Package 'MultiVarMI'

April 9, 2018

Type Package

Title Multiple Imputation for Multivariate Data

Version 1.0

Date 2018-04-08

Author Rawan Allozi, Hakan Demirtas

Maintainer Rawan Allozi <ralloz2@uic.edu>

Description
Fully parametric Bayesian multiple imputation framework for massive multivariate data of different variable types as seen in Demirtas, H. (2017) <doi:10.1007/978-981-10-3307-0_8>.

License GPL-2 | GPL-3

Imports BinOrdNonNor, CorrToolBox, corpcor, Matrix, moments, norm, PoisNonNor

Suggests PoisBinOrdNonNor

NeedsCompilation no

Repository CRAN

Date/Publication 2018-04-09 11:55:15 UTC

R topics documented:

MultiVarMI-package 2
countrate 3
MI 4
MVN.corr 6
MVN.dat 9
nctsum 10
ordmps 12
trMVN.dat 13

Index **15**

`MultiVarMI-package` *Multiple Imputation for Multivariate Data*

Description

This package implements a Bayesian multiple imputation framework for multivariate data. Most incomplete data sets constist of interdependent binary, ordinal, count, and continuous data. Furthermore, planned missing data designs have been developed to reduce respondent burden and lower the cost associated with data collection. The unified, general-purpose multiple imputation framework described in Demirtas (2017) can be utilized in developing power analysis guidelines for intensive multivariate data sets that are collected via increasingly popular real-time data capture (RTDC) approaches. This framework can accommodate all four major types of variables with a minimal set of assumptions. The data are prepared for multivariate normal multiple imputation for use in the `norm` package and subsequently backtransformed to the original distribution.

This package consists of one main function and six auxiliary functions. Multiple imputation can be performed using the function `MI`. While the auxiliary functions are utilized in `MI`, they can be used as stand-alone functions. `nctsum` outputs a list with summary statistics and Fleishman coefficients and standardized forms of each variable. `ordmps` is utilized for ordinal variables and outputs a list with empirical marginal probabilities and the associated observations for each ordinal variable. `countrate` is designed for variables and outputs a list with empirical rates and the associated observations for each ordinal variable. `MVN.corr` calculates the intermediate correlation matrix, `MVN.dat` transforms variables to a standard normal variable, and `trMVN.dat` transforms standard normal variables to ordinal, count, and/or non-normal continuous variables through specified parameters.

Details

Package:	MultiVarMI
Type:	Package
Version:	1.0
Date:	2018-04-08
License:	GPL-2 | GPL-3

Author(s)

Rawan Allozi, Hakan Demirtas

Maintainer: Rawan Allozi <ralloz2@uic.edu>

References

Demirtas, H. and Hedeker, D. (2011). A practical way for computing approximate lower and upper correlation bounds. *The American Statistician*, **65(2)**, 104-109.

Demirtas, H., Hedeker, D., and Mermelstein, R. J. (2012). Simulation of massive public health data by power polynomials. *Statistics in Medicine*, **31(27)**, 3337-3346.

Demirtas, H. (2016). A note on the relationship between the phi coefficient and the tetrachoric correlation under nonnormal underlying distributions. *The American Statistician*, **70(2)**, 143-148.

Demirtas, H. and Hedeker, D. (2016). Computing the point-biserial correlation under any underlying continuous distribution. *Communications in Statistics-Simulation and Computation*, **45(8)**, 2744-2751.

Demirtas, H., Ahmadian, R., Atis, S., Can, F.E., and Ercan, I. (2016). A nonnormal look at polychoric correlations: modeling the change in correlations before and after discretization. *Computational Statistics*, **31(4)**, 1385-1401.

Demirtas, H. (2017). A multiple imputation framework for massive multivariate data of different variable types: A Monte-Carlo technique. *Monte-Carlo Simulation-Based Statistical Modeling*, edited by Ding-Geng (Din) Chen and John Dean Chen, Springer, 143-162.

Ferrari, P.A. and Barbiero, A. (2012). Simulating ordinal data. *Multivariate Behavioral Research*, **47(4)**, 566-589.

Fleishman A.I. (1978). A method for simulating non-normal distributions. *Psychometrika*, **43(4)**, 521-532.

Vale, C.D. and Maurelli, V.A. (1983). Simulating multivariate nonnormal distributions. *Psychometrika*, **48(3)**, 465-471.

