Package ‘npsurv’

July 22, 2025

Title Nonparametric Survival Analysis

Version 0.5-0

Date 2020-09-19

Author Yong Wang

Maintainer Yong Wang <`yongwang@auckland.ac.nz`>

Depends lsei

Imports methods

Description Non-parametric survival analysis of exact and interval-censored observations. The methods implemented are developed by Wang (2007) <doi:10.1111/j.1467-9868.2007.00583.x>, Wang (2008) <doi:10.1016/j.csda.2007.10.018>, Wang and Taylor (2013) <doi:10.1007/s11222-012-9341-9> and Wang and Fani (2018) <doi:10.1007/s11222-017-9724-z>.

Encoding UTF-8

License GPL (>= 2)

URL `https://www.stat.auckland.ac.nz/~yongwang/`

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-09-20 14:40:11 UTC

Contents

`acfail` *Air Conditioner Failure Data*

Description

Contains the number of operating hours between successive failure times of the air conditioning systems in Boeing airplanes

Format

A numeric vector storing the failure times.

Source

Proschan (1963)

References

Proschan, F. (1963). Theoretical explanation of observed decreasing failure rate. *Technometrics*, **5**, 375-383.

See Also

`Uhaz`.

Examples

```
data(acfail)
r = Uhaz(acfail, deg=2)
plot(r$h, fn="h")
plot(r$h, fn="d")
```

`ap` *Angina Pectoris Survival Data*

Description

Contains the survival times in years from the time of diagnosis for 2418 male patients with angina pectoris. Some patients are lost to follow-up, hence giving right-censored observations. Each integer-valued survival time is treated as being censored within a one-year interval.

Format

A data frame with 30 observations and 3 variables:

`L`: left-end point of an interval-censored retraction time;

`R`: right-end point of an interval-censored retraction time;

`count`: number of patients in the interval.

Source

Lee and Wang (2003), page 92.

References

Lee, E. T. and Wang, J. W. (2003). *Statistical Methods for Survival Data Analysis*. Wiley.

See Also

`npsurv`.

Examples

```
data(ap)
r = Uhaz(ap, deg=2)           # smooth U-shaped hazard
plot(r$h, fn="h")             # hazard
plot(r$h, fn="d")             # density

# NPMLE and shape-restricted estimation
plot(npsurv(ap), fn="s")      # survival under no shape restriction
plot(r$h, fn="s", add=TRUE)   # survival with smooth U-shaped hazard
```

`cancer` *Breast Retraction Times after Beast Cancer Treatments.*

Description

Contains the breast retraction times in months for 94 breast cancer patients who received either radiation therapy or radiation therapy plus adjuvant chemotherapy.

Format

A data frame with 94 observations and 3 variables:

L: left-end points of the interval-censored retraction times;

R: right-end points of the interval-censored retraction times;

group: either `RT` (radiation therapy) or `RCT` (radiation therapy plus adjuvant chemotherapy).

Source

Finkelstein and Wolfe (1985).

References

Finkelstein, D. M. and R. A. Wolfe (1985). A semiparametric model for regression analysis of interval-censored failure time data. *Biometrics*, **41**, pp.933-945.

See Also

`npsurv`.

Examples

```
data(cancer)
i = cancer$group == "RT"
plot(npsurv(cancer[i,1:2]), xlim=c(0,60))
plot(npsurv(cancer[!i,1:2]), add=TRUE, col="green3")
```

`Deltamatrix` *Delta matrix*

Description

`Deltamatrix` computes the Delta matrix, along with maximal intersection intervals, for a set of intervals.

Usage

```
Deltamatrix(LR)
```

Arguments

`LR`	two-column matrix, each row of which stores an censoring interval of the form $(L_i, R_i]$. If $L_i = R_i$, it is an exact observation.

Details

An intersection interval is a nonempty intersection of any combination of the given intervals, and a maximal intersection interval is an intersection interval that contains no other intersection interval.

The Delta matrix is a matrix of indicators (`TRUE` or `FALSE`). The rows correspond to the given interval-censored observations, and the columns the maximal intersection intervals. A `TRUE` value of the (i,j)-th element means that the i-th observation covers the j-th maximal intersection interval, and a `FALSE` value means the opposite.

Value

A list with components:

`left`	left endpoints of the maximal intersection intervals.
`right`	right endpoints of the maximal intersection intervals.
`Delta`	logical matrix, for the Delta matrix.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. (2008). Dimension-reduced nonparametric maximum likelihood computation for interval-censored data. *Computational Statistics & Data Analysis*, **52**, 2388-2402.

See Also

`icendata`, `idf`.

Examples

```
(x = cbind(1:5,1:5*3-2))
Deltamatrix(x)
```

`gastric` *Gastric Cancer Survival Data*

Description

Contains the survival times of 45 gastrointestinal tumor patients who were treated with both chemotherapy and radiotherapy. It has both exact and right-censored observations.

