Package 'landpred'

November 16, 2021

Type Package

Title Landmark Prediction of a Survival Outcome

Version 1.1

Author Layla Parast

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

Description Provides functions for landmark prediction of a survival outcome incorporating covariate and short-term event information. For more information about landmark prediction please see: 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, <doi:10.1002/bimj.201000150>.

License GPL

Imports survival

NeedsCompilation no

Repository CRAN

Date/Publication 2021-11-16 17:20:02 UTC

R topics documented:

`landpred-package` *Landmark Prediction of a Survival Outcome*

Description

This package includes functions for landmark prediction of a survival outcome incorporating covariate and short-term event information. For more information about landmark prediction please see: 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.

Details

Package:	landpred
Type:	Package
Version:	1.0
License:	GPL

Author(s)

Layla Parast

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

####Landmark prediction with no covariate or short term information
Prob.Null(t0=t0,tau=tau,data=data_example_landpred)
out = Prob.Null(t0=t0,tau=tau,data=data_example_landpred)
out$Prob
out$data
```

```
newdata = matrix(c(1,1,3,0,4,1,10,1,11,0), ncol = 2, byrow=TRUE)
out = Prob.Null(t0=t0,tau=tau,data=data_example_landpred,newdata=newdata)
out$Prob
out$newdata

#Landmark prediction with covariate information only
Prob.Covariate(t0=t0,tau=tau,data=data_example_landpred)
out = Prob.Covariate(t0=t0,tau=tau,data=data_example_landpred)
out$Prob
out$data

newdata = matrix(c(1,1,1, 3,0,1, 4,1,1, 10,1,0, 11,0,1), ncol = 3, byrow=TRUE)
out = Prob.Covariate(t0=t0,tau=tau,data=data_example_landpred,newdata=newdata)
out$Prob
out$newdata

#Landmark prediction with covariate information and short term event information
#note: computationally intensive commands below
#Prob.Covariate.ShortEvent(t0=t0,tau=tau,data=data_example_landpred)
#out = Prob.Covariate.ShortEvent(t0=t0,tau=tau,data=data_example_landpred)
#out$data
#data.plot = out$data
#plot(data.plot$XS[data.plot$Z ==1], data.plot$Probability[data.plot$Z ==1],
#pch = 20, xlim = c(0,t0))
#points(data.plot$XS[data.plot$Z ==0], data.plot$Probability[data.plot$Z ==0],
#pch = 20, col = 2)

newdata = matrix(c(1,1,0.5,1,0,
3,0,1,1,1,
4,1,1.5,1,0,
10,1,5,1,0,
11,0,11,0,1), ncol = 5, byrow=TRUE)
#note: computationally intensive command below
#out=Prob.Covariate.ShortEvent(t0=t0,tau=tau,data=data_example_landpred,newdata=newdata)
#out$newdata
```

`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, short = TRUE, weight=NULL)
```

Arguments

`t0`	the landmark time.
`tau`	the residual survival time of interest.
`data`	n by k matrix, where k = 4 or 6. 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), the second to last column is the covariate vector (can be NULL) and the last column is the estimated probability P(TL<t0+tau | TL>t0).
`short`	logical value indicating whether data includes short term event information. Should be TRUE if short term XS and DS are includes as third and fourth columns of data matrix, FALSE if not. Default is TRUE.
`weight`	an optional weight to be incorporated in all estimation.

Value

`AUC.est`	Estimated AUC

Author(s)

Layla Parast

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

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

AUC.landmark(t0=t0,tau=tau, data = out$data)
```

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, short = TRUE, weight=NULL)
```

Arguments

`t0`	the landmark time.
`tau`	the residual survival time of interest.
`data`	n by k matrix, where k = 4 or 6. 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), the second to last column is the covariate vector (can be NULL) and the last column is the estimated probability P(TL<t0+tau | TL>t0).
`short`	logical value indicating whether data includes short term event information. Should be TRUE if short term XS and DS are includes as third and fourth columns of data matrix, FALSE if not. Default is TRUE.
`weight`	an optional weight to be incorporated in all estimation.

