

Get started

Open in app



towards
data science

Follow

603K Followers



8 Metrics to Measure Classification Performance

...explained in plain English



Rebecca Vickery

5 hours ago · 7 min read ★



Photo by [Mauro Gigli](#) on [Unsplash](#)

Classification is a type of supervised machine learning problem where the goal is to predict, for one or more observations, the category or class they belong to.

An important element of any machine learning workflow is the evaluation of the performance of

the model. This is the process where we use the trained model to make predictions on previously unseen, labelled data. In the case of classification, we then evaluate how many of these predictions the model got right.

In real-world classification problems, it is usually impossible for a model to be 100% correct. When evaluating a model it is, therefore, useful to know, not only how wrong the model was, but in which way the model was wrong.

“All models are wrong, but some are useful” , George Box

For example, if we are trying to predict if a tumour is benign or cancerous, we might be happier to trade off the model incorrectly predicting that a tumour is cancerous in a small number of cases. Rather than have the serious consequences of missing a cancer diagnosis.

On the flip side if we were a retailer deciding which transactions were fraudulent we might be happier for a small number of fraudulent transactions to be missed. Rather than risk turning away good customers.

In both of these cases, we would optimise a model to perform better for certain outcomes and therefore we may use different metrics to select the final model to use. As a consequence of these trade-offs when selecting a classifier there are a variety of metrics you should use to optimise a model for your specific use case.

In the following article, I am going to give a simple description of eight different performance metrics and techniques you can use to evaluate a classifier.

. . .

1. Accuracy

The overall **accuracy** of a model is simply the number of correct predictions divided by the total number of predictions. An accuracy score will give a value between 0 and 1, a value of 1 would indicate a perfect model.

$$accuracy = \frac{CorrectPredictions}{TotalPredictions}$$

Accuracy. Image by Author

This metric should rarely be used in isolation, as on imbalanced data, where one class is much larger than another, the accuracy can be highly misleading.

If we go back to the cancer example. Imagine we have a dataset where only 1% of the samples are cancerous. A classifier that simply predicts all outcomes as benign would achieve an accuracy score of 99%. However, this model would, in fact, be useless and dangerous as it would never detect a cancerous observation.

. . .

2. Confusion Matrix

A **confusion matrix** is an extremely useful tool to observe in which way the model is wrong

(or right!). It is a matrix that compares the number of predictions for each class that are correct and those that are incorrect.

In a confusion matrix, there are 4 numbers to pay attention to.

True positives: The number of positive observations the model correctly predicted as positive.

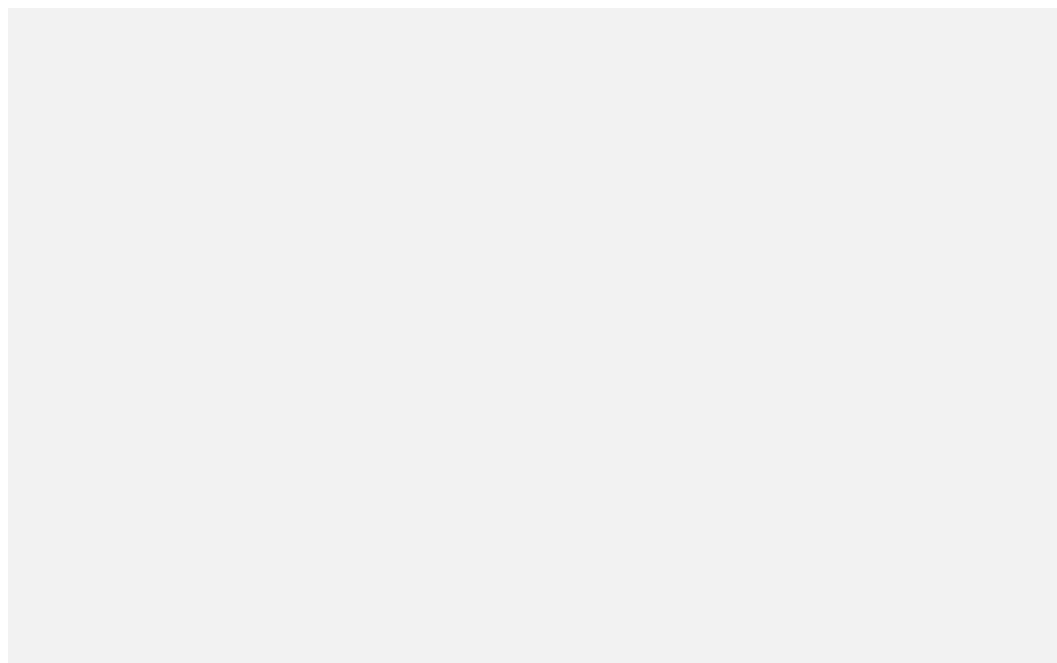
False-positive: The number of negative observations the model incorrectly predicted as positive.

