Package: tvrmst (via r-universe)

May 18, 2026

Type Package

Title Time-Varying Restricted Mean Survival Time from Survival Matrices

Version 0.0.6

Author Imad EL BADISY [aut, cre]

Maintainer Imad EL BADISY <elbadisyimad@gmail.com>

Description Utilities for restricted mean survival time (RMST) and time-varying restricted mean survival time quantities computed from survival curves provided on a time grid. The package is model-agnostic and accepts only a time vector and survival matrices, returning RMST-based quantities and bootstrap summaries. For restricted mean survival time methodology, see Royston and Parmar (2013) <doi:10.1186/1471-2288-13-152>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports stats

Suggests ggplot2, survival, testthat (>= 3.0.0), roxygen2 (>= 7.0.0)

Config/testthat/edition 3

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

Date/Publication 2026-03-11 15:19:28 UTC

RemoteUrl https://github.com/ielbadisy/tvrmst

RemoteRef HEAD

RemoteSha 9441577df3385f0f403221c991cd7415e357ed74

Contents

`as_survmat` *Construct a survival matrix object*

Description

Construct a survival matrix object

Usage

```
as_survmat(S, time, id = NULL, group = NULL)
```

Arguments

`S`	Numeric matrix of survival probabilities (rows = subjects, columns = time).
`time`	Numeric strictly increasing time grid with `length(time)` == `ncol(S)`.
`id`	Optional subject identifiers. Defaults to `seq_len(nrow(S))`.
`group`	Optional group labels with length `nrow(S)`.

Value

An object of class `"survmat"`.

Examples

```
time <- c(0, 1, 2)
S <- rbind(
  c(1.0, 0.8, 0.6),
  c(1.0, 0.7, 0.5)
)
x <- as_survmat(S, time, group = c("A", "B"))
x
```

as_survprob_matrix *Coerce survival predictions to a numeric matrix*

Description

Coerce survival predictions to a numeric matrix

Usage

```
as_survprob_matrix(pred, times = NULL)
```

Arguments

`pred`	Prediction output from a model.
`times`	Unused placeholder for API compatibility.

Value

Numeric matrix with rows as subjects and columns as times.

Examples

```
pred <- data.frame(
  t0 = c(1.0, 1.0),
  t1 = c(0.8, 0.7),
  t2 = c(0.6, 0.5)
)
as_survprob_matrix(pred)
```

bind_survmat *Row-bind survmat objects on a common time grid*

Description

Row-bind survmat objects on a common time grid

Usage

```
bind_survmat(...)
```

Arguments

`...`	One or more `survmat` objects.

Value

A combined `survmat`.

Examples

```
time <- c(0, 1, 2)
xA <- as_survmat(matrix(c(1.0, 0.9, 0.7), nrow = 1), time, group = "A")
xB <- as_survmat(matrix(c(1.0, 0.8, 0.6), nrow = 1), time, group = "B")
x <- bind_survmat(xA, xB)
nobs_survmat(x)
```

`boot_rmst_delta` *Bootstrap CI for dynamic RMST delta curve*

Description

Bootstrap CI for dynamic RMST delta curve

Usage

```
boot_rmst_delta(xA, xB, R = 300, conf = 0.95, seed = NULL)
```

Arguments

`xA`	A `survmat` object for arm A.
`xB`	A `survmat` object for arm B.
`R`	Number of bootstrap replicates.
`conf`	Confidence level in `(0, 1)`.
`seed`	Optional RNG seed.

Value

A list with point estimate and percentile confidence bands.

Examples

```
time <- c(0, 1, 2)
xA <- as_survmat(rbind(c(1.0, 0.9, 0.7), c(1.0, 0.8, 0.6)), time, group = c("A", "A"))
xB <- as_survmat(rbind(c(1.0, 0.95, 0.8), c(1.0, 0.85, 0.7)), time, group = c("B", "B"))
boot <- boot_rmst_delta(xA, xB, R = 10, seed = 1)
boot$estimate
```

`bootstrap_curve` *Generic bootstrap for curve-valued estimators*

Description

Generic bootstrap for curve-valued estimators

Usage

```
bootstrap_curve(est_fun, R = 300, conf = 0.95, seed = NULL, keep_reps = FALSE)
```

Arguments

`est_fun`	Function of one argument `r`; called with `NULL` for the point estimate and with integers `1:R` for bootstrap replicates.
`R`	Number of replicates.
`conf`	Confidence level in `(0, 1)`.
`seed`	Optional RNG seed.
`keep_reps`	If `TRUE`, include replicate matrix in output.

