Package 'misspi'

October 17, 2023

Type Package

Title Missing Value Imputation in Parallel

Version 0.1.0

Description A framework that boosts the imputation of 'missForest' by Stekhoven, D.J. and Bühlmann, P. (2012) <doi:10.1093/bioinformatics/btr597> by harnessing parallel processing and through the fast Gradient Boosted Decision Trees (GBDT) implementation 'LightGBM' by Ke, Guolin et al.(2017) <https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision>. 'misspi' has the following main advantages:
1. Allows embrassingly parallel imputation on large scale data.
2. Accepts a variety of machine learning models as methods with friendly user portal.
3. Supports multiple initializations methods.
4. Supports early stopping that prohibits unnecessary iterations.

License GPL-2

Encoding UTF-8

LazyData true

Imports lightgbm, doParallel, doSNOW, foreach, ggplot2, glmnet, SIS, plotly

Suggests e1071, neuralnet

RoxygenNote 7.2.3

NeedsCompilation no

Author Zhongli Jiang [aut, cre]

Maintainer Zhongli Jiang <jiang548@purdue.edu>

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2023-10-17 09:50:02 UTC

R topics documented:

`evaliq` *Evaluate the Imputation Quality*

Description

Calculates Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) and Normalized Root Mean Squared Error (NRMSE). It also performs visualization for imputation quality evaluation.

Usage

```
evaliq(x.true, x.impute, plot = TRUE, interactive = FALSE)
```

Arguments

`x.true`	a vector with true values.
`x.impute`	a vector with estimated values.
`plot`	a Boolean that indicates whether to plot or not.
`interactive`	a Boolean that indicates whether to use interactive plot when the plot option is invoked (plot = "TRUE").

Value

rmse root mean squared error.

mae mean absolute error.

nrmse normalized root mean squared error.

Author(s)

Zhongli Jiang <jiang548@purdue.edu>

See Also

`misspi`, `missar`

Examples

```
# A very quick example
n <- 100
x.true <- rnorm(n)
x.est <- x.true
na.idx <- sample(1:n, 20)
x.est[na.idx] <- x.est[na.idx] + rnorm(length(na.idx), sd = 0.1)
```

```
# Default plot
er.eval <- evaliq(x.true[na.idx], x.est[na.idx])

# Interactive plot
er.eval <- evaliq(x.true[na.idx], x.est[na.idx], interactive = TRUE)

# Turn off plot
# All of the three case will return the value of error
er.eval <- evaliq(x.true[na.idx], x.est[na.idx], plot = FALSE)
er.eval



# Real data example
set.seed(0)
data(toxicity, package = "misspi")
toxicity.miss <- missar(toxicity, 0.4, 0.2)
impute.res <- misspi(toxicity.miss)
x.imputed <- impute.res$x.imputed

na.idx <- which(is.na(toxicity.miss))
evaliq(toxicity[na.idx], x.imputed[na.idx])
evaliq(toxicity[na.idx], x.imputed[na.idx], interactive = TRUE)
```

`missar` *Generate Data that is Missing At Random (MAR)*

Description

Simulates missing value at random as NA for a given matrix.

Usage

```
missar(x, miss.rate = 0.2, miss.var = 1)
```

Arguments

`x`	a matrix to be used to fill in missing values as NA.
`miss.rate`	a value of missing rate within the range (0, 1) for variables that contain missing values.
`miss.var`	proportion of variables (columns) that contain missing values.

Value

x a matrix with missing values in "NA".

Author(s)

Zhongli Jiang <jiang548@purdue.edu>

See Also

misspi

Examples

```
set.seed(0)
data(toxicity, package = "misspi")
toxicity.miss <- missar(toxicity, 0.4, 1)
toxicity.miss[1:5, 1:5]
```

---

misspi *Missing Value Imputation in Parallel*

Description

Enables embarrassingly parallel computing for imputation. Some of the advantages include

- Provides fast implementation especially for high dimensional datasets.
- Accepts a variety of machine learning models as methods with friendly user portal.
- Supports multiple initializations.
- Supports early stopping that prohibits unnecessary iterations.

Usage

```
misspi(
  x,
  ncore = NULL,
  init.method = "rf",
  method = "rf",
  earlystopping = TRUE,
  ntree = 100,
  init.ntree = 100,
  viselect = NULL,
  lgb.params = NULL,
  lgb.params0 = NULL,
  model.train = NULL,
  pmm = TRUE,
  nn = 3,
  intcol = NULL,
  maxiter = 10,
  rdiff.thre = 0.01,
  verbose = TRUE,
  progress = TRUE,
  nlassofold = 5,
```

```
  isis = FALSE,
  char = " * ",
  iteration = TRUE,
  ndecimal = NULL,
  ...
)
```

