Package 'NPBayesImputeCat'

January 14, 2021

Type Package

Title Non-Parametric Bayesian Multiple Imputation for Categorical Data

Version 0.3

Date 2020-12-21

Author Quanli Wang, Daniel Manrique-Vallier, Jerome P. Reiter and Jingchen Hu

Maintainer Jingchen Hu <jingchen.monika.hu@gmail.com>

Description These routines create multiple imputations of missing at random categorical data, and create multiply imputed synthesis of categorical data, with or without structural zeros. Imputations and syntheses are based on Dirichlet process mixtures of multinomial distributions, which is a non-parametric Bayesian modeling approach that allows for flexible joint modeling, described in Manrique-Vallier and Reiter (2014) <doi:10.1080/10618600.2013.844700>.

License GPL (>= 3)

Depends Rcpp (>= 0.10.2),tidyverse

Imports methods, rlang, reshape2, ggplot2,dplyr

LinkingTo Rcpp

RcppModules clcm

NeedsCompilation yes

Repository CRAN

Date/Publication 2021-01-14 21:40:02 UTC

R topics documented:

`NPBayesImputeCat-package` *Bayesian Multiple Imputation for Large-Scale Categorical Data with Structural Zeros*

Description

This package implements a fully Bayesian, joint modeling approach to multiple imputation for categorical data based on latent class models with structural zeros. The idea is to model the implied contingency table of the categorical variables as a mixture of independent multinomial distributions, estimating the mixture distributions nonparametrically with Dirichlet process prior distributions. Mixtures of multinomials can describe arbitrarily complex dependencies and are computationally expedient, so that they are effective general purpose multiple imputation engines. In contrast to other approaches based on loglinear models or chained equations, the mixture models avoid the need to specify (potentially many) models, which can be a very time-consuming task with no guarantee of a theoretically coherent set of models. The package is designed to include for structural zeros, i.e., certain combinations of variables are not possible a priori.

Details

Package:	NPBayesImputeCat
Type:	Package
Version:	0.1
Date:	2014-04-05
License:	GPL(>=3)

Author(s)

Quanli Wang, Daniel Manrique-Vallier, Jerome P. Reiter and Jingchen Hu

Maintainer: Quanli Wang<quanli@stat.duke.edu>

References

Manrique-Vallier, D. and Reiter, J.P. (2013), "Bayesian Estimation of Discrete Multivariate Latent Structure Models with Structural Zeros", JCGS.

Si, Y. and Reiter, J.P. (2013), "Nonparametric Bayesian multiple imputation for incomplete categorical variables in large-scale assessment surveys", Journal of Educational and Behavioral Statistics, 38, 499 - 521

Manrique-Vallier, D. and Reiter, J.P. (2014), "Bayesian Multiple Imputation for Large-Scale Categorical Data with Structural Zeros", Survey Methodology.

Examples

```
require(NPBayesImputeCat)
#Please use NYexample data set for a more realistic example
data('NYMockexample')

#create the model
model <- CreateModel(X,MCZ,10,10000,0.25,0.25,8888)

#run 1 burnins, 2 mcmc iterations and thin every 2 iterations
model$Run(1,2,2,TRUE)

#retrieve parameters from the final iteration
result <- model$snapshot

#convert ImputedX matrix to dataframe, using proper factors/names etc.
ImputedX <- GetDataFrame(result$ImputedX,X)
#View(ImputedX)

#Most exhauststic examples can be found in the demo below
#demo(example_short)
#demo(example)
```

`compute_probs` *Estimating marginal and joint probabilities in imputed or synthetic datasets*

Description

Estimating marginal and joint probabilities in imputed or synthetic datasets

Usage

```
compute_probs(InputData, varlist)
```

Arguments

InputData	a list of imputed or synthetic datasets
varlist	a list of variable names (or combination of names) to evaluate (marginal or joint) probabilities for

Value

Results: a list of marginal and joint probability results after combining rules

---

CreateModel *Create and initialize the Rcpp_Lcm model object*

Description

CreateModel creates and initializes an Rcpp_Lcm Rcpp_Lcm-class object for non-parametric multiple imputation of discrete multivariate categorical data with or without structural zeros.

