Package ‘uniReg’

February 20, 2015

Type Package
Title Unimodal penalized spline regression using B-splines
Version 1.0
Date 2014-04-28
Author Claudia Koellmann
Maintainer Claudia Koellmann <koellmann@statistik.tu-dortmund.de>
Imports DoseFinding, MASS, mvtnorm, quadprog, SEL, splines
Description Univariate spline regression. It is possible to add the shape constraint of unimodality and predefined or self-defined penalties on the B-spline coefficients.
License GPL (>= 2)
NeedsCompilation no
Repository CRAN
Date/Publication 2014-04-29 07:59:12

R topics documented:

  equiknots ................................................................. 1
  unimat ................................................................. 3
  unireg ................................................................. 4

Index

  equiknots  Determine the knot sequence.

Description

Determines $g+2k+2$ knots for the spline basis. The inner knots lie equidistant in $[a, b]$. If coinc=T, $k$ knots are equal to each $a$ and $b$, otherwise the outer knots are also equidistant beyond $[a, b]$.  


Usage

equiknots(a, b, g, k, coinc)

Arguments

a  
  Left boundary of the spline interval.

b  
  Right boundary of the spline interval.

g  
  Number of inner knots.

k  
  Spline degree.

coinc  
  Should the outer knots be coincident or not? If coinc=TRUE, there are k coincident outer knots at each a and b.

Value

A numeric vector of length g+2k+2 with knot locations.

Author(s)

Claudia Koellmann

See Also

unimat, unireg

Examples

equiknots(0,5,3,3,TRUE)
equiknots(0,5,3,3,FALSE)
Create the matrix of unimodality constraints.

Description
Returns a matrix \( A \) that can be used to specify linear constraints \( Ab \geq 0 \) to impose unimodality with mode at the \( m \)-th element on a numeric vector \( b \) of length \( p \).

Usage
unimat(p, m)

Arguments
- \( p \) Length (\( \geq 2 \)) of the vector, for which the constraints will be used.
- \( m \) Location of the mode within the vector. Should be \( 1 \leq m \leq p \).

Value
Matrix with coefficients for the linear constraints.

Author(s)
Claudia Koellmann

See Also
equiknots, unireg

Examples
unimat(4,2)
unimat(5,3)
**unireg**

Fitting a unimodal penalized spline regression.

Description

Function for fitting spline regressions to data. The fit can be constrained to be unimodal, inverse-unimodal, isotonic or antitonic and an arbitrary penalty on the B-spline coefficients can be used.

Usage

```r
unireg(x, y, w=NULL, sigma=NULL, a=min(x), b=max(x), g=10, k=3,
       constr=c("unimodal","none","invuni","isotonic","antitonic"),
       penalty=c("diff", "none", "sigEmax", "self", "diag"), Vinv=NULL,
       beta0=NULL, coinc=NULL, tuning=TRUE, abstol=0.01, vari=5, ordpen=2)
```

Arguments

- **x**: A numeric vector of x-values, length \( n \). Contains \( p \leq n \) distinct values.
- **y**: A numeric vector of observed y-values of length \( n \).
- **w**: A numeric vector of positive weights of length \( n \), which will be used as the diagonal of the covariance matrix sigma. The weights do not have to sum to \( n \), but will be transformed to do so internally. If sigma is given, w should be NULL.
- **sigma**: Estimates of the residual (co-)variances. Can be a positive numeric vector of length \( p \), giving estimates for the variance at each of the distinct x-values. If it is a vector of length 1, a vector of length \( p \) will be created from it by repetition. It is also possible to provide a whole covariance matrix, if correlated data are present. If sigma is given, w should be NULL.
- **a**: The left boundary of the interval, on which the spline is defined. If coinc=TRUE, the spline is zero to the left of this value. By default a is equal to the minimal x-value.
- **b**: The right boundary of the interval, on which the spline is defined. If coinc=TRUE, the spline is zero to the right of this value. By default b is equal to the maximal x-value.
- **g**: The number of inner knots of the spline, the default is 10.
- **k**: The degree of the spline. Default is 3 (cubic spline).
- **constr**: Shape constraint for the fit. Can be one of "unimodal" (default), "none", "invuni" (inverse-unimodal), "isotonic" or "antitonic".
- **penalty**: Which penalty on the B-spline coefficients should be used? Possible choices are "none" (no penalty should be used), "diff" (default) for the differences penalty of order ordpen and "sigEmax" for the sigmoid Emax penalty and "self" for a self-defined penalty. In the latter case, Vinv and beta0 have to be provided. In the other cases they should be NULL.
If a self-defined penalty on the B-spline coefficients should be used, \( \text{Vinv} \) is the penalty matrix. \( \text{Vinv} \) is a \((g + k + 1) \times (g + k + 1)\) matrix. Default: NULL.

If a self-defined penalty on the B-spline coefficients should be used, \( \text{beta0} \) is the penalty vector of length \( g+k+1 \). Default: NULL.

Should the outer knots of the knot sequence be coincident with the boundary knots or not? Default is NULL and altering has no effect, if a pre-defined penalty is used. If penalty="self", it has to be specified as TRUE or FALSE.

Should the tuning parameter lambda be optimized with or without consideration of the shape constraint? Default: TRUE. Changing tuning has no effect, when constr="none" or penalty="none".

