
ubiqum code academy
Confusion Matrix
Let's Go

What is it?

The confusion matrix is a table used to describe the performance of a classifier (classification model) on a set of test data for which the true values are known.

The values shown in the matrix are used to calculate the metrics used to choose between models.

Exploring the matrix

True Positive: actual cases of the positive class that have been correctly predicted
True Negative: actual cases of the negative class that have been correctly predicted
False Positive: actual cases of the negative class that have been predicted as positive
False Negative: actual cases of the positive class that have been predicted as negative

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Accuracy: How often is the classifier correct?

(TP + TN)/TOTAL CASES
=
(100 + 50)/165
=
0.91

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Error Rate: How often is the classifier wrong?

(FP + FN)/TOTAL CASES
=
(10 + 5)/165
=
0.09

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Recall or Sensitivity: How many actual cases of the positive class have been predicted correctly?

(TP)/ACTUAL YES
=
(100)/105
=
0.95

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Precision: How many cases predicted as belonging to the positive class are correct?

TP/PREDICTED YES
=
100/110
=
0.91

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Specificity (true negative reate): how many actual cases belonging to the negative class have been predicted correctly?

TN/ACTUAL NEGATIVE
=
50/60
=
0.83

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

F-score: How to compare two models with high precision and low recall or vice versa?

$$\text{F-score} = \frac{(2*\text{Recall}*\text{Precision})}{(\text{Recall} + \text{Precision})}$$

Spam Filter Example

Let’s imagine your boss askes you to develop a model to detect whether an incoming email is legit or spam. If labelled as legit, the email goes into the inbox, if labelled as spam, it goes into the spam folder.

Which of the following two models would you go for and why??

Spam Filter: Which metric should the model be optimized for?

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Model 1

Precision: 0.91

Accuracy: 0.91

Specificity: 0.83

Spam Filter: Which metric should the model be optimized for?

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 58	FP = 2	60
Actual: YES	FN = 15	TP = 90	105
	73	92	

Model 2

Precision: 0.98

Accuracy: 0.90

Specificity: 0.97

To recap:

- The confusion matrix describes the performance of a classifier
- To choose among models, the metrics calculated using the values of the matrix should be used
- The metric which the model should be optimized for should be chosen depending on the objective of the analysis (what's the scenario that would make you boss happier?!)
- Deciding which class of the categorical variable will represent the 'positive class' is utlimately an arbitrary choice (NOTE: using the positive/negative class nomenclature on polytomous variable is not encouraged as it would make the matrix very hard to analyse)

Resources

- https://www.youtube.com/watch?v=8Oog7TXHvFY
- https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/

Thanks