Value

A list with `estimate`, `lo`, `hi`, and `conf`.

Examples

```
vals <- c(1, 2, 4, 8)
boot <- bootstrap_curve(
  function(r) {
    if (is.null(r)) {
      mean(vals)
    } else {
      mean(sample(vals, replace = TRUE))
    }
  },
  R = 20,
  seed = 1
)
boot$estimate
```

nobs_survmat *Number of rows in a survmat*

Description

Number of rows in a survmat

Usage

```
nobs_survmat(x)
```

Arguments

x	A survmat object.

Value

Number of subjects.

Examples

```
time <- c(0, 1, 2)
S <- rbind(
  c(1.0, 0.8, 0.6),
  c(1.0, 0.7, 0.5)
)
x <- as_survmat(S, time)
nobs_survmat(x)
```

plot_boot_curve *Plot bootstrap estimate with confidence ribbon*

Description

Plot bootstrap estimate with confidence ribbon

Usage

```
plot_boot_curve(
  boot,
  grid = NULL,
  title = "Bootstrap curve",
  xlab = "t",
  ylab = "estimate",
  x_scale = 1,
  y_scale = 1,
  x_unit = NULL,
  y_unit = NULL
)
```

Arguments

boot	List returned by bootstrap_curve() or boot_rmst_delta().
grid	Optional x-axis grid. If omitted, uses boot$time.
title	Plot title.
xlab	X-axis label.
ylab	Y-axis label.
x_scale	Positive divisor applied to the x-axis values before plotting.
y_scale	Positive divisor applied to the y-axis values before plotting.
x_unit	Optional unit label appended to the x-axis title.
y_unit	Optional unit label appended to the y-axis title.

Value

A ggplot object.

Examples

```
time <- c(0, 1, 2)
xA <- as_survmat(rbind(c(1.0, 0.9, 0.7), c(1.0, 0.8, 0.6)), time, group = c("A", "A"))
xB <- as_survmat(rbind(c(1.0, 0.95, 0.8), c(1.0, 0.85, 0.7)), time, group = c("B", "B"))
boot <- boot_rmst_delta(xA, xB, R = 10, seed = 1)
plot_boot_curve(boot)
```

---

plot_delta_curve *Plot a delta curve*

Description

Plot a delta curve

Usage

```
plot_delta_curve(
  grid,
  delta,
  title = "Delta curve",
  xlab = "t",
  ylab = "Delta",
  x_scale = 1,
  y_scale = 1,
  x_unit = NULL,
  y_unit = NULL
)
```

Arguments

grid	X-axis grid.
delta	Y values.
title	Plot title.
xlab	X-axis label.
ylab	Y-axis label.
x_scale	Positive divisor applied to the x-axis values before plotting.
y_scale	Positive divisor applied to the y-axis values before plotting.
x_unit	Optional unit label appended to the x-axis title.
y_unit	Optional unit label appended to the y-axis title.

Value

A ggplot object.

Examples

```
time <- c(0, 1, 2)
xA <- as_survmat(rbind(c(1.0, 0.9, 0.7), c(1.0, 0.8, 0.6)), time, group = c("A", "A"))
xB <- as_survmat(rbind(c(1.0, 0.95, 0.8), c(1.0, 0.85, 0.7)), time, group = c("B", "B"))
d <- rmst_delta(xA, xB)
plot_delta_curve(d$time, d$delta)
```

---

plot_rmst_individual_by_group

Plot individual dynamic RMST curves by group

Description

Plot individual dynamic RMST curves by group

Usage

```
plot_rmst_individual_by_group(
  res,
  group,
  n_show_per_group = 30,
  title = "Individual dynamic RMST by group",
  x_scale = 1,
  y_scale = 1,
  x_unit = NULL,
  y_unit = NULL
)
```

Arguments

res	Result from rmst_dynamic() containing individual and time.
group	Group vector aligned with rows of res$individual.
n_show_per_group	Maximum number of individual curves shown per group.
title	Plot title.
x_scale	Positive divisor applied to the x-axis values before plotting.
y_scale	Positive divisor applied to the y-axis values before plotting.
x_unit	Optional unit label appended to the x-axis title.
y_unit	Optional unit label appended to the y-axis title.

