

Package ‘sme’

February 9, 2018

Version 1.0.2

Date 2018-02-09

Title Smoothing-Splines Mixed-Effects Models

Depends R (>= 2.15.2), splines, lattice, stats

Description Fit smoothing-splines mixed-effects models to replicated functional data sets and visualise the results.

License GPL (>= 3)

URL <https://github.com/mberk/sme>

BugReports <https://github.com/mberk/sme/issues>

NeedsCompilation yes

Author Maurice Berk [aut, cre]

Maintainer Maurice Berk <maurice@mauriceberk.com>

Repository CRAN

Date/Publication 2018-02-09 16:43:56 UTC

R topics documented:

AICc	2
BICn	3
getRoughnessMatrix	4
inflammatory	5
MTB	5
plot.sme	6
plotSmeDiagnostic	7
plotSmeModel	8
plotSmeRaw	9
sme	10
sme.data.frame	12
sme.list	14
smeObject	16

Index	18
--------------	-----------

AICc

*The corrected AIC***Description**

Calculates the *corrected* AIC (AICc) of Hurvich and Tsai (1989). The AICc modifies the standard AIC with a correction for small sample sizes.

Usage

```
AICc(object)
```

Arguments

`object` a fitted model object for which there exists a `logLik` method to extract the corresponding log-likelihood or an object inheriting from class `logLik`

Details

AIC is an asymptotic result and may be inappropriate when the sample size is small, the number of estimated model parameters is large, or both. For the specific case of a linear model with homogeneous errors then Hurvich and Tsai (1989) derived a corrected AIC (AICc) which includes a correction for small sample sizes as $AICc = AIC + (2*k*(k+1))/(n-k-1)$ where *AIC* is the standard AIC, *k* is the number of parameters in the model and *n* is the number of observations.

While this is an exact result, it only applies in the very specific circumstances in which it was derived. However, as Burnham and Anderson (2002) point out, whenever the sample size is small some form of correction to the standard AIC is necessary, to the extent that they argue the AICc of Hurvich and Tsai (1989) should be used regardless of context unless a specific correction can be derived. In fact Burnham and Anderson (2004) go so far as to argue that it should be used irrespective of sample size as it tends to the standard AIC when *n* is large.

Value

A numeric value with the AICc of the model

Author(s)

Maurice Berk <maurice.berk01@imperial.ac.uk>

References

- Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint
- Hurvich, C. M. & Tsai, C.-L. (1989). *Regression and Time Series Model Selection in Small Samples*. *Biometrika*, 76, 297-307
- Burnham, K. P. & Anderson, D. R. (2002). *Model Selection and Multimodel Inference: a Practical Information-theoretic Approach*. Springer
- Burnham, K. P. & Anderson, D. R. (2004). *Multimodel Inference: Understanding AIC and BIC in Model Selection*. *Sociological Methods Research*, 33, 261-304

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
AICc(fit)
```

BICn*An alternative BIC for longitudinal models*

Description

Calculates the BIC as $-2 * \log\text{-likelihood} + \log(n) * npar$ for a longitudinal model where $npar$ is the number of parameters in the fitted-model and n is the *number of subjects*

Usage

```
BICn(object, ...)
```

Arguments

object	a fitted longitudinal model object
...	some methods for this generic function may require additional arguments

Details

When applying the BIC in a longitudinal context, there is some debate as to whether the sample size should be taken to mean the number of subjects or the total number of observations across all subjects (see Section 7.3 of Hedeker and Gibbons, 2006).

Assuming the default BIC function accounts for the latter case, this generic function can be implemented for longitudinal models where the number of subjects can be extracted in order to calculate the BIC under the alternative definition.

Value

A numeric value with the BIC of the longitudinal model, with the penalty taken as a function of the number of subjects as described.

Author(s)

Maurice Berk <maurice.berk01@imperial.ac.uk>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint
Hedeker, D. & Gibbons, D. R. (2006). *Longitudinal Data Analysis*. Wiley

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
BICn(fit)
```

getRoughnessMatrix *Extract the roughness matrix used in an SME model fit*

Description

Given an sme object, returns the roughness matrix which can be used to quickly calculate the integrated squared second derivative of the mean and individual level curves

