Data Science

Week 13

What is a Good Model?

Introduction

- For data science to add value to an application, it is important for the data scientists and other stakeholders to consider carefully what they would like to achieve by mining data.
- Both data scientists themselves and the people who work with them often avoid—perhaps without even realizing it—connecting the results of mining data back to the goal of the undertaking.
- The goal depends on the application. We cannot offer a single evaluation metric that is "right" for any classification problem, or regression problem, or whatever problem you may encounter.

Outline

- Evaluating Classifiers
- A Key Analytical Framework: Expected Value
- Visualizing Model Performance

Evaluating Classifiers

- Let's consider binary classification, for which the classes often are simply called "positive" and "negative." How shall we evaluate how well such a model performs?
  - Plain Accuracy and Its Problems
  - The Confusion Matrix
  - Problems with Unbalanced Classes

Plain Accuracy

- Classification accuracy is a popular metric because it's very easy to measure. Unfortunately, it is usually too simplistic for applications of data mining techniques to real problems.
- We will discuss it and some of the alternatives.
- The term "classifier accuracy" is sometimes used informally to mean any general measure of classifier performance.
- Here we will reserve accuracy for its specific technical meaning as the proportion of correct decisions:

$$\text{accuracy} = \frac{\text{Number of correct decisions made}}{\text{Total number of decisions made}}$$

Advantage and Limitation of Accuracy

- Accuracy is a common evaluation metric that is often used in data mining studies because it reduces classifier performance to a single number and it is very easy to measure.

- To understand these problems we need a way to decompose and count the different types of correct and incorrect decisions made by a classifier. For this we use the confusion matrix.

Confusion Matrix

- To evaluate a classifier properly it is important to understand the notion of class confusion and the confusion matrix.
- A confusion matrix for a problem involving n classes is an $n \times n$ matrix with the columns labeled with actual classes and the rows labeled with predicted classes. Each example in a test set has an actual class label as well as the class predicted by the classifier (the predicted class), whose combination determines which matrix cell the instance counts into.
- For simplicity we will deal with two-class problems having 2×2 confusion matrices.

Confusion Matrix

- We will consider two-class problems, and will denote the true classes as p(ositive) and n(egative), and the classes predicted by the model (the "predicted" classes) as Y(es) and N(o), respectively (think: the model says "Yes, it is a positive" or "No, it is not a positive").

- In the confusion matrix, the main diagonal contains the counts of correct decisions. The errors of the classifier are the false positives (negative instances classified as positive) and false negatives (positives classified as negative).

	p	n
Y	True positives	False positives
N	False negatives	True negatives

Problems with Unbalanced Classes

- As an example of how we need to think carefully about model evaluation, consider a classification problem where one class is rare.
- This is a common situation in applications, because classifiers often are used to sift through a large population of normal or uninteresting entities in order to find a relatively small number of unusual ones;
- For example, looking for defrauded customers, checking an assembly line for defective parts.
- Because the unusual or interesting class is rare among the general population, the class distribution is unbalanced.

Accuracy for unbalanced data

- Unfortunately, as the class distribution becomes more skewed, evaluation based on accuracy breaks down.
- Consider a domain where the classes appear in a 999:1 ratio. A simple rule—always choose the most prevalent class—gives 99.9% accuracy.
- With such skewed domains the base rate for the majority class could be very high, so a report of 99.9% accuracy may tell us little about what data mining has really accomplished.

Confusion matrices for unbalanced data

- Even when the skew is not so great, in domains where one class is more prevalent than another accuracy can be greatly misleading. Consider again our cellular-churn example.
- Let's say you are a manager at MegaTelCo and as an analyst I report that our churn prediction model generates 80% accuracy. This sounds good, but is it? My coworker reports that her model generates an accuracy of 37%. That's pretty bad, isn't it?
- we need more information about the data.

