Package 'validstats'

June 23, 2017

Type Package

Title Various Prediction Model Validation Statistics

Version 1.4

Date 2017-06-22

Author Olli Saarela and Aki S. Havulinna

Maintainer Olli Saarela <olli.saarela@utoronto.ca>

Description

Various prediction model validation statistics (ROC, AUC, c-index, reclassification/NRI, decision curve, calibration), all adapted for censored survival data and allowing for weights.

License GPL (>= 2)

Suggests survival

URL individual.utoronto.ca/osaarela/

R topics documented:

chaz *Cumulative (baseline) hazard*

Description

Nelson-Aalen/Breslow estimate for cumulative (baseline) hazard, allowing for weights.

Usage

```
chaz(timestop, censvar, timestart=NULL, lp=NULL, weight=NULL)
```

Arguments

timestop	A vector of times at the end of the follow-up
censvar	A vector indicating case status (logical or {0,1})
timestart	A vector of times at the start of the follow-up (optional)
lp	A vector of linear predictor values from a Cox regression model (optional)
weight	A vector of weights (optional)

Value

A list with elements

ch	Cumulative (baseline) hazard
times	Increment times

Author(s)

Olli Saarela

cical *Cumulative incidence estimates for model calibration*

Description

Cumulative incidence function estimates to check model calibration, allowing for weights.

Usage

```
cical(p, ftime, censvar, ngroups=10, weight=NULL, casetype=1)
```

Arguments

p	A vector of cumulative incidences
ftime	A vector of follow-up times
censvar	A vector indicating case status (0 for censoring; 1, 2, ... for competing causes)
ngroups	The number of risk categories to be used
weight	A vector of weights (optional)
casetype	Competing cause of interest

Value

A list with elements

totals	Number of observations by risk category
observed	Observed number of events by risk category
expected	Expected number of events by risk category

Author(s)

Olli Saarela

---

cif *Cumulative incidence function*

Description

Non-parametric estimate for cumulative incidence function, allowing for weights.

Usage

```
cif(timestop, censvar, timestart=NULL, weight=NULL, casetype=1)
```

Arguments

timestop	A vector of times at the end of the follow-up
censvar	A vector indicating case status (0 for censoring; 1, 2, ... for competing causes)
timestart	A vector of times at the start of the follow-up (optional)
weight	A vector of weights (optional)
casetype	Competing cause of interest

Value

A list with elements

ci	Cumulative incidence
times	Increment times

Author(s)

Olli Saarela

`ciroc` *Cumulative incidence ROC curve*

Description

Cumulative incidence function ROC curve for competing risks, allowing for weights.

Usage

```
ciroc(p, ftime, censvar, cutpoints=NULL, weight=NULL, casetype=1, cores=1)
```

Arguments

`p`	A vector of cumulative incidences
`ftime`	A vector of follow-up times
`censvar`	A vector indicating case status (0 for censoring; 1, 2, ... for competing causes)
`cutpoints`	A vector of risk cutoffs (optional; if not given, uses all risk levels found in the data)
`weight`	A vector of weights (optional)
`casetype`	Competing cause of interest
`cores`	Number of cores available for parallel computation

Value

A list with elements

`auc`	AUC (by trapedzoid integration)
`truepos`	True positives by cutoff (for plotting)
`falsepos`	False positives by cutoff (for plotting)
`cutoffs`	Risk cutoffs (for plotting)

Author(s)

Olli Saarela

ctest *C-index improvement*

Description

Jackknife estimation for c-index improvement as in Antolini et al. (2004), allowing for weights.

Usage

```
ctest(pnew, pold, ftime, censvar, weight=NULL)
```

Arguments

pnew	A vector of risk estimates (new model)
pold	A vector of risk estimates (old model)
ftime	A vector of follow-up times
censvar	A vector indicating case status (logical or {0,1})
weight	A vector of weights (optional)

Value

A list with elements

cnew	C-index for the new model
cold	C-index for the old model
cdiffse	Jackknife standard error for c-index improvement

Author(s)

Olli Saarela

kmcal *Kaplan-Meier estimates for model calibration*

Description

Kaplan-Meier estimates for model calibration, allowing for weights.