`countrate` *Computation of Rates for Count Data*

Description

This function computes the empirical rates for count data.

Usage

```
countrate(count.dat)
```

Arguments

`count.dat`	A matrix consisting of count variables.

Value

A list of length ncol(count.dat) containing the data and empirical rates for each variable in count.dat.

See Also

`MI`, `MVN.corr`

Examples

```
library(PoisBinOrdNonNor)
set.seed(123)
n<-1e4

lambdas<-list(1, 3)

#generate Poisson data
cmat.star <- find.cor.mat.star(cor.mat = .4 * diag(2) + .6,
                               no.pois = length(lambdas),
                               pois.list = lambdas)

cntdata <- genPBONN(n,
                    no.pois = length(lambdas),
                    cmat.star = cmat.star,
                    pois.list = lambdas)

#set a sample of the data to missing
cntdata<-apply(cntdata, 2, function(x) {
  x[sample(1:n, size=n/10)]<-NA
  return(x)
})

cntdata<-data.frame(cntdata)
cntinfo<-countrate(cntdata)
```

MI *Bayesian Multiple Imputation for Multivariate Data*

Description

This function implements the multiple imputation framework as described in Demirtas (2017) "A multiple imputation framework for massive multivariate data of different variable types: A Monte-Carlo technique."

Usage

```
MI(dat, var.types, m)
```

Arguments

dat	A data frame containing multivariate data with missing values.
var.types	The variable type corresponding to each column in dat, taking values of "NCT" for continuous data, "O" for ordinal or binary data, or "C" for count data.
m	The number of stochastic simulations in which the missing values are replaced.

Value

A list containing m imputed data sets.

References

Demirtas, H. and Hedeker, D. (2011). A practical way for computing approximate lower and upper correlation bounds. *The American Statistician*, **65(2)**, 104-109.

Demirtas, H., Hedeker, D., and Mermelstein, R. J. (2012). Simulation of massive public health data by power polynomials. *Statistics in Medicine*, **31(27)**, 3337-3346.

Demirtas, H. (2016). A note on the relationship between the phi coefficient and the tetrachoric correlation under nonnormal underlying distributions. *The American Statistician*, **70(2)**, 143-148.

Demirtas, H. and Hedeker, D. (2016). Computing the point-biserial correlation under any underlying continuous distribution. *Communications in Statistics-Simulation and Computation*, **45(8)**, 2744-2751.

Demirtas, H., Ahmadian, R., Atis, S., Can, F.E., and Ercan, I. (2016). A nonnormal look at polychoric correlations: modeling the change in correlations before and after discretization. *Computational Statistics*, **31(4)**, 1385-1401.

Demirtas, H. (2017). A multiple imputation framework for massive multivariate data of different variable types: A Monte-Carlo technique. *Monte-Carlo Simulation-Based Statistical Modeling*, edited by Ding-Geng (Din) Chen and John Dean Chen, Springer, 143-162.

Ferrari, P.A. and Barbiero, A. (2012). Simulating ordinal data. *Multivariate Behavioral Research*, **47(4)**, 566-589.

Fleishman A.I. (1978). A method for simulating non-normal distributions. *Psychometrika*, **43(4)**, 521-532.

Vale, C.D. and Maurelli, V.A. (1983). Simulating multivariate nonnormal distributions. *Psychometrika*, **48(3)**, 465-471.

