Package 'drgee'

July 2, 2014

Type Package

Title Doubly Robust Generalized Estimating Equations

Version 1.0.1

Date 2014-01-27

Author Johan Zetterqvist <johan.zetterqvist@ki.se> , Arvid Sjölander <arvid.sjolander@ki.se> with contributions from Alexander Ploner.

Maintainer Johan Zetterqvist <johan.zetterqvist@ki.se>

Description The package provides functions to estimate parameters quantifying an association between an exposure and an outcome in a generalized estimating equations model in the presence of nuisance variables. Three methods are implemented: outcome nuisance model based estimation where a model for the association between the nuisance factors and the outcome is assumed, exposure nuisance model based estimation where a model for the association between the nuisance factors and the exposure is assumed and doubly robust estimation where both models are used. In doubly robust estimation, the estimates will be consistent when at least one of the models are correctly specified, not necessarily both.

Depends R (>= 3.0.0)

Imports stats, nleqslv

License GPL-2 | GPL-3

NeedsCompilation no

Encoding UTF-8

Repository CRAN

Date/Publication 2014-01-27 15:31:41

R topics documented:

`drgee-package` *Doubly Robust Generalized Estimating Equations*

Description

The main function is `drgee`. For convenience, the help functions are supplied. `drgeeData` extracts variables in a format needed in estimation. `obeFit`, `ebeFit` and `drFit` performs outcome nuisance model based estimation, exposure nuisance model based estimation and doubly robust estimation respectively. `robVcov` calculates robust variance of the estimates.

Author(s)

Johan Zetterqvist, Arvid Sjölander with contributions from Alexander Ploner.

`drFit` *Doubly Robust Generalized Estimating Equations*

Description

Doubly robust estimation given a `drgeeData` object.

Usage

```
drFit(object, rootFinder=findRoots, ...)
```

Arguments

`object`	A `drgeeData` object.
`rootFinder`	A function used to solve a system of non linear equations. Default is `findRoots`.
`...`	Further arguments to be passed to `geeFit` and `rootFinder`.

Details

drFit performs doubly robust estimation (as described in the man page for drgee) given a drgeeData object.

The models are specified by variables y, a, x, yx, ax, v, z and as character variables olink and elink which are assumed to be found in the input argument object as a drgeeData object.

The rootFinder argument is not needed when object$olink is "identity". Any user supplied function to the argument rootFinder is expected to have the same input and output as findRoots.

Robust variance is calculated by robVcov.

drFit is meant to be called by drgee.

Value

A list containing:

coefficients	The estimated parameters in the main model.
vcov	The estimated covariance matrix.
optim.object	The optimization object returned from the function rootFinder, if called.

Author(s)

Johan Zetterqvist, Arvid Sjölander

See Also

drgee, drgeeData, findRoots and robVcov.

---

drgee *Doubly Robust Generalized Estimating Equations*

Description

drgee performs outcome nuisance model based estimation, exposure nuisance model based estimation or doubly robust estimation given symbolic representations of an outcome nuisance model and an exposure nuisance model.

Usage

```
drgee(oformula, eformula, iaformula = formula(~1),
      olink = c("identity", "log", "logit"),
      elink = c("identity", "log", "logit"),
      estimationMethod = c("dr", "obe", "ebe"),
      data = NULL, rootFinder = findRoots,
      clusterid = NULL, ...)
```

Arguments

oformula	An expression or formula for the outcome nuisance model. The outcome is identified as the response in this formula. Therefore, a LHS is required for all choices for estimationMethod.
eformula	An expression or formula for the exposure nuisance model. The exposure is identified as the response in this formula. Therefore, a LHS is required for all choices for estimationMethod.
iaformula	An expression or formula where the RHS should contain the variables that "interact" (i.e. are supposed to be multiplied with) with the exposure in the main model. "1" will always added. Default value is no interactions, i.e. iaformula=formula(~1).
olink	A character string naming the link function in the outcome nuisance model. Has to be "identity", "log" or "logit". Default is "identity".
elink	A character string naming the link function in the exposure nuisance model. Has to be "identity", "log" or "logit". Default is "identity".
estimationMethod	A character string naming the desired estimation method. Choose "obe" for outcome nuisance model based estimation, "ebe" for exposure nuisance model based estimation or "dr" for doubly robust estimation. Default is "dr".
data	A data frame or environment containing the variables appearing in iaformula, oformula and eformula. Default is NULL in which case data are expected to be found in the environment of the oformula argument.
rootFinder	A function to solve a system of non linear equations. Default is findRoots.
clusterid	A optional character string naming a cluster-defining variable in the data argument.
...	Further arguments to be passed to the functions geeFit and rootFinder.

