Package ‘MultiOrd’

May 21, 2016

Type Package
Title Generation of Multivariate Ordinal Variates
Version 2.2
Date 2016-05-20
Author Anup Amatya and Hakan Demirtas
Maintainer Anup Amatya <aamatya@nmsu.edu>
Depends mvtnorm, corpcor, Matrix, psych
Description A method for multivariate ordinal data generation given marginal distributions and correlation matrix based on the methodology proposed by Demirtas (2006).
License GPL-2
LazyLoad yes
NeedsCompilation no
Repository CRAN
Date/Publication 2016-05-21 00:03:38

R topics documented:

MultiOrd-package ................................................................. 2
BinToOrd ................................................................. 2
compute.sigma.star ......................................................... 4
conformity.Check ......................................................... 4
find.binary.prob .......................................................... 5
generate.binary .......................................................... 5
genOrd ................................................................. 6
simBinCorr ............................................................. 7
validation.CorrMat ....................................................... 8
validation.ordPmat ....................................................... 9

Index 11
MultiOrd-package  Generation of multivariate ordinal data.

Description

A package for multivariate ordinal data generation given marginal distributions and correlation matrix based on the methodology proposed by Demirtas (2006).

Details

```
Package: MultiOrd
Type: Package
Version: 2.1
Date: 2014-03-05
License: GPL-2
```

This package can be used to generate multivariate ordinal data. Two main input required are matrix of marginal probabilities of each variable and the correlation matrix of the ordinal variables. Due to the limitation on the magnitude of the binary correlations which depends on the marginal probabilities, off-diagonal entries of ordinal correlation matrix are not free to vary between -1 and 1.

The main function in this package is `genord` which generates the multivariate ordinal data. Another important function is `simBinCorr` which calculates the intermediate binary correlation.

Author(s)

Anup Amatya and Hakan Demirtas

Maintainer: Anup Amatya <aamatya@nmsu.edu>

References


BinToOrd  Converts multivariate binary data to multivariate ordinal data

Description

Converts multivariate binary data to multivariate ordinal data using original ordinal probabilities.
Usage

BinToOrd(prop.vec.bin, ordPmat, Mlocation, bin.data)

Arguments

prop.vec.bin Vector of marginal probabilities. It is usually a first component of the list returned by find.binary.prob
ordPmat Input matrix of ordinal marginal probabilities
Mlocation Vector of locations where dichotomization is done. It is usually a second component of the list returned by find.binary.prob
bin.data Matrix of binary data generated using generate.binary

Details

As a part of the multivariate ordinal data generation, intermediate multivariate binary data are generated. This function converts multivariate binary data generated by generate.binary to the multivariate ordinal data.

Value

y Matrix of multivariate ordinal data
corr Correlation matrix of y

Examples

```r
## Not run: nObs = 1000; nSim = 100000
## Not run: ordPmat1 = matrix( c(0.15,0.70,0.40,
## 0.55,0.10,0.25,
## 0.25,0.10,0.15,
## 0.05,0.10,0.20),4,3,byrow=TRUE)
## End(Not run)

## Not run: \ cmat1= matrix( c(1,0.2,0.2,
## 0.2,1,0.2,
## 0.2,0.2,1),3,3,byrow=TRUE)
## End(Not run)

## Not run: binObj = simBinCorr(ordPmat1, cmat1, nSim)
## Not run: ep0 = generate.binary( nObs, binObj$pvec, binObj$del.next)
## Not run: Mydata= BinToOrd(binObj$pvec, ordPmat1, binObj$Mlocation, ep0)
```
compute.sigma.star  Computes the tetrachoric correlation matrix. If it is non-positive definite, a nearest positive definite matrix is used.

Description
It computes the tetrachoric correlation matrix using the algorithm described in Emrich and Piedmonte (1991). If the resulting matrix is non-positive definite, a nearest positive definite matrix is returned and the warning message will be printed.

Usage
compute.sigma.star(prop.vec.bin, corr.mat)

Arguments
prop.vec.bin   Vector of marginal probabilities
corr.mat      Correlation matrix of the binary data

Value
Tetrachoric correlation matrix

See Also
phi2poly and nearPD

conformity.Check  Checks whether the dimension of marginal probability matrix matches the dimension of correlation matrix.

Description
Checks whether the dimension of marginal probability matrix matches the dimension of correlation matrix.

Usage
conformity.Check(ordPmat, CorrMat)

Arguments
ordPmat        Input matrix of ordinal marginal probabilities
CorrMat        Correlation matrix of the multivariate ordinal data.
find.binary.prob

Collapses the ordinal categories to binary ones

Description
Collapses the ordinal categories to binary ones and counts the number of categories in each variable.

Usage
find.binary.prob(ordPmat)

Arguments
ordPmat Input matrix of ordinal marginal probabilities.

Value
p Vector of binary probabilities
mlocation Vector of points where ordinal variables will be dichotomized

See Also
validation.ordPmat

generate.binary

Generates multivariate binary data given marginal probabilities and correlation.

Description
Usage

generate.binary(no.rows, prop.vec.bin, corr.mat)

Arguments

no.rows Number of observations
prop.vec.bin Vector of binary marginal probabilities
corr.mat correlation matrix of the binary data

Details

It generates multivariate binary data from the marginal probabilities and correlation matrix. It uses the algorithm described in Emrich and Piedmonte (1991). In the process, if the tetrachoric correlation matrix is non-positive definite, a nearest positive definite matrix is used.