Usage

```
CreateModel(X, MCZ, K, Nmax, aalpha, balpha,seed)
```

Arguments

X	a data frame with the dataset with missing values. All variables must be unordered factors.
MCZ	a dataframe with the definition of the structural zeros. Placeholder components are represented with NAs. Variables in MCZ must be factors with the same levels as X. Rows do not need to define disjoint regions of the contingency table. See Manrique-Vallier and Reiter (2014) for details of the definition of structural zeros. MCZ should be set to NULL when there are no structure zeros.
K	the maximum number of mixture components.
Nmax	An upper truncation limit for the augmented sample size. This parameter will be ignored(set to 0) when there is no structural zeros.
aalpha	the hyper parameter 'a' for alpha in stick-breaking prior distribution.
balpha	the hyper parameter 'b' for alpha in stick-breaking prior distribution.
seed	the random seed for sampling. When setting to NULL(default), the random seed will be set randomly.

Details

This should be the first function one should call to use the library. The returned model object will be referenced in all subsequent calls.

Value

CreateModel returns an Rcpp_lcm object. The returned model object will be referenced in all subsequent calls.

References

Manrique-Vallier, D. and Reiter, J.P. (2013), "Bayesian Estimation of Discrete Multivariate Latent Structure Models with Structural Zeros", JCGS.

Si, Y. and Reiter, J.P. (2013), "Nonparametric Bayesian multiple imputation for incomplete categorical variables in large-scale assessment surveys", Journal of Educational and Behavioral Statistics, 38, 499 - 521

Manrique-Vallier, D. and Reiter, J.P. (2014), "Bayesian Multiple Imputation for Large-Scale Categorical Data with Structural Zeros", Survey Methodology.

Examples

```
require(NPBayesImputeCat)
#Please use NYexample data set for a more realistic example
data('NYMockexample')

#create the model
model <- CreateModel(X,MCZ,10,10000,0.25,0.25,8888)

#run 1 burnins, 2 mcmc iterations and thin every 2 iterations
model$Run(1,2,2,FALSE)

#retrieve parameters from the final iteration
result <- model$snapshot

#convert ImputedX matrix to dataframe, using proper factors/names etc.
ImputedX <- GetDataFrame(result$ImputedX,X)
#View(ImputedX)
```

`DPMPM_nozeros_imp` *Use DPMPM models to impute missing data where there are no structural zeros*

Description

Use DPMPM models to impute missing data where there are no structural zeros

Usage

```
DPMPM_nozeros_imp(X, nrun, burn, thin, K, aalpha, balpha, m, seed, silent)
```

Arguments

X	data frame for the data containing missing values
nrun	number of mcmc iterations
burn	number of burn-in iterations
thin	thining parameter for outputing iterations
K	number of latent classes
aalpha	the hyperparameters in stick-breaking prior distribution for alpha
balpha	the hyperparameters in stick-breaking prior distribution for alpha
m	number of imputations
seed	choice of random seed
silent	Default to TRUE. Set this parameter to FALSE if more iteration info are to be printed

Value

impdata	m imputed datasets
origdata	original data containing missing values
alpha	save posterior draws of alpha, which can be used to check MCMC convergence
kstar	saved number of occupied mixture components, which can be used to track whether K is large enough

---

DPMPM_nozeros_syn *Use DPMPM models to synthesize data where there are no structural zeros*

Description

Use DPMPM models to synthesize data where there are no structural zeros

Usage

```
DPMPM_nozeros_syn(X, dj, nrun, burn, thin, K, aalpha, balpha, m, vars, seed, silent)
```

Arguments

X	data frame for the original data
dj	a vector recording the number of categories of the variables
nrun	number of mcmc iterations
burn	number of burn-in iterations
thin	thining parameter for outputing iterations
K	number of latent classes

aalpha	the hyperparameters in stick-breaking prior distribution for alpha
balpha	the hyperparameters in stick-breaking prior distribution for alpha
m	number of imputations
vars	the names of variables to be synthesized
seed	choice of random seed
silent	Default to TRUE. Set this parameter to FALSE if more iteration info are to be printed

Value

impdata	m imputed datasets
origdata	original data containing missing values
alpha	save posterior draws of alpha, which can be used to check MCMC convergence
kstar	saved number of occupied mixture components, which can be used to track whether K is large enough

---

`DPMPM_zeros_imp` *Use DPMPM models to impute missing data where there are no structural zeros*

Description

Use DPMPM models to impute missing data where there are no structural zeros

Usage

```
DPMPM_zeros_imp(X, MCZ, Nmax, nrun, burn, thin, K, aalpha, balpha, m, seed, silent)
```

Arguments

X	data frame for the data containing missing values
MCZ	data frame containing the structural zeros definition
Nmax	an upper truncation limit for the augmented sample size
nrun	number of mcmc iterations
burn	number of burn-in iterations
thin	thining parameter for outputing iterations
K	number of latent classes
aalpha	the hyperparameters in stick-breaking prior distribution for alpha
balpha	the hyperparameters in stick-breaking prior distribution for alpha
m	number of imputations
seed	choice of random seed
silent	Default to TRUE. Set this parameter to FALSE if more iteration info are to be printed