Format

A data frame with 30 observations and 3 variables:

L: left-end points of the interval-censored survival times;

R: right-end points of the interval-censored survival times.

Source

Klein and Moeschberger (2003), page 224.

References

Klein, J. P. and Moeschberger, M. L. (2003). *Survival Analysis: Techniques for Censored and Truncated Data (2nd ed.).* Springer.

See Also

`npsurv`, `Uhaz`.

Examples

```
data(gastric)
plot(npsurv(gastric), col="grey")      # survival function
plot(h0<-Uhaz(gastric, deg=0)$h, fn="s", add=TRUE, col="green3")
plot(h1<-Uhaz(gastric, deg=1)$h, fn="s", add=TRUE)
plot(h2<-Uhaz(gastric, deg=2)$h, fn="s", add=TRUE, col="red3")

plot(h0, fn="h", col="green3")         # hazard function
plot(h1, fn="h", add=TRUE)
plot(h2, fn="h", add=TRUE, col="red3")

plot(h0, fn="d", col="green3")         # density function
plot(h1, fn="d", add=TRUE)
plot(h2, fn="d", add=TRUE, col="red3")
```

`hazuh` *Distributional Functions given a U-shaped Hazard Function*

Description

Given an object of class `uh`:

Usage

```
hazuh(t, h)
chazuh(t, h)
survuh(t, h)
denuh(t, h)
```

Arguments

`t`	time points at which the function is to be evaluated.
`h`	an object of class `uh`.

Details

`hazuh` computes the hazard values;

`chazuh` computes the cumulative hazard values;

`survuh` computes the survival function values;

`denuh` computes the density function values.

Value

A numeric vector of the function values.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. and Fani, S. (2018). Nonparametric maximum likelihood computation of a U-shaped hazard function. *Statistics and Computing*, **28**, 187-200.

See Also

`Uhaz`, `icendata`, `plot.uh`

Examples

```
data(ap)
h = Uhaz(icendata(ap), deg=2)$h
hazuh(0:15, h)     # hazard
chazuh(0:15, h)    # cumulative hazard
survuh(0:15, h)    # survival probability
denuh(0:15, h)     # density
```

`icendata` *Class of Interval-censored Data*

Description

Class `icendata` can be used to store general interval-censored data, which may possibly contain exact observations.There are several functions associated with the class.

Usage

```
icendata(x, w=1)
is.icendata(x)
```

Arguments

`x`	vector or matrix.
`w`	weights or multiplicities of the observations.

Details

Function `icendata` creates an object of class 'icendata', which can be used to save both interval-censored and exact observations.

Function `is.icendata` simply checks if an object is of class 'icendata'.

If `x` is a vector, it contains only exact observations, with weights given in `w`.

If `x` is a two-column matrix, it contains interval-censored observations and stores their left and right endpoints in the first and second column, respectively. If the left and right endpoints are equal, then the observation is exact. Weights are provided by `w`.

If `x` is a three-column matrix, it contains interval-censored observations and stores their left and right endpoints in the first and second column, respectively. The weight of each observation is the third-column value multiplied by the corresponding weight value in `w`.

It is useful to turn interval-censored (and exact) observations into the format imposed by `icendata` so that they can be processed in a standardized format by other functions. Also, exact and interval-censored observations are stored separately in this format and can hence be dealt with more easily. Most functions in the package `npsurv` first ensure that the data has this format before processing.

Observations of zero weights are removed. Identical observations are aggregated.

An interval-valued observation is either $(L_i, R_i]$ if $L_i < R_i$, or $[L_i, R_i]$ if $L_i = R_i$.

Value

`t`	numeric vector, storing exact observations.
`wt`	numeric vector, storing the weights of exact observations.
`o`	two-column numeric matrix, storing interval-censored observations.
`wo`	numeric vector, storing the weights of interval-censored observations.
`i1`	logical vector, indicating whether exact observations are less than `upper`.
`upper`	the largest finite value of `t` and `o`.
`u`	numeric vector, containing 0 and all unique finite values in `t` and `o`.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. (2008). Dimension-reduced nonparametric maximum likelihood computation for interval-censored data. Computational Statistics & Data Analysis, 52, 2388-2402.

Wang, Y. and Fani, S. (2017). Nonparametric maximum likelihood computation of a U-shaped hazard function. *Statistics and Computing*, (in print).

See Also

`npsurv`, `Uhaz`.

Examples

```
data(ap)
(x = icendata(ap))
is.icendata(x)

data(gastric)
icendata(gastric)

data(leukemia)
i = leukemia[,"group"] == "6-MP"
icendata(leukemia[i,1:2])
```

`idf` *Interval Distribution Function*

Description

Class `idf` can be used to store a distribution function defined on a set of intervals. There are several functions associated with the class.

