Package 'WindAnalysis'

April 18, 2018

Type Package

Title Modelling wind turbine acceptance rates.

Version 0.1.0

Author Michael Harper

Maintainer Michael Harper <m.harper@soton.ac.uk>

Description A colleciton of function used for summarising data, statical modelling, and presenting of model results.

License file LICENSE

Imports tidyverse, caret, corrplot, ggplot2, magrittr, Hmisc, broom, car, plyr, psych, stats, ggthemes, grid, DiagrammeR, rsvg, DiagrammeRsvg, scales, knitr

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1.9000

R topics documented:

`AddNameStat` *Function will update the name with the statistic of your choice*

Description

Function will update the name with the statistic of your choice

Usage

```
AddNameStat(df, category, count_col, stat = c("sd", "mean", "count"),
  dp = 0)
```

Arguments

`df`	a dataframe
`category`	the category being named
`count_col`	the column being aggregated against
`stat`	the statistics. "sd" is standard deviation, "mean" or "count".
`dp`	decimal places of returned value

ChiSquared — *Calculate Chi squared statistics*

Description

Internal function used to calculates the Chi Squared statistics for a model

Usage

```
ChiSquared(Model)
```

Arguments

Model	A glm object

CrossValidation — *Function for internally validating a glm model.*

Description

Function for internally validating a glm model.

Usage

```
CrossValidation(dataframe, iterations, foldSize, fullstats = FALSE)
```

Arguments

dataframe	the dataframe which contains the parameters to be modelled. The first column must contain the outcome variable
iterations	the number of iterations for the function to run
foldSize	the size of the test dataset expressed as a value between 0 and 1
fullstats	TRUE/FALSE. Should the full stats be exported?

DurbinWatsonCheck — *Calculates Durbin-Watson statistic for glm model*

Description

Calculate the Durbin-Watson statistic to detect the presence of autocorrelation in the residuals from a regression analysis.

Usage

```
DurbinWatsonCheck(Model)
```

Arguments

Model	A glm object

FiltNumericColumns *Produces summary statistics for an entire numeric dataframe*

Description

Produces summary statistics for an entire numeric dataframe

Usage

```
FiltNumericColumns(dataframe)
```

Arguments

dataframe	the dataframe

Value

a dataframe of summary statistics

FindBestModel *Runs a stepwise optimisation to select the best fitting parameters*

Description

Runs a stepwise optimisation to select the best fitting parameters

Usage

```
FindBestModel(df, variableList, direction = "backward", steps = 1000)
```

Arguments

df	a dataframe
variableList	a list of parameters from the dataframe
direction	"forward" or "backwards. Default "backward"
steps	the number of steps for the model to assess

`grViz_pdf` *# Save a file from DiagrammeR as a PDF figure which can be included in a knitr report. Saves PDF output in the same directory as the ".gv" file specified.*

Description

Save a file from DiagrammeR as a PDF figure which can be included in a knitr report. Saves PDF output in the same directory as the ".gv" file specified.

Usage

```
grViz_pdf(filepath, saveSuffix = "")
```

Arguments

`filepath`	The input location of the .gz file
`saveSuffix`	An optional term to be added to the end of the PDF.

`ksource` *Runs an external Knitr script*

Description

Runs an external Knitr script

Usage

```
ksource(x, ...)
```

Arguments

`x`	A filepath to an external knitr script
`...`	options to be passed to the purl function

`loessfit` *Fit a LOESS curve to a dataset*

Description

Fit a LOESS curve to a dataset

Usage

```
loessfit(x, DegreeValue, SpanValue)
```

Arguments

`x`	the parameter name for the model
`DegreeValue`	the degree of polynomials to be used
`SpanValue`	the parameter ?? which controls the degree of smoothing

LogisticDiagnostics *Logistic Regression Model Diagnostics*

Description

Prints results for the Chi Squared, Psuedo R sqaured values, Variance Inflation Factors and Durbin Watson Test

