Package: mmmgee (via r-universe)

October 25, 2024

Title Simultaneous Inference for Multiple Linear Contrasts in GEE Models

Version 1.20

Imports stats, methods, Matrix, mvtnorm

Suggests geepack, testthat

Description Provides global hypothesis tests, multiple testing procedures and simultaneous confidence intervals for multiple linear contrasts of regression coefficients in a single generalized estimating equation (GEE) model or across multiple GEE models. GEE models are fit by a modified version of the 'geeM' package.

Date 2019-06-18

Author Robin Ristl [aut, cre], Lee McDaniel [ctb] (Author of 'geeM' package), Nick Henderson [ctb] (Author of 'geeM' package), Melanie Prague [ctb] (Contributor to 'geeM' package)

Maintainer Robin Ristl <robin.ristl@meduniwien.ac.at>

Depends R (>= 3.0.0)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

NeedsCompilation no

Date/Publication 2019-06-21 15:30:03 UTC

Repository https://rmaster1.r-universe.dev

RemoteUrl https://github.com/cran/mmmgee

RemoteRef HEAD

RemoteSha 8b35e1029e8b18e30ae3cf8aaade6826139c85eb

Contents

---

datasim	*Simulated Data Set With Three Endpoints*

Description

A data set was simulated with repeated observations of a continous outcome variable Y.lin, a count
data outcome Y.poi and a binary outcome Y.bin. In the simulation, the mean of an outcome variable
depends on the binary grouping variable gr.lang and one of the continuos predictors x1, x2 and
x3. Observations of all outcomes in the same subject, indicated by the variable id, are correlated.
Also x1, x2 and x3 are correlated within subjects. Data are independent between subjects. The
example code shows how to fit a marginal GEE model for each outcome variable and how to test,
with different methods, the null hypothesis that the grouping variable has no effect on any of the
three endpoints.

Usage

```
data(datasim)
```

Format

A data frame.

Examples

```
data(datasim)
head(datasim)
mod1<-geem2(Y.lin~gr.lang+x1,id=id,data=datasim,family="gaussian",corstr="exchangeable")
mod2<-geem2(Y.poi~gr.lang+x2,id=id,data=datasim,family="poisson",corstr="exchangeable")
mod3<-geem2(Y.bin~gr.lang+x3,id=id,data=datasim,family="binomial",corstr="exchangeable")
L1<-L2<-L3<-matrix(c(0,1,0),nrow=1)
mmmgee.test(list(mod1,mod2,mod3),L=list(L1,L2,L3),statistic="Wald",type="maximum",
 biascorr=TRUE,asymptotic=FALSE,closed.test=TRUE)
## Not run:
mmmgee.test(list(mod1,mod2,mod3),L=list(L1,L2,L3),statistic="score",closed.test=TRUE)
mmmgee.test(list(mod1,mod2,mod3),L=list(L1,L2,L3),statistic="score",type="quadratic",
 closed.test=TRUE)
mmmgee.test(list(mod1,mod2,mod3),L=list(L1,L2,L3),statistic="Wald",type="quadratic",
 biascorr=TRUE,asymptotic=FALSE,closed.test=TRUE)
mmmgee.test(list(mod1,mod2,mod3),L=list(L1,L2,L3),statistic="Wald",type="quadratic",
 scaled=TRUE,biascorr=TRUE,asymptotic=FALSE,closed.test=TRUE)
```

```
## End(Not run)
```

---

geem2   *Fit Generalized Estimating Equation Models*

Description

geem2 is a modified version of [geem](#) to fit generalized estimating equation models and to provide objects that can be used for simultaneous inference across multiple marginal models using [mmmgee](#) and [mmmgee.test](#). Like geem, geem2 estimates coefficients and nuisance parameters using generalized estimating equations. The link and variance functions can be specified by the user and the syntax is similar to [glm](#).

