Package ‘AdaptFitOS’

October 12, 2022

Title Adaptive Semiparametric Additive Regression with Simultaneous Confidence Bands and Specification Tests

Version 0.69

Date 2022-07-11

Imports mgcv, SemiPar

Depends nlme, MASS, splines

Description Fits semiparametric additive regression models with spatially adaptive penalized splines and computes simultaneous confidence bands and associated specification (lack-of-fit) tests. Simultaneous confidence bands cover the entire curve with a prescribed level of confidence and allow us to assess the estimation uncertainty for the whole curve. In contrast to point-wise confidence bands, they permit statements about the statistical significance of certain features (e.g. bumps) in the underlying curve. The method allows for handling of spatially heterogeneous functions and their derivatives as well as heteroscedasticity in the data. See Wiesenfarth et al. (2012) <doi:10.1080/01621459.2012.682809>.

License GPL (>= 2)

NeedsCompilation yes

Author Manuel Wiesenfarth [aut, cre],
Tatyana Krivobokova [aut],
Matt Wand [ctb] (Some R code from package Semipar),
Catherine Loader [cph, ctb] (C code from locfit package),
Jiayang Sun [cph, ctb] (C code from locfit package),
Lucent Technologies [cph] (C code from locfit package)

Maintainer Manuel Wiesenfarth <m.wiesenfarth@dkfz.de>

Repository CRAN

Date/Publication 2022-07-21 13:00:01 UTC

R topics documented:

AdaptFitOS-package .................................................. 2
asp2 ................................................................. 4
aspFormula .......................................................... 8
aspHetero ............................................................ 10
AdaptFitOS-package

Adaptive Semiparametric Additive Regression with Simultaneous Confidence Bands and Specification Tests

Description

Fits semiparametric regression models with spatially adaptive penalized splines and computes simultaneous confidence bands and associated specification (lack-of-fit) tests.

For computation of the critical value for simultaneous confidence bands based on Hotelling’s volume-of-tube formula, some functions of the libtube library by Catherine Loader (see package locfit) are used. See the references for details on the construction of the confidence bands.

Details

The DESCRIPTION file:

Package: AdaptFitOS
Title: Adaptive Semiparametric Additive Regression with Simultaneous Confidence Bands and Specification Tests
Version: 0.69
Date: 2022-07-11
Authors@R: c(person("Manuel", "Wiesenfarth", email = "m.wiesenfarth@dkfz.de", role = c("aut", "cre")), person("Tatyana", "Krivobokova", email = "tatyana.krivobokova@dkfz.de", role = c("aut")), person("Matt Wand", email = "matt.wand@lancaster.ac.uk", role = c("ctb")), person("Jiayang Sun", email = "jiayang.sun@lancaster.ac.uk", role = c("cph", "ctb")), person("Lucent Technologies", role = "cph", comment = "C code from locfit package"))
Imports: mgcv, SemiPar
Depends: nlme, MASS, splines
License: GPL (>=2)
Author: Manuel Wiesenfarth [aut, cre], Tatyana Krivobokova [aut], Matt Wand [ctb] (Some R code from package SemiPar), Jiayang Sun [cph, ctb] (C code from locfit package), Lucent Technologies [cph] (C code from locfit package)
Maintainer: Manuel Wiesenfarth <m.wiesenfarth@dkfz.de>

Index of help topics:

AdaptFitOS-package Adaptive Semiparametric Additive Regression with Simultaneous Confidence Bands and Specification Tests
asp2 Fit a semiparametric regression model with spatially adaptive penalized splines
AdaptFitOS-package

aspFormula An asp formula
aspHetero Estimate varying residual variance
default.knots Compute default knots for a given x vector
fitted.asp Fitted values for semiparametric regression.
plot.asp Plots fitted curves or their derivatives
predict.asp Semiparametric regression prediction.
residuals.asp Residuals for semiparametric regression.
scbM Calculate simultaneous confidence bands for penalized splines
sigma Extract estimated varying residual variance
summary.asp Summaries and hypothesis tests

Model estimation using the mixed model representation of penalized splines in combination with simultaneous probability calculations based on the volume-of-tube formula enable the simultaneous inference directly, that is, without resampling methods.