Classifications on a balanced population

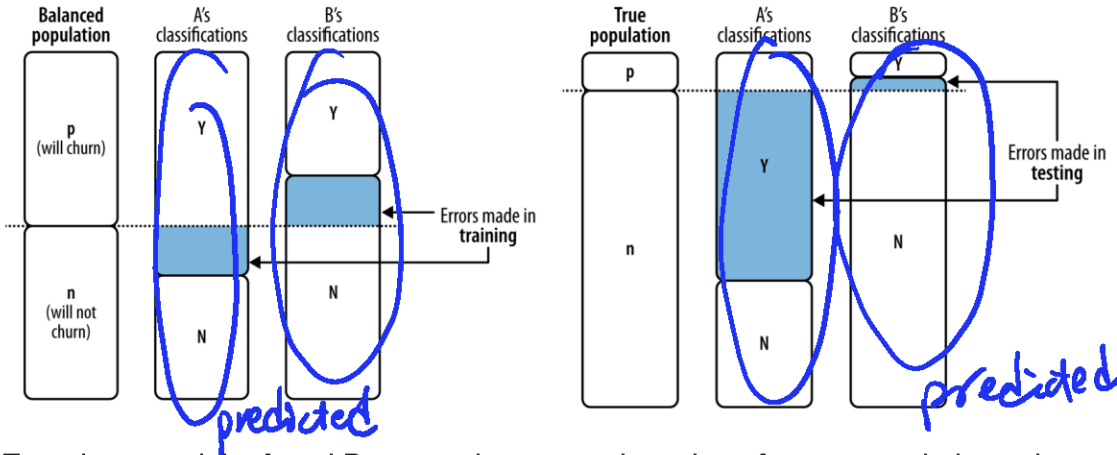
- Two tables illustrate these classifications on a balanced population and on a representative population. As mentioned, both models correctly classify 80% of the balanced population, but the confusion matrices and the figure show that they operate very differently.

	churn	not churn
Y	500	200
N	0	300

error

	churn	not churn
Y	300	0
N	200	500

Errors in training and test



Two churn models, A and B, can make an equal number of errors on a balanced population used for training. But a very different number of errors when tested against the true population.

Problems with Unequal Costs and Benefits

- Another problem with simple classification accuracy as a metric is that it makes no distinction between false positive and false negative errors.
- By counting them together, it makes an assumption that both errors are equally important. With real-world domains this is rarely the case. These are typically very different kinds of errors with very different costs because the classifications have consequences of differing severity.
- Returning to our cellular-churn example, consider the cost of giving a customer a retention incentive which still results in departure (a false positive error). Compare this with the cost of losing a customer because no incentive was offered (a false negative). Whatever costs you might decide for each, it is unlikely they would be equal; and the errors should be counted separately regardless.

Outline

- Evaluating Classifiers
- A Key Analytical Framework: Expected Value
- Visualizing Model Performance

Expected value

- We now are ready to discuss a very broadly useful conceptual tool to aid data analytic thinking: expected value.
- The expected value computation provides a framework that is extremely useful in organizing thinking about data-analytic problems. Specifically, it decomposes data-analytic thinking into

(i) the structure of the problem,

(ii) the elements of the analysis that can be extracted from the data, and

(iii) the elements of the analysis that need to be acquired from other sources

Expected value

- In an expected value calculation the possible outcomes of a situation are enumerated. The expected value is then the weighted average of the values of the different possible outcomes, where the weight given to each value is its probability of occurrence.
- For example, if the outcomes represent different possible levels of profit, an expected profit calculation weights heavily the highly likely levels of profit, while unlikely levels of profit are given little weight.
- For this lecture, we will assume that we are considering repeated tasks (like targeting a large number of consumers, or diagnosing a large number of problems) and we are interested in maximizing expected profit.

Calculation of expected value

- The expected value framework provides structure to an analyst's thinking (i) via the general form:

$$EV = p(o_1) \cdot v(o_1) + p(o_2) \cdot v(o_2) + p(o_3) \cdot v(o_3) \ldots$$

- Each o_i is a possible decision outcome; $p(o_i)$ is its probability and $v(o_i)$ is its value. The probabilities often can be estimated from the data (ii), but the business values often need to be acquired from other sources (iii).

Using Expected Value to Frame Classifier Use

- Consider that we have an offer for a product that, for simplicity, is only available via this offer. If the offer is not made to a consumer, the consumer will not buy the product. We have a model, mined from historical data, that gives an estimated probability of response $p_R(x)$ for any consumer whose feature vector description x is given as input.
- The model could be a classification tree or a logistic regression model or some other model we haven't talked about yet. Now we would like to decide whether to target a particular consumer described by feature vector x.

Using Expected Value to Frame Classifier Use

- Let's calculate the expected benefit (or cost) of targeting consumer **x**:

$$\text{Expected benefit of targeting} = p_R(\mathbf{x}) \cdot v_R + [1 - p_R(\mathbf{x})] \cdot v_{NR}$$

- where v_R is the value we get from a response and v_{NR} is the value we get from no response. Since everyone either responds or does not, our estimate of the probability of not responding is just $(1 - p_R(x))$.
- As mentioned, the probabilities came from the historical data, as summarized in our predictive model. The benefits v_R and v_{NR} need to be determined separately.
- Since a customer can only purchase the product by responding to the offer (as discussed above), the expected benefit of not targeting her conveniently is zero.

