Package 'GLSME'

September 16, 2019

Type Package

Title Generalized Least Squares with Measurement Error

Version 1.0.5

Date 2019-09-15

Author Krzysztof Bartoszek <krzbar@protonmail.ch>

Maintainer Krzysztof Bartoszek <krzbar@protonmail.ch>

Description Performs linear regression with correlated predictors, responses and correlated measurement errors in predictors and responses, correcting for biased caused by these.

Depends R(>= 2.9.1), mvtnorm, corpcor

Suggests ape, mvSLOUCH

License GPL (>= 2) | file LICENCE

LazyLoad yes

Collate GLSME.R

NeedsCompilation no

Repository CRAN

Date/Publication 2019-09-15 22:50:06 UTC

R topics documented:

GLSME-package *Generalized Least Squares with Measurement Error*

Description

The code fits the general linear model with correlated data and observation error in both dependent and independent variables. The code fits the model

$$y = D\beta + r, r \sim N(0, V), V = \sigma^2 T + V_e + Var[U\beta|D],$$

where y is a vector of observed response variables, D is an observed design matrix, β is a vector of regression parameters to be estimated, σ^2T is a matrix representing the true residual variance, V_e is a matrix of known measurement variance in the response variable, and Var[Uβ|D] is a matrix representing effects of measurement error in the predictor variables (see Hansen and Bartoszek 2012).

Details

Package:	GLSME
Type:	Package
Version:	1.0.5
Date:	2019-09-15
License:	GPL (>= 2)
LazyLoad:	yes

The code fits the general linear model with correlated data and observation error in both dependent and independent variables. The code fits the model

$$y = D\beta + r, r \sim N(0, V), V = \sigma^2 T + V_e + Var[U\beta|D],$$

where y is a vector of observed response variables, D is an observed design matrix, β is a vector of regression parameters to be estimated, σ^2T is a matrix representing the true residual variance, V_e is a matrix of known measurement variance in the response variable, and Var[Uβ|D] is a matrix representing effects of measurement error in the predictor variables (see Hansen and Bartoszek 2012).

The estimation function is GLSME. It is an iterated (if the variance parameters are unknown) generalized least squares estimation procedure.

The motivation for the approach is that the observations and errors are correlated due to an underlying phylogeny but the program allows for any dependence structure.

In the mvSLOUCH package an alternative method of correcting for observation error is used. The error variance-covariance matrix enters the likelihood function by being added to the biological variance-covariance matrix.

Author(s)

Krzysztof Bartoszek Maintainer: <bartoszekkj@gmail.com>

References

Bartoszek, K. and Pienaar, J. and Mostad. P. and Andersson, S. and Hansen, T. F. (2012) A phylogenetic comparative method for studying multivariate adaptation. Journal of Theoretical Biology 314:204-215.

Hansen, T.F. (1997) Stabilizing selection and the comparative analysis of adaptation. Evolution 51:1341-1351.

Hansen, T.F. and Bartoszek, K. (2012) Interpreting the evolutionary regression: the interplay between observational and biological errors in phylogenetic comparative studies. Systematic Biology 61(3):413-425.

Hansen, T.F. and Pienaar, J. and Orzack, S.H. (2008) A comparative method for studying adaptation to randomly evolving environment. Evolution 62:1965-1977.

See Also

mvSLOUCH

Examples

```
n<-3 ## number of species
apetree<-ape::rtree(n)
### Define Brownian motion parameters to be able to simulate data under the Brownian motion model.
BMparameters<-list(vX0=matrix(0,nrow=2,ncol=1),Sxx=rbind(c(1,0),c(0.2,1)))
### Now simulate the data and remove the values corresponding to the internal nodes.
xydata<-mvSLOUCH::simulBMProcPhylTree(apetree,X0=BMparameters$vX0,Sigma=BMparameters$Sxx)
xydata<-xydata[(nrow(xydata)-n+1):nrow(xydata),]

x<-xydata[,1]
y<-xydata[,2]

yerror<-diag((stats::rnorm(n,mean=0,sd=0.1))^2) #create error matrix
y<-mvtnorm::rmvnorm(1,mean=y,sigma=yerror)[1,]
xerror<-diag((stats::rnorm(n,mean=0,sd=0.1))^2) #create error matrix
x<-mvtnorm::rmvnorm(1,mean=x,sigma=xerror)[1,]
GLSME(y=y, CenterPredictor=TRUE, D=cbind(rep(1, n), x), Vt=ape::vcv(apetree),
Ve=yerror, Vd=list("F",ape::vcv(apetree)), Vu=list("F", xerror))
```

GLSME *Estimate regression parameters with correlated observations measurement errors.*

Description

The GLSME function estimates parameters of a linear model via generalized least squares. It allows for correlated predictors and responses. Furthermore it allows for correlated measurement errors both in predictors and responses. The program specifically corrects for biase caused by these errors.