The function 

\texttt{asp2()} 

is used to fit the model. Using the resulting \texttt{asp} object, fitted curves or their derivatives can be plotted with \texttt{plot.asp} and information on the parametric effects as well as specification tests for the nonparametric effects can be printed using \texttt{summary.asp}.

See Wiesenfarth et al (2012) for technical details and Wiesenfarth (2012, Chapter 5.1) for some more details on the use of the package (including a demonstration on how plots in Wiesenfarth et al are obtained).

Author(s)

Manuel Wiesenfarth and Tatyana Krivobokova

References


See Also

\texttt{spm} (package SemiPar)
asp2

fit a semiparametric regression model with spatially adaptive penalized splines

Description

Fits semiparametric additive regression models using the mixed model representation of penalized splines with spatially adaptive penalties. It includes the availability of simultaneous confidence bands (also for the derivatives of the smooth curves) and B-spline basis functions. Note that random effects, autocorrelations and interaction surfaces are not supported. Further, only Gaussian responses are supported. Also note that estimated curves are centered to have zero mean. See aspHetero for incorporation of heteroscedastic errors, scbM for some more details on the simultaneous confidence bands and summary.asp for computation of associated specification (lack-of-fit) tests.

Usage

asp2(form, spar.method = "REML", contrasts=NULL, omit.missing = NULL, returnFit=FALSE, niter = 20, niter.var = 50, tol=1e-6, tol.theta=1e-6, control=NULL)

Arguments

form

a formula describing the model to be fitted. See aspFormula for further information. Note, that an intercept is always included, whether given in the formula or not.

spar.method

method for automatic smoothing parameter selection. May be "REML" (restricted maximum likelihood) or "ML" (maximum likelihood).

contrasts

an optional list. See the contrasts.arg of model.matrix.default.

omit.missing

a logical value indicating whether fields with missing values are to be omitted.

niter

a maximum number of iterations for the mean estimation, default is 20.

niter.var

a maximum number of iterations for the variance of random effects estimation, default is 50.

tol

tolerance for the convergence criterion. Default is 1e-6.

tol.theta

tolerance for the convergence criterion (smoothing parameter function routine). Default is 1e-6.

returnFit

a logical value indicating whether the fitted object should be returned when the maximum number of iterations is reached without convergence of the algorithm. Default is FALSE.

control

see lmeControl in the documentation to nlme.
Details

See Wiesenfarth et al (2012) for technical details and Wiesenfarth (2012, Chapter 5.1) for some more details on the use of the package (including a demonstration on how plots in Wiesenfarth et al are obtained).

Value

A list object of class asp containing the fitted model. The components are:

- **fitted**: fitted values.
- **coef.mean**: estimated mean coefficients.
- **design.matrices**: design matrices both for knots und subknots.
- **x**: x values.
- **knots**: knots.
- **y.cov**: estimated covariance matrix of the response.
- **random.var**: estimated covariance matrix of the random effects.
- **subknots**: subknots.
- **coef.random**: estimated spline coefficients of the covariance matrix of the random effects.
- **var.random.var**: estimated variance of the spline coefficients of the covariance matrix of the random effects.
- **fit**: mimics fit object of lme().
- **info**: information about the inputs.
- **aux**: auxiliary information such as variability estimates.

References


Semiparametric Regression Cambridge University Press.
https://web.stat.tamu.edu/~carroll/semiregbook/


See Also

- **gam** (in package ‘mgcv’), **lme** (in package ‘nlme’)
Examples

#######
## scatterplot smoothing
x <- 1:1000/1000
mu <- exp(-400*(x-0.6)^2)+
  5*exp(-500*(x-0.75)^2)/3+2*exp(-500*(x-0.9)^2)
y <- mu+0.5*rnorm(1000)

#fit with default knots
y.fit <- asp2(y~f(x,adap=TRUE))
plot(y.fit,residuals=TRUE,lwd=2,scb.lwd=2,scb.lty="dashed")
# with shaded confidence region.
# Use scb.lty="blank" to plot the shades only.
  plot(y.fit,residuals=TRUE,shade=TRUE,scb.lty="blank")

