Package ‘newTestSurvRec’

April 16, 2018

Type Package
Title Statistical Tests to Compare Curves with Recurrent Events
Version 1.0.2
Date 2018-04-01
Depends R (>= 3.4.0)
Author Dr Carlos Miguel Martinez Manrique
Maintainer Carlos Martinez <cmmm7031@gmail.com>
Description Implements the routines to compare the survival curves with recurrent events, including the estimations of survival curves. The first model is a model for recurrent event, when the data are correlated or not correlated. It was proposed by Wang and Chang (1999) <doi:10.2307/2669690>. In the independent case, the survival function can be estimated by the generalization of the limit product model of Pena (2001) <doi:10.1198/016214501753381922>.
License GPL (>= 2)
URL https://www.r-project.org
Encoding UTF-8
RoxygenNote 6.0.1
NeedsCompilation no
Repository CRAN
Date/Publication 2018-04-16 09:01:31 UTC

R topics documented:

  newTestSurvRec-package .................................................. 2  
  DataColonDukesABvsC .................................................. 4  
  DataColonDukesABvsD .................................................. 5  
  DataColonDukesCvsD .................................................. 6  
  Dif.Surv.Rec .......................................................... 8  
  fit.Data.Survrecu ................................................... 10  
  FitSurvRec ............................................................ 11  
  is.Survrecu ........................................................... 13
package newTestSurvRec

Description

package newTestSurvRec

Details

Recurrent events are common in many areas: psychology, engineering, medicine, physics, astronomy, biology, economics and so on. Such events are very common in the real world: viral diseases, carcinogenic tumors, machinery and equipment failures, births, murders, rain, industrial accidents, car accidents and so on. The availability of computerized tools for the analysis is indispensable.

The survival analysis is a branch of statistics that allows us to model the time until the occurrence of an event. In general, the objectives of analysis are: the modeling of the survival function to estimate the risk or benefit of the occurrence of an event, the probability occurrence of this event and comparing population groups. The development of tools for the statistical analysis of recurrent event is relatively recent and are not fully known. The purpose of this package is to present statistical tests for the analysis of recurrent event data. Martinez et al. (2009) published a statistical test to compare survival curves of two groups with recurrent events. The hypothesis of the problem is:

\[ H_0 : S_1(t) = S_2(t) \]
\[ H_1 : S_1(t) \neq S_2(t) \]

Where, \( S_1(t) \) and \( S_2(t) \) are the survival curves of the both group. The statistic of test is,

\[ Z = \frac{\sum_{t \leq z} w_z [\Delta N(s, z; r) - E \{ \Delta N(s, z; r) \}]}{\sqrt{\sum_{t \leq z} w_z^2 Var \{ \Delta N(s, z; r) \}}} \]
The statistic $Z$ has a normal asymptotic behavior. Its square has a chi-square approximate behavior with a degree of freedom. So,

$$\Delta N(s, z; 1) = N(s, z + \Delta z; 1) - N(s, z; 1)$$

Now, $\Delta z$ is approaches to zero and as $\Delta N(s, z; 1)$ has a hyper-geometric behavior and expected value is equal to

$$Y(s, z; 1)\Delta N(s, z)/Y(s, z)$$

and variance equal to,

$$Var[\Delta N(s, z; 1)] = \frac{Y(s, z) - Y(s, z; 1)}{Y(s, z) - 1} \frac{\Delta N(s, z)}{Y(s, z)} \left[ 1 - \frac{\Delta N(s, z)}{Y(s, z)} \right]$$

This author proposed various types of weights ($w_z$),

$$w_z = [S(z)]^{\gamma} \left[ 1 - S(z) \right]^{\eta} \frac{[Y(s, z)]^\alpha}{[Y(s, z) + 1]^\beta}$$

The appropriate selection of weights depends on the behavior of the curves. With the selection of the values of the parameters ($\alpha, \beta, \gamma$ and $\eta$), on the proposal, is possible adjust its behavior. With the proposal, we are able of make studies on survival analysis with recurrent events and generate tests for analysis others, including the classical tests type: logrank, Gehan, Peto-Peto, Fleming-Harrington and so on. Note that, if all parameters are zero, $w_z = 1$, its generates the test type logrank for recurrent events. If, $\alpha = 1$ and the other parameters are zero $w_z = Y(s, z)$, its generates the test type Gehan. If, $\gamma = 1$ and the other parameters are zero $w_z = S(z)$, its generate the test of Peto-Peto. If, $\gamma = 1$ and $\eta = 1$ and the rest of the parameters are zero, its generate Fleming-Harrington test. On the other hand, if you analyze the test statistical of comparison for recurrent events, it depends on the counting processes $N$ and $Y$, which are doubles indexed. The index $S$ measures calendar time and $Z$ index measures the gap times. So, if the observation time tends to infinity and unity event study can only occur once in each unit and the statistical comparison becomes the weighted classical statistical comparison of groups of the survival analysis. We can conclude that test proposed by Martinez et al. (2009) are useful on diverse fields of research, such as: medicine, public health, insurance, social science, reliability and others.

Note

This package have some functions that them were originally performed by the survrec package, which solved the adjustment problem of the PSH and WC estimators using Fortran routines. With the permission of the author, Dr. Juan R. Gonzalez, the algorithm of base was taken, modified, the algorithm, WC estimator was reprogrammed and adapted to the needs of the newTestSurvRec package and thus avoid dependence. Thanks to Dr. Gonzalez

Author(s)

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

References

Description

This data set contains the re-hospitalization times of patients diagnosed with stage AB and patients diagnosed with stage C.

