x, y)

h_log_odds_ratio(x, y)

h_jac_log_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_risk_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 = c("difference", "risk_ratio", "odds_ratio", "log_risk_ratio",
    "log_odds_ratio"),
  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 robin_output object, with marginal_mean and contrast components.

Examples

```
robin_glm(
  y ~ treatment * s1,
  data = glm_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 robin_output object, with marginal_mean and contrast components.

Examples

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

robin_surv *Covariate Adjusted and Stratified Survival Analysis*

Description

Calculate log-rank test as well as hazard ratio estimates for survival data, optionally adjusted for covariates and a stratification factor.

Usage

```
robin_surv(
  formula,
  data,
  treatment,
  comparisons,
  contrast = c("hazardratio", "none"),
  test = "logrank",
  control = surv_control(),
  ...
)
```

Arguments

formula	(formula) A formula of analysis, of the form Surv(time, status) ~ covariates + strata(x, y, z). If no covariates should be adjusted for, use 1 instead on the right hand side. The intercept must not be removed. If no stratification factors should be used for the analysis, do not use strata() in the formula.
data	(data.frame) Input data frame.
treatment	(formula) A formula of treatment assignment or assignment by stratification, of the form treatment ~ scheme(vars). Note that currently the randomization scheme is not used in the analysis. However, any variables that were used in the randomization scheme must be included in the model formula, either as covariates, or as strata().
comparisons	(list) An optional list of comparisons between treatment levels to be performed, see details. By default, all pairwise comparisons are performed automatically.
contrast	(character(1)) The contrast statistic to be used, currently only "hazardratio" is supported. Can be disabled by specifying "none", in which case only the log-rank test is performed.
test	(character(1)) The test to be used, currently only "logrank" is supported.
control	(list) Control parameters for the root-finding algorithm used to estimate the log hazard ratio, usually created with surv_control(). The arguments correspond to tol, maxiter, and trace in stats::uniroot().
...	Additional arguments passed to the survival analysis functions, in particular hr_se_plugin_adjusted (please see here for details).

Details

The user can optionally specify a list of comparisons between treatment levels to be performed. The list must have two elements:

- Treatment level indices of the treatment group.
- Treatment level indices of the control group.

So for example if you would like to compare level 3 with level 1, and also level 3 with level 2 (but not level 2 with level 1) then you can specify: `comparisons = list(c(3, 3), c(1, 2))`

Value

A `surv_effect` object containing the results of the survival analysis.

See Also

surv_effect_methods for S3 methods.

Examples

```
# Adjusted for covariates meal.cal and age and adjusted for stratification by strata:
robin_surv(
  formula = Surv(time, status) ~ meal.cal + age + strata(strata),
  data = surv_data,
  treatment = sex ~ pb(strata)
)

# Adjusted for stratification by strata and ecog but not for covariates:
robin_surv(
  formula = Surv(time, status) ~ 1 + strata(strata, ecog),
  data = surv_data,
  treatment = sex ~ sr(1)
)

# Unadjusted for covariates and stratification:
robin_surv(
  formula = Surv(time, status) ~ 1,
  data = surv_data,
  treatment = sex ~ sr(1)
)
```

`surv_control` *Control Survival Analysis Root Finding*

Description

Control parameters for the root-finding algorithm used by `robin_surv()` to estimate the log hazard ratio.

Usage

```
surv_control(tol = .Machine$double.eps^0.25, maxiter = 1000, trace = 0)
```

Arguments

tol	(number) The desired accuracy, passed to stats::uniroot().
maxiter	(count) The maximum number of iterations, passed to stats::uniroot().
trace	(count) Tracing level, passed to stats::uniroot().

Value

A named list with elements tol, maxiter, and trace.

---

surv_data *Survival Example Data*

Description

This dataset contains survival data from the survival package's survival::lung dataset, modified to include factors for sex and strata, as well as a binary status variable which is 1 for death and 0 for censored.

Usage

```
surv_data
```

Format

An object of class data.frame with 228 rows and 12 columns.

Source

The data is generated by the create_surv_data.R script.

surv_effect_methods *S3 Methods for* surv_effect

Description

S3 Methods for surv_effect

Usage

```
## S3 method for class 'surv_effect'
print(x, ...)

table(x, ...)

## Default S3 method:
table(x, ...)

## S3 method for class 'surv_effect'
table(x, ...)
```

Arguments

x	(surv_effect) the obtained result from robin_surv().
...	ignored additional arguments (for compatibility).

Functions

- print(surv_effect): prints the surv_effect object.
- table(surv_effect): prints and returns invisibly the events table of the surv_effect object.

Examples

```
x <- robin_surv(
  formula = Surv(time, status) ~ meal.cal + age + strata(strata),
  data = surv_data,
  treatment = sex ~ pb(strata)
)
print(x)
table(x)
```

`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),
  contrast_name,
  ...
)

difference(object, ...)

risk_ratio(object, ...)

odds_ratio(object, ...)

log_risk_ratio(object, ...)

log_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.
`contrast_name`	(`string`) Name of the contrast.
`...`	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.

- variance: the variance of the treatment effect.
- jacobian: the Jacobian matrix.
- contrast_mat: contrast summary matrix.

---

vcovG *Generalized Covariance (ANHECOVA)*

Description

Generalized Covariance (ANHECOVA)

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

∗ **dataset**
    glm_data, 4
    surv_data, 12

against_ref *(*h_diff*)*, 5

confint, 3
custom_contrast *(*h_diff*)*, 5

difference *(*treatment_effect*)*, 14

eff_jacob *(*h_diff*)*, 5

find_data, 4

glm_data, 4

h_diff, 5
h_jac_diff *(*h_diff*)*, 5
h_jac_log_odds_ratio *(*h_diff*)*, 5
h_jac_log_risk_ratio *(*h_diff*)*, 5
h_jac_odds_ratio *(*h_diff*)*, 5
h_jac_risk_ratio *(*h_diff*)*, 5
h_log_odds_ratio *(*h_diff*)*, 5
h_log_risk_ratio *(*h_diff*)*, 5
h_odds_ratio *(*h_diff*)*, 5
h_risk_ratio *(*h_diff*)*, 5
here, *10*

log_odds_ratio *(*treatment_effect*)*, 14
log_risk_ratio *(*treatment_effect*)*, 14

odds_ratio *(*treatment_effect*)*, 14

pairwise *(*h_diff*)*, 5
predict_counterfactual, 6
print.surv_effect
        *(*surv_effect_methods*)*, 13

risk_ratio *(*treatment_effect*)*, 14
robin_glm, 7
robin_lm, 9
robin_surv, 10
robin_surv(), *11*, *13*
RobinCar2 *(*RobinCar2-package*)*, 2
RobinCar2-package, 2

stats::uniroot(), *10*, *12*
surv_control, 11
surv_control(), *10*
surv_data, 12
surv_effect_methods, *11*, 13
survival::lung, *12*

table *(*surv_effect_methods*)*, 13
treatment_effect, 14

vcovG, 15
vcovHC, 15