Usage

```
GLSME(y, D, Vt, Ve, Vd, Vu, EstimateVariance = c(TRUE, TRUE), CenterPredictor = TRUE,
InitialGuess = NULL, eps = 0.001, MaxIter = 50, MaxIterVar = 50, epsVar = 0.001,
OutputType = "short", Vttype = NULL, Vetype = NULL, Vdtype = NULL, Vutype = NULL,
ED = NULL, EDtype = "SingleValue")
```

Arguments

y	A vector of observed response variables.
D	a design matrix in which each column corresponds to a parameter to be estimated in the B-vector/matrix. Each entry in these columns corresponds to a data point (e.g. a species in comparative studies). The first column will typically be a column of ones, which will estimate an intercept. Columns with indicators for categorical fixed effects can also be added. Each regression variable is added as a column vector. The program will automatically estimate one coefficient for each column in the design matrix and these will be output in the order of the columns in the design matrix. Note that columns corresponding to "random effects", indicated by nonzero entry in the Vd matrix below, will be centered on their means unless the option CenterPredictor = FALSE is used to instruct the program to not do this. If there is to be an INTERCEPT the user needs to put into D a constant column of 1s.
Vt	The response biological residual covariance matrix (see Details).
Ve	The response observation error covariance matrix (see Details). observation errors in the response variable. In a comparative study in which the response consists of species means, this will typically be a diagonal matrix with squared standard errors of the means along the diagonal.
Vd	Represents the true variance structure for the predictor variables. (see Details).
Vu	The predictor observation variances (see Details)
EstimateVariance	Option to turn off estimation of the variance parameters. This is a vector of TRUE or FALSE values, where the first value corresponds to the true residual variance, and the others correspond to the rest of the true predictor variances. All the predictor variances can also be turned on or off jointly by providing a single TRUE or FALSE value. The default is to estimate all variance components. If a FALSE value is given the program assumes that the input variance matrices are exact.
CenterPredictor	TRUE or FALSE option to turn off automatic centering of predictors.
InitialGuess	Starting value for the regression in the iterated GLS. The default is NULL, in which case the program will use an OLS estimate. A specific starting value can be given as a vector of numbers corresponding to each entry in the B-vector. An additional number can also be given to specify the starting value of the residual variance parameter.
eps	tolerance for iterated GLS
MaxIter	maximum number of iterations for iterated GLS

MaxIterVar	maximum number of iterations for iterated GLS
epsVar	tolerance for estimating variance parameters in predictors
OutputType	should just the estimates be presented and their standard errors ("short") or more detailed information ("long")
Vttype	Vt matrix type (see Details)
Vetype	Ve matrix type (see Details)
Vdtype	Vd matrix type (see Details)
Vutype	Vu matrix type (see Details)
ED	the expected value of the design matrix, can be NULL then is estimated from the data
EDtype	if ED is provided then specifies what is provided, allowed values are : • "constant" ED is a number and each value of D has mean equal to this number • "variablemean" ED is a vector of length of number of variables, each value is a mean for the given predictor variable • NULL ED is assumed to be calculated

Details

The code fits the general linear model with correlated data and observation error in both dependent and independent variables. The code fits the model

$$y = D\beta + r, r \sim N(0, V), V = \sigma^2 V_t + V_e + Var[U\beta|D],$$

where y is a vector of observed response variables, D is an observed design matrix, β is a vector of parameters to be estimated, V_t is a matrix representing the true residual variance up to a scale parameter, σ^2, that is estimated by the program, V_e is a matrix of known measurement variance in the response variable, and Var[Uβ|D] is a matrix representing effects of measurement error in the predictor variables (see appendix of Hansen and Bartoszek 2012). To build the Var[Uβ|D] matrix, the program needs a known measurement variance matrix V_u and a true variance matrix V_{xt} for each of the predictor variables (these will be zero for fixed effects). The true variance matrices are assumed to be on the form $V_{xt} = \sigma_x^2 Sx$, where Sx is a matrix supplied by the user, and σ_x^2 is a scale parameter that the program estimates by maximum likelihood.

Note that this program cannot be used to fit parameters that enter nonlinearly into the variance or the design matrix, as the α in the adaptation-inertia model, but it can be used to fit the other parameters in such models conditionally on given values of the parameterized values of the matrices (and could hence be used as a subroutine in a program for fitting such models).