Usage

data(DataColonDukesABvsC)

Format

A data frame with 655 observations on the following 10 variables.

- j Observation number
- iden identification of each subject. Repeated for each recurrence
- id identification of each subject. Repeated for each recurrence
- tinicio Initial time of observation just before each recurrence
- time re-hospitalization or censoring gap time
- tcal re-hospitalization or censoring calendar time
- event censoring status. All event are 1 for each subject excepting last one that it is 0
- chemoter Did patient receive chemotherapy? 1: No or 2: Yes
- dukes Dukes tumor stage: 1:A-B or 2:C
- distance distance from living place to hospital 1:<=30 Km. or 2:>30 Km.
Details

The patients included in the study have been operated between January 1996 and December 1998. For each patient, we have considered this date as the beginning of the observational period. All patients were followed until June 2002. Consequently, the length of the monitoring period can differ for each patient, depending on its surgery date. The first inter occurrence time has been considered as the time between the surgical intervention and the first hospitalization related to cancer. Four hundred and three patients with colon and rectum cancer have been included in the study. Information about their sex (male or female), age (60, 60-74 or 75), and tumor stage using Dukes classification (A-B, C, or D) have been recorded. The following inter- occurrence times have been considered as the difference between the last hospitalization and the current one. Only re-admissions related to cancer have been considered.

Source

This data were obtained from Gonzalez, J.R. et al. (2009)

References


Examples

data(DataColonDukesABvsC)

data(DataColonDukesABvsD)  Re-hospitalization of patients with colorectal cancer

Description

This data contains re-hospitalization times of patients diagnosed with stage AB and patients diagnosed with stage D.

Usage

data(DataColonDukesABvsD)
DataColonDukesCvsD

Format

A data frame with 527 observations on the following 10 variables

This data frame contains the following columns:

- **j**: Observation number
- **iden**: Observation of each subject. Repeated for each recurrence
- **id**: Observation of each subject. Repeated for each recurrence
- **tinicio**: Initial time of observation just before each recurrence
- **time**: Re-hospitalization or censoring gap time
- **tcal**: Re-hospitalization or censoring calendar time
- **event**: Censoring status. All events are 1 for each subject excepting the last one that is 0
- **chemoter**: Did patient receive chemotherapy? 1: No or 2: Yes
- **dukes**: Dukes tumoral stage: 1:A-B or 3:D
- **distance**: Distance from living place to hospital 1:<=30 Km. or 2:>30 Km.

Details

See details on DataColonDukesABvsD

Source

This data were obtained from Gonzalez, J. R. et al. (2009)

References


Examples

data(DataColonDukesABvsD)
XL<-data(DataColonDukesABvsD)
print(XL)

| DataColonDukesCvsD | Rehospitalization of patients with colorectal cancer |

Description

This data contains the re-hospitalization times of patients diagnosed with stage C and patients diagnosed with stage D
Usage

data(DataColonDukesCvsD)

Format

A data frame with 537 observations on the following 10 variables

This data.frame contains the following columns

- **j**: Observation number
- **Iden**: Identification of each subject. Repeated for each recurrence
- **id**: Identification of each subject. Repeated for each recurrence
- **tinicio**: Initial time of observation just before each recurrence
- **time**: Re-hospitalization or censoring gap time
- **tcal**: Re-hospitalization or censoring calendar time
- **event**: Censoring status. All event are 1 for each subject excepting last one that it is 0
- **chemoter**: Did patient receive chemotherapy? 1: No or 2: Yes
- **dukes**: Dukes tumor stage: 2:C or 3:D
- **distance**: Distance from living place to hospital 1: <=30 Km. or 2: >30 Km.

Details

See details on DataColonDukesABvs

Source

This data were obtained from Gonzalez, J.R. *et al.* (2009)

References


Examples

data(DataColonDukesCvsD)
XL<-data(DataColonDukesCvsD)
print(XL)
This function computes statistical difference between two survival curves

Description

p-values of these tests are computed.

Usage

Dif.Surv.Rec(XX, type, alfa, beta, gamma, eta)

Arguments

- **XX**: Object type recurrent events data
- **type**: "LRrec","Grec","TWrec","PPrec","PMrec","FHrec","CMrec","Mrec","all"
- **alfa**: The appropriate choice, see \( w_z \). Defect value is equal zero
- **beta**: The appropriate choice, see \( w_z \). Defect value is equal zero
- **gamma**: The appropriate choice, see \( w_z \). Defect value is equal zero
- **eta**: The appropriate choice, see \( w_z \). Defect value is equal zero

Details

This function contains tests to compare survival curves with recurrent events. The curves are estimated using Pena-Strawderman-Hollander or Wang-Chang estimator. **GPLE or PSH model**: Pena et al. (2001) defined an estimator of the survival function to recurrent events or Kaplan-Meier estimator GPLE. They used two counting processes \( N \) and \( Y \). The PSH estimator was defined as,

\[
\hat{S}(z) = \prod_{t \leq z} \left[ 1 - \frac{\Delta N (s, z)}{Y (s, z)} \right]
\]

