Performance evaluation of binary classifiers

Kevin P. Murphy

Last updated October 10, 2007

1 ROC curves

We frequently design systems to detect events of interest, such as diseases in patients, faces in images, spam in an email queue, etc. We often want to compare the performance of such systems. One way to do this is to use a **ROC** (receiver operating characteristic) curve. This is a way of evaluating expected loss using a range of possible loss functions, where we vary the ratio of costs of false positives and false negatives. Below we summarize the main ideas. See [Faw03] and [MRS08, ch8] for details.

Let S_i be the score assigned to object x_i, where the score is proportional to our belief (confidence) that x_i is a positive instance (e.g., x_i has the disease/ is a face/ is spam), which we denote by $y_i = 1$. If x_i is not positive, it is negative, denoted by $y_i = 0$. Note that S_i does not have to be the probability $p(y_i = 1|x_i)$, but should be monotonically related to it. For any given threshold θ, we can convert S_i into an estimated label (best guess), by setting $\hat{y}_i = 1$ (positive) if $S_i > \theta$, otherwise $\hat{y}_i = 0$ (negative). Let y_i be the true label. By comparing $\hat{y}_i$ to y_i we can evaluate the quality of the system in terms of the number of errors it makes. If we use a low threshold, we will detect lots of events, but many will be wrong (false alarms); conversely if we use a high threshold, we may not detect many true events.

This is illustrated in Table 1. We see that the model is very confident that examples 1–3 are positive, and examples 7–9 are negative. It is not very sure about examples 4–6 (since the probabilities are all near 0.5), but nevertheless it is possible to find a threshold (namely $\theta = 0.5$) that perfectly separate the classes into positive and negative. Obviously if $\theta = 0$, all the examples are classified as positive, and conversely, if $\theta = 1$, all the examples are classified as negative.

In Table 2, we see the output of another estimate of $p(y_i|x_i)$, perhaps using a different model. Its behavior is similar to the previous model, except it gets examples 5 and 6 wrong, i.e., it assigns too little probability to the event $y_5 = 1$ and too much to $y_6 = 1$. Consequently it makes some errors when using a threshold of $\theta = 0.5$. In fact, it is easy to see that there is no threshold that will perfectly reproduce the vector of true labels, $y = (y_1, \ldots, y_n)$.

For any threshold θ, we can compute how many entities we correctly and incorrectly classify. This gives rise to four numbers. TP is the number of true positives, i.e., how many entities are "called" as positive which actually are

i	y_i	$p(y_i = 1|x_i)$	$\hat{y}_i(\theta = 0)$	$\hat{y}_i(\theta = 0.5)$	$\hat{y}_i(\theta = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	0.5	1	1	0
6	0	0.4	1	0	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0

Table 1: Example output from classifier 1.

i	y_i	$p(y_i = 1|x_i)$	$\hat{y}_i(\theta = 0)$	$\hat{y}_i(\theta = 0.5)$	$\hat{y}_i(\theta = 1)$
1	1	0.9	1	1	0
2	1	0.8	1	1	0
3	1	0.7	1	1	0
4	1	0.6	1	1	0
5	1	**0.2**	1	**0**	0
6	0	**0.6**	1	**1**	0
7	0	0.3	1	0	0
8	0	0.2	1	0	0
9	0	0.1	1	0	0

Table 2: Example output from classifier 2. The differences from classifier 1 are in boldface.

		Truth		
		1	0	Σ
Estimate	1	TP	FP	$\hat{P} = TP + FP$
	0	FN	TN	$\hat{N} = FN + TN$
	Σ	$P = TP + FN$	$N = FP + TN$	

Table 3: Summary of definitions used for evaluating binary classification systems.