## Not run:
## Model with heteroscedastic errors
attach(mcycle)
y=accel
kn1 <- default.knots(times,20)
# fit model with constant residual variance
  fit= asp2(accel~f(times,basis="os",degree=3,knots=kn1,adap=FALSE),
niter = 20, niter.var = 200)

# fit model with varying residual variance
fith=aspHetero(fit,times,tol=1e-8)
op <- par(mfrow = c(1,3))
plot(fit);plot(fith)
#sigma() returns the fitted varying residual variance
plot(sort(times),sigma(fith)[order(times)],type="l")
par(op)

## additive models
x1 <- 1:300/300
x2 <- runif(300)
mu1 <- exp(-400*(x1-0.6)^2)+
  5*exp(-500*(x1-0.75)^2)/3+2*exp(-500*(x1-0.9)^2)
mu2 <- sin(2*pi*x2)
y2 <- mu1+mu2+0.3*rnorm(300)

y2.fit <- asp2(y2~f(x1,adap=TRUE)+f(x2,adap=TRUE))
# switch off adaptive fitting for the first function
  y21.fit <- asp2(y2-f(x1,adap=FALSE)+f(x2,adap=TRUE))
op <- par(mfrow = c(2, 2))
plot(y2.fit)
plot(y21.fit)
par(op)

## scatterplot smoothing with specified knots and subknots
x <- 1:400/400
mu <- sqrt(x*(1-x))*sin((2*pi*(1+2^((9-4*6)/5)))/(x+2^((9-4*6)/5)))
y <- mu+0.2*rnorm(400)

kn <- default.knots(x,80)
kn.var <- default.knots(kn,20)

y.fit <- asp2(y=f(x,knots=kn))
y.fit2 <- asp2(y=f(x,knots=kn,var.knots=kn.var,adap=TRUE))
op <- par(mfrow = c(1, 2))
plot(y.fit)
plot(y.fit2)
par(op)

# more examples
beta=function(l,m,x)
  return(gamma(l+m)*(gamma(l)*gamma(m))^(-1)*x^(l-1)*(1-x)^(m-1))

f1 = function(x) return((0.6*beta(30,17,x)+0.4*beta(3,11,x))*1/0.958)
f2 = function(z) return((sin(2*pi*(z-0.5))^2)*1/.3535)
f3 = function(z)
  return((exp(-400*(z-0.6)^2)+5/3*exp(-500*(z-0.75)^2)+2*exp(-500*(z-0.9)^2))*1/0.549)

set.seed(1)
N <- 500
x1 = runif(N,0,1)
x2 = runif(N,0,1)
x3 = runif(N,0,1)

kn1 <- default.knots(x1,40)
kn2 <- default.knots(x2,40)
kn3 <- default.knots(x3,40)
kn.var3 <- default.knots(kn3,5)

y <- f1(x1)+f2(x2)+f3(x3)+0.3*rnorm(N)

# semiparametric model
fit1= asp2(y~x1+f(x2,basis="os",degree=3,knots=kn2,adap=FALSE)
  +f(x3,basis="os",degree=3,
    knots=kn3,var.knots=kn.var3,adap=FALSE),
    niter = 20, niter.var = 200)
summary(fit1)
plot(fit1,pages=1)

# all effects flexible
# fit model with all smoothing parameters constant
fit2a= asp2(y=f(x1,basis="os",degree=3,knots=kn1,adap=FALSE)
  +f(x2,basis="os",degree=3,knots=kn2,adap=FALSE)
  +f(x3,basis="os",degree=3,knots=kn3,adap=FALSE),
    niter = 20, niter.var = 200)
plot(fit2a,pages=1)
# fit model with last smoothing parameter adaptive
fit2b = asp2(y~f(x1,basis="os",degree=3,knots=kn1,adap=FALSE)  
   +f(x2,basis="os",degree=3,knots=kn2,adap=FALSE)  
   +f(x3,basis="os",degree=3,knots=kn3,adap=TRUE,  
   var.knots=kn.var3,var.basis="os",var.degree=3),  
   niter = 20, niter.var = 200)

