Package 'BMLCimpute'

May 26, 2018

Type Package

Title BMLCimpute: Bayesian Multilevel Latent Class Models for the Multiple Imputation of Nested Categorical Data

Version 0.1

Date 2018-05-24

Author Davide Vidotto

Maintainer Davide Vidotto <d.vidotto@uvt.nl>

Description A package for the multiple imputation of single-level and nested categorical data by means of Bayesian Multilevel Latent Class models.

License GPL (>= 2)

Imports Rcpp (>= 0.12.5)

LinkingTo Rcpp

Archs x64

R topics documented:

`BMLCimpute-package` *BMLCimpute: Multilevel Latent Class Models for the Multiple Imputation of Nested Categorical Data*

Description

BMLCimpute performs multiple imputation of nested categorical data by means of Bayesian Multilevel Latent Class models.

Details

Through the function multilevelLCMI the routine performs multiple imputation of single- or multi-level categorical data via single- or multi-level latent class (mixture) models. Data should be first processed with the function convData; the resulting list is then passed as input to the functions multilevelLCMI in order to perform the imputations. Complete datasets are obtained via the function compData.

Author(s)

Davide Vidotto

Maintainer: Davide Vidotto <d.vidotto@uvt.nl>

References

Vidotto, Vermunt, van Deun (in press). Multilevel Latent Class models for the Multiple imputation of Nested Categorical Data.

See Also

Rcpp package

Examples

```
library(BMLCimpute)
data(simul_incomplete)

# PreProcess the Data
cd <- convData(simul_incomplete, GID = 1, UID = 2, var2 = 8:12)

# Model Selection
mmLC<- multilevelLCMI( convData = cd, L = 10, K = 15, it1 = 1000, it2 = 3000, it3 = 100, it.print = 250, v = 10,
 I = 0, pri2 = 1 / 10, pri1 = 1 / 15, priresp = 0.01, priresp2 = 0.01, random = TRUE, estimates = FALSE,
 count = TRUE, plot.loglik = FALSE, prec = 3, scale = 1.0)

# Select posterior maxima of the number of classes for the imputations (Other alternatives are possible,
such as posterior modes or posterior quantiles)
L = max(which(mmLC[[12]] != 0))
K = max(apply(mmLC[[13]], 1, function(x) max(which( x != 0))), na.rm = TRUE)

# Impute
mmLC<- multilevelLCMI( convData = cd, L = L, K = K, it1 = 2000, it2 = 4000, it3 = 100, it.print = 250, v = 10,
I = 5, pri2 = 500, pri1 = 50, priresp = 0.01, priresp2 = 0.01, random = TRUE, estimates = FALSE,
count = TRUE, plot.loglik = TRUE, prec = 4, scale = 1.0)

# Obtain the dataset completed with the first set of imputations (ind = 1)
complete_data = compData( convData = cd, implev1 = mmLC[[1]], implev2 = mmLC[[2]], ind = 1 )
```

compData *compData*

Description

Plug the imputations obtained with multilevelLCMI into the original dataset, in order to obtain a complete dataset.

Usage

```
compData(convData,implev1,implev2,ind)
```

Arguments

convData	Ouptut list produced by the 'convData' function.
implev1	The set of imputations for the level-1 variables provided by the 'multilevelLCMI' function. It corresponds to the first element of the list returned by 'multilevelLCMI'.
implev2	The set of imputations for the level-2 variables (when present) provided by the 'multilevelLCMI' function. It corresponds to the second element of the list returned by 'multilevelLCMI'. Default to NULL.
ind	The imputation index; an integer value that ranges from 1 to M, where M is the number of imputations (specified as I in the multilevelLCMI input.

Details

This function takes a 'convData' list, the imputations provided by 'multilevelLCMI' and the imputation index (ind is an integer which can ragne from 1 to M, where M is the number of imputation sets) and returns the completed dataset.

Value

dataset	The imputed dataset.

convData *convData*

Description

This function takes a categorical dataset as input (categories can be denoted by numbers) and returns a list of objects that will be used by the 'multilevelLCMI' function to perform the imputations.

