Package ‘ForImp’

February 19, 2015

Type Package
Title Imputation of Missing Values Through a Forward Imputation Algorithm
Version 1.0.3
Date 2014-11-24
Author Alessandro Barbiero, Pier Alda Ferrari, Giancarlo Manzi
Maintainer Alessandro Barbiero <alessandro.barbiero@unimi.it>
Description Imputation of missing values in datasets of ordinal variables through a forward imputation algorithm
License GPL
LazyLoad yes
Depends homals, sampling, mvtnorm
Repository CRAN
Date/Publication 2015-01-02 17:47:37
NeedsCompilation no

R topics documented:

ForImp-package .................................................. 2
ForImp .......................................................... 3
ld ................................................................. 4
meanimp .......................................................... 5
medianimp ........................................................ 6
missingmat ...................................................... 7
missingmat2 ..................................................... 8
missingness ..................................................... 9
modeimp ......................................................... 10
rancatmat ....................................................... 11
transfmatcat .................................................... 12
vcosw ......................................................... 13

Index 15
ForImp-package

**Forward Imputation**

**Description**

The package contains a function for the imputation of missing values in matrices of ordinal data, called Forward Imputation, and other functions for generating ordinal data or imputing missing values.

**Details**

<table>
<thead>
<tr>
<th>Package:</th>
<th>ForImp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>Package</td>
</tr>
<tr>
<td>Version:</td>
<td>1.0</td>
</tr>
<tr>
<td>Date:</td>
<td>2013-01-30</td>
</tr>
<tr>
<td>License:</td>
<td>GPL</td>
</tr>
<tr>
<td>LazyLoad:</td>
<td>yes</td>
</tr>
</tbody>
</table>

**Author(s)**

Alessandro Barbiero<alessandro.barbiero@unimi.it>,
Giancarlo Manzi<giancarlo.manzi@unimi.it>,
Pier Alda Ferrari<pieralda.ferrari@unimi.it>

Maintainer: Alessandro Barbiero<alessandro.barbiero@unimi.it>

**References**

http://ideas.repec.org/a/eee/csdana/v55y2011i7p2410-2420.html


ForImp  

Forward Imputation procedure

Description

Forward Imputation of missing data

Usage

ForImp(mat, p=2)

Arguments

- **mat**: a matrix/dataframe
- **p**: the parameter for computing the Minkowski distance used in the nearest neighbor procedure for missing value imputation. p can be any positive number (p=2 gives the euclidean distance); if a negative number or Inf is entered, the procedure will use the maximum distance (or supremum norm)

Details

The function implements the Forward Imputation algorithm (see reference) on a matrix of ordinal data with missing values. The algorithm alternates NonLinear Principal Component Analysis (NLPCA) on a subset of the data with no missing data and sequential imputations of missing values by the nearest neighbor method. This sequential process starts from the units with the lowest number of missing values and ends with the units with the highest number of missing values.

Value

the imputed matrix

Author(s)

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

References

http://ideas.repec.org/a/eee/csdana/v55y2011i7p2410-2420.html

See Also

modeimp, medianimp, meanimp

Examples

```r
set.seed(1)
# correlation matrix
sigma<-matrix(c(1,0.5,0.5,0.5,0.5,1,0.5,0.5,0.5,0.5,0.5,0.5,0.5,0.5,1),4,4)
# generate a 500*4 matrix from a multivariate normal
matc<-rmvnorm(n=500, mean=rep(0,4), sigma=sigma)
# transform the numerical values into ordinal categories (Likert scale)
# obtaining matrix mato
mato<-transfmatcat(matc,4)
# set the number of desired missing values
nummissing<-100
# create the random missing values, obtaining matrix mat
mat<-missingmat(mato, nummissing, pattern="r")
# use function \code{forImp} to impute missing values, obtaining matrix mati
mati<-forImp(mat)
# number of correct imputations
nummissing=sum(mati!=mato)
```

### ld

**Listwise deletion**

**Description**

Listwise deletion

**Usage**

```r
ld(mat)
```

**Arguments**

- `mat` a matrix or a dataframe

**Details**

This function implements the listwise deletion on a given dataset, removing all the rows or units containing at least one missing value

**Value**

The matrix/dataframe in input with the rows/units with missing values removed

**Author(s)**

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari
**meanimp**

**See Also**

*meanimp, modeimp, medianimp*

**Examples**

```r
n<-10
m<-4
mat<-matrix(rnorm(n*m),n,m)
mat[c(3,6),1]<-NA
mat[10,2]<-NA
mat
ld(mat)
```

---

**meanimp**  
*Mean imputation*

**Description**

Mean imputation

**Usage**

```r
meanimp(mat)
```

**Arguments**

*mat*  
A numerical matrix

**Details**

The function implements the unconditional mean imputation on a numerical matrix with missing values, substituting to each missing value the arithmetic mean of the corresponding variable

