

Package ‘gee4’

March 15, 2017

Type Package

Title Generalised Estimating Equations (GEE/WGEE) using 'Armadillo' and S4

Version 0.1.0.0

Maintainer Yi Pan <ypan1988@gmail.com>

Description Fit joint mean-covariance models for longitudinal data within the framework of (weighted) generalised estimating equations (GEE/WGEE). The models and their components are represented using S4 classes and methods. The core computational algorithms are implemented using the 'Armadillo' C++ library for numerical linear algebra and 'RcppArmadillo' glue.

License GPL (>= 2)

LazyData TRUE

SystemRequirements C++11

Depends R (>= 3.2.2)

Imports Formula, methods, Rcpp (>= 0.12.4)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 6.0.1

NeedsCompilation yes

Author Jianxin Pan [aut],
Yi Pan [aut, cre]

URL <https://github.com/ypan1988/gee4/>

BugReports <https://github.com/ypan1988/gee4/issues/>

Repository CRAN

Date/Publication 2017-03-15 08:49:09

R topics documented:

aids	2
bootcurve	3

CalculateIPWcumprob	3
CalculateIPWprob	4
cattle	4
fittedcurve	5
geer	6
geerControl	7
geerMod-class	7
gees_estimation	8
GenerateCattleMAR	8
getGEER	9
ipw_estimation	10
modular	11
panss	12
show,geerMod-method	13
Index	14

aids

Aids Data

Description

The aids dataset comprises a total of 2376 CD4+ cell counts for 369 HIV infected men with a follow up period of approximately eight and half year. The number of measurements for each individual varies from 1 to 12 and the times are not equally spaced. The CD4+ cell data are highly unbalanced.

Usage

```
data(aids)
```

Format

A data frame with 2376 rows and 8 variables

Details

- id: subject id
- time: measurement time
- cd4: CD4+ cell count

bootcurve	<i>Plot Fitted Curves and Corresponding Confidence Interval using bootstrapping method</i>
-----------	--

Description

Plot fitted curves and corresponding 95% confidence interval using bootstrapping method.

Usage

```
bootcurve(object, nboot)
```

Arguments

object	a fitted joint mean covariance model of class "geerMod", i.e., typically the result of geer().
nboot	number of the bootstrap replications.

Examples

```
## Not run:
# It may take hours for large bootstrap replications
fitgee.ar1 <- geer(cd4|id|time ~ 1|1, data = aids, triple = c(6,3,3),
  method = 'gee-mcd', corr.struct = 'ar1', rho = 0.5,
  control = geerControl(trace=TRUE))
bootcurve(fitgee.ar1, nboot = 1000)

## End(Not run)
```

CalculateIPWcumprob	<i>Calculate the Cumulative Remaining Probabilities in Inverse Probability Weights</i>
---------------------	--

Description

Calculate a vector of cumulative remaining probabilities in inverse probability weights for one specific or all subjects.

Usage

```
CalculateIPWcumprob(m, Y, order, alpha, sub.num = 0)
```

Arguments

m	an integer vector of number of measurements for each subject.
Y	a vector of responses for all subjects.
order	the order for MAR remaining model.
alpha	parameters in MAR remaining model.
sub.num	subject number (0 = all subjects).

CalculateIPWprob	<i>Calculate the Remaining Probabilities in Inverse Probability Weights</i>
------------------	---

Description

Calculate a vector of remaining probabilities in inverse probability weights for one specific or all subjects

Usage

```
CalculateIPWprob(m, Y, order, alpha, sub.num = 0)
```

Arguments

m	an integer vector of number of measurements for each subject.
Y	a vector of responses for all subjects.
order	the order for MAR remaining model.
alpha	parameters in MAR remaining model.
sub.num	subject number (0 = all subjects).

cattle	<i>Cattle Data</i>
--------	--------------------

Description

Kenward (1987) reported an experiment in which cattle were assigned randomly to two treatment groups A and B, and their body weights were recorded in kilogram. Thirty animals received treatment A and another 30 received treatment B. The animals were weighted 11 times over a 133-day period; the first 10 measurements for each animal were made at two-week intervals and the last measurement was made one week later. Since no observation was missing, it is considered to be a balanced longitudinal dataset.