Usage

```
# Dataset called dat the group-ID is in the first column and the level-2 variables are in the
5th - 10th columns:
convData(dat, GID=1, UID=NULL, var2=5:10)
```

Arguments

dat	Raw (categorical) dataset with missing data. The GID and UID variables, if passed to the function, must be in the first two columns of the dataset.
GID	Group Indicator (expressed as column number corresponding to the group ID in the dataset). It can be omitted in single-level datasets.
UID	Lower-level unit indicator (expressed as column number corresponding to the unit ID in the dataset). Optional.
var2	Higher-level variables (expressed as a vector of column numbers in the dataset corresponding to the variables measured at the higher levels). Optional.

Details

Convert a raw categorical dataset into one ready to be imputed with the multilevelLCMI function. In particular, the function will transform factor variables into numeric ones, in which numbers denote a different category. A coding list is returned along with the converted dataset.

Value

convDat	The converted dataset.
codLev1	List containing the new (and original) scores which will be used for the imputations (Level-1 variables).
nCatLev1	Vector containing the number of categories observed for each variable (Level-1 variables).
codLev2	List containing the new (and original) scores which will be used for the imputations (Level-2 variables).
nCatLev2	Vector containing the number of categories observed for each variable (Level-2 variables).
GroupIDs	Matrix containing original and new Group ID's.
GID	The column Group ID number (as entered in the input).
UID	The column Unit ID number (as entered in the input).
var2	The column numbers for level-2 variables (as entered in the input).
doVar2	Boolean. Shall the BMLC model impute variables at level-2? (Result of !is.null(var2)).
namesLev1	Vector of variable names (level-1 variables).
namesLev2	Vector of variable names (level-2 variables).
GroupName	Group ID variable name.
CaseName	Unit ID variable name.
caseID	Unit ID vector (re-permuted).
sort_	Vector containing the original permutation of the dataset rows.

multilevelLCMI *multilevelLCMI*

Description

Perform single- and multi-level multiple imputation of categorical data through single/multi level Bayesian Latent Class analysis.

Usage

```
multilevelLCMI(convData,L,K,it1,it2,it3,it.print,v,I=5,pri2=1.0,pri1=1.0,priresp=1.0,
priresp2=1.0,random=TRUE,estimates=TRUE,count=FALSE,plot.loglik=FALSE,
prec=4,scale=1.0)
```

Arguments

convData	Dataset produced as output by the 'convData' function.
L	Number of higher-level mixture components. When L=1, single-level Latent Class multiple imputation is performed.
K	Number of Latent Classes at the lower-level.
it1	Number of Gibbs sampler iterations for the burn-in (must be larger than 0).
it2	Number of Gibbs sampler iterations for the imputations.
it3	Every it3 iterations, the sampler stores new parameter estimates for model estimation. Meaningful only when estimates=TRUE.
it.print	Every it.print iterations, the state of the Gibbs sampler is screen-printed.
v	The Gibbs sampler will produce the first set of imputations at the iteration number (it1+V), where V~Unif(1,v). Subsequent imputations are automatically spaced from each others across the remaining iterations, so that the last imputation (imputation I) occurs at the iteration number (it1+it2).
I	Number of imputations to be performed.
pri2	Hyperparameter value for the higher-level mixture probabilities. Default to 1.
pri1	Hyperparameter value for the lower-level mixture probabilities. Default to 1.
priresp	Hyperparameter value for the lower-level conditional response probabilities. Default to 1.
priresp2	Hyperparameter value for the higher-level conditional response probabilities. Default to 1.
random	Logical. Should the model parameters be initialized at random values? If TRUE, parameters are initialized through draws from uniform Dirichlet distributions. If FALSE, parameters are initialized to be equal to 1/D, with D the number of categories in the (observed/latent) variable of interest.
estimates	Logical. If TRUE, the function returns the posterior means of the model parameters. Default to TRUE.
count	Logical. Should the output include the posterior distribution of L and K? (only K if L=1)
plot.loglik	Logical. Should the output include a traceplot of the log-likelihood ratios obtained through the Gibbs sampler iterations? Helpful for assessing convergence.

prec	When estimates=TRUE, prec defines the number of digits with which the estimates are returned.
scale	Re-scale the log-likelihood value by a factor equal to 'scale'; this parameter is useful to avoid underflow in the calculation of the log-likelihood (which can occur in large datasets) and consequently to prevent error messages for the visualization of the log-likelihood traceplot. This parameter is meaningful only when 'plot.loglik' is set to TRUE. Default value equal to 1.0.