Details

drgee estimates the parameter β in a main model $g\{E(Y|A,L)\} = \beta^T\{A \cdot X(L)\} + Q(L)$, where L is a vector of nuisance variables and $X(L)$ and $Q(L)$ are functions of L. Note that $A \cdot X(L)$ should be interpreted as a columnwise multiplication and that $X(L)$ will always contain a column of 1's. Given a specification of an outcome nuisance model $Q(L) = \gamma^T V(L)$ (where $V(L)$ is a function of L) outcome nuisance model based estimation can be performed. Alternatively, leaving $Q(L)$ unspecified and using an exposure nuisance model $h\{E(A|L)\} = \alpha^T Z(L)$ (where h is a link function and $Z(L)$ is a function of L) exposure nuisance model based estimation can be performed. When g is $logit$, the exposure nuisance model is required be of the form $logit\{E(A|Y=0,L)\} = \alpha^T Z(L)$. In this case the exposure needs to binary. Given both an outcome and an exposure nuisance model, doubly robust estimation can be performed. Doubly robust estimation gives a consistent estimate of the parameter β when either the outcome nuisance model or the exposure nuisance model is correctly specified, not necessarily both.

Usage is best explained through an example. Suppose that we are interested in the parameter vector β_0 and β_1 in a main model $logit\{E(Y|A,L_1,L_2)\} = \beta_0 A + \beta_1 A \cdot L_1 + Q(L_1,L_2)$ where L_1 and L_2 are nuisance variables and $Q(L_1,L_2)$ is some (unspecified) function of L_1 and L_2.

To adjust for L_1 and L_2, we can use an outcome nuisance model $Q(L_1,L_2;\gamma) = \gamma_0 + \gamma_1 L_1$ or an exposure nuisance model $logit\{E(A|Y=0,L_1,L_2)\} = \alpha_0 + \alpha_1 L1 + \alpha_2 L2$ to calculate estimates of β_0 and β_1 in the main model.

We specify the outcome nuisance model as `oformula=Y~L_1` and `olink="logit"`. The exposure nuisance model is specified as `eformula=A~L_1+L_2` and `elink="logit"`. Since the outcome Y and the exposure A are identified as the LHS of `oformula` and `eformla` respectively and since the outcome link is specified in the `olink` argument, the only thing left to specify for the main model is the (multiplicative) interactions $X(L) = (1, L_1)^T$. This is done as `iaformula=~L_1`, since 1 is always included in $X(L)$. We can then perform outcome or exposure nuisance model based estimation or doubly robust estimation by setting `estimationMethod` to `"obe"`, `"ebe"` or `"dr"` respectively.

When `estimationMethod="obe"`, the RHS of `eformula` will be ignored with a warning message.

When `estimationMethod="ebe"`, the RHS of `oformula` will be ignored with a warning message.

Outcome nuisance model based estimation is implemented for generalized estimating equation models with the identity, log or logit link and independent observations. The estimated coefficients are identical to those obtained with `glm`, but since no distributional assumptions are made, robust variance is calculated.

When exposure nuisance model based estimation or doubly robust estimation estimation is chosen with `olink="logit"` the exposure link will be changed to `"logit"` with a warning message.

Robust variance for the estimated parameter is calculated using `robVcov`. A cluster robust variance is calculated when a character string naming a cluster variable is supplied in the `clusterid` argument.

`drgee` calls `geeData` to create a `geeData` object containing the elements needed in the calculations. The estimation of the coefficents in the main model is performed by `obeFit`, `ebeFit` or `drFit`.

For exposure nuisance model based estimation when g is the identity or log link, see Robins et al. (1992).

For doubly robust estimation when g is the identity or log link, see Robins (1999). For doubly robust estimation when g is the logit link, see Tchetgen et al. (2010).