Arguments

object a fitted SME model object returned from the sme function

Details

The parameters of the SME model are estimated using maximum *penalized* likelihood, where the penalized likelihood is the standard likelihood with the addition of penalty terms corresponding to the roughness of the mean and individual level curves. Typically the roughness of a curve is quantified as its integrated squared second derivative. Green and Silverman (1994) show that, for a natural cubic spline interpolating the vector of points f , there exists a *roughness matrix* G such that the integrated squared second derivative is $f'Gf$ where f' denotes f transposed. For details on constructing the matrix G , either refer to the original source of Green and Silverman (1994) or it may prove easier to access Berk and Montana (2009) where they can be found in the appendix.

Value

The roughness matrix corresponding to the SME model fit

Author(s)

Maurice Berk <maurice.berk01@imperial.ac.uk>

References

- Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint
- Berk, M. & Montana, G. (2009). *Functional modelling of microarray time series with covariate curves*. *Statistica*, 2-3, 158-187
- Green, P. J. & Silverman, B. W. (1994). *Nonparametric Regression and Generalized Linear Models*. Chapman and Hall

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
getRoughnessMatrix(fit)
```

inflammatory	<i>Example gene transcript from an experiment on children suffering from an inflammatory condition</i>
--------------	--

Description

The inflammatory data frame has 318 rows and 3 columns.

Format

This data frame contains the following columns:

y a numeric vector of observed gene expression levels

tme a numeric vector of time points corresponding to the observations in y

ind a factor indicating which subject the observations in y belong to

Source

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

MTB	<i>Ten example gene transcripts from an experiment investigating the in vitro transcriptional response to M.Tuberculosis</i>
-----	--

Description

The MTB data frame has 290 rows and 4 columns.

Format

This data frame contains the following columns:

y a numeric vector of observed gene expression levels

tme a numeric vector of time points corresponding to the observations in y

ind a factor indicating which subject the observations in y belong to

variable a factor indicating which gene transcript the observations in y belong to

Source

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

`plot.sme`*Visualisation of smoothing-splines mixed-effects models*

Description

A visualisation of the smoothing-splines mixed-effects model fit is obtained. For `type="model"`, a scatter plot of the observations overlaid with the fitted mean and (optionally) individual curves. Confidence limits for the mean curve can optionally also be shown. See `plot.sme.model` for further details. For `type="raw"`, a trellis plot of the raw data for each subject, with the fitted individual curves optionally overlaid. See `plot.sme.raw` for further details. For `type="diagnostic"`, a diagnostic plot of the model residuals. See `plot.sme.diagnostic` for further details.

Usage

```
## S3 method for class 'sme'  
plot(x, type="model", ...)
```

Arguments

<code>x</code>	an object inheriting from class <code>sme</code> , representing a fitted smoothing-splines mixed-effects model
<code>type</code>	the type of plot desired, currently either <code>"model"</code> for visualising the model fit, <code>"raw"</code> for visualising the raw data or <code>"diagnostic"</code> for visualising the model residuals
<code>...</code>	further arguments to <code>plotSmeModel</code> , <code>plotSmeRaw</code> or <code>plotSmeDiagnostic</code>

Value

Depending on `type`, either a plot of the fitted model, a plot of the raw data or a diagnostic plot of the model residuals

Author(s)

Maurice Berk <maurice@mauriceberk.com>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

See Also

[plotSmeModel](#), [plotSmeRaw](#), [plotSmeDiagnostic](#)

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
plot(fit,type="model")
plot(fit,type="raw")
plot(fit,type="diagnostic")
```

plotSmeDiagnostic	<i>Diagnostic plot of the residuals for a fitted smoothing-splines mixed-effects model</i>
-------------------	--

Description

This type of plot is heavily inspired by Wu and Zhang (2006). A plot containing four panels is produced: (1) standardised residuals against fitted values, (2) standardised residuals against time, (3) standardised residuals against response and (4) a normal Q-Q plot of the standardised residuals. These plots can be used to assess how well the fitted model has accounted for the temporal correlation and between-subject variance in the data.