Usage

```
kmcal(p, ftime, censvar, ngroups=10, weight=NULL)
```

Arguments

p	A vector of risk estimates
ftime	A vector of follow-up times
censvar	A vector indicating case status (logical or {0,1})
ngroups	The number of risk categories to be used
weight	A vector of weights (optional)

Value

A list with elements

totals	Number of observations by risk category
observed	Observed number of events by risk category
expected	Expected number of events by risk category

Author(s)

Olli Saarela

Examples

```
library(survival)
library(validstats)

set.seed(1)

a1 <- 6.0
a2 <- 95.0
b <- 0.25
N <- 10000
firstyr <- 1990
lastyr <- 2000
years <- lastyr - firstyr

# Simulate cross-sectional cohort data:

x <- rnorm(N)
byr <- runif(N, firstyr-65, firstyr-25)
ageb <- firstyr - byr
agec <- lastyr - byr
aget <- rweibull(N, a1, a2 * exp(x * b)^(-1/a1))
yrstop <- byr + aget
case <- yrstop < lastyr
aget[!case] <- (lastyr - byr)[!case]
yrstop[!case] <- (byr + aget)[!case]

# Omit prevalent cases:

leftcen <- yrstop < firstyr
N <- sum(!leftcen)
```

```
byr <- byr[!leftcen]
ageb <- ageb[!leftcen]
aget <- aget[!leftcen]
agec <- agec[!leftcen]
yrstop <- yrstop[!leftcen]
case <- case[!leftcen]
x <- x[!leftcen]

# Case-cohort selection:

model <- glm(case ~ ageb, family=binomial(link='logit'))
ps <- 1/(1 + exp(-(coef(model)[1] + coef(model)[2] * ageb)))
ps <- 2.0 * sum(case) * ps/sum(ps)
subcoh <- runif(N) < ps
ccset <- case | subcoh
sum(ccset)
table(ccset, case)

# Kalbfleisch & Lawless weights:

prob <- ps
prob[case] <- 1.0
sum(prob)
weight <- 1/prob[ccset]
sum(weight)

# Full cohort analysis:

model1 <- coxph(Surv(ageb, aget, case) ~ x)
summary(model1)

lp1 <- predict(model1, type='lp')
ch1 <- chaz(aget, case, ageb, lp=lp1)

# Calculate absolute risk estimates:

chstart <- numeric(N)
chstop <- numeric(N)
chstart[ageb >= min(ch1$time)] <-
ch1$ch[findInterval(ageb[ageb >= min(ch1$time)], ch1$time)]
chstop[ageb + years >= min(ch1$time)] <-
ch1$ch[findInterval(ageb[ageb + years >= min(ch1$time)] + years, ch1$time)]
p1 <- 1.0 - exp(-exp(x * coef(model1)) * (chstop-chstart))
sum(p1)

# Observed vs. expected events:

freq1 <- kmcal(p1, aget-ageb, case, ngroups=10)
freq1
barplot(rbind(freq1$observed, freq1$expected), beside=TRUE,
        col=c('red','green'), names.arg=1:10, xlab='Risk category',
        ylab='Count')
legend('topleft', legend=c('Observed events','Expected events'),
```

```
       fill=c('red','green'))

# Case-cohort analysis:

model2 <- coxph(Surv(ageb[ccset], aget[ccset], case[ccset]) ~ x[ccset],
                weight=weight)
summary(model2)

lp2 <- predict(model2, type='lp')
ch2 <- chaz(aget[ccset], case[ccset], ageb[ccset], lp=lp2, weight=weight)

plot(ch1$time, ch1$ch, type='l', lwd=2, xlab='Age', ylab='Cumulative hazard')
lines(ch2$time, ch2$ch, col='red', lwd=2)
legend('topleft', legend=c('Cohort analysis','Case-cohort analysis'),
       lwd=c(2,2), col=c('black','red'))

# Calculate absolute risk estimates:

chstart <- numeric(sum(ccset))
chstop <- numeric(sum(ccset))
chstart[ageb[ccset] >= min(ch2$time)] <-
ch2$ch[findInterval((ageb[ccset])[ageb[ccset] >= min(ch2$time)], ch2$time)]
chstop[ageb[ccset] + years >= min(ch2$time)] <-
ch2$ch[findInterval((ageb[ccset])[ageb[ccset] + years >= min(ch2$time)] + years,
       ch2$time)]
p2 <- 1.0 - exp(-exp(x[ccset] * coef(model2)) * (chstop-chstart))
sum(p2)
sum(p2 * weight)

# Observed vs. expected events:

freq2 <- kmcal(p2, aget[ccset]-ageb[ccset], case[ccset], ngroups=10,
               weight=weight)
barplot(rbind(freq2$observed, freq2$expected), beside=TRUE,
        col=c('red','green'), names.arg=1:10, xlab='Risk category',
        ylab='Count')
legend('topleft', legend=c('Observed events','Expected events'),
       fill=c('red','green'))
```

---

kmdec *Kaplan-Meier decision curve*

Description

Kaplan-Meier decision curve as in Vickers et al. (2008), allowing for weights.