positive:

$$TP = \sum_{i=1}^{n} I(\hat{y}_i = 1 \wedge y_i = 1) \tag{1}$$

Similarly we can define TN as the number of true negatives, FP as the number of false positives and FN as the number of false negatives. Also, let $\hat{P} = TP + FP$ be the number of called positives, and $P = TP + FN$ be the true number of positives; similarly, define $\hat{N} = FN + TN$ as the number of called negatives, and $N = FP + TN$ be the true number of negatives. See Table 3 for a summary of these definitions; this is called a **confusion matrix**. We have that $TP + FP + FN + TN = n$, the total number of test points. Thus by normalizing this **contingency table** of counts, we can approximate the following conditional probabilities:

$$
\begin{align}
p(\hat{y} = 1, y = 1) &= TP/n \tag{2}\\
p(\hat{y} = 0, y = 0) &= TN/n \tag{3}\\
p(\hat{y} = 1, y = 0) &= FP/n \tag{4}\\
p(\hat{y} = 0, y = 1) &= FN/n \tag{5}
\end{align}
$$

The **true positive rate**, also called **sensitivity** or **recall** or **hit rate**, is defined as

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} = \frac{p(\hat{y} = 1, y = 1)}{p(y = 1)} = p(\hat{y} = 1|y = 1) \tag{6}$$

The **false positive rate**, also called the **false acceptance rate** or **type I error rate**, is defined as

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} = \frac{p(\hat{y} = 1, y = 0)}{p(y = 0)} = p(\hat{y} = 1|y = 0) \tag{7}$$

A **ROC** (receiver operating characteristic) curve is a plot of TPR vs FPR for different thresholds θ. See Figure 1 for an example. Any system can achieve the point on the bottom left, $(FPR = 0, TPR = 0)$, by setting $\theta = 1$ and thus classifying everything as negative; similarly any system can achieve the point on the top right, $(FPR = 1, TPR = 1)$, by setting $\theta = 0$ and thus classifying everything as positive. A system that sets $p(y_i|x_i) = 0.5$ can achieve any point on the diagonal line $TPR = FPR$ by choosing an appropriate threshold; thus this represents chance performance.
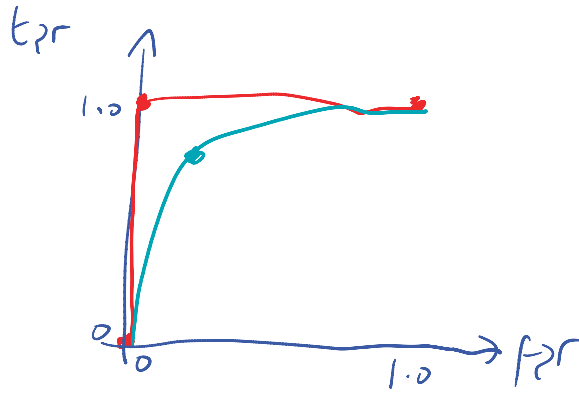
Figure 1: ROC curves for two classification systems. We plot the true positive rate (TPR) vs the false positive rate (FPR) as we vary the threshold θ. The red curve corresponds to the system in Table 1. The bottom left point corresponds to $\theta = 1$, which has a TPR of $0/5 = 0$ and an FPR of $0/0 = 0$; the top left point corresponds to $\theta = 0.5$, which has a TPR of $5/5 = 1$ and an FPR of $0/4 = 0$; the top right point corresponds to $\theta = 0$, which has a TPR of $5/5 = 1$ and an FPR of $4/4 = 1$. The green curve corresponds to the system in Table 2. Here the top left point corresponds to $\theta = 0.5$, which has a TPR of $4/5 = 0.8$ and an FPR of $1/4 = 0.25$. Clearly the red curve is better than the green curve.

A system that perfectly separates the positives from negatives has a threshold that can achieve the top left corner, $(FPR = 0, TPR = 1)$; by varying the threshold such a system will "hug" the left axis and then the top axis, as shown in Figure 1.

The quality of a ROC curve is often summarized using the **area under the curve** (AUC). Higher AUC scores are better; the maximum is obviously 1. Another summary statistic that is used is the **equal error rate**, also called the **cross over rate**, defined as the value which satisfies $FPR = FNR$; lower EER scores are better, the minimum is obviously 0. Here FNR is the **false negative rate**, also called the **false rejection rate**, or **type II error rate**, and is defined as

$$FNR = \frac{FN}{P} = \frac{FN}{FN + TP} = 1 - TPR = \frac{p(\hat{y} = 0, y = 1)}{p(y = 1)} = p(\hat{y} = 0 | y = 1) \tag{8}$$

The EER corresponds to the point where the line $FPR = 1 - TPR$ intersects the ROC curve: see Figure 2.

