Package 'survC'

October 29, 2025

Title Survival Model Validation Utilities

Version 0.1.0

Author Minhyuk Kim [aut, cre] (ORCID: <https://orcid.org/0009-0000-8083-5099>)

Maintainer Minhyuk Kim <mhkim@zarathu.com>

Description Provides helper functions to compute linear predictors, time-dependent ROC curves, and Harrell's concordance index for Cox proportional hazards models as described in Therneau (2024) <https://CRAN.R-project.org/package=survival>, Therneau and Grambsch (2000, ISBN:0-387-98784-3), Hung and Chiang (2010) <doi:10.1002/cjs.10046>, Uno et al. (2007) <doi:10.1198/016214507000000149>, Blanche, Dartigues, and Jacqmin-Gadda (2013) <doi:10.1002/sim.5958>, Blanche, Latouche, and Viallon (2013) <doi:10.1007/978-1-4614-8981-8_11>, Harrell et al. (1982) <doi:10.1001/jama.1982.03320430047030>, Peto and Peto (1972) <doi:10.2307/2344317>, Schemper and Uno et al. (2011) <doi:10.1002/sim.4154>.

Imports survival, stats, timeROC, officer, rvg

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

URL https://newjoseph.github.io/survC/

Suggests magrittr, jstable, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2025-10-29 19:40:02 UTC

Contents

`calc_risk_score` *Compute risk scores from a fitted survival model*

Description

This helper wraps `stats::predict()` for `coxph` objects so that package users can easily obtain linear predictors (default) or risk scores to feed into downstream metrics such as time-dependent ROC or Harrell's C-index.

Usage

```
calc_risk_score(model, data = NULL, type = "lp", ...)
```

Arguments

`model`	A fitted `coxph` object.
`data`	Optional dataset on which to score the model. Defaults to the training data stored within `model`.
`type`	Scale of the predictions to return. Either `"lp"` (linear predictor, the default) or `"risk"`. If `NULL` or omitted, `"lp"` is used.
`...`	Additional arguments passed to `stats::predict()`.

Value

A numeric vector containing the requested risk scores.

Examples

```
if (requireNamespace("survival", quietly = TRUE)) {
  fit <- survival::coxph(survival::Surv(time, status) ~ age, data = survival::lung)
  # Linear predictor on the training data
  calc_risk_score(fit)

  # Risk scale predictions on new data
  calc_risk_score(fit, survival::lung, type = "risk")
}
```

cindex_calc *Calculate Harrell's C-index with 95% CI*

Description

Calculate Harrell's C-index with 95% CI

Usage

```
cindex_calc(model, newdata = NULL, digits = 3)
```

Arguments

model	a 'coxph' object
newdata	optional validation dataset
digits	number of decimal places for rounding (default 3).

Value

numeric vector of C-index (lower, upper)

Examples

```
library(survival)
fit <- coxph(Surv(time, status) ~ age + sex, data = lung)
cindex_calc(fit)
```

tdroc_calc *Calculate time-dependent ROC and AUC*

Description

Calculate time-dependent ROC and AUC

Usage

```
tdroc_calc(time, status, marker, times)
```

Arguments

time	Survival time vector
status	Event indicator (1 = event, 0 = censor)
marker	Risk score or linear predictor
times	Vector of time points (e.g., c(365, 730, 1095))

Value

A data.frame with AUCs for each time

`validation_report` *Generate survival model validation report*

Description

Generate survival model validation report

Usage

```
validation_report(
  train_data,
  val_data,
  model,
  time_col,
  status_col,
  times = c(365, 730, 1095),
  time_unit = "days",
  output = NULL
)
```

Arguments

`train_data`	training dataset containing survival outcomes.
`val_data`	validation dataset containing survival outcomes.
`model`	fitted 'coxph'
`time_col`	name of the survival time column present in both datasets
`status_col`	name of the event indicator column present in both datasets
`times`	follow-up timepoints
`time_unit`	character label for the time axis (default = "days")
`output`	file path (.pptx or .html). Defaults to a temporary file.

Value

Invisibly returns the file path containing the validation report

Index