True negative: The number of negative observations the model correctly predicted as negative.

False-negative: The number of positive observations the model incorrectly predicted as negative.

The image below shows a confusion matrix for a classifier. Using this we can understand the following:

- The model correctly predicted 3,383 negative samples but incorrectly predicted 46 as positive.
- The model correctly predicted 962 positive observations but incorrectly predicted 89 as negative.
- We can see from this confusion matrix that the data sample is imbalanced, with the negative class having a higher volume of observations.



Confusion matrix example (plotted using Pycaret). Image by Author

• • •

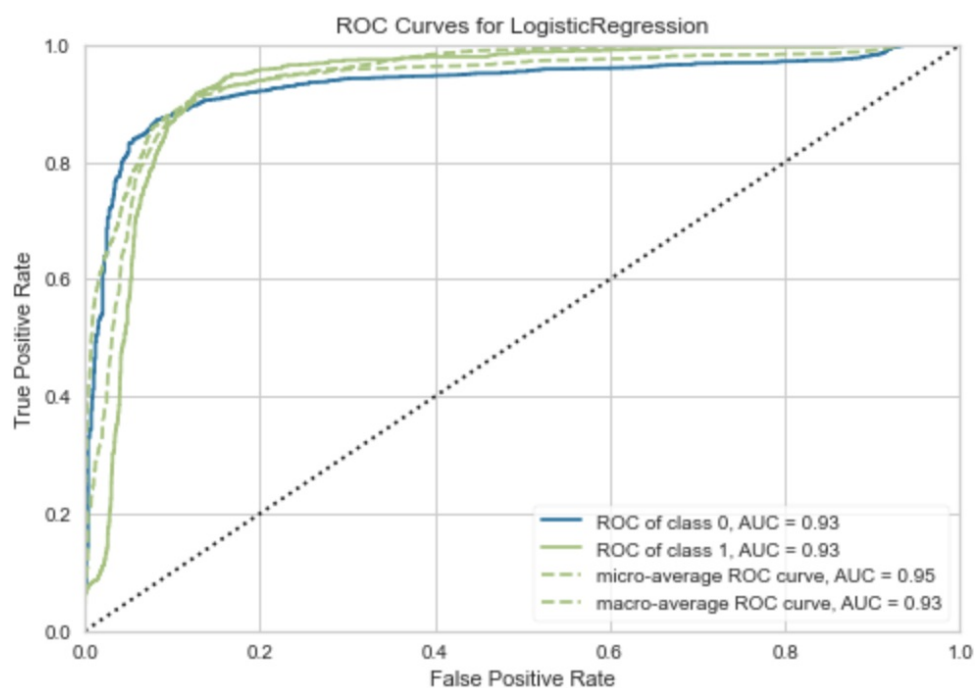
3. AUC/ROC

A classifier such as logistic regression will return the probability of an observation belonging to a

particular class as the prediction output. For the model to be useful this is usually converted to a binary value e.g. either the sample belongs to the class or it doesn't. To do this a classification threshold is used, for example, we might say that if the probability is above 0.5 then the sample belongs to class 1.