Value

A ggplot object.

Examples

```
time <- c(0, 1, 2)
S <- rbind(
  c(1.0, 0.9, 0.7),
  c(1.0, 0.8, 0.6),
  c(1.0, 0.95, 0.8),
  c(1.0, 0.85, 0.7)
)
grp <- c("A", "A", "B", "B")
x <- as_survmat(S, time, group = grp)
res <- rmst_dynamic(x)
plot_rmst_individual_by_group(res, grp, n_show_per_group = 2)
```

---

plot_rmst_two_arms *Plot mean RMST curves for two arms*

Description

Plot mean RMST curves for two arms

Usage

```
plot_rmst_two_arms(
  xA,
  xB,
  labels = c("Arm A", "Arm B"),
  title = NULL,
  xlab = "Time (tau)",
  ylab = "RMST(tau)",
```

```
  curve_colors = c("#1B6CA8", "#D95F02"),
  x_scale = 1,
  y_scale = 1,
  x_unit = NULL,
  y_unit = NULL
)
```

Arguments

xA	A survmat object for arm A.
xB	A survmat object for arm B.
labels	Two legend labels.
title	Plot title. Use NULL to omit it.
xlab	X-axis label.
ylab	Y-axis label.
curve_colors	Colors for the two curves.
x_scale	Positive divisor applied to the x-axis values before plotting.
y_scale	Positive divisor applied to the y-axis values before plotting.
x_unit	Optional unit label appended to the x-axis title.
y_unit	Optional unit label appended to the y-axis title.

Value

A ggplot object.

Examples

```
time <- c(0, 1, 2)
xA <- as_survmat(rbind(c(1.0, 0.9, 0.7), c(1.0, 0.8, 0.6)), time, group = c("A", "A"))
xB <- as_survmat(rbind(c(1.0, 0.95, 0.8), c(1.0, 0.85, 0.7)), time, group = c("B", "B"))
plot_rmst_two_arms(xA, xB, labels = c("Control", "Treatment"))
```

---

rmst_delta *Dynamic RMST difference between two arms*

Description

Computes Delta(tau) = RMST_B(tau) - RMST_A(tau).

Usage

```
rmst_delta(xA, xB, tau = NULL)
```

Arguments

xA	A survmat object for arm A.
xB	A survmat object for arm B.
tau	Optional numeric horizons for interpolation.

Value

A list with RMST curves for both arms and the delta curve.

Examples

```
time <- c(0, 1, 2)
xA <- as_survmat(rbind(c(1.0, 0.9, 0.7), c(1.0, 0.8, 0.6)), time, group = c("A", "A"))
xB <- as_survmat(rbind(c(1.0, 0.95, 0.8), c(1.0, 0.85, 0.7)), time, group = c("B", "B"))
rmst_delta(xA, xB, tau = 1.5)$at_tau
```

---

rmst_dynamic *Dynamic RMST trajectories*

Description

Computes RMST_i(tau) = integral_0^tau S_i(u) du over the provided grid.

Usage

```
rmst_dynamic(x, tau = NULL, by = x$group)
```

Arguments

x	A survmat object.
tau	Optional numeric horizons for interpolation.
by	Optional grouping variable. Defaults to x$group.

Value

A list with individual and mean RMST curves.

Examples

```
time <- c(0, 1, 2)
S <- rbind(
  c(1.0, 0.8, 0.6),
  c(1.0, 0.7, 0.5)
)
x <- as_survmat(S, time, group = c("A", "A"))
res <- rmst_dynamic(x, tau = c(0.5, 1.5))
res$mean
res$at_tau
```

tvrmst *tvrmst: Matrix-first dynamic restricted mean survival time utilities*

Description

Tools for dynamic restricted mean survival time estimation, restricted mean survival time contrasts, bootstrap uncertainty, and plotting from survival probability matrices evaluated on a common time grid.

Index