**Value**

The imputed matrix

**Author(s)**

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

**See Also**

*modeimp, medianimp*
Examples

```r
set.seed(1)
n<-10
m<-3
mat<-matrix(rnorm(n*m),n,m)
matm<-mat
matm[1,1]<-NA
matm[2,2:3]<-NA
# matrix with missing values
matm
# imputed matrix
meanimp(matm)
# original matrix with no missing values
mat
```

---

### medianimp

**Median imputation**

#### Description

Median imputation

#### Usage

```r
medianimp(mat)
```

#### Arguments

- `mat`  
  A matrix of ordinal values, ordered according to the Likert scale (1, 2, 3,...)

#### Details

The function implements the median imputation on a matrix of ordinal data with missing values. The function substitutes to each missing value the median of the corresponding variable.

#### Value

The imputed matrix

#### Author(s)

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

#### See Also

`modeimp`, `meanimp`
Examples

```r
set.seed(1)
n<-10
m<-3
mat<-matrix(ceiling(runif(n*m)*4),n,m)
matm<-mat
matm[1,3]<-NA
matm[9:10,1]<-NA
# matrix with missing values
mat
# imputed matrix
medianimp(matm)
# original matrix with no missing values
mat
```

## missingmat

**Random generation of missing values**

### Description
Random generation of missing values in matrices of numerical data or preferably categorical data coded as integers

### Usage

```r
missingmat(mat, nummissing, pattern = "r", nk = 1, p = 0.1, w = 3)
```

### Arguments

- **mat**: A matrix of numerical values
- **nummissing**: number of missing values
- **pattern**: pattern of missing values ("r" random, "l" lowest value, "b" block, "n" not at random)
- **nk**: category
- **p**: percentage of missing values
- **w**: weight for the lowest category in pps sampling (pattern "n")

### Details
The function generates random missing values on a matrix of categorical data according to a specific pattern. "r" is the random pattern, "l" generates a percentage p of missing values on the lowest values of variable nk, "b" generates random blocks of missing values on the group of variables indexed by nk, "n" generates a kind of not at random missing values: specifically, lowest values are more likely to be missing, since they are assigned a weight w (greater than 1, the default is 3) and the values are sampled according to an unequal probability sampling design (pivotal, see the reference for more details).
Value

The original matrix with the desired number of values randomly substituted by missing values

Author(s)

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

References


Examples

```r
set.seed(1)
# correlation matrix
sigma <- matrix(c(1,0.5,0.5,0.5,0.5,1,0.5,0.5,0.5,0.5,1,0.5,0.5,0.5,0.5,0.5,1),4,4)
# generate a n*m matrix from a multivariate normal
n <- 500
m <- 4
matc <- rmvnorm(n, mean = rep(0, m), sigma = sigma)
# transform the numerical values into ordinal categories (Likert scale)
# obtaining matrix mato
mato <- transfmatcat(matc, c(2,3,4,5))
# set the number of desired missing values
nummissing <- 150
# create the random missing values
# random missing values
mato <- missingmat(mato, nummissing, pattern = "r")
matc
# random blocks of missing values on variables 1,2 and 3
matc <- missingmat(mato, nummissing, pattern = "b", nk = c(2,3))
matc
# missing values on lowest category of variable 4
matl <- missingmat(mato, nummissing, pattern = "l", nk = 4, p = 0.1)
matl
# not at random missing values on variable 4
matn <- missingmat(mato, nummissing, pattern = "n", nk = 4, w = 4)
matn
```

**missingmat2**

Random generation of missing values

Description

Random generation of missing values in matrices
Usage

missingmat2(mat, missing)

Arguments

mat a matrix (n rows, m columns)
missing a vector: element i contains the desired number of rows with i missing values (1<=i<=m)

Value

a matrix with the specified pattern of missing values

Author(s)

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

See Also

missingmat, missingness

Examples

mat<-matrix(rnorm(500),100,5)
# if you want 20 rows with 1 missing, 10 rows with 2 missing,
# 4 rows with 3 missing, 1 row with 4 missing
missing<-c(20,10,4,1)
matm<-missingmat2(mat, missing)
matm
# check that the function works
missingness(matm)

---

missingness           Missing values

Description

Summary for the missing values in a matrix

Usage

missingness(mat)

Arguments

mat a matrix/dataframe with missing values
Details

The function provides a summary for the missing values in a matrix (units for variables)

Value

number_of_missing_values
Total number of missing values in the matrix

missing_values_per_unit
Number of units with a certain number of missing values

missing_values_per_variable
Number of missing values for each variable

Author(s)

Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

Examples

```r
n<-100
m<-3
mat<-matrix(rnorm(n*m),n,m)
nummissings<-50
index<-sample(n*m,nummissings,replace=FALSE)
mat[index]<-NA
missingness(mat)
```

---

Usage

```r
modeimp(mat)
```

Arguments

mat A matrix of categorical or ordinal values, coded as integer values (1, 2, 3, ...)