This package was inspired by the STATA package `drglm` described in Orsini et al. (2013). Basically, it also provides the same functionality.

Value

`drgee` return an object of class `drgee` containing:

`coefficients`	Estimates of the parameters in the main model.
`vcov`	Robust variance of the parameter estimates.
`optim.object`	An estimation object returned from the function specified in the `rootFinder`, if this function is called.
`call`	The matched call.
`geeData`	The `geeData` object used in the calculations.
`estimationMethod`	The value of the input argument `estimationMethod`.

The class methods `coef` and `vcov` can be used to extract the estimated parameters and their covariance matrix from a `drgee` object. `summary.drgee` produces a summary of the calculations.

Author(s)

Johan Zetterqvist, Arvid Sjölander

References

Orsini N., Belocco R., Sjölander A. (2013), Doubly Robust Estimation in Generalized Linear Models, *Stata Journal*, **13**, 1, pp.185-205

Robins J.M., Mark S.D., Newey W.K. (1992), Estimating Exposure Effects by Modelling the Expectation of Exposure Conditional on Confounders, *Biometrics*, **48**, 479–495

Robins JM (1999), Robust Estimation in Sequentially Ignorable Missing Data and Causal Inference Models, *Proceedings of the American Statistical Association Section on Bayesian Statistical Science*, pp. 6–10

Tchetgen E.J.T., Robins J.M., Rotnitzky A. (2010), On Doubly Robust Estimation in a Semiparametric Odds Ratio Model, *Biometrika*, **97**, 1, 171–180

See Also

`obeFit` for outcome nuisance model based estimation, `ebeFit` for exposure nuisance model based estimation, `drFit` for doubly robust estimation of the parameters in the main model, `drgeeData` for data preparation and `findRoots` for nonlinear equation solving, `robVcov` for estimation of variance.

Examples

```
## Doubly robust estimation when
## the main model is
## E(Y|A,L1,L2)-E(Y|A=0,L1,L2)=beta0*A+beta1*A*L1
## and the outcome nuisance model is
## E(Y|A=0,L1,L2)=gamma0+gamma1*L1+gamma2*L2
## and the exposure nuisance model is
## E(A|Y=0,L1,L2)=expit(alpha0+alpha1*L1+alpha2*l2)

library(drgee)

expit<-function(x) exp(x)/(1+exp(x))

n<-5000

# nuisance
l1<-rnorm(n, mean = 0, sd = 1)
l2<-rnorm(n, mean = 0, sd = 1)

beta0<-1.5
beta1<-1
gamma0<--1
gamma1<--2
gamma2<-2
alpha0<-1
alpha1<-5
alpha2<-3
```

```
# Exposure
a<-rbinom(n,1,expit(alpha0 + alpha1*l1 + alpha2*l2))
# Outcome
y<-rnorm(n,beta0*a + beta1*a*l1 + gamma0 + gamma1*l1 + gamma2*l2,sd=1)

data<-data.frame(y,a,l1,l2)

## outcome nuisance model misspecified and
## exposure nuisance model correctly specified

# Doubly robust estimation
dr.est <- drgee(y~l1,a~l1+l2,~l1,"identity","logit","dr",data)
summary(dr.est)

# Outcome nuisance model based estimation
obe.est <- drgee(y~l1,a~1,~l1,"identity","logit","obe",data)
summary(obe.est)

# Exposure based estimation
ebe.est <- drgee(y~1,a~l1+l2,~l1,"identity","logit","ebe",data)
summary(ebe.est)
```

`drgeeData` *Extracting Variables and Model Matrices for Generalized Estimating equations*

Description

Given a main model, an outcome nuisance model and an exposure nuisance model `drgeeData` extracts the model variables and matrices from a `data.frame` or an `environment` object. It also performs some data cleaning and error checking.