Using Expected Value to Frame Classifier Use

- To be concrete, let's say that a consumer buys the product for $200 and our product-related costs are $100. To target the consumer with the offer, we also incur a cost. Let's say that we mail some flashy marketing materials, and the overall cost including postage is $1, yielding a value (profit) of v_R = $99 if the consumer responds (buys the product).
- Now, what about v_{NR}, the value to us if the consumer does not respond? We still mailed the marketing materials, incurring a cost of $1 or equivalently a benefit of -$1. Now we are ready to say precisely whether we want to target this consumer: do we expect to make a profit? Technically, is the expected value (profit) of targeting greater than zero? Mathematically, this is:

$$p_R(x) \cdot \$99 - [1 - p_R(x)] \cdot \$1 > 0$$

$$\rightarrow \quad p_R(x) > 0.01$$

- With these example values, we should target the consumer as long as the estimated probability of responding is greater than 1%.

Using Expected Value to Frame Classifier Evaluation

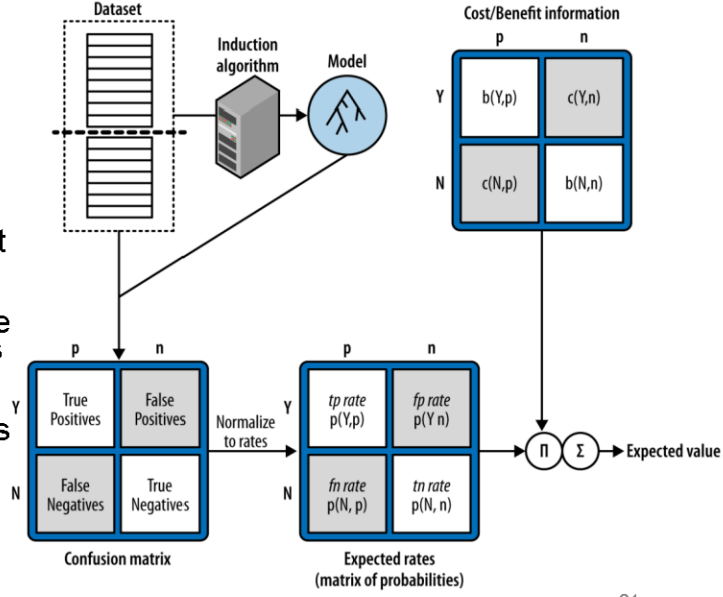
- At this point we want to shift our focus from individual decisions to collections of decisions. Specifically, we need to evaluate the set of decisions made by a model when applied to a set of examples. Such an evaluation is necessary in order to compare one model to another.
- For example, does our data-driven model perform better than the hand-crafted model suggested by the marketing group? Does a classification tree work better than a linear discriminant model for a particular problem? Do any of the models do substantially better than a baseline "model," such as randomly choosing consumers to target?
- It is likely that each model will make some decisions better than the other model. What we care about is, *in aggregate*, how well does each model do: what is its *expected* value.

Using Expected Value to Frame Classifier Evaluation

- We can use the expected value framework just described to determine the best decisions for each particular model, and then use the expected value in a different way to compare the models. If we are to calculate the expected profit for a model in aggregate, each o_i in Equation on page 18 corresponds to one of the possible combinations of the class we predict, and the actual class.
- Each o_i corresponds to one cell of the confusion matrix. For example, what is the probability associated with the particular combination of a consumer being *predicted to churn* and *actually does not churn*? That would be estimated by the number of test-set consumers who fell into the confusion matrix cell (Y,n), divided by the total number of test-set consumers.

Diagram of the expected value calculation

Figure shows a schematic diagram of the expected value calculation in the context of model induction and evaluation. At the top left of the diagram, a training portion of a dataset is taken as input by an induction algorithm, which produces the model that we will evaluate. That model is applied to a holdout (test) portion of the data, and the counts for the different cells of the confusion matrix are tallied.



From confusion matrix to Estimated probabilities

- Let's consider a concrete example of a classifier confusion matrix in Table.

A sample confusion matrix with counts.