Usage

```
data(cattle)
```

Format

A data frame with 660 rows and 4 variables

Details

- id: subject id
- day: measurement time
- group: Treatment A or Treatment B
- weight: cattle weight

 fittedcurve

Plot Fitted Curves for One or More geerMod Objects

Description

Plot fitted curves and corresponding 95% confidence interval for one or more geerMod objects

Usage

```
fittedcurve(object, text = "fitted curve", ..., include.CI = FALSE)
```

Arguments

object	a fitted joint GEE-MCD/WGEE-MCD model of class "geerMod", i.e., typically the result of geer().
text	some corresponding descriptions for the objects.
...	additional pairs of 'object' and 'text'
include.CI	whether or not 95 for the first object should be plotted.

Examples

```
fitgee.ar1 <- geer(cd4|id|time ~ 1|1, data = aids, triple =
  c(6,3,3), method = 'gee-mcd', corr.struct = 'ar1', rho = 0.5)
fittedcurve(fitgee.ar1, text = "GEE-MCD fitted curve", include.CI = TRUE)
```

Description

Fit a modified Cholesky decomposition (MCD) based joint mean covariance model to longitudinal data within the framework of (weighted) generalised estimating equations (GEE/WGEE), via the Newton-Raphson method.

Usage

```
geer(formula, data = NULL, triple = c(3, 3, 3), method = c("gee-mcd",
  "wgee-mcd"), corr.struct = c("id", "cs", "ar1"), rho = 0.5,
  ipw.order = 1, weights.vec = NULL, control = geerControl(),
  start = NULL)
```

Arguments

formula	a two-sided linear formula object describing the covariates for both the mean and covariance matrix part of the model, with the response, the corresponding subject id and measurement time on the left of a operator~, divided by vertical bars (" ").
data	a data frame containing the variables named in formula.
triple	an integer vector of length three containing the degrees of the three polynomial functions for the mean structure, the log innovation -variances and the autoregressive coefficients.
method	choose 'gee-mcd' (Ye and Pan, 2006) or 'wgee-mcd' (Pan et al. 2012).
corr.struct	choose 'id' (independent), 'cs' (compound symmetry) or 'ar1' (AR(1)).
rho	a parameter used in the 'working' covariance structure.
ipw.order	the order for MAR remaining model.
weights.vec	a user specified vector for the weights H in WGEE-MCD.
control	a list (of correct class, resulting from geerControl()) containing control parameters, see the *geerControl documentation for details.
start	starting values for the parameters in the model.

Examples

```
fitgee.normal <- geer(cd4 | id | time ~ 1 | 1, data = aids, triple
  = c(6,3,3), method = 'gee-mcd', corr.struct = 'id', control =
  geerControl(trace=TRUE))
```

geerControl *Control of GEE-MCD/WGEE-MCD Model Fitting*

Description

Construct control structures for GEE-MCD/WGEE-MCD model fitting

Usage

```
geerControl(trace = FALSE, profile = TRUE, ignore.const.term = TRUE,
            errorMsg = FALSE, use.weights.vec = FALSE)
```

Arguments

trace	whether or not the value of the objective function and the parameters should be print on every trace'th iteration.
profile	whether or not parameters should be estimated sequentially using the idea of profile likelihood.
ignore.const.term	whether or not the constant term should be considered when calculating quasi-likelihood and QIC.
errorMsg	whether or not the error message should be print.
use.weights.vec	whether or not a user-specified weights.vec should be used

geerMod-class *Class "geerMod" of Fitted GEE-MCD/WGEE-MCD Models.*

Description

Class "geerMod" of Fitted GEE-MCD/WGEE-MCD Models.

Slots

call	the matched call
opt	the optimization result returned by optimizeGeer
args	arguments m, Y, X, Z, W, H, time
triple	an integer vector of length three containing the degrees of the three polynomial functions for the mean structure, the log innovation -variances and the autoregressive coefficients.
rho	a parameter used in the 'working' covariance structure.
devcomp	the deviance components list.

gees_estimation *Fit (Weighted) Generalized Estimating Equations based on MCD*

Description

Fit (weighted) generalized estimating equations based on MCD.

