Package ‘CPHshape’

February 19, 2015

Type  Package
Title  Find the maximum likelihood estimator of the shape constrained hazard baseline and the effect parameters in the Cox proportional hazards model
Version 1.0.1
Date  2012-02-27
Author Rihong Hui and Hanna Jankowski <hkj@mathstat.yorku.ca>
Maintainer Hanna Jankowski <hkj@mathstat.yorku.ca>
Description This package computes the maximum likelihood estimator (MLE) of the shape-constrained hazard baseline and the effect parameters in the Cox proportional hazards model under IID sampling. We assume that the data are continuous and allow for right censoring. The function 'find.shapeMLE' allows for four different shape constraints: increasing, decreasing, unimodal, and u-shaped.
License GPL (>= 2)
URL http://www.math.yorku.ca/~hkj
NeedsCompilation no
Repository CRAN
Date/Publication 2014-03-07 08:21:11

R topics documented:

CPHshape-package .......................................................... 2
find.H ................................................................. 3
find.hazard ............................................................ 4
find.shapeCPH ......................................................... 5
find.shapeMLE ........................................................ 7
plot.CPHshape ........................................................ 9

Index  10
Compute the MLE in the Cox proportional hazard model with shape constrained baseline

**Description**

This package calculates the semi-parametric maximum likelihood estimator (MLE) of the effect parameters and the nonparametric hazard constrained to be either increasing, decreasing, unimodal, or u-shaped. We assume that the times are continuous, and allow for right-censoring.

**Details**

<table>
<thead>
<tr>
<th>Package:</th>
<th>CPHshape</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>1.0</td>
</tr>
<tr>
<td>Date:</td>
<td>2012-03-06</td>
</tr>
<tr>
<td>License:</td>
<td>GPL (&gt;=2)</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>

The two main function in the package is

**find.shapeCPH**: Compute the maximum likelihood estimator of the effect parameters and the non-parametric shape constrained hazard in the proportional hazard model.

The package also provides the function:

**find.shapeMLE**: Compute the MLE of a nonparametric shape-constrained hazard.

**Author(s)**

Rihong Hui and Hanna Jankowski <hkj@mathstat.yorku.ca>

**References**


### Examples

```r
# random sample from the proportional hazard model
n <- 200
beta1 <- -1
beta2 <- -2
z1 <- rbinom(n, 1, 0.5)
z2 <- runif(n, -1, 1)
w <- exp(beta1 * z1 + beta2 * z2)
x <- rexp(n, rate = 0.3 * w)
delta <- ifelse(x <= 2.5)
x <- pmin(x, 2.5)

# compute MLE
mle <- find.shapeCPH(x, cbind(z1, z2), delta, print = TRUE, type = "decreasing")

# estimates of the effect parameter
mle$beta

# plot resulting estimate of baseline hazard
plot(mle)
abline(h = 0.3, col = "red")  # add true baseline
rug(x)
```

---

### find.H

*Calculate the cumulative hazard function*

**Description**

This function calculates the cumulative hazard function at location t based on input from either `find.shapeCPH` or `find.shapeMLE`.

**Usage**

```r
find.H(t, h.val, h.ranges)
```

**Arguments**

- **t**: time at which to evaluate the cumulative hazard (must be univariate)
- **h.val**: vector specifying changes of values in the hazard baseline MLE (output from e.g. `find.shapeCPH`)
- **h.ranges**: vector specifying locations of changes in values in the hazard baseline MLE (output from e.g. `find.shapeCPH`
Value
A number giving the value of the estimated cumulative hazard at t.

Author(s)
Rihong Hui and Hanna Jankowski <hkj@mathstat.yorku.ca>

See Also
find.shapeCPH find.shapeMLE

Examples
# random sample from the uniform density
n <- 500
x <- runif(n)

# compute MLE of increasing hazard
mle <- find.shapeMLE(x, type="increasing")

# find fitted cumulative hazard function at t=0.2
find.H(0.2, mle$h.val, mle$h.range)

find.hazard Calculate the hazard function

Description
This function calculates the hazard function at location t based on input resulting from either find.shapeCPH or find.shapeMLE.

