Package ‘lm.br’

September 13, 2013

Type Package

Title Linear Model with Breakpoint

Version 1.7.1

Date 2013-09-18

Author Marc Adams. Source code includes a copy of the JAMA package from the U.S. National Institute of Standards and Technology (NIST), and borows from 'lm' and 'MASS::lm.gls'.

Maintainer Marc Adams <lm.br.pkg@gmail.com>

Description Exact significance tests for a changepoint in linear or multivariate linear regression. Confidence regions with exact coverage probabilities for the changepoint.

License GPL (>= 2)

Depends R(>= 2.15.1), methods, Rcpp (>= 0.10.2)

LinkingTo Rcpp

ByteCompile yes

BuildVignettes no

R topics documented:

'lm.br' *Fit a Linear Model with a Breakpoint*

Description

Exact significance tests for a coefficient changepoint in linear or multivariate linear regression, assuming continuity. Confidence intervals and confidence regions with exact coverage probabilities for the changepoint.

Usage

```
lm.br( formula, type = "LL", data, subset, weights, inverse =FALSE,
var.known = FALSE, offset, contrasts, na.action, ... )
```

Arguments

`formula`	a formula expression as for regression models, of the form `response ~ predictors`, see `formula` for more
`type`	'LL', 'LT' or 'TL' which stand for line-line, line-threshold or threshold-line, defined below
`data`	an optional data frame that defines variables in `formula`
`subset`	expression saying which subset of the data to use
`weights`	vector or positive-definite matrix
`inverse`	if TRUE, 'weights' specifies the inverse of the weights vector or matrix, as for a covariance matrix
`var.known`	TRUE or FALSE
`offset`	a constant vector to be subtracted from the responses vector
`contrasts`	an optional list, see 'contrasts.arg' in `model.matrix.default`
`na.action`	a function to filter missing data
`...`	other arguments to `lm.fit` or `lm.wfit`

Details

A broken-line model consists of two straight lines joined at a changepoint. Three versions are

LL: y = alpha + B * min(x-theta , 0) + B'* max(x-theta, 0) + e

LT: y = alpha + B * min(x-theta , 0) + e

TL: y = alpha + B'* max(x-theta , 0) + e

where e ~ N(0 , var * inv(weights)). All parameters are unknown except for 'weights'. The LT and TL models omit 'alpha' if the formula is without intercept, such as 'y~x+0'.

The same models apply for a multivariate formula such as 'y ~ x1 + x2 + ... + xn' where 'alpha' becomes the coefficient of the "1"-vector and 'theta' becomes the changepoint for the cofficient of the first term, 'x1'.

Exact inferences about the changepoint 'theta' or '(theta,alpha)' are based on the distribution of its likelihood-ratio statistic, conditional on sufficient statistics for the other parameters.

Test for the presence of a changepoint by the significance level of a postulate value outside of the x-values. Thus, 'sl(min(x1)-1)' would give the significance level of the null hypothesis "single line" versus the alternate hypothesis "broken line," in the 'LL' model.

Value

'lm.br' returns a list that includes a C++ object with accessor functions.

Function 'ci' gets confidence intervals, 'cr' confidence regions, and 'sl' significance levels for the changepoint's x-coordinate or (x,y)-coordinates.

Other functions 'mle' gets maximum likelihood estimates and 'sety' sets new y-values.

The returned list also contains the components of an 'lm' list including 'coefficients', 'fitted.values' and 'residuals'.

Note

If variance is known, 'weights' is the inverse of the variances vector or variance-covariance matrix, and 'var'=1 in the algebraic expressions of the model above.

References

Siegmund, D. and Zhang, H.P. (1994), "Confidence regions in broken line regression," in _Changepoint Problems_, IMS Lecture Notes – Monograph Series, vol. 23, eds. E. Carlstein, H. Muller and D. Siegmund, Hayward, CA: Institute of Mathematical Statistics, pp. 292-316.