Value

impdata	m imputed datasets
origdata	original data containing missing values
alpha	save posterior draws of alpha, which can be used to check MCMC convergence
kstar	saved number of occupied mixture components, which can be used to track whether K is large enough
Nmax	saved posterior draws of the augmented sample size, which can be used to check MCMC convergence

---

fit_GLMs *Fit GLM models for imputed or synthetic datasets*

Description

Fit GLM models for imputed or synthetic datasets

Usage

```
fit_GLMs(InputData, exp)
```

Arguments

InputData	a list of imputed or synthetic datasets
exp	GLM expression (for polr and nnet, those libraries should be loaded first)

Value

Results: a list of GLM results

GetDataFrame *Convert imputed data to a dataframe, using the same setting from original input data.*

Description

This is a utility function to convert the imputed data matrix to a dataframe. This function will be implemented as a RCPP internal function later on.

Usage

```
GetDataFrame(dest, from, cols = 1:NCOL(from))
```

Arguments

dest	the imputed output data matrix.
from	the original input dataframe.
cols	optinal. Always use default for now.

Value

The returned dataframe object for imputed data.

Examples

```
require(NPBayesImputeCat)
#Please use NYexample data set for a more realistic example
data('NYMockexample')

#create the model
model <- CreateModel(X,MCZ,10,10000,0.25,0.25,8888)

#run 1 burnins, 2 mcmc iterations and thin every 2 iterations
model$Run(1,2,2,TRUE)

#retrieve parameters from the final iteration
result <- model$snapshot

#convert ImputedX matrix to dataframe, using proper factors/names etc.
ImputedX <- GetDataFrame(result$ImputedX,X)
#View(ImputedX)
```

GetMCZ *Convert disjointed structrual zeros to a dataframe, using the same setting from original structrual zero data.*

Description

This is a utility function to convert the disjointed structrual zero matrix to a dataframe. This function will be implemented as a RCPP internal function later on.

Usage

```
GetMCZ(dest, from, mcz, cols = 1:NCOL(from))
```

Arguments

dest	the output data matrix for disjointed structrual zeros.
from	the original input dataframe.
mcz	the original input dataframe for structrual zeros.
cols	optinal. Always use default for now.

Value

The returned dataframe object for disjointed structrual zeros.

References

Manrique-Vallier, D. and Reiter, J.P. (2013), "Bayesian Estimation of Discrete Multivariate Latent Structure Models with Structural Zeros", JCGS.

Si, Y. and Reiter, J.P. (2013), "Nonparametric Bayesian multiple imputation for incomplete categorical variables in large-scale assessment surveys", Journal of Educational and Behavioral Statistics, 38, 499 - 521

Manrique-Vallier, D. and Reiter, J.P. (2014), "Bayesian Multiple Imputation for Large-Scale Categorical Data with Structural Zeros", Survey Methodology.

`Lcm` *RCPP implemenation of the library*

Description

Rcpp_Lcm-class

`marginal_compare_all_imp` *Plot estimated marginal probabilities from observed data vs imputed datasets*

Description

Plot estimated marginal probabilities from observed data vs imputed datasets

Usage

```
marginal_compare_all_imp(obsdata, impdata, vars)
```

Arguments

`obsdata`	he observed data
`impdata`	the list of m imputed datasets
`vars`	the variable of interest

Value

`Plot`	the barplot
`Comparison`	a table of marginal probabilies from observed data vs imputed datasets

marginal_compare_all_syn

Plot estimated marginal probabilities from observed data vs synthetic datasets

Description

Plot estimated marginal probabilities from observed data vs synthetic datasets

Usage

```
marginal_compare_all_syn(obsdata, syndata, vars)
```

Arguments

obsdata	the observed data
syndata	the list of m imputed datasets
vars	the variable of interest

Value

Plot	the barplot
Comparison	a table of marginal probabilies from observed data vs imputed datasets

MCZ

Example dataframe for structrual zeros based on the NYMockexample dataset.

Description

Example dataframe for structrual zeros based on the NYMockexample dataset. It contains 8 structural zero cases with 10 variables.

