Package 'riskdiff'

June 23, 2025

Title Risk Difference Estimation with Multiple Link Functions

Version 0.2.0

Date 2025-05-30 # or current date

Description Calculates risk differences (or prevalence differences for cross-sectional data) using generalized linear models with automatic link function selection. Provides robust model fitting with fallback methods, support for stratification and adjustment variables, inverse probability of treatment weighting (IPTW) for causal inference, and publication-ready output formatting. Handles model convergence issues gracefully and provides confidence intervals using multiple approaches. Methods are based on approaches described in Mark W. Donoghoe and Ian C. Marschner (2018) ``logbin: An R Package for Relative Risk Regression Using the Log-Binomial Model'' <doi:10.18637/jss.v086.i09> for robust GLM fitting, Peter C. Austin (2011) ``An Introduction to Propensity Score Methods for Reducing the Effects of Confounding in Observational Studies'' <doi:10.1080/00273171.2011.568786> for IPTW methods, and standard epidemiological methods for risk difference estimation as described in Kenneth J. Rothman, Sander Greenland and Timothy L. Lash (2008, ISBN:9780781755641) ``Modern Epidemiology''.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Depends R (>= 3.5.0)

Imports dplyr (>= 1.0.0), purrr, tibble, rlang, scales, stringr, stats

Suggests kableExtra, testthat (>= 3.0.0), knitr, rmarkdown, ggplot2, covr

Config/testthat/edition 3

URL https://github.com/jackmurphy2351/riskdiff

BugReports https://github.com/jackmurphy2351/riskdiff/issues

LazyData true

VignetteBuilder knitr

NeedsCompilation no

Author John D. Murphy [aut, cre] (ORCID:
<https://orcid.org/0000-0002-7714-9976>, MPH, PhD)

Maintainer John D. Murphy <jackmurphy2351@gmail.com>

Repository CRAN

Date/Publication 2025-06-23 15:50:05 UTC

Contents

`.supports_unicode` *Safe Unicode Display Functions*

Description

Functions to safely display Unicode characters with fallbacks for systems that don't support them properly.

Usage

```
.supports_unicode()
```

`cachar_sample` *Synthetic Cancer Risk Factor Study Data*

Description

A synthetic dataset inspired by cancer screening and risk factor patterns observed during an opportunistic screening program conducted at the Cachar Cancer Hospital and Research Centre in Northeast India, specifically designed to reflect authentic epidemiological relationships without using real patient data.

Usage

```
cachar_sample
```

Format

A data frame with 2,500 rows and 12 variables:

id Participant identifier (1 to 2500)

age Age in years (continuous, range 18-84)

sex Biological sex: "male" or "female"

residence Residence type: "rural", "urban", or "urban slum"

smoking Current smoking status: "No" or "Yes"

tobacco_chewing Current tobacco chewing: "No" or "Yes"

areca_nut Current areca nut use: "No" or "Yes"

alcohol Current alcohol use: "No" or "Yes"

abnormal_screen Binary outcome: 1 = abnormal screening (precancerous lesions or cancer), 0 = normal

head_neck_abnormal Binary outcome: 1 = head/neck abnormality detected, 0 = normal

age_group Age categories: "Under 40", "40-60", "Over 60"

tobacco_areca_both Combined exposure: "Yes" if both tobacco_chewing and areca_nut are "Yes", "No" otherwise

Details

This synthetic dataset was designed to reflect authentic epidemiological patterns observed in Northeast India, particularly the distinctive tobacco and areca nut use patterns of the region. All data points are mathematically generated rather than collected from real individuals.

Key epidemiological features modeled:

- **Areca nut use**: Very high prevalence (~69%) reflecting regional cultural practices
- **Tobacco chewing**: Moderate to high prevalence (~53%), often used with areca nut
- **Smoking**: Lower prevalence (~13%) with strong male predominance

- **Cancer outcomes**: Realistic prevalence (~3.5%) for population-based screening, including both precancerous lesions and invasive cancers
- **Geographic patterns**: Predominantly rural population (~87%)

Synthetic Data Advantages: The synthetic approach preserves authentic statistical relationships while:

- Avoiding any privacy or ethical concerns
- Ensuring reproducible examples and tests
- Providing controlled demonstration scenarios
- Maintaining cultural authenticity for educational purposes

Risk Factor Relationships: The data models realistic dose-response relationships between multiple tobacco exposures and cancer outcomes, with particularly strong associations for areca nut use and head/neck abnormalities, reflecting authentic epidemiological patterns from this region.

