Package 'SynthTools'

March 11, 2020

Title Tools and Tests for Experiments with Partially Synthetic Data Sets

Version 1.0.1

Description A set of functions to support experimentation in the utility of partially synthetic data sets. All functions compare an observed data set to one or a set of partially synthetic data sets derived from the observed data to (1) check that data sets have identical attributes, (2) calculate overall and specific variable perturbation rates, (3) check for potential logical inconsistencies, and (4) calculate confidence intervals and standard errors of desired variables in multiple imputed data sets. Confidence interval and standard error formulas have options for either synthetic data sets or multiple imputed data sets. For more information on the formulas and methods used, see Reiter & Raghunathan (2007) <doi:10.1198/016214507000000932>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

Depends R (>= 2.10)

Imports dplyr, magrittr, Rdpack, utils

Suggests synthpop, testthat (>= 2.1.0), spelling

RdMacros Rdpack

Language en-US

NeedsCompilation no

Author Charlotte Looby [aut, cre]

Maintainer Charlotte Looby <clooby@rti.org>

Repository CRAN

Date/Publication 2020-03-11 13:00:02 UTC

R topics documented:

`ContCI` *Confidence intervals and standard errors of multiple imputation for a specific imputed continuous variable.*

Description

This function will calculate confidence intervals and standard errors from the responses of multiple imputed datasets for a specified continuous variable, and also give a YES/NO indicator for whether or not the observed value is within the confidence interval. The confidence intervals and standard errors are calculated by first taking the means of the variable from the partially synthesized datasets, then using `t.test()` to get the confidence intervals.

Usage

```
ContCI(obs_data, imp_data_list, var, sig = 6, alpha = 0.05)
```

Arguments

`obs_data`	The original dataset to which the next will be compared, of the type "data.frame".
`imp_data_list`	A list composed of `m` synthetic data sets.
`var`	The continuous variable being checked.
`sig`	The number of significant digits in the output data frame. Defaults to 6.
`alpha`	Test size, defaults to 0.05.

Details

This function was developed with the intention of making the job of researching partially synthetic data utility a bit easier by providing another way of measuring utility.

Value

This function returns a data frame with the variable's observed mean, lower and upper limits of the confidence interval, standard error, and a YES/NO indicating whether or not the observed value is within the confidence interval.

Examples

```
#"PPA" is the observed data set
#"PPAm5" is a list of 5 partially synthetic data sets derived from PPA
#"age" is a continuous variable present in the synthesized data sets.
#3 significant digits are desired from the output data frame.

ContCI(PPA, PPAm5, "age", sig=3)
```

---

dataComp *Checking for equality in the features of two data sets.*

Description

This function will check for comparability between two data sets, including dimensions, order of variables, variable classifications, and levels of factors. When a data set is fully or partially synthesized from an observed data set, these are the features that should be equal between the data sets so the utility of the synthetic data can be measured.

Usage

```
dataComp(obs_data, new_data)
```

Arguments

obs_data	The original data set to which the next will be compared, of the type "data.frame".
new_data	The fully or partially synthetic data set to be compared to the observed data, of the type "data.frame".

Details

This function was developed with the intention of making the job of researching synthetic data utility a bit easier by making preliminary data set comparisons quickly.

Value

A list containing the following components:

same.dim	A logical value indicating whether or not obs_data and new_data have the same dimensions.
same.order	A logical value indicating whether or not the variables in obs_data and new_data are in the same order.

`class.identical` A logical value indicating where or not the variable classifications are identical.

`class.table` A table of types of variable classifications.

`fac.num.same` A logical value indicating whether or not the factors in the data sets have the same number of levels.

`fac.lev.same` A logical value indicating whether or not the factors in the data sets have the same levels.

Examples

```
#PPA is observed data set, PPAps1 is a partially synthetic data set derived from the observed data.

dataComp(PPA, PPAps1)
```

`logicCheck` *Checking for logical consistency between two categorical variables in a synthesized data set.*

Description

This function will check for logical consistency between two categorical variables in a fully or partially synthesized data set.

Usage

```
logicCheck(obs_data, new_data, vars, NAopt = T)
```

Arguments

`obs_data` The original data set to which the next will be compared, of the type "data.frame".

`new_data` The fully or partially synthetic data set to be compared to the observed data, of the type "data.frame".

`vars` A vector of two categorical variables in the data sets to check for logical consistency.

`NAopt` Defaults to TRUE to use NAs in tables. If you do not wish to check for NAs, put FALSE.