Usage

```
geem2(formula, id, waves = NULL, data = parent.frame(),
  family = gaussian, corstr = "independence", Mv = 1,
  weights = NULL, corr.mat = NULL, init.beta = NULL,
  init.alpha = NULL, init.phi = 1, scale.fix = FALSE,
  nodummy = FALSE, sandwich = TRUE, useP = TRUE, maxit = 20,
  tol = 1e-05, restriction = NULL, conv.criterion = c("ratio",
  "difference"))
```

Arguments

formula	a formula expression similar to that for [glm](#), of the form response~predictors. An offset is allowed, as in glm.
id	a vector identifying the clusters. By default, data are assumed to be sorted such that observations in a cluster are in consecutive rows and higher numbered rows in a cluster are assumed to be later. If NULL, then each observation is assigned its own cluster.
waves	a non-negative integer vector identifying components of a cluster. For example, this could be a time ordering. If integers are skipped within a cluster, then dummy rows with weight 0 are added in an attempt to preserve the correlation structure (except if corstr = "exchangeable" or "independent"). This can be skipped by setting nodummy=TRUE. When assessing missing values, waves are assumed to start at 1, starting at a larger integer is therefore computationally inefficient.
data	an optional data frame containing the variables in the model.
family	will determine the link and variance functions. The argument can be one of three options: a family object, a character string, or a list of functions. For more information on how to use family objects, see [family](#). If the supplied argument is a character string, then the string should correspond to one of the family objects. In order to define a link function, a list must be created with the components (LinkFun, VarFun, InvLink, InvLinkDeriv), all of which are

	vectorized functions. If the components in the list are not named as (LinkFun, VarFun, InvLink, InvLinkDeriv), then geem2 assumes that the functions are given in that order. LinkFun and VarFun are the link and variance functions. InvLink and InvLinkDeriv are the inverse of the link function and the derivative of the inverse of the link function and so are decided by the choice of the link function.
corstr	a character string specifying the correlation structure. Allowed structures are: "independence", "exchangeable", "ar1", "m-dependent", "unstructured", "fixed", and "userdefined". Any unique substring may be supplied. If "fixed" or "userdefined", then corr.mat must be specified. If "m-dependent", then Mv is relevant.
Mv	for "m-dependent", the value for m.
weights	A vector of weights for the inverse of the scale factor each observation. If an observation is assigned weight 0, it is excluded from the calculations of any parameters. Observations with a NA in any variable will be assigned a weight of 0. Note that weights are defined differently in geem2 and geem, see details.
corr.mat	the correlation matrix for "fixed". Matrix should be symmetric with dimensions >= the maximum cluster size. If the correlation structure is "userdefined", then this is a matrix describing which correlations are the same. In that case, all entries have to be integers, and values less or equal zero indicate a correlation of zero. The information regarding the user-defined structure are extracted from the upper triangle of the provided matrix.
init.beta	an optional vector with the initial values of beta. If not specified, then the intercept will be set to InvLink(mean(response)). init.beta must be specified if not using an intercept.
init.alpha	an optional scalar or vector giving the initial values for the correlation. If provided along with Mv>1 or unstructured correlation, then the user must ensure that the vector is of the appropriate length.
init.phi	an optional initial scale parameter. If not supplied, initialized to 1.
scale.fix	if set to TRUE, then the scale parameter is fixed at the value of init.phi. See details.
nodummy	if set to TRUE, then dummy rows will not be added based on the values in waves.
sandwich	if TRUE, calculate robust variance.
useP	if set to FALSE, do not use the n-p correction for dispersion and correlation estimates, as in Liang and Zeger. This can be useful when the number of observations is small, as subtracting p may yield correlations greater than 1.
maxit	maximum number of iterations.
tol	tolerance in calculation of coefficients.
restriction	either a contrast matrix or a list of a contrast matrix and a right hand side vector, defining a restriction on the regression coefficients. See details.
conv.criterion	convergence criterion, either "ratio" or "difference". The default is "ratio", using the relative change in regression coefficient estimates as convergence criterion, like in geem. With "difference" the maximum absolute difference in regression coefficient estimates is used. The latter is required if some coefficient is 0, e.g. by estimation under some restriction.

Details

The function is a modification of geem from the geeM package, such that additional output is returned that is required for the calculation of covariance matrix across multiple marginal models. In particular the contributions of each subject to the estimating equation are made available in the output. Internal functions regarding the calculation of matrix inverses were modified to improve the handling of missing data.

In geem2, weights are defined as scale weights, similar to most othe software. Note that, in contrast, the current version of geem (version 0.10.1) uses residual weights.

The scale parameter phi is used in estimating the residual working correlation parameters and in estimating the model based (naiv) covariance matrix of the regression coefficients. Similar as in most other software, requesting scale.fix=TRUE only has an impact on the latter, while the working correlation is still estimated using an empirical scale factor for the residuals. In contrast, geem uses the fixed scale factor also when estimating the working correlation.

geem2 allows for estimation of regression coefficients under linear restrictions $L\beta = r$, where L is a contrast matrix, β the vector of regression coefficients and r a real valued right hand side vector. Using the argument restriction, L and r can be specified. If only L is specified, r is assumed as null vector. The functionality is in particular required to calculate the generalized score test for linear hypotheses about β. Use conv.criterion="difference" if any regression coefficient is restricted to 0.