Usage

```
idf(left, right, p)
## S3 method for class 'idf'
print(x, ...)
```

Arguments

`left`, `right`	left and right endpoints of intervals on which the distribution function is defined.
`p`	probabilities allocated to the intervals. Probability values will be normalized inside the function.
`x`	an object of class `idf`.
`...`	other arguments for printing.

Details

`idf` creates an object of class `idf`. An `idf` object stores a distribution function defined on a set of intervals.

When left and right endpoints are identical, the intervals just represent exact points.

`print.idf` prints an object of class `idf` as a three-coumn matrix.

Value

`left`, `right`	left and right endpoints of intervals on which the distribution function is defined.
`p`	probabilities allocated to the intervals.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

See Also

`icendata`, `Deltamatrix`, `npsurv`.

Examples

```
idf(1:5, 1:5*3-2, c(1,1,2,2,4))
npsurv(cbind(1:5, 1:5*3-2))$f    # NPMLE
```

`km` *Kaplan-Meier Estimation*

Description

`km` computes the nonparametric maximum likelihood esimate (NPMLE) of a survival function for right-censored data.

Usage

```
km(data, w = 1)
```

Arguments

`data`	vector or matrix, or an object of class `icendata`.
`w`	weights/multiplicities of observations.

Details

For details about the arguments, see `icendata`.

Value

A list with components:

`f`	NPMLE, an object of class `idf`.
`ll`	log-likelihood value of the NPMLE `f`.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Kaplan, E. L. and Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, **53**, 457-481.

See Also

`icendata`, `npsurv`, `idf`.

Examples

```
x = cbind(1:5, c(1,Inf,3,4,Inf))
(f = km(x)$f)
plot(f)

data(leukemia)
i = leukemia[,"group"] == "Placebo"
plot(km(leukemia[i,1:2])$f, xlim=c(0,40), col="green3") # placebo
plot(km(leukemia[!i,1:2])$f, add=TRUE)                  # 6-MP
```

`leukemia` *Remission Times for Acute Leukemia Patients*

Description

Contains remission times in weeks of 42 acute leukemia patients, who received either the treatment of drug 6-mercaptopurine or the placebo treatment. Each remission time is either exactly observed or right-censored.

Format

A data frame with 42 observations and 3 variables:

L: left-end points of the interval-censored remission times in weeks;

R: right-end points of the interval-censored remission times;

group: either 6-MP (6-mercaptopurine) or Placebo.

Source

Freireich et al. (1963).

References

Freireich, E. O. et al. (1963). The effect of 6-mercaptopmine on the duration of steroid induced remission in acute leukemia. *Blood*, **21**, 699-716.

See Also

`npsurv`.

Examples

```
data(leukemia)
i = leukemia[,"group"] == "Placebo"
plot(npsurv(leukemia[i,1:2]), xlim=c(0,40), col="green3") # placebo
plot(npsurv(leukemia[!i,1:2]), add=TRUE)                  # 6-MP

## Treat each remission time as interval-censored:
```

```
x = leukemia
ii = x[,1] == x[,2]
x[ii,2] = x[ii,1] + 1
plot(npsurv(x[i,1:2]), xlim=c(0,40), col="green3")        # placebo
plot(npsurv(x[!i,1:2]), add=TRUE)                         # 6-MP
```

`logLikuh` *Computes the Log-likelihood Value of a U-shaped Hazard Function*

Description

`logLikuh` returns the log-likelihood value of a U-shaped hazard function, given a data set.

Usage

```
logLikuh(h, data)
```

Arguments

`h`	an object of class `uh`.
`data`	numeric vector or matrix for exact or interval-censored observations, or an object of class `icendata`.

Value

Log-likelihood value evaluated at `h`, given `data`.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. and Fani, S. (2018). Nonparametric maximum likelihood computation of a U-shaped hazard function. *Statistics and Computing*, **28**, 187-200.

See Also

`Uhaz`, `icendata`, `plot.uh`

Examples

```
data(ap)
(h0 = uh(.2, NULL, NULL, NULL, NULL, 15, 1))   # Uniform hazard
plot(h0, ylim=c(0,.3))
logLikuh(h0, ap)

r = Uhaz(ap, deg=2)
r$ll
logLikuh(r$h, ap)
plot(r$h, add=TRUE, col="red3")
```

`marijuana` *Angina Pectoris Survival Data*

Description

Contains the answers of 191 California high school students to the question: "When did you first use marijuana?". An answer can be an exact age, or "I have never used it", which gives rise to a right-censored observation, or "I have used it but cannot recall just when the first time was", which gives rise to a left-censored observation.

Format

A data frame with 21 observations and 3 variables:

L: left-end point of an interval-censored time;

R: right-end point of an interval-censored time;

count: number of students in the interval.

Source

Turnbull and Weiss (1978). See also Klein and Moeschberger (1997), page 17.

References

Turnbull and Weiss (1978). A likelihood ratio statistic fortesting goodness of fit with randomly censored data. *Biometrics*, **34**, 367-375.