Details

Function for performing Multiple Imputation with the Bayesian Multilevel Latent Class Model. The model takes as input the list produced by the 'convData' function, in which the dataset converted and prepared for the imputations is present, along with other parameters specified by the user (e.g., number of latent classes and specification of the prior distribution hyperparameters). The function can also offer (when the corresponding boolean parameter is activated) a graphical representation of the posterior distribution of the number of occupied classes during the Gibbs sampler iterations. In this way, 'multilevelLCMI' can also perform model selection in a pre-imputation stage. Symmetric Dirichlet priors are used.

Value

imp	Set of imputations for the level-1 variables and units.
imp2	Set of imputations for the level-2 variables and units.
piL	osterior means of the level-2 class probabilities. Calculated only if estimates = TRUE.
piLses	Posterior standard deviations of the level-2 class probabilities. Calculated only if estimates = TRUE.
piK	Posterior means of the level-1 class probabilities. Calculated only if estimates = TRUE.
piKses	Posterior standard deviations of the level-2 class probabilities. Calculated only if estimates = TRUE.
picondlev1	Posterior means of the level-1 conditional probabilities. Calculated only if estimates = TRUE.
picondlev1ses	Posterior standard deviations of the level-1 conditional probabilities. Calculated only if estimates = TRUE.
picondlev2	Posterior means of the level-2 conditional probabilities. Calculated only if estimates = TRUE.
picondlev2ses	Posterior standard deviations of the level-2 conditional probabilities. Calculated only if estimates = TRUE.
DIC	DIC index for the BMLC model. Calculated only if estimates = TRUE.
freqL	Posterior distribution of the number of latent classes at level-2. Calculated only if count is set to TRUE.
freqK	Posterior distribution of the number of latent classes at level-1. Calculated only if count is set to TRUE.
time	Running time of the Gibbs sampler iterations.

Author(s)

Davide Vidotto (Tilburg University)

References

Vidotto, Vermunt, van Deun. Multilevel Latent Class models for the Multiple imputation of Nested Categorical Data.

See Also

Rcpp.

`simul` *Toy multilevel categorical dataset*

Description

The dataset contains 1000 level-1 artificial observations grouped into 200 level-2 groups. The dataset consists of: one identifier for the group, one identifier for the units, six level-1 binary variables (5 predictors and 1 response), five level-2 binary variables. The dataset was generated following the simulation study in Vidotto, Vermunt, van Deun (in press) : Multilevel Latent Class models for the Multiple imputation of Nested Categorical Data.

Usage

```
data(simul)
```

Format

A data frame with 13 columns and 1,000 rows from 200 groups.

[,1]	GroupID	numeric	Level-2 unit Identifier
[,2]	UnitID	numeric	Area Identifier
[,3]	X1,...,X5	binary	Level-1 predictors
[,4]	Z1,...,Z5	binary	Level-2 predictors
[,5]	Y	binary	Response Variable

References

Vidotto, Vermunt, van Deun (in press). Multilevel Latent Class models for the Multiple imputation of Nested Categorical Data.

`simul_incomplete` *Toy multilevel categorical dataset with missing entries*

Description

The dataset contains 1000 level-1 artificial observations grouped into 200 level-2 groups. The dataset consists of: one identifier for the group, one identifier for the units, six level-1 binary variables (5 predictors and 1 response), five level-2 binary variables. The level-1 variables X1, X2, X5 and the level-2 variables Z1 and Z3 have missing observations, generated through a MAR mechanism. The dataset was generated following the simulation study in Vidotto, Vermunt, van Deun (in press) : Multilevel Latent Class models for the Multiple imputation of Nested Categorical Data.

Usage

```
data(simul)
```

Format

A data frame with 13 columns and 1,000 rows from 200 groups.

[,1]	GroupID	numeric	Level-2 unit Identifier
[,2]	UnitID	numeric	Area Identifier
[,3]	X1,...,X5	binary	Level-1 predictors
[,4]	Z1,...,Z5	binary	Level-2 predictors
[,5]	Y	binary	Response Variable

References

Vidotto, Vermunt, van Deun (in press). Multilevel Latent Class models for the Multiple imputation of Nested Categorical Data.

Index