Value

A list with class geem2, similar to the output of geem from the geeM package. The additional slot sandwich.args contains components to calculate the sandwich variance estimator for the fitted model and across models if applied in the multiple marginal model framework.

Note

The option to fit a model with linear restrictions concerning the coefficients is implemented to enable the calculation of a generalized score test. It may also be used to obtain estimates of the coefficients under restrictions. The model based and robust variance estimates of the restricted coefficient estimates are found in the slots restricted.naiv.var and restricted.var, respectively. Note that the variance of estimates restricted to a single value is supposed to be zero, however the calculated variance estimate may deviate from zero within machine accuracy.

Author(s)

The geem function was written by Lee McDaniel and Nick Henderson, modifications for geem2 are by Robin Ristl <robin.ristl@meduniwien.ac.at>

References

Lee S. McDaniel, Nicholas C. Henderson, Paul J. Rathouz. Fast pure R implementation of GEE: application of the matrix package. The R journal 5.1 (2013): 181.

See Also

mmmgee, geem, mmmgee.test

Examples

```
data(keratosis)
m1<-geem2(clearance~trt,id=id,data=keratosis,family=binomial,corstr="independence")
summary(m1)
m2<-geem2(pain~trt,id=id,data=keratosis[keratosis$lesion==1,],family=gaussian,corstr="independence")
summary(m2)
geem2(pain~trt,id=id,data=keratosis[keratosis$lesion==1,],family=gaussian,corstr="exchangeable")
#
data(datasim)
mod1<-geem2(Y.lin~gr.lang+x1,id=id,data=datasim,family="gaussian",corstr="exchangeable")
summary(mod1)
mod2<-geem2(Y.poi~gr.lang+x2,id=id,data=datasim,family="poisson",corstr="unstructured")
summary(mod2)
mod3<-geem2(Y.bin~gr.lang+x3,id=id,data=datasim,family="binomial",corstr="user",
 corr.mat=matrix(c(1,2,3,0, 2,1,2,3, 3,2,1,2, 0,3,2,1),4,4))
summary(mod3)
```

---

keratosis   *Simulated Data Set for a Study of Actinic Keratosis Treatments*

Description

A data set simulated under the planning assumptions for a study comparing four radiation regimens for a photodynamic treatment of actinic keratosis. Each patient receives each treatment in a different skin patch and each patch contains four lesions. Variables are patient identifier (id), treatment (trt), lesion identifier within a patient (lesion), the binary outcome clearance success (1=success, 0=no success) reported for each lesion and the metric outcome pain (larger values indicating more pain) reported for each skin patch. The aim of the study is to compare the treatments B, C and D to the reference treatment A in terms of both outcomes.

Usage

```
data(keratosis)
```

Format

A data frame.

Examples

```
data(keratosis)
head(keratosis)
```

---

mmmgee   *Covariance Matrix Estimation for Multiple Marginal GEE Models*

Description

Calculate the covariance matrix for a stacked vector of regression coefficients from multiple marginal GEE models fitted with [geem2](#).

Usage

```
mmmgee(x, biascorr = FALSE)
```

Arguments

x	a list of geem objects fitted with geem2. The geem objects must be different models calculated with data from the same subjects. In particular, the parameter id in the call to geem2 must refer to the same subjects in each model.
biascorr	logical, if TRUE, a bias corrected covariance matrix is calculate by extending the method due to Mancl and DeRouen to multiple models. See references.

Value

A list with class mmmgee containing the following components:

beta The stacked vector of regression coefficient estimates from the models in x.

V The estimated covariance matrix of the regression coefficient estimates.

A The outer component of $V = ABA$.

B The inner component of $V = ABA$.

biascorr The value of the input argument biascorr (logical).

n A vector with the number of clusters in each model in x.

p A vector with number of regression coefficients in each model in x.

Author(s)

Robin Ristl, <robin.ristl@meduniwien.ac.at>

References

Lloyd A. Mancl, Timothy A. DeRouen. A covariance estimator for GEE with improved small sample properties. Biometrics, 2001, 57(1):126-134.

