Package ‘mimi’

March 7, 2019

Type Package
Title Main Effects and Interactions in Mixed and Incomplete Data
Version 0.2.0
Author Geneviève Robin
Maintainer Genevieve Robin <genevieve.robin@polytechnique.edu>
Description Generalized low-rank models for mixed and incomplete data frames. The main function may be used for dimensionality reduction of imputation of numeric, binary and count data (simultaneously). Main effects such as column means, group effects, or effects of row-column side information (e.g. user/item attributes in recommendation system) may also be modelled in addition to the low-rank model. Geneviève Robin, Olga Klopp, Julie Josse, Éric Moulines, Robert Tibshirani (2018) <arXiv:1806.09734>.
Depends R (>= 2.10)
License GPL-3
Imports glmnet, softImpute, stats, FactoMineR, parallel, doParallel, foreach, data.table, rARPACK
Encoding UTF-8
LazyData true
RoxygenNote 6.1.0
Suggests knitr, rmarkdown
NeedsCompilation no
Repository CRAN
Date/Publication 2019-03-07 06:20:04 UTC

R topics documented:

acs2016 ................................................................. 2
covmat ................................................................. 3
cv.mimi ................................................................. 3
mimi ................................................................. 5

Index 7
Excerpt of the 2016 Public Use American Census Survey (Alabama only)

Description
A dataset containing answers of 24614 Alabama households to 20 questions

Usage
acs2016

Format
survey A data frame with 24614 rows and 20 columns:

- **NP** Number of persons in household
- **ACCESS** Access to the internet. 1 yes 0 no.
- **AGS** Sales of agriculture products ($, yearly)
- **BATH** Bathtub or shower. 0 yes 1 no.
- **BDSP** Number of bedrooms in household.
- **BROADBND** Cellular data plan for a smartphone or other mobile device 1 yes 2 no
- **COMPOThX** Other computer equipment. 1 yes 2 no
- **CONP** Condo fee ($, monthly)
- **ELEP** Electricity ($, monthly)
- **FS** Food Stamps. 0 no 1 yes
- **FULP** Fuel cost ($, yearly)
- **GASP** Gas ($, monthly)
- **MHP** Mobile home costs $, yearly
- **REFR** Refrigerator. 1 yes, 2 no.
- **RMSP** Number of rooms in household
- **RWAT** Hot and cold running water. 1 yes 2 no
- **SATELLITE** Satellite internet service. 1 yes 2 no.
- **WATP** Water ($, yearly)
- **FFINCP** Family income allocation flag (past 12 months) 0 No 1 yes.

Source
https://factfinder.census.gov/faces/nav/jsf/pages/searchresults.xhtml?refresh=1
covmat

construct covariate matrix (predictor matrix) in the right format for input to the mimi or cv.mimi functions from tables of attributes about the rows or columns of data frames.

Description

construct covariate matrix (predictor matrix) in the right format for input to the mimi or cv.mimi functions from tables of attributes about the rows or columns of data frames.

Usage

covmat(n, p, R = NULL, C = NULL, E = NULL, center = T)

Arguments

n  number of rows
p  number of columns
R  nxK1 matrix of row covariates
C  nxK2 matrix of column covariates
E  (n+p)xK3 matrix of row-column covariates
center boolean indicating whether the returned covariate matrix should be centered (for identifiability)

Value

the joint product of R and C column-binded with E, a (np)x(K1+K2+K3) matrix in order row1col1,row2col1,...,rowncol1, row1col2, row2col2,...,rowncolp

Examples

R <- matrix(rnorm(10), 5)
C <- matrix(rnorm(9), 3)
covs <- covmat(5,3,R,C)

cv.mimi

selection of the regularization parameters (lambda1 and lambda2) of the mimi function by cross-validation