Usage

```
drgeeData(oformula, eformula, iaformula = formula(~1),
              olink = c("identity", "log", "logit"),
              elink = c("identity", "log", "logit"),
              data = NULL, clusterid = NULL)
```

Arguments

`oformula`	An expression or formula for the outcome nuisance model. The outcome is identified as the response in this formula.
`eformula`	An expression or formula for the exposure nuisance model. The exposure is identified as the response in this formula.

iaformula	An expression or formula where the RHS should contain the variables that "interact" (i.e. are supposed to be multiplied with) with the exposure in the main model to create the terms associated with the parameters of interest. "1" will always added. Default value is no interactions, i.e. formula(~1).
olink	A character string naming the link function in the outcome nuisance model. Have to be "identity", "log" or "logit". Default is "identity".
elink	A character string naming the link function in the exposure nuisance model. Have to be "identity", "log" or "logit". Default is "identity". When olink="logit" this is replaced by "logit".
data	A data frame or environment containing the variables in iaformula, oformula and eformula. Default is NULL in which case the variables will be expected to be found in the environment of the oformula argument.
clusterid	A optional character string naming a cluster-defining variable in the data argument.

Details

drgeeData is called by drgee and uses the oformula, eformula and iaformula arguments to extract data from a data.frame or environment object. The data can then be used to for outcome/exposure nuisance model based estimation or doubly robust estimation by calling obeFit, ebeFit or drFit respectively. drgeeData uses model.frame and model.matrix to remove incomplete observations and to convert factors to dummy variables. Factor outcome is only allowed when olink="logit" and then only when the factor has two levels. There are no restrictions on the exposure (RHS of eformula) or on factors appearing on the RHS of oformula, eformula and iaformula.

The class method summary.drgeeData produces strings for the formulas with terms referring to the columns in the produced design matrices.

Value

drgeeData returns an object of class drgeeData containing

y	The outcome matrix.
a	The exposure matrix.
x	The matrix of of interactions defined in iaformula. This matrix will always contain a column with 1's.
ax	The matrix of elementwise product(s) of a and each column in x.
v	The matrix of terms in the outcome nuisance model.
z	The matrix of terms in the exposure nuisance model.
yx	The matrix of elementwise product(s) of y and each column in x.
id	A factor defining clusters. For independent observations, the number of levels equals the number of complete observations.
olink	A character string naming the link function in the outcome nuisance model.
elink	A character string naming the link function in the outcome nuisance model.

All matrix elements have rows corresponding to the complete observations in the original data.

Author(s)

Johan Zetterqvist, Arvid Sjölander

See Also

drgee, model.frame and model.matrix.

---

ebeFit *Exposure nuisance model based estimation*

Description

Exposure nuisance model based estimation given a drgeeData object.

Usage

```
ebeFit(object, rootFinder=findRoots, ...)
```

Arguments

object	A drgeeData object.
rootFinder	A function used to solve a system of nonlinear equations. Default is findRoots.
...	Further arguments to be passed to drgeeFit or to the function rootFinder.

Details

ebeFit performs exposure nuisance model based estimation (as described in the man page for drgee) given a drgeeData object.

The models are specified by variables y, a, x, yx, ax, z and as character variables olink and elink which are assumed to be found in the input argument object as a drgeeData object.

The rootFinder argument is only needed when object$olink is "log". The rootFinder argument is not needed when object$olink is "identity". Any user supplied function to rootFinder is expected to have the same input and output as findRoots.

Robust variance is calculated by robVcov.

ebeFit is meant to be called by drgee.

Value

A list containing:

coefficients	The estimated parameters in the main model.
vcov	The estimated covariance matrix.
optim.object	The optimization object returned from the function rootFinder function. Is NULL when object$olink is "logit".

Note

When `object$olink` is `"logit"`, the result is equivalent with logistic regression where the outcome and exposure have switched places in the model and robust variance is used.

Author(s)

Johan Zetterqvist, Arvid Sjölander

See Also

`drgeeData`, `drgee`, `findRoots` and `robVcov`.

`findRoots` *Non Linear Equation System Solving*

Description

A wrapper around `nleqslv` from the **nleqslv** package to solve a non linear system of equations.

Usage

```
findRoots(beta.init, eq.func, d.eq.func = NULL, arg.list, ...)
```

Arguments

`beta.init`	An initial guess for the zero.
`eq.func`	A function of two variables for which the zero are sought. Its first argument `beta` should be a vector over which the zeros are sought and the second argument `arg.list` a list of additional arguments.
`d.eq.func`	A function to return the Jacobian of `eq.func` taking the same arguments as `eq.func`. Supplying this function can speed up calculations. Default is `NULL`.
`arg.list`	The second argument to `eq.func` and `d.eq.func`
`...`	A list of additional arguments to be passed to `nleqslv`

Details

`findRoots` calculates zeros fo the function `eq.func` and is meant to be called from `ebeFit` and `drFit`. It is the default equation solving function in `drgee`. It is supplied as a separate function in order to allow users to use other equation solvers by writing their own wrapper with the same interface as `findRoots`.

