Package ‘drord’

October 13, 2022

Title Doubly-Robust Estimators for Ordinal Outcomes
Version 1.0.1
Description Efficient covariate-adjusted estimators of quantities that are useful for establishing the effects of treatments on ordinal outcomes.
Depends R (>= 3.5.0)
Imports MASS, VGAM, ordinal, ggplot2
Suggests testthat, knitr, rmarkdown, ggsci
License MIT + file LICENSE
URL https://github.com/benkeser/drord
BugReports https://github.com/benkeser/drord/issues
Encoding UTF-8
LazyData true
VignetteBuilder knitr
RoxygenNote 7.1.1
NeedsCompilation no
Author David Benkeser [aut, cre, cph]
Maintainer David Benkeser <benkeser@emory.edu>
Repository CRAN
Date/Publication 2021-05-20 22:30:03 UTC

R topics documented:

  bca_interval ................................................................. 3
  bca_logodds ................................................................. 3
  bca_mannwhitney ............................................................ 4
  bca_marg_dist ............................................................... 6
  bca_wmean ................................................................. 7
  compute_trt_spec_bca_intervals ......................................... 8
R topics documented:

compute_trt_spec_marg_dist_ptwise_ci ................................. 9
compute_trt_spec_marg_dist_simul_ci ................................ 9
covid19 ................................................................. 10
drodr ................................................................. 10
eif_pmf_k .............................................................. 13
eif_theta_k .............................................................. 14
estimate_cdf ............................................................ 14
estimate_ci_logodds ...................................................... 15
estimate_ci_mannwhitney ............................................... 16
estimate_ci_marg_dist .................................................. 17
estimate_ci_wmean ..................................................... 19
estimate_cond_mean .................................................... 20
estimate_eif_wmean ..................................................... 21
estimate_logodds ........................................................ 21
estimate_mannwhitney .................................................. 22
estimate_pmf ............................................................. 22
estimate_treat_prob ..................................................... 24
estimate_wmean ........................................................ 24
evaluate_beta_cov ....................................................... 25
evaluate_mannwhitney_gradient ....................................... 26
evaluate_marg_cdf_eif .................................................. 26
evaluate_marg_cdf_ptwise_ci ......................................... 27
evaluate_marg_dist_simul_ci .......................................... 27
evaluate_marg_pmf_eif .................................................. 28
evaluate_marg_pmf_ptwise_ci ......................................... 29
evaluate_theta_cov ..................................................... 29
evaluate_trt_spec_pmf_eif ............................................. 30
evaluate_trt_spec_theta_eif .......................................... 30
fit_trt_spec_reg .......................................................... 31
getResponseFromFormula ................................................ 33
get_one_logodds ......................................................... 33
get_one_mannwhitney .................................................. 34
get_one_marg_dist ...................................................... 35
get_one_wmean ........................................................ 36
jack_logodds ............................................................ 37
jack_mannwhitney ........................................................ 37
jack_marg_cdf ........................................................... 38
jack_wmean ............................................................... 39
marginalize_cdf ........................................................ 40
marginalize_pmf ........................................................ 40
one_boot_logodds ....................................................... 41
one_boot_mannwhitney ................................................ 42
one_boot_marg_dist .................................................... 43
one_boot_wmean ........................................................ 44
plot.drodr ............................................................... 45
P0plugin ................................................................. 45
predict.POplugin ....................................................... 46
print.drodr .............................................................. 46
**bca_interval**

*Compute a BCa confidence interval*

**Description**

Compute a BCa confidence interval

**Usage**

```r
bca_interval(pt_est, boot_samples, jack_samples, alpha = 0.05)
```

**Arguments**

- `pt_est`: The point estimate of the parameter of interest
- `boot_samples`: A collection of bootstrap realizations of the estimator of the parameter of interest
- `jack_samples`: A vector of jackknife estimates of the parameter of interest.
- `alpha`: Confidence intervals have nominal level $1 - \alpha$.

**Value**

2-length vector containing BCa confidence interval limits.

---

**bca_logodds**

*Compute a BCa bootstrap confidence interval for the weighted mean. The code is based on the slides found here: [http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf](http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf)*

**Description**

Compute a BCa bootstrap confidence interval for the weighted mean. The code is based on the slides found here: [http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf](http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf)
Usage

\texttt{bca\_logodds(}
  \texttt{treat,}
  \texttt{covar,}
  \texttt{out,}
  \texttt{nboot,}
  \texttt{treat\_form,}
  \texttt{out\_levels,}
  \texttt{out\_form,}
  \texttt{out\_model,}
  \texttt{logodds\_est,}
  \texttt{alpha = 0.05}
\texttt{)}

Arguments

treat A numeric vector containing treatment status. Should only assume a value 0 or 1.
covar A data.frame containing the covariates to include in the working proportional odds model.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
nboot Number of bootstrap replicates used to compute bootstrap confidence intervals.
treat\_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out\_levels A numeric vector containing all ordered levels of the outcome.
out\_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out\_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
logodds\_est The estimated log-odds.
alpha Level of confidence interval.

Value

matrix with treatment-specific log-odds CIs and CI for difference.

\texttt{bca\_mannwhitney} \hspace{1cm} \textit{Compute a BCa bootstrap confidence interval for the Mann-Whitney parameter. The code is based on the slides found here: http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf}
Description

Compute a BCa bootstrap confidence interval for the Mann-Whitney parameter. The code is based on the slides found here: http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf

Usage

```r
bcamannwhitney(
  treat,
  covar,
  out,
  nboot,
  treat_form,
  out_levels,
  out_form,
  mannwhitney_est,
  out_model,
  alpha = 0.05
)
```

Arguments

treat A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in `out` is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.

covar A data.frame containing the covariates to include in the working proportional odds model.

out A numeric vector containing the outcomes. Missing outcomes are allowed.

nboot Number of bootstrap replicates used to compute bootstrap confidence intervals.

treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates

out_levels A numeric vector containing all ordered levels of the outcome.

out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.

mannwhitney_est The point estimate of the Mann-Whitney parameter.

out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

alpha Level of confidence interval.