Value

`Brier.score`	Estimated Brier score

Author(s)

Layla Parast

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

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

BS.landmark(t0=t0,tau=tau, data = out$data)
```

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)
```

Ghat.FUN *Calculates the Kaplan Meier survival probability for censoring*

Description

Calculates the survival probability for censoring i.e. P(C > tt) where C is censoring; used in inverse probability of censoring weights (IPCW). This function is called by Wi.FUN; this function should not be called on its own.

Usage

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

Arguments

tt	the time (or vector of times) at which the survival probability should be estimated.
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)
type	type sent to survfit function, default is "fl".
weight.given	a weight to be used in estimation.

Value

survival probability for censoring at time tt

Author(s)

Layla Parast

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

This calculates the kernel matrix needed for estimating the probability incorporating short term event information

Usage

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

Arguments

zz	zz
zi	zi
bw	bandwidth

Value

the kernel matrix

Author(s)

Layla Parast

`mse.BW` *Helper function for optimize.mse.BW.*

Description

Helper function for optimize.mse.BW.

Usage

```
mse.BW(data, t0,tau,h, folds = 3,reps=2)
```

Arguments

`data`	n by 5 matrix. 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), the third column is XS = min(TS, C) where TS is the time of the short term event, C is the censoring time, the fourth column is DS =1*(TS<C), and the fifth column is the covariate. These are the data used to calculate the estimated probability.
`t0`	the landmark time.
`tau`	the residual survival time of interest.
`h`	bandwidth
`folds`	Number of folds wanted for K-fold cross-validation. Default is 3.
`reps`	Number of repitions wanted of K-fold cross-validation. Default is 2.

Value

mean of MSE

Author(s)

Layla Parast

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.

`optimize.mse.BW` *Calculates initial optimal bandwidth.*

Description

Calculates initial optimal bandwidth with respect to mean squared error using K-fold cross-validation.

Usage

```
optimize.mse.BW(data, t0,tau,h.grid=seq(.01,2,length=50), folds=3, reps=2)
```

Arguments

`data`	n by 5 matrix. 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), the third column is XS = min(TS, C) where TS is the time of the short term event, C is the censoring time, the fourth column is DS =1*(TS<C), and the fifth column is the covariate. These are the data used to calculate the estimated probability.
`t0`	the landmark time.
`tau`	the residual survival time of interest.
`h.grid`	The grid of possible bandwidths that the user would like the function to search through. Default is h.grid = seq(.01,2,length=50).
`folds`	Number of folds wanted for K-fold cross-validation. Default is 3.
`reps`	Number of repitions wanted of K-fold cross-validation. Default is 2.

Value

`h`	Selected bandwidth.

Author(s)

Layla Parast

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.

`Prob.Covariate` *Estimates P(TL <t0+tau | TL > t0, Z), i.e. given discrete covariate.*

Description

This function calculates the probability that the an individual has the event of interest before t0 + tau given the discrete covariate and given the event has not yet occurred and the individual is still at risk at time t0; this estimated probability does not incorporate any information about the short term event information.

Usage

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

Arguments

`t0`	the landmark time.
`tau`	the residual survival time for which probabilities are calculated. Specifically, this function estimates the probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0.
`data`	n by k matrix, where k =3 or k=5. 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). If short term event information is included in this dataset then the third column is XS = min(TS, C) where TS is the time of the short term event, C is the censoring time, and the fourth column is DS =1*(TS<C), and the fifth column is the covariate. If short term event information is not included then the third column is the covariates (see "short" parameter). These are the data used to calculate the estimated probabilities.
`weight`	an optional weight to be incorporated in all estimation.
`short`	logical value indicating whether data includes short term event information. Should be TRUE if short term XS and DS are includes as third and fourth columns of data matrix meaning that the covariates is in the fifth column, FALSE if not meaning that the covariate is in the third column. Default is TRUE.
`newdata`	n by k matrix, where k =3 or k=5. 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), and the last column (either 3rd or 5th) contains covariate values. Predicted probabilities are estimated for these data.