Note

This synthetic dataset is designed for educational and software demonstration purposes. While the statistical relationships reflect authentic epidemiological patterns, the data should not be used for research conclusions about real populations. The cultural patterns represented (high areca nut use, specific tobacco consumption practices) are authentic to Northeast India.

Source

Synthetic dataset created for the riskdiff package. Inspired by cancer screening patterns observed in Northeast India but contains no real patient data. Statistical relationships designed to reflect authentic epidemiological patterns from this region for educational and methodological purposes.

References

Epidemiological patterns modeled after studies of tobacco use and cancer risk in Northeast India. For research involving actual populations from this region, consult published literature on areca nut and tobacco-related cancer risks in South Asian populations.

Warnakulasuriya S, Trivedy C, Peters TJ (2002). "Areca nut use: an independent risk factor for oral cancer." BMJ, 324(7341), 799-800.

Gupta PC, Ray CS (2004). "Epidemiology of betel quid use." Annals of the Academy of Medicine, Singapore, 33(4 Suppl), 31-36.

Examples

```
data(cachar_sample)
head(cachar_sample)

# Basic descriptive statistics
table(cachar_sample$areca_nut, cachar_sample$abnormal_screen)

# Regional tobacco use patterns
with(cachar_sample, table(areca_nut, tobacco_chewing))
```

```
# Simple risk difference for areca nut and abnormal screening
rd_areca <- calc_risk_diff(
  data = cachar_sample,
  outcome = "abnormal_screen",
  exposure = "areca_nut"
)
print(rd_areca)

# Age-adjusted analysis
rd_adjusted <- calc_risk_diff(
  data = cachar_sample,
  outcome = "abnormal_screen",
  exposure = "areca_nut",
  adjust_vars = "age"
)
print(rd_adjusted)

# Stratified by sex
rd_stratified <- calc_risk_diff(
  data = cachar_sample,
  outcome = "head_neck_abnormal",
  exposure = "smoking",
  strata = "sex"
)
print(rd_stratified)

# Multiple tobacco exposures comparison
rd_smoking <- calc_risk_diff(cachar_sample, "abnormal_screen", "smoking")
rd_chewing <- calc_risk_diff(cachar_sample, "abnormal_screen", "tobacco_chewing")
rd_areca <- calc_risk_diff(cachar_sample, "abnormal_screen", "areca_nut")

# Compare risk differences
cat("Risk differences for abnormal screening:\n")
cat("Smoking:", sprintf("%.1f%%", rd_smoking$rd * 100), "\n")
cat("Tobacco chewing:", sprintf("%.1f%%", rd_chewing$rd * 100), "\n")
cat("Areca nut:", sprintf("%.1f%%", rd_areca$rd * 100), "\n")

# Create summary table
cat(create_simple_table(rd_areca, "Abnormal Screening Risk by Areca Nut Use"))
```

`calc_iptw_weights` *Calculate Propensity Scores and IPTW Weights*

Description

Calculates propensity scores and inverse probability of treatment weights for use in standardized risk difference estimation. Implements multiple approaches for weight calculation and includes diagnostic tools.

Usage

```
calc_iptw_weights(
  data,
  treatment,
  covariates,
  method = "logistic",
  weight_type = "ATE",
  stabilize = TRUE,
  trim_weights = TRUE,
  trim_quantiles = c(0.01, 0.99),
  verbose = FALSE
)
```

Arguments

data	A data frame containing treatment and covariate data
treatment	Character string naming the binary treatment variable
covariates	Character vector of covariate names for propensity score model
method	Method for propensity score estimation: "logistic" (default), "probit", or "cloglog"
weight_type	Type of weights to calculate: "ATE" (average treatment effect, default), "ATT" (average treatment effect on treated), "ATC" (average treatment effect on controls)
stabilize	Logical indicating whether to use stabilized weights (default: TRUE)
trim_weights	Logical indicating whether to trim extreme weights (default: TRUE)
trim_quantiles	Vector of length 2 specifying quantiles for weight trimming (default: c(0.01, 0.99))
verbose	Logical indicating whether to print diagnostic information (default: FALSE)

Details

Propensity Score Estimation:

The function fits a model predicting treatment assignment from covariates:

- **Logistic regression**: Standard approach, assumes logit link
- **Probit regression**: Uses probit link, may be more robust with extreme probabilities
- **Complementary log-log**: Useful when treatment is rare

Weight Types:

- **ATE weights**: 1/pi(X) for treated, 1/(1-pi(X)) for controls
- **ATT weights**: 1 for treated, pi(X)/(1-pi(X)) for controls
- **ATC weights**: (1-pi(X))/pi(X) for treated, 1 for controls

Where pi(X) is the propensity score (probability of treatment given X).