Value

The value is a list containing the following arguments:

`root`	The zero(s) of the function `eq.func`.
`optim.object`	The optimization object returned from `nleqslv`.

Author(s)

Johan Zetterqvist, Arvid Sjölander

See Also

`nleqslv` in package **nleqslv**, `ebeFit`, `drFit`.

`geeFit` *Generalized Estimating Equation Solver*

Description

`geeFit` is basically a wrapper around `glm.fit` but also computes the residuals and their derivatives with respect to the estimated coefficients.

Usage

```
geeFit(y, x, link=c("identity","log","logit"), ...)
```

Arguments

`y`	The outcome observations.
`x`	The design matrix.
`link`	A character string naming the link function. Have to be `"identity"`, `"log"` or `"logit"`.
`...`	Further arguments to be passed to `glm.fit`

Details

`geeFit` estimates coefficients θ using an estimating equation $W\{R - k^{-1}(\theta^T W)\} = 0$ where k is the identity, log or logit link function. `glm.fit` is used to estimate the coefficients. The `family` object used in the `glm.fit` call depends on the argument `link`. It will be `gaussian()`, `quasipoisson()` or `binomial()` when `link` is `"identity"`, `"log"` and `"logit"` respectively.

`geeFit` is intended to be used as help function in `drFit`, `ebeFit` and `obeFit`.

Value

A list with the following elements:

`coefficients`	A named vector of the estimated coefficients.
`fitted.values`	A vector of the predicted outcomes evaluated with the estimated coefficients.
`res`	A vector of the residuals, i.e. the difference between the the observed outcome `y` and `fitted.values`.
`d.res`	The derivatives of the residuals with respect to the model parameters evaluated with the estimated coefficients. A matrix with rows corresponding to observations and columns corresponding the the estimated coefficients.

Author(s)

Johan Zetterqvist, Arvid Sjölander

See Also

glm.fit

---

obeFit *Outcome Nuisance Model Based estimation*

Description

Outcome nuisance based estimation given a drgeeData object.

Usage

```
obeFit(object, ...)
```

Arguments

object	A drgeeData object.
...	Further arguments to be passed to geeFit

Details

obeFit performs outcome nuisance model based estimation (as described in the man page for drgee)) given a drgeeData object.

The model is specified by variables y, a, x, ax, v and as a character variable olink which are assumed to be found in the input argument object as a drgeeData object.

Robust variance is calculated by robVcov.

obeFit is meant to be called by drgee.

Value

A list containing:

coefficients	The estimated parameters in the main model.
vcov	The estimated covariance matrix.

Note

When object$olink is "logit", the result is asymptotically equivalent with logistic regression.

Author(s)

Johan Zetterqvist, Arvid Sjölander

See Also

drgeeData, drgee and robVcov.

---

robVcov *Robust Variance Calculation*

Description

robVcov calculates the asymptotic variance for Z-estimators.

Usage

```
robVcov(U, d.U, id=NULL)
```

Arguments

U	A n x q matrix of the estimating equations evaluated at the estimated model parameters, where n is the number of observations and q is the number of estimating equations.
d.U	The mean of the jacobian of U evaluated at the solution to the estimating equations, with rows corresponding to the estimating equations and columns corresponding to the model parameters. The number of model parameters is assumed to equal the number of estimating equations such that d.U is a q x q square matrix.
id	A factor with levels corresponding the clusters in the data. Default is NULL in which case all observations are considered to be independent.

Details

For robust variance estimation, see van der Vaart (1998).

For clustered data, the rows in U are added clusterwise resulting in a cluster robust estimate of the variance.

Value

The estimated covariance matrix.

Author(s)

Johan Zetterqvist, Arvid Sjölander

References

van der Vaart, A.W. (2000), *Asymptotic Statistics*, Cambridge University Press, pp. 52–53.

Index