[,1] AGE = 15 and EDUC = 8
[,2] AGE = 16 and VESTAT = 2
[,3] OWNERSHIP = 0 and MORTGAGE = 4
[,4] AGE = 17 and EDUC = 11
[,5] AGE = [36, 50] and EMPSTAT = 0
[,6] AGE > 70 and DISABWRK = 0
[,7] AGE < 15 and EDUC = 10
[,8] OWNERSHIP = 2 and MORTGAGE = 1

pool_estimated_probs *Pool probability estimates from imputed or synthetic datasets*

Description

Pool probability estimates from imputed or synthetic datasets

Usage

```
pool_estimated_probs(ComputeProbsResults, method =
                       c("imputation", "synthesis_full", "synthesis_partial"))
```

Arguments

ComputeProbsResults	output from the compute_probs function
method	choose between "imputation", "synthesis_full", "synthesis_partial"

Value

Results: a list of marginal and joint probability results after combining rules

pool_fitted_GLMs *Pool estimates of fitted GLM models in imputed or synthetic datasets*

Description

Pool estimates of fitted GLM models in imputed or synthetic datasets

Usage

```
pool_fitted_GLMs(GLMResults, method =
                       c("imputation", "synthesis_full", "synthesis_partial"))
```

Arguments

GLMResults	output from the fit_GLMs function
method	choose between "imputation", "synthesis_full", "synthesis_partial"

Value

Results: a list of GLM results after combining rules

Rcpp_Lcm *RCPP implemenation of the library*

Description

Rcpp_Lcm-class

Rcpp_Lcm-class *Class* "Rcpp_Lcm"

Description

This class implements the MCMC sampler for non-parametric imputation of discrete multivariate data described in Manrique-Vallier and Reiter (2014). It provides methods for updating and monitoring the sampler.

Details

Rcpp_lcm objects should be created with CreateModel. Please see the examples in the demo folder for more detailed explanation on model fitting and parameter tracing.

Extends

Class "C++Object", directly.

All reference classes extend and inherit methods from "envRefClass".

Fields

CurrentIteration: the total number of iterations that have been run so far.

EnableTracer: to check tracer status or to enable/disable the tracer.

MCZ: the disjointed structural zero matrix.

snapshot: retrieve a list with the current state of all the parameters in the sampler, including the imputed sample. A call the the "snapshot" method returns a list with the following components:

alpha: the concentration parameter of the stick breaking prior.

k_star: the effective number number of latent classes (mixture components)

Nmis: the size of the augmented sample.

nu: a vector with the mixture weights

z: a matrix with the current latent class assignment of each member of the sample

ImputedX: the current raw imputed dataset. Use GetDataFrame to convert the raw data to a data frame of factors as defined in the input data set.

psi: The conditional multinomial probabilties. A Lmax * K * J array, where Lmax is the maximum number of levels of all discrete factors in the dataset, J is the number of factors in the dataset, and K is the number of latent classes. Since variables might have different numbers of levels, unused entries in the first dimension are filled with NAs to complete Lmax.

`traceable`**:** list of model parameters that can be traced by the tracer.

`traced`**:** list of model parameters that are traced.

Methods

`SetTrace(paralist,num_of_iterations)`**:** set parameters to be traced.

paralist: a list of parameters to be traced.

num_of_iterations: the maximum number of traced iterations.

`Run(burnin, iter, thinning,silent)`**:** run MCMC iterations.

burnin: number of burn in iterations.

iter: number of MCMC iterations.

thinning: thinning parameter.

silent: boolean indication if more iteration should be printed.

`Resume()`**:** resume from an interrupted call to run method.

`Parameters(paralist)`**:** retrieve a selected list of model parameters from last MCMC iteration.

paralist: a list of parameters to be traced.

`GetTrace()`**:** retrieve all traced iterations. Returns a list with all the parameters set using the method SetTrace(). See description of snapshotreference method for a description of the parameters.

References

Manrique-Vallier, D. and Reiter, J.P. (2013), "Bayesian Estimation of Discrete Multivariate Latent Structure Models with Structural Zeros", JCGS.

Si, Y. and Reiter, J.P. (2013), "Nonparametric Bayesian multiple imputation for incomplete categorical variables in large-scale assessment surveys", Journal of Educational and Behavioral Statistics, 38, 499 - 521

Manrique-Vallier, D. and Reiter, J.P. (2014), "Bayesian Multiple Imputation for Large-Scale Categorical Data with Structural Zeros", Survey Methodology.

