

Package ‘JPSurv’

February 19, 2015

Version 1.0.1

Date 2012/03/06

Title Methods for population-based cancer survival analysis

Author Yongwu Shao <ywshao@gmail.com>,

Maintainer Yongwu Shao <ywshao@gmail.com>

Description Functions, methods, and datasets for cancer survival analysis, including the proportional hazard relative survival model, the join point relative survival model.

License GPL (>= 2)

URL <http://www.r-project.org>

Repository CRAN

Date/Publication 2012-03-07 07:11:25

NeedsCompilation no

R topics documented:

aapc	2
CoxFit	2
CoxModel_Year	3
joinpoint	4
predict.joinpoint	5
prostate	6
read.seerstat	6

Index	7
--------------	----------

aapc

Trend summary measures for joint point relative survival model

Description

Get the trend summary measures for joint point relative survival model. Measures include annual percentage changes of hazard, annual percentage changes of cumulative relative survival, annual changes of cumulative relative survival.

Usage

```
aapc(fit, type="HAZ_AC(CS)", interval=5)
```

Arguments

fit	Object of class "joinpoint".
type	Type of trend summary measure. Supported measures are: HAZ_APC(HR) - annual percentage changes of hazard, HAZ_APC(CS) - annual percentage changes of cumulative relative survival, HAZ_AC(CS) - annual changes of cumulative relative survival. The default is HAZ_AC(CS).
interval	Years after diagnosis. Only needed for HAZ_APC(CS) and HAZ_AC(CS).

Value

The estimates and standard errors of the trend summary measure.

Examples

```
data(prostate);
# Fit the survival join point model with zero join points, i.e., fit the proportional hazard relative survival model
fit1 = joinpoint(~Year, data=prostate, numJPoints = 0);
# Get the estimates and standard errors of the annual changes of cumulative relative survival.
haz_ac = aapc(fit1, type="HAZ_AC(CS)", interval=5);
```

CoxFit

Fitting a proportional hazard relative survival model

Description

Fitting a proportional hazard relative survival model

Usage

```
CoxFit(X, nAlive, nDied, nLost, expSurv)
```

Arguments

X	The input design matrix.
nAlive	Number of people at risk.
nDied	Number of people who died.
nLost	Number of people who are censored.
expSurv	The expected survival rate.

Value

A CoxFit class object.

CoxModel_Year	<i>Fitting a proportional hazard relative survival model with year as a covariate</i>
---------------	---

Description

Fitting a proportional hazard relative survival model with year as a covariate

Usage

```
CoxModel_Year(formula, data, subset, ...)
```

Arguments

formula	an object of class "formula": a symbolic description of the model to be fitted. If the input data is from SEER*Stat, the formula can be ~Year. Otherwise, the full formula should be specified as: ~Alive_at_Start + Died + Lost_to_Followup + Expected_Survival_Interval + Interval + Year
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used in the fitting process.
...	additional arguments to be passed to the low level regression fitting functions.

Value

A list with attributes:

coefficients	a named vector of coefficients and standard errors
converged	convergence status
predicted	the fitted relative survival rates
xbeta	the linear predictor
ll	log likelihood
aic	AIC
bic	BIC

 joinpoint

Fitting a join point relative survival model

Description

Fitting a joinpoint relative survival model

Usage

```
joinpoint(formula, data, subset, numJPoints = 0, ...)
```

Arguments

formula	an object of class "formula": a symbolic description of the model to be fitted. If the input data is from SEER*Stat, the formula can be ~Year. Otherwise, the full formula should be specified as: ~Alive_at_Start + Died + Lost_to_Followup + Expected_Survival_Interval + Interval + Year
data	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in data, the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used in the fitting process.
numJPoints	The maximum number of join points allowed. The default is zero, which is equivalent to a proportional hazard relative survival model.
...	additional arguments to be passed to the low level regression fitting functions.

Value

An object of class "joinpoint" will be returned with attributes:

coefficients	a named vector of coefficients and standard errors
jp	the estimates of the join points
converged	convergence status
predicted	the fitted relative survival rates
xbeta	the linear predictor
ll	log likelihood
aic	AIC
bic	BIC

References

Yu, B., Huang, L., Tiwari, R. C., Feuer, E. J. and Johnson, K. A. (2009), Modeling population-based cancer survival trends by using join point models for grouped survival data. *Journal of the Royal Statistical Society: Series A*, 172, 405-425.

Examples

```
#load the SEER-9 prostate cancer dataset.
data(prostate);
# Fit the survival join point model with zero join points, i.e., fit the proportional hazard relative survival model.
fit1 = joinpoint(~Year, data=prostate, numJPoints = 0);
# Fit the survival join point model on a subset of the dataset.
fit2 = joinpoint(~Year, data=prostate, Site_and_age_for_total_stage == 18 & Sex == 1, numJPoints = 0);
# Fit the survival join point model for non-SEER*Stat datasets.
fit3 = joinpoint(~Alive_at_Start + Died + Lost_to_Followup + Expected_Survival_Interval
+ Interval + Year, prostate, numJPoints = 0);
plot(fit1);
```

predict.joinpoint	<i>Predict method for join point models</i>
-------------------	---

Description

Predicted values based on join point object

Usage

```
## S3 method for class 'joinpoint'
predict(object, ...)
```

Arguments

object	Object of class "joinpoint"
...	Two more arguments: years, intervals may be used to specify which years and intervals to predict.

Value

A data frame which contains the predicted interval survivals and cumulative survivals

Examples

```
data(prostate);
fit = joinpoint(~Year, data=prostate, numJPoints = 0);
pred1 = predict(fit);
pred2 = predict(fit, years = 1978, intervals = 5);
pred3 = predict(fit, years = 1976:1980, intervals = c(2, 3, 5));
```

prostate *Life Table for the Prostate Cancer*

Description

This data set is extracted from the SEER 9 Cancer Registries and it contains the life table of prostate cancer patients.

Usage

```
prostate
```

Format

An R dataframe of the size 433x15

read.seerstat *Reading a SEER*Stat dataset*

Description

Read the SEER*Stat source files

Usage

```
read.seerstat(fileName, keep.missing=TRUE)
```

Arguments

fileName The input SEER*Stat file name, no missing values allowed.
keep.missing Whether to delete the missing values.

Value

The SEER*Stat matrix.

Examples

```
#X = read.seerstat("allstages.bysiteage.sex.year.yearly");
```

Index

*Topic **datasets**

prostate, [6](#)

aapc, [2](#)

CoxFit, [2](#)

CoxModel_Year, [3](#)

joinpoint, [4](#)

plot.joinpoint (joinpoint), [4](#)

predict.joinpoint, [5](#)

print.joinpoint (joinpoint), [4](#)

prostate, [6](#)

read.seerstat, [6](#)

summary.joinpoint (joinpoint), [4](#)