Nuts and Bolts of Data Mining: Classifiers & ROC Curves

By Tim Graettinger

Yes or no. Buy or sell. Renew or cancel. Many customer behaviors have this flavor of a choice between two alternatives. Suppose software called a "classifier" (explained more below) is available to predict customer choices in advance. Would you use it? Why or why not? Perhaps you'd like to test it to see how well it performs before you commit. In this installment of my ongoing series on the nuts and bolts of data mining, I discuss the use of classifiers and the question of performance. Regarding performance, we specifically consider hits, misses, false alarms, and the ROC[1] curve that pulls them all together.

Classifiers

Think about the customer behaviors that you would like to predict. A publisher might like to distinguish prospects who will subscribe to a new outdoor sports magazine from those who won't. An insurance executive may want to differentiate policyholders who will renew from those who will terminate. A gift officer at a non-profit might wish to separate large-potential donors from the more typical, $25 donors. In each of these instances, there is value in making the distinction before taking any action. For example, the fundraiser would love to know which donors are most likely to make a large gift before spending the time and money to contact them.

But how can you make these predictions? Let me suggest using a classifier. A classifier is a piece of data mining software that attempts to predict a future outcome (respond/not, renew/terminate, or large/small donor) using other attributes that are readily available in the present. Such attributes can include, say, for an existing customer: customer age, gender, number of past purchases, total amount of past purchases, date of first purchase, and the like.

The job of the data miner is to build the classifier. But, alas, any real-world classifier is imperfect. To paraphrase George Box[2], "all classifiers are wrong sometimes, but some classifiers are useful." How will you know if a classifier is useful? Read on.

Warranty Fraud & Abuse

For purposes of illustration, consider a car manufacturer and the warranty claims submitted by their network of dealers. Suppose a customer brings her car into the

dealer for service during the warranty period. The dealer performs the service, and the customer pays nothing. The dealer does, however, bill the manufacturer for the work, and the manufacturer reimburses the dealer. Unfortunately, not all claims from the dealer are legitimate. Abuse, and in rare instances even fraud[3], does occur.

Suppose the car maker sets a business goal to reduce abuse in the payment of warranty claims. A project team is assembled to build a classifier to detect abuse – based upon a set of painstakingly-collected, human-audited, warranty claims. Suppose further that 10,000 of these audited claims, excerpted in Table 1, are held out for testing the classifier. The table consists of 10,000 rows (one per claim) and a collection of columns. The columns are attributes known at the time the claim is submitted, including: a claim identifier, the number of services performed, type of service, etc. Attached to each row is an audited outcome, which is zero if approved, and one if disallowed (that is, determined to be abusive or fraudulent by the human auditors). The final column of the table is the predicted outcome produced by the classifier, which is the subject of the test.

ClaimID	# Services	Type of Service	…	Audited Outcome	Predicted Outcome
910910	3	M	…	1	0.88
901917	1	G	…	0	0.31
…	…	…	…	…	…
1279817	1	G	…	0	0.05

Table 1- Excerpt of the audited claims used for testing

The Classification Matrix

Keep in mind that the goal of testing the classifier is to evaluate its performance before putting it into production use. A "common sense" evaluation is to simply count the right and wrong predictions, and then calculate the percentage correct. Sounds sensible, right? Let's find out.

Notice, first, that the predicted outcome in Table 1 is not simply zero or one, approved or disallowed, as is the audited outcome. Rather, the prediction is a continuous number between zero and one. This seems to complicate our common sense calculation. Never fear, though, there is an easy way out of this difficulty. We can draw a line in the sand, or less metaphorically, we can choose a threshold value, say 0.8. (Note that the threshold value **does not have to be 0.5**). Predictions greater than 0.8 are put in the disallowed bucket, and predictions less than 0.8 are put in the allowed bucket.

