Package 'PRTree'

January 16, 2024

Type Package

Date 2024-01-12

Title Probabilistic Regression Trees

Version 0.1.0

Depends R (>= 4.2.0)

Description Probabilistic Regression Trees (PRTree). Functions for fitting and predicting PRTree models with some adaptations to handle missing values. The main calculations are performed in 'FORTRAN', resulting in highly efficient algorithms.
This package's implementation is based on the PRTree methodology described in Alkhoury, S.; Devijver, E.; Clausel, M.; Tami, M.; Gaussier, E.; Oppenheim, G. (2020) - ``Smooth And Consistent Probabilistic Regression Trees'' <https://proceedings.neurips.cc/paper_files/paper/2020/file/8289889263db4a40463e3f358bb7c7a1-Paper.pdf>.

License GPL (>= 3)

Encoding UTF-8

NeedsCompilation yes

RoxygenNote 7.2.3

Author Alisson Silva Neimaier [aut, cre]
(<https://orcid.org/0000-0002-7524-0776>),
Taiane Schaedler Prass [aut, ths]
(<https://orcid.org/0000-0003-3136-909X>)

Maintainer Alisson Silva Neimaier <alissonneimaier@hotmail.com>

Repository CRAN

Date/Publication 2024-01-16 13:40:08 UTC

R topics documented:

predict.prtree *Predict method for PRTree*

Description

Predicted values based on a prtree object.

Usage

```
## S3 method for class 'prtree'
predict(object, newdata, ...)
```

Arguments

object	Object of class inheriting from "prtree"
newdata	A matrix with new values for the covariates.
...	further arguments passed to or from other methods.

Value

A list with the following arguments

yhat	The predicted values.
newdata	The matrix with the covariates new values.

pr_tree *Probabilistic Regression Trees (PRTrees)*

Description

Probabilistic Regression Trees (PRTrees)

Usage

```
pr_tree(y, X, sigma_grid = NULL, max_terminal_nodes = 15L, cp = 0.01,
  max_depth = 5L, n_min = 5L, perc_x = 0.1, p_min = 0.05)
```

Arguments

y	a numeric vector corresponding to the dependent variable
X	A numeric vector, matrix or dataframe corresponding to the independent variables, with the same number of observations as y.
sigma_grid	optionally, a numeric vector with candidate values for the parameter σ, to be passed to the grid search algorithm. If NULL, the standard deviations of the columns in X are used. The default is NULL.
max_terminal_nodes	a non-negative integer. The maximum number of regions in the output tree. The default is 15.
cp	a positive numeric value. The complexity parameter. Any split that does not decrease the MSE by a factor of cp will be ignored. The default is 0.01.
max_depth	a non-negative integer. The maximum depth of the decision tree. The depth is defined as the length of the longest path from the root to a leaf. The default is 5.
n_min	a non-negative integer, The minimum number of observations in a final node. The default is 5.
perc_x	a positive numeric value. Given a column of P, the value perc_x is the percentage of rows in this column that must have a probability higher than the threshold p_min for a splitting attempt to be made in the corresponding region. The default is 0.1.
p_min	a positive numeric value. A threshold probability that controls the splitting process. A splitting attempt is made in a given region only when the proportion of rows with probability higher than p_min, in the corresponding column of the matrix P, is equal to perc_x. The default is 0.05.

Value

yhat	the estimated values for y
P	the matrix of probabilities calculated with the observations in X for the returned tree
gamma	the values of the γ_j weights estimated for the returned tree
MSE	the mean squared error calculated for the returned tree
sigma	the σ of the returned tree
nodes_matrix_info	information related to each node of the returned tree
regions	information related to each region of the returned tree

Examples

```
set.seed(1234)
X = matrix(runif(200, 0, 10), ncol = 1)
eps = matrix(rnorm(200, 0, 0.05), ncol = 1)
y =  matrix(cos(X) + eps, ncol = 1)
reg = PRTree::pr_tree(y, X, max_terminal_nodes = 9)
```

```
plot(X[order(X)], reg$yhat[order(X)], xlab = 'x', ylab = 'cos(x)', col = 'blue', type = 'l')
points(X[order(X)], y[order(X)], xlab = 'x', ylab = 'cos(x)', col = 'red')
```

Index