# plot smoothing parameter function for covariate x3.
# Note that in the case of B-splines additional knots are added,
# see references.
plot(seq(0,1,length.out=42), fit2b$y.cov/fit2b$random.var[85:126],  
   ylab=expression(lambda(x3)),xlab="x3",type="l",lwd=3)

# compute 95% simultaneous confidence bands.
# You could skip this and use "fit2b" instead of "scb2b" later on, however,
# if N is large, computing the SCBs various times can take some time
# if you don't need fitted values and bounds for all covariate points
# (can be computationally intensive due to large matrix dimensions),
# set calc.stdev=F such that these are not computed.
scb2b<- scbM(fit2b,calc.stdev=FALSE)
plot(scb2b,pages=1)

# plot only f(x2).
plot(scb2b,select=2,mfrow=c(1,1),lwd=3,ylab="f(x2)",xlab="x2")

# plot.asp returns fitted values and confidence limits,
# if you only need the returned object set plot=FALSE
pscb=plot(scb2b,plot=FALSE)
# add pointwise confidence intervals to the plot
polygon(c(pscb$grid.x[[2]]), c(pscb$fitted[[2]]+1.96*pscb$Stdev[[2]],  
   rev(pscb$fitted[[2]]-1.96*pscb$Stdev[[2]])),  
   col = grey(0.85), border = NA)
lines(pscb$grid.x[[2]],pscb$lcb[[2]],lty="dotted",lwd=3)
lines(pscb$grid.x[[2]],pscb$fitted[[2]],lwd=3)
lines(pscb$grid.x[[2]],pscb$ucb[[2]],lty="dotted",lwd=3)

# plot first derivative of f(x1).
# Useful to check statistical significance of certain features (such
# as bumps) in a curve.
scb2bdrv<- scbM(fit2b,drv=1,calc.stdev=FALSE)
plot(scb2bdrv,select=1)
# the following would give the same result
#x11();plot(fit2b,select=1,drv=1)
# different style
plot(scb2bdrv,select=1,scb.lty="blank",  
   shade=TRUE,shade.col="steelblue")

## End(Not run)
Description

A formula to be used in asp2.

Dummies for categorical covariates are constructed automatically if a variable is given as factor (with contrasts as set by options("contasts") or specified by a list in argument contrasts). Note that only parametric interactions are supported and that interacting covariates have to be multiplied beforehand and given as a new variable in the formula. Smooth terms are given by

\[
f(x, \text{basis}="os", \text{degree}=3, \text{knots}, \text{var.knots}, \text{var.basis}, \text{var.degree}, \text{adap}=\text{TRUE})
\]

with the following arguments:

Arguments

- **x**
  - the covariate

- **basis**
  - the spline basis function to be used. "trunc.poly" for truncated polynomials, "tps" for thin plate splines and "os" for B-splines (default).

- **degree**
  - the degree of the basis. In the case of B-splines also a vector of the form c(p,q) with p the B-spline degree and q the penalty order (the integrated q-th squared derivative is penalized, see references). If only a scalar is given q is chosen such that p=2*q-1. Defaults are degree=3 (basis="tps"), degree=1 (basis="trunc.poly") and degree=c(3,2) (basis="os"), respectively.

- **knots**
  - the knots to be used. Using e.g. kn=default.knots(x,40) beforehand leads to 40 quantile based knots in the case of "tps" and "trunc.poly" bases. In the case of B-splines ("os"), knots are always equidistant and are automatically generated with the number equal to the length of the vector of knots given plus boundary knots. If no knots are given the number of knots is automatically chosen to be equal to floor(n/max(4, floor(n/35)) - 1).

- **adap**
  - TRUE for spatially adaptive smoothing parameter

- **var.knots**
  - the knots for the spline basis for adaptively estimating the smoothing parameter. If missing the number of knots is automatically chosen to be equal to floor(knots/max(4, floor(knots/35)) - 1).

