Package 'dblcens'

January 13, 2023

Title Compute the NPMLE of Distribution Function from Doubly Censored Data, Plus the Empirical Likelihood Ratio for F(T)

Version 1.1.9

Depends R (>= 3.5)

Suggests testthat (>= 3.0.0)

Description Doubly censored data, as described in Chang and Yang (1987) <doi:10.1214/aos/1176350608>), are commonly seen in many fields. We use EM algorithm to compute the nonparametric MLE (NPMLE) of the cummulative probability function/survival function and the two censoring distributions. One can also specify a constraint F(T)=C, it will return the constrained NPMLE and the -2 log empirical likelihood ratio for this constraint. This can be used to test the hypothesis about the constraint and, by inverting the test, find confidence intervals for probability or quantile via empirical likelihood ratio theorem. Influence functions of hat F may also be calculated, but currently, the it may be slow.

Maintainer Yifan Yang <yfyang.86@hotmail.com>

URL https://github.com/yfyang86/dblcens/

License GPL (>= 2)

Repository CRAN

NeedsCompilation yes

Date 2023-1-13 16:00:00

Config/testthat/edition 3

Author Mai Zhou [aut],
Li Lee [aut],
Kun Chen [aut],
Yifan Yang [aut, cre, cph]

Date/Publication 2023-01-13 13:50:05 UTC

R topics documented:

d011 *Compute NPMLE of CDF from doubly censored data*

Description

d011 computes the NPMLE of CDF from doubly censored data via EM algorithm starting from an initial estimator that have jumps at (1) uncensored points; (2) (mid-point of) consecutive survival times with censoring indicator pattern of (0,2), (see below for definition).

When there are ties, the left (right) censored points are treated as happened slightly before (after), to break tie. Also when the last observation happens to be right censored and/or when the first observation happens to be left censored, they are changed to uncensored. This is to ensure we obtain a proper distribution as the CDF estimator. (though this can be modified easily as they are written in R language).

It also computes the NPMLE of the two censoring distributions. There is an option that you may also try to compute the three influence functions (but could slow and memory hungry).

Usage

```
d011(z, d, identical = rep(0, length(z)),
     maxiter = 49, error = 0.00001, influence.fun = FALSE)
```

Arguments

`z` a vector of length n denoting observed times, (ties permitted)

`d` a vector of length n that contains censoring indicator: d= 2 or 1 or 0, (according to z being left, not, right censored)

`identical` optional. A vector of length n that has values either 0 or 1. identical[i]=1 means: even if (z[i],d[i]) is identical to (z[j],d[j]), for some $j \neq i$, they still stay as 2 observations, (not 1 obs. with weight 2, which only happen if identical[i]=0 and identical[j] =0). One reason for this is because they may have different covariates not shown here. This adds more flexibility for regression applications. Default value is identical = 0, (i.e. collapse if identical observations).

`maxiter` optional integer value. default to 49

`error` optional. Default to 0.00001

`influence.fun` optional. Default to FALSE. If TRUE, the code will try to compute the influence functions (3 of them) at the censored times. This computation can be very slow and memory intensive (for data with >500 censored times).

Details

The true NPMLE may have probability mass inside the interval where two consecutive times z[i] < z[j], having censoring pattern of d[i]=0 and d[j]=2. As the first example below show.

Value

a list contain the NPMLE of CDF and other information.

`time` Times of input z, with time corresponding to status=2 removed.

`status` Censoring status of the above times. Status = -1 means this is an added time because of the censoring pattern (0,2).

`surv` Survival probability at the above times.

`jump` Jumps of the NPMLE at the above times.

`exttime` Similar to times but those with status =2 not removed.

`extstatus` status of exttime

`extjump` jump pf NPMLE at exttime.

`extsurv.Sx` Estimated lifetime distribution.

`surv0.Sy` One of the censoring distributions.

`jump0` Jump of surv0.Sy

`surv2.Sz` Another censoring distribution.

`jump2` Jump of surv2.Sz

`conv` A vector of length 2: the actual number of iterations, and the actual error of successive iteration. If the iteration number equal to the maxiter you set, then the iteration has not converged.