Usage

```
plotSmeDiagnostic(x)
```

Arguments

`x` an object inheriting from class `sme`, representing a fitted smoothing-splines mixed-effects model

Value

A four panel plot as described above.

Author(s)

Maurice Berk <maurice@mauriceberk.com>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint
Wu, H. and Zhang, J.-T. (2006). *Nonparametric Regression Methods for Longitudinal Data Analysis*. Wiley

See Also

[plot.sme](#), [plotSmeRaw](#), [plotSmeModel](#)

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
plotSmeDiagnostic(fit)
```

plotSmeModel

Visualisation of a fitted smoothing-splines mixed-effects model

Description

The observations used to fit the given smoothing-splines mixed-effects model are shown as a scatter plot and the fitted mean curve is overlaid. Optionally, the fitted individual curves are also shown. Also, the 95% confidence band for the fitted mean curve can optionally be displayed.

Usage

```
plotSmeModel(x, xlab="Time", ylab="Y", showIndividuals=T, showConfidenceBands=F,
  col.meanCurve="red", ...)
```

Arguments

<code>x</code>	an object inheriting from class <code>sme</code> , representing a fitted smoothing-splines mixed-effects model
<code>xlab</code>	the label for the x-axis, defaulting to "Time"
<code>ylab</code>	the label for the y-axis, defaulting to "Y"
<code>showIndividuals</code>	if TRUE then each of the fitted subject specific curves will be displayed in the plot as dashed lines, otherwise only the mean curve will be shown
<code>showConfidenceBands</code>	if TRUE, then the 95% confidence band for the fitted mean curve will be shown
<code>col.meanCurve</code>	the color to be used for the fitted mean curve and, if <code>showConfidenceBands</code> is TRUE, the confidence band as well
<code>...</code>	additional arguments passed through to <code>plot</code> when producing the scatter plot of observations

Value

A plot visualising the fitted smoothing-splines mixed-effects model as described above.

Author(s)

Maurice Berk <maurice@mauriceberk.com>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

See Also

[plot.sme](#), [plotSmeRaw](#), [plotSmeDiagnostic](#)

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
plotSmeModel(fit,showIndividuals=FALSE)
plotSmeModel(fit,showConfidenceBands=TRUE)
```

plotSmeRaw

Visualisation of raw data for a smoothing-splines mixed-effects model

Description

A visualisation of the raw data used to fit the smoothing-splines mixed-effects model is obtained. This is presented as a trellis plot where each panel shows the observations for one of the subjects as a scatter plot (with optional lines connecting the observations if `showRawLines` is TRUE). If `showModelFits` is TRUE, then these observations are overlaid with the fitted individual curve for that subject.

Usage

```
plotSmeRaw(x,xlab="Time",ylab="Y",mainTitle="",showModelFits=TRUE,showRawLines=FALSE,...)
```

Arguments

<code>x</code>	an object inheriting from class <code>sme</code> , representing a fitted smoothing-splines mixed-effects model
<code>xlab</code>	the label for the x-axis, defaulting to "Time"
<code>ylab</code>	the label for the y-axis, defaulting to "Y"
<code>mainTitle</code>	the main title for the plot, defaulting to none
<code>showModelFits</code>	if TRUE, the fitted individual curve for each subject will be overlaid on their observations in their panel
<code>showRawLines</code>	if TRUE, the raw observations will be connected by lines
<code>...</code>	additional arguments passed through to <code>xypLOT</code>

Value

A trellis plot as described above.