Usage
find.hazard(t, h.val, h.ranges, type, mode)

Arguments
t time at which to evaluate the hazard (must be univariate)
h.val vector specifying changes of values in the hazard baseline MLE (output from e.g. find.shapeCPH)
h.ranges vector specifying locations of changes in values in the hazard baseline MLE (output from e.g. find.shapeCPH)
type type of shape constrained used (also given as output from e.g. find.shapeCPH)
mode location of mode or antimode (also given as output from e.g. find.shapeCPH).
Not required for the increasing or decreasing shapes.
find.shapeCPH

Value

A number giving the value of the estimated hazard at t.

Author(s)

Rihong Hui and Hanna Jankowski <hkj@mathstat.yorku.ca>

See Also

find.shapeCPH find.shapeMLE

Examples

# random sample from the uniform density
n <- 100
x <- runif(n)

# compute MLE of increasing hazard and evaluate fitted hazard at t=0.2
mle <- find.shapeMLE(x, type="increasing")
find.hazard(0.2, mle$h.val, mle$h.range, type=mle$type)

# compute MLE of unimodal hazard and evaluate fitted hazard at t=0.2
mle <- find.shapeMLE(x, type="unimodal")
find.hazard(0.2, mle$h.val, mle$h.range, type=mle$type, mode=mle$mode)

find.shapeCPH

Compute the MLE of the effect parameters and shape-constrained
baseline in the Cox model

Description

Compute the maximum likelihood estimator (MLE) of the shape-constrained hazard baseline and
the effect parameters in the Cox proportional hazards model under IID sampling. We assume that
the data are continuous and allow for right censoring. The function 'find.shapeMLE' allows for
four different shape constraints: increasing, decreasing, unimodal, and u-shaped.

Usage

find.shapeCPH(x, z, delta, type="increasing",
beta0=rep(1, length(as.matrix(z)[1,])), max.loop=250, eps=1e-5,
eps.beta=1e-5, print=FALSE)
Arguments

- **x**: vector of length n containing the data
- **z**: matrix of size n x p containing the covariate values
- **delta**: logical vector containing the (right) censoring information. If delta_i=1 then the observation was not censored. The default is delta_i=1 for all i, that is, no observations were censored.
- **type**: string indicating type of shape constraint. Options are "increasing", "decreasing", "unimodal", and "ushaped".
- **beta0**: vector of length p containing the starting value of beta for the algorithm. The default is all elements equal to one.
- **max.loop**: maximum number of iterations. The default is 250.
- **eps**: precision for stopping criterion. The default is 1e-05.
- **eps.beta**: precision for second stopping criterion. The default is 1e-05.
- **print**: logical, if TRUE, output from each iteration of the algorithm is shown. The default is FALSE.

Value

A list containing the following elements:

- **beta**: MLE of the effect parameters
- **h.range**: endpoints for the values of the hazard MLE
- **h.val**: values of the hazard MLE
- **phi**: the criterion function \( \Phi \) evaluated at the MLE
- **h**: the cumulative hazard MLE evaluated at the data points
- **mode**: location of the mode (for unimodal) or antimode (for u-shaped). Note that the antimode is not unique, and the midpoint of all possible values is returned.
- **type**: string indicating type of shape constraint used

Note

For the increasing and unimodal setting, the MLE takes the value of infinity at the mode (the largest observation for type="increasing"). In such situations, this value is removed from the likelihood in the maximization process. A similar approach was taken in Gernander (1956).

We note also that for the shapes type="unimodal" and type="ushaped" the algorithm can take some time, especially for larger sample sizes.