Usage

```
LogisticDiagnostics(Model)
```

Arguments

Model	A glm object

Value

A printout of diagnostics

LogisticModel *Function to build logistic model from a list of input parameters*

Description

Function to build logistic model from a list of input parameters

Usage

```
LogisticModel(variables, df)
```

Arguments

variables	a list of predictor variables
df	model dataframe

`LogisticModelInt` *Takes a list of variables, and updates the list to include the log transformations (Int)*

Description

Takes a list of variables, and updates the list to include the log transformations (Int)

Usage

```
LogisticModelInt(PredictorVariables, outcome, df)
```

Arguments

`PredictorVariables`	a list of varaibles to be assessed
`outcome`	the outcome variable
`df`	a datafram containing the observations

`LogisticOddsPlot` *Function builds the*

Description

Function builds the

Usage

```
LogisticOddsPlot(Model, Sort = FALSE, Title)
```

Arguments

`Model`	a glmmodel
`Sort`	indicates whether the parameters should be sorted
`Title`	a character string of the title of the of ggplot

LogisticOddsTable *Builds a Logistic Odds Table*

Description

Produces an odds table with confidence intervals for a logistic model

Usage

```
LogisticOddsTable(Model, Sort = FALSE)
```

Arguments

Model	a glmmodel
Sort	boolean indicator whether the parameters should be sorted. Default FALSE

LogisticPseudoR2s *Calculate Logistic Model R-squared values*

Description

Calculates the R squared values (Hosmer and Lemeshow, Cox and Snell and Nagelkerke) values for a logistic regression model.

Usage

```
LogisticPseudoR2s(Model)
```

Arguments

Model	A glm object

LogisticResultsTable *Creates a formatted odds table of the results using ggplot*

Description

Creates a formatted odds table of the results using ggplot

Usage

```
LogisticResultsTable(Model, roundby = 3)
```

Arguments

Model	a logistic regression object
roundby	the number of decimal places to display results. Default 3

LogOddsPlotGraph *Produces boxplots for estimated values from a regression model*

Description

Produces boxplots for estimated values from a regression model

Usage

```
LogOddsPlotGraph(OddsTable, PlotTitle, Sort = FALSE)
```

Arguments

OddsTable	a formatted odds table from the function "LogisticOddsTable"
PlotTitle	the title of the resulting plot
Sort	reorder the plot by variable fit

LogResults *Summarise results for a logistic odds model*

Description

Summarise results for a logistic odds model

Usage

```
LogResults(Model)
```

Arguments

Model	a glm object

LogResultsTable *Takes a list of logistic regression models and creates a summary table*

Description

Takes a list of logistic regression models and creates a summary table

Usage

```
LogResultsTable(...)
```

Arguments

...	a list of glm objects

`matchCategory` *Matches variable names with their full descriptive name. Used when plotting or showing results in a table*

Description

This function is designed to work with the PhD project

Usage

```
matchCategory(inputNames, path = "VariableDisplayNames.csv")
```

Arguments

`inputNames`	a list of names which are to looked up
`path`	the filepath of the lookup file

`matchNames` *Matches variable names with their full descriptive name. Used when plotting or showing results in a table*

Description

This function is designed to work with the PhD project

Usage

```
matchNames(inputNames, path = "VariableDisplayNames.csv")
```

Arguments

`inputNames`	a list of names which are to looked up
`path`	the filepath of the lookup file

`matchNamesColumns` *Checks the row names against the variable full names*

Description

This function is designed to work with the PhD project

Usage

```
matchNamesColumns(dataframe, dropNA = TRUE)
```

Arguments

`dataframe`	a list of names which are to looked up
`dropNA`	a TRUE/FALSE selection whether names should be dropped if empty

`matchUnits` *Matches variable names with their full descriptive name. Used when plotting or showing results in a table*

Description

This function is designed to work with the PhD project

Usage

```
matchUnits(inputNames, path = "VariableDisplayNames.csv")
```

Arguments

`inputNames`	a list of names which are to looked up
`path`	the filepath of the lookup file

