

ORES Metrics Documentation

In this documentation, an overview and explanation of the ORES damaging model fitness statistics, or simply *metrics*, is presented with the help of examples.

1 Confusion Matrix

As we are faced with the binary damaging classifier, there are four different classification cases:

1. Correctly classifying an edit as damaging - a *true positive*
2. Wrongly classifying an edit as damaging - a *false positive*
3. Correctly classifying an edit as good - a *true negative*
4. Wrongly classifying an edit as good - a *false negative*

A popular representation of those cases are confusion matrices as the one in Figure ???. The abbreviations of TP, FP, TN and FN will be used to denote the four mentioned cases.

2 Metrics Overview

An overview of the metrics is given by Figure ??, a table that comes with a quick definition and a value, if possible, based on the confusion matrix for each metric:

3 Example scenario

- Let's assume a total of 100 edits, of which 35 are damaging - an unreasonably high ratio of damaging edits, but useful for illustration purposes.

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Figure 1: Confusion matrix of a binary classifier.

Metric	Quick Definition	Value
recall	Ability to find all relevant cases	$\frac{TP}{TP+FN}$
precision	Ability to find only relevant cases	$\frac{TP}{TP+FP}$
f1	Harmonic mean of recall and precision	$2 * \frac{rec * prec}{rec + prec}$
fpr	Probability of a false alarm	$\frac{FP}{FP+TN}$
roc_auc	Measure of classification performance	
pr_auc	Measure of classification performance	
accuracy	Portion of correctly predicted data	$\frac{TP+TN}{Total}$
match_rate	Portion of observations predicted to be positive	$\frac{TP+FP}{Total}$
filter_rate	Portion of observations predicted to be negative	$1 - match_rate = \frac{TN+FN}{Total}$
!recall	Negated recall	$\frac{TN}{TN+FP}$
!precision	Negated precision	$\frac{TN}{TN+FN}$
!f1	Negated f1	$2 * \frac{!rec * !prec}{!rec + !prec}$

Figure 2: Metrics overview table.

- That leaves us with the following labels (or actual values): 35 posi-

tives and 65 negatives as visualized by Figure ??, where each edit is represented by one editor.

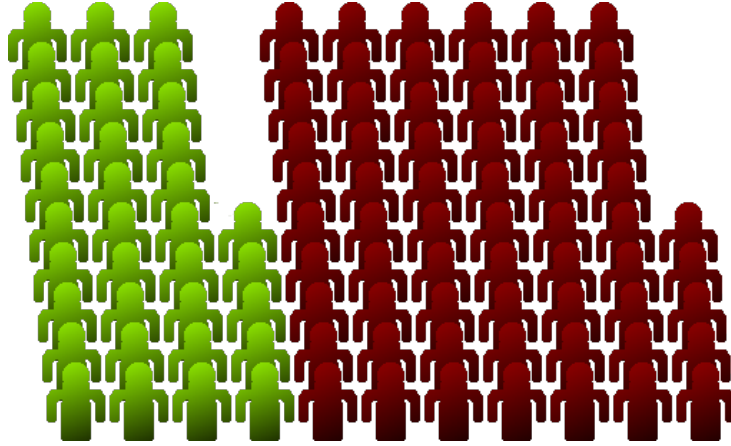


Figure 3: Total of 100 edits, represented by 100 editors, divided into actual positives in green and actual negatives in red.

- A binary classifier might now predict 40 positives, of which 30 actually are positive and 60 negatives of which 55 actually are negative. This also means that 10 non damaging edits have been predicted to be damaging and 5 damaging edits have been predicted not to be damaging. Figure ?? illustrates this state by marking predicted positives with a hazardous symbol ☢ and predicted negatives with a sun symbol ☀.
- Referring to the confusion matrix, we have

- 30 true positives ☢ (correctly predicted damaging edits)
- 5 false negatives ☢ (wrongly predicted damaging edits)
- 55 true negatives ☀ (correctly predicted non damaging edits)
- 10 false positives ☢ (wrongly predicted non damaging edits)

After establishing this example scenario, references to it will be included in the detailed descriptions of metrics in the next section.

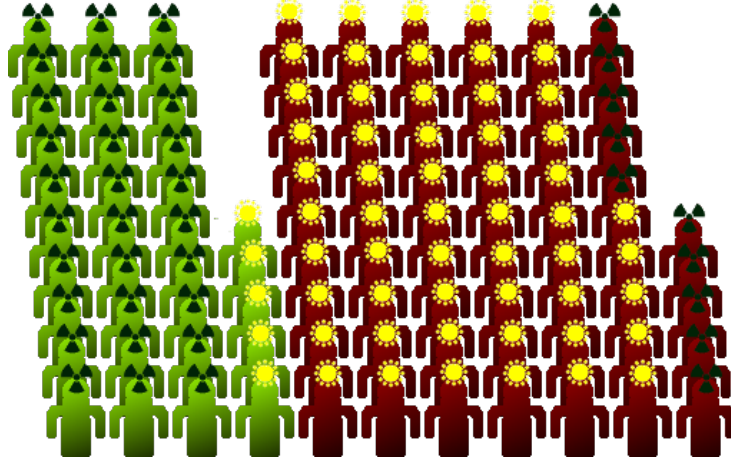



Figure 4: Edits divided into TP, FN, TN and FP.