Each cell of the confusion matrix contains a count of the number of decisions corresponding to the corresponding combination of (predicted, actual).
For the expected value calculation we reduce these counts to rates or estimated probabilities. We do this by dividing each count by the total number of instances:

	p	n
Y	56	7
N	5	42

$T = 110$

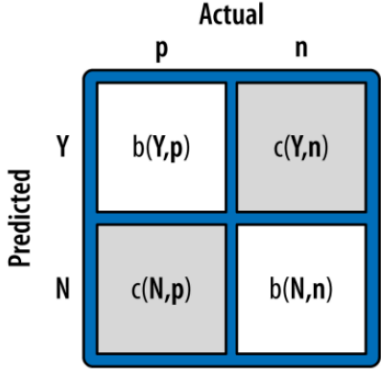
$p(\mathbf{Y},\mathbf{p}) = 56/110 = 0.51 \quad p(\mathbf{Y},\mathbf{n}) = 7/110 = 0.06$

$p(\mathbf{N},\mathbf{p}) = 5/110 = 0.05 \quad p(\mathbf{N},\mathbf{n}) = 42/110 = 0.38$

Costs and benefits

- To compute expected profit, we also need the cost and benefit values that go with each decision pair. These will form the entries of a cost-benefit matrix with the same dimensions (rows and columns) as the confusion matrix. However, the cost-benefit matrix specifies, for each (predicted,actual) pair, the cost or benefit of making such a decision.

- Correct classifications (true positives and true negatives) correspond to the benefits $b(\mathbf{Y},\mathbf{p})$ and $b(\mathbf{N},\mathbf{n})$, respectively. Incorrect classifications (false positives and false negatives) correspond to the "benefit" $b(\mathbf{Y},\mathbf{n})$ and $b(\mathbf{N},\mathbf{p})$, respectively, which may well actually be a cost (a negative benefit), and often are explicitly referred to as costs $c(\mathbf{Y},\mathbf{n})$ and $c(\mathbf{N},\mathbf{p})$.



Costs and benefits

- While the probabilities can be estimated from data, the costs and benefits often cannot. They generally depend on external information provided via analysis of the consequences of decisions in the context of the specific business problem.
- Indeed, specifying the costs and benefits may take a great deal of time and thought. In many cases they cannot be specified exactly but only as approximate ranges.
- For example, in our churn problem, how much is it really worth us to retain a customer? The value depends on future cell phone usage and probably varies a great deal between customers. It may be that data on customers' prior usage can be helpful in this estimation. In many cases, average estimated costs and benefits are used rather than individual-specific costs and benefits, for simplicity of problem formulation and calculation.

Understanding benefits in targeted marketing example

- So, let's return to our targeted marketing example. What are the costs and benefits? We will express all values as benefits, with costs being negative benefits, so the function we're specifying is b(predicted, actual). For simplicity, all numbers will be expressed as dollars.
- These cost-benefit estimations can be summarized in a 2 × 2 cost-benefit matrix, as in following Figure. Note that the rows and columns are the same as for our confusion matrix, which is exactly what we'll need to compute the overall expected value for the classification model.

		Actual	
		p	**n**
Predicted	**Y**	99	-1
	N	0	0

Benefits in the targeted marketing example

- A *false positive* occurs when we classify a consumer as a likely responder and therefore target her, but she does not respond. We've said that the cost of preparing and mailing the marketing materials is a fixed cost of $1 per consumer. The benefit in this case is negative: $b(\mathbf{Y}, \mathbf{n}) = -1$.
- A *false negative* is a consumer who was predicted not to be a likely responder (so was not offered the product), but would have bought it if offered. In this case, no money was spent and nothing was gained, so $b(\mathbf{N}, \mathbf{p}) = 0$.
- A *true positive* is a consumer who is offered the product and buys it. The benefit in this case is the profit from the revenue ($200) minus the product-related costs ($100) and the mailing costs ($1), so $b(\mathbf{Y}, \mathbf{p}) = 99$.
- A *true negative* is a consumer who was not offered a deal and who would not have bought it even if it had been offered. The benefit in this case is zero (no profit but no cost), so $b(\mathbf{N}, \mathbf{n}) = 0$.

Expected profit in the targeted marketing example

- Given a matrix of costs and benefits, these are multiplied cell-wise against the matrix of probabilities, then summed into a final value representing the total expected profit. The result is:

$$\begin{aligned} \textit{Expected profit} = {} & p(\mathbf{Y}, \mathbf{p}) \cdot b(\mathbf{Y}, \mathbf{p}) + p(\mathbf{N}, \mathbf{p}) \cdot b(\mathbf{N}, \mathbf{p}) + \\ & p(\mathbf{N}, \mathbf{n}) \cdot b(\mathbf{N}, \mathbf{n}) + p(\mathbf{Y}, \mathbf{n}) \cdot b(\mathbf{Y}, \mathbf{n}) \end{aligned}$$

- Using this equation, we can now compute and compare the expected profits for various models and other targeting strategies. All we need is to be able to compute the confusion matrices over a set of test instances, and to generate the cost-benefit matrix.

