


The Best Metric to Measure Accuracy of Classification Models

 clevertap.com/blog/the-best-metric-to-measure-accuracy-of-classification-models

Data Science



Jacob Joseph

November 28, 2016

Unlike evaluating the accuracy of models that predict a continuous or discrete dependent variable like Linear Regression models, evaluating the accuracy of a classification model could be more complex and time-consuming. Before measuring the accuracy of classification models, an analyst would first measure its robustness with the help of metrics such as AIC-BIC, AUC-ROC, AUC-PR, Kolmogorov-Smirnov chart, etc. The next logical step is to measure its accuracy. To understand the complexity behind measuring the accuracy, we need to know few basic concepts.

Model Output

Most of the classification models output a probability number for the dataset.

E.g. – A classification model like Logistic Regression will output a probability number between 0 and 1 instead of the desired output of actual target variable like Yes/No, etc.

The next logical step is to translate this probability number into the target/dependent variable in the model and test the accuracy of the model. To understand the implication of translating the probability number, let's understand few basic concepts relating to evaluating a classification model with the help of an example given below.

Goal: Create a classification model that predicts fraud transactions

Output: Transactions that are predicted to be Fraud and Non-Fraud

Testing: Comparing the predicted result with the actual results

Dataset: Number of Observations: 1 million; Fraud : 100; Non-Fraud: 999,900

The fraud observations constitute just **0.1%** of the entire dataset, representing a typical case of **Imbalanced Class**. Imbalanced Classes arises from classification problems where the classes are not represented equally. Suppose you created a model that predicted 95% of the transactions as Non-Fraud, and all the predictions for Non-Frauds turn out to be accurate. But, that high accuracy for Non-Frauds shouldn't get you excited since Frauds are just 0.1% whereas the Predicted Frauds constitute 5% of the observations.

Assuming you were able to translate the output of your model to Fraud/Non-Fraud, the predicted result could be compared to actual result and summarized as follows:

- a) **True Positives:** Observations where the actual and predicted transactions were fraud
- b) **True Negatives:** Observations where the actual and predicted transactions weren't fraud
- c) **False Positives:** Observations where the actual transactions weren't fraud but predicted to be fraud
- d) **False Negatives:** Observations where the actual transactions were fraud but weren't predicted to be fraud

Confusion Matrix is a popular way to represent the summarized findings.

True Positives (TP) False Negatives (FN)

False Positives (FP) True Negatives (TN)

Typically, a classification model outputs the result in the form of probabilities as shown below:

First 5 rows of the dataset:

Observation	Actual	Predicted
1	Non-Fraud	0.45
2	Non-Fraud	0.10
3	Fraud	0.67
4	Non-Fraud	0.60
5	Non-Fraud	0.11

Suppose we assume 0.5 as the cut-off probability i.e. observations with probability value of 0.5 and above are marked as Fraud and below 0.5 are marked as Non-Fraud as shown in the table below:

Accordingly, the above first 5 rows will be as below:

Observation	Actual	Predicted
1	Non-Fraud	Non-Fraud
2	Non-Fraud	Non-Fraud
3	Fraud	Fraud
4	Non-Fraud	Fraud
5	Non-Fraud	Non-Fraud

Let's summarize the results from the model of the entire dataset with the help of the confusion matrix:

TP = 90 FN = 10

FP = 10 TN = 999,890

We have all non-zero cells in the above matrix. So is this result ideal?

Wouldn't we love a scenario wherein the model accurately identifies the Frauds and the Non-Frauds i.e. zero entry for cells, FP and FN?

A BIG YES.

Consider a scenario wherein as a marketing analyst; you would like to identify users who were likely to buy but haven't bought yet. This particular class of users would be the ones who share the characteristics of the users who bought. Such a class would belong to False Positives – Users who were predicted to transact but didn't transact in reality. Hence, in addition to non-zero entries in TP and TN, you would prefer a non-zero entry in FP too. Thus, the model accuracy depends on the goal of the prediction exercise.

Key Testing Metrics

Since we are now comfortable with the interpretation of the Confusion Matrix, let's look at some popular metrics used for testing the classification models:

i) Sensitivity/Recall

Sensitivity also known as the True Positive rate or Recall is calculated as,

$$\text{Sensitivity} = \frac{\text{Number of True Positives}}{\text{Number of True Positives} + \text{Number of False Negatives}}$$

Since the formula doesn't contain FP and TN, Sensitivity may give you a biased result, especially for imbalanced classes.