Three important notes for the user :

- The program does NOT assume there will be an intercept -> hence the user needs to provide a column on 1s in the design matrix if an intercept is desired.
- The program by default centres predictors (controlled by CenterPredictor). This means that estimates of fixed effects will be changed due to them absorbing the mean of the predictors. Using the centering has been found to improve estimation especially of variance constants (PredictorVarianceConstantEstimate and ResponseVarianceConstantEstimate see Value). The user should try out the option with CenterPredictor TRUE and FALSE (here fixed effects will not be effected) and compare results.

- The program uses a Monte Carlo procedure as part of the estimation algorithm therefore the user should run the code a couple of times to see stability, and combine the results by e.g. a (weighted) average or choose the best estimate according to e.g. the likelihood or R^2.

The program tries to recognize the structure of the `Vt`, `Ve`, `Vd` and `Vu` matrices passed (see the supplementary information to Hansen and Bartoszek 2012) otherwise the user can specify how the matrix looks like in the appropriate matrix type variable, these can be in the respective `Vttype`, `Vetype`, `Vdtype` or `Vutype` parameter:

- `"SingleValue"` the matrix variable is a single number that will be on the diagonal of the covariance matrix, used when the deviations are assumed to be uncorrelated and homoscedastic
- `"Vector"` the matrix variable is a vector each value corresponding to one of the variables and the covariance matrix will have that vector appropriately on its diagonal, if an element of the list has the value `"F"` then this means that the variable is a fixed effect and will get a 0 covariance matrix
- `"CorrelatedPredictors"` the matrix is a covariance matrix, it assumes that the observations are independent so the resulting covariance structure is block diagonal, if some of the variables are fixed effects then in the matrix the values of the corresponding rows and columns have to be 0 (this is a special case of BM with the second element equal to the identity matrix)
- `"MatrixList"` a list of length equal to the number of variables, each list element is the covariance structure for the given variable, if an element of the list has the value `"F"` then this means that the variable is a fixed effect and will get a 0 covariance matrix
- `"BM"` the matrix variable `Vx` is to be a list of two values, "Vx = Vx[[1]] then the first value corresponds to the variable vector covariance while the second will be the matrix of distances between species, if the first value is a number or vector then it is changed to a diagonal matrix, if some of the variables are fixed effects then in the matrix of the first element of the list the values of the corresponding rows and columns have to be 0
- `NULL` or `"Matrix"` the matrix is assumed calculated as given

Value

- GLSestimate the GLS estimates without any correction (centering the predictors CHANGES fixed effects)
- errorGLSestim the estimates of their standard errors
- BiasCorrectedGLSestimate the bias corrected estimates (centering the predictors CHANGES fixed effects)
- K the bias attenuation factor matrix
- R2 R^2 of the model with the GLS estimates not bias corrected
- BiasCorrectedR2 R^2 of the model with the GLS estimates bias corrected
- PredictorVarianceConstantEstimate if `EstimateVariance[2]` is `TRUE` then the estimates of the unknown variance constants for the predictors otherwise not present
- ResponseVarianceConstantEstimate if `EstimateVariance[1]` is `TRUE` then the estimate of the unknown variance constant for the response otherwise not present if the `outputType` variable is set to `"long"` then the following additional fields will be in the output :
- CovarianceGLSestimate estimate of the covariance matrix of the bias uncorrected GLS estimates

- CovarianceBiasCorrectedGLSestimate estimate of the covariance matrix of the bias corrected GLS estimates
- response the provided y vector
- design the provided design matrix D
- Vt the final used Vt matrix with the unknown variance constant incorporated (if estimated)
- Ve the final used Ve matrix
- Vd the final used Vd matrix with the unknown variance constant(s) incorporated (if estimated)
- Vu the final used Vu matrix

Author(s)

Krzysztof Bartoszek

References

Bartoszek, K. and Pienaar, J. and Mostad. P. and Andersson, S. and Hansen, T. F. (2012) A phylogenetic comparative method for studying multivariate adaptation. Journal of Theoretical Biology 314:204-215.

Hansen, T.F. (1997) Stabilizing selection and the comparative analysis of adaptation. Evolution 51:1341-1351.

Hansen, T.F. and Bartoszek, K. (2012) Interpreting the evolutionary regression: the interplay between observational and biological errors in phylogenetic comparative studies. Systematic Biology 61(3):413-425.

Hansen, T.F. and Pienaar, J. and Orzack, S.H. (2008) A comparative method for studying adaptation to randomly evolving environment. Evolution 62:1965-1977.