Alternative formulation

- This equation is sufficient for comparing classifiers, but let's continue along this path a little further, because an alternative calculation of this equation is often used in practice.
- Furthermore, by examining the alternative formulation we can see exactly how to deal with the model comparison problem we introduced at the beginning of the lecture—where one analyst had reported performance statistics over a representative (but unbalanced) population, and another had used a class balanced population.
- A common way of expressing expected profit is to factor out the probabilities of seeing each class, often referred to as the class priors. The class priors, p(p) and p(n), specify the likelihood of seeing positive and negative instances, respectively.

Expected profit in an Alternative formulation

- A rule of basic probability is:

$$p(x,\ y) = p(y) \cdot p(x \mid y)$$

- This says that the probability of two different events both occurring is equal to the probability of one of them occurring times the probability of the other occurring if we know that the first occurs. Using this rule we can re-express our expected profit as:

$$\begin{aligned}\text{Expected profit} = {} & p(\mathbf{Y} \mid \mathbf{p}) \cdot p(\mathbf{p}) \cdot b(\mathbf{Y}, \mathbf{p}) + p(\mathbf{N} \mid \mathbf{p}) \cdot p(\mathbf{p}) \cdot b(\mathbf{N}, \mathbf{p}) + \\ & p(\mathbf{N} \mid \mathbf{n}) \cdot p(\mathbf{n}) \cdot b(\mathbf{N}, \mathbf{n}) + p(\mathbf{Y} \mid \mathbf{n}) \cdot p(\mathbf{n}) \cdot b(\mathbf{Y}, \mathbf{n})\end{aligned}$$

$$\begin{aligned}\text{Expected profit} = {} & p(\mathbf{p}) \cdot [p(\mathbf{Y} \mid \mathbf{p}) \cdot b(\mathbf{Y}, \mathbf{p}) + p(\mathbf{N} \mid \mathbf{p}) \cdot c(\mathbf{N}, \mathbf{p})] + \\ & p(\mathbf{n}) \cdot [p(\mathbf{N} \mid \mathbf{n}) \cdot b(\mathbf{N}, \mathbf{n}) + p(\mathbf{Y} \mid \mathbf{n}) \cdot c(\mathbf{Y}, \mathbf{n})]\end{aligned}$$

Expected profit in an Alternative formulation

- We now have one component (the first one) corresponding to the expected profit from the positive examples, and another (the second one) corresponding to the expected profit from the negative examples.
- Each of these is weighted by the probability that we see that sort of example. So, if positive examples are very rare, their contribution to the overall expected profit will be correspondingly small.
- In this alternative formulation, the quantities p(Y|p), p(Y|n), etc. correspond directly to the true positive rate, the false positive rate, etc., that also can be calculated directly from the confusion matrix

Calculation of Expected profit using example data

- Here again is our sample confusion matrix in Table.

	p	n
Y	56	7
N	5	42

- The next table shows the class priors and various error rates we need.

$T = 110$	
$P = 61$	$N = 49$
$p(\mathbf{p}) = 0.55$	$p(\mathbf{n}) = 0.45$
tp rate $= 56/61 = 0.92$	*fp rate* $= 7/49 = 0.14$
fn rate $= 5/61 = 0.08$	*tn rate* $= 42/49 = 0.86$

Calculation of Expected profit using example data

- Returning to the targeted marketing example, what is the expected profit of the model learned?

$$
\begin{aligned}
\text{expected profit} &= p(\mathbf{p}) \cdot [p(\mathbf{Y} \mid \mathbf{p}) \cdot b(\mathbf{Y}, \mathbf{p}) + p(\mathbf{N} \mid \mathbf{p}) \cdot c(\mathbf{N}, \mathbf{p})] + \\
&\quad p(\mathbf{n}) \cdot [p(\mathbf{N} \mid \mathbf{n}) \cdot b(\mathbf{N}, \mathbf{n}) + p(\mathbf{Y} \mid \mathbf{p}) \cdot c(\mathbf{Y}, \mathbf{n})] \\
&= 0.55 \cdot [0.92 \cdot b(\mathbf{Y}, \mathbf{p}) + 0.08 \cdot b(\mathbf{N}, \mathbf{p})] + \\
&\quad 0.45 \cdot [0.86 \cdot b(\mathbf{N}, \mathbf{n}) + 0.14 \cdot p(\mathbf{Y}, \mathbf{n})] \\
&= 0.55 \cdot [0.92 \cdot 99 + 0.08 \cdot 0] + \\
&\quad 0.45 \cdot [0.86 \cdot 0 + 0.14 \cdot -1] \\
&= 50.1 - 0.063 \\
&\approx \$\mathbf{50.04}
\end{aligned}
$$

This expected value means that if we apply this model to a population of prospective customers and mail offers to those it classifies as positive, we can expect to make an average of about $50 profit per consumer.