Value

`Prob`	matrix of estimated probability for each value of the covariate; first column shows all covariate values and second column contains predicted probability at that covariate value

data	the data matrix with an additional column with the estimated individual probabilities; note that the predicted probability is NA if TL <t0 since it is only defined for individuals with TL> t0
newdata	the newdata matrix with an additional column with the estimated individual probabilities; note that the predicted probability is NA if TL <t0 since it is only defined for individuals with TL> t0; if newdata is not supplied then this returns NULL

Author(s)

Layla Parast

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
Prob.Covariate(t0=t0,tau=tau,data=data_example_landpred)

out = Prob.Covariate(t0=t0,tau=tau,data=data_example_landpred)
out$Prob
out$data

newdata = matrix(c(1,1,1, 3,0,1, 4,1,1, 10,1,0, 11,0,1), ncol = 3, byrow=TRUE)
out = Prob.Covariate(t0=t0,tau=tau,data=data_example_landpred,newdata=newdata)
out$Prob
out$newdata
```

Prob.Covariate.ShortEvent

Estimates P(TL <t0+tau | TL > t0, Z, min(TS, t0), I(TS<=t0)), i.e. given discrete covariate and TS information.

Description

This function calculates the probability that the an individual has the event of interest before t0 + tau given the discrete covariate, given short term event information, and given the event has not yet occurred and the individual is still at risk at time t0.

Usage

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

Arguments

`t0` the landmark time.

`tau` the residual survival time for which probabilities are calculated. Specifically, this function estimates the probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0.

`data` n by 5 matrix. 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), the third column is XS = min(TS, C) where TS is the time of the short term event, C is the censoring time, the fourth column is DS =1*(TS<C), and the fifth column is the covariate. These are the data used to calculate the estimated probability.

`weight` a weight to be incorporated in all estimation.

`bandwidth` an optional bandwidth to be used in kernel smoothing; is not provided then function calculates an appropriate bandwidth using bw.nrd and then undersmoothing with c = .10 (See reference)

`newdata` an optional n by 5 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), the third column is XS = min(TS, C) where TS is the time of the short term event, C is the censoring time, the fourth column is DS =1*(TS<C), and the fifth column is the covariate. Predicted probabilities are estimated for these data.

Value

`data` the data matrix with an additional column with the estimated individual probabilities; note that the predicted probability is NA if TL <t0 since it is only defined for individuals with TL> t0

`newdata` the newdata matrix with an additional column with the estimated individual probabilities; note that the predicted probability is NA if TL <t0 since it is only defined for individuals with TL> t0; if newdata is not supplied then this returns NULL

Author(s)

Layla Parast

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
```

```
#note: computationally intensive command below
#Prob.Covariate.ShortEvent(t0=t0,tau=tau,data=data_example_landpred)

#out = Prob.Covariate.ShortEvent(t0=t0,tau=tau,data=data_example_landpred)
#out$data
#data.plot = out$data
#plot(data.plot$XS[data.plot$Z ==1], data.plot$Probability[data.plot$Z ==1],
#pch = 20, xlim = c(0,t0))
#points(data.plot$XS[data.plot$Z ==0], data.plot$Probability[data.plot$Z ==0],
#pch = 20, col = 2)

newdata = matrix(c(1,1,0.5,1,0,
3,0,1,1,1,
4,1,1.5,1,0,
10,1,5,1,0,
11,0,11,0,1), ncol = 5, byrow=TRUE)
#note: computationally intensive command below
#out = Prob.Covariate.ShortEvent(t0=t0,tau=tau,data=data_example_landpred,newdata=newdata)
#out$newdata
```

`Prob.Null` *Estimates P(TL <t0+tau | TL > t0).*

Description

This function calculates the probability that an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0; this estimated probability does not incorporate any information about the covariate or short term event information.

Usage

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

Arguments

`t0`	the landmark time.
`tau`	the residual survival time for which probabilities are calculated. Specifically, this function estimates the probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0.
`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). These are the data used to calculate the estimated probability.
`weight`	an optional weight to be incorporated in all estimation.

`newdata` an optional 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). Predicted probabilities are estimated for these data.

Value

`Prob` Estimated probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0; this estimated probability does not incorporate any information about the covariate or short term event information.

