Package ‘PLORN’

October 12, 2022

Type Package
Title Prediction with Less Overfitting and Robust to Noise
Version 0.1.1
Author Takahiko Koizumi, Kenta Suzuki, Yasunori Ichihashi
Maintainer Takahiko Koizumi <takahiko.koizumi@riken.jp>
Description A method for the quantitative prediction with much predictors. This package provides functions to construct the quantitative prediction model with less overfitting and robust to noise.
Depends R (>= 3.5.0)
License MIT + file LICENSE
Language en-US
Encoding UTF-8
LazyData true
RoxygenNote 7.1.2
Suggests knitr, rmarkdown, testthat (>= 3.0.0)
URL https://github.com/takakoizumi/PLORN
VignetteBuilder knitr
Config/testthat/edition 3
Imports ggplot2, kernlab
NeedsCompilation no
Repository CRAN
Date/Publication 2022-03-21 08:00:11 UTC

R topics documented:

p.clean ................................................................. 2
p.opt ................................................................. 3
p.pca ................................................................. 4
p.rank ................................................................. 5
p.clean

Clean data by eliminating predictors with many missing values

Usage

p.clean(x, missing = 0.1, lowest = 10)

Arguments

x A data matrix (raw: samples, col: predictors).
missing A ratio of missing values in each column allowed to be remained in the data.
lowest The lowest value recognized in the data.

Value

A data matrix (raw: samples, col: qualified predictors)

Author(s)

Takahiko Koizumi

Examples

data(Pinus)
train.raw <- Pinus$train
ncol(train.raw)

train <- p.clean(train.raw)
ncol(train)
**p.opt**  
*Estimate the optimal number of predictors to construct PLORN model*

**Description**

Estimate the optimal number of predictors to construct PLORN model.

**Usage**

```r
p.opt(x, y, range = 5:50, method = "linear", rep = 1)
```

**Arguments**

- **x**: A data matrix (row: samples, col: predictors).
- **y**: A vector of an environment in which the samples were collected.
- **range**: A sequence of numbers of predictors to be tested for MAE calculation (default: 5:50).
- **method**: A string to specify the method of regression for calculating R-squared values. "linear" (default), "quadratic" or "cubic" regression model can be specified.
- **rep**: The number of replications for each case set by range (default: 1).

**Value**

A sample-MAE curve

**Author(s)**

Takahiko Koizumi

**Examples**

```r
data(Pinus)
train <- p.clean(Pinus$train)
target <- Pinus$target
p.opt(train[1:10,], target[1:10], range = 5:15)
```
p.pca

Visualize predictors using principal coordinate analysis

Description

Visualize predictors using principal coordinate analysis

Usage

```r
p.pca(x, y, method = "linear", lower.thr = 0, n.pred = ncol(x), size = 1)
```

Arguments

- `y`: A vector of an environment in which the samples were collected.
- `method`: A string to specify the method of regression for calculating R-squared values. "linear" (default), "quadratic" or "cubic" regression model can be specified.
- `lower.thr`: The lower threshold of R-squared value to be indicated in a PCA plot (default: 0).
- `n.pred`: The number of candidate predictors for PLORN model to be indicated in a PCA plot (default: ncol(x)).
- `size`: The size of symbols in a PCA plot (default: 1).

Value

A PCA plot

Author(s)

Takahiko Koizumi

Examples

```r
data(Pinus)
train <- p.clean(Pinus$train)
target <- Pinus$target
p.pca(train, target)
```
**Description**

Visualize R-squared value distribution in predictor-environment interaction

**Usage**

```r
p.rank(
  x, 
  y, 
  method = "linear", 
  lower.thr = 0, 
  n.pred = ncol(x), 
  upper.xlim = ncol(x)
)
```

**Arguments**

- **x**: A data matrix (row: samples, col: predictors).
- **y**: A vector of an environment in which the samples were collected.
- **method**: A string to specify the method of regression for calculating R-squared values. "linear" (default), "quadratic" or "cubic" regression model can be specified.
- **lower.thr**: The lower threshold of R-squared value to be included in PLORN model (default: 0).
- **n.pred**: The number of predictors to be included in PLORN model (default: ncol(x)).
- **upper.xlim**: The upper limitation of x axis (i.e., the number of predictors) in the resulted figure (default: ncol(x)).

**Value**

A rank order plot

**Author(s)**

Takahiko Koizumi

**Examples**

```r
data(Pinus)
train <- p.clean(Pinus$train)
target <- Pinus$target
train <- p.sort(train, target)
p.rank(train, target)
```
p.sort  

Sort and truncate predictors according to the strength of predictor-environment interaction

Description

Sort and truncate predictors according to the strength of predictor-environment interaction

Usage

p.sort(x, y, method = "linear", n.pred = ncol(x), trunc = 1)

Arguments

x  A data matrix (raw: samples, col: predictors).
y  A vector of an environment in which the samples were collected.
method  A string to specify the method of regression for calculating R-squared values. "linear" (default), "quadratic" or "cubic" regression model can be specified.
n.pred  The number of predictors to be included in PLORN model (default: ncol(x)).
trunc  a threshold to be truncated (default: 1).

Value

A data matrix (raw: samples, col: sorted predictors)

Author(s)

Takahiko Koizumi

Examples

data(Pinus)
train <- p.clean(Pinus$train)
target <- Pinus$target
cor(target, train[, 1])

train <- p.sort(train, target, trunc = 0.5)
cor(target, train[, 1])
**Pinus**

**Transcriptomes of Pinus roots under a Temperature Gradient**

**Description**

This dataset gives the TPM values of 200 selected genes obtained from 60 Pinus root samples (30 samples each for training and test data) under a temperature gradient, generated by RNA-seq.

**Usage**

Pinus

**Details**

A gene expression data matrix of 30 root samples of *P. thunbergii* under five temperature conditions (8, 13, 18, 23, 28 °C) with six biological replicates is in the first element of the list.

A gene expression data matrix of another 30 root samples of *P. thunbergii* under the same condition is in the second one.

Temperature conditions where 30 root samples in each data matrix were generated are in the third one.

Gene expressions are normalized in the TPM value.

**Source**

original (not published)

**References**

original (not published)

---

**plorn**

*Construct and apply the PLORN model with your own data*

**Description**

Construct and apply the PLORN model with your own data

**Usage**

`plorn(x, y, newx = x, method = "linear", lower.thr = 0, n.pred = 0)`
**Arguments**

- `x` A data matrix (row: samples, col: predictors).
- `y` A vector of an environment in which the samples were collected.
- `method` A string to specify the method of regression for calculating R-squared values. "linear" (default), "quadratic" or "cubic" regression model can be specified.
- `lower.thr` The lower threshold of R-squared value to be used in PLORN model (default: 0).
- `n.pred` The number of candidate predictors to be used in PLORN model (default: 30).

**Value**

A vector of the environment in which the samples of newx were collected

**Author(s)**

Takahiko Koizumi

**Examples**

data(Pinus)
train <- p.clean(Pinus$train)
test <- Pinus$test
test <- test[, colnames(train)]
target <- Pinus$target
cor(target, plorn(train, target, newx = test, method = "cubic"))
Index

* dataset
  Pinus, 7

p.clean, 2
p.opt, 3
p.pca, 4
p.rank, 5
p.sort, 6
Pinus, 7
plorn, 7