

Package ‘featurefinder’

July 22, 2025

Type Package

Title Feature Finder

Version 1.2

Author Richard Davis [aut, cre]

Maintainer Richard Davis <davisconsulting@gmail.com>

Description Finds features through a detailed analysis of model residuals using rpart classification and regression trees. Scans the residuals of a model across subsets of the data to identify areas where the model differs from the actual data.

Depends R (>= 3.2.0)

License MIT + file LICENSE

LazyData true

Encoding UTF-8

RoxygenNote 7.2.3

Suggests knitr,rmarkdown, png

VignetteBuilder knitr

Imports rpart, rpart.plot, plyr, grDevices

BuildVignettes true

NeedsCompilation no

Repository CRAN

Date/Publication 2024-12-17 15:00:06 UTC

Contents

data	2
findFeatures	2
generateResidualCutoffCode	4
generateTrees	5
getVarAv	5
parseSplits	6
printResiduals	6
saveTree	7

Index	9
-------	---

data	<i>data</i>
------	-------------

Description

Sample data based on dataset EuStockMarkets in the datasets package.

Format

A data frame with 1860 rows and 4 variables

Author(s)

Richard Davis <richard.davis@cba.com.au>

Source

<https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/00Index.html>

Examples

```
data(mycsv)
thismodel=lm(formula=DAX ~ .,data=data)
expectedprob=predict(thismodel,data)
actualprob=data$DAX
residual=actualprob-expectedprob
data=cbind(data,expectedprob, actualprob, residual)
```

findFeatures	<i>findFeatures</i>
--------------	---------------------

Description

Perform analysis of residuals grouped by factor to identify features which explain the target variable

Usage

```
findFeatures(
  OutputPath,
  fcsv,
  ExclusionVars,
  FactorToNumericList,
  treeGenerationMinBucket = 50,
  treeSummaryMinBucket = 20,
  treeSummaryResidualThreshold = 0,
```

```

    treeSummaryResidualMagnitudeThreshold = 0,
    doAllFactors = TRUE,
    maxFactorLevels = 20
  )

```

Arguments

OutputPath	A string containing the location of the input csv file. Results are also stored in this location.
fcsv	A string containing the name of a csv file
ExclusionVars	A string consisting of a list of variable names with double quotes around each variable
FactorToNumericList	A list of variable names as strings
treeGenerationMinBucket	Desired minimum number of data points per leaf (default 50)
treeSummaryMinBucket	Minimum number of data points in each leaf for the summary (default 20)
treeSummaryResidualThreshold	Minimum residual in the summary (default 0 for positive residuals)
treeSummaryResidualMagnitudeThreshold	Minimum residual magnitude in the summary (default 0 i.e. no restriction)
doAllFactors	Flag to indicate whether to analyse the levels of all factor variables (default TRUE)
maxFactorLevels	(maximum number of levels per factor before it is converted to numeric (default 20))

Value

Saves residual CART trees and associated highlighted residuals for each to the path provided.

Examples

```

require(featurefinder)
data(mycsv)
data$SMIfactor=paste("smi",as.matrix(data$SMIfactor),sep="")
nn=floor(length(data$DAX)/2)

# Can we predict the relative movement of DAX and SMI?
data$y=data$DAX*0
data$y[1:(nn-1)]=((data$DAX[2:nn])-(data$DAX[1:(nn-1)]))/
  (data$DAX[1:(nn-1)]-(data$SMI[2:nn])-(data$SMI[1:(nn-1)]))/(data$SMI[1:(nn-1)])

thismodel=lm(formula=y ~ .,data=data)
expected=predict(thismodel,data)
actual=data$y
residual=actual-expected

```

```

data=cbind(data,expected, actual, residual)

OutputPath=tempdir()
fcsv <- file.path(OutputPath, "mycsv.csv")
write.csv(data[(nn+1):(length(data$y)),], file = fcsv, row.names=FALSE)

ExclusionVars="\"residual\", \"expected\", \"actual\", \"y\""
FactorToNumericList=c()
findFeatures(OutputPath, fcsv, ExclusionVars,FactorToNumericList,
             treeGenerationMinBucket=50,
             treeSummaryMinBucket=20)