Usage

```
kmdec(p, ftime, censvar, cutpoints=NULL, weight=NULL)
```

Arguments

p	A vector of risk estimates
ftime	A vector of follow-up times
censvar	A vector indicating case status (logical or {0,1})
cutpoints	A vector of risk cutoffs (optional; if not given, uses 1000 equally spaced values from [0,1])
weight	A vector of weights (optional)

Value

A list with elements

netbenefit	Net benefit by cutoff
truepos	True positives by cutoff
falsepos	False positives by cutoff
cutoffs	Risk cutoffs

Author(s)

Olli Saarela

---

kmnri *Kaplan-Meier estimation of NRI*

Description

Kaplan-Meier estimation of NRI, allowing for weights. For bootstrapping, use this instead of kmrc.

Usage

```
kmnri(risklimits, pold, pnew, ftime, censvar, weight=NULL)
```

Arguments

risklimits	A vector of probabilities indicating the risk classification to be used
pnew	A vector of risk estimates (new model)
pold	A vector of risk estimates (old model)
ftime	A vector of follow-up times
censvar	A vector indicating case status (logical or {0,1})
weight	A vector of weights (optional)

Value

A list with elements

casesup	Proportion of cases reclassified up
casesdown	Proportion of cases cases reclassified down
noncasesup	Proportion of non-cases reclassified up
noncasesdown	Proportion of non-cases reclassified down

Author(s)

Olli Saarela

---

kmrc *Kaplan-Meier estimation of reclassification frequencies*

Description

Kaplan-Meier estimation of reclassification frequencies, allowing for weights.

Usage

```
kmrc(risklimits, pold, pnew, ftime, censvar, weight=NULL)
```

Arguments

risklimits	A vector of probabilities indicating the classification to be used
pnew	A vector of risk estimates (new model)
pold	A vector of risk estimates (old model)
ftime	A vector of follow-up times
censvar	A vector indicating case status (logical or {0,1})
weight	A vector of weights (optional)

Value

A list with elements

cases	Reclassification table for cases
noncases	Reclassification table for non-cases
casesup	Proportion of cases reclassified up
casesdown	Proportion of cases cases reclassified down
noncasesup	Proportion of non-cases reclassified up
noncasesdown	Proportion of non-cases reclassified down

Author(s)

Olli Saarela

kmroc *Kaplan-Meier ROC curve*

Description

Kaplan-Meier ROC curve as in Heagerty et al. (2000), allowing for weights.

Usage

```
kmroc(p, ftime, censvar, cutpoints=NULL, weight=NULL, cores=1)
```

Arguments

p	A vector of risk estimates
ftime	A vector of follow-up times
censvar	A vector indicating case status (logical or {0,1})
cutpoints	A vector of risk cutoffs (optional; if not given, uses all risk levels found in the data)
weight	A vector of weights (optional)
cores	Number of cores available for parallel computation

Value

A list with elements

auc	AUC (by trapedzoid integration)
truepos	True positives by cutoff (for plotting)
falsepos	False positives by cutoff (for plotting)
cutoffs	Risk cutoffs (for plotting)

Author(s)

Olli Saarela

`logrank` *Log-rank test statistic*

Description

Log-rank test statistic for comparison between two groups, allowing for weights.

Usage

```
logrank(timestop, censvar, timestart=NULL, group, weight=NULL)
```

Arguments

`timestop`	A vector of times at the end of the follow-up
`censvar`	A vector indicating case status (logical or {0,1})
`timestart`	A vector of times at the start of the follow-up (optional)
`group`	A vector indicating the group of each observation (0 or 1)
`weight`	A vector of weights (optional)

Value

The value of the test statistic

Author(s)

Olli Saarela

Index