Stabilized Weights:

When stabilize=TRUE, weights are multiplied by marginal treatment probabilities to reduce variance while maintaining unbiasedness (Robins et al., 2000).

Weight Trimming:
Extreme weights can cause instability. Trimming replaces weights outside specified quantiles with the quantile values (Crump et al., 2009).

Value

A list containing:

data Original data with added propensity scores and weights

ps_model Fitted propensity score model

weights Vector of calculated weights

ps Vector of propensity scores

diagnostics List of diagnostic information including balance statistics

method Method used for propensity score estimation

weight_type Type of weights calculated

References

Austin PC (2011). "An Introduction to Propensity Score Methods for Reducing the Effects of Confounding in Observational Studies." Multivariate Behavioral Research, 46(3), 399-424. doi:10.1080/00273171.2011.568786

Crump RK, Hotz VJ, Imbens GW, Mitnik OA (2009). "Dealing with Limited Overlap in Estimation of Average Treatment Effects." Biometrika, 96(1), 187-199.

Hernan MA, Robins JM (2020). Causal Inference: What If. Boca Raton: Chapman & Hall/CRC.

Robins JM, Hernan MA, Brumback B (2000). "Marginal Structural Models and Causal Inference in Epidemiology." Epidemiology, 11(5), 550-560.

Examples

```
data(cachar_sample)

# Calculate ATE weights for areca nut use
iptw_result <- calc_iptw_weights(
  data = cachar_sample,
  treatment = "areca_nut",
  covariates = c("age", "sex", "residence", "smoking"),
  weight_type = "ATE"
)

# Check balance
print(iptw_result$diagnostics$balance_table)

# Calculate ATT weights (effect on the treated)
iptw_att <- calc_iptw_weights(
  data = cachar_sample,
  treatment = "tobacco_chewing",
  covariates = c("age", "sex", "residence", "areca_nut"),
  weight_type = "ATT"
)
```

`calc_risk_diff` *Calculate Risk Differences with Robust Model Fitting and Boundary Detection*

Description

Calculates risk differences with enhanced boundary detection and robust confidence interval methods. Version 0.2.0 adds explicit detection of boundary cases where the MLE lies at the edge of the parameter space, providing more reliable inference in these situations.

Usage

```
calc_risk_diff(
  data,
  outcome,
  exposure,
  adjust_vars = NULL,
  strata = NULL,
  link = "auto",
  alpha = 0.05,
  boundary_method = "auto",
  verbose = FALSE
)
```

Arguments

`data`	A data frame containing all necessary variables
`outcome`	Character string naming the binary outcome variable
`exposure`	Character string naming the exposure variable of interest
`adjust_vars`	Character vector of variables to adjust for (default: NULL)
`strata`	Character vector of stratification variables (default: NULL)
`link`	Character string specifying link function preference
`alpha`	Significance level for confidence intervals (default: 0.05)
`boundary_method`	Method for boundary case CIs: "auto", "profile", "bootstrap" (default: "auto")
`verbose`	Logical indicating whether to print diagnostic messages

Details

New in Version 0.2.0: Boundary Detection:

This version adds explicit detection of boundary cases where the maximum likelihood estimate lies at the edge of the valid parameter space. This commonly occurs with:

- **Identity link models**: When fitted probabilities approach 0 or 1
- **Log link models**: When fitted probabilities approach 1

- **Logit link models**: When complete or quasi-separation occurs

Statistical Theory:

When the MLE is on the boundary, standard asymptotic theory may not apply:

- Wald confidence intervals can be too narrow or asymmetric
- Standard errors from the information matrix may be inappropriate
- The sampling distribution may not be normal

Robust Inference Methods:

For boundary cases, the function implements:

- **Profile likelihood intervals** (preferred when computationally feasible)
- **Bootstrap confidence intervals** (robust but computationally intensive)
- **Modified Wald intervals** with boundary adjustments

