Package 'MLeval'

January 20, 2025

Title Machine Learning Model Evaluation

Version 0.3

Author Christopher R John

Maintainer Christopher R John <chris.r.john86@gmail.com>

Description Straightforward and detailed evaluation of machine learning models. 'MLeval' can produce receiver operating characteristic (ROC) curves, precision-recall (PR) curves, calibration curves, and PR gain curves. 'MLeval' accepts a data frame of class probabilities and ground truth labels, or, it can automatically interpret the Caret train function results from repeated cross validation, then select the best model and analyse the results. 'MLeval' produces a range of evaluation metrics with confidence intervals.

License AGPL-3

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

Suggests knitr, rmarkdown

Imports ggplot2

VignetteBuilder knitr

RoxygenNote 6.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-02-12 06:40:02 UTC

Contents

---

brier_score   *brier_score: A Brier score function*

Description

Calculates the Brier score to evaluate probabilities. A data frame of probabilities and ground truth labels must be passed in to evaluate. Raw probability data must be column1: prob G1, column2: prob G2, column3: obs labels, column4: Group (optional). Zero is optimal and more positive is less.

Usage

```
brier_score(preds, positive = colnames(preds)[2])
```

Arguments

preds	Data frame: Data frame of probabilities and ground truth labels.
positive	Character vector: The name of the positive group, must equal a column name consisting of probabilities.

Value

Brier score

Examples

```
r2 <- brier_score(preds)
```

---

evalm   *evalm: Evaluate Machine Learning Models in R*

Description

evalm is for machine learning model evaluation in R. The function can accept the Caret 'train' function results to evaluate machine learning predictions or a data frame of probabilities and ground truth labels can be passed in to evaluate. Probability data must be column1: probability group1 (column named as your group name 1), column2: probability group2 (column named as your group name 2), column3: observation labels (column named 'obs'), column4: Group, e.g. different models (column named 'Group'), optional to include if different models are combined horizontally.

Usage

```
evalm(list1, gnames = NULL, title = "", cols = NULL,
  silent = FALSE, rlinethick = 1.25, fsize = 12.5,
  dlinecol = "grey", dlinethick = 0.75, bins = 6, optimise = "INF",
  percent = 95, showplots = TRUE, positive = NULL, plots = c("prg",
  "pr", "r", "cc"))
```

Arguments

list1	List or data frame: List of Caret results objects from train, or a single train results object, or a data frame of probabilities and observed labels
gnames	Character vector: A vector of group names for the fit objects
title	Character string: A title for the ROC plot
cols	Character vector: A vector of colours for the group or groups
silent	Logical flag: whether to hide messages (default=FALSE)
rlinethick	Numerical value: Thickness of the ROC curve line
fsize	Numerical value: Font size for the ROC curve plots
dlinecol	Character string: Colour of the diagonal line
dlinethick	Numerical value: Thickness of the diagonal line
bins	Numerical value: Number of bins for calibration curve
optimise	Character string: Metric by which to select the operating point (INF, MCC, or F1)
percent	Numerical value: percentage for the confidence intervals (default = 95)
showplots	Logical flag: whether to show plots or not
positive	Character string: Name of the positive group (will effect PR metrics)
plots	Character vector: which plots to show: r = roc, pr = proc, prg = precision recall gain, cc = calibration curve

Value

List containing: 1) A ggplot2 ROC curve object for printing 2) A ggplot2 PROC object for printing 3) A ggplot2 PRG curve for printing 4) Optimised results according to defined metric 5) P cut-off of 0.5 standard results

Examples

```
r <- evalm(fit)
```

---

fit   *Random forest fitted object from Caret on Sonar data*

Description

Caret was run using 10 fold cross validation on the Sonar data with random forest used to predict the response variable.

Usage

```
fit
```

Format

A Caret train object

---

fit1   *Random forest fitted object from Caret on Sonar data*

Description

Caret was run using 10 fold repeated cross validation on the Sonar data with random forest used to predict the response variable.

Usage

```
fit1
```

Format

A Caret train object

---

fit2   *Gradient boosted machines fitted object from Caret on Sonar data*

Description

Caret was run using 10 fold repeated cross validation on Sonar data with GBM used to predict the response variable.

Usage

```
fit2
```

Format

A Caret train object

---

fit3 *Random forest fitted object from Caret on Sonar data with log-likelihood objective function*

Description

Caret was run using 10 fold repeated cross validation on the Sonar data using random forest to predict the response variable. Log-likelihood was set to be the objective function to select the best model from cross validation.

Usage

```
fit3
```

Format

A Caret train object

---

im_fit *Random forest fitted object from Caret on simulated imbalanced data*

Description

Caret was run using 10 fold repeated cross validation on the Sonar data with random forest to predict the response variable.

Usage

```
im_fit
```

Format

A Caret train object

LL   *LL: Log-likelihood function*

Description

Calculates the Log-likelihood to evaluate probabilities. A data frame of probabilities and ground truth labels must be passed in to evaluate. Raw probability data must be column1: prob G1, column2: prob G2, column3: obs labels, column4: Group (optional). Zero is optimal and more negative is less.

Usage

```
LL(preds, positive = colnames(preds)[2])
```

Arguments

preds	Data frame: Data frame of probabilities and ground truth labels.
positive	Character vector: The name of the positive group, must equal a column name consisting of probabilities.

Value

Log-likelihood

Examples

```
r1 <- LL(preds)
```

---

preds   *Predictions from gbm on the Sonar test data*

Description

The Sonar data was split into training (157 points) and testing (51 points), a gbm model was fitted using Caret on the training data. Then these are the predicted probabilities of the model on the test data.

Usage

```
preds
```

Format

A data frame with 51 rows as points and 4 variables

predsc	*Predictions from gbm and random forest on the Sonar test data*

Description

The Sonar data was split into training (157 points) and testing (51 points), a gbm model was fitted using Caret on the training data. Then these are the predicted probabilities of the model on the test data. A random forest model was then fit and tested in the same manner. The probabilities and ground truth labels were combined horizontally for further analysis.

Usage

```
predsc
```

Format

A data frame with 102 rows as points and 4 variables

Sonar	*Sonar data*

Description

The Sonar data consist of 208 data points collected on 60 predictors. The goal is to predict the two classes M for metal cylinder or R for rock. This data has been obtained from the 'mlbench' package. Response variable is in the Class column.

Usage

```
Sonar
```

Format

A data frame with 208 rows as points and 61 variables

Index