Description

selection of the regularization parameters (lambda1 and lambda2) of the mimi function by cross-validation
Usage

cv.mimi(y, model = c("low-rank", "covariates"), var.type, x = NULL, 
groups = NULL, N = 5, algo = c("mcgd", "bcgd"), thresh = 1e-05, 
maxit = 100, max.rank = NULL, trace.it = F, parallel = F, 
len = 15)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>[matrix, data.frame] incomplete and mixed data frame (nxp)</td>
</tr>
<tr>
<td>model</td>
<td>either one of &quot;groups&quot;, &quot;covariates&quot; or &quot;low-rank&quot;, indicating which model should be fitted</td>
</tr>
<tr>
<td>var.type</td>
<td>vector of length p indicating types of y columns (gaussian, binomial, poisson)</td>
</tr>
<tr>
<td>x</td>
<td>[matrix, data.frame] covariate matrix (npxq)</td>
</tr>
<tr>
<td>groups</td>
<td>factor of length n indicating groups (optional)</td>
</tr>
<tr>
<td>N</td>
<td>[integer] number of cross-validation folds</td>
</tr>
<tr>
<td>algo</td>
<td>type of algorithm to use, either one of &quot;bcgd&quot; (small dimensions, gaussian and binomial variables) or &quot;mcgd&quot; (large dimensions, poisson variables)</td>
</tr>
<tr>
<td>thresh</td>
<td>[positive number] convergence threshold, default is 1e-5</td>
</tr>
<tr>
<td>maxit</td>
<td>[integer] maximum number of iterations, default is 100</td>
</tr>
<tr>
<td>max.rank</td>
<td>[integer] maximum rank of interaction matrix, default is 2</td>
</tr>
<tr>
<td>trace.it</td>
<td>[boolean] whether information about convergence should be printed</td>
</tr>
<tr>
<td>parallel</td>
<td>[boolean] whether the N-fold cross-validation should be parallelized, default value is TRUE</td>
</tr>
<tr>
<td>len</td>
<td>[integer] the size of the grid</td>
</tr>
</tbody>
</table>

Value

A list with the following elements

- `lambda1`: regularization parameter estimated by cross-validation for nuclear norm penalty (interaction matrix)
- `lambda2`: regularization parameter estimated by cross-validation for l1 norm penalty (main effects)
- `errors`: a table containing the prediction errors for all pairs of parameters
mimi

**Description**

main function: low-rank models to analyze and impute mixed and incomplete data frames with numeric, binary and discrete variables, and missing values

**Usage**

```r
mimi(y, model = c("low-rank", "multilevel", "covariates"), x = NULL,
     groups = NULL, var.type = c("gaussian", "binomial", "poisson"),
     lambda1, lambda2, algo = c("mcgd", "bcgd"), maxit = 100,
     alpha0 = NULL, theta0 = NULL, thresh = 1e-05, trace.it = F,
     max.rank = NULL)
```

**Arguments**

- `y`: n xp matrix of observations
- `model`: either one of "groups", "covariates" or "low-rank", indicating which model should be fitted
- `x`: (np)xN matrix of covariates (optional)
- `groups`: factor of length n indicating groups (optional)
- `var.type`: vector of length p indicating the data types of the columns of y (gaussian, binomial or poisson)
- `lambda1`: positive number regularization parameter for nuclear norm penalty
- `lambda2`: positive number regularization parameter for l1 norm penalty
- `algo`: type of algorithm to use, either one of "bcgd" (small dimensions, gaussian and binomial variables) or "mcgd" (large dimensions, poisson variables)
- `maxit`: integer maximum number of iterations
- `alpha0`: vector of length N: initial value of regression parameter (optional)
- `theta0`: matrix of size nxp: initial value of interactions (optional)
- `thresh`: positive number, convergence criterion
- `trace.it`: boolean indicating whether convergence information should be printed
- `max.rank`: integer, maximum rank of interaction matrix theta

**Value**

A list with the following elements

- `alpha`: vector of main effects
- `theta`: interaction matrix
Examples

```r
n = 6; p = 2
y1 <- matrix(rnorm(mean = 0, n * p), nrow = n)
y2 <- matrix(rnorm(mean = 0, n * p), nrow = n)
y3 <- matrix(rnorm(mean = 2, n * p), nrow = n)
y <- cbind(matrix(rnorm(mean = c(y1), n * p), nrow = n),
          matrix(rbinom(n * p, prob = c(exp(y2)/(1+exp(y2))), size = 1), nrow = n),
          matrix(rpois(n * p, lambda = c(exp(y3))), nrow = n))
varNtype <- c(rep("gaussian", p), rep("binomial", p), rep("poisson", p))
idx_na <- sample(1:(3 * n * p), size = round(0.01 * 3 * n * p))
y[idx_na] <- NA
res <- mimi(y, model = "low-rank", var.type = varNtype, lambda1 = 1, maxit=5)
```
Index

*Topic datasets
  acs2016, 2

acs2016, 2

covmat, 3
cv.mimi, 3

mimi, 5