The authors considered two time scales: one related to calendar time (S) and other related to inter occurrences time (T). So, the counting process \( N(s, z) \) represents the number of observed events in the calendar period \([0, s]\) with \( t \leq z \) and \( Y(s, z) \) represents the number of observed events in the period \([0, s]\) with \( t \geq z \). The product-limit estimator was developed by Pena, Strawderman and Hollander, called PSH. This estimator is useful when the inter occurrence times are assumed to represent IID sample from some underlying distribution \( F \). The GPLE estimator is defined as:

\[
\hat{S}_r(z) = \prod_{t \leq z} \left[ 1 - \frac{\Delta N (s, z; r)}{Y (s, z; r)} \right] \nabla r = 1, 2.
\]

**WC model**: Wang-Chang (1999) proposed an estimator of the common marginal survivor function in the case where within-unit inter occurrences times are correlated. The correlation structure
considered by Wang and Chang (1999) is quite general and contains, the cases particular, both the
i.i.d. and multiplicative frailty model as special cases. The WC estimator was defined using two
new processes, $d^*$ and $R^*$.

$$
\hat{S}(t) = \prod_{T_k \leq t} \left[ 1 - \frac{d^*(T_k)}{R^*(T_k)} \right]
$$

The authors try to take into account in the definition of $N$ and $Y$ that an individual may have more
than one event. In fact, this estimator has the same way as the GPLE estimator but using these two
different processes. The index $d^*$ represents the sum of the proportion of individuals of the inter
occurrences times which are equal to $t$ when there is at least one event. On the other hand, $R^*$
represents an average of the individuals that are at risk time $t$, where for each individual the average
is the number of failures or censored times at least equal to $t$. This average is done regarding the
number of events that there are to each individual and in case $K$ is 0 is divided by 1. For definition
more formal see Martinez (2009) and Pena et. al (2001). The WC estimator of $S$ eliminates the
bias for the product-limit estimator developed by PSH (2001) when the inter occurrences times are
correlated within units. However, when applied to i.i.d. inter occurrence times, this estimator is not
expected to perform as well as the PSH estimator, especially with regard to efficiency.

**Value**

# Dif.Surv.Rec(TBCplapyr,"all",0,0,0,0). Values returned

<table>
<thead>
<tr>
<th>Nomb.Est</th>
<th>Chi.square</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>LRrec</td>
<td>0.3052411</td>
<td>0.5806152</td>
</tr>
<tr>
<td>Grec</td>
<td>1.4448446</td>
<td>0.2293570</td>
</tr>
<tr>
<td>TWrec</td>
<td>0.9551746</td>
<td>0.3284056</td>
</tr>
<tr>
<td>PPrec</td>
<td>1.1322772</td>
<td>0.2872901</td>
</tr>
<tr>
<td>PMrec</td>
<td>1.1430319</td>
<td>0.2850126</td>
</tr>
<tr>
<td>PPrec</td>
<td>1.1834042</td>
<td>0.2766641</td>
</tr>
<tr>
<td>HFrec</td>
<td>0.3052411</td>
<td>0.5806152</td>
</tr>
<tr>
<td>CMrec</td>
<td>0.3052411</td>
<td>0.5806152</td>
</tr>
<tr>
<td>Mrec</td>
<td>1.5298763</td>
<td>0.2161310</td>
</tr>
</tbody>
</table>

**Author(s)**

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

**References**

Martinez C., Ramirez, G., Vasquez M. (2009). Pruebas no parametricas para comparar curvas de
supervivencia de dos grupos que experimentan eventos recurrentes. Propuestas. Revista Ingenieria
paracion de curvas de supervivencia al caso de eventos de naturaleza recurrente. Tesis doctoral.
Universidad Central de Venezuela (UCV). Caracas-Venezuela.

**See Also**

Examples

```r
data(TBCplapyr)
#Return the p-values of the all tests
Dif.Surv.Rec(TBCplapyr,"all",0,0,0,0)
#Return the p-value of the LRrec test
Dif.Surv.Rec(TBCplapyr)
#Return the p-value of the Grec test
Dif.Surv.Rec(TBCplapyr,"Grec")
#Return the p-values of the CMrec tests
#The CMrec test with this parameters generates LRrec test
Dif.Surv.Rec(TBCplapyr,"all",0,0,0,0)
#The CMrec test with this parameters generates Grec test
Dif.Surv.Rec(TBCplapyr,"all",0,0,1,0)
#The CMrec test with this parameters generates TWrec test
Dif.Surv.Rec(TBCplapyr,"all",0,0,0.5,0)
```

Description

This function let to adjust the ID’s the database in case that it is not have the order numeric correct. Observation: this function only let to adjust the id variable not sort the rest of the data.

Usage

```r
fit.Data.Survrecu(x)
```

Arguments

- `x` a database type dataframe

Value

Returns the correct numeric order for the dataframe

Note

The last id on each unit of the database to have be a censored data and the occurrences have that to precede to this last it.

Author(s)

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>
References


See Also

FitSurvRec, Survrecu, is.Survrecu

Examples

data(MMC.TestSurvRec)
ID<-fit.Data.Survrecu(Survrecu(MMC.TestSurvRec$id,MMC.TestSurvRec$time,
                           MMC.TestSurvRec$event))

ID
fit<-PSH.fit(Survrecu(ID,MMC.TestSurvRec$time,
                      MMC.TestSurvRec$event))
fit$time
fit$surv
plot(fit$time,fit$surv)

write(fit)

data(DataColonDukesABvsD)
XL<-data(DataColonDukesABvsD)
DataColonDukesABvsD$Iden
Y<-fit.Data.Survrecu(Survrecu(DataColonDukesABvsD$Iden,DataColonDukesABvsD$time,
                                DataColonDukesABvsD$event))

Y
fit<-WC.fit(Survrecu(Y,DataColonDukesABvsD$time,DataColonDukesABvsD$event))
fit$time
fit$surv
plot(fit$time,fit$surv)

print(data.frame(time=fit$time,n.event=fit$n.event,
                 Surv=fit$survfunc,stderr=fit$stderr))

---

**FitSurvRec**  
*Compute a Survival Curve for Recurrent Event Data given a variable of group*

**Description**

Computes an estimate of a survival curve for recurrent event data using either the *Pena, Strawderman and Hollander* or *Wang and Chang* estimators. It also computes the asymptotic standard errors. The resulting object of class Survrecu is plotted.
Usage

FitSurvRec(formula, data, type = " pena-strawderman-hollander", ...)