See Also

MVN.corr, MVN.dat, trMVN.dat

Examples

```
library(PoisBinOrdNonNor)
set.seed(1234)
n<-1e5
lambdas<-list(1, 3) #2 count variables
mps<-list(c(.2, .8), c(.6, 0, .3, .1)) #1 binary variable, 1 ordinal variable with skip pattern
moms<-list(c(-1, 1, 0, 1), c(0, 3, 0, 2)) #2 continuous variables

##############################################
#Generate Poisson, Ordinal, and Continuous Data#
##############################################
#get intermediate correlation matrix
cmat.star <- find.cor.mat.star(cor.mat = .8 * diag(6) + .2, #all pairwise correlations set to 0.2
                               no.pois = length(lambdas),
                               no.ord = length(mps),
                               no.nonn = length(moms),
                               pois.list = lambdas,
                               ord.list = mps,
```

```
                                nonn.list = moms)

#generate dataset
mydata <- genPBONN(n,
                   no.pois = length(lambdas),
                   no.ord = length(mps),
                   no.nonn = length(moms),
                   cmat.star = cmat.star,
                   pois.list = lambdas,
                   ord.list = mps,
                   nonn.list = moms)

cor(mydata)
apply(mydata, 2, mean)

#Make 10 percent of each variable missing completely at random
mydata<-apply(mydata, 2, function(x) {
  x[sample(1:n, size=n*0.1)]<-NA
  return(x)
  }
)


#Create 5 imputed datasets
mydata<-data.frame(mydata)
mymidata<-MI(dat=mydata,
             var.types=c('C', 'C', 'O', 'O', 'NCT', 'NCT'),
             m=5)

#get the means of each variable for the m imputed datasets
do.call(rbind, lapply(mymidata, function(x) apply(x, 2, mean)))

#get m correlation matrices of for the m imputed dataset
lapply(mymidata, function(x) cor(x))

#Look at the second imputed dataset
head(mymidata$dataset2)

##run a linear model on each dataset and extract coefficients
mycoef<-lapply(mymidata, function(x) {
  fit<-lm(X6~., data=data.frame(x))
  fit.coef<-coef(fit)
  return(fit.coef)
})

do.call(rbind, mycoef)
```

Description

This function calculates an intermediate correlation matrix for Poisson, ordinal, and continuous random variables, with specified target correlations and marginal properties.

Usage

```
MVN.corr(indat, var.types, ord.mps=NULL, nct.sum=NULL, count.rate=NULL)
```

Arguments

`indat`	A data frame containing multivariate data. Continuous variables should be standardized.
`var.types`	The variable type corresponding to each column in dat, taking values of "NCT" for continuous data, "O" for ordinal or binary data, or "C" for count data.
`ord.mps`	A list containing marginal probabilities for binary and ordinal variables as packaged from output in `ordmps`. Default is NULL.
`nct.sum`	A matrix containing summary statistics for continuous variables as packaged from output in `nctsum`. Default is NULL.
`count.rate`	A vector containing rates for count variables as packaged from output in `countrate`. Default is NULL.

Value

The intermediate correlation matrix.

References

Demirtas, H. and Hedeker, D. (2011). A practical way for computing approximate lower and upper correlation bounds. *The American Statistician*, **65(2)**, 104-109.

Demirtas, H., Hedeker, D., and Mermelstein, R. J. (2012). Simulation of massive public health data by power polynomials. *Statistics in Medicine*, **31(27)**, 3337-3346.

Demirtas, H. (2016). A note on the relationship between the phi coefficient and the tetrachoric correlation under nonnormal underlying distributions. *The American Statistician*, **70(2)**, 143-148.

Demirtas, H. and Hedeker, D. (2016). Computing the point-biserial correlation under any underlying continuous distribution. *Communications in Statistics-Simulation and Computation*, **45(8)**, 2744-2751.

Demirtas, H., Ahmadian, R., Atis, S., Can, F.E., and Ercan, I. (2016). A nonnormal look at polychoric correlations: modeling the change in correlations before and after discretization. *Computational Statistics*, **31(4)**, 1385-1401.

Ferrari, P.A. and Barbiero, A. (2012). Simulating ordinal data. *Multivariate Behavioral Research*, **47(4)**, 566-589.

Fleishman A.I. (1978). A method for simulating non-normal distributions. *Psychometrika*, **43(4)**, 521-532.

Vale, C.D. and Maurelli, V.A. (1983). Simulating multivariate nonnormal distributions. *Psychometrika*, **48(3)**, 465-471.