Value

A tibble of class "riskdiff_result" with additional boundary information:

exposure_var Character. Name of exposure variable

rd Numeric. Risk difference estimate

ci_lower Numeric. Lower confidence interval bound

ci_upper Numeric. Upper confidence interval bound

p_value Numeric. P-value for test of RD = 0

model_type Character. Link function used

n_obs Integer. Number of observations

on_boundary Logical. TRUE if MLE is on parameter space boundary

boundary_type Character. Type of boundary: "none", "upper_bound", "lower_bound", "separation"

boundary_warning Character. Warning message for boundary cases (if any)

ci_method Character. Method used for confidence intervals

References

Marschner IC, Gillett AC (2012). Relative Risk Regression: Reliable and Flexible Methods for Log-Binomial Models. Biostatistics, 13(1), 179-192.

Venzon DJ, Moolgavkar SH (1988). A Method for Computing Profile-Likelihood-Based Confidence Intervals. Journal of the Royal Statistical Society, 37(1), 87-94.

Donoghoe MW, Marschner IC (2018). logbin: An R Package for Relative Risk Regression Using the Log-Binomial Model. Journal of Statistical Software, 86(9), 1-22. doi:10.18637/jss.v086.i09

Examples

```
data(cachar_sample)

# Basic usage with boundary detection
result <- calc_risk_diff(
  data = cachar_sample,
  outcome = "abnormal_screen",
  exposure = "smoking"
)

# Check for boundary cases
if (any(result$on_boundary)) {
  cat("Boundary case detected!\n")
  cat("Boundary type:", result$boundary_type[result$on_boundary], "\n")
  cat("CI method used:", result$ci_method[result$on_boundary], "\n")
}

# Force profile likelihood CIs for all cases
result_profile <- calc_risk_diff(
  data = cachar_sample,
  outcome = "abnormal_screen",
  exposure = "smoking",
  boundary_method = "profile"
)
```

---

calc_risk_diff_iptw *Calculate Standardized Risk Differences Using IPTW*

Description

Calculates standardized risk differences using inverse probability of treatment weighting. This approach estimates causal effects under the assumption of no unmeasured confounding by creating a pseudo-population where treatment assignment is independent of measured confounders.

Usage

```
calc_risk_diff_iptw(
  data,
  outcome,
  treatment,
  covariates,
  iptw_weights = NULL,
  weight_type = "ATE",
  ps_method = "logistic",
  stabilize = TRUE,
  trim_weights = TRUE,
  alpha = 0.05,
```

```
  bootstrap_ci = FALSE,
  boot_n = 1000,
  verbose = FALSE
)
```

Arguments

data	A data frame containing outcome, treatment, and covariate data
outcome	Character string naming the binary outcome variable
treatment	Character string naming the binary treatment variable
covariates	Character vector of covariate names for propensity score model
iptw_weights	Optional vector of pre-calculated IPTW weights
weight_type	Type of weights if calculating: "ATE", "ATT", or "ATC" (default: "ATE")
ps_method	Method for propensity score estimation (default: "logistic")
stabilize	Whether to use stabilized weights (default: TRUE)
trim_weights	Whether to trim extreme weights (default: TRUE)
alpha	Significance level for confidence intervals (default: 0.05)
bootstrap_ci	Whether to use bootstrap confidence intervals (default: FALSE)
boot_n	Number of bootstrap replicates if bootstrap_ci=TRUE (default: 1000)
verbose	Whether to print diagnostic information (default: FALSE)

Details

Causal Interpretation:

IPTW estimates causal effects by weighting observations to create balance on measured confounders. The estimand depends on the weight type:

- **ATE**: Average treatment effect in the population
- **ATT**: Average treatment effect among those who received treatment
- **ATC**: Average treatment effect among those who did not receive treatment

Standard Errors:

By default, uses robust (sandwich) standard errors that account for propensity score estimation uncertainty. Bootstrap confidence intervals are available as an alternative that may perform better with small samples.

