Package 'rmlnomogram'

January 8, 2025

Title Construct Explainable Nomogram for a Machine Learning Model

Version 0.1.2

Description Construct an explainable nomogram for a machine learning (ML) model to improve availability of an ML prediction model in addition to a computer application, particularly in a situation where a computer, a mobile phone, an internet connection, or the application accessibility are unreliable. This package enables a nomogram creation for any ML prediction models, which is conventionally limited to only a linear/logistic regression model. This nomogram may indicate the explainability value per feature, e.g., the Shapley additive explanation value, for each individual. However, this package only allows a nomogram creation for a model using categorical without or with single numerical predictors. Detailed methodologies and examples are documented in our vignette, available at <https://htmlpreview.github.io/?https://github.com/herdiantrisufriyana/rmlnomogram/blob/master/doc/ml_nomogram_exemplar.html>.

Depends R (>= 4.4)

Imports dplyr,
purrr,
broom,
stats,
ggplot2,
ggpubr,
stringr,
tidyr,
utils

Suggests tidyverse,
knitr,
caret,
randomForest,
iml,
testthat (>= 3.0.0)

VignetteBuilder knitr

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

LazyData true

Config/testthat/edition 3

Contents

create_nomogram 2
nomogram_features 4
nomogram_features2 5
nomogram_features3 6
nomogram_features4 6
nomogram_outputs 7
nomogram_outputs2 7
nomogram_outputs3 8
nomogram_outputs4 8
nomogram_shaps 9
nomogram_shaps2 9
nomogram_shaps3 10
nomogram_shaps4 11

Index **12**

`create_nomogram` *Construct nomogram for a machine learning model*

Description

This function constructs a nomogram for either binary or continuous outcomes based on provided sample features and outputs. It can also incorporate feature explainability values, such as SHAP values.

Usage

```
create_nomogram(
  sample_features,
  sample_output,
  feature_exp = NULL,
  threshold = 0.5,
  prob = FALSE,
  est = FALSE,
  verbose = FALSE
)
```

Arguments

sample_features	A data frame of feature values where each column represents a feature. The data frame must contain all possible combinations of feature values. There must be at least one categorical predictor and no more than one numerical predictor. Only factor and numeric data types are allowed. The column name 'output' is not allowed. Must not contain any NA values.
sample_output	A data frame with one column 'output' containing numeric values for either the predicted probabilities (for binary outcomes) or estimated values (for continuous outcomes). Must not contain any NA values.
feature_exp	Optional data frame containing feature explainability values (e.g., SHAP values) with one column for each feature. The structure must match sample_features in terms of column names. Each column must contain numeric values. Must not contain any NA values.
threshold	A numeric scalar between 0 and 1, used to define the threshold for classifying predicted probabilities into binary outcomes. A sample is predicted positive if the predicted probability is equal or greater than this threshold.
prob	A logical scalar indicating if the predicted probabilities should be shown in the nomogram.
est	A logical scalar indicating if the estimated values should be shown in the nomogram.
verbose	A logical scalar indicating whether to show a progress bar if it is required.

Value

A ggplot object representing the nomogram.

Examples

```
# Binary outcome (or class-wise multinomial outcome)

## 1 - Categorical predictors and binary outcome without probability
data(nomogram_features)
data(nomogram_outputs)
create_nomogram(nomogram_features, nomogram_outputs)

## 2 - Categorical predictors and binary outcome with probability
create_nomogram(nomogram_features, nomogram_outputs, prob = TRUE)

data(nomogram_shaps)
create_nomogram(
  nomogram_features, nomogram_outputs, nomogram_shaps
  , prob = TRUE
)

## 3 - Categorical and 1 numerical predictors and binary outcome with probability
data(nomogram_features2)
data(nomogram_outputs2)
```

```
create_nomogram(nomogram_features2, nomogram_outputs2, prob = TRUE)

data(nomogram_shaps2)
create_nomogram(
  nomogram_features2, nomogram_outputs2, nomogram_shaps2
  , prob = TRUE
)

# Continuous outcome

## 4 - Categorical predictors and continuous outcome
data(nomogram_features3)
data(nomogram_outputs3)
create_nomogram(nomogram_features3, nomogram_outputs3, est = TRUE)

data(nomogram_shaps3)
create_nomogram(
  nomogram_features3, nomogram_outputs3, nomogram_shaps3
  , est = TRUE
)

## 5 - Categorical and 1 numerical predictors and continuous outcome
data(nomogram_features4)
data(nomogram_outputs4)
create_nomogram(nomogram_features4, nomogram_outputs4, est = TRUE)

data(nomogram_shaps4)
create_nomogram(
  nomogram_features4, nomogram_outputs4, nomogram_shaps4
  , est = TRUE
)
```