Value

Confidence interval for the Mann-Whitney parameter
Compute a BCa bootstrap confidence interval for the weighted mean. The code is based on the slides found here: http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf

**Usage**

```r
bcas_marg_dist(
  treat,
  covar,
  out,
  nboot,
  treat_form,
  out_levels,
  out_form,
  out_model,
  marg_cdf_est,
  marg_pmf_est,
  alpha = 0.05
)
```

**Arguments**

- `treat`: A numeric vector containing treatment status. Should only assume a value 0 or 1.
- `covar`: A data.frame containing the covariates to include in the working proportional odds model.
- `out`: A numeric vector containing the outcomes. Missing outcomes are allowed.
- `nboot`: Number of bootstrap replicates used to compute bootstrap confidence intervals.
- `treat_form`: The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.
- `out_levels`: A numeric vector containing all ordered levels of the outcome.
- `out_form`: The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
- `out_model`: Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
- `marg_cdf_est`: Point estimate of treatment-specific CDF.
- `marg_pmf_est`: Point estimate of treatment-specific PMF.
- `alpha`: Level of confidence interval.
Value

List (cdf, pmf) of lists (treat=1, treat=0) of confidence intervals for distributions.

Description

Compute a BCa bootstrap confidence interval for the weighted mean. The code is based on the slides found here: http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf

Usage

bca_wmean(
  treat,  # A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
  covar,  # A data.frame containing the covariates to include in the working proportional odds model.
  out,  # A numeric vector containing the outcomes. Missing outcomes are allowed.
  nboot, # Number of bootstrap replicates used to compute bootstrap confidence intervals.
  treat_form,  # The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
  out_levels, # A numeric vector containing all ordered levels of the outcome.
  out_form,  # The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
  out_weights,  # The right-hand side of a regression formula for the working model of outcome weights.
  out_model,  # The right-hand side of a regression formula for the working model of outcome.
  wmean_est,  # The right-hand side of a regression formula for the working model of weighted mean estimate.
  alpha = 0.05  # Compute a BCa bootstrap confidence interval for the weighted mean. The code is based on the slides found here: http://users.stat.umn.edu/~helwig/notes/bootci-Notes.pdf
)
compute_trt_spec_bca_intervals

A vector of numeric weights with length equal to the length of out_levels.

Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

The estimated weighted means + estimated covariance matrix.

Level of confidence interval.

matrix with treatment-specific weighted mean CIs and CI for difference.

compute_trt_spec_bca_intervals

Used to compute treatment-specific BCa intervals for the CDF and PMF

Description

Used to compute treatment-specific BCa intervals for the CDF and PMF

Usage

compute_trt_spec_bca_intervals(
  dist = c("cdf", "pmf"),
  trt = c(1, 0),
  marg_est,
  boot_samples,
  jack_samples,
  alpha
)

Arguments

  dist Which one? CDF or PMF?
  trt Which treatment?
  marg_est The point estimate
  boot_samples A collection of bootstrap realizations of the estimator of the parameter of interest
  jack_samples A vector of jackknife estimates of the parameter of interest.
  alpha Confidence intervals have nominal level 1-alpha.

Value

List of pointwise and simultaneous confidence intervals for dist.
compute_trt_spec_marg_dist_ptwise_ci

Compute simultaneous confidence interval for treatment-specific marginal distribution

Description

Compute simultaneous confidence interval for treatment-specific marginal distribution

Usage

compute_trt_spec_marg_dist_ptwise_ci(pt_est, cov_est, alpha, cdf = TRUE)

Arguments

- pt_est: The point estimate of the treatment-specific marginal CDF/PMF
- cov_est: Covariance matrix estimates.
- alpha: Confidence intervals have nominal level 1-\alpha.
- cdf: Is this for CDF or PMF?

Value

Confidence interval

compute_trt_spec_marg_dist_simul_ci

Compute simultaneous confidence interval for treatment-specific marginal distribution

Description

Compute simultaneous confidence interval for treatment-specific marginal distribution

Usage

compute_trt_spec_marg_dist_simul_ci(
    pt_est,
    trt_spec_marg_dist_eif,
    remove_last = TRUE,
    alpha
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pt_est</td>
<td>The point estimate of the treatment-specific marginal CDF/PMF</td>
</tr>
<tr>
<td>trt_spec_marg_dist_eif</td>
<td>The EIF estimates for the treatment-specific marginal CDF/PMF estimates</td>
</tr>
<tr>
<td>remove_last</td>
<td>Should the last level be removed? Should be set equal to TRUE for CDF computations and FALSE for PMF computations.</td>
</tr>
<tr>
<td>alpha</td>
<td>Confidence intervals have nominal level 1-alpha.</td>
</tr>
</tbody>
</table>

Value

Confidence interval

---

covid19

*Simulated COVID-19 outcomes for hospitalized patients.*

Description

A simulated dataset containing outcomes, (hypothetical) treatment, and age group

Usage

covid19

Format

A data frame with 500 rows and 3 variables:

- **out**: study outcome, here 1 represents death, 2 intubation, 3 no adverse outcome
- **age_grp**: age category with 1 the youngest and 7 the oldest
- **treat**: hypothetical treatment, here 1 represents an (effective) active treatment and 0 a control

---

drord

*Doubly robust estimates of for evaluating effects of treatments on ordinal outcomes.*

Description

The available parameters for evaluating treatment efficacy are:

- Difference in (weighted) means: The outcome levels are treated numerically, with each level possibly assigned a weight. The difference in average outcomes is computed.
- Log odds ratio: The comparison describes the average log-odds (treatment level 1 versus 0) of the cumulative probability for each level of the outcome.
- Mann-Whitney: The probability that a randomly-selected individual receiving treatment 1 will have a larger outcome value than a randomly selected individual receiving treatment 0 (with ties assigned weight 1/2).
drord

Usage

dror(d
  out,
  treat,
  covar,
  out_levels = sort(unique(out)),
  out_form = paste0(colnames(covar), collapse = "+"),
  out_weights = rep(1, length(out_levels)),
  out_model = "pooled-logistic",
  treat_form = "1",
  param = c("weighted_mean", "log_odds", "mann_whitney"),
  ci = "wald",
  alpha = 0.05,
  nboot = 1000,
  return_models = TRUE,
  est_dist = TRUE,
  stratify = FALSE,
  ...
)