For completeness, we also define the quantity called **specificity** as

$$spec = \frac{TN}{N} = \frac{TN}{FP + TN} = 1 - FPR = \frac{p(\hat{y} = 0, y = 0)}{p(y = 0)} = p(\hat{y} = 0 | y = 0) \tag{9}$$

Also, the overall **accuracy** is defined as

$$acc = \frac{TP + TN}{TP + FP + FN + TN} = \frac{TP + TN}{n} = p(\hat{y} = 1, y = 1) + p(\hat{y} = 0, y = 0) \tag{10}$$

2 Precision recall curves

When the number of negatives is very large, such as when trying to detect a **rare event** (such as retrieving a relevant document or finding a face in an image), comparing $TP/P = p(\hat{y} = 1 | y = 1)$ to $FP/N = p(\hat{y} = 1 | y = 0)$ is not very meaningful. Instead, we can compare TP/P (the recall or TPR) to $TP/\hat{P}$, which is called the **precision** or **positive predictive value** (PPV):

$$prec = \frac{TP}{\hat{P}} = \frac{TP}{TP + FP} = p(y = 1 | \hat{y} = 1) \tag{11}$$

Precision measures what fraction of the entities that we called are actually positive, and recall measures what fraction of the true positives we called. A **precision recall** (PR) curve is a plot of precision vs recall as we vary the threshold θ.
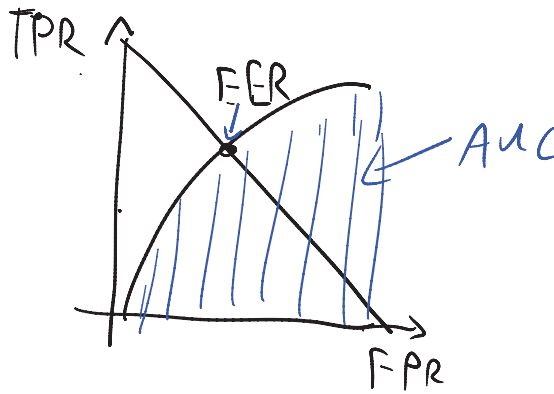
3

Figure 2: ROC curve, with the AUC and EER indicated.

See Figure 3. Hugging the top right is the best one can do. This curve can be summarized using the **mean precision** (averaging over recall values), which approximates the area under the curve. Alternatively, one can quote the precision for a fixed recall level (say, the first 10 entities in a retrieval system).

Precision and recall are often combined into a single statistic called the **F score**. This is the **harmonic mean** of precision and recall:

$$F = \frac{2}{1/P + 1/R} = \frac{2PR}{R + P} \tag{12}$$

In general, we can change the weightings on precision and recall:

$$F_\alpha = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} \tag{13}$$

with the usual measure corresponding to $\alpha = 1/2$. Note that $0 \leq F_\alpha \leq 1$. This is commonly used to rank **information retrieval systems**, that return a fixed set of documents. We use the harmonic mean instead of the arithmetic mean because an arithmetic mean can always trivially achieve a score of 0.5, by setting $R = 1$ (recalling all the entities). In contrast, if we assume that $p(y = 1) = 10^{-4}$, the harmonic mean of this strategy is 0.2%:

$$F = \frac{2PR}{P + R} = \frac{2 \times 10^{-4} \times 1}{1 + 10^{-4}} = 0.00019998 \tag{14}$$

F-scores can be used to evaluate the output of a classifier with a fixed probability threshold, or, in the context of information retrieval, a system that returns a fixed set of documents. If the system returns a probabilistic output, or a ranked set of documents, one should use a PR curve.

3 Using mutual information to compare classifiers

The following section, which is based on [Wal06], shows that using accuracy, precision, recall or F-scores to rank hard (i.e., non-probabilistic) binary classifiers can yield counterintuitive results, whereas computing the **mutual information** between the predicted label, $\hat{y}$, and the true label, y, yields sensible results.