`Nodes` Points where the influence function is computed.

`IC1tu` Influence function value at the nodes. See Chang (1990) for details.

`IC1tu2` Influence function values at other points. See Chang (1990) for details.

`IC2tu` ditto IC1tu

`IC3tu` ditto IC1tu

`VarFt` Estimated variances of $\hat{F}(t)$ at the Nodes.

Author(s)

Mai Zhou, Li Lee.

References

Chang, M. N. and Yang, G. L. (1987). Strong consistency of a nonparametric estimator of the survival function with doubly censored data. Ann. Statist. 15, 1536-1547.

Turnbull (1976) The empirical distribution function with arbitrarily grouped, censored and truncated data. JRSS B, 290-295.

Chang, M. N. (1990). Weak convergence in doubly censored data. Ann. Statist. 18, 390-405.

Chen, K. and Zhou, M. (2003). Nonparametric Hypothesis Testing and Confidence Intervals with Doubly Censored Data. Lifetime Data Analysis, 9, 71-91.

Examples

```
d011(z=c(1,2,3,4,5), d=c(1,0,2,2,1))
#
# you should get something like below (and more)
#
#       $time:
#       [1] 1.0 2.0 2.5 5.0    (notice the times, (3,4), corresponding
#                                   to d=2 are removed, and time 2.5 added
#       $status:               since there is a (0,2) pattern at
#       [1]  1  0 -1  1        times 2, 3. The status indicator of -1
#                                   show that it is an added time )
#       $surv
#       [1] 0.5000351 0.5000351 0.3333177 0.0000000
#
#       $jump
#       [1] 0.4999649 0.0000000 0.1667174 0.3333177
#
#       $exttime
#       [1] 1.0 2.0 2.5 3.0 4.0 5.0
#
#       $extstatus
#       [1]  1  0 -1  2  2  1
#
#       ......
#
#       $conv
#       [1] 3.300000e+01  8.788214e-06  ### did 33 iterations
#
# BTW, the true NPMLE of surv, i.e. 1-F(), is (1/2, 1/2, 1/3, 0) at times (1,2,2.5,5).
###### Example 2.
d011(c(1,2,3,4,5), c(1,2,1,0,1),influence.fun=TRUE)
#     we get
# ......
#$conv:
#[1] 3 0
#
#$Nodes:
#[1] 2 4
#
#$IC1tu:
#     [,1] [,2]
#[1,]   -1    0
#[2,]   -1   -2
#
#$IC2tu:
#           [,1] [,2]
#[1,]  0.0000000    0
#[2,] -0.3333333    0
#
#$IC3tu:
#     [,1]       [,2]
#[1,]   -1 -0.6666667
```

```
#[2,]   -1 -1.0000000
#
#$VarFt:
#[1] 0.24 0.24          ## est var of hat F(t) at t=nodes
######################################################
```

d011ch — *Compute NPMLE of CDF from doubly censored data, with and without a constraint, plus an empirical likelihood ratio*

Description

d011ch computes the NPMLE of CDF, with and without a constraint, from doubly censored data. It also computes the -2 log empirical likelihood ratio for testing the given constraint via empirical likelihood theorem, i.e. under Ho it should be distributed as chi-square with df=1.

It uses EM algorithm starting from an initial CDF estimator that have jumps at uncensored points as well as the mid-point of those censoring times that have a pattern of (0,2), (see below for definition and example.)

The constraint on the CDF are given in the form F(K) = konst. where you specify the time K and probability 'konst'.

When there are ties among censored and uncensored observations, the left (right) censored points are treated as happened slightly before (after), to break tie. Also the last right censored observation and first left censored observation are changed to uncensored, in order to obtain a proper distribution as estimator (though this can be modified easily as they are written in R language).

