

Package ‘PdM’

April 2, 2019

Type Package

Title R package for predictive maintenance

Version 0.1.0

Date 2019-02-30

Author Cuong Sai [aut, cre],
Maxim Shcherbakov [aut]

Maintainer Cuong Sai <svcuonghvktqs@gmail.com>

Description Contains tools to analyze, visualize multiple multivariate time series,
explore failure modes, and build predictive maintenance models for IoT.

License GPL-3

Depends R (≥ 2.10)

Imports ggplot2,
dplyr,
magrittr,
skimr,
viridis

URL <https://github.com/forvis/PdM>

BugReports <https://github.com/forvis/PdM/issues>

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

calculate_rul	2
handle_misv	2
PdM	3
runShinyPdM	3
summarize_data	4
transfrom_data	4
validate_data	5
visualize_data	6
visualize_RUL	7

Index 8

calculate_rul	<i>Remaining useful life calculation</i>
---------------	--

Description

Generates the remaining useful life (rul) colum for training data

Usage

```
calculate_rul(df)
```

Arguments

df	a training dataframe containing multiple multivariate time series formatted using the specific Table Schema, use <code>showDF()</code> to display schema specification details.
----	---

Value

a new training data frame with rul column

Author(s)

Cuong Sai and Maxim Shcherbakov.

Examples

```
train_data <- calculate_rul(train_data)
```

handle_misv	<i>Handling Missing Values</i>
-------------	--------------------------------

Description

Imputation and Removing Data

Usage

```
handle_misv(df, method)
```

Arguments

df	the input dataframe containing multiple multivariate time series formatted using the specific Table Schema with missing values, use <code>showDF()</code> to display schema specification details.
method	the method forhandling missing values. Possible methods are: omit: removes the rows containing any missing values mean: for point graphs (scatter plots) median: for both line and point graphs mod: for box plots h: for histogram hf: for histograms of the healthy vs failing sensor Values

Value

A new data frame without missing values

Author(s)

Cuong Sai and Maxim Shcherbakov.

Examples

```
## Not run:
new_data <- handle_misv(data, method = "omit")
new_data <- handle_misv(data, method = "mean")
new_data <- handle_misv(data, method = "median")

## End(Not run)
```

PdM	<i>R package for predictive maintenance</i>
-----	---

Description

Contains tools to analyze, visualize multiple multivariate time series, explore failure modes, and build predictive maintenance models for IoT.

runShinyPdM	<i>Run shinyPdM</i>
-------------	---------------------

Description

Run a local instance of shinyPdM.

Usage

```
runShinyPdM(...)
```

Arguments

... [any]
Additional arguments passed to shiny's `runApp()` function.

Examples

```
## Not run:
runShinyPdM()

## End(Not run)
```

`summarize_data`*Extract summary statistics for input data frame*

Description

This function used to summarize the input data frame

Usage

```
summarize_data(df)
```

Arguments

`df` dataframe containing multiple multivariate time series formatted using the specific Table Schema, use `showDF()` to display schema specification details.

Details

`summarize_data()` is an alternative to `summary()`, quickly providing a broad overview of a data frame. It handles data of all types, dispatching a different set of summary functions based on the types of columns in the data frame.

Value

a data frame containing the by variables and the statistical summaries

See Also

[showDF](#), [validate_data](#)

Examples

```
summary_data(train_data)
summary_data(test_data)
```

`transfrom_data`*Data transforms*

Description

The function for data transformation. For more details about the parameter arguments, use `?transfrom_data`