Arguments

out A numeric vector containing the outcomes. Missing outcomes are allowed.
treat A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
covar A data.frame containing the covariates to include in the working proportional odds model.
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_weights A vector of numeric weights with length equal to the length of out_levels.
out_model Which R function should be used to fit the proportional odds model. The recommended option is "pooled-logistic". Other options available include "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
param A vector of characters indicating which of the three treatment effect parameters should be estimated ("weighted_mean", "log_odds", and/or "mann_whitney").
ci A vector of characters indicating which confidence intervals should be computed ("bca" and/or "wald")
alpha Confidence intervals have nominal level 1-alpha.
nboot  Number of bootstrap replicates used to compute bootstrap confidence intervals.
return_models  If TRUE the fitted working proportional odds models and treatment probability models are returned.
est_dist  A boolean indicating whether estimates of the CDF and PMF should be computed and returned. For real data analysis, we generally recommend leaving as TRUE; however, when studying performance in simulations, it can save time to set to FALSE.
stratify  If TRUE, then a fully stratified estimator is computed, i.e., the empirical CDF of each treatment arm is estimated stratifying by levels of covar. For now, this option is limited to univariate covariates.
...  Other options (not currently used).

Details

In each case, estimates are constructed by obtaining a doubly robust estimate of the cumulative distribution function (CDF) for each treatment group. This is achieved by fitting a (working) proportional odds model that includes inverse probability of treatment weights. The inclusion of these weights ensures that, so long as the working model includes intercept terms, the resultant estimate of the CDF is an augmented inverse probability of treatment weighted estimate. This implies that the estimate is nonparametric efficient if the working model contains the truth; however, even if the working model does not contain the truth, the CDF estimates are consistent and asymptotically normal with variance expected to dominate that of an unadjusted estimate of the same treatment effect.

The CDF estimates are subsequently mapped into estimates of each requested parameter for evaluating treatment effects. The double robustness and efficiency properties of the CDF estimates extend to these quantities as well. Confidence intervals and hypothesis tests can be carried out in closed form using Wald-style intervals and tests or using a nonparametric corrected and accelerated bootstrap (BCa). Inference for the CDF and probability mass function is also returned and can be used for subsequent visualizations (see plot.drord).

Value

An object of class drord. In addition to information related to how drord was called, the output contains the following:

log_odds  inference pertaining to the log-odds parameter. NULL if this parameter not requested in call to drord.
mann_whitney  inference pertaining to the Mann-Whitney parameter. NULL if this parameter not requested in call to drord.
weighted_mean  inference pertaining to weighted mean parameter. NULL if this parameter not requested in call to drord.
cdf  inference pertaining to the treatment-specific CDFs. See the plot method for a convenient way of visualizing this information. NULL if est_dist = FALSE in call to drord.
pmf  inference pertaining to the treatment-specific PMFs. See the plot method for a convenient way of visualizing this information. NULL if est_dist = FALSE in call to drord.
treat_mod the fitted model for the probability of treatment as a function of covariates. NULL if return_models = FALSE
**eif_pmf_k**

Get EIF estimates for treatment-specific PMF at a particular level of the outcome

**Description**

Get EIF estimates for treatment-specific PMF at a particular level of the outcome

**Usage**

```
eif_pmf_k(k, out, treat, trt_level, trt_spec_prob_est, trt_k_spec_pmf_est)
```

**Arguments**

- **k**  The level of the outcome.
- **out**  A numeric vector containing the outcomes. Missing outcomes are allowed.
\textbf{eif Theta k} \hspace{1cm} \textit{Get EIF estimates for treatment-specific CDF at a particular level of the outcome}

**Description**

Get EIF estimates for treatment-specific CDF at a particular level of the outcome

**Usage**

\begin{verbatim}
eif_theta_k(k, out, treat, trt_level, trt_spec_prob_est, trt_k_spec_cdf_est)
\end{verbatim}

**Arguments**

- \textit{k} \hspace{1cm} \text{The level of the outcome.}
- \textit{out} \hspace{1cm} \text{A numeric vector containing the outcomes. Missing outcomes are allowed.}
- \textit{treat} \hspace{1cm} \text{A numeric vector containing treatment status. Should only assume a value 0 or 1.}
- \textit{trt_level} \hspace{1cm} \text{Treatment level}
- \textit{trt_spec_prob_est} \hspace{1cm} \text{Estimated propensity for \textit{trt_level}.}
- \textit{trt_k_spec_cdf_est} \hspace{1cm} \text{Estimated conditional CDF for \textit{trt_level} at \textit{k}.}

\textbf{estimate_cdf} \hspace{1cm} \textit{Map an estimate of the conditional PMF into an estimate of the conditional CDF}

**Description**

Map an estimate of the conditional PMF into an estimate of the conditional CDF

**Usage**

\begin{verbatim}
estimate_cdf(pmf_est)
\end{verbatim}
**estimate_ci_logodds**

**Arguments**

- `pmf_est` A list of the treatment-specific PMF estimates

**Value**

A list of treatment-specific CDF estimates

---

**estimate_ci_logodds**  
*Compute confidence interval/s for the log-odds parameters*

**Description**

Compute confidence interval/s for the log-odds parameters

**Usage**

```
estimate_ci_logodds(
  logodds_est,
  cdf_est,
  out_form,
  covar,
  treat_prob_est,
  treat,
  treat_form,
  out,
  ci,
  alpha = 0.05,
  nboot,
  out_levels,
  out_model,
  ...
)
```

**Arguments**

- `logodds_est` The point estimates for log-odds.
- `cdf_est` A list of treatment-specific CDF estimates.
- `out_form` The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
- `covar` A data.frame containing the covariates to include in the working proportional odds model.
- `treat_prob_est` Estimated probability of treatments, output from call to `estimate_treat_prob`.
- `treat` A numeric vector containing treatment status. Should only assume a value 0 or 1.
The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.

A numeric vector containing the outcomes. Missing outcomes are allowed.

A vector of characters indicating which confidence intervals should be computed ("bca" and/or "wald")

Confidence intervals have nominal level 1-alpha.

Number of bootstrap replicates used to compute bootstrap confidence intervals.