Klein and Moeschberger (2003). *Survival Analysis: Techniques for Censored and Truncated Data* (2nd ed.). Springer

See Also

`npsurv`.

Examples

```
data(marijuana)
r = Uhaz(marijuana, deg=2)
plot(r$h, fn="h")
plot(r$h, fn="s")
```

---

npsurv *Nonparametric Survival Function Estimation*

Description

`npsurv` computes the nonparametric maximum likelihood esimate (NPMLE) of a survival function for general interval-censored data.

Usage

```
npsurv(data, w = 1, maxit = 100, tol = 1e-06, verb = 0)
```

Arguments

`data`	vector or matrix, or an object of class `icendata`.
`w`	weights or multiplicities of the observations.
`maxit`	maximum number of iterations.
`tol`	tolerance level for stopping the algorithm. It is used as the threshold on the increase of the log-likelihood after each iteration.
`verb`	verbosity level for printing intermediate results at each iteration.

Details

If `data` is a vector, it contains only exact observations, with weights given in `w`.

If `data` is a matrix with two columns, it contains interval-censored observations, with the two columns storing their left and right end-points, respectively. If the left and right end-points are equal, then the observation is exact. Weights are provided by `w`.

If `data` is a matrix with three columns, it contains interval-censored observations, with the first two columns storing their left and right end-points, respectively. The weight of each observation is the third-column value multiplied by the corresponding weight value in `w`.

The algorithm used for computing the NPMLE is either the constrained Newton method (CNM) (Wang, 2008), or the hierachical constrained Newton method (HCNM) (Wang and Taylor, 2013) when there are a large number of maximal intersection intervals.

Inside the function, it examines if data has only right censoring, and if so, the Kaplan-Meier estimate is computed directly by function `km`.

An interval-valued observation is either $(L_i, R_i]$ if $L_i < R_i$, or $[L_i, R_i]$ if $L_i = R_i$.

Value

An object of class `npsurv`, which is a list with components:

`f`	NPMLE, an object of class `idf`.
`upper`	largest finite value in the data.
`convergence`	= `TRUE`, converged successfully; = `FALSE`, maximum number of iterations reached.
`method`	method used internally, either `cnm` or `hcnm`.
`ll`	log-likelihood value of the NPMLE `f`.
`maxgrad`	maximum gradient value of the NPMLE `f`.
`numiter`	number of iterations used.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. (2008). Dimension-reduced nonparametric maximum likelihood computation for interval-censored data. *Computational Statistics & Data Analysis*, **52**, 2388-2402.

Wang, Y. and Taylor, S. M. (2013). Efficient computation of nonparametric survival functions via a hierarchical mixture formulation. *Statistics and Computing*, **23**, 713-725.

See Also

`icendata`, `Deltamatrix`, `idf`, `km`.

Examples

```
## all exact observations
data(acfail)
plot(npsurv(acfail))

## right-censored (and exact) observations
data(gastric)
plot(npsurv(gastric))

data(leukemia)
i = leukemia[,"group"] == "Placebo"
plot(npsurv(leukemia[i,1:2]), xlim=c(0,40), col="blue") # placebo
plot(npsurv(leukemia[!i,1:2]), add=TRUE, col="red")     # 6-MP

## purely interval-censored data
data(ap)
plot(npsurv(ap))

data(cancer)
cancerRT = with(cancer, cancer[group=="RT",1:2])
```

```
plot(npsurv(cancerRT), xlim=c(0,60))                 # survival of RT
cancerRCT = with(cancer, cancer[group=="RCT",1:2])
plot(npsurv(cancerRCT), add=TRUE, col="green")       # survival of RCT
```

---

nzmort *New Zealand Mortality in 2000*

Description

Contains the number of deaths of Maori and Non-Maori people at each age in New Zealand in 2000.

Format

A data frame with 210 observations and 3 variables:

age: at which age the deaths occurred;

deaths: number of people died at the age;

ethnic: either Maori or Non-Maori.

Details

Data contains no age with zero death.

Source

https://www.mortality.org/

See Also

`Uhaz`.