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

In the example of Fraud detection, it gives you the percentage of Correctly Predicted Frauds from the pool of Actual Frauds.

$$\text{Sensitivity} = \frac{90}{90 + 10} = 0.90$$

ii) Specificity

Specificity, also known as True Negative Rate is calculated as,

$$\text{Specificity} = \frac{\text{Number of True Negatives}}{\text{Number of True Negatives} + \text{Number of False Positives}}$$

Since the formula does not contain FN and TP, Specificity may give you a biased result, especially for imbalanced classes.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

In the example of Fraud detection, it gives you the percentage of Correctly Predicted Non-Frauds from the pool of Actual Non-Frauds.

$$Specificity = \frac{999,890}{999,890 + 10} = 1$$

iii) Precision

Precision also known as Positive Predictive Value is calculated as,

$$Precision = \frac{\text{Number of True Positives}}{\text{Number of True Positives} + \text{Number of False Positives}}$$

Since the formula does not contain FN and TN, Precision may give you a biased result, especially for imbalanced classes.

$$Precision = \frac{TP}{TP + FP}$$

In the example of Fraud detection, it gives you the percentage of Correctly Predicted Frauds from the pool of Total Predicted Frauds.

$$Precision = \frac{90}{90 + 10} = 0.90$$

iv) F1 score

F1 score incorporates both Recall and Precision and is calculated as,

The F1 score represents a more balanced view compared to the above 3 metrics but could give a biased result in the scenario discussed later since it doesn't include TN.

$$F_1 \text{ score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F_1 \text{ score} = 2 * \frac{0.90 * 0.90}{0.90 + 0.90} = 0.90$$

v) Matthews Correlation Coefficient (MCC)

Unlike the other metrics discussed above, MCC takes all the cells of the Confusion Matrix into consideration in its formula.

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

Similar to Correlation Coefficient, the range of values of MCC lie between -1 to +1. A model with a score of +1 is a perfect model and -1 is a poor model. This property is one of the key usefulness of MCC as it leads to easy interpretability.

$$MCC = \frac{90 * 999,890 - 10 * 10}{\sqrt{(90 + 10) * (90 + 10) * (999,890 + 10) * (999,890 + 10)}} = 0.90$$

Metric Comparison

We will test and compare the result of the classification model at few probability cut-off values using the above-mentioned testing metrics.

Scenario A: Confusion Matrix at cut-off value of 0.5

We shall take this scenario (cut-off value of 0.5) as the base case and compare the result of the base case with different cut-off values.

Confusion Matrix

TP = 90 FN = 10

FP = 10 TN = 999,890

Testing Metrics

Sensitivity	Specificity	Precision	F1 Score	MCC
0.90	1.00	0.90	0.90	0.90

Scenario B: Confusion Matrix at cut-off value of 0.4

Confusion Matrix

TP = 90 FN = 10

FP = 1910 TN = 997,990

It can be clearly observed that for Scenario B, there is a substantial increase in FP compared to Scenario A. Hence, there should be deterioration in the metrics.

Testing Metrics

Sensitivity	Specificity	Precision	F1 Score	MCC
0.90	1.00	0.05	0.09	0.20

There is no change in Sensitivity & Specificity, which is constant.

Scenario C: Confusion Matrix at cut-off value of 0.6

Confusion Matrix

TP = 90 FN = 1910

FP = 10 TN = 997,990

There is a substantial increase in FN compared to Scenario A. Hence, there should be deterioration in the metrics compared to A.

Testing Metrics

Sensitivity	Specificity	Precision	F1 Score	MCC
0.05	1.00	0.90	0.09	0.20

Here there is no change in Specificity & Precision while there is a general decline in other metrics.

Based on our findings, we can say that F1 score and MCC is making more sense compared to Sensitivity and Specificity.

In the example, we have built a model to predict Fraud. We can use the same model to predict Non-Fraud. In such a case, the Confusion Matrix will be as given below:

Scenario D: Confusion Matrix at cut-off value of 0.5

Confusion Matrix

TP = 999,890 FN = 10

FP = 10 TN = 90

The above confusion matrix is just the transpose of the matrix given in Scenario A since the model is predicting Non-Frauds instead of Frauds. So the True Negatives in Scenario A will be the True Positives for Scenario D, likewise for other cells. Ideally, the testing metrics should be the same for Scenario A and D.

Testing Metrics

Sensitivity	Specificity	Precision	F1 Score	MCC
1	0.90	1	1	0.90

Except for MCC all the other testing metrics have changed.

Summary of Testing Metrics for all the scenarios:

Scenario	Sensitivity	Specificity	Precision	F1 Score	MCC
A	0.90	1.00	0.90	0.90	0.90
B	0.90	1.00	0.05	0.09	0.20
C	0.05	1.00	0.90	0.09	0.20
D	1	0.90	1	1	0.90

Conclusion

As an analyst, if you are looking at a metric to measure and maximize the overall accuracy of the classification model, MCC seems to be the best bet since it is not only easily interpretable but also robust to changes in the prediction goal.

Make your apps smarter

Experience the benefits of using a single product for your app analytics and user engagement.

[Schedule a Demo Now!](#)

Popular Tags

Related Posts

[Leading vs Lagging Indicators: Using the Right KPIs for App Marketing](#)

[RFM analysis for Customer Segmentation](#)

[Why We Chose Sunbursts Over Sankey Charts to Depict User Journeys](#)

[Visualizing and Comparing Trends with Vastly Varying Scale](#)

[Funnel Analysis: How Funnel Analytics Can Increase Conversions](#)