Examples

```
require(NPBayesImputeCat)
#Please use NYexample data set for a more realistic example
data('NYMockexample')

#create the model
model <- CreateModel(X,MCZ,10,10000,0.25,0.25,8888)
```

```
#run 1 burnins, 2 mcmc iterations and thin every 2 iterations
model$Run(1,2,2,TRUE)

#retrieve parameters from the final iteration
result <- model$snapshot

#convert ImputedX matrix to dataframe, using proper factors/names etc.
ImputedX <- GetDataFrame(result$ImputedX,X)
#View(ImputedX)
```

`ss16pusa_ds_MCZ` *Example dataframe for structrual zeros based on the ss16pusa_sample_zeros dataset.*

Description

Example dataframe for structrual zeros based on the ss16pusa_sample_zeros dataset. It contains 8 structural zero cases with 5 variables.

[,1] AGEP = 16 and SCHL = Bachelor's degree
[,2] AGEP = 16 and SCHL = Doctorate degree
[,3] AGEP = 16 and SCHL = Master's degree
[,4] AGEP = 16 and SCHL = Professional degree
[,5] AGEP = 17 and SCHL = Bachelor's degree
[,6] AGEP = 17 and SCHL = Doctorate degree
[,7] AGEP = 17 and SCHL = Master's degree
[,8] AGEP = 17 and SCHL = Professional degree

`ss16pusa_mi_MCZ` *Example dataframe for structrual zeros based on the ss16pusa_sample_zeros dataset.*

Description

Example dataframe for structrual zeros based on the ss16pusa_sample_zeros dataset. It contains 8 structural zero cases with 5 variables.

[,1] AGEP = 16 and SCHL = Bachelor's degree
[,2] AGEP = 16 and SCHL = Doctorate degree
[,3] AGEP = 16 and SCHL = Master's degree
[,4] AGEP = 16 and SCHL = Professional degree
[,5] AGEP = 17 and SCHL = Bachelor's degree
[,6] AGEP = 17 and SCHL = Doctorate degree
[,7] AGEP = 17 and SCHL = Master's degree
[,8] AGEP = 17 and SCHL = Professional degree

ss16pusa_sample_nozeros *Example dataframe for input categorical data without structural zeros (without missing values).*

Description

Example dataframe for input categorical data without structural zeros (without missing values). It contains 1000 observations and 3 variables.

[,1]	MAR	marital status	5 levels
[,2]	SEX	sex	2 levels
[,3]	WKL	When last worked	3 levels

ss16pusa_sample_nozeros_miss *Example dataframe for input categorical data without structural zeros (with missing values).*

Description

Example dataframe for input categorical data without structural zeros (with missing values). It contains 1000 observations and 3 variables.

[,1]	MAR	marital status	5 levels
[,2]	SEX	sex	2 levels
[,3]	WKL	When last worked	3 levels

ss16pusa_sample_zeros *Example dataframe for input categorical data with structural zeros (without missing values).*

Description

Example dataframe for input categorical data with structural zeros (without missing values). It contains 1000 observations and 5 variables.

[,1]	AGEP	age	7 levels
[,2]	MAR	marital status	5 levels
[,3]	SCHL	educational attainment	9 levels

[,4]	SEX	sex	2 levels
[,5]	WKL	When last worked	3 levels

`ss16pusa_sample_zeros_miss` *Example dataframe for input categorical data with structural zeros (with missing values).*

Description

Example dataframe for input categorical data with structural zeros (with missing values). It contains 1000 observations and 5 variables.

[,1]	AGEP	age	7 levels
[,2]	MAR	marital status	5 levels
[,3]	SCHL	educational attainment	9 levels
[,4]	SEX	sex	2 levels
[,5]	WKL	When last worked	3 levels

`UpdateX` *Allow user to update the model with data matrix of same kind.*

Description

Allow user to replace initial matrix with a new data matrix of same size and same number of factors. This is not intended for general use and is only useful for very specific circumstance.

Usage

```
UpdateX(model, X)
```

Arguments

`model`	The Rcpp model object created by the CreateModel function.
`X`	a data frame with the dataset with missing values. All variables must be unordered factors.

X *Example dataframe for input categorical data with missing values based on the NYMockexample dataset.*

Description

Example dataframe for input categorical data with missing values based on the NYMockexample dataset. It contains 2000 observations and 10 variables.

[,1]	OWNERSHIP	ownership of dwelling	3 levels
[,2]	MORTGAGE	mortgate status	4 levels
[,3]	AGE	age	9 levels
[,4]	SEX	sex	2 levels
[,5]	MARST	martial status	6 levels
[,6]	RACESING	single race identification	5 levels
[,7]	EDUC	educational attainment	11 levels
[,8]	EMPSTAT	employment status	4 levels
[,9]	DISABWRK	work disability status	3 levels
[,10]	VESTAT	veteran status	3 levels

Index