Details

When a data set is fully or partially synthesized from an observed data set, sometimes there are logical consistencies in the observed data set which must be adhered to in the synthesized data set that may be violated during the course of the synthesis. For example, if there is a data set which contains an age variable and a variable that represents whether or not a person has a drivers license in the state of Pennsylvania, the age variable should indicate that the person is at least 16-years-old if the license indicator shows that the person has a drivers license. It is recommended that you check for data comparability with `dataComp()` prior to using this function.

This function creates cross-tabulations of the specified variables of both the observed data set and synthesized data set, then checks that the corresponding cell values are either zero or a positive value accordingly. It was developed with the intention of making the job of researching synthetic data utility a bit easier by quickly checking for logical consistency.

Value

This function returns a message stating whether or not there were any potential logical inconsistencies found in the data sets for the variables specified. Then the cross-tabulations will be printed (in either case) for the analyst to review.

This function will also return a list of the following components:

`consistent`	A logical value indicating whether the variable cross-tabulation is logically consistent.
`obs.table`	The original data set cross-tabulation.
`new.table`	The new data set cross-tabulation.
`which`	A matrix indicating if values are logically consistent. 0=consistent, otherwise=inconsistent.

Examples

```
#PPA is observed data set, PPAps2 is a partially synthetic data set derived from the observed data.
#age17plus and marriage are two categorical variables within these data sets.

logicCheck(PPA, PPAps2, c("age17plus", "marriage"))
```

---

oneCatCI *Confidence intervals and standard errors for one synthetic categorical variable of derived with multiply imputed datasets.*

Description

This function will calculate confidence intervals and standard errors from the proportional responses of multiply imputed datasets for a specified categorical variable, and also gives a YES/NO indicator for whether or not the observed value is within the confidence interval. The confidence intervals and standard errors are calculated from variance formulas that are specific to whether the multiple imputed datasets are fully or partially synthetic. See reference for more information.

Usage

```
oneCatCI(obs_data, imp_data_list, type, var, sig = 6, alpha = 0.05)
```

Arguments

obs_data	The original dataset to which the next will be compared, of the type "data.frame".
imp_data_list	A list of datasets that are either synthetic or contain imputed values.
type	Specifies which type of datasets are in imp_data_list. Options are "fully" and "partially".
var	The categorical variable being checked. Should be of type "factor".
sig	The number of significant digits in the output dataframe. Defaults to 6.
alpha	Test size, defaults to 0.05.

Details

This function was developed with the intention of making the job of researching synthetic data utility a bit easier by providing another way of measuring utility.

Value

This function returns a dataframe with the variable's responses, observed values, lower and upper limits of the confidence interval, standard error, and "YES"/"NO" indicating whether or not the observed value is within the confidence interval.

References

Reiter JP, Raghunathan TE (2007). "The Multiple Adaptations of Multiple Imputation." *Journal of the American Statistical Association*.

Examples

```
#PPA is observed data set, PPAm5 is a list of 5 partially synthetic data sets derived from PPA.
#sex is a categorical variable within these data sets. 3 significant digits are desired.

oneCatCI(obs_data=PPA, imp_data_list=PPAm5, type="partially", var="sex", sig=3)
```

---

pertRates *Calculates perturbation rates of overall data set and specific variables.*

Description

This function will calculate the overall perturbation rate of an imputed data set and for specific variables requested.

Usage

```
pertRates(obs_data, new_data, imp_vars, desc = FALSE, sig = 4)
```

Arguments

obs_data	The original dataset to which the next will be compared, of the type "data.frame".
new_data	The fully or partially synthetic data set to be compared to the observed data, of the type "data.frame".
imp_vars	The variable or a vector of variables which were imputed and are to be used in the overall perturbation rate calculation.
desc	Whether or not the variable perturbation rates should be output in descending rate order. Defaults to FALSE.
sig	The number of significant digits desired for the overall perturbation rate. Defaults to 4.

Details

A record in a data set is considered "perturbed" when at least one value in the record is different from the observed data. The overall perturbation rate is therefore the number of records that are found to be perturbed over the number of records in a data set.

The variable perturbation rate is simply the rate at which the values for a given variable are different from those in the observed data set.

This function was developed with the intention of making the job of researching synthetic data utility a bit easier by quickly calculating perturbation rates.

Value

Returns the overall perturbation rate of the synthetic data set and the specific variable perturbation rates in percentages, rounded to 0.1. The function will also output in list format with the following components:

overall	The overall perturbation rate.
variable	A vector of variable perturbation rates.