The **ROC** (Receiver Operating Characteristics) curve is a plot of the performance of the model (a plot of the true positive rate and the false positive rate) at all classification thresholds. The **AUC** is the measurement of the entire two-dimensional area under the curve and as such is a measure of the performance of the model at all possible classification thresholds.

ROC curves plot the accuracy of the model and therefore are best suited to diagnose the performance of models where the data is not imbalanced.



ROC curve example (plotted using Pycaret). Image by Author

• • •

4. Precision

Precision measures how good the model is at correctly identifying the positive class. In other words out of all predictions for the positive class how many were actually correct? Using alone this metric for optimising a model we would be minimising the false positives. This might be desirable for our fraud detection example, but would be less useful for diagnosing cancer as we would have little understanding of positive observations that are missed.

$$precision = TP / TP + FP$$

Precision. Image by author

• • •

5. Recall

Recall tell us how good the model is at correctly predicting **all** the positive observations in the dataset. However, it does not include information about the false positives so would be more useful in the cancer example.

$$recall = TP / TP + FN$$

Usually, precision and recall are observed together by constructing a precision-recall curve. This can help to visualise the trade-offs between the two metrics at different thresholds.

• • •

6. F1 score

The **F1 score** is the harmonic mean of precision and recall. The F1 score will give a number between 0 and 1. If the F1 score is 1.0 this indicates perfect precision and recall. If the F1 score is 0 this means that either the precision or the recall is 0.

$$F1 = 2 \times precision \times recall / precision + recall$$

F1 score. Image by Author

• • •

7. Kappa

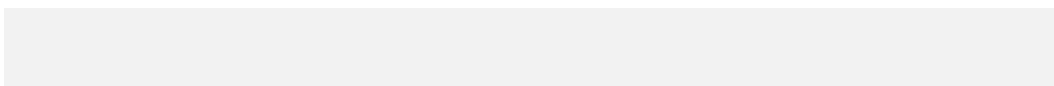
The **kappa** statistic compares the observed accuracy to an expected accuracy or the accuracy expected from random chance. One of the flaws of pure accuracy is that if a class is imbalanced then making predictions at random could give a high accuracy score. Kappa accounts for this by comparing the model accuracy to the expected accuracy based on the number of instances in each class.

Essentially it tells us how the model is performing compared to a model that classifies observations at random according to the frequency of each class.

$$kappa = (ObservedAccuracy - ExpectedAccuracy) / (1 - ExpectedAccuracy)$$

Kappa statistic. Image by Author

Kappa returns a value at or below 1, negative values are possible. One drawback of this statistic is that there is no agreed standard to interpret its values. Although, a general interpretation of the metric was given by Landis and Koch in 1977.



Kappa interpretation (Landis and Koch 19771). Image by Author

• • •

8. MCC

MCC (Matthews Correlation Coefficient) is generally considered one of the best measurements of performance for a classification model. This is largely because, unlike any of the previously mentioned metrics, it takes all possible prediction outcomes into account. If there are imbalances in the classes this will therefore be accounted for.

The MCC is essentially a correlation coefficient between the observed and predicted classifications. As with any correlation coefficient, its value will lie between -1.0 and +1.0. A value of +1 would indicate a perfect model.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Matthews Correlation Coefficient. Image by Author

• • •

In this article, we have covered simple explanations for eight metrics for measuring the performance of classification models. In practice, rarely should any of these metrics be used alone. More commonly a data scientist would assess a number of these scores and weigh up the trade-offs that they reveal when optimising a model.

Assessing the performance of a classifier is generally not straightforward and highly dependant on the use case and available dataset. It is particularly important to understand the risk of being wrong in a particular direction so that you can produce a truly useful model.

Thanks for reading!



19



[Data Science](#)[Machine Learning](#)[Artificial Intelligence](#)[Education](#)[Editors Pick](#)

More from Towards Data Science

[Follow](#)

Your home for data science. A Medium publication sharing concepts, ideas and codes.

[Read more from Towards Data Science](#)