Package ‘bootStepAIC’
October 12, 2022

Title  Bootstrap stepAIC
Version  1.3-0
Date  2022-03-07
Author  Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>
Maintainer  Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>
Description  Model selection by bootstrapping the stepAIC() procedure.
Depends  R(>= 4.0.0), MASS
LazyLoad  yes
License  GPL (>= 2)
NeedsCompilation  no
Repository  CRAN
Date/Publication  2022-03-08 13:40:05 UTC

R topics documented:

  boot.stepAIC  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1

Index

  boot.stepAIC  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

Description

  Implements a Bootstrap procedure to investigate the variability of model selection under the stepAIC()
  stepwise algorithm of package MASS.

Usage

  boot.stepAIC(object, data, B = 100, alpha = 0.05, direction = "backward",
               k = 2, verbose = FALSE, seed = 1L, ...)

1
**Arguments**

*object* 
an object representing a model of an appropriate class; currently, "lm", "aov", "glm", "negbin", "polr", "survreg", and "coxph" objects are supported.

*data* 
a `data.frame` or a `matrix` that contains the response variable and covariates.

*B* 
the number of Bootstrap samples.

*alpha* 
the significance level.

*direction* 
the direction argument of `stepAIC()`.

*k* 
the k argument of `stepAIC()`.

*verbose* 
logical; if TRUE information about the evolution of the procedure is printed in the screen.

*seed* 
numeric scalar denoting the seed used to create the Bootstrap samples.

*...* 
extra arguments to `stepAIC()`, e.g., scope.

**Details**

The following procedure is replicated B times:

**Step 1:** Simulate a new data-set taking a sample with replacement from the rows of data.

**Step 2:** Refit the model using the data-set from Step 1.

**Step 3:** For the refitted model of Step 2 run the `stepAIC()` algorithm.

Summarize the results by counting how many times (out of the B data-sets) each variable was selected, how many times the estimate of the regression coefficient of each variable (out of the times it was selected) it was statistically significant in significance level `alpha`, and how many times the estimate of the regression coefficient of each variable (out of the times it was selected) changed signs (see also Austin and Tu, 2004).

**Value**

An object of class `BootStep` with components

*Covariates* 
a numeric matrix containing the percentage of times each variable was selected.

*Sign* 
a numeric matrix containing the percentage of times the regression coefficient of each variable had sign + and −.

*Significance* 
a numeric matrix containing the percentage of times the regression coefficient of each variable was significant under the `alpha` significance level.

*OrigModel* 
a copy of `object`.

*OrigStepAIC* 
the result of applying `stepAIC()` in `object`.

*direction* 
a copy of the `direction` argument.

*k* 
a copy of the `k` argument.

*BootStepAIC* 
a list of length B containing the results of `stepAIC()` for each Bootstrap data-set.

**Author(s)**

Dimitris Rizopoulos <d.rizopoulos@erasmusmc.nl>
References


See Also

`stepAIC` in package MASS

Examples

```r
## lm() Example ##
n <- 350
x1 <- runif(n, -4, 4)
x2 <- runif(n, -4, 4)
x3 <- runif(n, -4, 4)
x4 <- runif(n, -4, 4)
x5 <- runif(n, -4, 4)
x6 <- runif(n, -4, 4)
x7 <- factor(sample(letters[1:3], n, rep = TRUE))
y <- 5 + 3 * x1 + 2 * x2 - 1.5 * x3 - 0.8 * x4 + rnorm(n, sd = 2.5)
data <- data.frame(y, x1, x2, x3, x4, x5, x6, x7)
rm(n, x1, x2, x3, x4, x5, x6, x7)

lmFit <- lm(y ~ (.-x7) * x7, data = data)
boot.stepAIC(lmFit, data)
```

```r
## glm() Example ##
n <- 200
x1 <- runif(n, -3, 3)
x2 <- runif(n, -3, 3)
x3 <- runif(n, -3, 3)
x4 <- runif(n, -3, 3)
x5 <- factor(sample(letters[1:2], n, rep = TRUE))
eta <- 0.1 + 1.6 * x1 - 2.5 * as.numeric(as.character(x5) == levels(x5)[1])
y1 <- rbinom(n, 1, plogis(eta))
y2 <- rbinom(n, 1, 0.6)
data <- data.frame(y1, y2, x1, x2, x3, x4, x5)
rm(n, x1, x2, x3, x4, x5, eta, y1, y2)

glmFit1 <- glm(y1 ~ x1 + x2 + x3 + x4 + x5, family = binomial(), data = data)
glmFit2 <- glm(y2 ~ x1 + x2 + x3 + x4 + x5, family = binomial(), data = data)

boot.stepAIC(glmFit1, data, B = 50)
boot.stepAIC(glmFit2, data, B = 50)
```
Index

* regression
  boot.stepAIC, 1

boot.stepAIC, 1