Examples

```
data(nzmort)
x = with(nzmort, nzmort[ethnic=="maori",])[,1:2]      # Maori mortality
# x = with(nzmort, nzmort[ethnic!="maori",])[,1:2]    # Non-Maori mortality

## As exact observations
# Plot hazard functions
h0 = Uhaz(x[,1]+0.5, x[,2], deg=0)$h    # U-shaped hazard
plot(h0, fn="h", col="green3", pch=2)
h1 = Uhaz(x[,1]+0.5, x[,2], deg=1)$h    # convex hazard
plot(h1, fn="h", add=TRUE, pch=1)
h2 = Uhaz(x[,1]+0.5, x[,2], deg=2)$h    # smooth U-shaped hazard
plot(h2, fn="h", add=TRUE, col="red3")

# Plot densities
age = 0:max(x[,1])
```

```
count = integer(length(age))
count[x[,"age"]+1] = x[,"deaths"]
barplot(count/sum(count), space=0, col="lightgrey", ylab="Density")
axis(1, pos=NA, at=0:10*10)
plot(h0, fn="d", add=TRUE, col="green3", pch=2)
plot(h1, fn="d", add=TRUE, col="blue3", pch=1)
plot(h2, fn="d", add=TRUE, col="red3", pch=19)

## As interval-censored observations
# Plot hazard functions
x2 = cbind(x[,1], x[,1]+1, x[,2])
h0 = Uhaz(x2, deg=0)$h      # U-shaped hazard
plot(h0, fn="h", col="green3", pch=2)
h1 = Uhaz(x2, deg=1)$h      # convex hazard
plot(h1, fn="h", add=TRUE, pch=1)
h2 = Uhaz(x2, deg=2)$h      # smooth U-shaped hazard
plot(h2, fn="h", add=TRUE, col="red3", pch=1)

# Plot densities
barplot(count/sum(count), space=0, col="lightgrey")
axis(1, pos=NA, at=0:10*10)
plot(h0, fn="d", add=TRUE, col="green3", pch=2)
plot(h1, fn="d", add=TRUE, col="blue3", pch=1)
plot(h2, fn="d", add=TRUE, col="red3", pch=19)
```

`plot.npsurv` *Plot Functions for Nonparametric Survival Estimation*

Description

Functions for plotting nonparametric survival functions and related ones.

Usage

```
## S3 method for class 'npsurv'
plot(x, ...)
## S3 method for class 'idf'
plot(x, data, fn=c("surv","grad"), ...)
plotsurvidf(f, style=c("box","uniform","left","right","midpoint"),
            xlab="Time", ylab="Survival Probability", col="blue3", fill=0,
            add=FALSE, lty=1, lty.inf=2, xlim, ...)
plotgradidf(f, data, w=1, col1="red3", col2="blue3",
            xlab="Survival Time", ylab="Gradient", xlim, ...)
```

Arguments

`x`	an object of class `npsurv` (i.e., an output of function `npsurv`) or an object of class `idf`.

`...`	arguments for other graphical parameters (see `par`).
`fn`	either "surv" or "grad", to indicate plotting either the survival or the gradient function.
`f`	an object of class `idf`.
`style`	for how to plot the survival function on a "maximal intersection interval": = `box`, plot a rectangle, which shows the uncertainty of probability allocation within the interval; = `uniform`, treat it as a uniform distribution and hence the diagonal line of the rectangle is plotted; = `left`, plot only the left side of the rectangle; = `right`, plot only the right side of the rectangle; = `midpoint`, plot a vertical line at the midpoint of the interval.
`xlab`, `ylab`	x- or y-axis label.
`add`	= `TRUE`, adds the curve to the existing plot; = `FALSE`, plots the curve in a new one.
`col`	color for all line segments, including box/rectangle borders.
`fill`	color for filling a box/rectangle. By default, a lighter semi-transparent color is used.
`lty`	line type
`lty.inf`	line type for the rectangle that may extend to infinity.
`data`	vector or matrix that stores observations, or an object of class `icendata`.
`w`	additional weights/multiplicities of the observations stored in `x`.
`col1`	color for drawing maximal intersection intervals allocated with positive probabilities.
`col2`	color for drawing all gradients and the maximal intersection intervals allocated with zero probabilities.
`xlim`	x-coordinate limit points.

Details

`plot.npsurv` and `plot.idf` are wrapper functions that call either `plotsurvidf` or `plotgradidf`.

`plotsurvidf` plots the survival function of the nonparametric maximum likelihood estimate (NPMLE).

`plotgradidf` plots the gradient function of the NPMLE.

`plotsurvidf` by default chooses a less saturated color for `fill` than `col`.

`plotgradidf` plots gradient values as vertical lines located as the left endpoints of the maximal intersection intervals. Each maximal intersection interval is plotted as a wider line on the horizontal zero-gradient line, with a circle to represent the open left endpoint of the interval and a solid point the closed right endpoint of the interval. The maximal intersection intervals allocated with positive probabilities have zero gradients, and hence no vertical lines are drawn for them.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. (2008). Dimension-reduced nonparametric maximum likelihood computation for interval-censored data. *Computational Statistics & Data Analysis*, **52**, 2388-2402.

See Also

`icendata`, `idf`, `npsurv`.