Usage

```
d011ch(z, d, K, konst,
     identical = rep(0, length(z)), maxiter = 49, error = 0.00001)
```

Arguments

z	a vector of length n denoting observed times, (ties permitted)
d	a vector of length n that contains censoring indicator: d= 2 or 1 or 0, (according to z being left, not, right censored)
K	the constraint time.
konst	the constraint value, i.e. F(K)=konst.
identical	optional. a vector of length n that has values either 0 or 1. identical[i]=1 means even if (z[i],d[i]) is identical with (z[j],d[j]), for some $j \neq i$, they still stay as 2 observations, not 1 observation with weight 2, which only happen if identical[i]=0 and identical[j] =0. One reason to do this is because they may have different covariates not shown here. This flexibility may be useful for regression applications. Default value is identical = 0.
maxiter	optional integer value. Default to 49
error	optional. Default to 0.00001. maxiter and error are used to control the time of computing.

Details

A true NPMLE may have probability mass inside an interval when two consecutive time having censoring pattern d= (0, 2). See the help file of d011() for more details.

The definition of the double censoring is same as the Example 6.4 of Owen's 2001 book (except he uses d=0, 1, -1. we use d=0, 1 and 2). Our empirical likelihood function for the doubly censored data is also the same with the definition L(F) given on the middle of p. 148 of the book. In our natation it is

$$EL_c(F) = \prod_{i=1}^{n} [\Delta F(z_i)]^{d_i=1} [1 - F(z_i)]^{d_i=0} [F(z_i)]^{d_i=2} .$$

Value

a list contain the NPMLE of CDF with and without the constraint, -2loglik ratio and other informations.

`time` survival times. Those corresponding to d=2 are removed. Those corresponding to (0,2) censoring pattern are added, at mid-point.

`status` Censoring status of the above times. Since left censored times are removed, there is no status =2. There may be -1, indicating that this is an added time for (0,2) censoring pattern.

`surv` The survival function at the above times.

`jump` Jumps of NPMLE at the above times.

`exttime` Similar to time but now include the left censored times.

`extstatus` Censoring status of exttime. -1 has same meaning as status before.

`extjump` Jumps of the unconstrained NPMLE on extended times.

`extsurv.Sx` Survival probability at exttime.

`konstdist` The constrained NPMLE of distribution.

`konstjump` Jumps of the constrained NPMLE of CDF.

`konsttime` Location of the constraint, same as K in the input.

`theta` is the same value `konst` in the input.

`"-2loglikR"` the Wilks statistics. Distributed approximately chi-square df=1 under Ho

`maxiter` the actual number of iterations for the unconstrained NPNLE. The constrained NPMLE usually took less iterations to converge.

Author(s)

Kun Chen, Mai Zhou

References

Chang, M. N. and Yang, G. L. (1987). Strong consistency of a nonparametric estimator of the survival function with doubly censored data. Ann. Statist. 15, 1536-1547.

Murphy, S. and Van der Varrt. (1997). Semiparametric Likelihood Ratio Inference. Ann. Statist. 25, 1471-1509.

Chen, K. and Zhou, M. (2003). Nonparametric Hypothesis Testing and Confidence Intervals with Doubly Censored Data. Lifetime Data Analysis. 9, 71-91.

Owen, A. (2001). Empirical Likelihood. Chapman and Hall CRC press, Boca Raton.

