Different Measures to Evaluate Classifiers

For model performance classification accuracy, i.e., the number of correct predictions as a percentage of all predictions made is not enough. Other scores are discussed.

The robustness of a model for making predictions on unseen data is ensured by using multiple cross validation. We used classification accuracy, its average and divergence. But, depending on the problem, there are other more relevant aspects to consider. Once you have a model that you believe can make robust predictions you need to decide whether it is a good enough model to solve your problem. Classification accuracy alone is typically not enough information to make this decision.

Figure 1: Squash male and female flowers. To classify them is a binary classification problem.

We will introduce two important terms here: Precision and Recall as performance measures. This is often used to evaluate the classifier for a binary classification problem.

In the squash male/female flower classification, Actual female (positive) = 50 samples, Actual Male (negative) = 50 samples. Out of that, our model classifier 41 as female which are actually female, and it misclassified 9 females as male. Similarly, out of 50 males, the classifier correctly classified 46 males as male flowers. 4 male flowers are misclassified as female (positive).

100 flower data, 50 female (positive) and 50 male (negative). The problem is to classify them correctly.	Actual positive (female)	Actual negative (male)	
Predicted by model female (positive)	True positive (TP) = 41	False positive (FP) = 4	41+4 = 45
Predicted by model negative (male)	False negative (FN) = 9	True negative (TN) = 46	46+9 = 55
	41+9 = 50	4+46 = 50	

Example: Breast Cancer data

The UCI breast cancer dataset have 9 attributes, like age, menopause, tumor size, left/right breast, etc. There are two classes, no-recurrence and recurrence in 5 years. Total samples are 286 from women. This is a binary classification problem. Of the 286 women, for 201 there was no recurrence, and for rest 85 it did. The classifier takes the attributes as input and predict whether there will be recurrence of cancer or not. We define recurrence as positive and no-recurrence as negative.
Depending on the problem, either false negative or false positive becomes more important. In this case, assuring someone that there will be no-recurrence (negative) but actually cancer recurred is worse. In other words, false negative (FN) is bad. A Classifier should be tuned to reduce FN.

Different parameters for evaluating a classifier

1) ***Classification Accuracy***
   Classification accuracy is the number of correct predictions made divided by the total number of predictions made (multiplied by 100 to convert into percentage). A trivial classifier, which predicts no-recurrence for all samples, will still have a high classification accuracy of (201/286)*100% = 70.28%. But, it fails to do what it is supposed to do – predict recurrence so that the patients can take regular check-up, and preventive measures. On the other hand, if the classifier predicts everyone as recurrence case, accuracy will be (85/286)*100 = 29.72%, a very poor result. Safe, but no good.
   Now suppose the classifier result is as shown in the following confusion matrix:

286 cancer patient data, 9 attributes, 86 recurred (positive); 201 no-recurrence (negative).	Actual positive (recurrence)	Actual negative (no-recurrence)	
Predicted by model recurrence (positive)	True positive (TP) = 75	False positive (FP) = 13	75+13 = 88
Predicted by model no-recurrence (negative)	False negative (FN) = 10	True negative (TN) = 188	10+188 = 198
	10+75 = 85	13+188 = 201	

Out of 85 recurrence cases, the classifier correctly predicts 75; it predicts another 13 as recurrence though they are not. Similarly, out of 201 no-recurrences, it classifies 188 correctly; 10 recurrence cases it misclassifies as no-recurrence. Thus, the accuracy is ((TP+TN)/total population)*100=((75+188)/286)*100 = 91.9%. There are other ways to judge the quality of the classifier, using terms like precision, recall, f-score, and ROC-curve (receiver operating curve) – to see how the performance of a binary classifier changes with tuning a threshold of the classifier.

2) ***Precision or PPV:***

Precision or positive predictive value (PPV) is TP/(TP+FP), i.e., when the classifier says it is positive, how reliable is the model, what percentage of times it is true when it declares a sample as positive. In other words, it is the percentage of "true positives" divided by "the total number of positives" as predicted by the model. In the above example, 75/(75+13)=0.85 or 85% is the precision.