Examples

```
data(ap)
plot(r<-npsurv(ap))              # survival function
plot(r$f, ap, fn="g")            # all gradients virtually zeros.

data(cancer)
cancerRT = with(cancer, cancer[group=="RT",1:2])
plot(rt<-npsurv(cancerRT), xlim=c(0,60))                 # survival of RT
cancerRCT = with(cancer, cancer[group=="RCT",1:2])
plot(rct<-npsurv(cancerRCT), add=TRUE, col="green3") # survival of RCT
## as uniform dististrbutions.
plot(rt, add=TRUE, style="uniform", col="blue3")
plot(rct, add=TRUE, style="uniform", col="green3")

## plot gradients; must supply data
plot(rt, cancerRT, fn="g")        # for group RT
plotgradidf(rct$f, cancerRCT)   # or, for group RCT
```

`plot.Uhaz` *Plot Functions for U-shaped Hazard Estimation*

Description

Functions for plotting various functions in U-shaped hazard estimation

Usage

```
## S3 method for class 'Uhaz'
plot(x, ...)
## S3 method for class 'uh'
plot(x, data, fn=c("haz","grad","surv","den","chaz"), ...)
plothazuh(h, add=FALSE, col="darkblue", lty=1, xlim, ylim,
          lwd=2, pch=19, len=500, vert=FALSE, add.knots=TRUE,
          xlab="Time", ylab="Hazard", ...)
plotchazuh(h, add=FALSE, lwd=2, len=500, col="darkblue",
           pch=19, add.knots=TRUE, vert=FALSE, xlim, ylim, ...)
plotdenuh(h, add=FALSE, lty=1, lwd=2, col="darkblue",
          add.knots=TRUE, pch=19, ylim, len=500, vert=FALSE, ...)
```

```
plotsurvuh(h, add=FALSE, lty=1, lwd=2, len=500, vert=FALSE,
           col="darkblue", pch=19, add.knots=TRUE, xlim, ylim, ...)
plotgraduh(h, data, w=1, len=500, xlim, ylim, vert=TRUE,
           add=FALSE, xlab="Time", ylab="Gradient",
           col0="red3", col1="blue3", col2="green3", order=0, ...)
```

Arguments

`x`	an object of class `Uhaz`, i.e., an output of function `Uhaz`, or an object of class `uh`..
`...`	arguments for other graphical parameters (see `par`).
`h`	an object of class `uh`.
`data`	vector or matrix that stores observations, or an object of class `icendata`.
`w`	additional weights/multiplicities for the observations stored in `data`.
`fn`	function to be plotted. It can be = `haz`, for hazard function; = `chaz`, for cumulative hazard function; = `den`, for density function; = `surv`, for survival function; = `gradient`, for gradient functions.
`xlim`, `ylim`	numeric vectors of length 2, giving the x and y coordinates ranges.
`xlab`, `ylab`	x- or y-axis labels.
`add`	= `TRUE`, adds the curve to the existing plot; = `FALSE`, plots the curve in a new one.
`col`	color used for plotting the curve.
`lty`	line type for plotting the curve.
`lwd`	line width for plotting the curve.
`len`	number of points used to plot a curve.
`add.knots`	logical, indicating if knots are also plotted.
`pch`	point character/type for plotting knots.
`vert`	logical, indicating if grey vertical lines are plotted to show the interval that separates the two discrete measures.
`col0`	color for gradient function 0, i.e., for the hazard-constant part, or alpha.
`col1`	color for gradient function 1, i.e., for the hazard-decreasing part.
`col2`	color for gradient function 1, i.e., for the hazard-increasing part.
`order`	= 0, the gradient functions are plotted; = 1, their first derivatives are plotted; = 2, their second derivatives are plotted.

Details

`plot.Uhaz` and `plot.uh` are wrapper functions that can be used to invoke `plot.hazuh`, `plot.chazuh`, `plot.survuh`, `plot.denuh` or `plot.graduh`.

`plothazuh` plots a U-shaped hazard function.

`plotchazuh` plots a cumulative hazard function that has a U-shaped hazard function.

`plotsurvuh` plots the survival function that has a U-shaped hazard function.

`plotdenuh` plots the density function that has a U-shaped hazard function.

`plotgraduh` plots the gradient function that has a U-shaped hazard function.

A U-shaped hazard function is given by

$$h(t) = \alpha + \sum_{j=1}^{k} \nu_j (\tau_j - t)_+^p + \sum_{j=1}^{m} \mu_j (t - \eta_j)_+^p,$$

where $\alpha, \nu_j, \mu_j \geq 0$, $\tau_1 < \cdots < \tau_k \leq \eta_1 < \cdots < \eta_m$, and $p \geq 0$.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Wang, Y. and Fani, S. (2018). Nonparametric maximum likelihood computation of a U-shaped hazard function. *Statistics and Computing*, **28**, 187-200.

See Also

`icendata`, `uh`, `npsurv`.

