Package ‘PosiR’

April 30, 2025

Type Package

Title Post-Selection Inference via Simultaneous Confidence Intervals

Version 0.1.2

Description Post-selection inference in linear regression models, constructing simultaneous confidence intervals across a user-specified universe of models. Implements the methodology described in Kuchibhotla, Kolassa, and Kuffner (2022) ``Post-Selection Inference'' <doi:10.1146/annurev-statistics-100421-044639> to ensure valid inference after model selection, with applications in high-dimensional settings like Lasso selection.

License MIT + file LICENSE

Encoding UTF-8

Suggests knitr, pbapply, rmarkdown, testthat (>= 3.0.0), dplyr, glmnet

Config/testthat/edition 3

RoxygenNote 7.3.2

URL `https://github.com/Chukyhenry/PosiR`

BugReports `https://github.com/Chukyhenry/PosiR/issues`

Imports graphics, parallel, stats

VignetteBuilder knitr

Depends R (>= 4.0.0)

NeedsCompilation no

Author Henry Chukwuma [aut, cre]

Maintainer Henry Chukwuma <`chukyhenry55@gmail.com`>

Repository CRAN

Date/Publication 2025-04-30 15:10:06 UTC

Contents

---

plot.simultaneous_ci_result *Plot Simultaneous Confidence Intervals*

Description

Visualizes confidence intervals returned by `simultaneous_ci()` using base R graphics. Estimates are shown as points with corresponding CI segments, grouped and labeled by model and coefficient name. Supports customization for log scale, character sizes, label trimming, and reference lines.

Usage

```
## S3 method for class 'simultaneous_ci_result'
plot(
  x,
  y = NULL,
  subset_pars = NULL,
  log.scale = FALSE,
  cex = 0.8,
  cex.labels = 0.8,
  las.labels = 1,
  pch = 16,
  col.estimate = "blue",
  col.ci = "darkgray",
  col.ref = "red",
  ref.line.pos = 0,
  lty.ref = 2,
  main = "Simultaneous Confidence Intervals",
  xlab = NULL,
  label.trim = NULL,
  ...
)
```

Arguments

`x`	An object of class `simultaneous_ci_result`, typically returned by `simultaneous_ci()`.
`y`	Ignored.
`subset_pars`	Optional character vector. Coefficient names to subset the plot. Default: all.
`log.scale`	Logical. Plot on logarithmic scale. Intervals crossing 0 or with nonpositive bounds are excluded.
`cex`	Point size for estimates. Default = 0.8.
`cex.labels`	Label size for y-axis. Default = 0.8.
`las.labels`	Orientation of y-axis labels (0, 1, 2, or 3). Default = 1.
`pch`	Plot character for point estimates. Default = 16.

`col.estimate`	Color of point estimates. Default = "blue".
`col.ci`	Color of confidence interval lines. Default = "darkgray".
`col.ref`	Color of reference line(s). Default = "red".
`ref.line.pos`	Position(s) for vertical reference line(s). Default = 0. Set to NULL to omit.
`lty.ref`	Line type for reference lines. Default = 2 (dashed).
`main`	Plot title. Default = "Simultaneous Confidence Intervals".
`xlab`	X-axis label. If NULL and `log.scale = TRUE`, label defaults to "Log Estimate".
`label.trim`	Integer. Trims long coefficient labels to this width (adds "..."). Optional.
`...`	Additional arguments passed for future use (currently ignored).

Value

Invisibly returns a list:

- `ycoords`: Named vector of y-axis positions for each label
- `xlim`: Range of x-axis limits used
- `ylim`: Range of y-axis limits used

If no valid intervals are available for plotting, returns `invisible(NULL)`.

Examples

```
set.seed(1)
X <- matrix(rnorm(100*2), 100, 2, dimnames = list(NULL, c("X1", "X2")))
y <- 1 + X[,1] - X[,2] + rnorm(100)
res <- simultaneous_ci(X, y, list(mod = 1:3), B = 100, add_intercept = TRUE)
plot(res)
```

`simultaneous_ci` *Compute Simultaneous Confidence Intervals via Bootstrap (Post-Selection Inference)*

Description

Implements Algorithm 1 from the reference paper using bootstrap-based max-t statistics to construct valid simultaneous confidence intervals for selected regression coefficients across a user-specified universe of linear models.

Usage

```
simultaneous_ci(
  X,
  y,
  Q_universe,
  alpha = 0.05,
  B = 1000,
  add_intercept = TRUE,
  bootstrap_method = "pairs",
  cores = 1,
  use_pbapply = TRUE,
  seed = NULL,
  verbose = TRUE,
  ...
)
```

Arguments

`X`	Numeric matrix (n x p): Design matrix. Must have unique column names. Do not include an intercept if `add_intercept = TRUE`.
`y`	Numeric vector (length n): Response vector.
`Q_universe`	Named list of numeric vectors. Each element specifies a model as a vector of column indices (accounting for intercept if `add_intercept = TRUE`). Names are used to identify each model in results.
`alpha`	Significance level for the confidence intervals. Default is 0.05.
`B`	Integer. Number of bootstrap samples. Default is 1000.
`add_intercept`	Logical. If TRUE, adds an intercept as the first column of the design matrix. Default is TRUE.
`bootstrap_method`	Character. Bootstrap type. Only "pairs" is currently supported.
`cores`	Integer. Number of CPU cores to use for bootstrap parallelization. Default is 1.
`use_pbapply`	Logical. Use `pbapply` for progress bars if available. Default is TRUE.
`seed`	Optional numeric. Random seed for reproducibility. Used for parallel-safe RNG.
`verbose`	Logical. Whether to display status messages. Default is TRUE.
`...`	Reserved for future use.

Details

Supports parallel execution, internal warnings capture, and returns structured results with estimates, intervals, bootstrap diagnostics, and inference statistics.

Value

A list of class `simultaneous_ci_result` with elements:

- `intervals`: Data frame with estimates, confidence intervals, variances, and SEs
- `K_alpha`: Bootstrap (1 - alpha) quantile of max-t statistics
- `T_star_b`: Vector of bootstrap max-t statistics
- `n_valid_T_star_b`: Number of finite bootstrap max-t statistics
- `alpha`, `B`, `bootstrap_method`: Metadata
- `warnings_list`: Internal warnings collected during bootstrap/model fitting
- `valid_bootstrap_counts`: Valid bootstrap replicates per parameter
- `n_bootstrap_errors`: Total bootstrap fitting errors

References

Kuchibhotla, A., Kolassa, J., & Kuffner, T. (2022). Post-selection inference. *Annual Review of Statistics and Its Application*, 9(1), 505–527.

Examples

```
set.seed(123)
X <- matrix(rnorm(100 * 2), 100, 2, dimnames = list(NULL, c("X1", "X2")))
y <- X[,1] * 0.5 + rnorm(100)
Q <- list(model = 1:2)
res <- simultaneous_ci(X, y, Q, B = 100, cores = 1)
print(res$intervals)
plot(res)
```

Index