See Also

'ci', 'cr', 'sl', 'mle', 'sety'

The 'sl' help page explains the evaluation methods.

The 'sety' page gives an example simulations test.

'lm' describes the arguments and output values.

Examples

```
library(lm.br)

#  Data for Patient B from Smith and Cook (1980),
#  reciprocal of blood creatinine L/micromol  versus
#  day after kidney transplant:
crea = c( 37.3, 47.1, 51.5, 67.6, 75.9, 73.3, 69.4, 61.5, 31.8, 19.4 )
day = c( 1., 2., 3., 4., 5., 6., 7., 8., 9., 10. )
sc <- lm.br( crea ~ day )
sc$ci()
sc$cr(.90,'af')
sc$sl( crea[1] - 1.5 )      # test for the presence of a changepoint
sc$mle()



#  A 'TL' example, from figure 1 of  Chiu et al. (2006),
#  log(salmon abundance) vs year:
salmon = c( 2.50, 2.93, 2.94, 2.83, 2.43, 2.84, 3.06, 2.97, 2.94, 2.65, 2.92,
 2.71, 2.93, 2.60, 2.12, 2.08, 1.81, 2.45, 1.71, 0.55, 1.30 )
year = 1980:2000
chiu <- lm.br( salmon ~ year, 'TL' )
chiu$ci(0.90)
```

```
#  An ’LT’ example with repeat observations, from Robbins, Saxton and Southern (2006).
#  pig weight-gain in grams-per-day  versus   % dietary isoleucine:

wgt_gain = c( 533, 452, 433, 418,   686, 543, 648, 530,
       819, 657, 561, 712,   757, 743, 560, 721,
       691, 757, 657, 809,   667, 721, 651, 862  )
isoleu = c( 0.38, 0.38, 0.38, 0.38,  0.42, 0.42, 0.42, 0.42,
       0.46, 0.46, 0.46, 0.46,  0.50, 0.50, 0.50, 0.50,
       0.54, 0.54, 0.54, 0.54,  0.58, 0.58, 0.58, 0.58 )

lm.br( wgt_gain ~ isoleu, ’LT’ )



#  A multivariate example, using one of R’s included datasets,
#  automobile miles-per-gallon versus weight and horsepower:

lm.br( mpg ~ wt + hp, data = mtcars )



#  An example with variance known, for the Normal approximations
#  of binomial random variables.
#      Drinking-and-driving yearly surveys by TIRF, Ottawa,
#  with seasonal adjustment based on monthly surveys by CAMH, Toronto,
#  log-odds vs year:

log_odds = c( -1.194, -2.023, -2.285, -1.815, -1.673, -1.444, -1.237, -1.228 )
year = c( 1998.92, 2001.25, 2002.29, 2003.37, 2004.37, 2005.71, 2006.71, 2007.71 )

VarCov = matrix(  c(   0.0361, 0, 0, 0, 0, 0, 0, 0,
          0, 0.0218, 0.0129, 0, 0, 0, 0, 0,
          0, 0.0129, 0.0319, 0, 0, 0, 0, 0,
          0, 0, 0, 0.0451, 0.0389, 0, 0, 0,
          0, 0, 0, 0.0389, 0.0445, 0, 0, 0,
          0, 0, 0, 0, 0, 0.0672, 0.0607, 0.0607,
          0, 0, 0, 0, 0, 0.0607, 0.0664, 0.0607,
          0, 0, 0, 0, 0, 0.0607, 0.0607, 0.0662 ) ,  nrow = 8, ncol = 8 )

dd <- lm.br( log_odds ~ year, w = VarCov, inv = TRUE, var.known = TRUE )

dd$ci()
dd$ci(.95,’af’)
dd$cr()



#  An example that shows different confidence regions
#  by conditional likelihood-ratio inference and by approximate-F inference:
y = c( 1.55, 3.2, 6.3, 4.8, 4.3, 4.0, 3.5, 1.8 )
x = 1:8
eg <- lm.br(y~x)
eg$cr()
eg$cr(.95,”af”)
```

ci — *Confidence Interval for the Changepoint*

Description

Confidence interval for ’theta’, the changepoint’s x-coordinate.