`ModelAccuracy` *Assesses the accuracy of a logistic regression model*

Description

Assesses the accuracy of a logistic regression model

Usage

```
ModelAccuracy(dataframe, outcomeVariable, predictorVariables,
  iterations = 300, foldSize = 0.05, fullstats = FALSE)
```

Arguments

`dataframe`	a data frame
`outcomeVariable`	a list of the outcome variables
`predictorVariables`	a string of the column name for the outcome variable
`iterations`	number of iterations to run for the model accuracy. Defaults 300
`foldSize`	size of data proportion used to assess model accuracy. Defaults 0.05
`fullstats`	TRUE/FALSE. Should the full results be returned? Default FALSE

ModelAccuracyFromModel *Assess the model accuracy*

Description

Wrapper for the ModelAccuracy function to extract the variables from the model Would probably be better to rewrite the original function

Usage

```
ModelAccuracyFromModel(Model, Dataframe, ...)
```

Arguments

Model	the glm object
Dataframe	the full dataframe used to build the model
...	additional arguments passed to the 'ModelAccuracy' function.

my.summary *Produces a summary of a column within a dataframe*

Description

Produces a summary of a column within a dataframe

Usage

```
my.summary(x, rounded = 1, ...)
```

Arguments

x	a vector of values
rounded	the number of decimal places to return
...	additional arguments parsed by summary statistic calculations

Value

a list of summary statistics for the variable

OddsTable *Build Odds Table*

Description

Creates an odds table for parameters within the regression model

Usage

```
OddsTable(Model, round = 3)
```

Arguments

Model	A glm object
round	the number of decimal places the figure should be printed to

ParameterUpdate *Update the list of parameters for the model building. Used in conjunction with the "LogisticModel" and "LogsticModelInt" functions*

Description

Update the list of parameters for the model building. Used in conjunction with the "LogisticModel" and "LogsticModelInt" functions

Usage

```
ParameterUpdate(input, add = NULL, remove = NULL)
```

Arguments

input	an existing list of input parameters
add	a list of parameters to be added to the model (optional)
remove	a list of values to be removed from the model (optional)

Value

An updateded model parameter list

`ReducedDataframe` *Reduces a dataframe to a list of parameters for use within regression modelling*

Description

Reduces a dataframe to a list of parameters for use within regression modelling

Usage

```
ReducedDataframe(dataframe, outcomeVariable, inputVariables)
```

Arguments

`dataframe`	a dataframe
`outcomeVariable`	the outcome variable
`inputVariables`	a list of input variables

`scale_colour_Publication` *A colour scale fill for the publication*

Description

A colour scale fill for the publication

Usage

```
scale_colour_Publication(...)
```

Arguments

`...`	option arguments passed to 'discrete_scale'

`scale_fill_Publication` *A fill scale fill for the publication*

Description

A fill scale fill for the publication

Usage

```
scale_fill_Publication(...)
```

Arguments

`...`	option arguments passed to 'discrete_scale'

ScatterPlotOdds *Produces a scatter plot for the logistic regression variables*

Description

Produces a scatter plot for the logistic regression variables

Usage

```
ScatterPlotOdds(df, variable, quantiles = 20)
```

Arguments

df	the dataframe which contains the data
variable	the column reference to search
quantiles	the number of bins in which the data should be grouped

Segmented_Dataset *Splits a dataset based on a categorical variable*

Description

Splits a dataset based on a categorical variable

Usage

```
Segmented_Dataset(df, by)
```

Arguments

df	a dataframe
by	the column used to split the data

Value

a list of dataframes

Segmented_FullOddsPlot

Splits a dataset and forms a segmented Odds plot

Description

This function splits data based on a categorical variable, builds separate glm models for each dataset and compares the results using a Odds Plot

Usage

```
Segmented_FullOddsPlot(df, split_category, predictors, outcome, Title,
  linebreak)
```

Arguments

df	a dataframe to be split
split_category	the column to be used to split the dataset by
predictors	a list of variables to be used within the regression model
outcome	the outcome variable
Title	a string to be used within the ggplot title
linebreak	the spacing to be used between gridlines in the plot

Segmented_LogisticModels

Build Segmented Logistic Regression Models

Description

Combines the SplitDatasetbyVariable function and "Logstic Model"

Usage

```
Segmented_LogisticModels(SegmentedDatasets, predictors, outcome)
```

Arguments

SegmentedDatasets	the full dataframe to be split
predictors	the name of the column for the dataset to be split using
outcome	the outcome variable

Value

a list of logistic regression models

Segmented_LogisticModelsComplete

Returns the a list of results for split logistic regresion model.

