`masal-package` *Multivariate Adaptive Splines for Analysis of Longitudinal Data*

Description

This package facilitates non-parametric longitudinal data analysis.

Details

Package:	masal
Type:	Package
Version:	1.0
Date:	2009-11-17

Very convenient to use. Accepts a longitudinal study data set and a nonparametric fit to the data is produced. Two functions are included:
`masal`
`print.masal`

Author(s)

Heping Zhang for C++ code and Yunxiao He for R version
Maintainer: Yunxiao He ⟨yunxiao.he@yale.edu⟩

References

Zhang, H. (1997), *Multivariate adaptive splines for analysis of longitudinal data.*
Zhang, H. and Singer, B. (1999) *Recursive partitioning in the health sciences*, Springer Verlag.

Examples

```
data(s1)
fit=masal(s1)
print(fit)
summary(fit)
```

`masal` *Multivariate Adaptive Splines for Analysis of Longitudinal Data*

Description

Multivariate Adaptive Splines for Analysis of Longitudinal Data

Usage

```
masal(data.masal, max.term=ncol(data.masal)-2, max.interaction=2,
          variance.list=c(1,1,0), enforce.term=0, max.iter=2)
```

Arguments

`data.masal`	Input data. See details for the format of the data.
`max.term`	Maximum number of terms allowed in the model (without counting intercept which is enforced all the time). Default is the number of covariates plus 1 (for time).
`max.interaction`	Maximum order of interaction allowed in the model. Default is 2.
`variance.list`	The covariance of random effect is allowed to be a quadratic function of time. `variance.list` is a vector of length three and of values at 0 or 1 to indicate whether certain terms should be included (1) or not (0) in the second degree polynomial. Default is that only the constant and linear terms are added. See details.
`enforce.term`	The function `masal` allows user to enforce certain covariates (or time) in the model. `enforce.term` should be a vector of integers indicating the covariates (or time) to be enforced. Default is none (0). See details.
`max.iter`	The model fitting with `masal` is an iterative process. `max.iter` sets the maximum number of iterations which serves as the stopping criterion. Default is 2.

Details

To run the function `masal`, the input data set `data.masal` must be a matrix of the format similar to the one used in SAS PROC MIXED. For example, the data file should look like

```
1001 0 2 28 2 1 40 506 10.6
1001 0 2 28 2 1 40 464 10.6
1001 0 2 28 2 1 40 304 9.8
1001 0 2 28 2 1 40 243 9.5
1001 0 2 28 2 1 40 234 9.6
1003 0 0 22 1 1 40 509 11.5
1003 0 0 22 1 1 40 502 11.5
1003 0 0 22 1 1 40 285 9.6
```

where the first column is the ID number, columns 2 - 7 are covariates (can be time dependent, denoted by $x.1, \cdots, x.6$), column 8 is time (denoted by $x.7$), and the response is in the last column. We suggest the user to sort the data by ID number (and break ties by time) while preparing the data set so that the observations for each subject are grouped together. If not appropriately sorted, the function `masal` will sort the data and a warning message is generated in the end.

The model fitted by the function `masal` is a special case of the model introduced in Section 10.5 of Zhang & Singer (1999) and only time-related random effect is considered here. Formally, the model is

$$Y_{ij} = f(t_{ij}, X_{ij}) + Z_{ij} b_i + \varepsilon_{ij},$$

where $Y_{ij}, X_{ij}, Z_{ij}, i = 1, \cdots, p, j = 1, \cdots, q_i$, represent response, covariates in fixed effect, and covariates in random effect for the i^{th} subject at its j^{th} time points t_{ij}. The fixed effect $f(t_{ij}, X_{ij})$, a smooth function, is estimated with the methods introduced in Chapter 9 of Zhang & Singer (1999) as discussed in Section 10.5 of the same book.

In the extreme case of `variance.list`=(0, 0, 0), no random effect is considered and a regression model is fitted. Otherwise, let I_v be the matrix truncated from I_3, the identity matrix of order 3, according to `variance.list` in the following way: the k^{th} column of I_3 is kept in I_v if and only if the k^{th} element of `variance.list` is 1, $k = 1, 2, 3$. Then $Z_{ij} = (1, \sqrt{t_{ij}}, t_{ij})I_v$, $b_i = [(b_{i1}, b_{i2}, b_{i3})I_v]'$ with $b_{ik} \sim N(0, \sigma_k^2)$, $\varepsilon_{ij} \sim N(0, \sigma_0^2)$, and all of b_{ij} and ε_{ij} are independent to each other. Let $v =$ `variance.list`, then the variance of Y_{ij} is $\sigma_0^2 + v[1]\sigma_1^2 + v[2]t_{ij}\sigma_2^2 + v[3]t_{ij}^2\sigma_3^2$ and the covariance between Y_{ij} and Y_{il} is $v[1]\sigma_1^2 + v[2]\sqrt{t_{ij}t_{il}}\sigma_2^2 + v[3]t_{ij}t_{il}\sigma_3^2$ for $j \neq l$. Under this model, the covariance matrix of Y_i is automatically positive definite under very general conditions, e.g., σ_0^2 is positive.

