Package 'imputevalR'

December 14, 2021

Title Multiple Imputation of Missing Data Using Predictive Sampling

Version 0.1

Date 2021-12-15

Description imputvalR contains methods to impute all missing values in a data frame for all variables in the data, so long as the variables are either continuous or binary. The imputation uses a Gibbs sampler to draw from the predictive distributions of the variable of interest, conditional on all other variables in the data set, and then returns a list of data.frames that can be used to complete pooled regression analyses (e.g. using Rubin's rules) or other sample statistics.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

LazyData true

Imports Rcpp (>= 1.0.7), methods, Rdpack, stats, mice, utils

Depends R (>= 2.10)

NeedsCompilation no

Author Yijing Feng [aut, cre],
Jenny Jin [aut],
Matt Lee [aut],
Meg Salvia [aut],
Jose Villa-Uribe [aut],
Qingru Xu [aut]

Maintainer Yijing Feng <yfeng@g.harvard.edu>

R topics documented:

`compare_by_cell` *Squared error between means for all cells in a dataset*

Description

Squared error between means for all cells in a dataset

Usage

```
compare_by_cell(df1, df2)
```

Arguments

`df1`	The first data.frame for which comparisons should be made
`df2`	The second data.frame for which comparisons should be made

Value

`summed_diff`, a named vector of length equal to the number of variables, where each value is the summed squared difference between the imputed and true value of a cell, for each variable

`compare_by_column` *Squared error between means for all columns in a dataset*

Description

Squared error between means for all columns in a dataset

Usage

```
compare_by_column(df1, df2)
```

Arguments

`df1`	The first data.frame for which comparisons should be made
`df2`	The second data.frame for which comparisons should be made

Value

`squarediff`, a named vector with length equal to the the number of variables for the comparison

imputer — *Run Imputation Procedure and Generate Imputed Datasets*

Description

Takes in a data.frame with missing values, and runs the imputation algorithm to return 5 imputed data sets (stored as a list) with missing values replaced by imputed values.

Usage

```
imputer(df, nchains = 5, niter = 100)
```

Arguments

df	A data.frame for which imputed data sets should be generated
nchains	The number of chains (ie number of imputed datasets) to generate
niter	The number of iterations to complete in each chain (default is 100 for convergence)

Value

A list (returnSets) that contains 5 imputed data.frames with missing values that were originally in df with imputed values based on the predicted imputed values

Examples

```
data(nhanes)
nhanes <- nhanes[1:100, ] # subset for speed
miss_data <- imputevalR::makeNA(nhanes, proportionNA = 0.2)
imputed <- imputer(miss_data, nchains = 1, niter = 5)
```

makeNA — *Set random values in a data frame to missing*

Description

Set random values in a data frame to missing

Usage

```
makeNA(df, proportionNA = 0.2)
```

Arguments

df	A data.frame for which NA values should be generated
proportionNA	The proportion of all cells that should be NA, across variables and observations

Value

A data.frame (`df`) that is the same dimensions as the input data frame, but now has values missing (set by `proportionNA`)

Examples

```
data(nhanes, package = "imputevalR")
miss_data <- imputevalR::makeNA(nhanes, proportionNA = 0.2)
```

`nhanes` *Example NHANES data with no missing values*

Description

A dataset containing sample variables from the National Health and Nutrition Examination Survey (2017-18) cycle

Usage

```
nhanes
```

Format

A data frame with 9254 rows and 21 variables:

gender gender (male/female)

age age in years

white Non-Hispanic White race/ethnicity

poverty ratio of household income to federal poverty level

weight measured weight in kg

height measured height in cm

bmi BMI in kg/m^2

waist_circum waist circumference in cm

hip_circum hip circumference in cm

sbp1 systolic blood pressure (4 readings)

sbp2 systolic blood pressure (4 readings)

sbp3 systolic blood pressure (4 readings)

sbp4 systolic blood pressure (4 readings)

dbp1 diastolic blood pressure (4 readings)

dbp2 diastolic blood pressure (4 readings)

dbp3 diastolic blood pressure (4 readings)

dbp4 diastolic blood pressure (4 readings)

android_pfat percent fat in android

gynoid_pfat percent fat in gyneoid

selfreported_weight self rep weight in lbs

selfreported_ht self rep height in inches

Source

https://www.cdc.gov/nchs/nhanes/index.htm

`pool_df` *Pool imputed data frame results*

Description

Pool imputed data frame results

Usage

```
pool_df(dfList)
```

Arguments

`dfList`	A list of imputed data.frames generated by imputer()

Value

`finalDF`: A single data.frame containing values that have been pooled across the imputations using Rubin's rules

---

`simulate_nhanes_study` *Simulation study to evaluate imputer() vs. MICE using predictive mean matching*

Description

Completes a simulation study to compare the differences between `mice()` from the mice package and `imputer()`, using an example complete case NHANES dataset from 2017-2018. Differences are generated as the absolute difference between (1) `imputer` and the original NHANES data and (2) `mice` and the original NHANES data, scaled by the difference `mice` and the original NHANES data. Results are therefore interpreted as percent difference between `imputer` and `mice` as a function of the difference between `mice` and the true cell values.

Usage

```
simulate_nhanes_study(
  numSims = 50,
  proportionNA = 0.2,
  nchains = 3,
  niter = 100
)
```

Arguments

numSims	Number of simulations to run
proportionNA	The proportion of all cells that should be NA, across variables and observations, to be passed to makeNA()
nchains	The number of chains (ie number of imputed datasets) to generate for imputer()
niter	The number of iterations to complete in each chain (default is 100 for convergence) for imputer()

Value

A data.frame (differences) that includes the results of the simulation, with a row for each simulation and a column for each variable

Examples

```
# NOT RUN FOR RUNTIME
## Not run:
library(imputevalR)
simulate_nhanes_study(numSims = 20, proportionNA = 0.2)

## End(Not run)
```

Index