`nomogram_features` *Nomogram features using categorical predictors*

Description

An example of a data frame for `sample_features` argument in `create_nomogram` function, must only include all possible combinations of feature values, where one column is available for each feature.

Usage

```
nomogram_features
```

Format

A data frame with 16 rows and 4 columns:

cyl.6 A categorical predictor with values of 0 and 1.

cyl.8 A categorical predictor with values of 0 and 1.

qsec.1 A categorical predictor with values of 0 and 1.

vs.1 A categorical predictor with values of 0 and 1.

Source

Derived from `mtcars` for examples in this package.

`nomogram_features2` *Nomogram features using categorical and 1 numerical predictors*

Description

An example of a data frame for `sample_features` argument in `create_nomogram` function, must only include all possible combinations of feature values, where one column is available for each feature.

Usage

```
nomogram_features2
```

Format

A data frame with 80 rows and 4 columns:

.

qsec A numerical predictor without decimal.

cyl.6 A categorical predictor with values of 0 and 1.

cyl.8 A categorical predictor with values of 0 and 1.

vs.1 A categorical predictor with values of 0 and 1.

Source

Derived from `mtcars` for examples in this package.

nomogram_features3 *Nomogram features using categorical predictors*

Description

An example of a data frame for `sample_features` argument in `create_nomogram` function, must only include all possible combinations of feature values, where one column is available for each feature.

Usage

```
nomogram_features3
```

Format

A data frame with 16 rows and 4 columns:

cyl.6 A categorical predictor with values of 0 and 1.

cyl.8 A categorical predictor with values of 0 and 1.

qsec.1 A categorical predictor with values of 0 and 1.

vs.1 A categorical predictor with values of 0 and 1.

Source

Derived from `mtcars` for examples in this package.

nomogram_features4 *Nomogram features using categorical and 1 numerical predictors*

Description

An example of a data frame for `sample_features` argument in `create_nomogram` function, must only include all possible combinations of feature values, where one column is available for each feature.

Usage

```
nomogram_features4
```

Format

A data frame with 80 rows and 4 columns:

.

qsec A numerical predictor without decimal.

cyl.6 A categorical predictor with values of 0 and 1.

cyl.8 A categorical predictor with values of 0 and 1.

vs.1 A categorical predictor with values of 0 and 1.

Source

Derived from `mtcars` for examples in this package.

`nomogram_outputs` *Nomogram outputs using the predicted probability of binary outcome*

Description

An example of a data frame for `sample_output` argument in `create_nomogram` function, must only include the predicted probabilities for binary outcome.

Usage

```
nomogram_outputs
```

Format

A data frame with 16 rows and 1 column:

output A binary outcome with values from 0 to 1.

Source

Generated by a caret randomforest model using categorical predictors for examples in this package.

`nomogram_outputs2` *Nomogram outputs using the predicted probability of binary outcome*

Description

An example of a data frame for `sample_output` argument in `create_nomogram` function, must only include the predicted probabilities for binary outcome.

Usage

```
nomogram_outputs2
```

Format

A data frame with 80 rows and 1 column:

output A binary outcome with values from 0 to 1.

Source

Generated by a caret randomforest model using categorical and 1 numerical predictors for examples in this package.

`nomogram_outputs3` *Nomogram outputs using the estimated value of numerical outcome*

Description

An example of a data frame for `sample_output` argument in `create_nomogram` function, must only include the estimated values for numerical outcome.

