

Package ‘ConfusionTableR’

October 7, 2021

Type Package

Title Confusion Matrix Toolset

Version 1.0.2

Maintainer Gary Hutson <hutsons-hacks@outlook.com>

Description Takes the outputs of a 'caret' confusion matrix and allows for the quick conversion of these list items to lists.

The intended usage is to allow the tool to work with the outputs of machine learning classification models.

This tool works with classification problems for binary and multi-classification problems and allows for the record level conversion of the confusion matrix outputs.

This is useful, as it allows quick conversion of these objects for storage in database systems and to track ML model performance over time.

Traditionally, this approach has been used for highlighting model representation and feature slip-page.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.1.1

Imports dplyr, tidyr, magrittr, caret, mlbench, lattice, scales, ggplot2, e1071, randomForest, purrr, furr

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Collate 'MultiFramer.R' 'SingleFramer.R' 'binaryVisualise.R' 'dummymcoder.R' 'globals.R'

Language en-US

Author Gary Hutson [aut, cre] (<<https://orcid.org/0000-0003-3534-6143>>)

Date/Publication 2021-10-07 20:40:08 UTC

R topics documented:

dummy_encoder 2

dummy_encoder	<i>Dummy Encoder function to encode multiple columns at once</i>
---------------	--

Description

This function has been designed to encode multiple columns at once and allows the user to specify whether to drop the reference columns or retain them in the data

Usage

```
dummy_encoder(df, columns, map_fn = furrr::future_map, remove_original = TRUE)
```

Arguments

df	- data.frame object to pass to the function
columns	- vector of columns to be encoded for dummy encoding
map_fn	- choice of mapping function purrr:map or furrr::future_map accepted
remove_original	- remove the variables that the dummy encodings are based off

Value

A tibble containing the dummy encodings

Examples

```
## Not run:
#Use the NHR dataset
df <- NHRdatasets::stranded_data
#Create a function to select categorical variables
sep_categorical <- function(df){
  cats <- df %>%
    dplyr::select_if(is.character)
  return(cats)
}
cats <- sep_categorical(df) %>%
  dplyr::select(-c(admit_date))
#Dummy encoding
columns_vector <- c(names(cats))
dummy_encodings <- dummy_encoder(cats, columns_vector)
glimpse(dummy_encodings)

## End(Not run)
```

Index

`dummy_encoder`, [2](#)