Usage

```
gees_estimation(m, Y, X, Z, W, H, method, corrStruct, rho, start,
  trace = FALSE, profile = TRUE, errorMsg = FALSE)
```

Arguments

m	an integer vector of number of measurements for each subject.
Y	a vector of responses for all subjects.
X	model matrix for mean structure model.
Z	model matrix for the diagonal matrix.
W	model matrix for the lower triangular matrix.
H	a vector of weights used in WGEE-MCD.
method	choose 'gee-mcd' (Ye and Pan, 2006) or 'wgee-mcd' (Pan et al. 2012).
corrStruct	choose 'id' (independent), 'cs' (compound symmetry) or ar1' (AR(1)).
rho	a parameter used in the 'working' covariance structure.
start	starting values for the parameters in the model.
trace	the values of the objective function and the parameters are printed for all the trace'th iterations.
profile	whether parameters should be estimated sequentially using the idea of profile likelihood or not.
errorMsg	whether or not the error message should be print.

GenerateCattleMAR *Generate Cattle B Data with Pre-specified Dropout Rate*

Description

We assume that all cattle have the first four observations and there is a certain chance that some cows will quit the study (dropout) at the fifth measurement if their weights are below a certain threshold. The threshold value for the weight is chosen so that some fixed rates of MAR dropout at the fifth measurement time.

Usage

```
GenerateCattleMAR(dropout.rate)
```

Arguments

dropout.rate the dropout rate at the fifth measurement.

getGEER	<i>Extract or Get Generalized Components from a Fitted GEE-MCD/WGEE-MCD Model</i>
---------	---

Description

Extract (or get) "components" - in a generalized sense - from a fitted joint mean covariance model from an object of class "geerMod".

Usage

```
getGEER(object, name, sub.num)
```

```
## S3 method for class 'geerMod'
getGEER(object, name = c("m", "Y", "X", "Z", "W", "H", "D",
  "T", "Sigma", "mu", "theta", "beta", "lambda", "gamma", "alpha", "sd", "FIM",
  "quasilik", "BIC", "iter", "triple", "pij", "cpij"), sub.num = 0)
```

Arguments

object	a fitted joint mean covariance model of class "geerMod", i.e., typically the result of geer().
name	a character vector specifying the name(s) of the "component". When sub.num is not specified or equal to 0, possible values are: "m" a vector of number of measurement for each subject "Y" response vector "X" model matrix for mean structure "Z" model matrix for covariance structure (the diagonal matrix) "W" model matrix for covariance structure (the lower triangular matrix) "H" a vector of weights used in WGEE-MCD "FIM" Fisher Information matrix "theta" parameter estimates of GEE-MCD/WGEE-MCD model "sd" standard deviations for parameter estimates of GEE-MCD/WGEE-MCD model "beta" parameter estimates for mean structure model "lambda" parameter estimates for covariace structure (the diagonal matrix) "gamma" parameter estimates for covariance structure (the lower triangular matrix)

"quasilik" quasi-likelihood
 "QIC" Quasi information criterion
 "iter" number of iterations until convergence
 "triple" (p, d, q)
 "pij" a vector of remaining probability
 "cpij" a vector of cumulative remaining probability
 When sub.num is specified, possible values are:
 "m" number of measurements for subject i
 "Y" response vector for subject i
 "X" model matrix of subject i for mean structure
 "Z" model matrix of subject i for covariance structure (the diagonal matrix)
 "W" model matrix of subject i for covariance structure (the lower triangular matrix)
 "D" the estimated diagonal matrix for subject i
 "T" the estimated lower triangular matrix for subject i
 "Sigma" the estimated covariance matrix for subject i
 "mu" the estimated mean for subject i
 "pij" a vector of remaining probability for subject i
 "cpij" a vector of cumulative remaining probability for subject i
 sub.num refer to i's subject

Methods (by class)

- geerMod: Extract or Get Generalized Components from a Fitted GEE-MCD/WGEE-MCD Model

Examples

```
fitgee.ar1 <- geer(cd4|id|time ~ 1|1, data = aids, triple =
  c(6,3,3), method = 'gee-mcd', corr.struct = 'ar1', rho = 0.5, control =
  geerControl(trace=TRUE))
```

```
sd <- getGEER(fitgee.ar1, "sd")
QIC <- getGEER(fitgee.ar1, "QIC")
Di <- getGEER(fitgee.ar1, "D", 10)
```

ipw_estimation

Fit Inverse Probability Weights Model

Description

Fit inverse probability weights model.