Author(s)

Maurice Berk <maurice@mauriceberk.com>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

See Also

[plot.sme](#), [plotSmeModel](#), [plotSmeDiagnostic](#)

Examples

```
data(MTB)
fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")])
plotSmeRaw(fit,showModelFits=FALSE)
plotSmeRaw(fit,showModelFits=FALSE,showRawLines=TRUE)
plotSmeRaw(fit)
```

sme

Smoothing-splines mixed-effects models

Description

This generic function fits a smoothing-splines mixed-effects model

Usage

```
sme(object, tme, ind, verbose=F, lambda.mu=NULL, lambda.v=NULL, maxIter=500,
knots=NULL, zeroIntercept=F, deltaEM=1e-3, deltaNM=1e-3, criteria="AICc",
initial.lambda.mu=10000, initial.lambda.v=10000, normalizeTime=FALSE, ...)
```

Arguments

object	either a vector of observations, a data.frame object or a list of vectors of observations. The method functions <code>sme.data.frame</code> and <code>sme.list</code> are documented separately
tme	either a vector of time points corresponding to the observations given in <code>object</code> or a list of vectors of time points in the case of <code>sme.list</code> . Ignored in the case of <code>sme.data.frame</code>
ind	a factor (or a vector that can be coerced to a factor) of subject identifiers corresponding to the observations given in <code>object</code> or a list of factors of subject identifiers in the case of <code>sme.list</code> . Ignored in the case of <code>sme.data.frame</code>
verbose	if TRUE, debug information will be output while fitting the model
lambda.mu	smoothing parameter used for the fixed-effect function. If NULL, the optimal values for this and <code>lambda.v</code> will be found according to <code>criteria</code> using Nelder-Mead search
lambda.v	smoothing parameter used for the random-effects functions. If NULL, the optimal values for this and <code>lambda.mu</code> will be found according to <code>criteria</code> using Nelder-Mead search

<code>maxIter</code>	maximum number of iterations to be performed for the EM algorithm
<code>knots</code>	location of spline knots. If NULL, an incidence matrix representation will be used. See ‘Details’
<code>zeroIntercept</code>	experimental feature. If TRUE, the fitted values of the fixed- and random-effects functions at the intercept will be zero
<code>deltaEM</code>	convergence tolerance for the EM algorithm
<code>deltaNM</code>	(relative) convergence tolerance for the Nelder-Mead optimisation
<code>criteria</code>	one of "AICc", "AIC", "BICn" or "BICn" indicating which criteria to use to score a particular combination of <code>lambda.mu</code> and <code>lambda.v</code> in the Nelder-Mead search
<code>initial.lambda.mu</code>	value to initialise the smoothing parameter for the fixed-effects to in the Nelder-Mead search. See details below
<code>initial.lambda.v</code>	value to initialise the smoothing parameter for the random-effects to in the Nelder-Mead search. See details below
<code>normalizeTime</code>	should time be normalized to lie in $[0,1]$? See details below
<code>...</code>	additional arguments to <code>sme.data.frame</code> or <code>sme.list</code>

Details

Prior to package version 0.9, starting values for the smoothing parameters in the Nelder-Mead search were fixed to 10000 for both `lambda.mu` and `lambda.v`. As it turns out, the appropriate scale for the smoothing parameters depends on the scale for `tme` and so `tme` will now automatically be rescaled to lie in $[0,1]$ and much smaller initial values for the smoothing parameters will be used, although these can now optionally be changed to achieve best results. To reproduce results obtained using previous versions of the package, set `initial.lambda.mu=10000`, `initial.lambda.v=10000` and `normalizeTime=FALSE`.

The default behaviour is to use an incidence matrix representation for the smoothing-splines. This works well in most situations but may incur a high computational cost when the number of distinct time points is large, as may be the case for irregularly sampled data. Alternatively, a basis projection can be used by giving a vector of knots of length (much) less than the number of distinct time points.