Assumptions:

1. **No unmeasured confounding**: All confounders are measured and included
2. **Positivity**: All subjects have non-zero probability of receiving either treatment
3. **Correct model specification**: Propensity score model is correctly specified

Value

A tibble of class "riskdiff_iptw_result" containing:

treatment_var Character. Name of treatment variable

rd_iptw Numeric. IPTW-standardized risk difference

ci_lower Numeric. Lower confidence interval bound

ci_upper Numeric. Upper confidence interval bound

p_value Numeric. P-value for test of null hypothesis

weight_type Character. Type of weights used

effective_n Numeric. Effective sample size

risk_treated Numeric. Risk in treated group

risk_control Numeric. Risk in control group

Examples

```
data(cachar_sample)

# Standard ATE estimation
rd_iptw <- calc_risk_diff_iptw(
  data = cachar_sample,
  outcome = "abnormal_screen",
  treatment = "areca_nut",
  covariates = c("age", "sex", "residence", "smoking")
)
print(rd_iptw)

# ATT estimation with bootstrap CI
rd_att <- calc_risk_diff_iptw(
  data = cachar_sample,
  outcome = "head_neck_abnormal",
  treatment = "tobacco_chewing",
  covariates = c("age", "sex", "residence", "areca_nut"),
  weight_type = "ATT",
  bootstrap_ci = TRUE,
  boot_n = 500
)
print(rd_att)
```

---

check_iptw_assumptions

Check IPTW Assumptions

Description

Provides diagnostic checks for key IPTW assumptions including positivity, balance, and model specification. Returns a comprehensive summary with recommendations for potential issues.

Usage

```
check_iptw_assumptions(
  iptw_result,
  balance_threshold = 0.1,
  extreme_weight_threshold = 10,
  verbose = TRUE
)
```

Arguments

iptw_result	An iptw_result object from calc_iptw_weights()
balance_threshold	Threshold for acceptable standardized difference (default: 0.1)
extreme_weight_threshold	Threshold for flagging extreme weights (default: 10)
verbose	Whether to print detailed diagnostics (default: TRUE)

Value

A list containing:

overall_assessment Character indicating "PASS", "CAUTION", or "FAIL"

positivity List with positivity checks and recommendations

balance List with balance assessment and problematic variables

weights List with weight distribution diagnostics

recommendations Character vector of specific recommendations

Examples

```
data(cachar_sample)

iptw_result <- calc_iptw_weights(
  data = cachar_sample,
  treatment = "areca_nut",
  covariates = c("age", "sex", "residence", "smoking")
)

# Check assumptions
assumptions <- check_iptw_assumptions(iptw_result)
print(assumptions$overall_assessment)
print(assumptions$recommendations)
```

`create_balance_plots` *Create Balance Plots for IPTW Analysis*

Description

Creates visualizations to assess covariate balance before and after IPTW weighting. Includes love plots (standardized differences) and propensity score distribution plots.

Usage

```
create_balance_plots(
  iptw_result,
  plot_type = "both",
  threshold = 0.1,
  save_plots = FALSE,
  plot_dir = "plots"
)
```

Arguments

`iptw_result`	An iptw_result object from calc_iptw_weights()
`plot_type`	Type of plot: "love" for standardized differences, "ps" for propensity score distributions, or "both"
`threshold`	Threshold for acceptable standardized difference (default: 0.1)
`save_plots`	Whether to save plots to files (default: FALSE)
`plot_dir`	Directory to save plots if save_plots=TRUE (default: "plots")

Details

Love Plot:

Shows standardized differences for each covariate before and after weighting. Points represent standardized differences, with lines connecting before/after values. Horizontal lines show common thresholds (0.1, 0.25) for acceptable balance.

Propensity Score Plot:

Shows distributions of propensity scores by treatment group before and after weighting. Good overlap indicates positivity assumption is met.

Value

A ggplot object (if plot_type is "love" or "ps") or a list of ggplot objects (if plot_type is "both"). If ggplot2 is not available, returns a message and creates base R plots.

Examples

```
data(cachar_sample)

# Calculate IPTW weights
iptw_result <- calc_iptw_weights(
  data = cachar_sample,
  treatment = "areca_nut",
  covariates = c("age", "sex", "residence", "smoking")
)

# Create balance plots
if (requireNamespace("ggplot2", quietly = TRUE)) {
  plots <- create_balance_plots(iptw_result, plot_type = "both")
  print(plots$love_plot)
  print(plots$ps_plot)
}
```

`create_rd_table` *Create Formatted Table of Risk Difference Results*

Description

Creates a publication-ready table of risk difference results with appropriate grouping and formatting. Requires the kableExtra package for full functionality.