Usage

```
transfrom_data(train_df, test_df, method = "range")
```

Arguments

<code>train_df</code>	the training dataframe containing multiple multivariate time series formatted using the specific Table Schema, use <code>showDF()</code> to display schema specification details.
<code>test_df</code>	the test dataframe containing multiple multivariate time series formatted using the specific Table Schema, use <code>showDF()</code> to display schema specification details.
<code>method</code>	<p>The transform method. Possible methods are:</p> <p>BoxCox: apply a Box-Cox transform, values must be non-zero and positive.</p> <p>YeoJohnson: apply a Yeo-Johnson transform, like a BoxCox, but values can be negative.</p> <p>expoTrans: apply a power transform like BoxCox and YeoJohnson.</p> <p>zv: remove attributes with a zero variance (all the same value).</p> <p>nzv: remove attributes with a near zero variance (close to the same value).</p> <p>center: divide values by standard deviation.</p> <p>scale: subtract mean from values.</p> <p>range: normalize values.</p> <p>pca: transform data to the principal components.</p> <p>ica: transform data to the independent components.</p> <p>spatialSign: project data onto a unit circle.</p>

Value

Returns the transformed training and test datasets

Author(s)

Cuong Sai and Maxim Shcherbakov.

See Also

[showDF](#), [validate_data](#), [summarize_data](#)

Examples

```
train
```

validate_data

Training and Test datasets Validation

Description

Checks if the input data is correctly formatted in accordance with the training and test datasets

Usage

```
validate_data(df)
```

Arguments

df dataframe containing multiple multivariate time series formatted using the specific Table Schema, use `showDF()` to display schema specification details.

Details

Checks that `fc` contains necessary column and the composite primary key values are not duplicated.

Value

TRUE if the checks are passed, FALSE otherwise.

Author(s)

Cuong Sai and Maxim Shcherbakov.

See Also

[showDF](#)

Examples

```
validate_data(train_data)
```

<code>visualize_data</code>	<i>Training and Test data Visualization for predictive maintenance</i>
-----------------------------	--

Description

The function for plotting of ggplot objects. For more details about the graphical parameter arguments, use `?visualize_data`

Usage

```
visualize_data(df, id_engine, cols = NULL, type = "l", n_step = 20)
```

Arguments

df dataframe containing multiple multivariate time series formatted using the specific Table Schema, use `showDF()` to display schema specification details.

id_engine id for the input data to be shown.

cols for the variable names of the input data to be shown.

type what type of plot should be drawn. Possible types are:
l: for line graphs

- p:** for point graphs (scatter plots)
- b:** for both line and point graphs
- bp:** for box plots
- h:** for histogram
- hf:** for histograms of the healthy vs failing sensor Values

Value

a ggplot object containing the subgraphs of each variable from the input data.

Author(s)

Cuong Sai and Maxim Shcherbakov.

See Also

[showDF](#), [validate_data](#),[summarize_data](#)

Examples

```
visualize_data(train_data, id_engine = 1:10, type = "l")
visualize_data(train_data, id_engine = 1:10, cols = c("s1", "s2", "S8", "c2"), type = "p")
visualize_data(train_data, id_engine = 1:20, type = "bp")
visualize_data(train_data, id_engine = 1:100, type = "h")
visualize_data(train_data, id_engine = 1:100, type = "hp", n_step = 30)
```

visualize_RUL	<i>RUL visualization</i>
---------------	--------------------------

Description

Generates data_visualization

Usage

```
visualize_RUL(df, id_engine, type = "bar")
```

Arguments

df	data
id_engine	indentificator
cols	specisly cols

Details

load input data

Value

a list with the following variables

Author(s)

Cuong Sai and Maxim Shcherbakov.

Index

*Topic **Remaining**
calculate_rul, 2

*Topic **data**
transfrom.data, 4
visualize.data, 6
visualize.RUL, 7

*Topic **life**
calculate_rul, 2

*Topic **missing**
handle_misv, 2

*Topic **transform**
transfrom.data, 4

*Topic **useful**
calculate_rul, 2

*Topic **values**
handle_misv, 2

*Topic **visualization**
visualize.data, 6

calculate_rul, 2

data.visualization (visualize.RUL), 7

handle_misv, 2

PdM, 3

PdM-package (PdM), 3

runShinyPdM, 3

showDF, 4–7

summarize_data, 4, 5, 7

transfrom.data, 4

validate.data, 4, 5, 5, 7

visualize.data, 6

visualize.RUL, 7