Value

An object of class `sme` representing the smoothing-splines mixed-effects model fit. See `smeObject` for the components of the fit and `plot.sme` for visualisation options

Author(s)

Maurice Berk <maurice.berk01@imperial.ac.uk>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

See Also

[smeObject](#), [sme.data.frame](#), [sme.list](#), [plot.sme](#)

Examples

```
data(MTB)
fit.AIC <- sme(MTB[MTB$variable==6031,c("y","tme","ind")],criteria="AIC")
fit.BICN <- sme(MTB[MTB$variable==6031,c("y","tme","ind")],criteria="BICN")
fit.BICn <- sme(MTB[MTB$variable==6031,c("y","tme","ind")],criteria="BICn")
fit.AICc <- sme(MTB[MTB$variable==6031,c("y","tme","ind")],criteria="AICc")

fit <- sme(MTB[MTB$variable==6031,c("y","tme","ind")],lambda.mu=1e5,lambda.v=1e5)

data(inflammatory)
system.time(fit <- sme(inflammatory,knots=c(29.5,57,84.5),deltaEM=0.1,deltaNM=0.1))
```

<code>sme.data.frame</code>	<i>Smoothing-splines mixed-effects model fit(s) from a data.frame object</i>
-----------------------------	--

Description

Carry out one or more independent smoothing-splines mixed-effects model fits simultaneously

Usage

```
## S3 method for class 'data.frame'
sme(object,tme,ind,verbose=F,lambda.mu=NULL,lambda.v=NULL,maxIter=500,
knots=NULL,zeroIntercept=F,deltaEM=1e-3,deltaNM=1e-3,criteria="AICc",
initial.lambda.mu=10000,initial.lambda.v=10000,normalizeTime=FALSE,...)
```

Arguments

<code>object</code>	a <code>data.frame</code> with named variables <code>y</code> , <code>tme</code> , <code>ind</code> and, optionally, <code>variable</code> . The first three represent observations, corresponding time points and corresponding subjects respectively. If <code>variable</code> is missing then these are used to carry out a single model fit. If <code>variable</code> is present then it denotes variable membership, and a separate smoothing-splines mixed-effects model is fit to each unique variable
<code>tme</code>	for consistency with the generic function. Ignored in this case
<code>ind</code>	for consistency with the generic function. Ignored in this case
<code>verbose</code>	if TRUE, debug information will be output while fitting the model(s)
<code>lambda.mu</code>	in the case of carrying out a single model fit, either a smoothing parameter to be used for the fixed-effect function or NULL if the optimal values for this and <code>lambda.v</code> should be found according to <code>criteria</code> using Nelder-Mead search. For the case of multiple model fits, either a single smoothing parameter to be used for all fits, or a vector of smoothing parameters, one for each fit, or NULL if Nelder-Mead search should be used to find the optimal values for this and <code>lambda.v</code> for all variables

<code>lambda.v</code>	in the case of carrying out a single model fit, either a smoothing parameter to be used for the random-effects functions or NULL if the optimal values for this and <code>lambda.mu</code> should be found according to <code>criteria</code> using Nelder-Mead search. For the case of multiple model fits, either a single smoothing parameter to be used for all fits, or a vector of smoothing parameters, one for each fit, or NULL if Nelder-Mead search should be used to find the optimal values for this and <code>lambda.mu</code> for all variables
<code>maxIter</code>	maximum number of iterations to be performed for the EM algorithm
<code>knots</code>	location of spline knots. If NULL, an incidence matrix representation will be used. See ‘Details’
<code>zeroIntercept</code>	experimental feature. If TRUE, the fitted values of the fixed- and random-effects functions at the intercept will be zero
<code>deltaEM</code>	convergence tolerance for the EM algorithm
<code>deltaNM</code>	(relative) convergence tolerance for the Nelder-Mead optimisation
<code>criteria</code>	one of "AICc", "AIC", "BICn" or "BIC" indicating which criteria to use to score a particular combination of <code>lambda.mu</code> and <code>lambda.v</code> in the Nelder-Mead search
<code>initial.lambda.mu</code>	value to initialise the smoothing parameter for the fixed-effects to in the Nelder-Mead search. See details below
<code>initial.lambda.v</code>	value to initialise the smoothing parameter for the random-effects to in the Nelder-Mead search. See details below
<code>normalizeTime</code>	should time be normalized to lie in $[0,1]$? See details below
<code>...</code>	additional arguments used when carrying out multiple fits, specifically <code>numberOfThreads</code> indicating the number of threads used to carry out the multiple fits in parallel. See <code>sme.list</code> for details

