

Package ‘PAmeasures’

February 12, 2025

Type Package

Title Prediction and Accuracy Measures for Nonlinear Models and for
Right-Censored Time-to-Event Data

Version 0.1.0

Date 2018-01-18

Author Xiaoyan Wang, Gang Li

Maintainer Xiaoyan Wang<xywang@ucla.edu>

Description We propose a pair of summary measures for the predictive power of a prediction function based on a regression model. The regression model can be linear or nonlinear, parametric, semi-parametric, or nonparametric, and correctly specified or mis-specified. The first measure, R-squared, is an extension of the classical R-squared statistic for a linear model, quantifying the prediction function's ability to capture the variability of the response. The second measure, L-squared, quantifies the prediction function's bias for predicting the mean regression function. When used together, they give a complete summary of the predictive power of a prediction function. Please refer to Gang Li and Xiaoyan Wang (2016) <[arXiv:1611.03063](#)> for more details.

Depends R (>= 3.1)

Imports survival, stats

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests testthat

NeedsCompilation no

Repository CRAN

Date/Publication 2018-01-22 10:06:41 UTC

Contents

pam.performance_metrics	2
pam.predicted_performance_metric	3
Index	5

pam.performance_metrics

Performance Metrics for Survival Analysis Models

Description

This function computes a comprehensive set of performance metrics for survival analysis models. It provides metrics such as `R_square`, `L_square`, `Pseudo_R`, Harrell's C, Uno's C, `R_sph` (distance-based estimator for survival predictive accuracy), `R_sh`, Brier Score, and Time-dependent AUC. Users can specify particular metrics and model types, enabling tailored performance evaluation for various survival models.

Usage

```
pam.performance_metrics(
  data,
  time_var,
  status_var,
  covariates,
  model = "coxph",
  metrics = "all",
  newdata = NULL,
  t_star = NULL
)
```

Arguments

<code>data</code>	A data frame containing the survival data.
<code>time_var</code>	The name of the time variable in 'data' indicating survival time.
<code>status_var</code>	The name of the status variable in 'data' indicating event occurrence.
<code>covariates</code>	A character vector of covariate names to include in the model.
<code>model</code>	A character string or vector specifying the model types to fit (e.g., "coxph", "exp", "lognormal", "weibull"). Default is "coxph" to fit all models.
<code>metrics</code>	A character string or vector specifying the metrics to compute. Default is "all" to compute all available metrics.
<code>newdata</code>	(Optional) A data frame containing validation data. If 'NULL', the function uses the same data as 'data' for model evaluation.
<code>t_star</code>	A positive numeric value specifying the time point at which the Brier score is calculated.

Value

A data frame containing the selected model's performance metrics.

Examples

```

library(PAmeasures)
library(survival)
library(rms)
library(dplyr)

Use Mayo Clinic Primary Biliary Cirrhosis Data
data(pbc)
pbc <- pbc %>%
  filter(is.na(trt)==F) %>%
  mutate(log_albumin = log(albumin),
         log_bili = log(bili),
         log_protime = log(protime),
         status = ifelse(status==2, 1, 0))
time_var <- "time"
status_var <- "status"
covariates <- c("age", "log_albumin", "log_bili", "log_protime", "edema")
Call the function with all metrics and all models
results <- pam.performance_metrics(data = pbc,
                                time_var = time_var,
                                status_var = status_var,
                                covariates = covariates
                                )
results2 <- pam.performance_metrics(data = pbc,
                                time_var = time_var,
                                status_var = status_var,
                                covariates = covariates,
                                model = c("lognormal", "weibull"),
                                metrics = c("R_square", "L_square", "Brier Score")
                                )

```

pam.predicted_performance_metric

Summary of Performance Metrics for Predicted Survival Data

Description

This function computes and summarizes various performance metrics for survival data using predicted values and observed survival data. Users can specify the desired metrics or compute all available metrics by default.

Usage

```

pam.predicted_performance_metric(
  predicted_data,
  survival_time,
  metric,
  status = NULL,
  tau = NULL,
  t_star = NULL,
  start_time = NULL
)

```

Arguments

<code>predicted_data</code>	A numeric vector of predicted survival probabilities or scores.
<code>survival_time</code>	A numeric vector of observed survival times.
<code>status</code>	A numeric or logical vector indicating event status (1 for event, 0 for censoring).
<code>tau</code>	An optional numeric value for restricted time horizon. Default is NULL.
<code>t_star</code>	An optional numeric value specifying the time point for certain metrics. Default is NULL.
<code>metrics</code>	A character vector specifying the desired metrics to compute. Options include: <ul style="list-style-type: none"> • <code>"R_square"</code>: R-squared metric. • <code>"L_square"</code>: L-squared metric. • <code>"Pesudo_R"</code>: Pseudo-R-squared metric. • <code>"Harrells_C"</code>: Harrell's Concordance Index. • <code>"Unos_C"</code>: Uno's Concordance Index. • <code>"R_sph"</code>: Explained variation (<code>R_sph</code>). • <code>"R_sh"</code>: Explained variation (<code>R_sh</code>). • <code>"Brier_Score"</code>: Brier Score. • <code>"Time_Dependent_Auc"</code>: Time-dependent AUC.

Value

A data frame summarizing the requested performance metrics.

Examples

```
predicted_data <- c(0.8, 0.6, 0.4, 0.2)
survival_time <- c(5, 8, 3, 10)
status <- c(1, 0, 1, 1)
metric <- "R_square"
pam.predicted_performance_metric(predicted_data, survival_time, metric, status)
```

Index

pam.performance_metrics, [2](#)
pam.predicted_performance_metric, [3](#)