Author(s)

Rihong Hui and Hanna Jankowski <hkj@mathstat.yorku.ca>
find.shapeMLE

References


Examples

# random sample from the proportional hazard model
n <- 200
beta1 <- -1
beta2 <- -2
z1 <- rbinom(n, 1, 0.5)
z2 <- runif(n, -1, 1)
w <- exp(beta1*z1 + beta2*z2)
x <- rexp(n, rate = 0.3*w)
delta <- 1*(w <= 2.5)
x <- pmin(x, 2.5)

# compute MLE
mle <- find.shapeMLE(x, cbind(z1, z2), delta, print = TRUE, type = "decreasing")

# estimates of the effect parameter
mle$beta

# plot resulting estimate of baseline hazard
plot(mle)
abline(h = 0.3, col = "red") # add true baseline
rug(x)

find.shapeMLE

Compute the MLE of a shape-constrained hazard function

Description

Compute the maximum likelihood estimator (MLE) of a shape-constrained hazard function under IID sampling. We assume that the data are continuous and allow for right censoring. The function ‘find.shapeMLE’ allows for four different shape constraints: increasing, decreasing, unimodal, and u-shaped.

Usage

find.shapeMLE(x, delta = rep(1, length(x)), type = "increasing", plot = FALSE)
find.shapeMLE

Arguments

- **x**: vector of length n containing the data
- **delta**: logical vector containing the (right) censoring information. If delta_i=1 then the observation was not censored. The default is delta_i=1 for all i, that is, assuming that no observations were censored.
- **type**: string indicating type of shape constraint. Options are "increasing", "decreasing", "unimodal", and "u-shaped".
- **plot**: logical, if TRUE, the graphical representation of the MLE is shown

Value

A list containing the following elements:

- **h.range**: endpoints for the values of the hazard MLE
- **h.val**: values of the hazard MLE in between the endpoints
- **phi**: the criterion function \(\Phi\) (negative log-likelihood) evaluated at the MLE
- **H**: the cumulative hazard MLE evaluated at the data points
- **mode**: location of the mode (for unimodal) or antimode (for u-shaped). Note that the antimode is not unique, and the midpoint of all possible values is returned.
- **type**: string indicating type of shape constraint used

Note

The MLE can be found in different ways. We use the graphical representation via convex minorants or concave majorants of appropriate functions. Also, for the increasing and unimodal setting, the MLE takes the value of infinity at the mode (the largest observation for type="increasing"). In such situations, this value is removed from the likelihood in the maximization process. A similar approach was taken in Gernander (1956).

Author(s)

Rihong Hui and Hanna Jankowski <hkj@mathstat.yorku.ca>

References


Examples

```r
# random sample from the uniform density
n <- 500
x <- runif(n)

# compute MLE of increasing hazard
mle <- find.shapeMLE(x, type="increasing")
```
# plot the fitted hazard
plot(mle)
rug(x)

# add true hazard to the plot
h.true <- function(x) 1/(1-x)
plot(h.true, col="red", add=TRUE)

---

**plot.CPHshape**

*Plot a shape constrained MLE of a hazard function*

**Description**

`plot` method for class "CPHshape".

**Usage**

```r
## S3 method for class 'CPHshape'
plot(x, cex=0.5, xlim=range(x$h.range), ylim=range(x$h.val),
     xlab="", ylab="hazard function", ...)
```

**Arguments**

- `x` object of class "CPHshape" (typically output from either `find.shapeMLE` or `find.shapeCPH`).
- `cex` A numerical value giving the amount by which plotting text and symbols should be magnified relative to the default. In this setting, it changes the size of the dots representing continuity points of the hazard. Default is set to 0.5.
- `xlim` limits of the x axis in the plot. Defaults to full range of data.
- `ylim` limits of the y axis in the plot. Defaults to full range of the MLE.
- `xlab` a label for the x axis. Defaults to empty.
- `ylab` a label for the y axis. Default is set to "hazard function".
- `...` Additional arguments

**Details**

For examples, see `find.shapeMLE` or `find.shapeCPH`.

**Author(s)**

Hanna Jankowski <hkj@mathstat.yorku.ca>
Index

*Topic package
  CPHshape-package, 2

CPHshape (CPHshape-package), 2
CPHshape-package, 2

find.H, 3
find.hazard, 4
find.shapeCPH, 2–5, 5, 9
find.shapeMLE, 2–5, 7, 9

plot, 9
plot.CPHshape, 9