Details

Prior to package version 0.9, starting values for the smoothing parameters in the Nelder-Mead search were fixed to 10000 for both `lambda.mu` and `lambda.v`. As it turns out, the appropriate scale for the smoothing parameters depends on the scale for `tme` and so `tme` will now automatically be rescaled to lie in $[0,1]$ and much smaller initial values for the smoothing parameters will be used, although these can now optionally be changed to achieve best results. To reproduce results obtained using previous versions of the package, set `initial.lambda.mu=10000`, `initial.lambda.v=10000` and `normalizeTime=FALSE`.

The default behaviour is to use an incidence matrix representation for the smoothing-splines. This works well in most situations but may incur a high computational cost when the number of distinct time points is large, as may be the case for irregularly sampled data. Alternatively, a basis projection can be used by giving a vector of knots of length (much) less than the number of distinct time points.

Value

In the case of a single model fit, an object of class `sme`. For multiple model fits, a list of such objects. See `smeObject` for the components of the fit and `plot.sme` for visualisation options

Author(s)

Maurice Berk <maurice@mauriceberk.com>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

See Also

[smeObject](#), [sme](#), [sme.list](#), [plot.sme](#)

Examples

```
## Not run: data(MTB)
## Not run: system.time(fits <- sme(MTB,numberOfThreads=1))
## Not run: sapply(fits,logLik)

## Not run: system.time(fits <- sme(MTB,numberOfThreads=10))
## Not run: sapply(fits,logLik)
```

sme.list	<i>Carry out multiple independent smoothing-splines mixed-effects model fits simultaneously</i>
----------	---

Description

Carry out multiple independent smoothing-splines mixed-effects model fits simultaneously

Usage

```
## S3 method for class 'list'
sme(object, tme, ind, verbose=F, lambda.mu=NULL, lambda.v=NULL, maxIter=500,
knots=NULL, zeroIntercept=F, deltaEM=1e-3, deltaNM=1e-3, criteria="AICc",
initial.lambda.mu=10000, initial.lambda.v=10000, normalizeTime=FALSE, numberOfThreads=-1,
...)
```

Arguments

object	a list of vectors of observations
tme	a list of vectors of time points corresponding to the observations in object
ind	a list of factors (or vectors that can be coerced to factors) of subject identifiers corresponding to the observations in object
verbose	if TRUE, debug information will be output while fitting the model(s)
lambda.mu	either a single smoothing parameter to be used for the fixed-effect function for all fits, or a vector of smoothing parameters, one for each fit, or NULL if Nelder-Mead search should be used to find the optimal values for this and lambda.v for all fits

<code>lambda.v</code>	either a single smoothing parameter to be used for the random-effects functions for all fits, or a vector of smoothing parameters, one for each fit, or NULL if Nelder-Mead search should be used to find the optimal values for this and <code>lambda.mu</code> for all fits
<code>maxIter</code>	maximum number of iterations to be performed for the EM algorithm
<code>knots</code>	location of spline knots. If NULL, an incidence matrix representation will be used. See ‘Details’
<code>zeroIntercept</code>	experimental feature. If TRUE, the fitted values of the fixed- and random-effects functions at the intercept will be zero
<code>deltaEM</code>	convergence tolerance for the EM algorithm
<code>deltaNM</code>	(relative) convergence tolerance for the Nelder-Mead optimisation
<code>criteria</code>	one of "AICc", "AIC", "BICn" or "BIC" indicating which criteria to use to score a particular combination of <code>lambda.mu</code> and <code>lambda.v</code> in the Nelder-Mead search
<code>initial.lambda.mu</code>	value to initialise the smoothing parameter for the fixed-effects to in the Nelder-Mead search. See details below
<code>initial.lambda.v</code>	value to initialise the smoothing parameter for the random-effects to in the Nelder-Mead search. See details below
<code>normalizeTime</code>	should time be normalized to lie in $[0,1]$? See details below
<code>numberOfThreads</code>	The number of threads to use to fit the multiple smoothing-splines mixed-effects models simultaneously. When <code>numberOfThreads=-1</code> , as is the default, the OpenMP system will handle thread creation dynamically
<code>...</code>	additional arguments, currently not used