Details

The function implements the mode imputation on a matrix of categorical or ordinal data with missing values. The function substitutes to each missing value the mode of the corresponding variable.

Value

The imputed matrix
Author(s)
Alessandro barbiero, Giancarlo Manzi, Pier Alda Ferrari

See Also
medianimp, modeimp

Examples

```r
set.seed(1)
n<-10
m<-3
mat<-matrix(ceiling(runif(n*m)*4),n,m)
matm<-mat
matm[1,3]<-NA
matm[9:10,1]<-NA
# matrix with missing values
matm
# imputed matrix
modeimp(matm)
# original matrix with no missing values
mat
```

rancatmat Generating a random matrix of ordinal variables

Description
The function generates a random matrix of integer (ordinal) variables, with independent and uniform marginal distributions

Usage
rancatmat(n, m, cat = 3)

Arguments

- `n` number of rows/units
- `m` number of columns, variables
- `cat` number of categories for each variable

Details
The function generates a random matrix of integer (ordinal) variables (coded with 1, 2, 3...), with independent and uniform marginal distributions

Value

a matrix of ordinal values
transfmatcat

**Author(s)**
Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari

**See Also**
transfmatcat

**Examples**
```r
n<-500
m<-3
mat<-rancatmat(n,m,c(3,4,5))
# let's check the marginal distributions...
apply(mat,2,tabulate)
#... should be "quite" uniform
```

---

**transfmatcat**

*Transforming a matrix of continuous values into a matrix of ordinal values*

**Description**
The function transforms a matrix of continuous numerical values into a matrix of integer (ordinal) values, with uniform marginal distributions and the desired number of categories.

**Usage**
```r
transfmatcat(mat, cat = 3)
```

**Arguments**
- `mat`: a matrix or a dataframe
- `cat`: the number of categories, one for each column/variable of the matrix/dataframe

**Details**
The function converts the matrix in input, containing continuous numerical values, into a matrix of ordinal values (1,2,3,... i.e.: Likert scale) according to the cat-1 normal quantiles corresponding to each variable (column) of mat.

**Value**
the matrix of ordinal values

**Author(s)**
Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari
vcosw

References


See Also
rancatmat

Examples
# generate a 40*3 matrix from a multivariate normal r.v.
# whose independent components have mean 10 and standard deviation 4
mat<-matrix(rnorm(40,3),10,4)
# transform the matrix of normal data into a matrix of ordinal data
transfmatcat(mat, cat=c(2,3,4,3))

vcosw   Cosine of the angle between two vectors

Description
The function calculates the cosine of the angle between two vectors, defined as the inner product of the vectors divided by the product of their euclidean norms

Usage
vcosw(v, w)

Arguments
v      a vector
w      a vector, of the same length of v

Value
The cosine of the angle between the two vectors

Author(s)
Alessandro Barbiero, Giancarlo Manzi, Pier Alda Ferrari
See Also


Examples

```r
a<-1:10
b<-2:11
cosw(a,b)
#
e<-c(1,2,3)
f<-c(3,-3,1)
cosw(e,f)
# e and f are orthogonal vectors!
```
## Index

*Topic **datagen**

- ForImp, 3
- ld, 4
- meanimp, 5
- medianimp, 6
- missingmat, 7
- missingmat2, 8
- missingness, 9
- modeimp, 10
- rancatmat, 11
- transfmatcat, 12
- vcosw, 13

*Topic **multivariate**

- ForImp, 3
- ld, 4
- meanimp, 5
- medianimp, 6
- missingmat, 7
- missingmat2, 8
- missingness, 9
- modeimp, 10
- rancatmat, 11
- transfmatcat, 12
- vcosw, 13

*Topic **package**

- ForImp-package, 2

- ForImp, 3
- ForImp-package, 2

- ld, 4

- meanimp, 4, 5, 5, 6
- medianimp, 4, 5, 6, 11
- missingmat, 7, 9
- missingmat2, 8
- missingness, 9, 9
- modeimp, 4–6, 10, 11

- rancatmat, 11, 13

transfmatcat, 12, 12
vcosw, 13