`data` the data matrix with an additional column with the estimated individual probabilities; note that the predicted probability is NA if TL <t0 since it is only defined for individuals with TL> t0

`newdata` the newdata matrix with an additional column with the estimated individual probabilities; note that the predicted probability is NA if TL <t0 since it is only defined for individuals with TL> t0; if newdata is not supplied then this returns NULL

Author(s)

Layla Parast

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

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

newdata = matrix(c(1,1,3,0,4,1,10,1,11,0), ncol = 2, byrow=TRUE)
out = Prob.Null(t0=t0,tau=tau,data=data_example_landpred,newdata=newdata)
out$Prob
out$newdata
```

`Prob2` *Estimates P(TL <t0+tau | TL > t0, Z, TS>t0).*

Description

This function calculates the probability that the an individual has the event of interest before t0 + tau given the discrete covariate, given the short term event has not yet occurred by t0, and given the long term event has not yet occurred and the individual is still at risk at time t0. This function is called by Prob.Covariate.ShortEvent; this function should not be called on its own.

Usage

```
Prob2(t0, tau, data, covariate.value, weight = NULL)
```

Arguments

- `t0` — the landmark time.
- `tau` — the residual survival time for which probabilities are calculated. Specifically, this function estimates the probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0.
- `data` — n by 5 matrix. 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), the third column is log(XS) = log(min(TS, C)) where TS is the time of the short term event, C is the censoring time, the fourth column is DS =1*(TS<C), and the fifth column is the covariate. These are the data used to calculate the estimated probability.
- `covariate.value` — the discrete covariate value at which to calculate the estimated probability.
- `weight` — an optional weight to be incorporated in all estimation.

Value

Estimated probability = P(TL <t0+tau | TL > t0, Z, TS>t0).

Author(s)

Layla Parast

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.

Prob2.k.t *Estimates P(TL <t0+tau | TL > t0, Z, TS==ts).*

Description

This function calculates the probability that the an individual has the event of interest before t0 + tau given the discrete covariate, given the short term event occurred before t0 and occurred at time ts, and given the long term event has not yet occurred and the individual is still at risk at time t0.This function is called by Prob.Covariate.ShortEvent; this function should not be called on its own.

Usage

```
Prob2.k.t(t, t0, tau, data.use, bandwidth, covariate.value, weight = NULL)
```

Arguments

`t`	time of the short term event, ts, on the log scale.
`t0`	the landmark time.
`tau`	the residual survival time for which probabilities are calculated. Specifically, this function estimates the probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0.
`data.use`	n by 5 matrix. 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), the third column is log(XS) = log(min(TS, C)) where TS is the time of the short term event, C is the censoring time, the fourth column is DS =1*(TS<C), and the fifth column is the covariate.
`bandwidth`	bandwidth to be used.
`covariate.value`	covariate value at which to calculate probability.
`weight`	an optional weight to be incorporated in all estimation.

Value

returns estimated probabilities for each ts value (parameter t) at the specified covariate value; returns NA if ts>t0.

Author(s)

Layla Parast

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.

prob2.single — *Estimates P(TL <t0+tau | TL > t0, Z, TS==ts) for a single t.*

Description

Helper function for Prob2.k.t; should not be called directly.

Usage

```
prob2.single(K, W2i, Xi.long, tau, Di.short, Xi.short, Zi, t0, covariate.value)
```

Arguments

`K`	Kernel matrix.
`W2i`	inverse probability of censoring weights.
`Xi.long`	XL = min(TL, C) where TL is the time of the long term event, C is the censoring time.
`tau`	the residual survival time for which probabilities are calculated. Specifically, this function estimates the probability that the an individual has the event of interest before t0 + tau given the event has not yet occurred and the individual is still at risk at time t0.
`Di.short`	DS =1*(TS<C), where TS is the time of the short term event, C is the censoring time.
`Xi.short`	log(XS) = log(min(TS, C)) where TS is the time of the short term event, C is the censoring time.
`Zi`	covariate vector.
`t0`	landmark time.
`covariate.value`	specific covariate at which to estimate the conditional probability.

Value

returns estimated probability for values corresponding to the kernel matrix at the specified covariate value;

Author(s)

Layla Parast

`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 (see Ghat.FUN)

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