A numeric vector containing all ordered levels of the outcome.

Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Other options (not currently used).

List with wald and bca-estimated confidence intervals for the weighted mean parameters.

Compute confidence interval/s for the Mann-Whitney parameter

Compute confidence interval/s for the Mann-Whitney parameter

Usage

```r
estimate_ci_mannwhitney(
  mannwhitney_est,
  cdf_est,
  pmf_est,
  treat_prob_est,
  treat_form,
  out_form,
  treat,
  ci,
  out,
  alpha,
  nboot,
  out_levels,
  covar,
  out_model
)
```
**Arguments**

- `mannwhitney_est`: The point estimates for log-odds.
- `cdf_est`: The estimated conditional CDF.
- `pmf_est`: The estimated conditional PMF.
- `treat_prob_est`: Estimated probability of treatments, output from call to `estimate_treat_prob`.
- `treat_form`: The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.
- `out_form`: The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
- `treat`: A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in `out` is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
- `ci`: A vector of characters indicating which confidence intervals should be computed ("bca" and/or "wald").
- `out`: A numeric vector containing the outcomes. Missing outcomes are allowed.
- `alpha`: Confidence intervals have nominal level 1-\alpha.
- `nboot`: Number of bootstrap replicates used to compute bootstrap confidence intervals.
- `out_levels`: A numeric vector containing all ordered levels of the outcome.
- `covar`: A `data.frame` containing the covariates to include in the working proportional odds model.
- `out_model`: Which R function should be used to fit the proportional odds model. Options are "polr" (from the `MASS` package), "vglm" (from the `VGAM` package), or "clm" (from the `ordinal` package).

**Value**

List with wald and bca-estimated confidence intervals for the Mann-Whitney parameter.

**Description**

Compute confidence interval/s for the treatment specific PMF and CDF.
Usage

estimate_ci_marg_dist(
    marg_cdf_est,
    marg_pmf_est,
    cdf_est,
    pmf_est,
    covar,
    treat_prob_est,
    treat_form,
    out_form,
    treat,
    ci,
    out_levels,
    out_model,
    out,
    alpha,
    nboot
)

Arguments

marg_cdf_est  Point estimate of treatment-specific CDF.
marg_pmf_est  Point estimate of treatment-specific PMF.
cdf_est       Estimates of treatment-specific conditional CDF.
pmf_est       Estimates of treatment-specific conditional PMF.
covar         A data.frame containing the covariates to include in the working proportional odds model.
treat_prob_est Estimated probability of treatments, output from call to estimate_treat_prob.
treat_form    The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out_form      The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
treat         A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
ci            A vector of characters indicating which confidence intervals should be computed ("bca" and/or "wald")
out_levels    A numeric vector containing all ordered levels of the outcome.
out_model     Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
out           A numeric vector containing the outcomes. Missing outcomes are allowed.
alpha         Confidence intervals have nominal level 1-alpha.
nboot         Number of bootstrap replicates used to compute bootstrap confidence intervals.
estimate_ci_wmean

Value

List of lists (cdf and pmf) with wald and bca-estimated confidence intervals for the marginal treatment-specific distribution functions.

estimate_ci_wmean  Compute confidence interval/s for the weight mean parameters

Description

Compute confidence interval/s for the weight mean parameters

Usage

estimate_ci_wmean(
  out, treat, covar, wmean_est, alpha = 0.05, out_levels = order(unique(out)), out_form = NULL, out_weights = rep(1, length(out_levels)), out_model, treat_form = "1", ci = c("bca", "wald"), nboot = 10000
)

Arguments

out A numeric vector containing the outcomes. Missing outcomes are allowed.
treat A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
covar A data.frame containing the covariates to include in the working proportional odds model.
wmean_est The point estimates for weighted means
alpha Confidence intervals have nominal level 1-alpha.
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_weights A vector of numeric weights with length equal to the length of out_levels.
out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates

ci A vector of characters indicating which confidence intervals should be computed ("bca" and/or "wald")

nboot Number of bootstrap replicates used to compute bootstrap confidence intervals.

Value

List with wald and bca-estimated confidence intervals for the weighted mean parameters.

estimate_cond_mean Map an estimate of treatment-specific PMF into an estimate of treatment specific conditional mean for each observation.

Description

Map an estimate of treatment-specific PMF into an estimate of treatment specific conditional mean for each observation.

Usage

estimate_cond_mean(trt_spec_pmf_est, ordered_out_levels, ordered_out_weights)

Arguments

trt_spec_pmf_est
  The treatment-specific PMF estimates

ordered_out_levels
  Self explanatory

ordered_out_weights
  Self explanatory

Value

Vector of estimated conditional means
Estimate EIF WMean

Obtain an estimate of the efficient influence function for the treatment-specific weighted mean parameter.

**Usage**

```r
estimate_eif_wmean(
  trt_spec_cond_mean_est,
  trt_spec_prob_est,
  trt_level,
  out,
  treat
)
```

**Arguments**

- `trt_spec_cond_mean_est`: Conditional mean for `trt_level`.
- `trt_spec_prob_est`: Propensity for `trt_level`.
- `trt_level`: Treatment level.
- `out`: A numeric vector containing the outcomes. Missing outcomes are allowed.
- `treat`: A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in `out` is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.

Estimate Log Odds

Implements a plug-in estimator of equation (2) in Diaz et al.

**Usage**

```r
estimate_logodds(cdf_est)
```
estimate_pmf

Arguments

cdf_est A list of treatment-specific CDF estimates

Value

Log odds of treatment = 1, = 0, and the difference.

estimate_mannwhitney Compute the estimate of Mann-Whitney based on conditional CDF and PMF

Description

Compute the estimate of Mann-Whitney based on conditional CDF and PMF

Usage

estimate_mannwhitney(cdf_est, pmf_est)

Arguments

cdf_est Conditional CDF estimates
pmf_est Conditional PMF estimates

Value

Mann-Whitney point estimate

estimate_pmf Get a treatment-specific estimate of the conditional PMF. Essentially this is a wrapper function for fit_trt_spec_reg, which fits the proportion odds model in a given treatment arm.