See Also

MI, MVN.dat, ordmps, nctsum, countrate

Examples

```
library(PoisBinOrdNonNor)
n<-1e4
lambdas<-list(1, 3)
mps<-list(c(.2, .8), c(.6, 0, .3, .1))
moms<-list(c(-1, 1, 0, 1), c(0, 3, 0, 2))

#generate Poisson, ordinal, and continuous data
cmat.star <- find.cor.mat.star(cor.mat = .8 * diag(6) + .2,
                               no.pois = length(lambdas),
                               no.ord = length(mps),
                               no.nonn = length(moms),
                               pois.list = lambdas,
                               ord.list = mps,
                               nonn.list = moms)

mydata <- genPBONN(n,
                   no.pois = length(lambdas),
                   no.ord = length(mps),
                   no.nonn = length(moms),
                   cmat.star = cmat.star,
                   pois.list = lambdas,
                   ord.list = mps,
                   nonn.list = moms)

#set a sample of each variable to missing
mydata<-apply(mydata, 2, function(x) {
  x[sample(1:n, size=n/10)]<-NA
  return(x)
})

mydata<-data.frame(mydata)

#get information for use in function
ord.info<-ordmps(ord.dat=mydata[,c('X3', 'X4')])
nct.info<-nctsum(nct.dat=mydata[,c('X5', 'X6')])
count.info<-countrate(count.dat=mydata[,c('X1', 'X2')])

#extract marginal probabilites, continuous properties, and count rates
mps<-sapply(ord.info, "[[", 2)
nctsum<-sapply(nct.info, "[[", 2)
rates<-sapply(count.info, "[[", 2)

#replace continuous with standardized forms
mydata[,c('X5', 'X6')]<-sapply(nct.info, "[[", 1)[,c('X5', 'X6')]

var.types<-c('C', 'C', 'O', 'O', 'NCT', 'NCT')
```

```
mvn.cmat<-MVN.corr(indat=mydata,
                   var.types=var.types,
                   ord.mps=mps,
                   nct.sum=nctsum,
                   count.rate=rates)
```

MVN.dat *Computation of Normal Scores for Multivariate Data*

Description

This function assigns a normal score to binary and ordinal variables using normal quantiles in this appropriate range dictated by marginal proportions; a normal score to count variables based on the equivalence of CDFs of Poisson and normal distribution in the appropriate range dictated by the rate parameters; and a normal score for each continuous measurement by finding the normal root in the Fleishman equation.

Usage

```
MVN.dat(ord.info=NULL, nct.info=NULL, count.info=NULL)
```

Arguments

ord.info	A list containing binary and ordinal data and corresponding marginal probabilities as packaged in ordmps. Default is NULL.
nct.info	A list containing standardized continuous data and corresponding summary statistics for continuous variables as packaged in nctsum. Default is NULL.
count.info	A list containing count data and corresponding rates as packaged in countrate. Default is NULL.

Value

A matrix containing normal scores for each variable input.

References

Fleishman A.I. (1978). A method for simulating non-normal distributions. *Psychometrika*, **43(4)**, 521-532.

See Also

MI, ordmps, nctsum, countrate

Examples

```
library(PoisBinOrdNonNor)
n<-1e4
lambdas<-list(1)
mps<-list(c(.2, .8))
moms<-list(c(-1, 1, 0, 1))

#generate Poisson, ordinal, and continuous data
cmat.star <- find.cor.mat.star(cor.mat = .8 * diag(3) + .2,
                               no.pois = length(lambdas),
                               no.ord = length(mps),
                               no.nonn = length(moms),
                               pois.list = lambdas,
                               ord.list = mps,
                               nonn.list = moms)

mydata <- genPBONN(n,
                   no.pois = length(lambdas),
                   no.ord = length(mps),
                   no.nonn = length(moms),
                   cmat.star = cmat.star,
                   pois.list = lambdas,
                   ord.list = mps,
                   nonn.list = moms)

#set a sample of each variable to missing
mydata<-apply(mydata, 2, function(x) {
  x[sample(1:n, size=n/10)]<-NA
  return(x)
})

mydata<-data.frame(mydata)

#get information for use in function
count.info<-countrate(count.dat=data.frame(mydata[,c('X1')]))
ord.info<-ordmps(ord.dat=data.frame(mydata[,c('X2')]))
nct.info<-nctsum(nct.dat=data.frame(mydata[,c('X3')]))

mvn.dat<-MVN.dat(ord.info=ord.info,
                 nct.info=nct.info,
                 count.info=count.info) #outputs in order of continuous, ordinal, count
```

---

nctsum *Summary and Standardization of Continuous Data*

Description

This function calculates mean, variance, skewness, excess kurtosis, and Fleishman coefficients for continuous data and also standardizes each variable.

Usage

```
nctsum(nct.dat)
```

Arguments

nct.dat	A data frame consisting of continuous variables.

Value

A list of length ncol(nct.dat) containing the standardized data and summary statistics for each variable in nct.dat.

References

Fleishman A.I. (1978). A method for simulating non-normal distributions. *Psychometrika*, **43(4)**, 521-532.

