

Package: trainsplit (via r-universe)

August 21, 2024

Title Split a Dataframe, Tibble, or Data.table into Training and Test Sets

Version 1.0

Description Split a dataframe, tibble, or data.table into a list containing training and test sets. Can specify either number or percentage of observations to go into the training set.

URL <https://github.com/eastnile/trainsplit>

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Imports data.table

Suggests tibble, dplyr

License MIT + file LICENSE

Repository <https://eastnile.r-universe.dev>

RemoteUrl <https://github.com/eastnile/trainsplit>

RemoteRef HEAD

RemoteSha 74ece4b3864a93e66301504b3ce8dd43048fa59c

Contents

trainsplit	2
Index	3

`trainsplit`*trainsplit*

Description

Splits a dataframe, tibble, or data.table into a test set and training set. Specify either the number or percentage of observations to put into training set.

Usage

```
trainsplit(  
  data,  
  ntrain = NULL,  
  trainpct = NULL,  
  round_ntrain = "round",  
  seed = NULL  
)
```

Arguments

<code>data</code>	The dataset you want to split
<code>ntrain</code>	The number of observations to go into the training set. Must be ≥ 0 and $\leq \text{nrow}(\text{data})$.
<code>trainpct</code>	Fraction of observations to go into training set. Must be ≥ 0 and ≤ 1 . If set to 0 or 1, the empty test or training set will still inherit the same column names and types as the original dataset.
<code>round_ntrain</code>	What to do when $\text{nrow}(\text{data}) * \text{trainpct}$ is not a whole number. Default behavior is to round the size of the training set. Use 'ceiling' or 'floor' to instead set the size of training set to next highest or lowest whole number.
<code>seed</code>	Sets the random seed; use this argument if you want to always get the same result. Note: sets seed only locally within the function.

Value

A list, containing the training and test sets.

Examples

```
# Returns a list containing the training and test sets:  
trainsplit(mtcars, trainpct = 0.75)  
# Specify size of training set by number of rows, not percent:  
trainsplit(mtcars, ntrain = 10)  
# Size of training set rounds to one:  
trainsplit(mtcars, trainpct = 0.01, round_ntrain = 'ceiling')  
# Also works with data.table:  
trainsplit(data.table::as.data.table(mtcars), trainpct = 0.75)
```

Index

`trainsplit`, 2