Usage

```
## S4 method for signature ’Cpp_Clmbr’
ci(  CL =0.95, method ="clr" )
```

Arguments

CL	confidence level, between 0 and 1
method	"clr" or "af" which stand for conditional likelihood-ratio or approximate-F

Details

This subroutine scans to determine the postulate values of ’theta’ that have significance levels greater than 1-CL.

Examples

```
##  Data for Patient B from Smith and Cook (1980)
require(lm.br)
y = c(37.3, 47.1, 51.5, 67.6, 75.9, 73.3, 69.4, 61.5, 31.8, 19.4)
x = 1:10
sc <- lm.br(y~x)
sc$ci()
sc$ci(0.90)
sc$ci(.99,’af’)
```

cr — *Confidence Region for the Changepoint*

Description

Joint confidence region for (theta, alpha), the changepoint’s (x,y)-coordinates.

Usage

```
## S4 method for signature ’Cpp_Clmbr’
cr(  CL =0.95 ,  method ="clr",  incr  )
```

Arguments

CL	confidence level, between 0 and 1
method	"clr" or "af" which stand for conditional likelihood-ratio or approximate-F
incr	increment of theta values in the confidence region printout

Details

This subroutine scans to determine the postulate values of (theta,alpha) that have significance levels greater than 1-CL. It scans first along the (theta, alpha-MLE) ridge to determine 'theta' boundaries.

Examples

```
##  Data for Patient B from Smith and Cook (1980)
library(lm.br)
y = c(37.3, 47.1, 51.5, 67.6, 75.9, 73.3, 69.4, 61.5, 31.8, 19.4)
x = 1:10
sc <- lm.br(y~x)
sc$cr()
sc$cr(.9,'clr',0.1)
sc$cr(.99,"af")
```

mle *Maximum Likelihood Estimates*

Description

Maximum-likelihood estimates of parameters. Estimates are without bias correction except for the variance.

Usage

```
## S4 method for signature 'Cpp_Clmbr'
mle()
```

Examples

```
##  Data for Patient B from Smith and Cook (1980):
library(lm.br)
y = c(37.3, 47.1, 51.5, 67.6, 75.9, 73.3, 69.4, 61.5, 31.8, 19.4)
x = 1:10
sc <- lm.br(y~x)
sc$mle()
```

sety *Set y-Values*

Description

Reset the y-values in the C++ object.

Usage

```
## S4 method for signature 'Cpp_Clmbr'
sety( rWy )
```

Arguments

rWy	vector of y-values, pre-multiplied by the square-root of 'weights'

Details

This method changes the y-values for the accessor functions 'sl', 'ci', 'cr' and 'mle'. But the output list of 'coefficients' and 'fitted.values' and so on retains its values from the original call.

Note

The pre-multiplied vector is more convenient as input during simulation tests. This vector is simply the y-vector if 'weights' is omitted.

The square-root of a vector 'w' is the vector of the square roots of elements. The square-root of a matrix 'W' here is the matrix 'rW' such that rW*rW = W (stricter than rW*transpose(rW) = W).