- **var.basis**
  - spline basis function for adaptive smoothing parameter estimation. If missing, the same basis as for estimation of f is used.

- **var.degree**
  - spline degree for adaptive smoothing parameter estimation. If missing, the same degree as for estimation of f is used.

- **spar**
  - the smoothing parameter if desired. Usually this is left unspecified, such that the smoothing parameter is estimated by restricted maximum likelihood (see references). Currently doesn’t work for basis="os".

- **df**
  - the number of degrees of freedom corresponding to the REML choice of smoothing parameter if desired. Usually this is left unspecified, such that the smoothing parameter is estimated by restricted maximum likelihood (see references). Currently doesn’t work for basis="os".
Estimate varying residual variance

Description

Estimates a varying residual variance on basis of an asp object. Resulting object can be plotted with simultaneous confidence bands corrected for heteroscedasticity.

Usage

```r
aspHetero(object, xx, nknots=5, knots=NULL, basis="os",
        degree=c(3,2), tol=1e-8, niter=100, niter.var=250)
```

Arguments

- **object**: an asp object.
- **xx**: the covariate.
- **nknots**: the number of knots. Does not apply when knots are given.
- **knots**: the knots. Does not apply if basis="os". Otherwise, if NULL nknots equidistant knots are used.
- **basis**: the spline basis: "os" (default), "trunc.poly" or "tps".
- **degree**: the spline degree (and penalty order in case of B-splines). Defaults to c(3,2).
- **tol**: tolerance for the convergence criterion. Default is 1e-8.
- **niter**: a maximum number of iterations for residual variance function estimation, default is 100.
- **niter.var**: a maximum number of iterations for the variance of random effects estimation within the residual variance function estimation routine, default is 250.

Value

An object of class asp with varying variances, with additional element `sigma` including information on the spline of the varying variance.

References

**Examples**

```r
attach(mcycle)

y=accel
kn1 <- default.knots(times,20)
# fit model with constant residual variance
fit= asp2(accel~f(times,basis="os",degree=3,knots=kn1,adap=FALSE),
       niter = 20, niter.var = 200)

# fit model with varying residual variance
fith=aspHetero(fit,times,tol=1e-8)
op <- par(mfrow = c(1,3))
plot(fit);plot(fith)
#sigma() returns the fitted varying residual variance
plot(sort(times),sigma(fith)[order(times)],type="l")
par(op)
```

**default.knots**

*Compute default knots for a given x vector*

**Description**

Computes default knots for a given x vector.

**Usage**

```r
default.knots(x, num.knots, knotchoice="quantiles")
```

**Arguments**

- `x` The covariate. Note that for B-splines, only the range of x is considered.
- `num.knots` The number of knots. Defaults to `floor(n/max(4, floor(n/35)) - 1)`.  
- `knotchoice` Either "equidistant" or "quantiles" for equidistant and quantile based knots, respectively. Note that in case of B-splines, knots are always equidistant.

**fitted.asp**

*Fitted values for semiparametric regression.*

**Description**

Extracts fitted values from a semiparametric regression fit object.

**Usage**

```r
## S3 method for class 'asp'
fitted(object,...)
```
Arguments

object a fitted asp object as produced by asp2().
... other possible arguments.

Details

Extracts fitted from a semiparametric regression fit object. The fitted are defined to be the set of values obtained when the predictor variable data are substituted into the fitted regression model.

Value

The vector of fitted.

See Also

plot.asp, predict.asp, summary.asp, residuals.asp.