Examples

```
#PPA is observed data set, PPAps2 is a partially synthetic data set derived from the observed data.
#age17plus, marriage, and vet are three categorical variables within these data sets.

pertRates(PPA, PPAps2, c("age17plus", "marriage", "vet"))
```

PPA *Characteristics of 1000 People in Pennsylvania.*

Description

A dataset containing some variables about 1000 people in Pennsylvania. This is a subset of the 2017 ACS PUMS data with one indicator variable added.

Usage

```
PPA
```

Format

A data frame with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

---

PPAm5 *A list containing 5 partially synthetic data sets.*

Description

This is a list that has the 5 partially synthetic versions of PPA (PPAps1 - PPAps5).

Usage

```
PPAm5
```

Format

5 data frames with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

PPAps1 *Characteristics of 1000 People in Pennsylvania, partially synthetic (set 1).*

Description

This is a version of the PPA data set that is partially synthetic. Some of the values of "sex", "marriage", and "age17plus" were imputed.

Usage

```
PPAps1
```

Format

A data frame with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

PPAps2 *Characteristics of 1000 People in Pennsylvania, partially synthetic (set 2).*

Description

This is a version of the PPA data set that is partially synthetic. Some of the values of "sex", "marriage", and "age17plus" were imputed.

Usage

```
PPAps2
```

Format

A data frame with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

---

`PPAps3` *Characteristics of 1000 People in Pennsylvania, partially synthetic (set 3).*

Description

This is a version of the PPA data set that is partially synthetic. Some of the values of "sex", "marriage", and "age17plus" were imputed.

Usage

```
PPAps3
```

Format

A data frame with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

PPAps4 *Characteristics of 1000 People in Pennsylvania, partially synthetic (set 4).*

Description

This is a version of the PPA data set that is partially synthetic. Some of the values of "sex", "marriage", and "age17plus" were imputed.

Usage

```
PPAps4
```

Format

A data frame with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

PPAps5 *Characteristics of 1000 People in Pennsylvania, partially synthetic (set 5).*

Description

This is a version of the PPA data set that is partially synthetic. Some of the values of "sex", "marriage", and "age17plus" were imputed.

Usage

```
PPAps5
```

Format

A data frame with 1000 rows and 7 variables:

age age of respondent, in years, ("AGEP")

sex sex of respondent, ("SEX")

race recoded detailed race code, ("RAC1P")

marriage married/spouse present/spouse absent, ("MSP")

emp employment status recode, ("ESR")

vet veteran period of service, ("VPS")

age17plus age >= 17 indicator

Source

https://factfinder.census.gov/faces/tableservices/jsf/pages/productview.xhtml?pid=ACS_pums_csv_2017&prodType=document

---

`twoCatCI` *Confidence intervals and standard errors for the cross-tabulation of two categorical variables of derived with multiply imputed datasets.*

Description

This function will calculate confidence intervals and standard errors from the proportional tabular responses of multiply imputed datasets for the cross-tabulation of two categorical variables, and also give a YES/NO indicator for whether or not the observed value is within the confidence interval. The confidence intervals and standard errors are calculated from formulas that are adapted for fully and partially synthetic data sets. See reference for more information.

Usage

```
twoCatCI(obs_data, imp_data_list, type, vars, sig = 4, alpha = 0.05)
```

Arguments

`obs_data`	The original dataset to which the next will be compared, of the type "data.frame".
`imp_data_list`	A list composed of `m` synthetic data sets.
`type`	Specifies which type of datasets are in `imp_data_list`. Options are "fully" and "partially".
`vars`	A vector of the two categorical variable being checked. Should be of type "factor".
`sig`	The number of significant digits in the output dataframes. Defaults to 4.
`alpha`	Test size, defaults to 0.05.

Details

This function was developed with the intention of making the job of researching synthetic data utility a bit easier by providing another way of measuring utility.

Value

This function returns a list of five data frames:

`Observed`	A cross-tabular proportion of observed values
`Lower`	Lower limit of the confidence interval
`Upper`	Upper limit of the confidence interval
`SEs`	Standard Errors
`CI_Indicator`	"YES"/"NO" indicating whether or not the observed value is within the confidence interval

References

Reiter JP, Raghunathan TE (2007). "The Multiple Adaptations of Multiple Imputation." *Journal of the American Statistical Association*.

Examples

```
#PPA is the observed data set. PPAm5 is a list of 5 partially synthetic data sets derived from PPA.
#"sex" and "race" are categorical variables present in the synthesized data sets.
#3 significant digits are desired in the output dataframes.

twoCatCI(PPA, PPAm5, "partially", c("sex", "race"), sig=3)
```

Index