Conclusion for Expected profit

- We now can see one way to deal with our motivating example from the beginning of the lecture.
- Instead of computing accuracies for the competing models, we would compute expected values. Furthermore, using this alternative formulation, we can compare the two models even though one analyst tested using a representative distribution and the other tested using a class-balanced distribution.
- In each calculation, we simply can replace the priors. Using a balanced distribution corresponds to priors of $p(\mathbf{p}) = 0.5$ and $p(\mathbf{n}) = 0.5$.

Outline

- Evaluating Classifiers
- A Key Analytical Framework: Expected Value
- Visualizing Model Performance

Ranking Instead of Classifying

- "A Key Analytical Framework: Expected Value" discusses how the score assigned by a model can be used to compute a decision for each individual case based on its expected value.
- A different strategy for making decisions is to rank a set of cases by these scores, and then take actions on the cases at the top of the ranked list.
- Instead of deciding each case separately, we may decide to take the top *n* cases (or, equivalently, all cases that score above a given threshold). There are several practical reasons for doing this.

Reasons for Ranking: probability is not accurate

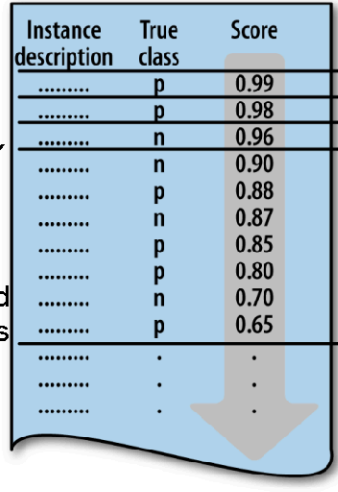
- It may be that the model gives a score that ranks cases by their likelihood of belonging to the class of interest, but which is not a true probability.
- More importantly, for some reason we may not be able to obtain accurate probability estimates from the classifier.
- The classifier scores may still be very useful for deciding which prospects are better than others, even if a 1% probability estimate doesn't exactly correspond to a 1% probability of responding.

Reasons for Ranking: Different requirements in different applications

- When working with a classifier that gives scores to instances, in some situations the classifier decisions should be very conservative, corresponding to the fact that the classifier should have high certainty before taking the positive action.
- This corresponds to using a high threshold on the output score. Conversely, in some situations the classifier can be more permissive, which corresponds to lowering the threshold.
- This introduces a complication for which we need to extend our analytical framework for assessing and comparing models. “The Confusion Matrix” stated that a classifier produces a confusion matrix.
- With a ranking classifier, a classifier plus a threshold produces a single confusion matrix. Whenever the threshold changes, the confusion matrix may change as well because the numbers of true positives and false positives change.

Different “Confusion Matrix” for different thresholds based on ranking

Figure illustrates this basic idea. As the threshold is lowered, instances move up from the N row into the Y row of the confusion matrix: an instance that was considered a negative is now classified as positive, so the counts change. Which counts change depends on the example’s true class.



How to choose the threshold?

- Technically, each different threshold produces a different classifier, represented by its own confusion matrix.
- This leaves us with two questions:
  - How do we compare different rankings?
  - And, how do we choose a proper threshold?
- If we have accurate probability estimates and a well specified cost-benefit matrix, then we already answered the second question in our discussion of expected value: we determine the threshold where our expected profit is above a desired level (usually zero).

Profit Curves

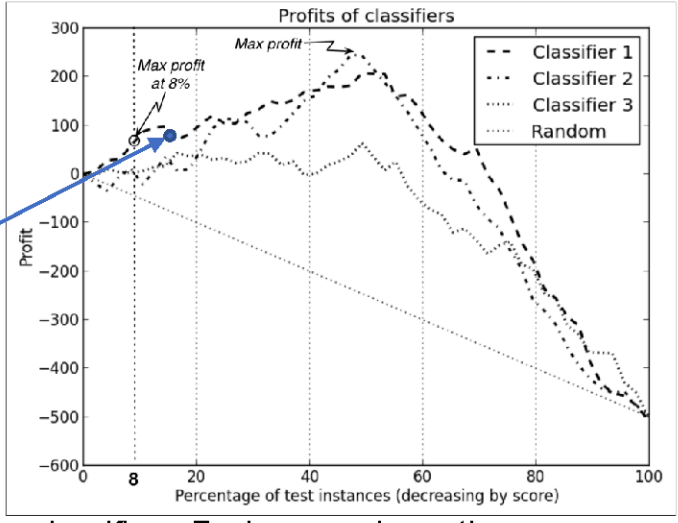
- We have known how to compute expected profit, and we've just introduced the idea of using a model to rank instances. We can combine these ideas to construct various performance visualizations in the form of curves.
- Each curve is based on the idea of examining the effect of thresholding the value of a classifier at successive points, implicitly dividing the list of instances into many successive sets of predicted positive and negative instances. As we move the threshold "down" the ranking, we get additional instances predicted as being positive rather than negative.
- Each threshold, i.e., each set of predicted positives and negatives, will have a corresponding confusion matrix. The previous lecture showed that once we have a confusion matrix, along with knowledge of the cost and benefits of decisions, we can generate an expected value corresponding to that confusion matrix.