Arguments

formula A formula object. If a formula object is supplied it must have a SurvRecu object as the response on the left of the operator and a term on the right. For a single survival curve as part of the formula is required.
data a data frame in which to interpret the variables named in the formula.
type a character string specifying the type of survival curve. Possible value are "pena-strawderman-hollander" or "wang-chang". The default is "pena-strawderman-hollander".
additional arguments passed to the type of estimator.

Details

See the help details of PSH.fit or WC.fit depending on the type chosen

Value

A FitSurvRec object. Methods defined for FitSurvRec objects are provided for print, lines and plot.

Author(s)

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

References


See Also


Examples

data(MMC.TestSurvRec)
# fit a PSH survival function and plot it
fitPSH <- FitSurvRec(SurvRecu(id, time, event)~1, data = MMC.TestSurvRec)
plot(fitPSH$time, fitPSH$survfunc, type = "s", ylim = c(0, 1),
     xlim = c(0, max(fitPSH$time)))
title(main = list("Survival Curve with Recurrent Event Data",
        cex = 0.8, font = 2.3, col = "dark blue"))
mtext("Research Group: AVANCE USE R!", cex = 0.7, font = 2,
      col = "dark blue", line = 1)
mtext("Software made by: Dr. Carlos Martinez", cex = 0.6, font = 2,
is.Survrecu

This function verify if the formula type of survival recurrent is object type newTestSurvRec

Description
To verify if the create object type Survrecu is a formula model type newTestSurvRec

Usage
is.Survrecu(x)

Arguments
x Object type formula of the class newTestSurvRec

Value
False if the object is not type formula
True if the object is type formula

Author(s)
Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

References

See Also
FitSurvRec, Dif.Surv.Rec, Survrecu, FitSurvRec
Examples

```r
data(MMC.TestSurvRec)
x<-Survrecu(MMC.TestSurvRec$id,MMC.TestSurvRec$time,MMC.TestSurvRec$event)-1
is.Survrecu(x)
```

<table>
<thead>
<tr>
<th>MMC.TestSurvRec</th>
<th>Migratory Motor Complex</th>
</tr>
</thead>
</table>

Description

This contains the Migratory Motor Complex data

Usage

```r
data(MMC.TestSurvRec)
```

Format

A data frame with 99 observations on the following 5 variables.
- `j` Number of the observation on dataset
- `id` ID of each subject. Repeated for each recurrence
- `time` recurrence or censoring time
- `event` censoring status. All event are 1 for each subject excepting last one that it is 0
- `group` A factor with levels Females Males

Details

The data correspond a study from the Section for Gastroenterology of Department of Internal Medicine, Ulleal University Hospital of Oslo.

Source


References


Examples

```r
data(MMC.TestSurvRec)
XL<-data(MMC.TestSurvRec)
print(XL)
Print.Summary(MMC.TestSurvRec)
## maybe str(MMC.TestSurvRec) ; plot(MMC.TestSurvRec) ...
```
Plot.Cusum.Events

Plot data with recurrent events

Description

This function plot data with recurrent events

Usage

Plot.Cusum.Events(yy, xy = 1, xf = 1, coevent = "blue", colcensor = "red",
 ltyx = 1, lwdx = 1)

Arguments

yy
Data type recurrent events. Examples: TBCplapyr, TBCplathi or TBCpyrthi
xy
Initial unit to start the plotted
xf
Final unit of the plotted
colevent
It is color that identifies the event
colcensor
It is color that identifies the censor
ltyx
The line type. Line types can either be specified as an integer (0="blank",
1="solid" (default), 2="dashed", 3="dotted", 4="dotdash", 5="longdash", 6="twodash")
or as one of the character strings: "blank", "solid", "dashed", "dotted", "dot-
dash", "longdash", or "twodash", where blank uses invisible lines (i.e., does not
draw them)
lwdx
The line width, a positive number, defaulting to 1. The interpretation is device-
specific, and some devices do not implement line widths less than one. (See the
help on the device for details of the interpretation.)

Details

This function print and plot as max 5 units each intent.

Value

Print the data correspond to the selects units

Note

This graph is useful because it facilitates the processes of counting in the units

Author(s)

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>
References


See Also

Plot.Data, Events, Plot.Surv.Rec

Examples

```r
XL<-data(TBCPlapyr)
#TBCPlapyr
# See, the unit number 1 to 24
Plot.Cusum.Events(TBCPlapyr,1,24,"green","red",2,1)
# See, the unit number 10 to 12
Plot.Cusum.Events(TBCPlapyr,10,12,"pink","blue",1,3)
# See, the unit number 5 to 9
Plot.Cusum.Events(TBCPlapyr,5,11,,2,3)
```

Description

This function plot data with recurrent events

Usage

```r
Plot.Data.Events(yy, paciente, inicio, dias, censored, especiales, 
colevent="red",colcensor="blue")
```

Arguments

- **yy**: Data type recurrent events. Examples: TBCPlapyr, TBCPlathi or TBCPyrthi
- **paciente**: Vector of number of units on the data base
- **inicio**: Vector, its assumed that the units are observed from one time equal to zero.
- **dias**: Vector of the periods of observations of the study untiis
- **censored**: vector of times of censorship for each unit
- **especiales**: Three-column matrix containing the identification of the units of study in each observation, the times of occurrence of the event or censorship and type of event.
- **colevent**: Color event identifier.
- **colcensor**: Color censored data identifier.

