Package 'landpred'

May 8, 2026

Type Package

Title Landmark Prediction of a Survival Outcome

Version 2.0

Description Nonparametric methods for landmark prediction of long-term survival outcomes, incorporating covariate and short-term event information. The package supports the construction of flexible varying-coefficient models that use discrete covariates, as well as multiple continuous covariates. The goal is to improve prediction accuracy when censored short-term events are available as predictors, using robust nonparametric procedures that do not require correct model specification and avoid restrictive parametric assumptions found in alternative methods. More information on these methods can be found in Parast et al. 2012 <doi:10.1080/01621459.2012.721281>, Parast et al. 2011 <doi:10.1002/bimj.201000150>, and Parast and Cai 2013 <doi:10.1002/sim.5776>. A tutorial for this package is available here: <https://www.laylaparast.com/landpred>.

License GPL

Imports survival, stats, quantreg, splines, sm, quantreg

NeedsCompilation no

Suggests testthat (>= 3.0.0)

Author Dylan Huynh [aut], Layla Parast [aut, cre]

Maintainer Layla Parast <parast@austin.utexas.edu>

Repository CRAN

Date/Publication 2025-12-15 15:10:02 UTC

Contents

`AUC.landmark` *Estimates the area under the ROC curve (AUC).*

Description

This function calculates the AUC given the data (truth) and corresponding estimated probabilities; uses a continuity correction.

Usage

```
AUC.landmark(t0, tau, data, weight=NULL)
```

Arguments

`t0`	The landmark time.
`tau`	The prediction window.
`data`	A data matrix where the first column is XL = min(TL, C) where TL is the time of the long term event, C is the censoring time; the second column is DL =1*(TL<C); the third column is the estimated probability P(TL<t0+tau | TL>t0).
`weight`	an optional weight to be incorporated in all estimation.

Value

`AUC.est`	Estimated AUC

References

Parast, Layla, Su-Chun Cheng, and Tianxi Cai. Incorporating short-term outcome information to predict long-term survival with discrete markers. Biometrical Journal 53.2 (2011): 294-307.

Examples

```
data(data_example_landpred)
t0=2
tau = 8

out = Prob.Null(t0=t0,tau=tau,data=data_example_landpred)

#get data with predictions
data_pred = out$data

#calculate training AUC
AUC.landmark(t0=t0,tau=tau, data = data_pred[,c("XL","DL","prob_est")])
```

`BS.landmark` *Estimates the Brier score.*

Description

This function calculates the Brier score given the data (truth) and corresponding estimated probabilities.

Usage

```
BS.landmark(t0, tau, data, weight=NULL)
```

Arguments

`t0`	The landmark time.
`tau`	The prediction window.
`data`	A data matrix where the first column is XL = min(TL, C) where TL is the time of the long term event, C is the censoring time; the second column is DL =1*(TL<C); the third column is the estimated probability P(TL<t0+tau | TL>t0).
`weight`	an optional weight to be incorporated in all estimation.

Value

`Brier.score`	Estimated Brier score

References

Parast, Layla, Su-Chun Cheng, and Tianxi Cai. Incorporating short-term outcome information to predict long-term survival with discrete markers. Biometrical Journal 53.2 (2011): 294-307.

Examples

```
data(data_example_landpred)
t0=2
tau = 8

out = Prob.Null(t0=t0,tau=tau,data=data_example_landpred)

#get data with predictions
data_pred = out$data

#calculate training BS
BS.landmark(t0=t0,tau=tau, data = data_pred[,c("XL","DL","prob_est")])
```

`coef.landpred_model_continuous` *Extract Coefficients from Landpred Continuous Model*

Description

Extracts coefficients. If `t_s` is provided, it fits the short-term GLM and returns its coefficients.

Usage

```
## S3 method for class 'landpred_model_continuous'
coef(object, t_s = NULL, ...)
```

Arguments

`object`	A landpred_model_continuous object.
`t_s`	Optional short-term event time.
`...`	Additional arguments.

Value

A named vector of coefficients.

cumsum2 *Helper function*

Description

Helper function; should not be called directly by user.

Usage

```
cumsum2(mydat)
```

Arguments

mydat	mydat

Value

out	matrix

Author(s)

Layla Parast

data_example_landpred *Hypothetical data to be used in examples.*

Description

Hypothetical data to be used in examples.

Usage

```
data(data_example_landpred)
```

Format

A data frame with 4868 observations on the following 5 variables.

- XL a numeric vector. XL = min(TL, C) where TL is the time of the long term event, C is the censoring time.
- DL a 0/1 vector. DL =1*(TL<C) where TL is the time of the long term event, C is the censoring time.
- XS a numeric vector. XS = min(TS, C) where TS is the time of the long term event, C is the censoring time.
- DS a 0/1 vector. DS =1*(TS<C) where TS is the time of the long term event, C is the censoring time.
- Z a 0/1 vector of discrete covariate values.