See Also

[geem2](#), [mmmgee.test](#)

Examples

```
data(keratosis)
m1<-geem2(clearance~trt,id=id,data=keratosis,family=binomial,corstr="independence")
m2<-geem2(pain~trt,id=id,data=keratosis[keratosis$lesion==1,],family=gaussian,corstr="independence")
mmmgee(x=list(m1,m2),biascorr=TRUE)
```

---

mmmgee.test	*Hypothesis Tests for Linear Contrasts in Multiple Marginal GEE Models*

Description

Global hypothesis tests, multiple testing procedures and simultaneous confidence intervals for multiple linear contrasts of regression coefficients in a single generalized estimating equation (GEE) model or across multiple GEE models.

Usage

```
mmmgee.test(x, L, r = NULL, statistic = c("wald", "score"),
  type = c("maximum", "quadratic"), asymptotic = TRUE,
  biascorr = FALSE, closed.test = FALSE, conf.int = FALSE,
  conf.level = 0.95, alternative = c("undirected", "greater", "less"),
  denomDF = NULL, scaled.F = FALSE, maxit = 20, tol = 10^(-8), ...)
```

Arguments

x	a geem2 object fitted with geem2 or a list of geem2. In the latter case, the geem2 objects must be different models calculated with data from the same subjects. In particular, the parameter id in the call to geem2 must refer to the same subjects in each model.
L	a contrast matrix defining a contrast for the stacked vector of regression coefficients of the marginal models, or a list of contrast matrices. In the latter case, the list must contain one matrix for each model listed in x, in the same order as the models. When using the the score test and x is a list, L must be a list.
r	right hand side vector of the null hypothesis or a list of vectors resembling the right hand side of the null hypothesis. If not specified, r is assumed to be a null vector of appropriate length. See details.
statistic	either "wald" or "score", see details. The default is "wald".
type	either "maximum" or "quadratic", see details. The default is "maximum".
asymptotic	logical, if TRUE the reference distribution for the maximum-type Wald test statistic is a multivariate normal distribution and the reference distribution for the quadratic form Wald test statistic is a chi-squared distribution. If FALSE, a multivariate t-distribution or an F-distribution is used instead. Ignored for the Score test, see details.

biascorr	logical indicating whether the Mancl and DeRouen Bias correction should be used when estimating the joint covariance matrix via [mmmgee]().
closed.test	logical, if TRUE, multiplicity adjusted p-values based on a closed test procedure using the selected type of test are calculated. With k hypotheses this involves the computation of 2^k tests, which may require considerable computation time.
conf.int	logical. If TRUE simultaneous confidence intervals corresponding to a single step maximum-type test are calculated using a multivariate normal or t approximation, depending on asymptotic.
conf.level	the nominal simultaneous coverage probability of the confidence intervals.
alternative	one of "undirected", "greater", or "less". Determines the direction of maximum-type tests and of confidence intervals. The default is "undirected".
denomDF	Defaults to NULL. In that case, denominator degrees of freedom for the multavariate t-distribution or F-distribution are calculated as min(n-p), where n and p are vectors of the number of independent clusters and the number of regression coefficients in the models in x. Alternatively, a numeric value may be entered to be used as denominator degrees of freedom.
scaled.F	logical. If TRUE and type="quadratic" and asymptotic=FALSE a scaled F distribution similar as for Hotelling's test is used. Ignored otherwise.
maxit	maximal number of iterations to be passed to [geem2](). Only required when using the score test, where the models are refitted under the restriction of the null hyptothesis.
tol	tolerance limit for the convergence criterion to be passed to [geem2](). Only required when using the score test, where the models are refitted under the restriction of the null hyptothesis.
...	additional arguments that are passed to [pmvnorm](), [qmvnorm](), [pmvt]() and [qmvt](). In particular the algorithm to solve the multivariate normal or t-distribution integrals may be selected.

Details

The null hypothesis is $H0 : L\beta = r$ where L is a contrast matrix, β the stacked vector of regression coefficients rom the marginal models and r a real values right hand side vector. L can be specified as matrix or, if it is a block diagonal matrix with each block corresponding to a contrast for one marginal GEE model, as list of the matrices on the diagonal. The right hand side r can be speficied as vector or as list of vectors each corresponding to the part of the right hand side vector for one model.

