Package 'manymodelr'

March 2, 2019

Title Build and Tune Several Models

Version 0.1.0

Description Frequently one needs a convenient way to build and tune,
several models in one go.The goal is to provide a number of convenience functions useful in machine learning
applications. It provides the ability to build,tune and obtain predictions of
several models in one function. The models are built using 'caret' functions with
easier to read syntax.
Kuhn(2014)<arXiv:1405.6974v14>.
Kuhn(2008)<doi10.18637/jss.v028.i05>.
Chambers,J.M.(1992)<doi:10.1371/journal.pone.0053143>.
Wilkinson,G.N. and Rogers, C. E. (1973)<doi:10.2307/2346786>.

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Imports dplyr(>= 0.7.8), caret(>= 6.0-81), tidyr(>= 0.8.2),
reshape2(>= 1.4.3), plyr(>= 1.8.4), magrittr(>= 1.5),
Metrics(>= 0.1.4), tibble(>= 2.0.1), e1071(>= 1.7-0.1)

URL https://github.com/Nelson-Gon/manymodelr

BugReports https://github.com/Nelson-Gon/manymodelr/issues

NeedsCompilation no

Author Nelson Gonzabato [aut, cre]

Maintainer Nelson Gonzabato <gonzabato@hotmail.com>

Repository CRAN

Date/Publication 2019-03-02 12:40:03 UTC

R topics documented:

---

expo *Convenience functions for use with get_exponent*

Description

Convenience functions for use with get_exponent

Usage

```
expo(n)
```

Arguments

n	The value to which a number should be raised. Useful for expo1

Details

See ?get_exponent for details

Examples

```
## Not run:
square<-expo(2)
square(4)
expo1(2,3)
## End(Not run)
```

expo1 *Convenience functions for use with get_exponent*

Description

Convenience functions for use with get_exponent

Usage

```
expo1(y, x)
```

Arguments

y	The value for which an exponential is required
x	The power to which y should be raised

Details

See ?get_exponent for detials

Examples

```
## Not run:
square<-expo(2)
square(4)
expo1(2,3)
## End(Not run)
```

get_data_Stats *A pipe friendly way to get summary stats for exploratory data analysis*

Description

A pipe friendly way to get summary stats for exploratory data analysis

Usage

```
get_data_Stats(x, func)
```

Arguments

x	The data for which stats are required
func	The nature of function to apply

Details

A convenient wrapper especially useful for get_mode

Value

A data.frame object showing the requested stats

Examples

```
library(dplyr)
mtcars %>%
get_data_Stats(mean)
mtcars %>%
get_data_Stats(get_mode)
## Not run:
get_data_Stats(airquality,min)
airquality%>%
get_data_Stats(get_mode)

## End(Not run)
```

---

get_exponent *Get the exponent of any number or numbers*

Description

Get the exponent of any number or numbers

Usage

```
get_exponent(y, x)
```

Arguments

y	The number for which an exponent is required
x	The power to which y is raised

Details

Depends on the expo and expo1 functions in expo

Value

A data.frame object showing the value,power and result

Examples

```
df<-data.frame(A=c(1123,25657,3987))
get_exponent(df,3)
```

get_mode *A convenience function that returns the mode*

Description

A convenience function that returns the mode

Usage

```
get_mode(x)
```

Arguments

x	The dataframe or vector for which the mode is required

Details

Useful when used together with get_stats in a pipe fashion. These functions are for exploratory data analysis The smallest number is returned if there is a tie in values The function is currently slow for greater than 300,000 rows. It may take up to a minute

Value

a data.frame or vector showing the mode of the variable(s)

Examples

```
test<-c(1,2,3,3,3,3,4,5)
get_mode(test)
## Not run:
mtcars %>%
get_stats(get_mode)
get_stats(mtcars,get_mode)
## End(Not run)
```

modeleR *Perform several kinds of models in one function*

Description

Perform several kinds of models in one function

Usage

```
modeleR(df, yname, xname, modeltype, na.rm = F, new_data, ...)
```

Arguments

df	The data for which analysis is required
yname	The dependent variable
xname	Theindependent variable. Supports formulae x1+x2+...
modeltype	Currently one of lm and aov. Other models may work with inaccuracies
na.rm	Logical. Should missing values be removed from analysis?
new_data	A data.frame object for which new predictions are to be made
...	Additional arguments to the modeltype

Details

This function provides a friendly way to perform any kind of model in one line. The model uses the inbuilt R functions aov and lm to make the predictions.