Details

The plot shows the recurrence of the events on the time
Plot.Data.Events

Value

This function returned the pictorial representation of the set of recurrence events data

Note

We recommend users to use routines similar to the example.

Author(s)

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

References


See Also


Examples

data(TBCplapyr)
XL<-data(TBCplapyr)
p<-ncol(TBCplapyr)
N<-nrow(TBCplapyr)
censor<-matrix(TBCplapyr$event)
especiales<-matrix(data=0,nrow(TBCplapyr),3)
especiales[,1]<-matrix(TBCplapyr$id)
especiales[,2]<-matrix(TBCplapyr$Tcal)
especiales[,3]<-matrix(TBCplapyr$event)
niveles<-levels(factor(especiales[,1]))
for(i in 1:N){
  for(j in 1:nrow(matrix(niveles))){
    if (as.character(especiales[i,1])==niveles[j]) especiales[i,1]<-j}
  StudyPeriod<-matrix(data=0,nrow(matrix(niveles)),1)
  start<-matrix(data=0,nrow(matrix(niveles)),1)
  k<-0
  for(j in 1:N){if (TBCplapyr$event[j]==0){k<-k+1;StudyPeriod[k,1]<-TBCplapyr$Tcal[j]}}
  units<-matrix(1:nrow(matrix(niveles)),nrow(matrix(niveles)),1)
  Plot.Data.Events(TBCplapyr,units,start,StudyPeriod,censor,especiales,"black","blue")
  Plot.Data.Events(TBCplapyr,units,start,StudyPeriod,censor,especiales,"red","black")
Plot.Event.Rec

This function plots the occurrence of an event in two scales time.

Description

Recurrent events are plotted. A plot is returned. The counting processes are a powerful tool in survival analysis. These processes consider two scale time, a calendar time and a gap time. This idea originally provides from Gill (1981) and the concept was extended by Pena et al. (2001).

Usage

Plot.Event.Rec(yy, xy, xf)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>yy</td>
<td>Object type recurrent events data. Example: TBCplayr</td>
</tr>
<tr>
<td>xy</td>
<td>Identification of the unit to plotted. ’xy = 1’ is defect value.</td>
</tr>
<tr>
<td>xf</td>
<td>Argument to plot the occurrence events of the unit ’xf’. ’xf = 1’ is defect value.</td>
</tr>
</tbody>
</table>

Value

Plot is returned. Pena et al. (2001) designed a special graphic, that allows to count the occurrence of events per unit time. Doubly-indexed processes illustration for an case. The graphic shows a case followed during 24.01 months. This patient presents four recurrences at months 7, 10, 16 and 24 from the beginning of study. This fact implies that interoccurrence times are 7, 3, 6, 8 and the censored time correspond to 0.01 months. Let us assume that we are interested in computing the single processes, \( N(t) \) and \( Y(t) \) for a selected interoccurrence time \( t = 5 \). In this case \( N(t = 5) = 1 \) and \( Y(t = 5) = 3 \). For the calendar time scale, \( s = 20 \), we have \( N(s = 20) = 3 \) and \( Y(s = 20) = 1 \). Now, let us assume that we would like to know double-indexed processes for both selected interoccurrence and calendar times. Using both time scales we observe that \( N_{14}(s = 20, t = 5) = 1 \), \( Y_{14}(s = 20, t = 5) = 2 \) and \( \Delta N_{14}(s = 20, t = 6) = 1 \).

Author(s)

Dr. Carlos M. Martinez M. <cmmm7031@gmail.com>

References


See Also

Dif.Surv.Rec, Plot.Data.Events
Examples

```r
XX<-data(TBCplapyr)
# See, the unit number 14
Plot.Event.Rec(TBCplapyr,14,14)
# See, the unit number 5
Plot.Event.Rec(TBCplapyr,5,5)
```

**Plot.Surv.Rec**  
Plots of the survival function from an object with class newTestSurvRec, using PHS or WC models

**Description**

The survival curves are plotted. Both curves are estimates using PSH o WC estimator. This package is available in language R. This important clearly, that the PHS estimator is of valid use when it assumed that the inter-occurrence times are IID. Its obvious that this assumption is restrictive in biomedical applications and its use is more valid on the field of engineering. For WC estimated not import if the data is correlated.

**Usage**

```r
Plot.Surv.Rec(XX,...)
```

**Arguments**

- **XX**  
  Data type recurrent events. Example: TBCplapyr

- **...**  
  Other objects

**Value**

The survival curves for both groups are plotted.

**Author(s)**

Dr. Carlos M. Martinez M. <cmmm7031@gmail.com>

**References**


**See Also**

Plot.Event.Rec, Dif.Surv.Rec
Print.Summary

Examples

XL<-data(TBCplapyr)
Plot.Surv.Rec(TBCplapyr)

Print.Summary

Function to print summary of statistics tests to comparison of the survival curves of the groups with recurrent events

Description

Returns matrices that contain the estimations of the survival curves for both groups. The estimations of survival curves of both groups are made using PSH estimator. The p.values of the tests are returned.