Description

Get a treatment-specific estimate of the conditional PMF. Essentially this is a wrapper function for fit_trt_spec_reg, which fits the proportion odds model in a given treatment arm.
estimate_pmf

Usage

estimate_pmf(
  out, treat, covar, out_levels, out_form = NULL, out_model, treat_prob_est, stratify = FALSE, return_models = TRUE,
  ...
)

Arguments

  out       A numeric vector containing the outcomes. Missing outcomes are allowed.
  treat     A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
  covar     A data.frame containing the covariates to include in the working proportional odds model.
  out_levels A numeric vector containing all ordered levels of the outcome.
  out_form  The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
  out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
  treat_prob_est Estimated probability of treatments, output from call to estimate_treat_prob.
  stratify   Boolean indicating whether to use nonparametric maximum likelihood (i.e., a stratified estimator). If out_form = "1", then a covariate-unadjusted estimate is computed.
  return_models If TRUE the fitted working proportional odds models and treatment probability models are returned.
  ...        Other options (not used).

Value

A list with fm the fitted model for treatment 1 and 0 (or, if !return_models then NULL) and pmf the estimated PMF under treatment 1 and 0 evaluated on each observation.
estimate_treat_prob  
*Estimate probability of receiving each level of treatment*

**Description**

Estimate probability of receiving each level of treatment

**Usage**

```r
estimate_treat_prob(treat, covar, treat_form, return_models)
```

**Arguments**

- `treat`: A numeric vector containing treatment status. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing.
- `covar`: A data.frame containing the covariates to include in the working proportional odds model.
- `treat_form`: The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.
- `return_models`: If TRUE the fitted working proportional odds models and treatment probability models are returned.

**Value**

A list where the first element is estimate of Pr(treat = 1 | covar) for covar equal to inputted values of covar and second element is estimate of Pr(treat = 0 | covar) for covar equal to inputted values of covar.

---

estimate_wmean  
*Compute the estimate of the weighted mean parameter based on estimated PMF in each treatment arm.*

**Description**

Compute the estimate of the weighted mean parameter based on estimated PMF in each treatment arm.
Usage

\[
\text{evaluate}_\text{beta}\_\text{cov}(\\text{cdf}_\text{est}, \ \text{theta}_\text{cov})
\]

Arguments

- \text{cdf}_\text{est} \quad \text{Estimated CDFs}
- \text{theta}_\text{cov} \quad \text{Covariance matrix for CDF estimates}

Description

Get the covariance matrix for beta
evaluate_mannwhitney_gradient  
*Compute the estimated gradient of the Mann-Whitney parameter. Needed to derive standard error for Wald confidence intervals.*

**Description**  
Compute the estimated gradient of the Mann-Whitney parameter. Needed to derive standard error for Wald confidence intervals.

**Usage**  
`evaluate_mannwhitney_gradient(cdf_est, pmf_est)`

**Arguments**  
- `cdf_est`: Conditional CDF estimates  
- `pmf_est`: Conditional PMF estimates

**Value**  
3-length vector for delta method calculus

evaluate_marg_cdf_eif  
*Get eif estimates for treatment-specific CDF*

**Description**  
Get eif estimates for treatment-specific CDF

**Usage**  
`evaluate_marg_cdf_eif(cdf_est, treat_prob_est, treat, out, out_levels)`

**Arguments**  
- `cdf_est`: Estimated conditional CDF for `trt_level`.  
- `treat_prob_est`: Estimated propensity for `trt_level`.  
- `treat`: A numeric vector containing treatment status. Should only assume a value 0 or 1.  
- `out`: A numeric vector containing the outcomes. Missing outcomes are allowed.  
- `out_levels`: A numeric vector containing all ordered levels of the outcome.
**evaluate_marg_cdf_ptwise_ci**

*Evaluate pointwise confidence interval for marginal CDF.*

**Description**

Evaluate pointwise confidence interval for marginal CDF.

**Usage**

```r
evaluate_marg_cdf_ptwise_ci(marg_cdf_est, marg_cdf_eif, alpha)
```

**Arguments**

- `marg_cdf_est`: The point estimate of the marginal CDF distribution
- `marg_cdf_eif`: The EIF estimates for the marginal CDF estimates
- `alpha`: Confidence intervals have nominal level 1-alpha.

**Value**

List by treatment of simultaneous confidence intervals

**evaluate_marg_dist_simul_ci**

*Evaluate simultaneous confidence interval for marginal PMF or CDF.*

**Description**

Evaluate simultaneous confidence interval for marginal PMF or CDF.

**Usage**

```r
evaluate_marg_dist_simul_ci(
    marg_dist_est, 
    marg_dist_eif, 
    alpha, 
    remove_last = FALSE
)
```
evaluate_marg_pmf_eif

Arguments

marg_dist_est The point estimate of the marginal CDF/PMF distribution
marg_dist_eif The EIF estimates for the marginal CDF/PMF estimates
alpha Confidence intervals have nominal level 1-alpha.
remove_last Should the last level be removed? Should be set equal to TRUE for CDF computations and FALSE for PMF computations.

Value

List by treatment of simultaneous confidence intervals

evaluate_marg_pmf_eif  Get eif estimates for treatment-specific PMF

Description

Get eif estimates for treatment-specific PMF

Usage

evaluate_marg_pmf_eif(pmf_est, treat_prob_est, treat, out, out_levels)

Arguments

pmf_est Estimated conditional PMF for trt_level.
treat_prob_est Estimated propensity for trt_level.
treat A numeric vector containing treatment status. Should only assume a value 0 or 1.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
out_levels A numeric vector containing all ordered levels of the outcome.

Value

a list of eif estimates
**evaluate_marg_pmf_ptwise_ci**

Evaluate pointwise confidence interval for marginal PMF.

**Description**

Evaluate pointwise confidence interval for marginal PMF.

**Usage**

```r
evaluate_marg_pmf_ptwise_ci(marg_pmf_est, marg_pmf_eif, alpha)
``` 

**Arguments**

- `marg_pmf_est`: The point estimate of the marginal PMF distribution
- `marg_pmf_eif`: The EIF estimates for the marginal PMF estimates
- `alpha`: Confidence intervals have nominal level 1-\( \alpha \).