When choosing statistic="wald" and type="maximum", the maximum of the standardized entries of $L\hat{\beta}$ is used as test statistic and the p-value is calculated from a multivariate normal or t-distribution (depending on asymptotic being TRUE or FALSE) with correlation matrix estimated for $L\hat{\beta}$. For the t-distribution, denominator degrees of freedom are used as specified in denomDF. When choosing statistic="wald" and type="quadratic", a quadratic form of $L\hat{\beta}$ and the inverse of the estimated covariance matrix of $L\hat{\beta}$ is used as test statistic and the p-value is calculated from a chi-squared distribution or an F-distribution (depending on asymptotic being TRUE or FALSE).

With statistic="score", generalized score tests are calculated by replacing $L\hat{\beta}$ by the first order approximation LAU where U is the stacked estimating equation (the score) and A is the negative

inverse of the matrix of first derivatives of U, both evaluated at the location of constrained estimates for β under the null hypothesis. Analogous to the Wald statistic, a maximum-type and a quadratic form score test are available. For the score test the option asymptotic is ignored and the reference distribution is multivariate normal or chi-squared.

Value

A list with class mmmgeetest containing the following components, if required:

test   Contains a data frame with the test statistic, degrees of freedom (depending in the type of test) and the p-value. If closed test was required, a further data frame is reported with estimates, right hand side vector, unadjusted p-values and adjusted p-values for each line of $H0 : L\beta - r = 0$.

hypothesis   A list containing the contrast matrix L and the right hand side vector r.

conf.int   The simultaneous confidence intervals.

denomDF   The type and value of the denominator degrees of freedom used in the procedure.

mmmgee   The mmmgee object containing in particular the estimated covariance matrix for the coefficents of the models in x. See mmmgee.

Note

A single value for the denominator degrees of freedom is calculated for the covariance matrix estimate across all contrasts. In the closed testing procedure, this value is used for the degrees of freedom associated with the covariance matrix of any subset of contrasts.

Usual linear models or generalized linear models can be regarded as special case of GEE models and can be included in the analysis framework. Note however that mmmgee.test always uses the robust sandwich covariance matrix estimate, even if the calculation of the sandwich covariance was suppressed in the model objects in x.

Author(s)

Robin Ristl, <robin.ristl@meduniwien.ac.at>

References

Dennis D. Boos. On generalized score tests. The American Statistician, 1992, 46(4):327-333.

Lloyd A. Mancl, Timothy A. DeRouen. A covariance estimator for GEE with improved small sample properties. Biometrics, 2001, 57(1):126-134.

See Also

geem2, mmmgee

Examples

```
data(keratosis)
m1<-geem2(clearance~trt,id=id,data=keratosis,family=binomial,corstr="independence")
m2<-geem2(pain~trt,id=id,data=keratosis[keratosis$lesion==1,],family=gaussian,corstr="independence")
L1<-L2<-diag(1,4)[-1,]
mmmgee.test(x=m1,L=list(L1),statistic="wald",type="maximum")
mmmgee.test(x=m1,L=list(L1),statistic="score",type="maximum")
mmmgee.test(x=list(m1,m2),L=list(L1,L2),type="maximum",asymptotic=FALSE,biascorr=TRUE)
mmmgee.test(x=list(m1,m2),L=list(L1,L2),type="maximum",closed.test=TRUE)
mmmgee.test(x=list(m1,m2),L=list(L1,L2),type="maximum",asymptotic=FALSE,
 alternative="less",conf.int=TRUE,denomDF=40)
mmmgee.test(x=list(m1,m2),L=list(L1,L2),type="quadratic",asymptotic=TRUE)
mmmgee.test(x=list(m1,m2),L=list(L1,L2),statistic="score",type="quadratic")
mmmgee.test(x=list(m1,m2),L=list(L1,L2),statistic="score",type="maximum")
#
## Not run:
data(datasim)
mod1<-geem2(Y.lin~gr.lang+x1,id=id,data=datasim,family="gaussian",corstr="exchangeable")
mod2<-geem2(Y.poi~gr.lang+x2,id=id,data=datasim,family="poisson",corstr="exchangeable")
mod3<-geem2(Y.bin~gr.lang+x3,id=id,data=datasim,family="binomial",corstr="exchangeable")
Li<-matrix(c(0,1,0),nrow=1)
mmmgee.test(list(mod1,mod2,mod3),L=list(Li,Li,Li),statistic="Wald",type="maximum",
 biascorr=TRUE,asymptotic=FALSE,closed.test=TRUE)
mmmgee.test(list(mod1,mod2,mod3),L=list(Li,Li,Li),statistic="score",closed.test=TRUE)

## End(Not run)
```

Index