Package 'chestR'

September 12, 2026

Title Kernel-Weighted Cox Regression for Treatment Effect Heterogeneity

Version 0.1.0

Description Explores treatment effect heterogeneity and candidate predictive biomarkers by re-fitting weighted Cox proportional hazards models on a biomarker grid. Kernel weights centred at each grid point produce local coefficient estimates that can be visualised across biomarker space. Builds on ideas related to graphical Cox treatment-covariate interaction methods [see Bonetti and Gelber (2004) <doi:10.1093/biostatistics/kxh002> and local partial-likelihood approaches Fan, Lin and Zhou (2006) <doi:10.1214/009053605000000796>].

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

VignetteBuilder knitr

Imports survival, ggplot2, scales

Suggests knitr, rmarkdown, mvtnorm, testthat (>= 3.0.0)

Config/testthat/edition 3

Language en-GB

NeedsCompilation no

Author Richard Jackson [aut, cre], Caroline Jeffery [aut]

Maintainer Richard Jackson <richj23@liverpool.ac.uk>

Repository CRAN

Date/Publication 2026-09-12 15:10:02 UTC

Contents

---

chestr *Kernel-weighted Cox estimates over a biomarker grid*

Description

Fits a weighted Cox model at each point on a biomarker grid. Observations near the grid point receive higher weight, yielding local estimates of treatment and covariate effects.

Usage

```
chestr(
  base,
  biom,
  grid.size = 25,
  method = c("distance", "square_distance", "legacy"),
  kern.adj = 4,
  bw = NULL,
  min_events_per_df = 10,
  treat_term = NULL
)
```

Arguments

base	A fitted coxph object (the global model).
biom	Biomarker values: a vector, matrix, or data.frame (1–2 columns supported for plot.chestr()).
grid.size	Number of grid points per biomarker dimension.
method	Weighting scheme: • "distance" (default): scaled biomarkers + Gaussian kernel. • "square_distance": inverse-square distance weights. • "legacy": unscaled Gaussian kernel; use kern.adj for bandwidth.
kern.adj	Bandwidth divisor when method = "legacy". Larger values give narrower kernels.
bw	Optional kernel standard deviation; reserved for future use.
min_events_per_df	Minimum Kish effective events per model degree of freedom required to fit a local model (default 10). Effective events use Kish ESS among event weights: (sum w_e)^2 / sum(w_e^2). Grid points below the threshold are skipped (coefficients set to NA) and a warning is issued. Set to 0 or NULL to disable.
treat_term	Optional name of the treatment coefficient to record on the returned object (used as the default in plot.chestr() / chestr_test()).

Value

An object of class "chestr": a list with

- estimates: data.frame of grid coordinates, coefficients, SEs, ESS columns
- base: the fitted coxph model
- biom: biomarker data.frame used for weighting / plotting
- data: analysis data recovered from base when available (for permutation tests)
- treat_term: optional treatment coefficient name
- method, kern.adj, grid.size, min_events_per_df: fitting settings

See Also

plot.chestr(), chestr_test()

Examples

```
library(survival)
set.seed(1)
n <- 60
dat <- data.frame(
  time = rexp(n, 0.1), status = 1L,
  trt = rbinom(n, 1, 0.5), biom = rnorm(n)
)
dat$status <- as.integer(dat$time < 5)
dat$time[dat$status == 0L] <- 5
base <- coxph(Surv(time, status) ~ trt, data = dat)
cr <- chestr(base, dat$biom, grid.size = 5, min_events_per_df = 0,
             treat_term = "trt")
head(cr$estimates)
plot(cr, trt.param = "trt")
```

---

chestr_test *Permutation test for treatment-effect heterogeneity*

Description

Tests the null that the treatment effect is constant over biomarker space by permuting treatment labels, re-fitting the global Cox model and chestr() surface, and comparing observed heterogeneity statistics to the permutation null.