Usage

```
create_rd_table(
  results,
  caption = "Risk Differences",
  include_model_type = FALSE,
  ...
)
```

Arguments

`results`	Results tibble from calc_risk_diff()
`caption`	Table caption (default: "Risk Differences")
`include_model_type`	Whether to include model type column (default: FALSE)
`...`	Additional arguments passed to kableExtra::kable()

Value

If kableExtra is available, returns a kable table object suitable for rendering in R Markdown or HTML. The table includes formatted risk differences, confidence intervals, and p-values with appropriate styling and footnotes. If kableExtra is not available, returns a formatted tibble with the same information in a basic data frame structure.

Examples

```
data(cachar_sample)
results <- calc_risk_diff(cachar_sample, "abnormal_screen", "smoking")

# Basic table (works without kableExtra)
basic_table <- create_rd_table(results, caption = "Risk of Abnormal Cancer Screening")
print(basic_table)

# Enhanced table (requires kableExtra)
if (requireNamespace("kableExtra", quietly = TRUE)) {
  enhanced_table <- create_rd_table(
    results,
    caption = "Risk of Abnormal Cancer Screening by Smoking Status",
    include_model_type = TRUE
  )
  print(enhanced_table)
}
```

---

create_simple_table *Create a Simple Summary Table*

Description

Creates a simple text-based summary table that doesn't require kableExtra.

Usage

```
create_simple_table(results, title = "Risk Difference Results")
```

Arguments

results	Results tibble from calc_risk_diff()
title	Optional title for the table

Value

A formatted character vector representing the table

Examples

```
data(cachar_sample)
results <- calc_risk_diff(cachar_sample, "abnormal_screen", "smoking")
cat(create_simple_table(results))
```

`format_risk_diff` *Format Risk Difference Results for Display*

Description

Formats numerical values in risk difference results for presentation, with appropriate percentage formatting and rounding.

Usage

```
format_risk_diff(results, digits = 2, p_accuracy = 0.001)
```

Arguments

`results`	Results tibble from calc_risk_diff()
`digits`	Number of decimal places for percentages (default: 2)
`p_accuracy`	Accuracy for p-values (default: 0.001)

Value

Tibble with additional formatted columns

Examples

```
data(cachar_sample)
results <- calc_risk_diff(cachar_sample, "abnormal_screen", "smoking")
formatted <- format_risk_diff(results)
print(formatted)
```

print.iptw_result *Print Method for IPTW Results*

Description

Print Method for IPTW Results

Usage

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

Arguments

x	An iptw_result object
...	Additional arguments passed to print

print.riskdiff_iptw_result *Print Method for IPTW Risk Difference Results*

Description

Print Method for IPTW Risk Difference Results

Usage

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

Arguments

x	A riskdiff_iptw_result object
...	Additional arguments passed to print

print.riskdiff_result *Print method for riskdiff_result objects*

Description

Prints risk difference results in a formatted, readable way showing key statistics including risk differences, confidence intervals, and model types used. Version 0.2.0+ includes boundary case detection.

Usage

```
## S3 method for class 'riskdiff_result'
print(x, show_boundary = TRUE, ...)
```

Arguments

x	A riskdiff_result object from calc_risk_diff()
show_boundary	Logical, whether to show boundary information when available (default: TRUE)
...	Additional arguments passed to print methods

Value

Invisibly returns the original riskdiff_result object (x). Called primarily for its side effect of printing formatted results to the console.

Examples

```
data(cachar_sample)
result <- calc_risk_diff(cachar_sample, "abnormal_screen", "smoking")
print(result)

# Show boundary information if available
print(result, show_boundary = TRUE)
```

summary.riskdiff_iptw_result *Summary Method for IPTW Risk Difference Results*

Description

Provides a comprehensive summary of IPTW risk difference analysis including effect estimates, diagnostics, and interpretation guidance.

Usage

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

Arguments

object	A riskdiff_iptw_result object
...	Additional arguments (currently ignored)

Value

Invisibly returns the input object. Called primarily for side effects (printing summary).

Examples

```
data(cachar_sample)

rd_iptw <- calc_risk_diff_iptw(
  data = cachar_sample,
  outcome = "abnormal_screen",
  treatment = "areca_nut",
  covariates = c("age", "sex", "residence", "smoking")
)

summary(rd_iptw)
```

Index