```

`generateResidualCutoffCode`

generateResidualCutoffCode

Description

For each tree print a summary of the significant residuals as specified by the user

Usage

```
generateResidualCutoffCode(data, filename, trees, names, runname, ...)
```

Arguments

<code>data</code>	A dataframe
<code>filename</code>	A string
<code>trees</code>	A list of trees generated by <code>saveTree</code>
<code>names</code>	A list of level names
<code>runname</code>	A string corresponding to the name of the factor variable being analysed
<code>...</code>	and parameters to be passed through

Value

A list of residuals for each tree provided.

generateTrees	<i>generateTrees</i>
---------------	----------------------

Description

Generate a residual tree for each level of factor mainfac

Usage

```
generateTrees(data, vars, expr, runname, ...)
```

Arguments

data	A dataframe
vars	A list of candidate predictors
expr	A expression to be modelled by the RPART tree
runname	A string corresponding to the name of the variable being modelled
...	and parameters to be passed through

Value

A list of residual trees for each level of the mainfac factor provided

getVarAv	<i>getVarAv</i>
----------	-----------------

Description

This function generates a residual tree on a subset of the data

Usage

```
getVarAv(dd, varAv, varString)
```

Arguments

dd	A dataframe
varAv	A string corresponding to the numeric field to be averaged within each leaf node
varString	A string

Value

An average of the numeric variable varString in the segment

parseSplits	<i>parseSplits</i>
-------------	--------------------

Description

Extract information relating to the paths and volume of data in the leaves of the tree

Usage

```
parseSplits(thistree)
```

Arguments

thistree	A tree
----------	--------

Value

A list of parsed splits.

printResiduals	<i>printResiduals</i>
----------------	-----------------------

Description

This function generates a residual tree on a subset of the data

Usage

```
printResiduals(
  fileConn,
  all,
  dat,
  runname,
  levelname,
  treeSummaryResidualThreshold,
  treeSummaryMinBucket,
  treeSummaryResidualMagnitudeThreshold,
  ...
)
```

Arguments

fileConn	A file connection
all	A dataframe
dat	The dataset
runname	A string corresponding to the name of the factor being analysed
levelname	A string corresponding to the factor level being analysed
treeSummaryResidualThreshold	The minimum residual threshold
treeSummaryMinBucket	The minumum volume per leaf
treeSummaryResidualMagnitudeThreshold	Minimun residual magnitude
...	and parameters to be passed through

Value

Residuals are printed and also saved in a simplified format.

saveTree

saveTree

Description

Generate a residual tree on a subset of the data specified by the factor level mainfaclev (main factor level)

Usage

```
saveTree(  
  data,  
  vars,  
  expr,  
  i,  
  varname,  
  mainfaclev,  
  treeGenerationMinBucket,  
  ...  
)
```

Arguments

data	A dataframe containing the residual and some predictors
vars	A list of candidate predictors
expr	A expression to be modelled by the RPART tree
i	An integer corresponding to the factor level
varname	A string corresponding to the name of the factor variable being analysed
mainfaclev	A level of the mainfac factor
treeGenerationMinBucket	Minimum size for tree generation
...	and parameters to be passed through

Value

A tree object

Index

- * **data**
 - data, [2](#)
- * **findFeatures**
 - findFeatures, [2](#)
- * **generateTrees**
 - generateTrees, [5](#)
- * **sample**
 - data, [2](#)
- * **saveTree**
 - generateResidualCutoffCode, [4](#)
 - getVarAv, [5](#)
 - parseSplits, [6](#)
 - printResiduals, [6](#)
 - saveTree, [7](#)

data, [2](#)

findFeatures, [2](#)

generateResidualCutoffCode, [4](#)

generateTrees, [5](#)

getVarAv, [5](#)

parseSplits, [6](#)

printResiduals, [6](#)

saveTree, [7](#)