Arguments

x	a matrix of numerical values for imputation, missing value should all be "NA".
ncore	number of cores to use, will be set to the cores detected as default.
init.method	initializing method to fill in the missing value before imputation. Support "rf" for random forest imputation as default, "mean" for mean imputation, "median" for median imputation.
method	method name for the imputation, support "rf" for random forest, "lgb" for lightgbm, "lasso" for LASSO, or "customize" if you want to use your own method.
earlystopping	a Boolean which indicates whether to stop the algorithm if the relative difference stop decreasing, with TRUE as default.
ntree	number of trees to use for imputation when method is "rf" or "gbm".
init.ntree	number of trees to use for initialization when method is "rf"
viselect	the number of variables with highest variable importance calculated from random forest initialization to work on if the value is not NULL. This would only work when init.method is "rf", and method is "rf" or "gbm".
lgb.params	parameters to customize for lightgbm models, could be invoked when method is "rf" or "gbm".
lgb.params0	parameters to customize for initialization using random forest, could be invoked when init.method is "rf".
model.train	machine learning model to be invoked for customizing the imputation. Only invoked when parameter method = "customize". The input model should be able to take y~x for fitting process where y, and x are matrices, also make sure that it could be called using method "predict" for model prediction. You could pass the parameters for the model through the additional arguments ...
pmm	a Boolean which indicated whether to use predictive mean matching.
nn	number of neighbors to use for prediction if predictive mean matching is invoked (pmm is "TRUE").
intcol	a vector of indices of columns that are know to be integer, and will be round to integer in every iteration.
maxiter	maximum number of iterations for imputation.
rdiff.thre	relative difference threshold for determining the imputation convergence.
verbose	a Boolean that indicates whether to print out the intermediate steps verbally.
progress	a Boolean that indicates whether to show the progress bar.
nlassofold	number of folds for cross validation when the method is "lasso".

isis	a Boolean that indicates whether to use isis if the method is "lasso", recommended to use for ultra high dimension.
char	a character to use which also accept unicode for progress bar. For example, u03c, u213c for pi, u2694 for swords, u2605 for star, u2654 for king, u26a1 for thunder, u2708 for plane.
iteration	a Boolean that indicates whether use iterative algorithm.
ndecimal	number of decimals to round for the result, with NULL meaning no intervention.
...	other arguments to be passed to the method.

Value

a list that contains the imputed values, time consumed and number of iterations.

x.imputed the imputed matrix.

time.elapsed time consumed for the algorithm.

niter number of iterations used in the algorithm.

Author(s)

Zhongli Jiang <jiang548@purdue.edu>

References

Azur, M. J., Stuart, E. A., Frangakis, C., & Leaf, P. J. (2011). Multiple imputation by chained equations: what is it and how does it work?. International journal of methods in psychiatric research, 20(1), 40-49.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30.

Stekhoven, D. J., & Bühlmann, P. (2012). MissForest—non-parametric missing value imputation for mixed-type data. Bioinformatics, 28(1), 112-118.

Fan, J., & Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. Journal of the Royal Statistical Society Series B: Statistical Methodology, 70(5), 849-911.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society Series B: Statistical Methodology, 58(1), 267-288.

Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

See Also

missar

Examples

```
# Quick example 1
# Load a small data
```

```
data(iris)
# Keep numerical columns
num.col <- which(sapply(iris, is.numeric))
iris.numeric <- as.matrix(iris[, num.col])
set.seed(0)
iris.miss <- missar(iris.numeric, 0.3, 1)
iris.impute <- misspi(iris.miss)
iris.impute

# Quick example 2
# Load a high dimensional data
data(toxicity, package = "misspi")
set.seed(0)
toxicity.miss <- missar(toxicity, 0.4, 0.2)
toxicity.impute <- misspi(toxicity.miss)
toxicity.impute

# Change cores
iris.impute.5core <- misspi(iris.miss, ncore = 5)

# Change initialization and maximum iterations (no iteration in the example)
iris.impute.mean.5iter <- misspi(iris.miss, init.method = "mean", maxiter = 0)

# Change fun shapes for progress bar
iris.impute.king <- misspi(iris.miss, char = " \u2654")


# Use variable selection
toxicity.impute.vi <- misspi(toxicity.miss, viselect = 128)


# Use different machine learning algorithms as method
# linear model
iris.impute.lm <- misspi(iris.miss, model.train = lm)

# From external packages
# Support Vector Machine (SVM)

library(e1071)
iris.impute.svm.radial <- misspi(iris.miss, model.train = svm)


# Neural Networks

library(neuralnet)
iris.impute.nn <- misspi(iris.miss, model.train = neuralnet)
```

`toxicity` *Toxicity Data*

Description

The data was created by Gul, S., Rahim, F., Isin, S. et al. (2021) doi:10.1038/s41598021979625, downloaded and cleaned from UCI Machine Learning Repository with doi:10.24432/C59313. The toxicity data consists of 171 molecules with 1203 molecule descriptors.

Usage

```
data(toxicity)
```

Format

A matrix with 171 rows and 1203 columns

References

doi:10.1038/s41598021979625 Gul, S., Rahim, F., Isin, S., Yilmaz, F., Ozturk, N., Turkay, M., & Kavakli, I. H. (2021). Structure-based design and classifications of small molecules regulating the circadian rhythm period. Scientific reports, 11(1), 18510.

Index