

Package ‘eatRep’

December 18, 2012

Type Package

Title eatRep

Version 0.0.1

Depends R (>= 2.14.0), eatTools, eatData, survey, reshape

Imports fmsb

Date 2012-11-20

Author Sebastian Weirich, Martin Hecht, Nicole Haag, Karoline Sachse and Thilo Siegle

Maintainer Sebastian Weirich <sebastian.weirich@iqb.hu-berlin.de>

Description compute descriptives for complex survey designs with multiple imputed data

License GPL (>= 2)

LazyLoad yes

LazyData yes

OS_type windows

R topics documented:

jk2.glm	2
jk2.mean	4
jk2.quantile	5
jk2.table	6
Index	9

jk2.glm

*JK2 for linear regression models.***Description**

Compute linear regression models for complex cluster designs with multiple imputed variables based on Jackknife (JK2) procedure. Conceptually, the function combines replication methods and methods for multiple imputed data. Technically, this is a wrapper for the svyglm() function of the survey package.

Usage

```
jk2.glm(dat, ID, wgt = NULL, JKZone, JKrep, group = list(), independent = list(),
        reg.statement = NULL, dependent = list(),
        complete.permutation = c("nothing", "groups", "independent", "all") ,
        glm.family)
```

Arguments

dat	Data frame containing all variables for analysis.
ID	Variable name or column number of ID variable.
wgt	Optional: Variable name or column number of weighting variable. If no weighting variable is specified, all cases will be equally weighted.
JKZone	Variable name or column number of variable indicating Jackknifing Zone.
JKrep	Variable name or column number of variable indicating replicate ID.
group	Optional: List of one or more grouping variables. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
independent	List of one or more independent variables. If independent variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
reg.statement	Optional: By default, the regression formula is created automatically by the independent variables connected by a "+" symbol. However, when interaction terms should be defined, the regression has to be specified by the user. The right side of the regression formula has to be defined in "reg.statement" as a string, whereas the names of the independent variables occurring in the "independent" argument have to be used. This is due to various names of one variable in multiple imputed data sets. See example 3 for further details.
dependent	List of one or more dependent variables. Each dependent variable will result in a separate analysis. If dependent variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
complete.permutation	Argument defines the number of multiple imputed data sets. In general, the number is defined by the number of imputations of a variable. Therefore, this argument only becomes relevant, if the number of imputation of one variable (e.g. the number of plausible values of the dependent variable) differs from the number of imputations used for independent variable(s). If "all", number

of datasets are determined through permutation, e.g. 5 plausible values and 3 imputations of the independent variable results in $3 \times 5 = 15$ imputed data sets. If "nothing", only 5 imputed data sets will be used, no matter whether the larger number of imputations is a whole multiple of the smaller number of imputations. If "groups", only the number of imputations of more than one grouping variable will be permuted. If "independent", only the number of imputations of more than one independent variable will be permuted.

`glm.family` Argument of class "family", specifying the link function. See help file of 'glm' for details.

Details

Function first creates replicate weights based on JKZone and JKrep variables according to JK2 procedure implemented in WesVar. According to multiple imputed data sets, a workbook with several analyses is created. The function afterwards serves as a wrapper for `svyglm()` implemented in the 'survey' package. The results of the several analyses are then pooled according to Rubin's rule.

Value

A list of data frames, one for each dependent variable. Each data frame contains columns for group variables (if specified), regression weights, standard errors and r squared. Each row in the data frame corresponds to one parameter (specified in column "reg") in one group (specified in the column which name refers to the group name) for one dependent variable (specified in the name of the corresponding list element).

Author(s)

Sebastian Weirich

Examples

```
data(reading_writing)
### Example 1: Computes linear regression from reading score on hisei separately for each country
mod1 <- jk2.glm(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country"),
  dependent = list(reading = paste("reading_score", 1:3, sep = "")), writing = paste("writing_score", 1:3, sep = "")),
  independent = list(hisei = paste("zehisei", 1:5, sep = "")),
  complete.permutation = "no", glm.family = gaussian(link="identity") )
### Example 2: Computes log linear regression from pass/fail on hisei and gender separately for each country
mod2 <- jk2.glm(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country"),
  dependent = list(reading.pass = paste("passed_reading", 1:3, sep = "")), writing.pass = paste("passed_writing", 1:3, sep = "")),
  independent = list(hisei = paste("zehisei", 1:5, sep = "")), gender = "sex",
  complete.permutation = "no", glm.family = binomial(link="logit") )
### Example 3: Computes linear regression from pass/fail on hisei and gender (and its' interaction) separately for each country
### The regression statement has to be defined manually. Note that, despite "hisei" occurs as a multiple in the dependent variable,
### one statement has to be defined.
mod3 <- jk2.glm(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country"),
  dependent = list(reading.pass = paste("passed_reading", 1:3, sep = "")), writing.pass = paste("passed_writing", 1:3, sep = "")),
  independent = list(hisei = paste("zehisei", 1:5, sep = "")), gender = "sex",
  reg.statement = "hisei*gender",
  complete.permutation = "no", glm.family = binomial(link="logit") )
```

