Package ‘ConSpline’

July 21, 2025

Type Package

Title Partial Linear Least-Squares Regression using Constrained Splines

Version 1.2

Date 2017-10-02

Author Mary C Meyer

Maintainer Mary C Meyer <meyer@stat.colostate.edu>

Description Given response y, continuous predictor x, and covariate matrix, the relationship between E(y) and x is estimated with a shape constrained regression spline. Function outputs fits and various types of inference.

License GPL-2 | GPL-3

Depends graphics, grDevices, stats, utils, coneproj (>= 1.12)

NeedsCompilation no

Repository CRAN

Date/Publication 2017-10-02 18:49:42 UTC

Contents

ConSpline-package *Partial Linear Least-squares Regression with Constrained Splines*

Description

Given a continuous response y and a continuous predictor x, and a design matrix Z of parametrically-modeled covariates, the model y=f(x)+Zb+e is fit using least-squares cone projection. The function f is smooth and has one of eight user-defined shapes: increasing, decreasing, convex, concave, or combinations of monotonicity and convexity. Quadratic splines are used for increasing and decreasing, while cubic splines are used for the other six shapes.

Details

Package:	ConSpline
Type:	Package
Version:	1.1
Date:	2015-08-27
License:	GPL-2 | GPL-3

The function `conspline` fits the partial linear model. Given a response variable y, a continuous predictor x, and a design matrix Z of parametrically modeled covariates, this function solves a least-squares regression assuming that y=f(x)+Zb+e, where f is a smooth function with a user-defined shape. The shape is assigned with the argument type, where 1=increasing, 2=decreasing, 3=convex, 4=concave, 5=increasing and convex, 6=decreasing and convex, 7=increasing and concave, 8= decreasing and concave.

Author(s)

Mary C. Meyer

Maintainer: Mary C. Meyer <meyer@stat.colostate.edu>

References

Meyer, M.C. (2008) Shape-Restricted Regression Splines, *Annals of Applied Statistics*, **2(3)**,1013-1033.

Examples

```
data(WhiteSpruce)
plot(WhiteSpruce$Diameter,WhiteSpruce$Height)
ans=conspline(WhiteSpruce$Height,WhiteSpruce$Diameter,7)
lines(sort(WhiteSpruce$Diameter),ans$muhat[order(WhiteSpruce$Diameter)])
```

conspline — *Partial Linear Least-Squares with Constrained Regression Splines*

Description

Given a response variable y, a continuous predictor x, and a design matrix Z of parametrically modeled covariates, this function solves a least-squares regression assuming that y=f(x)+Zb+e, where f is a smooth function with a user-defined shape. The shape is assigned with the argument type, where 1=increasing, 2=decreasing, 3=convex, 4=concave, 5=increasing and convex, 6=decreasing and convex, 7=increasing and concave, 8=decreasing and concave.

Usage

```
conspline(y,x,type,zmat=0,wt=0,knots=0,
   test=FALSE,c=1.2,nsim=10000)
```

Arguments

`y`	A continuous response variable
`x`	A continuous predictor variable. The length of x must equal the length of y.
`type`	An integer 1-8 describing the shape of the regression function in x. 1=increasing, 2=decreasing, 3=convex, 4=concave, 5=increasing and convex, 6=decreasing and convex, 7=increasing and concave, 8= decreasing and concave.
`zmat`	An optional design matrix of covariates to be modeled parametrically. The number of rows of zmat must be the length of y.
`wt`	Optional weight vector, must be positive and of the same length as y.
`knots`	Optional user-defined knots for the spline function. The range of the knots must contain the range of x.
`test`	If test=TRUE, a test for the "significance" of x is performed. For convex and concave shapes, the null hypothesis is that the relationship between y and x is linear, for any of the other shapes, the null hypothesis is that the expected value of y is constant in x.
`c`	An optional parameter for the variance estimation. Must be between 1 and 2 inclusive.
`nsim`	An optional specification of the number of simulated data sets to make the mixing distribution for the test statistic if test=TRUE.

Details

A cone projection is used to fit the least-squares regression model. The test for the significance of x is exact, while the inference for the covariates represented by the Z columns uses statistics that have approximate t-distributions.

Value

muhat	The fitted values at the design points, i.e. an estimate of E(y).
fhat	The estimated regression function, evaluated at the x-values, describing the relationship between E(y) and x, see above description of the model.
fslope	The slope of fhat, evaluated at the x-values.
knots	The knots used in the spline function estimation.
pvalx	If test=TRUE, this is the p-value for the test involving the predictor x. For convex and concave shapes, the null hypothesis is that the relationship between y and x is linear, versus the alternative that it has the assigned shape. For any of the other shapes, the null hypothesis is that the expected value of y is constant in x, versus the assigned shape.
zcoef	The estimated coefficients for the components of the regression function given by the columns of Z. An "intercept" is given if the column space of Z did not contain the constant vectors.
sighat	The estimate of the model variance. Calculated as SSR/(n-cD), where SSR is the sum of squared residuals of the fit, n is the length of y, D is the observed degrees of freedom of the fit, and c is a parameter between 1 and 2.
zhmat	The hat matrix corresponding the columns of Z, to compute p-values for contrasts, for example.
sez	The standard errors for the Z coefficient estimates. These are square roots of the diagonal values of zhmat, times the square root of sighat.
pvalz	Approximate p-values for the null hypotheses that the coefficients for the covariates represented by the Z columns are zero.