Having chosen a threshold, we can "bucket" the predictions and fill in the 2 x 2 classification matrix, shown in Table 2. In this matrix, the rows are the predicted outcomes from the classifier, and the columns are the audited outcomes. There are 8400 claims where the classifier predicts zero, or "approved", and the audited outcome is also approved. Similarly, there

		Audited		
		0 Approved	1 Disallowed	Total
Predicted	0 Approved	8400	100	8500
	1 Disallowed	600	900	1500
	Total	9000	1000	10000

Table 2 - Classification matrix

are 900 claims where the classifier predicted "disallowed", and the audited outcome is also disallowed. Together, these cells in the matrix represent all of the correct predictions. There are 9,300 correct out of 10,000 total claims, or 93%.

Is This Good?

Is 93% good? It sounds good, but compared to what? And why did we choose 0.8 as the threshold? What if we choose a different threshold? As we'll soon see, these questions are all important – and interrelated.

First, let's address the "Is 93% good?" question. Any response is ridiculous without some baseline for comparison. We can quickly establish a baseline from the column sums in Table 2. Notice that 9,000 of the 10,000 claims are approved by the auditors. That means that a simple classifier that always says "approved" will be correct 90% of the time. This baseline number puts the 93% correct result for the tested classifier into context. Keep in mind, however, that the "always approve" classifier never detects any claims that are likely to be abusive or fraudulent. In other words, you never save any money if you always approve.

The 93% correct metric for the tested classifier, versus 90% for the simple "always approve" classifier, provides some information about performance. For most purposes, though, this is insufficient. To glean more insight, we need to look at the classification matrix in more detail.

Let's begin by studying the column where the audited outcome is one, or "disallowed". Notice first that there are 1000 claims with this outcome. Of these, the classifier predicts "disallowed" for 900 (these are the "hits") and "approved" for 100 (these are the "misses"). To use a bit of data mining jargon, we say that the hit rate for the classifier is 900/1000, or 90 percent. In other words, the classifier detects 90% of the actual instances of abuse or fraud. This is where the car maker can save money.

Is your Spidey-sense tingling? Good for you if it is. It's probably telling you that there's no free lunch. In fact, there is a trade-off to hit rate that you must consider as well. Look again at Table 2 and study the row where the predicted outcome is one, or "disallowed". Notice that there are 1500 claims with this predicted outcome, and 900 of them have an audited outcome of "disallowed". Also critically important, there are 600 predictions out of 1500 (40%) where the audited outcome is "approved". In data mining terms, these are the "false alarms". These are claims the classifier disallows that should rightly be approved. These are the mistakes that will

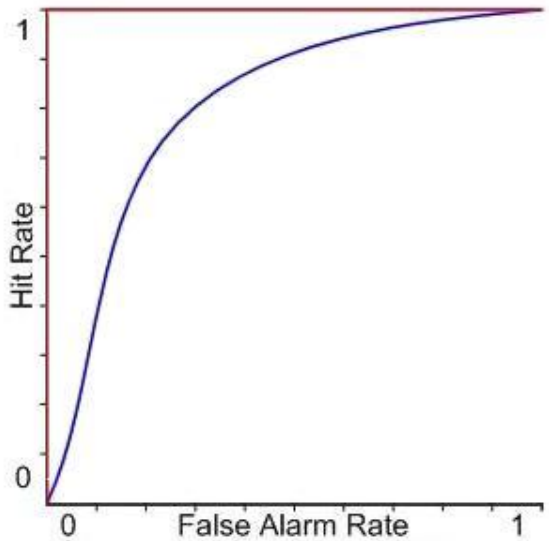

Figure 1- ROC curve for the classifier

really strain the relationship between the dealers and the manufacturer.

The trade-off between hit rate and false alarm rate depends on both the classifier and the choice of threshold. Since we are testing a particular classifier, we make no changes there. That leaves the threshold as the sole adjustment knob. Stop reading for a moment and imagine what happens to the hit rate and the false alarm rate as you raise the threshold. What happens when you lower it?

Here's the answer: when you lower the threshold, more claims fall into the disallowed bucket, which generally will include more hits. As a result, the hit rate goes up -- but so does the false alarm rate. Conversely, when you raise the threshold, you get higher-quality hits, but fewer of them. On the plus side, you produce fewer false alarms.