If \( \text{sigma} = \text{NULL} \), the iteration between estimation of the residual standard deviation and the B-spline coefficients is stopped, when the absolute change in the variance estimate is less than \( \text{abstol} \). Default: 0.01. Changing \( \text{abstol} \) has no effect, when \( \text{sigma} \) is provided.

Variance parameter \( \sigma^2 \) in the full-rank precision matrix of the prior for \( \beta \). By default 5.

Order of the difference penalty. Only has an effect, if penalty="diff". By default 2.

This function combines implementations of the spline methods described in Koellmann et al. Given paired data \((x_1, y_1), \ldots, (x_n, y_n)\) it is possible to fit regression splines using the B-spline basis and the maximum likelihood approach. If the spline is unrestricted, the problem reduces to a simple linear regression problem. If the spline is restricted to be unimodal, inverse unimodal, isotonic or antitonic, this leads to a quadratic programming problem. If a penalty is used on the spline coefficients, the tuning parameter is chosen via restricted maximum likelihood (REML).

The data should contain repeated measurements at certain points on the \( x \)-axis (at least 2 for each point), so that a start estimate of the residual variance can be calculated. Then the function iterates between estimation of the spline coefficients and of the variance. Both estimates will be weighted, if weights are given. If there is only one measurement (e.g. an already aggregated (mean) measurement), the function expects an input in \( \text{sigma} \), a vector with variances at each \( x \)-value. The reciprocal variances are used as weights in this case (no further weights can be used).

If no penalty is used, the number of estimable B-spline coefficients, which is \( g+k+1 \), equals the number of distinct \( x \)-values. \( g \) and \( k \) have to be chosen accordingly.

A list of

- \( \text{coef} \) The vector of estimated B-spline coefficients.
- \( \text{x} \) The original vector of \( x \)-values.
- \( \text{fitted.values} \) The fitted values at each \( x \)-value.
- \( \text{unimod.func} \) A function that evaluates the estimated spline at arbitrary \( x \)-values.
- \( \text{lambdaopt} \) The optimal tuning parameter found via REML.
sigma  The estimated residual variance. If the input for sigma was not NULL, sigma equals the input.
degree  The degree k of the spline.
knotsequence  The sequence of knots (length g+2k+2) used for spline fitting.
g  The number g of inner knots.
a  The left boundary of the domain [a,b].
b  The right boundary of the domain [a,b].
variter  The number of iterations used to estimate the spline coefficients and the variance.

Author(s)
Claudia Koellmann

References

See Also
unimat, equiknots

Examples
```r
# generate some data
x <- sort(rep(0:5,20))
n <- length(x)
set.seed(41333)
func <- function(mu){rnorm(1,mu,0.05)}
y <- sapply(dchisq(x,3),func)

# plot of data
plot(jitter(x),y,xlab="x (jittered)")

# fit with default settings
fit <- unireg(x,y,g=5)

# fit without penalty (we can use at most g=2 inner knots if k=3)
fit2 <- unireg(x,y,penalty="none",g=2)

# fit without penalty and without constraint
# (does not differ from fit2 with constraint in this case)
fit3 <- unireg(x,y,penalty="none",g=2,constr="none")

# plot of true and fitted functions
z <- seq(0,5,by=0.01)
plot(jitter(x),y,xlab="x (jittered)"#,ylim=c(-0.15,0.35))
points(z,dchisq(z,3),type="l",col="grey",lwd=2)
```
```r
# estimated variance
fit$sigma
fitR$sigma

## Not run:
# fit with isotonic, antitonic and inverse-unimodal constraint (just for completeness)
fit4 <- unireg(x,y,constr="antitonic",g=5)
fit5 <- unireg(x,y,constr="isotonic",g=5)
fit6 <- unireg(x,y,constr="invuni",g=5)

points(z,fit4$unimod.func(z),type="l",col="orange",lwd=2)
points(z,fit5$unimod.func(z),type="l",col="brown",lwd=2)
points(z,fit6$unimod.func(z),type="l",col="yellow",lwd=2)

# suppose only aggregated data had been given
means <- c(mean(y[1:20]),mean(y[21:40]),mean(y[41:60]),mean(y[61:80]),
mean(y[81:100]),mean(y[101:120]))
sigma <- c(sd(y[1:20]),sd(y[21:40]),sd(y[41:60]),sd(y[61:80]),sd(y[81:100]),sd(y[101:120]))^2

# unimodal fit with differences penalty
fit7 <- unireg(x=unique(x),y=means,g=5,w=NULL,sigma=sigma,abstol=NULL)
plot(unique(x),means,pch=19,ylim=range(y))
points(z,dchisq(z,type="l",col="grey",lwd=2)
points(z,fit7$unimod.func(z),type="l",col="green",lwd=2)
legend("bottomright", legend = c("true mean function", "observed mean values",
"diff. penalized unimodal fit for means"),
col=c("grey","black","green"),lty=c(1,1,1),lwd=c(2,0,2),pch=c(NA,19,NA))

## End(Not run)
```
Index

*Topic models
   unireg, 4
*Topic nonparametric
   unireg, 4
*Topic regression
   unireg, 4

equiknots, 1, 3, 6
unimat, 2, 3, 6
unireg, 2, 3, 4