jk2.mean

*JK2 for mean estimates.***Description**

Compute means, variances and standard deviations with standard errors for complex cluster designs with multiple imputed variables (e.g. plausible values) based on Jackknife (JK2) procedure. Conceptually, the function combines replication methods and methods for multiple imputed data. Technically, this is a wrapper for the svymean() and svyvar() functions of the survey package.

Usage

```
jk2.mean(dat, ID, wgt = NULL, JKZone, JKrep, group = list(), group.differences.by = NULL,
         dependent = list(), complete.permutation = c("nothing", "groups", "all"))
```

Arguments

dat	Data frame containing all variables for analysis.
ID	Variable name or column number of ID variable.
wgt	Optional: Variable name or column number of weighting variable. If no weighting variable is specified, all cases will be equally weighted.
JKZone	Variable name or column number of variable indicating Jackknifing Zone.
JKrep	Variable name or column number of variable indicating replicate ID.
group	Optional: List of one or more grouping variables. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
group.differences.by	Optional: Specifies variable group differences should be computed for.
dependent	List of one or more grouping variables. Each dependent variable will result in a separate analysis. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
complete.permutation	Argument defines the number of multiple imputed data sets. In general, the number is defined by the number of imputations of a variable. Therefore, this argument only becomes relevant, if the number of imputation of one variable (e.g. plausible values of the dependent variable) differs from the number of imputations used for group variable(s). If "all", number of datasets are determined through permutation, e.g. 5 plausible values and 3 imputations of one group variable results in 3x5=15 imputed data sets. If "nothing", only 5 imputed data sets will be used, no matter whether the larger number of imputations is a whole multiple of the smaller number of imputations. If "groups", only the number of imputations of more than one grouping variable will be permuted.

Details

Function first creates replicate weights based on JKZone and JKrep variables according to JK2 procedure implemented in WesVar. According to multiple imputed data sets, a workbook with several analyses is created. The function afterwards serves as a wrapper for svymean() called by svyby() implemented in the 'survey' package. The results of the several analyses are then pooled according to Rubin's rule.

Value

A list of data frames, one for each dependent variable. Each data frame contains means, variances, standard deviations and standard errors for all of them. If group differences were estimated, they occur as attributes of each data frame. See examples for further details.

Author(s)

Sebastian Weirich

Examples

```
data(reading_writing)
### First example: only means, SD and variances for each country
means <- jk2.mean(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country"),
  dependent = list(reading = paste("reading_score",1:3,sep=""), writing = paste("writing_score",1:3,sep="")),
  complete.permutation = "all" )
### Second example: Sex differences by country
means <- jk2.mean(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country", GENDER = "sex"), group.differences.by = "GENDER",
  dependent = list(reading = paste("reading_score",1:3,sep=""), writing = paste("writing_score",1:3,sep="")),
  complete.permutation = "all" )
attr(means[[1]], "difference")
```

jk2.quantile

JK2 method for quantiles.

Description

Compute quantiles with standard errors for complex cluster designs with multiple imputed variables (e.g. plausible values) based on Jackknife (JK2) procedure. Conceptually, the function combines replication methods and methods for multiple imputed data. Technically, this is a wrapper for the `svyquantile()` function of the `survey` package.

Usage

```
jk2.quantile(dat, ID, wgt = NULL, JKZone, JKrep, group = list(), dependent = list(),
  probs = seq(0, 1, 0.25), complete.permutation = c("nothing", "groups", "all"))
```

Arguments

<code>dat</code>	Data frame containing all variables for analysis.
<code>ID</code>	Variable name or column number of ID variable.
<code>wgt</code>	Optional: Variable name or column number of weighting variable. If no weighting variable is specified, all cases will be equally weighted.
<code>JKZone</code>	Variable name or column number of variable indicating Jackknifing Zone.
<code>JKrep</code>	Variable name or column number of variable indicating replicate ID.
<code>group</code>	Optional: List of one or more grouping variables. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.

dependent	List of one or more grouping variables. Each dependent variable will result in a separate analysis. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
probs	Numeric vector with probabilities for which to compute quantiles.
complete.permutation	Argument defines the number of multiple imputed data sets. In general, the number is defined by the number of imputations of a variable. Therefore, this argument only becomes relevant, if the number of imputation of one variable (e.g. number of plausible values of the dependent variable) differs from the number of imputations used for group variable(s). If "all", number of datasets are determined through permutation, e.g. 5 plausible values and 3 imputations of one group variable results in 3x5=15 imputed data sets. If "nothing", only 5 imputed data sets will be used, no matter whether the larger number of imputations is a whole multiple of the smaller number of imputations. If "groups", only the number of imputations of more than one grouping variable will be permuted.