Examples

data(fossil,package="SemiPar")
attach(fossil)
fit <- asp2(strontium.ratio~f(age))
plot(fit, bands=FALSE)
points(age,fitted(fit)-fit$coef[1],col="red")
Arguments

- **x**: an asp object created by `asp` or `aspHetero`.
- **select**: vector specifying which curves in an additive model should be plotted. If NULL, all curves are plotted.
- **drv**: the derivative order. Defaults to 0, i.e., the estimated curves themselves are plotted. First and second derivatives are supported. Does not apply to objects created by `scbM`.
- **bands**: TRUE in order to include simultaneous confidence bands.
- **grid**: number of points used for the plot, default value 50.
- **plot**: if FALSE no plot is given.
- **ylim**: vector or list of vectors of limits on y axes. If NULL limits are automatically chosen. If multiple curves are plotted and a two-dimensional vector is given, y axes for all curves will be equal. A list with length equal to the number of smooth curves in the model can be given to specify different y-axes for each smooth.
- **pages**: The number of pages over which to spread the output as in package `mgcv`. For example, if pages=1 then all terms will be plotted on one page in an automatic fashion. If pages=0 (default) all graphics settings are left as they are.
- **level**: the level of confidence (does not apply to objects created by `scbM`).
- **xlab**: label for the x axis. A list with length equal to the number of smooth curves in the model can be given to specify different labels for each smooth.
- **ylab**: label for the y axis. A list with length equal to the number of smooth curves in the model can be given to specify different labels for each smooth.
- **scb.lwd**: line width for simultaneous confidence bands.
- **scb.lty**: line type for simultaneous confidence bands. Use scb.lty="blank", if you only want to plot the shades.
- **shade**: set to TRUE to produce shaded regions as simultaneous confidence bands for smooths.
- **shade.col**: define the color used for shading confidence bands.
- **residuals**: if TRUE, partial residuals are added to the plot.
- **residuals.col**: color of partial residuals.
- **rug**: adds a rug representation (1-d plot) of the data to the plot.
- **bayes**: FALSE for simultaneous confidence bands with (approximate) frequentist coverage probability, TRUE for (approximate) Bayesian coverage probability. See Krivobokova et al. (2010) for details.
- **...**: further arguments to be passed to plot()

Details

`plot.asp()` first calls `scbM` and then `plot.scbm()` to plot an asp object. If plotting takes long (because of a large data set) and you want to plot multiple times with different settings, use `scbM` and then plot the resulting `scbm` object with `plot.scbm()`. Estimated fits are centred to have zero mean. The simultaneous confidence bands have (approximate) frequentist coverage probabilities with automatic bias correction (see references).
predict.asp

Value

- `grid.x`: list of the grid values used
- `fitted`: list of the fitted values on the grid
- `lcb`: list of the lower bounds of the confidence bands
- `ucb`: list of the upper bounds of the confidence bands
- `drv`: the derivative order
- `Stdev.fit`: the standard deviations on the grid
- `ylim`: list of ylim used for plotting
- `residuals`: the partial residuals.

References


See Also

- `plot.spm` in package SemiPar

Examples

```r
# see asp2()
```

Description

Takes a fitted asp object produced by asp2 and obtains predictions at new data values.

Usage

```r
## S3 method for class 'asp'
predict(object, newdata, se, ...)
```
Arguments

object

a fitted asp object as produced by asp2(). Does not work with basis="os".

newdata

a data frame containing the values of the predictors at which predictions are required. The columns should have the same name as the predictors. Further, minima and maxima should currently coincide with those of the predictors.

se

when this is TRUE standard error estimates are returned for each prediction. The default is FALSE.

... other arguments.

Details

Takes a fitted asp object produced by asp2() and obtains predictions at new data values as specified by the ‘newdata’ argument. If ‘se=TRUE’ then standard error estimates are also obtained.

Value

If se=FALSE then a vector of predictions at ‘newdata’ is returned. If se=TRUE then a list with components named ‘fit’ and ‘se’ is returned. The ‘fit’ component contains the predictions. The ‘se’ component contains standard error estimates.

Author(s)

Manuel Wiesenfarth, based on implementation of M.P. Wand (package SemiPar).

