

Logistic Regression
CONFUSION MATRIX
and
More Metrics
Arunkumar Nair

Accuracy and Other Metrics

- When there is a class imbalance, accuracy can be misleading too.
- For Imbalanced Data

Accuracy and Other Metrics

For a binary classification problem, we will know our positive class (like spam, fraud, has cancer, etc) and hence we can focus on the scores for the positive class.

But, for a multiclass classification problem, apart from the class-wise

recall,

precision,

f1 scores,

Case-Study Data

As an example, let's assume that our model predicted the fruit types- Apple (A), Banana(B) and Custard apple(C) and gave the confusion matrix.
Apple (A), Banana(B) and Custard apple(C):

Total number of observations in the data
(i.e., fruits) = 25

A=10

B=8

C=7

Case-Study Data

Apple (A), Banana(B) and Custard apple(C):

Total number of observations in the data (i.e., fruits) = 25

Number of A, B, C type fruits the data has = 10, 8, 7 respectively







Model saw more instances of apples (A = 10) than other fruit types

The diagonal observations are the true positives of each class i.e., the number of times the model correctly identified A as A, B as B and C as C

All other non-diagonal observations are incorrect classifications made by the model

Let's rebuild the same confusion matrix in python so that the metric values can be validated at each step.

Confusion Matrix

		Predicted (what our model says))			
Actual (what the data says)	CLASSES	 A	 B	 C	Row totals
	 A	5	2	3	10
	 B	2	6	0	8
	 C	3	2	2	7
	Column Totals	10	10	5	25

Diagonal numbers are rightly classified observations

Total number of observations/ records

Total number of observations in the data

(i.e., fruits) = 25

A=10

B=8

C=7

Accuracy:

Tells us how often the model can be correct.

Accuracy of the model = No. of correct predictions by the model / total observations
= Sum (diagonal values of confusion matrix) / Total observations
= $(5+6+2) / 25$
= $13/25 = 0.52$

Check in python using sklearn:

Model accuracy

```
metrics.accuracy_score(actuals,predicted)
```

0.52

Accuracy is not always Accurate

When there is a class imbalance, accuracy can be misleading too.

Let's assume that we have a sample of 100 fruits of which 90 are apples and 5 are bananas and 5 are custard apples. Even if our model predicts all the fruits as apples, the accuracy will be 90% while the truth is that our model is not actually doing a great job!

Micro Averages

Hence, the need for other metrics:

Accuracy is calculated for the model as a whole but recall and precision are calculated for individual classes.

We use macro or micro or weighted scores of

- Recall
- Precision
- F1 score

of a model for multiclass classification problems.

Micro Averages

- In the Micro-average method, you sum up the individual class's true positives, false positives, and false negatives of the system for different sets and then apply them to get the score.

Micro Average Recall Score

- For the fruit's confusion matrix, micro-average recall score is calculated as below (same as in the classification report above):

Micro average recall

sum of true positives of class A,B,C over sum of true positives of class A,B,C and False positives of A,B,C

$$(5+6+2) / ((5+6+2) + (5+4+3))$$

0.52

Macro Average Score-Recall

Macro average recall

Mean of recall scores of all classes

```
: (0.5+0.75+0.29)/3
```

```
: 0.5133333333333333
```

Weighted Average Score-Recall

Weighted average recall

Sum of recall scores of all classes after multiplying the scores with their respective class proportions

$$(10 * 0.5 + 8 * 0.75 + 7 * 0.29) / 25$$

0.5212

Recall

Recall (also called True Positive Rate or Sensitivity):

As an example, let us calculate recall for class B.

Recall for B would be, **from all the actual instances of class B (banana), how often it correctly predicts as B (banana)**

Recall for B = out of the total Bs (bananas = 8) in the data, how many Bs did the model identify correctly (6).

So, it would be $6/8 = 0.75$ (which is the same as in the classification report above for class B)

Precision:

Precision for class B is: **how often is the model correct when it predicts as B?**

As earlier, let's calculate precision for class B.

Precision for B: out of the total observations predicted as Bs (10) by the model, how many are correct Bs (6). So, it would be $6/10 = 0.6$

F1-Score

F1 score of class B will be

$$(2 \times \text{Recall} \times \text{precision}) / (\text{Recall} + \text{Precision})$$
$$= 0.75 \times 0.6 \times 2 / (0.75 + 0.6) = 0.67$$

Precision vs Recall

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

Precision:

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

True Positive + False Positive = Total **Predicted** Positive

$$\begin{aligned}\text{Precision} &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \\ &= \frac{\text{True Positive}}{\text{Total Predicted Positive}}\end{aligned}$$

Precision for class B is: **how often is the model correct when it predicts as B?**

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

True Positive + False Positive = Total **Predicted** Positive

$$\begin{aligned}
 \text{Precision} &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \\
 &= \frac{\text{True Positive}}{\text{Total Predicted Positive}}
 \end{aligned}$$

- Immediately, you can see that Precision talks about how precise/accurate your model is out of those predicted positive, how many of them are actual positive.
- Precision is a good measure to determine, **when the costs of False Positive is high**. For instance, email spam detection. In email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam). The email user might lose important emails if the precision is not high for the spam detection model.

Recall

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

True Positive + False Negative = Total **Actual** Positive

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$= \frac{\text{True Positive}}{\text{Total Actual Positive}}$$

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

True Positive + False Negative = Total **Actual** Positive

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

$$= \frac{\text{True Positive}}{\text{Total Actual Positive}}$$

- So Recall actually calculates how many of the Actual Positives our model capture through labeling it as Positive (True Positive). Applying the same understanding, we know that Recall shall be the model metric we use to select our best model when there is a **high cost associated with False Negative**.
- For instance, in fraud detection or sick patient detection. If a fraudulent transaction (Actual Positive) is predicted as non-fraudulent (Predicted Negative), the consequence can be very bad for the bank.
- Similarly, in sick patient detection. If a sick patient (Actual Positive) goes through the test and predicted as not sick (Predicted Negative). The cost associated with False Negative will be extremely high if the sickness is contagious.

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

True Positive + False Positive = Total Predicted Positive

$$\begin{aligned}
 \text{Precision} &= \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \\
 &= \frac{\text{True Positive}}{\text{Total Predicted Positive}}
 \end{aligned}$$

- Immediately, you can see that Precision talks about how precise/accurate your model is out of those predicted positive, how many of them are actual positive.
- Precision is a good measure to determine, when the costs of False Positive is high. For instance, email spam detection. In email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam). The email user might lose important emails if the precision is not high for the spam detection model.

F1-Score

- F1 Score is needed when you want to seek a balance between Precision and Recall.
Right...so what is the difference between F1 Score and Accuracy then?

F1-Score

- We have previously seen that accuracy can be largely contributed by a large number of True Negatives which in most business circumstances, we do not focus on much whereas False Negative and False Positive usually has business costs (tangible & intangible) thus F1 Score might be a better measure to use if we need to seek a balance between Precision and Recall AND there is an uneven class distribution (**large number of Actual Negatives**).

Thanks

End