Details

Prior to package version 0.9, starting values for the smoothing parameters in the Nelder-Mead search were fixed to 10000 for both `lambda.mu` and `lambda.v`. As it turns out, the appropriate scale for the smoothing parameters depends on the scale for `tme` and so `tme` will now automatically be rescaled to lie in $[0,1]$ and much smaller initial values for the smoothing parameters will be used, although these can now optionally be changed to achieve best results. To reproduce results obtained using previous versions of the package, set `initial.lambda.mu=10000`, `initial.lambda.v=10000` and `normalizeTime=FALSE`.

The default behaviour is to use an incidence matrix representation for the smoothing-splines. This works well in most situations but may incur a high computational cost when the number of distinct time points is large, as may be the case for irregularly sampled data. Alternatively, a basis projection can be used by giving a vector of knots of length (much) less than the number of distinct time points.

Value

A list of objects of class `sme`. See `smeObject` for the components of the fit and `plot.sme` for visualisation options

Author(s)

Maurice Berk <maurice@mauriceberk.com>

References

Berk, M. (2012). *Smoothing-splines Mixed-effects Models in R*. Preprint

See Also

[smeObject](#), [sme](#), [sme.data.frame](#), [plot.sme](#)

smeObject

Fitted sme Object

Description

An object returned by the `sme` function, inheriting from class `sme`, representing a fitted smoothing-splines mixed-effects model

Value

An `sme` object must contain the following components.

<code>call</code>	a list containing an image of the <code>sme</code> call that produced the object.
<code>coefficients</code>	an $n+1$ by p matrix where n is the number of subjects and p is the number of knots used in the spline basis. The first row of the matrix corresponds to the fitted values of the mean curve at the knots, and the remaining n rows correspond to the fitted values of the individual deviations from the mean curve at the knots.
<code>fitted</code>	a vector of model fitted values corresponding to the original observations.
<code>logLik</code>	the log-likelihood of the fitted model.
<code>residuals</code>	a vector of model residuals.
<code>nobs</code>	the number of observations.
<code>data</code>	a data frame of the original data used to fit the model with variables <code>y</code> , <code>tme</code> and <code>ind</code> corresponding to observations, time points and subject identifiers respectively.
<code>df</code>	a vector consisting of named components <code>mu</code> and <code>v</code> corresponding to the degrees of freedom of the mean and subject curves respectively.
<code>smoothingParameters</code>	a vector of named components <code>mu</code> and <code>v</code> corresponding to the smoothing parameters for the mean and subject curves respectively.
<code>parameters</code>	a list with named components <code>sigmaSquared</code> and <code>D</code> corresponding to the error variance and (unregularized) variance of the random-effects respectively.
<code>iterations</code>	the number of iterations of the EM algorithm that ran before convergence (or the limit was reached)

`info` a numeric code indicating diagnostic information from the EM algorithm with zero indicating a successful run.

In some instances, an `sme` object may also contain the following components.

`knots` a vector of the knots used in the spline basis when they do not coincide with the unique design time points.

Author(s)

Maurice Berk <maurice.berk01@imperial.ac.uk>

See Also

[sme](#)

Index

AICc, [2](#)

BICn, [3](#)

getRoughnessMatrix, [4](#)

inflammatory, [5](#)

MTB, [5](#)

plot.sme, [6](#), [7](#), [9](#), [10](#), [12](#), [14](#), [16](#)

plotSmeDiagnostic, [6](#), [7](#), [9](#), [10](#)

plotSmeModel, [6](#), [7](#), [8](#), [10](#)

plotSmeRaw, [6](#), [7](#), [9](#), [9](#)

sme, [10](#), [14](#), [16](#), [17](#)

sme.data.frame, [12](#), [12](#), [16](#)

sme.list, [12](#), [14](#), [14](#)

smeObject, [12](#), [14](#), [16](#), [16](#)