**Value**

List by treatment of simultaneous confidence intervals

**evaluate_theta_cov**

get a covariance matrix for the estimated CDF

**Description**

get a covariance matrix for the estimated CDF

**Usage**

```r
evaluate_theta_cov(cdf_est, treat_prob_est, treat, out, out_levels)
``` 

**Arguments**

- `cdf_est`: The estimates of the treatment-specific CDFs
- `treat_prob_est`: List of estimated probability of treatments, output from call to `estimate_treat_prob`.
- `treat`: A numeric vector containing treatment status. Should only assume a value 0 or 1.
- `out`: A numeric vector containing the outcomes. Missing outcomes are allowed.
- `out_levels`: A numeric vector containing all ordered levels of the outcome.

**Value**

Estimated covariance matrix for CDF estimates
evaluate_trt_spec_pmf_eif

Get a matrix of eif estimates for treatment-specific PMF

Description

Get a matrix of eif estimates for treatment-specific PMF

Usage

evaluate_trt_spec_pmf_eif(
  trt_spec_pmf_est,
  trt_spec_prob_est,
  trt_level,
  treat,
  out,
  out_levels
)

Arguments

trt_spec_pmf_est
  Estimated conditional PMF for trt_level.
trt_spec_prob_est
  Estimated propensity for trt_level.
trt_level
  Treatment level

treat
  A numeric vector containing treatment status. Should only assume a value 0 or 1.

out
  A numeric vector containing the outcomes. Missing outcomes are allowed.

out_levels
  A numeric vector containing all ordered levels of the outcome.

Value

a matrix of EIF estimates

evaluate_trt_spec_theta_eif

generate EIF estimates for the treatment-specific CDF estimates

Description

get a matrix of eif estimates for the treatment-specific CDF estimates
Usage
evaluate_trt_spec_theta_eif(
    trt_spec_cdf_est,
    trt_spec_prob_est,
    trt_level,
    treat,
    out,
    out_levels
)

Arguments
trt_spec_cdf_est
    Estimated conditional CDF for trt_level.
trt_spec_prob_est
    Estimated propensity for trt_level.
trt_level
    Treatment level
treat
    A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
out
    A numeric vector containing the outcomes. Missing outcomes are allowed.
out_levels
    A numeric vector containing all ordered levels of the outcome.

Value
matrix of EIF estimates for CDF.

fit_trt_spec_reg

Helper function to fit a treatment specific outcome regression. If there are more than 2 observed levels of the outcome for the specified treatment arm, then polr is used from the MASS package. Otherwise logistic regression is used. In both cases, inverse probability of treatment weights are included in the regression. If there are levels of the outcome that are not observed in this treatment group, then 0’s are added in. The function returns a matrix with named columns corresponding to each outcome (ordered numerically). The entries represent the estimated covariate-conditional treatment-specific PMF.

Description
Helper function to fit a treatment specific outcome regression. If there are more than 2 observed levels of the outcome for the specified treatment arm, then polr is used from the MASS package. Otherwise logistic regression is used. In both cases, inverse probability of treatment weights are
included in the regression. If there are levels of the outcome that are not observed in this treatment group, then 0’s are added in. The function returns a matrix with named columns corresponding to each outcome (ordered numerically). The entries represent the estimated covariate-conditional treatment-specific PMF.

Usage

```r
fit_trt_spec_reg(
    trt_level, 
    trt_spec_prob_est, 
    out, 
    treat, 
    covar, 
    out_levels, 
    out_form = NULL, 
    out_model, 
    stratify, 
    ...
)
```

Arguments

- **trt_level**: Which level of treatment to fit the proportional odds model for
- **trt_spec_prob_est**: A vector of estimates of Pr(treat = trt_level | covar).
- **out**: A numeric vector containing the outcomes. Missing outcomes are allowed.
- **treat**: A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in `out` is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
- **covar**: A `data.frame` containing the covariates to include in the working proportional odds model.
- **out_levels**: A numeric vector containing all ordered levels of the outcome.
- **out_form**: The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
- **out_model**: Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
- **stratify**: Boolean indicating whether to use nonparametric maximum likelihood (i.e., a stratified estimator). If `out_form = "1"`, then a covariate-unadjusted estimate is computed.
- **...**: Other options (not used).
getResponseFromFormula

*Get a response from model formula*

**Description**

Get a response from model formula

**Usage**

getResponseFromFormula(formula, data)

**Arguments**

- **formula**: The model formula
- **data**: The data frame associated with the model

get_one_logodds

*Compute one log odds based on a given data set.*

**Description**

Compute one log odds based on a given data set.

**Usage**

get_one_logodds(treat, covar, treat_form, out_model, out, out_levels, out_form)

**Arguments**

- **treat**: A numeric vector containing treatment status. Should only assume a value 0 or 1.
- **covar**: A data frame containing the covariates to include in the working proportional odds model.
- **treat_form**: The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
- **out_model**: Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
- **out**: A numeric vector containing the outcomes. Missing outcomes are allowed.
- **out_levels**: A numeric vector containing all ordered levels of the outcome.
- **out_form**: The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
get_one_mannwhitney

Description

Compute one estimate of Mann-Whitney parameter based on a given data set.

Usage

get_one_mannwhitney(treat, covar, treat_form, out, out_levels, out_form, out_model)

Arguments

treat A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.

covar A data.frame containing the covariates to include in the working proportional odds model.
treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value

Estimate of Mann-Whitney parameter for these input data.
get_one_marg_dist  

*Compute one estimate of the marginal CDF/PMF on a given data set.*

**Description**

Compute one estimate of the marginal CDF/PMF on a given data set.

**Usage**

```r
get_one_marg_dist(
  treat,  # A numeric vector containing treatment status. Should only assume a value 0 or 1.
  covar,  # A data.frame containing the covariates to include in the working proportional odds model.
  treat_form,  # The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
  out_model,  # Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
  out,  # A numeric vector containing the outcomes. Missing outcomes are allowed.
  out_levels,  # A numeric vector containing all ordered levels of the outcome.
  out_form  # The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
)
```

**Arguments**

- `treat`: A numeric vector containing treatment status. Should only assume a value 0 or 1.
- `covar`: A data.frame containing the covariates to include in the working proportional odds model.
- `treat_form`: The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.
- `out_model`: Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
- `out`: A numeric vector containing the outcomes. Missing outcomes are allowed.
- `out_levels`: A numeric vector containing all ordered levels of the outcome.
- `out_form`: The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.