Usage

Print.Summary(XX,...)

Arguments

XX Object type recurrent events data
...
other objects

Details

See Dif.Surv.Rec(XX,...)

Value

Put object type recurrent events data. #Print.Summary(TBCplapyr). #Values returned:

<table>
<thead>
<tr>
<th>time</th>
<th>n.event</th>
<th>n.risk</th>
<th>Surv_G1</th>
<th>std.error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>127</td>
<td>0.984</td>
<td>0.0110</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>124</td>
<td>0.913</td>
<td>0.0243</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>113</td>
<td>0.800</td>
<td>0.0340</td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>98</td>
<td>0.726</td>
<td>0.0380</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.....</td>
<td>.....</td>
</tr>
<tr>
<td>29</td>
<td>1</td>
<td>18</td>
<td>0.244</td>
<td>0.0422</td>
</tr>
<tr>
<td>31</td>
<td>1</td>
<td>13</td>
<td>0.225</td>
<td>0.0427</td>
</tr>
<tr>
<td>35</td>
<td>1</td>
<td>9</td>
<td>0.200</td>
<td>0.0439</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>time</th>
<th>n.event</th>
<th>n.risk</th>
<th>Surv_G2</th>
<th>std.error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>84</td>
<td>0.964</td>
<td>0.0199</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>81</td>
<td>0.893</td>
<td>0.0327</td>
</tr>
<tr>
<td>Numb.Est</td>
<td>Chi.square</td>
<td>p.value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>------------</td>
<td>---------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LRrec</td>
<td>0.3052411</td>
<td>0.5806152</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grec</td>
<td>1.4448446</td>
<td>0.2293570</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWrec</td>
<td>0.9551746</td>
<td>0.3284056</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPrrec</td>
<td>1.1322772</td>
<td>0.2872901</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMrec</td>
<td>1.1430319</td>
<td>0.2850126</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PPrrec</td>
<td>1.1834042</td>
<td>0.2766641</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HFrec</td>
<td>0.3052411</td>
<td>0.5806152</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMrec</td>
<td>0.3052411</td>
<td>0.5806152</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mrec</td>
<td>1.5298763</td>
<td>0.2161310</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Author(s)**

Dr. Carlos M. Martinez M. <cmmm7031@gmail.com>

**References**


**See Also**

Dif.Surv.Rec, Plot.Surv.Rec

**Examples**

data(TBCplapyr)
Print.Summary(TBCplapyr)
Estimator of the survival curve using the estimator developed by Pena, Strawderman and Hollander

Description

Estimation of survival function for recurrence time data by means the generalized product limit estimator (PLE) method developed by Pena, Strawderman and Hollander. The resulting object of class Survrecu is plotted by plot, before it is returned.

Usage

PSH.fit(x, tvals)

Arguments

x a survival recurrent event object
tvals vector of times where the survival function can be estimated.

Details

The estimator computed by this object is the nonparametric estimator of the inter-event time survivor function under the assumption of a renewal or IID model. This generalizes the product-limit estimator to the situation where the event is recurrent. For details and the theory behind this estimator, please refer to Pena, Strawderman and Hollander (2001, JASA).

Value

Value returned

n number of unit or subjects observed.
m vector of number of recurrences in each subject (length n)
fail vector of number of recurrences in each subject (length n*m). Vector ordered (e.g. times of first unit, times of second unit, ..., times of n-unit)
censored vector of times of censorship for each subject (length n)
numdistinct number of distinct failures times.
distinct vector of distinct failures times.
Atrisk matrix of number of persons-at-risk at each distinct time and for each subject
survfunc vector of survival estimated in distinct times
tvals copy of argument.

Note

This function was originally performed by the survrec package, which solved the adjustment problem of the PSH estimator using Fortran routines. With the permission of its author, the algorithm of the packet base was taken, modified, the algorithm of the PSH estimates was reprogrammed and adapted to the needs of the newTestSurvRec package and thus avoid dependence.
**Author(s)**

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

**References**


See Also


**Examples**

```r
data(MMC.TestSurvRec)
fitPSHa<-PSH.fit(Survrecu(MMC.TestSurvRec$id,MMC.TestSurvRec$time,
                        MMC.TestSurvRec$event))
fitPSHa$surv
plot(fitPSHa$time,fitPSHa$survfunc,type="s",ylim=c(0,1),xlab=c(0,max(fitPSHa$time))
     ,title(main = list("Survival Curve with Recurrent Event Data",
                    cex = 0.8, font = 2.3, col = "dark blue"))
     ,mtext("Research Group: AVANCE USE R!", cex = 0.7, font = 2,
                     col = "dark blue", line = 1)
     ,mtext("Software made by: Dr. Carlos Martinez", cex = 0.6, font = 2,
                    col = "dark red", line = 0)
```

---

**Qsearch.Fractil**

*Calculate the survival time to a selected quantile*

**Description**

Auxiliary function called from Dif.Surv.Rec function. Given a FitSurvRec object we obtain the quantile from a survival function using PHS or WC estimators.