Value

A list containing summary stats and a data.frame object of some stats.

References

Chambers, J. M. (1992) Linear models. Chapter 4 of Statistical Models in S eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

Wilkinson, G. N. and Rogers, C. E. (1973). Symbolic descriptions of factorial models for analysis of variance. Applied Statistics, 22, 392-399. doi: 10.2307/2346786.

Examples

```
iris1<-iris[1:60,]
iris2<-iris[60:nrow(iris),]
modeleR(iris1,Sepal.Length,Petal.Length,
         lm,na.rm=TRUE,iris2)
```

---

multi_model_1 *Simultaneously train and predict on new data.*

Description

This function provides a convenient way to train several model types. It allows a user to predict on new data and depending on the metrics, the user is able to decide which model predictions to finally use. The models are built based on Max Kuhn's models in the caret package.

Usage

```
multi_model_1(df, yname, xname, method, metric, control, ..., newdata)
```

Arguments

df	The data holding the training dataset
yname	The outcome variable
xname	The predictor variable(s)
method	A vector containing methods to be used as defined in the caret package
metric	One of several metrics. Accuraciy,RMSE,MAE,etc
control	See caret ?trainControl for details.
...	Other arguments to caret's train function
newdata	A data set to validate the model or for which predictions are required

Details

Most of the details of the parameters can be found in the caret package documentation. This function is meant to help in exploratory analysis to make an informed choice of the best models

Value

A list containing two objects. A tibble containing a summary of the metrics per model and a tibble containig predicted values

References

Kuhn (2014), "Futility Analysis in the Cross-Validation of Machine Learning Models" http://arxiv.org/abs/1405.6974,

Kuhn (2008), "Building Predictive Models in R Using the caret" (http://www.jstatsoft.org/article/view/v028i05/v28i05.pdf)

Examples

```
library(caret)
train_set<-createDataPartition(iris$Species,p=0.8,list=FALSE)
valid_set<-iris[-train_set,]
train_set<-iris[train_set,]
ctrl<-trainControl(method="cv",number=5)
set.seed(233)
m<-multi_model_1(train_set,"Species",".",c("knn","rpart"),
                 "Accuracy",ctrl,newdata =valid_set)
m$Predictions
m$Metrics
```

row_mean_na *Replacing all NAs with mean values of a given row*

Description

Replacing all NAs with mean values of a given row

Usage

```
row_mean_na(data, func, observations, na.rm = F, exc)
```

Arguments

data	is the data you for which the mean is needed
func	describes the function to use. Currrently only supports the mean(others may work with some inaccuracies)
observations	takes on column names for which manipulations are required
na.rm	Logical. Should NAs be removed from analysis?
exc	the column to exclude from analysis. Useful for removing factor columns

Value

Returns a data.frame object showing columns with NAs and their replacement if na.rm=T

Examples

```
#This merges our replacement values with the original data containing NAs
row_mean_na(airquality,mean,c("Ozone","Wind"),na.rm=TRUE,0)
## Not run:
row_mean_na(iris,max,c("Sepal.Length","Petal.Length"),na.rm = F,5)

## End(Not run)
```

select_col *A convenient selector gadget*

Description

A convenient selector gadget

Usage

```
select_col(df, x, ...)
```

Arguments

df	The data set from which to select a column
x	The name of a column to select(no quotes)
...	Other columns to select

Details

A friendly way to select a column or several columns. Mainly for non-pipe usage It is recommended to use known select functions to do pipe manipulations. Otherwise convert to tibble

Value

Returns a dataframe with selected columns

Examples

```
select_col(iris,Petal.Length,Sepal.Length,Species,Petal.Width)
# A pipe friendly example
## Not run:
library(dplyr)
as_tibble(iris) %>%
select_col(Species)

## End(Not run)
```

---

select_percentile *Get the row corresponding to a given percentile*

Description

Get the row corresponding to a given percentile

Usage

```
select_percentile(df, n)
```

Arguments

df	The dataframe for which a percentile is required
n	The percentile required eg 10th percentile

Value

A dataframe showing the row corresponding to a given percentile

Examples

```
select_percentile(iris,5)
## Not run:
select_percentile(mtcars,1)

## End(Not run)
```

Index