Examples

```
##  Data for Patient B from Smith and Cook (1980)
y <- c(36.0, 45.5, 50.0, 60.0, 73.3, 71.0, 66.7, 60.0, 30.5, 18.3)
x <- 1:10
sc <- lm.br(y~x)
sc$ci()
newy <- c(37.3, 47.1, 51.5, 67.6, 75.9, 73.3, 69.4, 61.5, 31.8, 19.4)
sc$sety(newy)
sc$ci()


## An example simulation test:

simtest <- function( x, theta, alpha, B, Bp, var, N =10000) {
   y <- x
   mod <- lm.br(y~x)     ## construct an 'lm.br' object with arbitrary y values.

   cat("\n   Monte Carlo simulation test:\n")
   cat("for model   theta=",theta," alpha=",alpha," B=",B," B'=",Bp," var=",var,"\n")
   cat("with observations at  x=",x,"\n\n")
   cat("     no. of     coverage frequency of the 0.95-confidence interval by\n")
   cat("   iterations               CLR                    AF\n")
   flush.console()
   sigma = sqrt(var)
   n = length(x)
   countCLR = countAF = 0

   for( i in seq(1,N) ) {
      err <- rnorm(n,0,sigma)
      for( j in seq(1,n) )
         y[j] = alpha + B*min(x[j]-theta,0) + Bp*max(x[j]-theta,0) + err[j]
      mod$sety(y)
      stest = mod$sl(theta,"clr",.0001,FALSE)
      if(stest>0.05) countCLR = countCLR + 1
      stest = mod$sl(theta,"af",.0001,FALSE)
      if(stest>0.05) countAF = countAF + 1

      if(i/1000 - floor(i/1000) == 0) {
         cat( format(i,width=10),
              format(countCLR/i,digits=4,nsmall=4,width=22),
              format(countAF/i,digits=4,nsmall=4,width=22), "\n" )
         flush.console()
      }
   }
   cat("\n")
}

x = c( 1.0, 1.1, 1.3, 1.7, 2.4, 3.9, 5.7, 7.6, 8.4, 8.6 )
simtest(x,3,0,-1,0.5,1)
```

sl *Significance Level for Changepoint*

Description

Significance level of a postulate value for the changepoint's x-coordinate or (x,y)-coordinates.

Usage

```
## S4 method for signature 'Cpp_Clmbr'
sl( theta0,  method ="clr", accuracy =0.001, verbose =TRUE )
sl( theta0, alpha0,  method ="clr", accuracy =0.001, verbose =TRUE )
```

Arguments

`theta0` postulate value for 'theta', the changepoint's x-coordinate

`alpha0` postulate value for 'alpha', the changepoint's y-coordinate

`method` 'clr', 'mc' or 'af' which stand for conditional likelihood-ratio, conditional likelihood-ratio by Monte Carlo, or approximate-F, details below

`accuracy` maximum absolute error in numerical integration for the 'clr' method, or in Monte Carlo evaluation for the 'mc' method, not referenced for the 'af' method

`verbose` if TRUE prints the result with gloss

Details

Siegmund and Zhang derived the conditional likelihood-ratio significance level of a postulate change-point value as a probability of a generic random variable.

The default method 'clr' evaluates this probability using a geometric-expectation formula derived by Siegmund-Zhang, which slightly over-estimates.

Method 'mc' evaluates the probability by Monte Carlo simulation, which avoids the over-estimate of the 'clr' method.

Method 'af' evaluates significance level by estimating the distribution of the likelihood-ratio statistic by the related F-distribution (or chi-squared distribution if variance is known) that would be exact for a linear model. This method is not exact. It is common for non-linear regression.

Value

Returns the significance level if parameter 'verbose' is input explicitly as TRUE or FALSE, otherwise only prints-out the result.

Note

The 'accuracy' error limit does not include the slight over-estimate that is inherent in the 'clr' method, nor the approximation inherent in the 'af' method.

Examples

```
require(lm.br)
##  Data for Patient B from Smith and Cook (1980):
y = c(37.3, 47.1, 51.5, 67.6, 75.9, 73.3, 69.4, 61.5, 31.8, 19.4)
x = 1:10
sc <- lm.br(y~x)

sc$sl(6.1)
sc$sl(6.1,"af")
sc$sl(6.1,"mc",0.00001)
tmp <- sc$sl(6.1,"clr",0.001,FALSE)
tmp

sc$sl(6.1,88.2)
sc$sl(6.1,88.2,"clr",0.00001)
sc$sl(6.1,88.2,"af")
sc$sl(6.1,88.2,"mc",0.00001)
tmp2 <- sc$sl(6.1,88.2,"af",.001,TRUE)
tmp2
```

Index