Profit Curves

- More specifically, with a ranking classifier, we can produce a list of instances and their predicted scores, ranked by decreasing score, and then measure the expected profit that would result from choosing each successive cut-point in the list.
- Conceptually, this amounts to ranking the list of instances by score from highest to lowest and sweeping down through it, recording the expected profit after each instance. At each cut-point we record the percentage of the list predicted as positive and the corresponding estimated profit. Graphing these values gives us a profit curve.
- Three profit curves (generated by different classification methods, e.g., SVM, Logistic regression, …) are shown on the next page.

Example of Profit Curves

One point on the curve is corresponding to one confusion matrix



Profit curves of three classifiers. Each curve shows the expected cumulative profit for that classifier as progressively larger proportions of the consumer base are targeted.

Explanation for Figures

- This graph is based on a test set of 1,000 consumers.
- For each curve, the consumers are ordered from highest to lowest probability of accepting an offer based on some model.

- For this example, let's assume our profit margin is small: each offer costs $5 to make and market, and each accepted offer earns $9, for a profit of $4. The cost matrix is thus:

	p	n
Y	$4	-$5
N	$0	$0

Explanation for Figures

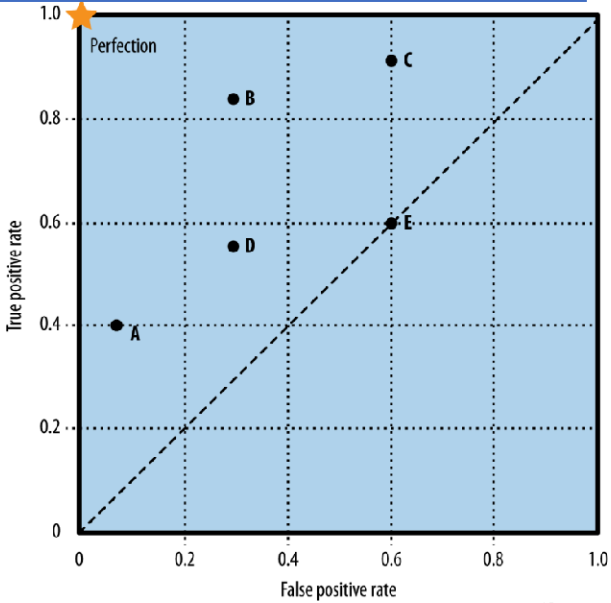
- Notice that all four curves begin and end at the same point. At the left side, when no customers are targeted there are no expenses and zero profit; at the right side everyone is targeted, so every classifier performs the same.
- In between, we'll see some differences depending on how the classifiers order the customers.
- The random classifier performs worst because it has an even chance of choosing a responder or a nonresponder.
- Among the classifiers tested here, the one labeled Classifier 2 produces the maximum profit of $200 by targeting the top-ranked 50% of consumers.
- If your goal was simply to maximize profit and you had unlimited resources, you should choose Classifier 2, use it to score your population of customers, and target the top half (highest 50%) of customers on the list.

ROC graph

- If both class priors and cost-benefit estimates are known and are expected to be stable, profit curves may be a good choice for visualizing model performance. However, in many domains these conditions are uncertain or unstable.
- Another approach is to use a method that can accommodate uncertainty by showing the entire space of performance possibilities. One such method is the Receiver Operating Characteristics (ROC) graph.
- A ROC graph is a two-dimensional plot of a classifier with false positive rate on the x axis against true positive rate on the y axis. As such, a ROC graph depicts relative trade-offs that a classifier makes between benefits (true positives) and costs (false positives).

ROC space

ROC space and five different classifiers (A-E) with their performance shown.