Examples

```
n<-3 ## number of species
apetree<-ape::rtree(n)
### Define Brownian motion parameters to be able to simulate data under the Brownian motion model.
BMparameters<-list(vX0=matrix(0,nrow=2,ncol=1),Sxx=rbind(c(1,0),c(0.2,1)))
### Now simulate the data and remove the values corresponding to the internal nodes.
xydata<-mvSLOUCH::simulBMProcPhylTree(apetree,X0=BMparameters$vX0,Sigma=BMparameters$Sxx)
xydata<-xydata[(nrow(xydata)-n+1):nrow(xydata),]

x<-xydata[,1]
y<-xydata[,2]

yerror<-diag((stats::rnorm(n,mean=0,sd=0.1))^2) #create error matrix
y<-mvtnorm::rmvnorm(1,mean=y,sigma=yerror)[1,]
xerror<-diag((stats::rnorm(n,mean=0,sd=0.1))^2) #create error matrix
x<-mvtnorm::rmvnorm(1,mean=x,sigma=xerror)[1,]
GLSME(y=y, CenterPredictor=TRUE, D=cbind(rep(1, n), x), Vt=ape::vcv(apetree),
Ve=yerror, Vd=list("F",ape::vcv(apetree)), Vu=list("F", xerror))
```

GLSME.predict *Prediction for a new observation using parameters estimated by the* GLSME *function*

Description

The function takes parameters estimated by the GLSME function and predicts the response for a new observation of predictors. It also returns confidence intervals on the prediction. The function is still under development.

Usage

```
GLSME.predict(xo, glsme.estimate, vy, vx, alpha = 0.95)
```

Arguments

xo	The new observed predictors. In a intercept is in the model then a 1 has to be included for it.
glsme.estimate	The output of the GLSME function. Has to have format "long".
vy	Residual variance, both biological and measurement error.
vx	Biological variance in predictor, NOT observation variance of predictor. If there is a predictor in the model then a 0 row and column have to included for it.
alpha	Level for confidence interval.

Value

BiasCorr Prediction using the bias corrected estimate.

- prediction Predicted value
- MSE Estimate of mean square error. They are calculated by the formula

$$v_y + x_o^T(MSE[\beta|XO])x_o + \beta'^T v_x \beta',$$

where β' is the bias corrected estimate of β.

- CI $1-\alpha$ level confidence intervals. They are calculated by the formula

$$\sqrt{1+1/n} * t_\alpha * (v_y + \beta'^T v_x \beta'),$$

where t_α is the $1-\alpha/2$ level quantile of the t-distribution with n-k degrees of freedom, k is the number of regression parameters to estimate, β' is the bias corrected estimate of β and n is the sample size used in the estimation.

BiasUncorr Prediction using the bias uncorrected estimate.

- prediction Predicted value
- MSE Estimate of mean square error. They are calculated by the formula

$$v_y + x_o^T(MSE[\beta|XO])x_o + \beta'^T v_x \beta',$$

where β' is the bias uncorrected estimate of β.

- CI $1-\alpha$ level confidence intervals. They are calculated by the formula

$$\sqrt{1+1/n} * t_\alpha * (v_y + \beta'^T v_x \beta'),$$

where t_α is the $1-\alpha/2$ level quantile of the t-distribution with n-k degrees of freedom, k is the number of regression parameters to estimate, β' is the bias uncorrected estimate of β and n is the sample size used in the estimation.

Author(s)

Krzysztof Bartoszek

References

Hansen, T.F. and Bartoszek, K. (2012) Interpreting the evolutionary regression: the interplay between observational and biological errors in phylogenetic comparative studies. Systematic Biology 61(3):413-425.

Examples

```
set.seed(12345)
n<-3 ## number of species
apetree<-ape::rtree(n)
### Define Brownian motion parameters to be able to simulate data under the Brownian motion model.
BMparameters<-list(vX0=matrix(0,nrow=2,ncol=1),Sxx=rbind(c(1,0),c(0.2,1)))
### Now simulate the data and remove the values corresponding to the internal nodes.
xydata<-mvSLOUCH::simulBMProcPhylTree(apetree,X0=BMparameters$vX0,Sigma=BMparameters$Sxx)
xydata<-xydata[(nrow(xydata)-n+1):nrow(xydata),]

x<-xydata[,1]
y<-xydata[,2]

yerror<-diag((stats::rnorm(n,mean=0,sd=0.1))^2) #create error matrix
y<-mvtnorm::rmvnorm(1,mean=y,sigma=yerror)[1,]
xerror<-diag((stats::rnorm(n,mean=0,sd=0.1))^2) #create error matrix
x<-mvtnorm::rmvnorm(1,mean=x,sigma=xerror)[1,]
glsme.res<-GLSME(y=y, CenterPredictor=TRUE, D=cbind(rep(1, n), x), Vt=ape::vcv(apetree),
Ve=yerror, Vd=list("F",ape::vcv(apetree)), Vu=list("F", xerror),OutputType="long")
GLSME.predict(c(1,1), glsme.res, vy=1, vx=rbind(c(0,0),c(0,1)))
```

Index