R Package Document: cdampute

Missing Values Simulations using cdampute

2023-11-25

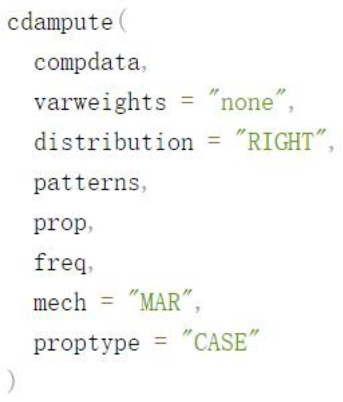
Installation:

STEP 1: Download the *.tar.gz* file from this website (https://github.com/SilviaChen1998/cdampute-missing-values-simulation) and store it in your folder.
STEP 2: Open *R*, and run this code - *install.packages(your file path, repos=NULL, type='source')*.

Introduction to *cdampute*

cdampute is an simple R package for generate missing values in complete datasets. *cdampute* provides more flexible choices for missing value settings. It will be a useful tool for researchers who care about missing data issues in their projects to evaluate their methods in data with different missing rates, missing mechanisms and missing patterns.

Parameters in *cdampute*



```
cdampute(
  compdata,
  varweights = "none",
  distribution = "RIGHT",
  patterns,
  prop,
  freq,
  mech = "MAR",
  proptype = "CASE"
)
```

Figure 1. Usage of cdampute function.

Parameters	Description
compdata	The complete data frame that need to be amputed, containing ***n*** records and ***p*** variables.
varweights	A ***p*p*** weight matrix for variables. The ***i-th*** row of this matrix defines how the values of all ***p*** variables contributes to the missingness of ***i-th*** variable. If the value in the ***i-th*** row and ***j-th*** column of varweights matrix equals to 0, it means that the absence of the ***i-th*** variable is not related to the value of the ***j-th*** variable. The weighted sum score of each variable is computed by *compdata%*%t(varweights)* in R. If mech='MCAR', the varweights will not be used. If mech='MAR' or 'MNAR', users can define varweights according to the definitions of mechanisms and their special requirements. If users do not input a weight matrix, varweights will be generated

	automatically with random values, and the diagonal values will be set to 0 if mech='MAR'.
distribution	A vector (length = ***p***) used to map the weighted sum scores of variables into probabilities of missingness. Four types of missing distributions, ***'LEFT','RIGHT','MID'*** and ***'TAIL'*** are provided. Users can also just input one certain word to represent that generate missing values for all variables using this one type of distribution. The default is ***'RIGHT'***.
patterns	A ***n*p*** matrix, where 0 indicates that the value of certain variable is missing, while 1 stands for that the variable is observed.
prop	The numerical missing rate.
proptype	The type of missing rate. Missing by ***'CASE'*** and ***'CELL'*** are provided. The default is by ***'CASE'***.
freq	A vector containing the proportion of the compdata assigned to each missing pattern. The sum of all values in a freq vector should be equal to 1. *freq*prop*the size of the compdata* is the final size of the incomplete data under each missing pattern.
mech	One of the ***'MCAR','MAR'*** and ***'MNAR'*** missing mechanisms. ***'MAR'*** is the default.

Weighted Sum Scores:

$$WSS_{k,i} = \sum_{j=1}^{p} w_{i,j} x_{k,j}\,;\ \ i,j = 1,2,\dots,p;\ k = 1,2,\dots,n$$

where $w_{i,j}$ is the value in i-th row and j-th column of *varweights*. $x_{k,j}$ is the value in the k-th row and j-th column of *compdata.*

LEFT distribution:

$$P(x_{k,i} = missing) = \frac{1}{1 + e^{centralized\_wss_{k,i}}}$$

RIGHT distribution:

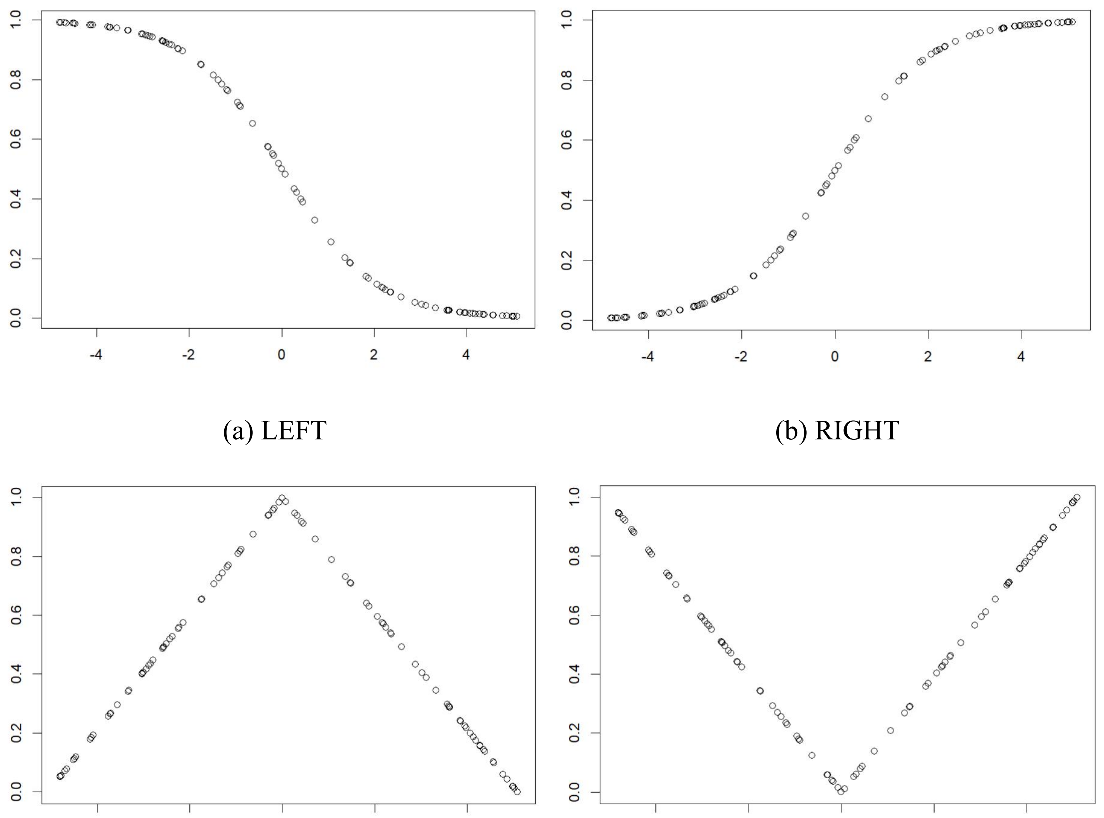
$$P(x_{k,i} = missing) = \frac{1}{1 + e^{-centralized\_wss_{k,i}}}$$

MID distribution:

$$P(x_{k,i} = missing) = \frac{|centralized\_wss_{k,i}|}{max(|centralized\_wss_{k,i}|)}$$

TAIL distribution:

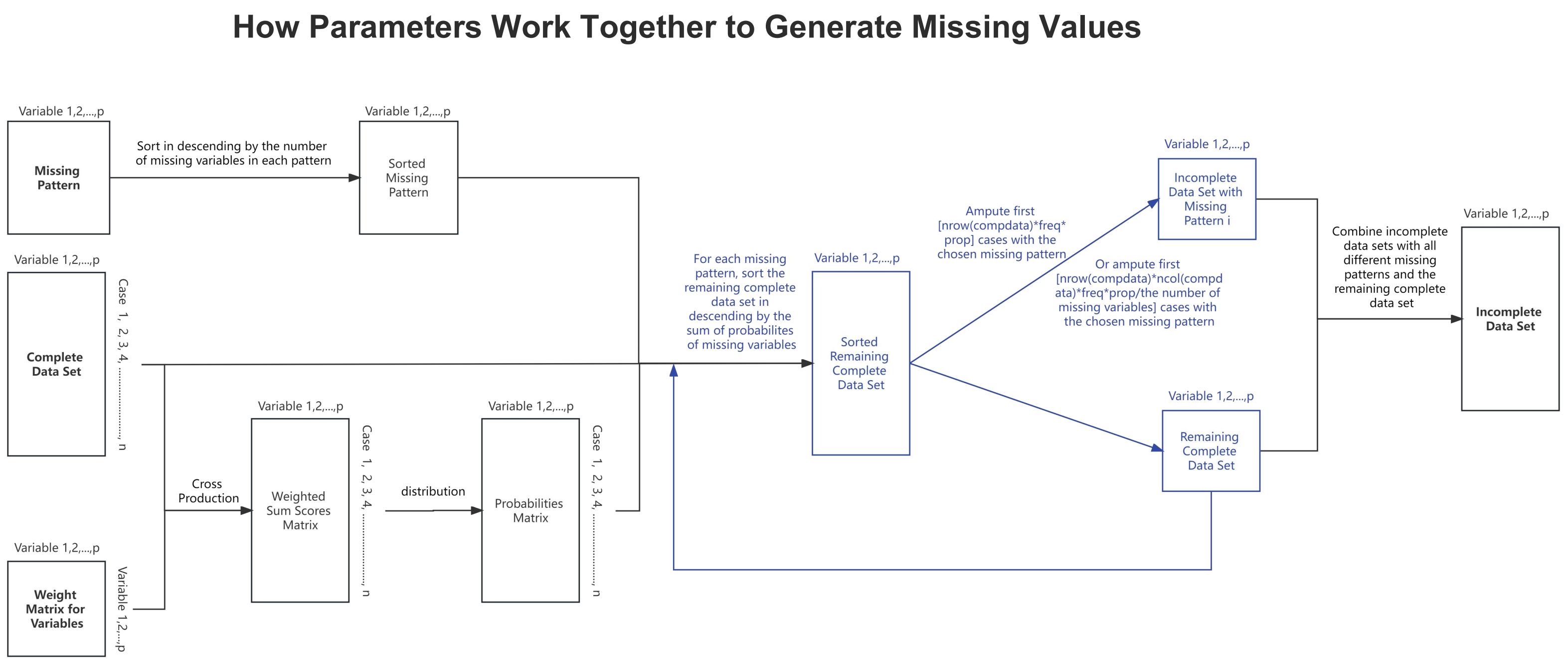
$$P(x_{k,i} = missing) = 1 - \frac{|centralized\_wss_{k,i}|}{max(|centralized\_wss_{k,i}|)}$$



(a) LEFT (b) RIGHT

(c) MID (d) TAIL

Figure 2. Types of distribution.



How Parameters Work Together to Generate Missing Values
Variable 1,2,...,p
Missing Pattern
Sort in descending by the number of missing variables in each pattern
Sorted Missing Pattern
Complete Data Set
Case 1, 2, 3, 4, ..., n
Weight Matrix for Variables
Variable 1,2,...,p
Cross Production
Weighted Sum Scores Matrix
distribution
Probabilities Matrix
For each missing pattern, sort the remaining complete data set in descending by the sum of probabilites of missing variables
Sorted Remaining Complete Data Set
Ampute first [nrow(compdata)*freq*prop] cases with the chosen missing pattern
Or ampute first [nrow(compdata)*ncol(compdata)*freq*prop/the number of missing variables] cases with the chosen missing pattern
Incomplete Data Set with Missing Pattern i
Remaining Complete Data Set
Combine incomplete data sets with all different missing patterns and the remaining complete data set
Incomplete Data Set

Figure 3. The process of generating missing values.

STEP 1: Cross product the *compdata* and *varweights*, obtaining the weighted sum score of each cells.
STEP 2: Transfer the weighted sum scores into probabilities of missingness using the chosen distribution. Now every cell has its probability of missingness.
STEP 3: Sort the *patterns* in descending by the number of missing variables. Patterns with more missing variables come first.
STEP 4: Select one missing pattern from *the sorted patterns* in order. Add up the missing probabilities of variables indicated as missing variables by the chosen missing pattern, and sort the complete data set in descending by the pattern specific missing probabilities sum.
STEP 5: Compute the number of incomplete cases, *n_missing*, by multiplying *n, freq* and *prop* if *proptype='CASE'*, by multiplying *n, p, freq, prop* then divided by *the number of missing variables* in the chosen missing pattern if *proptype='CELL'*.
STEP 6: Ampute first *n_missing* cases in the complete data set, obtaining the incomplete data set with the chosen missing pattern.
STEP 7: Repeat STEP 4-6 in the remaining complete data set until every missing pattern is used.