Value

data Matrix of multivariate binary data

See Also

nearPD, compute.sigma.star

Examples

```r
## Not run: ordmat1 = matrix( c(PN1U,PN7P,PNTP, PNNU,PN1P,PNRU, PNRU,PN1P,PN1U, PNPU,PN1P,PNRP),T,S,byrow=true)
## End(Not run)
## Not run: cmat1= matrix( c(1,PNR,PNR, PNR,1,PNR, PNR,PNR,1),S,S,byrow=true)
## End(Not run)
## Not run: p=find.binary.prob(ordmat1)
## Not run: finalcorr = simBinCorr(ordmat1, cmat1, nSim=100000)
## Not run: y=generate.binary( 1000, p&p, finalcorr$del.next)
```

---

**genOrd**

*Generates multivariate ordinal data from binary parameters*

**Description**

Generates multivariate ordinal data from the ordinal marginal probabilities and a list returned by the `simBinCorr` function.
**Usage**

```r
genOrd(nObs, ordPmat, binObj)
```

**Arguments**

- `nObs` Number of rows
- `ordPmat` Input matrix of ordinal marginal probabilities
- `binObj` A list returned by the `simBinCorr`

**Details**

It generates multivariate ordinal data. The argument `binObj` must be calculated using `simBinCorr` before executing this function.

**Value**

- `mydata` A list with two components. Two components are a matrix of multivariate ordinal data (y) and its correlation matrix (Corr)

**See Also**

- `simBinCorr`, `BinToOrd`, `generate.binary`

**Examples**

```r
CC not run: ordpmat1 = matrix( c(PN1U,PN7P,PNTP,
PN1U,PN7P,PNPU,
PN7P,PN1P,PN1U,
PN1U,PN1P,PN1U),S,S,byrow=true)
CC end(not run)
CC not run: cmat1= matrix( c(1,PNR,PNR,
PNR,1,PNR,
PNR,PNR,1),S,S,byrow=true)
CC end(not run)
CC not run: binobj=simbincorr(ordpmat1, cmat1, nsim=000000, steps=PNPRU)
CC not run: mydata = genOrd( 0000, ordpmat1, binObj)
```

---

**simBinCorr**

*Calculates intermediate binary correlation matrix*

**Description**

Calculates intermediate binary correlation matrix via simulation.

**Usage**

```r
simBinCorr(ordPmat, CorrMat, nSim, steps = 0.025)
```
**validation.CorrMat**

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ordPmat</td>
<td>Input matrix of ordinal marginal probabilities</td>
</tr>
<tr>
<td>CorrMat</td>
<td>Correlation matrix of the multivariate ordinal data</td>
</tr>
<tr>
<td>nSim</td>
<td>Number of simulations to use to calculate intermediate binary correlation matrix</td>
</tr>
<tr>
<td>steps</td>
<td>Fraction of difference between the current and target matrix to be added in each iteration.</td>
</tr>
</tbody>
</table>

**Value**

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>del.next</td>
<td>Calculated binary correlation matrix</td>
</tr>
<tr>
<td>Mlocation</td>
<td>Cutoff point for converting ordinal probabilities to binary ones.</td>
</tr>
<tr>
<td>pvec</td>
<td>Vector of binary probabilities</td>
</tr>
</tbody>
</table>

**See Also**

`generate.binary`, `BinToOrd`

**Examples**

```r
# Not run: ordPmat1 = matrix( c(PN1U,PN7P,PNTP,
0.15,0.70,0.40,
0.25,0.10,0.25,
0.25,0.10,0.15,
0.05,0.10,0.20),4,3,byrow=TRUE)
# End(Not run)
# Not run: cmat1= matrix( c(1,0.2,0.2,
0.2,1,0.2,
0.2,0.2,1),3,3,byrow=TRUE)
# End(Not run)
# Not run: simBinCorr(ordPmat1, cmat1, nSim=100000, steps = 0.025)
```

---

**validation.CorrMat**

*Validates input correlation matrix*

**Description**

Checks symmetry, positive definiteness, conformity and range of the correlation matrix.

**Usage**

`validation.CorrMat(prop.vec.bin, CorrMat)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>prop.vec.bin</td>
<td>Vector of binary (converted from ordinal) marginal probabilities</td>
</tr>
<tr>
<td>CorrMat</td>
<td>Correlation matrix to be validated</td>
</tr>
</tbody>
</table>
validation.ordPmat

Details
This function checks the correlation matrix for basic properties of correlation matrix, such as symmetry and positive definiteness. In addition it verifies that all the correlations are in valid range for the calculated binary marginal probabilities. Range violation error message indicates that ordinal data with the specified correlations cannot be generated due to distributional constraints.

See Also
find.binary.prob

Description
Validates the range of input matrix of marginal probabilities. It also counts the ordinal categories for each variable.

Usage
validation.ordPmat(ordPmat)

Arguments
ordPmat Matrix of marginal probabilities.

Details
Number of columns of input matrix is the number of variables and each column contains probability of each category within each variable. Any probability with 0 value must be entered at the end of corresponding column. For example if a column contains c(0.3,0.5,0.2,0), then it is assumed that particular variable has only 3 (1, 2 and 3) categories.

Value
J Number of ordinal variables
K Vector of number of categories for each variable

Examples
## Not run:
# 3 outcomes with 3, 4 and 4 categories.
ordPmat1 = matrix( c(0.15,0.70,0.40,
   0.55,0.10,0.25,
   0.30,0.10,0.15,
   0,0.10,0.20),4,3,byrow=TRUE)
validation.ordPmat(ordPmat1)
## End (Not run)
Index

BinToOrd, 2, 7, 8
compute.sigma.star, 4, 6
conformity.Check, 4
find.binary.prob, 3, 5, 9
generate.binary, 3, 5, 7, 8
genOrd, 2, 6
MultiOrd (MultiOrd-package), 2
MultiOrd-package, 2
nearPD, 4, 6
phi2poly, 4
simBinCorr, 2, 6, 7, 7
validation.CorrMat, 8
validation.ordPmat, 5, 9