4 Detailed definition of metrics

recall

- Recall ($\frac{TP}{TP+FN}$), true positive rate (tpr) or “sensitivity” is the ability of a model to find all relevant cases within the dataset.
- To us, relevant case means damaging edit: . The ability of the model to identify those depends on the ratio of actual positives being predicted as such:

$$\frac{\sum \text{green robot icon}}{\sum \text{green robot icon}}$$

with

$$\text{green robot icon} = \text{green robot icon with radiation symbol} + \text{green robot icon with star on head}$$

- In terms of numbers for our example that would be $\frac{30}{30+5} \approx 0.86$.

precision

- Precision ($\frac{TP}{TP+FP}$) or “specificity” is the ability of the model to find only relevant cases within the dataset.
- We are interested in how good the model is at only predicting edits to be damaging, that actually are. Therefore, we want the ratio of true

positives to all those predicted to be positive:

$$\frac{\sum \text{👤🟢}}{\sum \text{👤🟢} + \sum \text{👤🔴}} = \frac{30}{30 + 10} = 0.75$$

f1

- f1-score, the harmonic mean of recall and precision, a metric from 0 (worst) to 1 (best) is a possibility to evaluate the accuracy of a model
- It is defined by

$$2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

Note that, unlike the the average of recall and precision, the harmonic mean punishes extreme values.

- Referring to the example scenario, we get

$$2 * \frac{0.75 * \frac{30}{35}}{0.75 + \frac{30}{35}} = 0.8$$

fpr

- The false positive rate ($\frac{FP}{FP+TN}$) answers the question of ‘what is the portion, of all actual negatives, that are wrongly predicted?’ and can be described as the probability of a false alarm.
- In our example, a false alarm would be predicting an edit as damaging that isn’t. As a result we get

$$\frac{\sum \text{👤🔴}}{\sum \text{👤🔴} + \sum \text{👤🟡}} = \frac{10}{10 + 55} \approx 0.15$$

roc_auc

- The area under the ROC-curve, a measure between 0.5 (worthless) and 1.0 (perfect: getting no FPs), can be described as the probability of ranking a random positive higher than a random negative and serves as a measure of classification performance.

- The receiver operating characteristic (ROC) curve itself is used to visualize the performance of a classifier, plotting the TPR versus FPR as a function of the model's threshold for classifying a positive
- Assuming that we have had a threshold of 0.5 to get the previous results, one point on our ROC curve would be:

$$(\text{fpr}, \text{tpr}) = (0.15, 0.86).$$

Doing this for every threshold wanted results in the ROC curve. The area under the curve (auc) is a way of quantifying its performance.

pr_auc

- Similarly to the **roc_auc**, the area under the precision recall curve evaluates a classifiers performance. The main difference, however, is that, the PR curve plotting precision versus recall, does not make use of true negatives. It is therefore favourable to use **pr_auc** over **roc_auc** if true negatives are unimportant to the general problem or if there are a lot more negatives than positives, since differences between models will be more notable in the absence of a vast amount of true negatives in that second case.
- The point on the PR curve of our example for the standard threshold of 0.5 is

$$(\text{precision}, \text{recall}) = (0.75, 0.86)$$

To construct the PR-curve, it would be necessary to do this for every threshold wanted. Again, calculating the area under it is a way to quantify the curve's performance and therefore the model's performance as well.

accuracy

- Accuracy ($\frac{\text{TP}+\text{TN}}{\text{Total}}$) measures the ratio of correctly predicted data - positives and negatives.
- In the example, this is the proportion of correctly predicted damaging edits and correctly predicted non damaging edits to the total of edits and is given by

$$= \frac{\sum \text{👤} + \sum \text{👤}}{\sum \text{👤} + \sum \text{👤}} = \frac{30 + 55}{35 + 65} = 0.85$$

match_rate

- The match rate ($\frac{TP+FP}{Total}$) is the ratio of observations predicted to be positive.
- Concerning our damaging classifier, this is equal to wanting to know the ratio of edits predicted to be damaging, which is given by

$$\frac{\sum \text{👤} + \sum \text{👤}}{\sum \text{👤} + \sum \text{👤}} = \frac{30 + 10}{35 + 65} = 0.4$$

filter_rate

- The filter rate ($1 - \text{match_rate} = \frac{TN+FN}{Total}$) is the ratio of observations predicted to be negative. This is the complement to the match rate.
- In the example, the filter rate describes the ratio of edits predicted to be damaging, given by

$$1 - \text{match_rate} = \frac{\sum \text{👤} + \sum \text{👤}}{\sum \text{👤} + \sum \text{👤}} = \frac{55 + 5}{35 + 65} = 0.6$$

!<metric>

- Any <metric> with an exclamation mark is the same metric as without for the negative class:
 - !recall ($= \frac{TN}{TN+FP}$), the ability of a model to indentify all irrelevant cases (true negatives) as such
 - !precision $= \frac{TN}{TN+FN}$, the ability of a model to only predict irrelevant cases as such
 - !f1 $= 2 * \frac{!rec * !prec}{!rec + !prec}$, the harmonic mean of !recall and !precision
- Note that these metrics are also particularly useful for multi-class classifier as they permit queries to reference all but one class, e.g. in the ORES *itemquality* model, the recall for all classes except the “E” class comes down to the !recall of the “E” class.