See Also

plot.asp, summary.asp

Examples

data(fossil,package="SemiPar")
attach(fossil)
fit <- asp2(strontium.ratio~f(age, basis="tps"))
newdata.age <- data.frame(age=c(90,100,110,120,130))
preds <- predict(fit,newdata=newdata.age,se=TRUE)
print(preds)

# Use predict to avoid centering of smooths in case of scatterplot
# smoothing
fit <- asp2(strontium.ratio~f(age,basis="tps"))
newdata.age <- data.frame(age=seq(90,130,length.out=50) )
preds <- predict(fit,newdata=newdata.age,se=TRUE)
plot(age,strontium.ratio)
lines(newdata.age$age,preds$fit,col="red")
lines(unlist(newdata.age),preds$fit+2*preds$se,col="blue")
lines(unlist(newdata.age),preds$fit-2*preds$se,col="green")
Description

Extracts residuals from a semiparametric regression fit object.

Usage

```r
## S3 method for class 'asp'
residuals(object, ...)
```

Arguments

- `object` a fitted asp object as produced by asp2().
- `...` other possible arguments.

Details

Extracts residuals from a semiparametric regression fit object. The residuals are defined to be the difference between the response variable and the fitted values.

Value

The vector of residuals.

See Also

- `plot.asp`, `predict.asp`, `summary.asp`, `fitted.asp`

Examples

```r
data(fossil, package="SemiPar")
attach(fossil)
fit <- asp2(strontium.ratio~f(age))
plot(age, residuals(fit))
abline(0,0)
```
scbM

Calculate simultaneous confidence bands for penalized splines

Description

Calculates simultaneous (uniform) confidence bands for the mixed model representation of penalized splines based on volume-of-tube formula. Simultaneous confidence bands cover the entire curve with a prescribed level of confidence and allow us to assess the estimation uncertainty for the whole curve. In contrast to pointwise confidence bands, they permit statements about the statistical significance of certain features in the underlying curve.

Usage

scbM(object, select=NULL, drv=0, level=0.95, div=1000, calc.stdev=TRUE, bayes=FALSE)

Arguments

object: an asp object.
select: vector specifying which curves in an additive model should be considered. If NULL, all curves are considered.
drv: the derivative order. Defaults to 0, i.e. the estimated function itself is plotted. First and second derivatives are supported.
level: level of confidence.
div: precision for the integral used for calculation of the length of the curve, default is 1000.
calc.stdev: TRUE to compute standard deviation and confidence bands for each value of the covariates. Computationally intensive for large data sets. Use plot.scbm() or plot.asp() to compute standard deviation and bounds only for a grid. If FALSE only critical values are computed.
bayes: FALSE for confidence bands with (approximate) frequentist coverage probability, TRUE for (approximate) Bayesian coverage probability. See Krivobokova et al. (2010) for details.

Details

Returns a scbm object and prints critical values. The resulting confidence bands have (approximate) frequentist coverage probabilities with automatic bias correction (see references). Makes use of the libtube library by Catherine Loader (see package locfit).

Value

A list object of class scbm containing

aspobject: an asp object.
drv: the derivative order.
crit
sigma2
a list of critical values.
the variance of the residuals.
cov.coef
a list of covariance matrices of spline coefficients in the mixed model framework.
Stdev
the standard deviations of estimates. Only given if calc.stdev=TRUE.
fitted
a list of fitted values. Only given if calc.stdev=TRUE.
lcb
a list of lower bounds of confidence bands. Only given if calc.stdev=TRUE.
ucb
a list of upper bounds of confidence bands. Only given if calc.stdev=TRUE.

References


Examples

```r
## Not run:
beta=function(l,m,x)
return(gamma(l+m)*(gamma(l)*gamma(m))^(-1)*x^(l-1)*(1-x)^(m-1))
f1 = function(x) return((0.6*beta(30,17,x)+0.4*beta(3,11,x))*1/0.958)
f2 = function(z) return((sin(2*pi*(z-0.5))^2)*1/.3535)
f3 = function(z)
return((exp(-400*(z-0.6)^2)+5/3*exp(-500*(z-0.75)^2)+2*exp(-500*(z-0.9)^2))*1/0.549)
center=function(x) return(x-mean(x))

set.seed(1)
N <- 500
x1 = runif(N,0,1)
x2 = runif(N,0,1)
x3 = runif(N,0,1)

kn1 <- default.knots(x1,40)
k2n <- default.knots(x2,40)
k3n <- default.knots(x3,40)
k.var3 <- default.knots(kn3,5)
y <- f1(x1)+f2(x2)+f3(x3)+0.3*rnorm(N)

# fit model with last smoothing parameter adaptive
fit2b= asp2(y~f(x1, basis="os", degree=3, knots=kn1, adap=FALSE)
+ f(x2, basis="os", degree=3, knots=kn2, adap=FALSE)
+ f(x3, basis="os", degree=3, knots=kn3, adap=TRUE,
```
## sigma

**Extract estimated varying residual variance**

### Description

Extracts the estimated varying residual variance on basis of an object created by `aspHetero()`.