Usage

```
nomogram_outputs3
```

Format

A data frame with 16 rows and 1 column:

output A numerical outcome.

Source

Generated by a caret randomforest model using categorical predictors for examples in this package.

`nomogram_outputs4` *Nomogram outputs using the estimated value of numerical outcome*

Description

An example of a data frame for `sample_output` argument in `create_nomogram` function, must only include the estimated values for numerical outcome.

Usage

```
nomogram_outputs4
```

Format

A data frame with 80 rows and 1 column:

output A numerical outcome.

Source

Generated by a caret randomforest model using categorical and 1 numerical predictors for examples in this package.

`nomogram_shaps` *Nomogram SHAP values using categorical predictors and binary outcome*

Description

An example of a data frame for `feature_exp` argument in `create_nomogram` function, must only include feature explainability value per sample (i.e., SHAP value), where one column is available for each feature.

Usage

```
nomogram_shaps
```

Format

A data frame with 16 rows and 4 columns:

cyl.6 A predictor with SHAP values.

cyl.8 A predictor with SHAP values.

qsec.1 A predictor with SHAP values.

vs.1 A predictor with SHAP values.

Source

Computed by iml from a caret randomforest model using categorical predictors for examples in this package.

`nomogram_shaps2` *Nomogram SHAP values using categorical and 1 numerical predictors and binary outcome*

Description

An example of a data frame for `feature_exp` argument in `create_nomogram` function, must only include feature explainability value per sample (i.e., SHAP value), where one column is available for each feature.

Usage

```
nomogram_shaps2
```

Format

A data frame with 80 rows and 4 columns:

cyl.6 A predictor with SHAP values.

cyl.8 A predictor with SHAP values.

qsec A predictor with SHAP values.

vs.1 A predictor with SHAP values.

Source

Computed by iml from a caret randomforest model using categorical and 1 numerical predictors for examples in this package.

`nomogram_shaps3` *Nomogram SHAP values using categorical predictors and numerical outcome*

Description

An example of a data frame for `feature_exp` argument in `create_nomogram` function, must only include feature explainability value per sample (i.e., SHAP value), where one column is available for each feature.

Usage

```
nomogram_shaps3
```

Format

A data frame with 16 rows and 4 columns:

cyl.6 A predictor with SHAP values.

cyl.8 A predictor with SHAP values.

qsec.1 A predictor with SHAP values.

vs.1 A predictor with SHAP values.

Source

Computed by iml from a caret randomforest model using categorical predictors for examples in this package.

nomogram_shaps4 *Nomogram SHAP values using categorical and 1 numerical predictors and numerical outcome*

Description

An example of a data frame for `feature_exp` argument in `create_nomogram` function, must only include feature explainability value per sample (i.e., SHAP value), where one column is available for each feature.

Usage

```
nomogram_shaps4
```

Format

A data frame with 80 rows and 4 columns:

cyl.6 A predictor with SHAP values.

cyl.8 A predictor with SHAP values.

qsec A predictor with SHAP values.

vs.1 A predictor with SHAP values.

Source

Computed by iml from a caret randomforest model using categorical and 1 numerical predictors for examples in this package.

Index

∗ **dataset**
nomogram_features, 4
nomogram_features2, 5
nomogram_features3, 6
nomogram_features4, 6
nomogram_outputs, 7
nomogram_outputs2, 7
nomogram_outputs3, 8
nomogram_outputs4, 8
nomogram_shaps, 9
nomogram_shaps2, 9
nomogram_shaps3, 10
nomogram_shaps4, 11
∗ **ml-nomogram**
create_nomogram, 2

create_nomogram, 2, *4–11*

nomogram_features, 4
nomogram_features2, 5
nomogram_features3, 6
nomogram_features4, 6
nomogram_outputs, 7
nomogram_outputs2, 7
nomogram_outputs3, 8
nomogram_outputs4, 8
nomogram_shaps, 9
nomogram_shaps2, 9
nomogram_shaps3, 10
nomogram_shaps4, 11