Details

Function first creates replicate weights based on JKZone and JKrep variables according to JK2 procedure implemented in WesVar. According to multiple imputed data sets, a workbook with several analyses is created. The function afterwards serves as a wrapper for svyquantile() called by svyby() implemented in the 'survey' package. The results of the several analyses are then pooled according to Rubin's rule.

Value

A list of data frames, one for each dependent variable. Each data frame contains percentile values and their standard errors.

Author(s)

Sebastian Weirich

Examples

```
data(reading_writing)
### Computes percentile for reading and writing scores conditionally on country
percent <- jk2.quantile(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country"),
  dependent = list(reading = paste("reading_score", 1:3, sep = ""), writing = paste("writing_score", 1:3, sep = "")),
  probs = seq(0.1, 0.9, 0.2),
  complete.permutation = "no" )
```

Description

Compute frequency tables with standard errors for complex cluster designs with multiple imputed variables based on Jackknife (JK2) procedure. Conceptually, the function combines replication methods and methods for multiple imputed data. Technically, this is a wrapper for the `svymean()` function of the survey package called to dichotomous indicators of the dependent variable.

Usage

```
jk2.table(dat, ID, wgt = NULL, JKZone, JKrep, group = list(), dependent = list(),
          separate.missing.indikator = FALSE, expected.values = list(),
          complete.permutation = c("nothing", "groups", "all"))
```

Arguments

<code>dat</code>	Data frame containing all variables for analysis.
<code>ID</code>	Variable name or column number of ID variable.
<code>wgt</code>	Optional: Variable name or column number of weighting variable. If no weighting variable is specified, all cases will be equally weighted.
<code>JKZone</code>	Variable name or column number of variable indicating Jackknifing Zone.
<code>JKrep</code>	Variable name or column number of variable indicating replicate ID.
<code>group</code>	Optional: List of one or more grouping variables. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
<code>dependent</code>	List of one or more grouping variables. Each dependent variable will result in a separate analysis. If grouping variable is a multiple imputation variable, all names concerning one variable are interpreted as its imputations. See details for more information.
<code>separate.missing.indikator</code>	Logical.
<code>expected.values</code>	Optional. A vector auf values expected in dependent variable. Recommend to left this argument empty.
<code>complete.permutation</code>	Argument defines the number of multiple imputed data sets. In general, the number is defined by the number of imputations of a variable. Therefore, this argument only becomes relevant, if the number of imputation of one variable (e.g. dependent variable) differs from the number of imputations used for group variable(s). If "all", number of datasets are determined through permutation, e.g. 5 plausible values and 3 imputations of one group variable results in 3x5=15 imputed data sets. If "nothing", only 5 imputed data sets will be used, no matter whether the larger number of imputations is a whole multiple of the smaller number of imputations. If "groups", only the number of imputations of more than one grouping variable will be permuted.

Details

Function first creates replicate weights based on `JKZone` and `JKrep` variables according to JK2 procedure implemented in `WesVar`. According to multiple imputed data sets, a workbook with several analyses is created. The function afterwards serves as a wrapper for `svymean()` called by `svyby()`

implemented in the 'survey' package. Relative frequencies of the categories of the dependent variable are computed by the means of the indicators of each category. The results of the several analyses are then pooled according to Rubin's rule.

Value

A list of data frames, one for each dependent variable. Each data frame contains columns for group variables (if specified), a columns called "suffix" which indicates the value of the corresponding categorie, the frequency of this categorie and it's standard error.

Author(s)

Sebastian Weirich

Examples

```
data(reading_writing)
### Computes frequencies of the hisei group conditionally on the groups of passed and failed with respect to
freq.tab <- jk2.table(dat = reading_writing, ID = "idstud", wgt = "wgtSTUD", JKZone = "JKZone", JKrep = "JKrep",
  group = list(country = "country", passed.reading = paste("passed_reading",1:3,sep="") ),
  dependent = list(hisei = paste("zehisei", 1:5, sep = "") ) ,
  complete.permutation = "no" )
```


Index

*Topic \textasciitildekw1

jk2.glm, [2](#)
jk2.mean, [4](#)
jk2.quantile, [5](#)
jk2.table, [6](#)

*Topic \textasciitildekw2

jk2.glm, [2](#)
jk2.mean, [4](#)
jk2.quantile, [5](#)
jk2.table, [6](#)

jk2.glm, [2](#)
jk2.mean, [4](#)
jk2.quantile, [5](#)
jk2.table, [6](#)