3) ***Recall or Sensitivity or True Positive Rate:***

Recall on the other hand is TP/(TP+FN), i.e., the percentage of real positives that are identified by the model, what percentage of actual positives could be recalled by the model. (TP+FN) are the actual number of positives, out of which TP are identified as positive correctly. In the above example, 75/(75+10)=0.88 or 88% is the recall. ***Specificity (SPC)*** is defined as True negative rate, TN/(TN+FP) = TN/(total actual negative). ***False Positive Rate*** is FP/(total actual negative) = FP/(TN+FP) = ***1 – Specificity***.

Examples *(from Wiki):* Suppose a picture contains 9 dogs and some cats. A classifier identifies 7 dogs, of which 4 are real dogs and 3 cats falsely identified as dog. Then precision is 4/7 and recall is 4/9. OR – A search engine returns 30 pages out of which 20 are relevant. In fact, in total there are 60 pages which are relevant out of which only 20 are found. Then precision is 20/30 and recall is 20/60. Precision is how reliable the search result is, and recall is good its searching capability is.

4) ***F1 Score or F Score or F Measure:***

Depending on application of the model, either precision or recall may be more important. We adjust the classifier threshold to achieve required value of precision or recall on test samples. There are a host of problems, where we need a balance of the two, both precision and recall. Then we would like to maximize "F Score" defined as 2*((precision*recall) / (precision + recall)).

5) ***F_β Measure:***

$$F_\beta = (1 + \beta^2).\frac{precision.recall}{\beta^2.precision + recall}$$

Clearly, when $\beta > 1$ recall is given more importance than precision, and when $\beta < 1$ precision is given more importance. Let us see what happens when,

precision is 0.7 and recall is 0.8. When $\boldsymbol{\beta}$ = 2, $\boldsymbol{F_2 = 0.\dot{7} \approx 0.8}$,i.e., very near to recall value, whereas when $\boldsymbol{\beta}$ = 0.5, $\boldsymbol{F_{0.5} = 0.717 \approx 0.7}$, i.e., very near to precision value. For β = 1, $\boldsymbol{F_1 = 0.74\dot{6} \approx 0.75}$, which is the average of *precision and recall.* F-measure was introduced by Van Rijsbergen (1979), so that $\boldsymbol{F_\beta}$ "measures the effectiveness of retrieval with respect to a user who attaches $\boldsymbol{\beta}$ times as much importance to recall as precision."

6) ***ROC Curve:***

ROC (Receiver Operating Characteristic) is to see the relative performance of precision and recall of the classifier, as the decision threshold is varied.

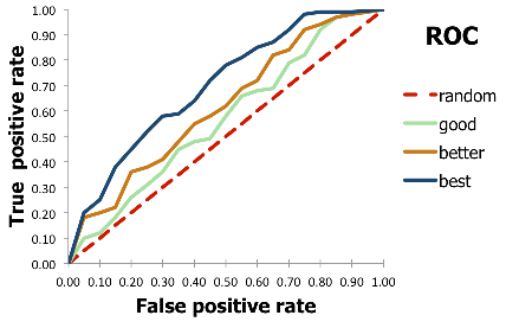
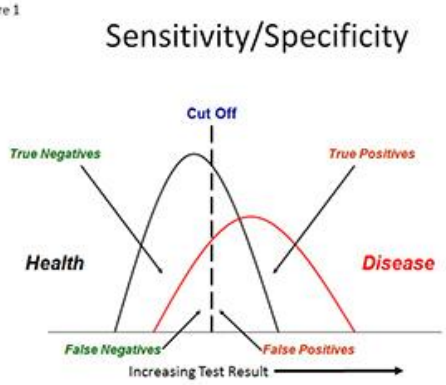


Figure 1

On the left is a ROC curve for three classifiers. The x-axis is False positive rate, i.e., FP/(total actual negative), and the y-axis is True positive rate, i.e., TP/(total actual positive). The red dashed line is what happens when the classification is blind/random. The blue line is the best classifier. Thus, when we have different classifiers to choose from, we plot their performances as ROC, and evaluate them. From ROC curve we can also judge the optimum threshold.



In the figure below, is the distribution of disease and healthy samples' distribution. We can shift the "cut off", i.e., the threshold and draw ROC curve.