Examples

```
data(data_example_landpred)
```

`get_model` *Get Landpred Model*

Description

Creates a landpred model object for a specific landmark time and prediction window. Dispatches to continuous or discrete model creation based on the landpred object type.

Usage

```
get_model(landpred_obj, t0, tau, bw = NULL, transform = identity)
```

Arguments

`landpred_obj`	A landpred object.
`t0`	The landmark time.
`tau`	The prediction window.
`bw`	The bandwidth.
`transform`	Transformation function.

Value

A landpred_model object (continuous or discrete).

`Ghat.FUN` *Estimate Survival Function*

Description

Estimate Survival Function

Usage

```
Ghat.FUN(tt, data, type = "fl", weight.given)
```

Arguments

`tt`	Time points.
`data`	Data frame.
`type`	Type of estimator.
`weight.given`	Optional weights.

helper.si *Helper function for AUC.landmark*

Description

Helper function for AUC.landmark; should not be called directly by user.

Usage

```
helper.si(yy,FUN,Yi,Vi=NULL)
```

Arguments

yy	yy
FUN	FUN
Yi	Yi
Vi	Vi

Value

out	matrix

Author(s)

Layla Parast

Kern.FUN *Calculates kernel matrix*

Description

Calculates the kernel matrix

Usage

```
Kern.FUN(zz, zi, bw)
```

Arguments

zz	zz
zi	zi
bw	bandwidth

Value

the kernel matrix

Author(s)

Layla Parast

`landpred` *Create a Landpred Object*

Description

Parses the formula and data to create a landpred object used for landmark prediction. Call '?landpred.pacakge' for more information on the legacy API.

Usage

```
landpred(formula, data, discrete = FALSE, no.covariates = FALSE)
```

Arguments

`formula`	A formula object with a Surv object on the LHS and covariates on the RHS.
`data`	The data frame.
`discrete`	Logical, whether to use the discrete method (legacy).
`no.covariates`	Logical, whether there are covariates or not.

Value

A landpred_object.

Examples

```
library(landpred)
library(survival)

# Load example data
data(data_example_landpred)

# Define landmark time and prediction window
t0 <- 2
tau <- 8

# Create a landpred object using the formula interface
# The formula specifies: Long-term survival ~ Short-term survival + Covariates
# Note: The short-term event must be a Surv object
obj <- landpred(
  Surv(XL, DL) ~ Surv(XS, DS) + Z,
  data = data_example_landpred,
```

```
  discrete = FALSE
)

# 1. Optimize bandwidth (Optional but recommended)
# This uses cross-validation to find the optimal bandwidth for the short-term event
# We use log transformation for the time variable as it's often more appropriate
bw <- optimize_bandwidth(
  landpred_obj = obj,
  t0 = t0,
  tau = tau,
  lower = 0.5,
  upper = 5,
  transform = log
)

print(paste("Optimal bandwidth:", bw))

# 2. Fit the model
# We pass the optimized bandwidth and the transformation used
model <- get_model(
  landpred_obj = obj,
  t0 = t0,
  tau = tau,
  bw = bw,
  transform = log
)

print(model)
summary(model,t_s = 1)

# 3. Predict on new data
# For demonstration, we use the first 10 rows of the original data as "new data"
new_data <- data_example_landpred[1:10, ]

# The predict function expects a data frame with the same column names as used in the formula
probs <- predict(model, newdata = new_data)

print("Predicted probabilities:")
print(probs)
```

---

optimize_bandwidth *Optimize Bandwidth for Continuous Landpred Models*

Description

Selects the optimal bandwidth by minimizing the Mean Squared Error (MSE) using cross-validation.

Usage

```
optimize_bandwidth(
  landpred_obj,
  t0,
  tau,
  lower = 0.05,
  upper = 5,
  transform = identity,
  reps = 50,
  train_prop = 0.66
)
```

Arguments

landpred_obj	A landpred object.
t0	The landmark time.
tau	The prediction window.
lower	Lower bound for bandwidth search.
upper	Upper bound for bandwidth search.
transform	Transformation function for the short-term event (e.g., log). Default is identity.
reps	Number of cross-validation repetitions. Default is 50.
train_prop	Proportion of data used for training in each fold. Default is 0.66.

Value

The optimal bandwidth.

---

predict.landpred_model_continuous

Predict Method for Landpred Continuous Model

Description

Predicts the probability of the event occurring given new data.

Usage

```
## S3 method for class 'landpred_model_continuous'
predict(object, newdata = NULL, type = "response", ...)
```

Arguments

object	A landpred_model_continuous object.
newdata	New data frame containing covariates and short-term event info.
type	Type of prediction (default "response").
...	Additional arguments

Value

A vector of predicted probabilities.

`predict.landpred_model_discrete` *Predict Method for Discrete Landpred Model*

Description

Predicts probabilities using the discrete landpred model.