See Also

MI, MVN.corr

Examples

```
library(PoisBinOrdNonNor)
set.seed(123)
n<-1e4

#first four moments for each simulated variable
moms<-list(c(-1, 1, 0, 1), c(0, 3, 0, 2))

#generate continuous data
cmat.star <- find.cor.mat.star(cor.mat = .8 * diag(2) + .2,
                               no.nonn = 2,
                               nonn.list = moms)

nctdata <- genPBONN(n,
                    no.nonn = length(moms),
                    cmat.star = cmat.star,
                    nonn.list = moms)

#set a sample of each variable to missing
nctdata<-apply(nctdata, 2, function(x) {
  x[sample(1:n, size=n/10)]<-NA
  return(x)
})

nctdata<-data.frame(nctdata)
nctinfo<-nctsum(nctdata)
```

ordmps *Computation of Marginal Probabilities for Binary and Ordinal Data*

Description

This function computes the empirical marginal probabilities for binary and ordinal data.

Usage

```
ordmps(ord.dat)
```

Arguments

ord.dat	A data frame consisting of binary and ordinal variables.

Value

A list of length ncol(ord.dat) containing the data and empirical marginal probabilities for each variable in ord.dat.

See Also

MI, MVN.corr

Examples

```
library(PoisBinOrdNonNor)
set.seed(123)
n<-1e4
mps<-list(c(.2, .8), c(.6, 0, .3, .1))

#generate ordinal data
cmat.star <- find.cor.mat.star(cor.mat = .8 * diag(2) + .2,
                               no.ord = length(mps),
                               ord.list = mps)

orddata <- genPBONN(n,
                    no.ord = length(mps),
                    cmat.star = cmat.star,
                    ord.list = mps)

#set a sample of each variable to missing
orddata<-apply(orddata, 2, function(x) {
  x[sample(1:n, size=n/10)]<-NA
  return(x)
})

orddata<-data.frame(orddata)
ordinfo<-ordmps(orddata)
```

trMVN.dat *Transformation of Normal Scores*

Description

This function backtransforms normal scores for ordinal variables using the thresholds determined by the marginal proportions using quantiles of the normal distribution; normal scores for continuous variables by the sum of linear combinations of standard normals using the corresponding Fleishman coefficients; and normal scores for count variables by the inverse cdf matching procedure.

Usage

```
trMVN.dat(indat, ord.mps=NULL, nct.sum=NULL, count.rate=NULL)
```

Arguments

indat	A list of data frames of normal scores to be backtransformed.
ord.mps	A list containing marginal probabilities for binary and ordinal variables as packaged from output in ordmps. Default is NULL.
nct.sum	A matrix containing summary statistics for continuous variables as packaged from output in nctsum. Default is NULL.
count.rate	A vector containing rates for count variables as packaged from output in countrate. Default is NULL.

Value

A list containing backtransformed data.

References

Fleishman A.I. (1978). A method for simulating non-normal distributions. *Psychometrika*, **43(4)**, 521-532.

See Also

MI, ordmps, nctsum, countrate

Examples

```
sndat<-data.frame(matrix(rnorm(1e4), ncol=5, nrow=1e4/5))

#ordinal marginal probabilities
m1<-c(0.4, 0.6)
names(m1)<-c(0,1)
m2<-c(0.2, 0.3, 0.5)
names(m2)<-c(0,2,3)
mps<-list(X1=m1, X2=m2)
```

```
#count rates
rates<-c(2, 3)
names(rates)<-c('X3', 'X4')

#continuous
nctsum<-data.frame(X5=c(1, 1, -0.31375, 0.82632, 0.31375, 0.02271)) #Weibull(1,1)
rownames(nctsum)<-c('Mean', 'Variance', 'a', 'b', 'c', 'd')

trdat<-trMVN.dat(indat=list(sndat), ord.mps=mps, nct.sum=nctsum, count.rate=rates)
```

Index

countrate, *2*, 3, *8, 9*, *13*

MI, *2, 3*, 4, *8, 9*, *11–13*
MultiVarMI *(*MultiVarMI-package*)*, 2
MultiVarMI-package, 2
MVN.corr, *2, 3*, *5*, 6, *11, 12*
MVN.dat, *2*, *5*, *8*, 9

nctsum, *2*, *8, 9*, 10, *13*

ordmps, *2*, *8, 9*, 12, *13*

trMVN.dat, *2*, *5*, 13