**Value**

List of estimated cdf/pmf for these input data.
get_one_wmean

Description

Compute one weighted mean based on a given data set.

Usage

get_one_wmean(
  treat,  # A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
  covar,  # A data.frame containing the covariates to include in the working proportional odds model.
  treat_form,  # The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
  out,  # A numeric vector containing the outcomes. Missing outcomes are allowed.
  out_levels,  # A numeric vector containing all ordered levels of the outcome.
  out_form,  # The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
  out_model,  # Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
  out_weights  # A vector of numeric weights with length equal to the length of out_levels.
)

Arguments

treat  # A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
covar  # A data.frame containing the covariates to include in the working proportional odds model.
treat_form  # The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out  # A numeric vector containing the outcomes. Missing outcomes are allowed.
out_levels  # A numeric vector containing all ordered levels of the outcome.
out_form  # The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_model  # Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
out_weights  # A vector of numeric weights with length equal to the length of out_levels.
**jack_logodds**

*Compute jackknife log-odds estimates.*

---

**Description**

Compute jackknife log-odds estimates.

**Usage**

```r
jack_logodds(treat, covar, out, treat_form, out_model, out_levels, out_form)
```

**Arguments**

- `treat` A numeric vector containing treatment status. Should only assume a value 0 or 1.
- `covar` A data.frame containing the covariates to include in the working proportional odds model.
- `out` A numeric vector containing the outcomes. Missing outcomes are allowed.
- `treat_form` The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
- `out_model` Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).
- `out_levels` A numeric vector containing all ordered levels of the outcome.
- `out_form` The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.

**Value**

Jackknife estimated log-odds

---

**jack_mannwhitney**

*Compute Mann-Whitney log-odds estimates.*

---

**Description**

Compute Mann-Whitney log-odds estimates.
Usage

jack_mannwhitney(
  treat,  
  covar,  
  out,  
  treat_form,  
  out_levels,  
  out_form,  
  out_model
)

Arguments

treat  A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
covar  A data.frame containing the covariates to include in the working proportional odds model.
out  A numeric vector containing the outcomes. Missing outcomes are allowed.
treat_form  The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out_levels  A numeric vector containing all ordered levels of the outcome.
out_form  The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_model  Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value

Jackknife estimate of Mann-Whitney parameter

---

jack_marg_cdf  Compute jackknife distribution estimates.

Description

Compute jackknife distribution estimates.

Usage

jack_marg_cdf(treat, covar, out, treat_form, out_levels, out_form, out_model)
Arguments

treat
A numeric vector containing treatment status. Should only assume a value 0 or 1.

covar
A data.frame containing the covariates to include in the working proportional odds model.

out
A numeric vector containing the outcomes. Missing outcomes are allowed.

treat_form
The right-hand side of a regression formula for the working model of treatment probability as a function of covariates.

out_levels
A numeric vector containing all ordered levels of the outcome.

out_form
The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.

out_model
Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value

Jackknife estimated distributions

---

jack_wmean

Compute jackknife weighted mean estimates.

Description

Compute jackknife weighted mean estimates.

Usage

jack_wmean(
  treat,
  covar,
  out,
  treat_form,
  out_levels,
  out_form,
  out_weights,
  out_model
)

Arguments

treat
A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
covar A data.frame containing the covariates to include in the working proportional odds model.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_weights A vector of numeric weights with length equal to the length of out_levels.
out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value
Jackknife-estimated weighted mean

marginalize_cdf Marginalize over empirical distribution to obtain marginal treatment-specific CDF estimate.

Description
Marginalize over empirical distribution to obtain marginal treatment-specific CDF estimate.

Usage
marginalize_cdf(cdf_est)

Arguments
cdf_est Estimates of treatment-specific conditional CDF.

marginalize_pmf Marginalize over empirical distribution to obtain marginal treatment-specific PMF estimate.

Description
Marginalize over empirical distribution to obtain marginal treatment-specific PMF estimate.

Usage
marginalize_pmf(pmf_est)
one_boot_logodds

Arguments

pmf_est Estimates of treatment-specific conditional PMF.

Description

Get one bootstrap computation of the log odds parameters.

Usage

one_boot_logodds(
  treat,
  covar,
  out,
  treat_form,
  out_levels,
  out_form,
  out_model
)

Arguments

treat A numeric vector containing treatment status. Should only assume a value 0 or 1.
covar A data.frame containing the covariates to include in the working proportional odds model.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value

Estimates of log odds for a particular bootstrap sample.
one_boot_mannwhitney  

Get one bootstrap computation of the Mann-Whitney parameter.

Description

Get one bootstrap computation of the Mann-Whitney parameter.

Usage

```r
one_boot_mannwhitney(
  treat,
  covar,
  out,
  treat_form,
  out_levels,
  out_form,
  out_model
)
```

Arguments

- **treat**  
  A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in `out` is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.

- **covar**  
  A `data.frame` containing the covariates to include in the working proportional odds model.

- **out**  
  A numeric vector containing the outcomes. Missing outcomes are allowed.

- **treat_form**  
  The right-hand side of a regression formula for the working model of treatment probability as a function of covariates

- **out_levels**  
  A numeric vector containing all ordered levels of the outcome.

- **out_form**  
  The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.

- **out_model**  
  Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value

Estimates of Mann-Whitney parameter for a particular bootstrap sample.
one_boot_marg_dist

Get one bootstrap computation of the CDF and PMF estimates

Description

Get one bootstrap computation of the CDF and PMF estimates

Usage

one_boot_marg_dist(treat, covar, out, treat_form, out_levels, out_form, out_model)

Arguments

treat A numeric vector containing treatment status. Should only assume a value 0 or 1.
covar A data.frame containing the covariates to include in the working proportional odds model.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value

Estimates of CDF and PMF for a particular bootstrap sample.
one_boot_wmean

Get one bootstrap computation of the weighted mean parameters.

Description
Get one bootstrap computation of the weighted mean parameters.