The ROC Curve

But all of this is fairly vague and wishy-washy. Wouldn't you like to try a few different thresholds to see how the hit rates and false alarm rates change? Why not try ALL threshold values?! That, in fact, is the purpose of the ROC curve shown in Figure 1. The horizontal axis is the false alarm rate, and the vertical axis is the hit rate. One choice of threshold produces one point on the curve, a combination of false alarm rate and hit rate. Another choice produces another point. By varying the threshold across a range of values, say from 0 to 1, you generate the entire curve (in blue) for the classifier. Small values for the threshold produce points in the northeast corner of the graph, corresponding to high hit rates and high false alarm rates. High threshold values, conversely, generate points in the southwest corner, where the hit rates and false alarm rates are both low.

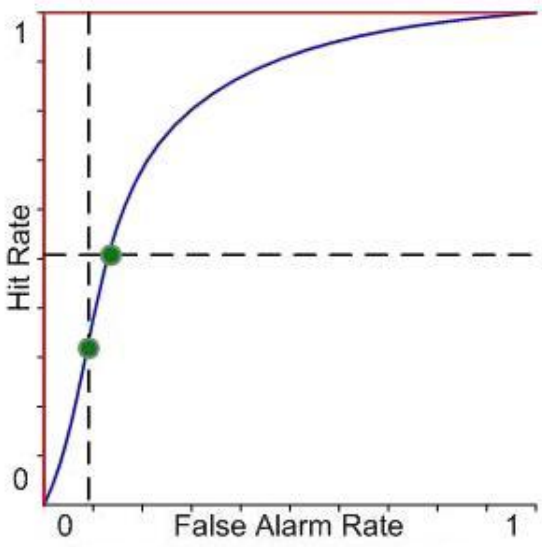

Figure 2 - ROC Curve with a hit rate requirement (dotted horizontal line) or a false alarm rate limit (dotted vertical line)

With the ROC curve as a tool, you can proceed to judge a classifier in terms of business performance. Consider again the car maker. Suppose management sets a goal of reducing abuse and fraud by 50%. That implies a hit rate of 50% (detecting and rejecting 50% of the bad claims). The hit rate requirement appears as a horizontal line on the ROC curve in Figure 2. For this classifier, a 50% hit rate will generate a 15% false alarm rate.

On the other hand suppose management decides that false alarms must be minimal, less than 10%, to maintain good relations with their dealers. The selected false alarm

rate appears as a vertical line in Figure 2. Notice that this line intersects the ROC curve at a hit rate value of about 30%.

Now, the key point: you can specify either a false alarm rate or a hit rate. The other rate is determined by the classifier (and is shown graphically via the ROC curve). You cannot specify both. The car maker decides that the dealer relationship is most important, so they set the false alarm rate low and must settle for a slightly lower hit rate than originally desired. Over time, as more and richer data is collected, improved classifiers can be built to improve the hit rate for the chosen false alarm rate.

Wrap Up

In this "Nuts & Bolts" article, we focused on classifiers and classifier performance. Classifiers have a wide range of uses in business, from reducing customer turnover to detecting abuse and fraud. To realize the full benefit of classifiers, you must evaluate and test them. In this article, we saw that a simple metric --percentage correct-- makes sense only relative to an established benchmark. But, as an evaluation method, it is fairly weak. We progressed to the classification matrix as an enhanced performance tool, and introduced the concepts of hits, misses, and false alarms. Finally, we studied the ROC curve that allows you to make real business decisions that trade off hits and false alarms – decisions that can make or save money while maintaining good customer or partner relations.

Tim Graettinger, Ph.D., is the President of Discovery Corps, Inc. (http://www.discoverycorpsinc.com), a Pittsburgh-area company specializing in data mining, visualization, and predictive analytics.

Your comments and questions about this article are welcome. Please contact Tim at (724)-743-3642 or tgraettinger@discoverycorpsinc.com

---

[1] ROC is pronounced "rock", and it is an acronym for Receiver Operating Characteristic. Don't ask. The name doesn't really illuminate anything, but it does sound cool – at least to data miners.

[2] George Box is a famous statistician who once said, "All models are wrong, but some are useful."

[3] The difference between fraud and abuse is generally intent – did someone know they were committing a crime. And that's a difference for lawyers to sort out, not data miners.