Description

This function splits data based on a categorical variable, builds separate glm models for each dataset and compares the results using a Odds Plot. This returns the full stages of the analysis 1) dataset 2) glm models 3) Summary Odds table 4) logistic regresion plot.

Usage

```
Segmented_LogisticModelsComplete(df, split_category, variables_list,
  outcome_variable, linebreak = 0.1, limits = 1)
```

Arguments

df	a dataframe to be split
split_category	the column to be used to split the dataset by
variables_list	a list of variables to be used within the regression model
outcome_variable	the outcome variable
linebreak	the spacing to be used between gridlines in the plot
limits	the extent of the plot

Segmented_OddsPlot *Builds an odds plot for a segmented logistic regression model*

Description

Builds an odds plot for a segmented logistic regression model

Usage

```
Segmented_OddsPlot(OddsTables, PlotTitle, linebreak)
```

Arguments

OddsTables	a list of odds as produced from the function "OddsTableSegmented"
PlotTitle	the title for the ggplot object
linebreak	numeric. Spacing between minor line breaks in plot

Segmented_OddsPlotGroupedCustom *Plots a faceted odds ratio plot for a list of segmented odds tables*

Description

Plots a faceted odds ratio plot for a list of segmented odds tables

Usage

```
Segmented_OddsPlotGroupedCustom(OddsTables, linebreak = 0.2, scale = 1)
```

Arguments

OddsTables	a list of odds as produced from the function "OddsTableSegmented"
linebreak	numeric. Spacing between minor line breaks in plot
scale	the limits of the plot

Segmented_OddsTable *Builds Segmented Regression Odds Tables*

Description

Split dataset into segments and build logistic models into a list

Usage

```
Segmented_OddsTable(LogisticModelList)
```

Arguments

LogisticModelList	a list of logistic regression models

Value

a single odds table which contains the statistics for the segmented model

SummariseDataframe *Summarise a Dataframe*

Description

Produces summary statistics for rows in a dataframe

Usage

```
SummariseDataframe(dataframe)
```

Arguments

dataframe	a datatable
...	additional arguments parsed to ‘my.sumarry‘

theme_Publication *A theme for use within the publication*

Description

A theme for use within the publication

Usage

```
theme_Publication(base_size = 14, base_family = "helvetica")
```

Arguments

base_size	the font size of the plot Defaults to 14
base_family	the font family of the plot. Defaults to "helvetica"

TwoWayFrequency *Creates a 2 way frequency table from a dataframe and calculates the percentage of a specified category*

Description

Creates a 2 way frequency table from a dataframe and calculates the percentage of a specified category

Usage

```
TwoWayFrequency(Rows, Columns, SumPercentage, inputDataframe)
```

Arguments

Rows	the variable to counted in the rows
Columns	the variable to be counted in the columns
SumPercentage	the parameter to be summed in the column
inputDataframe	the dataframe containing the row and column parameters

---

VariableCluster *Divides a set of numeric variables into disjoint or hierarchical cluster which can be used to diagnose collinearity between variables*

Description

Divides a set of numeric variables into disjoint or hierarchical cluster which can be used to diagnose collinearity between variables

Usage

```
VariableCluster(PredictorVariables, df)
```

Arguments

PredictorVariables	Parameter names to be considered
df	The dataframe containing the parameters

VIFcheck *Calculate Variance Inflation Factor (VIF)*

Description

Calculates variance-inflation and generalized variance-inflation factors for linear and generalized linear models, and returns any which are above 10.

Usage

```
VIFcheck(Model)
```

Arguments

Model	an object that responds to coef, vcov, and model.matrix, such as an lm or glm object

Index