Usage

one_boot_wmean(
  treat,
  covar,
  out,
  treat_form,
  out_levels,
  out_form,
  out_weights,
  out_model
)

Arguments

treat A numeric vector containing treatment status. Missing values are not allowed unless the corresponding entry in out is also missing. Only values of 0 or 1 are treated as actual treatment levels. Any other value is assumed to encode a value for which the outcome is missing and the corresponding outcome value is ignored.
covar A data.frame containing the covariates to include in the working proportional odds model.
out A numeric vector containing the outcomes. Missing outcomes are allowed.
treat_form The right-hand side of a regression formula for the working model of treatment probability as a function of covariates
out_levels A numeric vector containing all ordered levels of the outcome.
out_form The right-hand side of a regression formula for the working proportional odds model. NOTE: THIS FORMULA MUST NOT SUPPRESS THE INTERCEPT.
out_weights A vector of numeric weights with length equal to the length of out_levels.
out_model Which R function should be used to fit the proportional odds model. Options are "polr" (from the MASS package), "vglm" (from the VGAM package), or "clm" (from the ordinal package).

Value
Estimates of weighted mean for a particular bootstrap sample.
plot.drord \hspace{1cm} \textit{Print the output of a "drord" object.}

\underline{Description}
Print the output of a "drord" object.

\underline{Usage}
\begin{verbatim}
## S3 method for class 'drord'
plot(
  x,
  treat_labels = c(1, 0),
  dist = "pmf",
  out_labels = if (dist == "pmf") { x$out_levels } else {
    x$out_levels[-length(x$out_levels)] },
...
)
\end{verbatim}

\underline{Arguments}
\begin{itemize}
  \item \texttt{x} \hspace{1cm} A "drord" object.
  \item \texttt{treat_labels} \hspace{1cm} Labels for the treatment variables (treat = 1 followed by treat = 0).
  \item \texttt{dist} \hspace{1cm} Which distribution to plot. Valid options are "cdf" or "pmf".
  \item \texttt{out_labels} \hspace{1cm} Labels for the ordered outcome levels. If \texttt{dist} = "cdf", the highest level of outcome will be dropped.
  \item \texttt{...} \hspace{1cm} Other arguments (not used)
\end{itemize}

\underline{Value}
A list with named entries \texttt{plot} (a \texttt{ggplot2} object) and \texttt{plot_data}, the \texttt{data.frame} from which the plot is made. The latter is included for additional modifications to the plot that are desired.

\underline{POplugin} \hspace{1cm} \textit{Fits a proportional odds model via pooled logistic regression.}

\underline{Description}
The outcome in \texttt{data} (indicated in the \texttt{form} \texttt{object}) should be an ordered factor.

\underline{Usage}
\begin{verbatim}
POplugin(form, data, weights = 1)
\end{verbatim}
Arguments

form         The model formula
data         The data set used to fit the model
weights      Either equal to 1 (no weights) or a vector of length equal to nrow(data)

Value

A list with the fitted glm, the original data, levels of the outcome, and the outcome name

predict.POplugin predict method for a POplugin object

Description

Predict method for a POplugin object

Usage

## S3 method for class 'POplugin'
predict(object, newdata = NULL)

Arguments

object         An object of class POplugin
newdata        A data.frame on which to predict

Value

A data frame with nrow = number of rows in newdata (or the original data frame) and with the number of columns equal to the number of levels of the outcome observed in the original data frame

print.drord    Print the output of a "drord" object.

Description

Print the output of a "drord" object.

Usage

## S3 method for class 'drord'
print(x, ci = "bca", ...)


trimmed_logit

Arguments

- **x**: A "drord" object
- **ci**: Which confidence interval should be printed. Defaults to BCa, but it BCa was not computed in call to drord, defaults back to Wald.
- **...**: Other arguments (not used)

Description

Trimmed logistic function

Usage

```r
trimmed_logit(x)
```

Arguments

- **x**: A numeric between 0 and 1

wald_ci_wmean

Compute a Wald confidence interval for the weighted mean

Description

Compute a Wald confidence interval for the weighted mean

Usage

```r
wald_ci_wmean(wmean_est, alpha)
```

Arguments

- **wmean_est**: The estimated weighted means + estimated covariance matrix.
- **alpha**: Level of confidence interval.

Value

matrix with treatment-specific weighted mean CIs and CI for difference.
Index

* datasets
covid19, 10

bcad intervals, 3
bcalogodds, 3
bca_mannwhitney, 4
bca_marg_dist, 6
bca_wmean, 7

compute_trt_spec_bca_intervals, 8
compute_trt_spec_marg_dist_ptwise_ci, 9
compute_trt_spec_marg_dist_simul_ci, 9
covid19, 10

drord, 10
eif_pmf_k, 13
eif_theta_k, 14
estimate_cdf, 14
estimate_ci_logodds, 15
estimate_ci_mannwhitney, 16
estimate_ci_marg_dist, 17
estimate_ci_wmean, 19
estimate_cond_mean, 20
estimate_eif_wmean, 21
estimate_logodds, 21
estimate_mannwhitney, 22
estimate_pmf, 22
estimate_treat_prob, 24
estimate_wmean, 24
evaluate_beta_cov, 25
evaluate_mannwhitney_gradient, 26
evaluate_marg_cdf_eif, 26
evaluate_marg_cdf_ptwise_ci, 27
evaluate_marg_dist_simul_ci, 27
evaluate_marg_pmf_eif, 28
evaluate_marg_pmf_ptwise_ci, 29
evaluate_theta_cov, 29
evaluate_trt_spec_pmf_eif, 30

evaluate_trt_spec_theta_eif, 30
fit_trt_spec_reg, 31

get_one_logodds, 33
get_one_mannwhitney, 34
get_one_marg_dist, 35
get_one_wmean, 36
getResponseFromFormula, 33

jack_logodds, 37
jack_mannwhitney, 37
jack_marg_cdf, 38
jack_wmean, 39

marginalize_cdf, 40
marginalize_pmf, 40

one_boot_logodds, 41
one_boot_mannwhitney, 42
one_boot_marg_dist, 43
one_boot_wmean, 44

plot.drord, 45
POplugin, 45
predict.POplugin, 46
print.drord, 46

trimmed_logit, 47

wald_cdi_wmean, 47