### Usage

```r
sigma(object)
```

### Arguments

- `object` an object created by `aspHetero()`.

### Examples

```r
#see aspHetero()
```
**Summary**

Takes a fitted `asp` object produced by `asp2()` and summarises the fit, including tests for significance of nonparametric effects as well as their deviation from a parametric fit.

**Usage**

```r
## S3 method for class 'asp'
summary(object,test1=FALSE,test2=FALSE,signif=0.05,...)
```

**Arguments**

- `object`: a fitted `asp` object as produced by `asp2()`.
- `test1`: `TRUE` in order to include a test for significance of a nonparametrically estimated effect. The test corresponds to checking whether the zero line is entirely inside the simultaneous confidence band.
- `test2`: `TRUE` in order to include the nonparametric specification test proposed in Wiesenfarth et al. (2012). Only works with B-splines. The function under the null hypothesis is a polynomial of degree q-1 where q is the penalty order.
- `signif`: the significance level.
- `...`: other arguments.

**Details**

Produces tables for the linear (parametric) and non-linear (nonparametric) components. The linear table provides coefficient estimates, standard errors and p-values. The non-linear table provides degrees of freedom values and other information including tests for significance of nonparametric effects as well as their deviation from a parametric fit. See Wiesenfarth et al (2011, 2012) and Wiesenfarth (2012) for details on the hypothesis tests.

**Value**

The function generates summary tables.

**References**

*Semiparametric Regression* Cambridge University Press.  
[https://web.stat.tamu.edu/~carroll/semiregbook/](https://web.stat.tamu.edu/~carroll/semiregbook/)

Wiesenfarth, M., Krivobokova, T., & Sperlich, S. (2011)  
A Volume-of-tube based Test for Penalized Splines Estimators.  


**See Also**

`plot.asp`, `predict.asp`

**Examples**

```r
data(onions,package="SemiPar")
attach(onions)
log.yield <- log(yield)
fit <- asp2(log.yield~location+f(dens, degree=c(3,2)))
summary(fit,test1=TRUE,test2=TRUE)
```
Index

* **adaptive**
  asp2, 4
* **aplot**
  plot.asp, 12
* **dplot**
  plot.asp, 12
  scbM, 17
* **models**
  asp2, 4
  fitted.asp, 11
  predict.asp, 14
  residuals.asp, 16
  summary.asp, 20
* **nonlinear**
  asp2, 4
* **package**
  AdaptFitOS-package, 2
* **regression**
  asp2, 4
  fitted.asp, 11
  predict.asp, 14
  residuals.asp, 16
  summary.asp, 20
* **smooth**
  asp2, 4
  aspHetero, 10
  fitted.asp, 11
  predict.asp, 14
  residuals.asp, 16
  scbM, 17
  sigma, 19
  summary.asp, 20
  default.knots, 11
  fitted.asp, 11, 16
  gam, 5
  lme, 5
  mgcv, 13
  model.matrix.default, 4
  plot.asp, 3, 12, 12, 15, 16, 21
  plot.scbm(plot.asp), 12
  plot.spm, 14
  predict.asp, 12, 14, 16, 21
  residuals.asp, 12, 16
  scbM, 4, 13, 17
  sigma, 19
  spm, 3
  summary.asp, 3, 4, 12, 15, 16, 20

AdaptFitOS (AdaptFitOS-package), 2
AdaptFitOS-package, 2
asp, 13
asp2, 4, 9
aspFormula, 4, 8
aspHetero, 4, 10, 13