Usage

```
ipw_estimation(m, Y, order, trace = FALSE)
```

Arguments

m	an integer vector of number of measurements for each subject.
Y	a vector of responses for all subjects.
order	the order for MAR remaining model.
trace	whether or not the optimization iteration should be printed.

 modular

Modular Functions for GEE-MCD and WGEE-MCD Fits

Description

Modular Functions for a modified Cholesky decomposition (MCD) based (weighted) generalised estimating equations (GEE/WGEE) fits

Usage

```
ldFormula(formula, data = NULL, triple = c(3, 3, 3), method = c("gee-mcd",
  "wgee-mcd"), corr.struct = c("id", "cs", "ar1"), rho = 0.5,
  ipw.order = 1, weights.vec = NULL, control = geerControl(),
  start = NULL)
```

```
optimizeGeer(m, Y, X, Z, W, H, time, method, corr.struct, rho, ipw.order,
  control, start)
```

```
mkGeerMod(opt, args, triple, rho, corr.struct, mc)
```

Arguments

formula	a two-sided linear formula object describing the covariates for both the mean and covariance matrix part of the model, with the response, the corresponding subject id and measurement time on the left of a operator~, divided by vertical bars (" ").
data	a data frame containing the variables named in formula.
triple	an integer vector of length three containing the degrees of the three polynomial functions for the mean structure, the log innovation -variances and the autoregressive coefficients.
method	choose 'gee-mcd' (Ye and Pan, 2006) or 'wgee-mcd' (Pan et al. 2012).
corr.struct	choose 'id' (independent), 'cs' (compound symmetry) or 'ar1' (AR(1)).
rho	a parameter used in the 'working' covariance structure.
ipw.order	the order for MAR remaining model.

<code>weights.vec</code>	a user specified vector for the weights H in WGEE-MCD.
<code>control</code>	a list (of correct class, resulting from <code>geerControl()</code>) containing control parameters, see the <code>*geerControl</code> documentation for details.
<code>start</code>	starting values for the parameters in the model.
<code>m</code>	an integer vector of number of measurements for each subject.
<code>Y</code>	a vector of responses for all subjects.
<code>X</code>	model matrix for mean structure model.
<code>Z</code>	model matrix for the diagonal matrix.
<code>W</code>	model matrix for the lower triangular matrix.
<code>H</code>	a vector of weights used in WGEE-MCD.
<code>time</code>	a vector of time from the data.
<code>opt</code>	optimized results returned by <code>optimizeGeer</code> .
<code>args</code>	arguments returned by <code>ldFormula</code> .
<code>mc</code>	matched call from the calling function.

panss

PANSS Data

Description

The PANSS or the Positive and Negative Syndrome Scale is a medical scale used for measuring symptom severity of patients with schizophrenic conditions. `panss` contains data from a longitudinal study where 3 different treatments were considered. Patients were followed for 8 weeks and PANSS score was recorded on week 0, 1, 2, 4, 6 and 8. The lower PANSS score a patient has, the less symptoms. Data was extracted from a larger, and confidential, set of clinical trial data from a randomised clinical trial.

Usage

`data(panss)`

Format

A data frame with 685 rows and 4 variables

Details

- `treat`: a factor variable with 3 levels
- `time`: measurement time
- `Y`: PANSS score
- `id`: subject id

show,geerMod-method *Print information for geerMod-class*

Description

Print information for geerMod-class

Usage

```
## S4 method for signature 'geerMod'  
show(object)
```

Arguments

object a fitted GEE-MCD/WGEE-MCD model of class "geerMod", i.e., typically the result of `geer()`.

Index

*Topic **datasets**

- aids, [2](#)
- cattle, [4](#)
- panss, [12](#)

aids, [2](#)

bootcurve, [3](#)

CalculateIPWcumprob, [3](#)

CalculateIPWprob, [4](#)

cattle, [4](#)

fittedcurve, [5](#)

geer, [6](#)

geerControl, [7](#)

geerMod-class, [7](#)

gees_estimation, [8](#)

GenerateCattleMAR, [8](#)

getGEER, [9](#)

ipw_estimation, [10](#)

ldFormula (modular), [11](#)

mkGeerMod (modular), [11](#)

modular, [11](#)

optimizeGeer (modular), [11](#)

panss, [12](#)

show,geerMod-method, [13](#)