Examples

```
## Angina Pectoris Survival Data
data(ap)
plot(r<-Uhaz(ap))              # hazard function for a convex hazard
plot(r, fn="c")                # cumulative hazard function
plot(r, fn="s")                # survival function
plot(r, fn="d")                # density function
plot(r, ap, fn="g")            # gradient functions
plot(r, ap, fn="g", order=1)   # first derivatives of gradient functions
plot(r, ap, fn="g", order=2)   # second derivatives of gradient functions

## New Zealand Mortality in 2000
data(nzmort)
i = nzmort$ethnic == "maori"
x = nzmort[i,1:2]                            # Maori mortality
h = Uhaz(x[,1]+0.5, x[,2], deg=2)$h          # smooth U-shaped hazard
plot(h)                        # hazard function
plot(h, fn="d")                # density function
plot(h, fn="s")                # survival function
```

```
x2 = nzmort[!i,1:2]                        # Non-Maori mortality
h2 = Uhaz(x2[,1]+0.5, x2[,2], deg=2)$h
plot(h2, fn="s", add=TRUE, col="green3")
```

`uh` *U-shaped Hazard Function*

Description

Class `uh` can be used to store U-shaped hazard functions. There are a couple of functions associated with the class.

Usage

```
uh(alpha, tau, nu, eta, mu, upper=Inf, deg=1, collapse=TRUE)
## S3 method for class 'uh'
print(x, ...)
```

Arguments

`alpha`	a nonnegative value, for the constant coefficient.
`tau`	vector of nonnegative real values, for left knots.
`nu`	vector of nonnegative values, for masses associated with the left knots.
`eta`	vector of nonnegative real values, for right knots.
`mu`	vector of nonnegative real values, for masses associated with the right knots.
`upper`	a positive value, at which point the hazard starts to become infinite.
`deg`	nonnegative real number for spline degree (i.e., p in the formula below).
`collapse`	logical, indicating if identical knots should be collapsed.
`x`	an object of class `uh`.
`...`	other auguments for printing.

Details

`uh` creates an object of class `uh`, which stores a U-shaped hazard function.

`print.uh` prints an object of class `uh`.

A U-shape hazard function, as generalized by Wang and Fani (2018), is given by

$$h(t) = \alpha + \sum_{j=1}^{k} \nu_j (\tau_j - t)_+^p + \sum_{j=1}^{m} \mu_j (t - \eta_j)_+^p,$$

where $\alpha, \nu_j, \mu_j \geq 0$, $\tau_1 < \cdots < \tau_k \leq \eta_1 < \cdots < \eta_m$, and $p \geq 0$ is the the spline degree which determines the smoothness of the U-shaped hazard. As p increases, the family of hazard functions becomes increasingly smoother, but at the same time, smaller. When $p = 0$, the hazard function is U-shaped, as studied by Bray et al. (1967). When $p = 1$, the hazard function is convex, as studied by Jankowski and Wellner (2009a,b).

`print.uh` prints an object of class `uh`. While `alpha`, `upper` and `deg` are printed as they are, `tau` and `nu` are printed as a two-column matrix, and so are `eta` and `mu`.

Value

`uh` returns an object of class `uh`. It is a list with components `alpha`, `tau`, `nu`, `eta`, `mu`, `upper` and `deg`, which store their corresponding values as described above.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Bray, T. A., Crawford, G. B., and Proschan, F. (1967). *Maximum Likelihood Estimation of a U-shaped Failure Rate Function*. Defense Technical Information Center.

Jankowski, H. K. and Wellner, J. A. (2009a). Computation of nonparametric convex hazard estimators via profile methods. *Journal of Nonparametric Statistics*, **21**, 505-518.

Jankowski, H. K. and Wellner, J. A. (2009b). Nonparametric estimation of a convex bathtub-shaped hazard function. *Bernoulli*, **15**, 1010-1035.

Wang, Y. and Fani, S. (2018). Nonparametric maximum likelihood computation of a U-shaped hazard function. *Statistics and Computing*, **28**, 187-200.

See Also

`Uhaz`, `icendata`, `plot.uh`

Examples

```
(h0 = uh(3, 2, 3, 4, 5, 7, deg=0))              # deg = 0
plot(h0, ylim=c(0,20))
(h1 = uh(4, 2, 3, 5, 6, 7, deg=1))              # deg = 1
plot(h1, add=TRUE, col="green3")
(h2 = uh(1, 1:2, 3:4, 5:6, 7:8, 9, deg=2))      # deg = 2
plot(h2, add=TRUE, col="red3")
```

`Uhaz` *U-shaped Hazard Function Estimation*

Description

`Uhaz` computes the nonparametric maximum likelihood esimate (NPMLE) of a U-shaped hazard function from exact or interval-censored data, or a mix of the two types of data.

Usage

```
Uhaz(data, w = 1, deg = 1, maxit = 100, tol = 1e-06, verb = 0)
```

Arguments

`data`	vector or matrix, or an object of class `icendata`.
`w`	weights or multiplicities of the observations.
`deg`	nonnegative real number for spline degree (i.e., p in the formula below).
`maxit`	maximum number of iterations.
`tol`	tolerance level for stopping the algorithm. It is used as the threshold on the increase of the log-likelihood after each iteration.
`verb`	verbosity level for printing intermediate results in each iteration.

Details

If `data` is a vector, it contains only exact observations, with weights given in `w`.