Usage

```
## S3 method for class 'landpred_model_discrete'
predict(object, newdata = NULL, ...)
```

Arguments

`object`	A landpred_model_discrete object.
`newdata`	Optional new data.
`...`	Additional arguments.

Value

Predicted probabilities.

`print.landpred_model_continuous` *Print Method for Landpred Continuous Model*

Description

Prints the continuous landpred model results.

Usage

```
## S3 method for class 'landpred_model_continuous'
print(x, ...)
```

Arguments

`x`	A landpred_model_continuous object.
`...`	Additional arguments.

print.landpred_model_discrete *Print Method for Discrete Landpred Model*

Description

Prints the discrete landpred model results.

Usage

```
## S3 method for class 'landpred_model_discrete'
print(x, ...)
```

Arguments

x	A landpred_model_discrete object.
...	Additional arguments.

print.landpred_object *Print Method for Landpred Object*

Description

Prints a summary of the landpred object.

Usage

```
## S3 method for class 'landpred_object'
print(x, ...)
```

Arguments

x	A landpred_object.
...	Additional arguments.

Prob.Covariate *Calculate Probability with Covariate Information*

Description

Calculates the probability of the event occurring before `t0 + tau`, given survival up to `t0`, using a single covariate.

Usage

```
Prob.Covariate(t0, tau, data, weight = NULL, short = TRUE, newdata = NULL)
```

Arguments

`t0`	The landmark time.
`tau`	The prediction window.
`data`	The data frame for training.
`weight`	Optional weights.
`short`	Logical, whether the covariate is short-term.
`newdata`	Dataframe of new data for prediction.

Value

A landpred_result object.

Prob.Covariate.ShortEvent *Calculate Probability with Short Event Information*

Description

Calculates the probability of the event occurring before `t0 + tau`, given survival up to `t0`, using information on a short-term event.

Usage

```
Prob.Covariate.ShortEvent(
  t0,
  tau,
  data,
  weight = NULL,
  bandwidth = NULL,
  newdata = NULL
)
```

Arguments

t0	The landmark time.
tau	The prediction window.
data	The data frame.
weight	Optional weights.
bandwidth	Bandwidth for kernel smoothing.
newdata	Optional new data for prediction.

Value

A landpred_result object.

Prob.Null *Calculate Probability with No Information*

Description

Calculates the probability of the event occurring before t0 + tau, given survival up to t0, without using any covariate information.

Usage

```
Prob.Null(t0, tau, data, weight = NULL, newdata = NULL)
```

Arguments

t0	The landmark time.
tau	The prediction window.
data	The data frame.
weight	Optional weights.
newdata	Optional new data for prediction.

Value

A landpred_result object.

summary.landpred_model_continuous

Summary Method for Landpred Continuous Model

Description

Prints a summary of the model, including coefficients and standard errors.

Usage

```
## S3 method for class 'landpred_model_continuous'
summary(object, t_s = NULL, ...)
```

Arguments

object	A landpred_model_continuous object.
t_s	Optional short-term event time.
...	Additional arguments.

summary.landpred_object

Summary Method for Landpred Object

Description

Prints a detailed summary of the landpred object.

Usage

```
## S3 method for class 'landpred_object'
summary(object, ...)
```

Arguments

object	A landpred_object.
...	Additional arguments.

VTM *Helper function, repeats a row.*

Description

This function creates a matrix that repeats vc, dm times where each row is equal to the vc vector.

Usage

```
VTM(vc, dm)
```

Arguments

vc	the vector to repeat.
dm	number of rows.

Value

a matrix that repeats vc, dm times where each row is equal to the vc vector

Wi.FUN *Computes the inverse probability of censoring weights for a specific t0 and tau*

Description

Computes the inverse probability of censoring weights for a specific t0 and tau i.e. this computes I(t0 < XL < t0+tau)*DL/G(XL) + I(XL>t0+tau)/G(t0+tau) where XL = min(TL, C), TL is the time of the long term event, C is the censoring time, DL =1*(TL<C) and G() is the estimate survival probability for censoring estimated using the Kaplan Meier estimator

Usage

```
Wi.FUN(data, t0, tau, weight.given = NULL)
```

Arguments

data	n by k matrix, where k>= 2. A data matrix where the first column is XL = min(TL, C) where TL is the time of the long term event, C is the censoring time, and the second column is DL =1*(TL<C)
t0	the landmark time..
tau	the residual survival time for which probabilities are calculated.
weight.given	an optional weight to be incorporated in estimation of this weight

Value

Inverse probability of censoring weight.

Author(s)

Layla Parast

Examples

```
data(data_example_landpred)
t0=2
tau = 8

W2i <- Wi.FUN(data_example_landpred[,1],data = data_example_landpred[,c(1:2)],t0=t0,tau=tau)
```

Index