**Usage**

```r
Qsearch.Fractil(fr, qr = 0.5)
```

**Arguments**

- `fr` : FitSurvRec object
- `qr` : quantile. Default is 0.5

**Value**

Returns the time in a selected quantile
**Survrecu**

**Author(s)**

Dr. **Carlos M Martinez M.**, <cmmm7031@gmail.com>

**References**


**See Also**

FitSurvRe, Survrecu, is.Survrecu

**Examples**

```r
XL<-data(MMC.TestSurvRec)
fit<-FitSurvRec(Survrecu(id,time,event)-1,data=MMC.TestSurvRec)
# 35th percentile from the survival function
Qsearch.Fractil(fit,q=0.35)
```

---

**Survrecu**  
*Create a Survival recurrent object type newTestSurvRec*

**Description**

Create a survival recurrent object, usually used as a response variable in a model formula

**Usage**

Survrecu(id, time, event)

**Arguments**

- **id**: Identifier of each subject. This value is the same for all recurrent times of each subject.
- **time**: time of recurrence. For each subject the last time are censored.
- **event**: The status indicator, 0=no recurrence 1=recurrence. Only these values are accepted.

**Value**

An object of class newTestSurvRec is returned. newTestSurRec object is implemented as a matrix of 3 columns. No method for print. In the case of is.Survrecu, a logical value TRUE if x inherits from class Survrecu, otherwise an FALSE.
**TBCplapyr**

**Author(s)**

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

**References**


**See Also**

FitSurvRec, is.Survrecu

**Examples**

```r
data(MMC.TestSurvRec)
Survrecu(MMC.TestSurvRec$id,MMC.TestSurvRec$time,MMC.TestSurvRec$event)-1
```

<table>
<thead>
<tr>
<th>TBCplapyr</th>
<th>Data in patients with bladder cancer treated with placebo or pyridoxine</th>
</tr>
</thead>
</table>

**Description**

This database corresponds to the time of recurrence of tumors in 78 patients with bladder cancer. Patients were randomly assigned to treatments: placebo (47 patients) and pyridoxine (31 patients). Data type data.frame with 222 observations on 8 variables.

**Usage**

```r
data(TBCplapyr)
```

**Format**

A data frame with 222 observations on the following 9 variables.

- **j**: Observation number
- **id**: ID of each unit. Repeated for each recurrence
- **tinicio**: Initial time
- **time**: Recurrence or censoring time. For each unit the last time is censored
- **tcal**: Time if observation for each unit
- **event**: Censoring status. 1 = occurrence of the event in the unit and 0 right censored time
- **strata**: Number of strata
- **trt**: A factor with levels Placebo or Pyridoxine
- **group**: A factor with levels. Group identification
Details

Experiment Byar (1980). The database Byar experiment is used and the time (months) of recurrence of tumors in 116 sick patients with superficial bladder cancer is measured. These patients were randomly allocated to the following treatments: placebo (47 patients), pyridoxine (31 patients) and thiotepa (38 patients).

Source


References


Examples

```
XL<-data(TBcplapyr)
XL<-data(TBcplapyr)
print(XL)
Print.Summary(TBcplapyr)
```

Data in patients with bladder cancer treated as placebo or thiotepa

Description

This database corresponds to the time of recurrence of tumors of 85 patients with bladder cancer. Patients were randomly assigned to treatments: placebo (47 patients) and thiotepa (38 patients). Data type data.frame with 217 observations on 8 variables.

Usage

```
data(TBcplathi)
```

Format

A data frame with 217 observations on the following 9 variables.

- j Observation number
- id ID of each unit. Repeated for each recurrence
- Tinicio Inicial time
time  recurrence or censoring time. For each unit the last time is censored
Tca1  Time if observation for each unit
event  censoring status. 1 = occurrence of the event in the unit and 0 right censored time
strata  Number of strata
trt  a factor with levels Placebo or Thiotepa
group  A factor with levels. Group identifier

Details

Experiment Byar (1980). The database Byar experiment is used and the time (months) of recurrence of tumors in 116 sick patients with superficial bladder cancer is measured. These patients were randomly allocated to the following treatments: placebo (47 patients), pyridoxine (31 patients) and thiotepa (38 patients).

Source


References


Examples

data(TBCplathi)
XL<-data(TBCplathi)
print(XL)
Print.Summary(TBCplathi)
## maybe str(TBCplathi) ; plot(TBCplathi) ...

---

**TBCpyrthi**

*Data in patients with bladder cancers and treated with pyridoxine or thiotepa*

Description

This database corresponds to the time of recurrence of tumors of 69 patients with bladder cancer. Patients were randomly assigned to treatments: pyridoxine (38 patients) and thiotepa (31 patients). Data type data.frame with 171 observations on 8 variables.

Usage

data(TBCpyrthi)
Format

A data frame with 171 observations on the following 9 variables.

j Observation number
id ID of each unit. Repeated for each recurrence
Tinicio Inicial time
time recurrence o censoring time. For each unit the last time is censored
Tcal Time if observation for each unit
event censoring status. 1 = occurrence of the event in the unit and' 0 right censored time
strata Number of strata
trt a factor with levels *Pyridoxine or Thiotepa*
group A factor with levels. Group identifier

Details

Experiment Byar (1980). The database Byar experiment is used and the time (months) of recurrence of tumors in 116 sick patients with superficial bladder cancer is measured. These patients were randomly allocated to the following treatments: placebo (47 patients), pyridoxine (31 patients) and thiotepa (38 patients).

Source


References


Examples

data(TBCpyrthi)
XL<-data(TBCpyrthi)
print(XL)
Print.Summary(TBCpyrthi)
## maybe str(TBCpyrthi) ; plot(TBCpyrthi) ...
**WC.fit**  

*Survival function estimator for recurrence time data using the estimator developed by Wang and Chang*

**Description**

Estimation of survival function for correlated by the product limit estimator PLE method developed by Wang and Chang.