If `data` is a matrix with two columns, it contains interval-censored observations, with the two columns storing their left and right end-points, respectively. If the left and right end-points are equal, then the observation is exact. Weights are provided by `w`.

If `data` is a matrix with three columns, it contains interval-censored observations, with the first two columns storing their left and right end-points, respectively. The weight of each observation is the third-column value multiplied by the corresponding weight value in `w`.

The algorithm used for the computing the NPMLE of a hazard function under the U-shape restriction is is proposed by Wang and Fani (2015). Such a hazard function is given by

A U-shaped hazard function is given by

$$h(t) = \alpha + \sum_{j=1}^{k} \nu_j (\tau_j - t)_+^p + \sum_{j=1}^{m} \mu_j (t - \eta_j)_+^p,$$

where $\alpha, \nu_j, \mu_j \geq 0$, $\tau_1 < \cdots < \tau_k \leq \eta_1 < \cdots < \eta_m$, and $p \geq 0$ is the the spline degree which determines the smoothness of the U-shaped hazard. As p increases, the family of hazard functions becomes increasingly smoother, but at the time, smaller. When p = 0, the hazard function is U-shaped, as studied by Bray et al. (1967). When p = 1, the hazard function is convex, as studied by Jankowski and Wellner (2009a,b).

Note that `deg` (i.e., p in the above mathematical display) can take on any nonnegative real value.

Value

An object of class `Uhaz`, which is a list with components:

`convergence`	= `TRUE`, converged successfully; = `FALSE`, maximum number of iterations reached.
`grad`	gradient values at the knots.
`numiter`	number of iterations used.
`ll`	log-likelihood value of the NPMLE `h`.
`h`	NPMLE of the U-shaped hazard function, an object of class `uh`.

Author(s)

Yong Wang <yongwang@auckland.ac.nz>

References

Bray, T. A., Crawford, G. B., and Proschan, F. (1967). *Maximum Likelihood Estimation of a U-shaped Failure Rate Function*. Defense Technical Information Center.

Jankowski, H. K. and Wellner, J. A. (2009a). Computation of nonparametric convex hazard estimators via profile methods. *Journal of Nonparametric Statistics*, **21**, 505-518.

Jankowski, H. K. and Wellner, J. A. (2009b). Nonparametric estimation of a convex bathtub-shaped hazard function. *Bernoulli*, **15**, 1010-1035.

Wang, Y. and Fani, S. (2018). Nonparametric maximum likelihood computation of a U-shaped hazard function. *Statistics and Computing*, **28**, 187-200.

See Also

`icendata`, `nzmort`.

Examples

```
## Interval-censored observations
data(ap)
(r = Uhaz(ap, deg=0))
plot(r, ylim=c(0,.3), col=1)
for(i in 1:6) plot(Uhaz(ap, deg=i/2), add=TRUE, col=i+1)
legend(15, 0.01, paste0("deg = ", 0:6/2), lwd=2, col=1:7, xjust=1, yjust=0)

## Exact observations
data(nzmort)
x = with(nzmort, nzmort[ethnic=="maori",])[,1:2]   # Maori mortality
(h0 = Uhaz(x[,1]+0.5, x[,2], deg=0)$h)     # U-shaped hazard
(h1 = Uhaz(x[,1]+0.5, x[,2], deg=1)$h)     # convex hazard
(h2 <- Uhaz(x[,1]+0.5, x[,2], deg=2)$h)    # smooth U-shaped hazard

plot(h0, pch=2)                            # plot hazard functions
plot(h1, add=TRUE, col="green3", pch=1)
plot(h2, add=TRUE, col="red3", pch=19)
```

```
age = 0:max(x[,1])                          # plot densities
count = integer(length(age))
count[x[,"age"]+1] = x[,"deaths"]
barplot(count/sum(count), space=0, col="lightgrey")
axis(1, pos=NA, at=0:10*10)
plot(h0, fn="d", add=TRUE, pch=2)
plot(h1, fn="d", add=TRUE, col="green3", pch=1)
plot(h2, fn="d", add=TRUE, col="red3", pch=19)

plot(h0, fn="s", pch=2)                     # plot survival functions
plot(h1, fn="s", add=TRUE, col="green3", pch=1)
plot(h2, fn="s", add=TRUE, col="red3", pch=19)

## Exact and right-censored observations
data(gastric)
plot(h0<-Uhaz(gastric, deg=0)$h)            # plot hazard functions
plot(h1<-Uhaz(gastric, deg=1)$h, add=TRUE, col="green3")
plot(h2<-Uhaz(gastric, deg=2)$h, add=TRUE, col="red3")

plot(npsurv(gastric), fn="s", col="grey")  # plot survival functions
plot(h0, fn="s", add=TRUE)
plot(h1, fn="s", add=TRUE, col="green3")
plot(h2, fn="s", add=TRUE, col="red3")
```

Index