The numbers in `enforce.term` correspond to the order of the covariates (or time) appearing in the input data. For example, if we have `enforce.term` = (1,2), then the covariates corresponding to the second and the third columns in the input data, i.e., $x.1$ and $x.2$, are enforced in the final model. Note that time is treated the same way as other covariates and it is documented as the last covariate ($x.7$ for the data set presented above). Make sure the variables indicated in `enforce.term` do exist in `data.masal`. Otherwise, the `masal` function will exit with an error message.

If there are missing values in the data, the `masal` function will delete the whole observations with missing values, proceeds with the new data set, and a warning message is generated in the end.

If there is only one time point (one observation) for each subject in the data set `data.masal`, a regression model is fitted. When a regression model is fitted, any user specified setting of `variance.list` and `max.iter` has no effect.

Value

A list with the following components:

`call`	The call that produced this object
`data`	The data set actually used for the fitting. It may be different with the input data `data.masal` because of the possible preprocessing steps mentioned above.
`nsub`	Number of subjects in `data`
`ncolm`	Number of columns in `data`
`ntime`	A vector of integers indicating the number of time points (observations) for each subject in `data`
`Mobs`	The maximum number in the vector `ntime`
`nobs`	Total number of rows in `data`
`regression`	An indicator for whether a regression model instead of a mixed effect model has been fitted. 1 means yes and 0 otherwise.
`beta`	The estimated β coefficients for fixed effect
`beta.std`	The standard deviations of the estimated β coefficients

`out.fixed`	A data frame that is responsible for extracting the fixed effect. Each row of `out.fixed` includes the estimated β coefficient, its standard deviation, variables and corresponding knots for each term. In each row, column `variable` indicates the covariate (or time), column `knot.ind` indicates whether a knot is associated with the covariate (or time), and column `knot` is the value of the associated knot if `knot.ind` = 1 and `NA` if `knot.ind` = 0. Note that the dimension of `out.fixed` is allocated to be fit for the term of maximum order. The columns `variable` and `knot` are set to be `NA` after all the variables in one term (row) have been listed.
`out.term`	The total number of terms in fixed effect of the final model. It is equal to the number of rows in `out.fixed`.
`out.term.length`	A vector of integers indicating the order (number of variables) of each term in fixed effect of the final model
`random.effect`	A data frame including the estimation of the random coefficients b_i. The method to estimate b_i can be found in various references such as Section 6.2 of Crowder and Hand (1990).
`resd`	A data frame including the computed residuals. The columns `resp`, `resd`, and `resd.id` give the original response, the ordinary residual (difference between response the estimated fixed effect), and the individual residual (response - (estimated fixed effect + estimated random effect)). When regression model is fitted, the two types of residuals are the same.
`sigma.sq`	Estimated values of $(\sigma_0^2, (\sigma_1^2, \sigma_2^2, \sigma_3^2)I_v)$

Author(s)

Heping Zhang for C++ code and Yunxiao He for R version
Maintainer: Yunxiao He ⟨yunxiao.he@yale.edu⟩

References

Crowder, M.J. and Hand, D.J. (1990), *Analysis of repeated measures*, Chapman & Hall/CRC.
Zhang, H. (1997), *Multivariate adaptive splines for analysis of longitudinal data.*
Zhang, H. and Singer, B. (1999) *Recursive partitioning in the health sciences*, Springer Verlag.

See Also

`print.masal`

Examples

```
## fitting with default setting
data(s1)
fit=masal(s1)

## fitting with only fixed effect
## equivalent to regression
fit=masal(s1,5,2,c(0,0,0),c(1,2))
```

```
## fitting with a data set in which each subject has only one observation
## equivalent to regression
s2=s1[5*(1:100),]
fit=masal(s2,5,2,c(1,1,0),c(1,2))

## fitting with unsorted data and missing values
s2=s1
s2[500,]=s1[1,]
s2[1,]=s1[500,]
s2[300,1]=NA
fit=masal(s2,5,2,c(1,1,0),c(1,2))

## look at the output of masal
print(fit)
summary(fit)
fit$out.fixed
fit$random.effect
fit$resd
```

`print.masal` *Print a masal object*

Description

Print a summary of the model fitted by `masal`.

Usage

```
## S3 method for class 'masal':
print(x,...)
```

Arguments

`x`	Fitted `masal` object
`...`	Additional arguments to `print`

Details

The call that produced the object `x`, the fitted function for fixed effect, the estimated β coefficients and their standard deviations, and the estimated $(\sigma_0^2, (\sigma_1^2, \sigma_2^2, \sigma_3^2)I_v)$ are printed on the screen.

Author(s)

Heping Zhang for C++ code and Yunxiao He for R version
Maintainer: Yunxiao He ⟨yunxiao.he@yale.edu⟩

References

Zhang, H. (1997), *Multivariate adaptive splines for analysis of longitudinal data.*
Zhang, H. and Singer, B. (1999) *Recursive partitioning in the health sciences*, Springer Verlag.

See Also

`masal`

Examples

```
## fitting with default setting
data(s1)
fit=masal(s1)
print(fit)
```

`s1` *A Simulated Logitudinal Study Data*

Description

See Section 10.5.3 of Zhang (1999)

Format

A matrix with 500 rows and 9 columns including 100 subjects and observations at five time points for each subject.

Source

See Section 10.5.3 of Zhang (1999).

References

Zhang, H. and Singer, B. (1999) *Recursive partitioning in the health sciences.* Springer Verlag.