Suppose the true distribution is that $p(y = 1) = 0.9$ and $p(y = 0) = 0.1$. Consider classifier A, which always classifies everything as positive (e.g., because it uses a threshold of $\theta = 0$): see Figure 4(left). Clearly A contains no useful information, yet its accuracy is 0.9, its precision is 0.9 and its recall is 1.0 (see Table 5).

Now consider classifier B, which classifies 80% of instances as positive and 20% as negative. It correctly classifies all negative instances, but also misclassifies some positive instances as negative. See Figure 4(middle). If classifier B claims $\hat{y} = 1$, then we know that in fact $y = 1$ (it makes no false positives), since

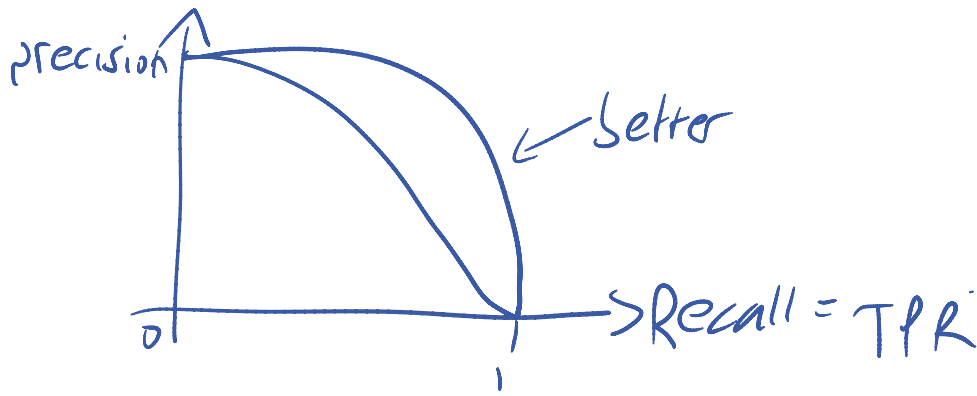$$p(y = 1 | \hat{y} = 1) = \frac{p(y = 1, \hat{y} = 1)}{p(y = 1)} = \frac{0.8}{0.8 + 0} = 1 \tag{15}$$

4

Figure 3: Precision-recall curve.

	A	.	B	.	C	.
	1	0	1	0	1	0
1	0.9	0.1	0.8	0	0.78	0
0	0	0	0.1	0.1	0.12	0.1

Table 4: Normalized confusion matrix for three different classifiers. Rows are the true labels, columns are the estimated labels for different models.

If classifier B claims that $\hat{y} = 0$, then we may be 50% sure that it really is negative, which is much higher than the overall class prior of $p(y = 0) = 0.1$.

Finally consider classifier C, which classifies 78% of instances as positive and 22% as negative. It correctly classifies all negative instances, but also misclassifies some positive instances as negative. See Figure 4(right). Intuitively this is not as good as classifier B since it puts more probability mass on the off-diagonal terms.

In Table 5, we evaluate all 3 classifiers according to various metrics. Intuitively, we would like the ranking $B \geq C \geq A$, but the only metric that gives this order is the mutual information, defined as

$$I(\hat{Y}, Y) = \sum_{\hat{y}=0}^{1} \sum_{y=0}^{1} p(\hat{y}, y) \log \frac{p(\hat{y}, y)}{p(\hat{y})p(y)} \tag{16}$$

This argues against using such measures as precision, recall and F-scores to compare binary classifiers.

References

[Faw03] Tom Fawcett. ROC Graphs: Notes and Practical Considerations for Data Mining Researchers. Technical report, HP labs, 2003.

[MRS08] C. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

Metric	A	B	C
Accuracy	0.9	0.9	0.88
Precision	0.9	1.0	1.0
Recall	1.0	0.888	0.8667
F-score	0.947	0.941	0.9286
Mutual information	0	0.1865	0.1735

Table 5: Various evaluation metrics for the three classifiers.

5

[Wal06] H. Wallach. Evaluation metrics for hard classifiers. Technical report, Cavendish Lab., Univ. Cambridge, 2006.