Author(s)

Mary C Meyer, Professor, Statistics Department, Colorado State University

References

Meyer, M.C. (2008) Shape-Restricted Regression Splines, *Annals of Applied Statistics*, **2(3)**,1013-1033.

Examples

```
n=60
x=1:n/n
z=sample(0:1,n,replace=TRUE)
mu=1:n*0+4
mu[x>1/2]=4+5*(x[x>1/2]-1/2)^2
mu=mu+z/4
y=mu+rnorm(n)/4
plot(x,y,col=z+1)
ans=conspline(y,x,5,z,test=TRUE)
points(x,ans$muhat,pch=20,col=z+1)
lines(x,ans$fhat)
lines(x,ans$fhat+ans$zcoef, col=2)
```

```
ans$pvalz  ## p-val for test of significance of z parameter
ans$pvalx  ## p-val for test for linear vs convex regression function
```

GAVoting — *Voting Data for Counties in Georgia, for the 2000 U.S. Presidential Election*

Description

Voting data by county, for the 150 counties in the state of Georgia, in the Bush vs Gore 2000 presidential election.

Usage

```
data("GAVoting")
```

Format

A data frame with 159 observations on the following 9 variables.

`county` the county name

`method` the voting method: `OS-CC` (optical scan, central count); `OS-PC` (optical scan, precinct count); `LEVER` (lever); `PUNCH` (punch card); `PAPER` (paper ballot)

`econ` the economic level of the county according to OneGeorgia: `poor`; `middle`; `rich`

`percent.black` proportion of registered voters who are black

`gore` number of votes recorded for Mr Gore

`bush` number of votes recorded for Mr Bush

`other` number of votes recorded for a third candidate

`votes` number of votes recorded

`ballots` number of ballots received

Details

The uncounted votes in the 2000 presidential election were a concern in the state of Florida, where 2.9 percent of the ballots did not have vote for president recorded. Because the election was close in that state, the voting methods and other issues were scrutinized. In the state of Georgia, 3.5 percent of the votes were uncounted. This data set gives votes by county, along with other data including voting method. A properly weighted ANOVA will show that proportions of uncounted votes are significantly higher with counties using the punch card method.

References

Meyer, M.C. (2002). Uncounted Votes: Does Voting Equipment Matter? Chance Magazine, 15(4), pp33-38.

Examples

```
data(GAVoting)
obs1=1:5
obs2=1:3
meth=1:159
econ=1:159
types=unique(GAVoting$method)
econs=unique(GAVoting$econ)
for(i in 1:159){
meth[i]=obs1[GAVoting$method[i]==types]
econ[i]=obs2[GAVoting$econ[i]==econs]
}
punc=100*(1-GAVoting$votes/GAVoting$ballots)
par(mar=c(4,4,1,1))
plot(GAVoting$percent.black,punc,xlab="Proportion of black voters",
  ylab="percent uncounted votes",col=meth,pch=econ)
legend(0,18.5,pch=1:3,legend=c("poor","middle","rich"))
legend(.63,18.5,pch=c(1,1,1,1,1),col=1:5,
  legend=c("lever","OS-CC","OS-PC","punch","paper"))

zmat=matrix(0,ncol=4,nrow=159)
for(i in 1:4){zmat[meth==i+1,i]=1}

ans1=conspline(punc,GAVoting$percent.black,1,zmat,wt=GAVoting$ballots)
lines(sort(GAVoting$percent.black),
   ans1$fhat[order(GAVoting$percent.black)],col=1)
for(i in 1:4){
lines(sort(GAVoting$percent.black),
ans1$fhat[order(GAVoting$percent.black)]+ans1$zcoef[i],col=i+1)
}
```

`WhiteSpruce` *Height and Diameter of 36 White Spruce trees.*

Description

A standard scatterplot example from various statistics text books, representing height versus diameter of White Spruce trees.

Usage

```
data("WhiteSpruce")
```

Format

A data frame with 36 observations on the following 2 variables.

`Diameter` Diameter at "breast height" of tree

`Height` Height of tree

Examples

```
data(WhiteSpruce)
plot(WhiteSpruce$Diameter,WhiteSpruce$Height)
ans=conspline(WhiteSpruce$Height,WhiteSpruce$Diameter,7)
lines(sort(WhiteSpruce$Diameter),ans$muhat[order(WhiteSpruce$Diameter)])
```

Index