**Usage**

`WC.fit(x, tvals)`

**Arguments**

- `x`  
a survival recurrent event object
- `tvals`  
vector of times where the survival function can be estimated.

**Details**

Wang-Chang (1999) proposed an estimator of the common marginal survivor function in the case where within-unit inter-occurrence times are correlated. The correlation structure considered by Wang and Chang (1999) is quite general and contains, in particular, both the i.i.d. and multiplicative (hence gamma) frailty model as special cases. This estimator removes the bias noted for the product-limit estimator developed by Pena, Strawderman and Hollander (PSH, 2001) when inter-occurrence times are correlated within units. However, when applied to i.i.d. inter-occurrence times, this estimator is not expected to perform as well as the PSH estimator, especially with regard to efficiency.

**Value**

Value returned

- `n`  
number of unit or subjects observed.
- `m`  
vector of number of recurrences in each subject (length n)
- `failed`  
vector of number of recurrences in each subject (length n*m). Vector ordered (e.g. times of first unit, times of second unit, ..., times of n-unit)
- `censored`  
vector of times of censorship for each subject (length n)
- `numdistinct`  
number of distinct failures times.
- `distinct`  
vector of distinct failures times.
- `AtRisk`  
matrix of number of persons-at-risk at each distinct time and for each subject
- `survfunc`  
vector of survival estimated in distinct times
- `tvals`  
copy of argument.
Note

This function was originally performed by the survrec package, which solved the adjustment problem of the WC estimator using Fortran routines. With the permission of its author, the algorithm was taken, modified, the algorithm, WC estimator was reprogrammed and adapted to the needs of the newTestSurvRec package and thus avoid dependence.

Author(s)

Dr. Carlos M. Martinez M., <cmmm7031@gmail.com>

References


See Also


Examples

```
XL<-data(MMC.TestSurvRec)
#---------------------------------------------------------------
fitPSHa<-PSH.fit(Survrecu(MMC.TestSurvRec$id,MMC.TestSurvRec$time, 
MMC.TestSurvRec$event))
fitPSHa$surv
fitPSHa$time
plot(fitPSHa$time,fitPSHa$survfunc,type="s" ,ylim=c(0,1), 
xlim=c(0,max(fitPSHa$time)))
title(main = list("Survival Curve with Recurrent Event Data", 
cex = 0.8, font = 2.3, col = "dark blue")

mtext("Research Group: AVANCE USE R!", cex = 0.7, font = 2, 
col = "dark blue", line = 1)

mtext("Software made by: Dr. Carlos Martinez", cex = 0.6, font = 2, 
col = "dark red", line = 0)

fitWCa<-WC.fit(Survrecu(MMC.TestSurvRec$id,MMC.TestSurvRec$time, 
MMC.TestSurvRec$event))
fitWCa$surv
fitWCa$time
plot(fitWCa$time,fitWCa$survfunc,type="s" ,ylim=c(0,1), 
xlim=c(0,max(fitWCa$time)))
```
Index

*Topic **Events**
  - Plot.Cusum.Events, 15
  - Plot.Data.Events, 16
*Topic **Event**
  - Plot.Event.Rec, 18
*Topic **Plot**
  - FitSurvRec, 11
  - Plot.Event.Rec, 18
  - Plot.Surv.Rec, 19
  - Print.Summary, 20
*Topic **Quantile**
  - Qsearch.Fractil, 23
*Topic **Recurrence**
  - Plot.Data.Events, 16
*Topic **Statistical**
  - Dif.Surv.Rec, 8
  - PSH.fit, 22
*Topic **Survival**
  - FitSurvRec, 11
  - is.Survrecu, 13
  - Plot.Surv.Rec, 19
  - Print.Summary, 20
  - Qsearch.Fractil, 23
  - Survrecu, 24
*Topic **Tests**
  - Dif.Surv.Rec, 8
  - is.Survrecu, 13
  - PSH.fit, 22
  - Survrecu, 24
*Topic **dataframe**
  - fit.Data.Survrecu, 10
*Topic **datasets**
  - DataColonDukesABvsC, 4
  - DataColonDukesABvsD, 5
  - DataColonDukesCvsD, 6
  - MMC.TestSurvRec, 14
  - TBCplapyr, 25
  - TBCplathi, 26
  - TBCpyrthi, 27
  - WC.fit, 29
*Topic **data**
  - fit.Data.Survrecu, 10
*Topic **package**
  - newTestSurvRec-package, 2
*Topic **plot**
  - Plot.Cusum.Events, 15
*Topic **survival**
  - WC.fit, 29
  - DataColonDukesABvsC, 4
  - DataColonDukesABvsD, 5
  - DataColonDukesCvsD, 6
  - Dif.Surv.Rec, 8
  - fit.Data.Survrecu, 10
  - FitSurvRec, 11
  - is.Survrecu, 13
  - MMC.TestSurvRec, 14
  - newTestSurvRec
    - (newTestSurvRec-package), 2
    - newTestSurvRec-package, 2
  - Plot.Cusum.Events, 15
  - Plot.Data.Events, 16
  - Plot.Event.Rec, 18
  - Plot.Surv.Rec, 19
  - Print.Summary, 20
  - PSH.fit, 22
  - Qsearch.Fractil, 23
  - Survrecu, 24
  - TBCplapyr, 25
  - TBCplathi, 26
  - TBCpyrthi, 27
  - WC.fit, 29