Examples

```
d011ch(z=c(1,2,3,4,5), d=c(1,0,2,2,1), K=3.5, konst=0.6)
#
# Here we are testing Ho: F(3.5) = 0.6 with a two-sided alternative
# you should get something like
#
#       $time:
#       [1] 1.0 2.0 2.5 5.0    (notice the times, (3,4), corresponding
#                                   to d=2 are removed, and time 2.5 added
#       $status:               since there is a (0,2) pattern at
#       [1]  1  0 -1  1        times 2, 3. The status indicator of -1
#                                   show that it is an added time )
#       $surv
#       [1] 0.5000351 0.5000351 0.3333177 0.0000000
#
#       $jump
#       [1] 0.4999649 0.0000000 0.1667174 0.3333177
#
#       $exttime
#       [1] 1.0 2.0 2.5 3.0 4.0 5.0       (exttime include all the times,
#                                         censor or not, plus the added time)
#       $extstatus
#       [1]  1  0 -1  2  2  1
#
#       $extjump
#       [1] 0.4999649 0.0000000 0.1667174 0.0000000 0.0000000 0.3333177
#
#       $extsurv.Sx
#       [1] 0.5000351 0.5000351 0.3333177 0.3333177 0.3333177 0.0000000
#
#       $konstdist
#       [1] 0.4999365 0.4999365 0.6000000 0.6000000 0.6000000 1.0000000
#
#       $konstjump
#       [1] 0.4999365 0.0000000 0.1000635 0.0000000 0.0000000 0.4000000
#
#       $konsttime
#       [1] 3.5
#
#       $theta
#       [1] 0.6
#
#       $"-2loglikR"                  (the Wilks statistics to test Ho:
#       [1] 0.05679897                  F(K)=konst)
#
#       $maxiter
#       [1] 33
```

```
#
#  The Wilks statistic is 0.05679897, there is no evidence against Ho: F(3.5)=0.6
```

IVaids *Data: AIDS patient among IV drug user*

Description

Time to AIDS among 232 patients infected with HIV. 136 left AIDS-free (right censored). 14 died with AIDS without prior diagnoses (left censored). 82 had AIDS while in the program (non-censored).

Usage

```
data(IVaids)
```

Details

A date set with 232 rows and 7 variables.

"A doubly censoring scheme occurs when the lifetimes T being measured, from a well-known time origin, are exactly observed only within a window [L, R] of observational time and are otherwise censored either from above by R (right-censored observations) or below by L (left-censored observations).

Sample data consists on the pairs (U, delta) where U = min[R, max(T, L)] and delta indicates whether T is exactly observed (delta = 0), right-censored (delta = 1) or left-censored (delta = -1). We are interested in the estimation of the marginal behaviour of the three random variables T, L and R based on the observed pairs (U, delta)." —— quote from the below reference paper.

The definition of the censoring indicator, (delta), here is different from the one we use, (d), in the functions d011( ) and d011ch( ).

- (delta=0) corresponds to (d=1);

- (delta=1) corresponds to (d=0);

- (delta= -1) corresponds to (d=2).

Therefore we need to make the change before call the function d011( ) or d011ch( ).

"The data set is from a cohort of drug users recruited in a detoxification program in Badalona (Spain). For these data we may estimate the survival function for the elapsed time from starting IV-drugs to AIDS diagnosis, as well as the potential follow-up time." ——quote from the below reference paper.

The entry "AIDSDate-FIRST_IV" is the observed AIDS-free times, U, in the above definition. The unit of measurement is "days".

According to the paper the estimated median of U is 15.44 years. And at time of 10 years, the probability of AIDS-free is about 0.7. (from my reading of the plot from the paper, this probability is more closer to 0.75).

References

Julia, Olga and Gomez, Guadalupe (2011) Simultaneous marginal survival estimators when doubly censored data is present. Lifetime Data Analysis, July 2011, Volume 17, Issue 3, pp 347-372.

Wdataclean2 *Internal dblcens functions*

Description

Internal dblcens functions

Usage

```
Wdataclean2(z, d, wt=rep(1,length(z)))
```

Arguments

`z`	a vector of length n denoting observed times, (ties permitted)
`d`	a vector of length n that contains censoring indicator: d = 2 or 1 or 0, (according to z being left, not, right censored)
`wt`	a vector of length n that is used to derive the number of ties. By default it is a “1” vector of length n.

Details

These are not intended to be called by the user.

`Wdataclean2` will sort the data and collaps those that are true ties, and the number of tied value is in the weights. Same code as in the package emplik.

Value

`value`	Cleaned survival times.
`d`	Cleaned Censoring status of the above times.
`weight`	a vector that indicate the number of ties.

Examples

```
z <- c(0.312 ,0.808 ,0.793 ,2.119 ,0.152 ,0.104 ,1.002 ,0.82 ,0.356 ,0.618)
d <- c(1, 0, 0, 0, 0, 0, 1, 0, 0, 0)
Wdataclean2(z,d)
```

Index