Usage

```
chestr_test(
  x,
  treat_term = NULL,
  data = NULL,
  treat_var = NULL,
  B = 100,
  seed = NULL,
  reliable_only = TRUE
)
```

Arguments

x	A "chestr" object from chestr().
treat_term	Treatment coefficient name. Optional if stored on x or uniquely inferable.
data	Analysis data frame. Optional if x$data is present or the data can be recovered from x$base$call.
treat_var	Treatment column in data. Optional if inferable.
B	Number of permutations (default 100).
seed	Optional random seed for reproducibility.
reliable_only	If TRUE (default), only reliable grid points enter the statistics, and the same ESS rule as x is used when rebuilding each permuted chestr() fit.

Details

Where possible, arguments are taken from x:

- treat_term from x$treat_term (or inferred)
- data from x$data (stored by chestr() when recoverable)
- treat_var inferred from treat_term
- grid / kernel / ESS settings from x

Observed statistics are the ESS-weighted L2 deviation and the maximum standardised deviation of local treatment coefficients from the global coefficient.

Value

An object of class "chestr_test" with observed statistics, permutation p-values, and null draws.

See Also

chestr(), plot.chestr()

Examples

```
library(survival)
set.seed(1)
n <- 80
dat <- data.frame(
  time = rexp(n, 0.15), status = 1L,
  trt = rbinom(n, 1, 0.5),
  b1 = rnorm(n), b2 = rnorm(n)
)
dat$status <- as.integer(dat$time < 4)
dat$time[dat$status == 0L] <- 4
base <- coxph(Surv(time, status) ~ trt, data = dat)
cr <- chestr(base, dat[, c("b1", "b2")], grid.size = 5,
             min_events_per_df = 0, treat_term = "trt")
tst <- chestr_test(cr, B = 19, seed = 1)
tst
```

`plot.chestr` *Plot local treatment effects from a* `chestr` *object*

Description

Visualises a local coefficient stored in `chestr()` output using **ggplot2**. For two biomarkers, a coloured grid of local log-hazard ratios is shown (point size reflects local information). For a single biomarker, a line of local hazard ratios is shown with a dashed horizontal reference at the global hazard ratio from `x$base`. Model and biomarker data are taken from `x` itself.

Usage

```
## S3 method for class 'chestr'
plot(
  x,
  trt.param = NULL,
  pnt.scale = 3,
  col.scale = NULL,
  pts = TRUE,
  data_cex = 0.6,
  reliable_only = TRUE,
  ...
)
```

Arguments

`x`	An object of class `"chestr"` from `chestr()`.
`trt.param`	Column name in `x$estimates` for the effect to colour (2D) or plot as a hazard-ratio curve (1D). If `NULL`, uses `x$treat_term` when set / uniquely inferable.

pnt.scale	Multiplier for point size (proportional to local information). If NULL, size is scaled automatically from grid size. Used for 2D plots.
col.scale	Colour limits for 2D plots: NULL (symmetric -3 to 3), "obs" (data-driven), or a length-2 numeric range (min, max).
pts	If TRUE, overlay observed biomarker values (2D points or 1D rug).
data_cex	Point size for observed biomarker data (when pts = TRUE).
reliable_only	If TRUE (default), only plot grid points that passed the events-per-df safeguard.
...	Ignored (for S3 compatibility).

Value

The ggplot object (invisibly).

See Also

chestr(), chestr_test()

Examples

```
library(survival)
set.seed(1)
n <- 60
dat <- data.frame(
  time = rexp(n, 0.1), status = 1L,
  trt = rbinom(n, 1, 0.5), biom = rnorm(n)
)
dat$status <- as.integer(dat$time < 5)
dat$time[dat$status == 0L] <- 5
base <- coxph(Surv(time, status) ~ trt, data = dat)
cr <- chestr(base, dat$biom, grid.size = 5, min_events_per_df = 0,
             treat_term = "trt")
head(cr$estimates)
plot(cr, trt.param = "trt")
```

Index