

# Package ‘mltest’

November 16, 2018

**Title** Classification Evaluation Metrics

**Version** 1.0.1

**Description** A fast, robust and easy-to-use calculation of multi-class classification evaluation metrics based on confusion matrix.

**License** GPL-2

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.0

**NeedsCompilation** no

**Author** G. Dudnik [aut, cre, cph]

**Maintainer** G. Dudnik <gl.dudnik@gmail.com>

**Repository** CRAN

**Date/Publication** 2018-11-16 17:00:03 UTC

## R topics documented:

ml_test . . . . .	1
<b>Index</b>	<b>5</b>

---

ml_test	<i>multi-class classifier evaluation metrics based on a confusion matrix (contingency table)</i>
---------	--

---

### Description

Calculates multi-class classification evaluation metrics: **balanced.accuracy**, balanced accuracy (**balanced.accuracy**), diagnostic odds ratio (**DOR**), error rate (**error.rate**), F.beta (**F0.5**, **F1** (F-measure, F-score), **F2** with where beta is 0.5, 1 and 2 respectively), false positive rate (**FPR**), false negative rate (**FNR**), false omission rate ((**FOR**)), false discovery rate (**FDR**), geometric mean (**geometric.mean**), **Jaccard**, positive likelihood ratio (p+, LR(+)) or simply **L**, negative likelihood ratio

(p-, LR(-) or simply **lambda**), Matthews correlation coefficient (**MCC**), markedness (**MK**), negative predictive value (**NPV**), optimization precision **OP**, **precision**, **recall** (sensitivity), **specificity** and finally **Youden's** index. The function calculates the aforementioned metrics from a confusion matrix (contingency matrix) where *TP*, *TN*, *FP* *FN* are abbreviations for *true positives*, *true negatives*, *false positives* and *false negatives* respectively.

### Usage

```
ml_test(predicted, true, output.as.table = FALSE)
```

### Arguments

predicted	class labels predicted by the classifier model (a set of classes convertible into type factor with levels representing labels)
true	true class labels (a set of classes convertible into type factor of the same length and with the same levels as predicted)
output.as.table	the function returns all metrics except for <b>accuracy</b> and <b>error.rate</b> in a tabular format if this argument is set to <i>TRUE</i>

### Value

the function returns a list of following metrics:

<b>accuracy</b>	<i>calculated as: <math>(TP+TN) / (TP+FP+TN+FN)</math> (doesn't show up when output.as.table = TRUE)</i>
<b>balanced.accuracy</b>	<i>calculated as: <math>(TP / (TP+FN)+TN / (TN+FP)) / 2 = (\text{recall}+\text{specificity}) / 2</math></i>
<b>DOR</b>	<i>calculated as: <math>TP*TN / (FP*FN) = L / \text{lambda}</math></i>
<b>error.rate</b>	<i>calculated as: <math>(FP+FN) / (TP+TN+FP+FN) = 1-\text{accuracy}</math> (doesn't show up when output.as.table = TRUE)</i>
<b>F0.5</b>	<i>calculated as: <math>1.25*(\text{recall}*\text{precision}/(0.25*\text{precision}+\text{recall}))</math></i>
<b>F1</b>	<i>calculated as: <math>2*(\text{precision}*\text{recall} / (\text{precision}+\text{recall}))</math></i>
<b>F2</b>	<i>calculated as: <math>5*(\text{precision}*\text{recall} / (4*\text{precision}+\text{recall}))</math></i>
<b>FDR</b>	<i>calculated as: <math>1-\text{precision}</math></i>
<b>FNR</b>	<i>calculated as: <math>1-\text{recall}</math></i>
<b>FOR</b>	<i>calculated as: <math>1-\text{NPV}</math></i>
<b>FPR</b>	<i>calculated as: <math>1-\text{specificity}</math></i>
<b>geometric.mean</b>	<i>calculated as: <math>(\text{recall}*\text{specificity})^{0.5}</math></i>
<b>Jaccard</b>	<i>calculated as: <math>TP / (TP+FP+FN)</math></i>
<b>L</b>	<i>calculated as: <math>\text{recall} / (1-\text{specificity})</math></i>
<b>lambda</b>	<i>calculated as: <math>(1-\text{recall}) / (\text{specificity})</math></i>
<b>MCC</b>	<i>calculated as: <math>(TP*TN-FP*FN) / (((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))^{0.5})</math></i>

<b>MK</b>	<i>calculated as:</i> precision + NPV - 1
<b>NPV</b>	<i>calculated as:</i> TN / (TN+FN)
<b>OP</b>	<i>calculated as:</i> accuracy -  recall-specificity  / (recall+specificity)
<b>precision</b>	<i>calculated as:</i> TP / (TP+FP)
<b>recall</b>	<i>calculated as:</i> TP / (TP+FN)
<b>specificity</b>	<i>calculated as:</i> TN / (TN+FP)
<b>Youden</b>	<i>calculated as:</i> recall+specificity-1

**Author(s)**

G. Dudnik

**References**

1. Sasaki Y. (2007). The truth of the F-measure.:1–5. [https://www.researchgate.net/publication/268185911\\_The\\_truth\\_of\\_the\\_F-measure](https://www.researchgate.net/publication/268185911_The_truth_of_the_F-measure).
2. Powers DMW. (2011). Evaluation: from Precision, Recall and F-measure to ROC, Informedness, Markedness & Correlation. Arch Geschwulstforsch. 2(1):37–63. [https://www.researchgate.net/publication/313610493\\_Evaluation\\_From\\_precision\\_recall\\_and\\_fmeasure\\_to\\_roc\\_informedness\\_markedness\\_and\\_correlation](https://www.researchgate.net/publication/313610493_Evaluation_From_precision_recall_and_fmeasure_to_roc_informedness_markedness_and_correlation).
3. Bekkar M, Djemaa HK, Alitouche TA. (2013). Evaluation Measures for Models Assessment over Imbalanced Data Sets. J Inf Eng Appl. 3(10):27–38. <https://www.iiste.org/Journals/index.php/JIEA/article/view/7633>.
4. Jeni LA, Cohn JF, De La Torre F. (2013). Facing Imbalanced Data Recommendations for the Use of Performance Metrics. Conference on Affective Computing and Intelligent Interaction. IEEE. p. 245–51. <http://ieeexplore.ieee.org/document/6681438/>.
5. López V, Fernández A, García S, Palade V, Herrera F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Inf Sci. 250:113–41. <http://dx.doi.org/10.1016/j.ins.2013.07.007>.
6. Tharwat A. (2018). Classification assessment methods. Appl Comput Informatics . <https://linkinghub.elsevier.com/retrieve/pii/S2210832718301546>.

**Examples**

```
library(mltest)

# class labels ("cat", "dog" and "rat") predicted by the classifier model
predicted_labels <- as.factor(c("dog", "cat", "dog", "rat", "rat"))

# true labels (test set)
true_labels <- as.factor(c("dog", "cat", "dog", "rat", "dog"))

classifier_metrics <- ml_test(predicted_labels, true_labels, output.as.table = FALSE)

# overall classification accuracy
accuracy <- classifier_metrics$accuracy
```

```
# F1-measures for classes "cat", "dog" and "rat"
F1 <- classifier_metrics$F1

# tabular view of the metrics (except for 'accuracy' and 'error.rate')
classifier_metrics <- ml_test(predicted_labels, true_labels, output.as.table = TRUE)
```

# Index

\*Topic **utilities**

ml\_test, 1

ml\_test, 1