Demystifying the Confusion Matrix

Dr. Stephen Hill

When we work with classification problems one of our main challenges is assessing the quality of a model. Common metrics for model evaluatio include: Accuracy, Sensitivity, and Specificity. To help us easily calculate these values we use a confusion matrix.

The Confusion Matrix

The confusion matrix is a table that shows the number of correct and incorrect classifications. A typical structure for a confusion matrix is shown in Table 1 below.

Table 1: Confusion Matrix Format

		Data
Predicted	Event	No Event
Event	True Positive	False Positive
No Event	False Negative	True Negative

From the confusion matrix we can then calculate Accuracy, Sensitivity, and Specificity.

Note that the display of the confusion matrix can differ. For example, you may see the predicted values in the columns and the data in the rows or "event" might be listed before "no event". Be cautious and carefully identify which elements in the table are the predictions and the data and in which order the predictions and data are displayed.

If we are using logistic regression class predictions (via a probability threshold), we manually create a confusion matrix with code similar to that below (assume that train$response is the response variable, predictions is an R object that stores the logistic regression predicted probabilities, and threshold is the probability threshold. This puts the predictions in the columns and the data in the rows. Flip the order if you wish.

```
table(train$response, predictions > threshold)
```

If you are using the R *caret* package and a method that produces direct class predictions (classification trees for example), the R code is then:

```
confusionMatrix(predictions, train$response)
```

Note that this approach does not work with logistic regression unless you have properly prepared the predictions to be displayed as predicted classes rather than predicted probabilities.

Accuracy

Accuracy is calculated as:

$$\frac{TP + TN}{TP + TN + FP + FN}$$

Simply put, accuracy is the ratio of the correct predictions to the number of total predictions. Accuracy can range from 0 to 1, with 1 being perfect. Accuracy is an easy measure to calculate and is commonly used to evaluate classification model quality. However, accuracy can be deceptive in the situation where the dataset is imbalanced. An imbalanced dataset is characterized by one of the classes dominating the other(s). [1] For example, in an insurance fraud dataset, fraud is actually fairly rare, so nearly all claims will be "Not Fraud". The accuracy of a *naive* model that simply predicts "Not Fraud" for all claims will then be very high.

[1] Beware of accuracy as misleading with an imbalanced dataset.

Sensitivity

Sensitivity is calculated as:

$$\frac{TP}{TP + FN}$$

Sensitivity captures how well a classification model identifies "positive" observations. For example, a model that naively classifies every observation as positive would have a sensitivity of 1. A highly sensitive model (even if it incorrectly labels "negative" observations as "positive") may be desirable in some applications. Such applications may include security and healthcare where the "cost" of labeling an observation as "negative" (safe or healthy) when the observation is actually "positive" (dangerous or sick) could be high.

Specificity

Specificity is calculated as:

$$\frac{TN}{TN + FP}$$

Specificity captures how well a classification model works for "negatives" as sensitivity does for "positives". In the insurance fraud example mentioned above, a naive model that classifies all claims as "Not Fraud" would have a specificity of 1.

Example

Let's say that we have built a model to predict whether or not a patient (given certain medical data) is likely to develop diabetes. Our

two classes in this problem are "Diabetes" and "No Diabetes". The confusion matrix will look as displayed in Table 2 below.

Table 2: Confusion Matrix Format

	Data	
Predicted	Diabetes	No Diabetes
Diabetes	True Positive	False Positive
No Diabetes	False Negative	True Negative

In our data we have 1,000 observations. Let's assume that 700 of these are will not develop diabetes so they are "No Diabetes" with the remaining 300 labeled "Diabetes". Assume that our model yields the confusion matrix shown in Table 3 below.

Table 3: Confusion Matrix Format

	Data	
Predicted	Diabetes	No Diabetes
Diabetes	175	100
No Diabetes	125	600

The accuracy of this model is:

$$\frac{175 + 600}{175 + 100 + 125 + 600} = 0.775$$

The sensitivity of this model is:

$$\frac{175}{175 + 125} = 0.583$$

The specificity of this model is:

$$\frac{600}{600 + 100} = 0.857$$

The accuracy of a naive model that predicts that all observations will be in the majority class would be:

$$\frac{600}{175 + 100 + 125 + 600} = 0.6$$