True positive rate =

$$\frac{\text{True positive}}{(\text{True positive} + \text{False Negative})}$$

False positive rate =

$$\frac{\text{False positive}}{(\text{False positive} + \text{True Negative})}$$

Point in ROC space

- The page 49 shows a ROC graph with five classifiers labeled A through E.
- Note that although the confusion matrix contains four numbers, we really only need two of the rates: either the true positive rate or the false negative rate, and either the false positive rate or the true negative rate.
- Given one from either pair the other can be derived since they sum to one. It is conventional to use the true positive rate (tp rate) and the false positive rate (fp rate), and we will keep to that convention so the ROC graph will make sense. Each discrete classifier produces an (fp rate, tp rate) pair corresponding to a single point in ROC space.

Several points in ROC space

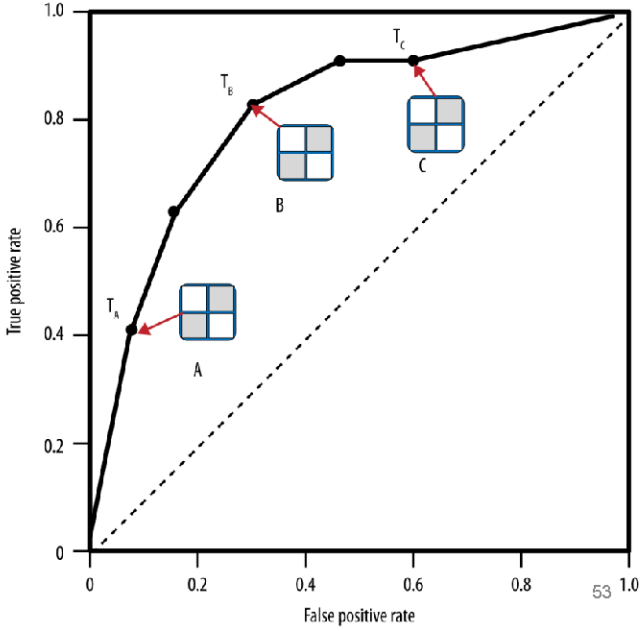
- Several points in ROC space are important to note.
- The lower left point (0, 0) represents the strategy of never issuing a positive classification; such a classifier commits no false positive errors but also gains no true positives.
- The opposite strategy, of unconditionally issuing positive classifications, is represented by the upper right point (1, 1).
- The point (0, 1) represents perfect classification, represented by a star. The diagonal line connecting (0, 0) to (1, 1) represents the policy of guessing a class.
- For example, if a classifier randomly guesses the positive class half the time, it can be expected to get half the positives and half the negatives correct; this yields the point (0.5, 0.5) in ROC space.

ROC curve in ranking model produce

- A ranking model produces a set of points (a curve) in ROC space.
- As discussed previously, a ranking model can be used with a threshold to produce a discrete (binary) classifier: if the classifier output is above the threshold, the classifier produces a Y, else an N. Each threshold value produces a different point in ROC space, as shown in Figure on the next page.

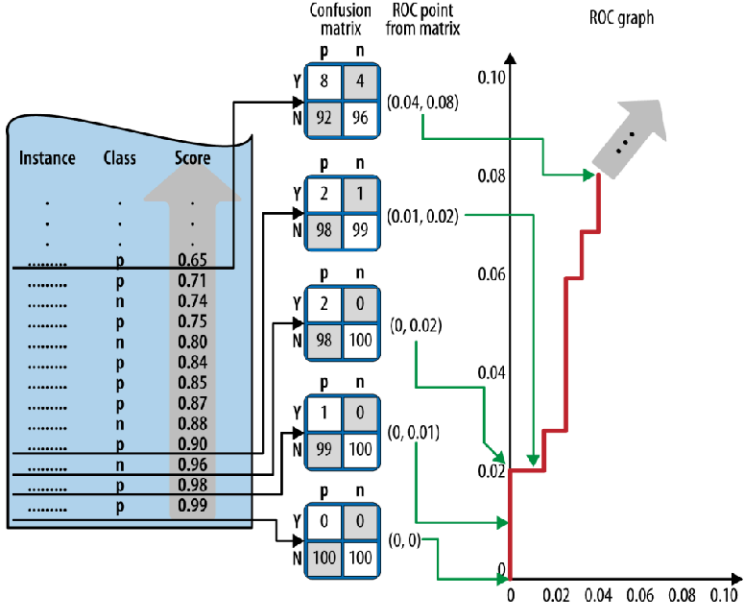
Example of ROC curve

Each different point in ROC space corresponds to a specific confusion matrix.



How a ROC "curve" is constructed from a test set

Conceptually, we may imagine sorting the instances by score and varying a threshold from $-\infty$ to $+\infty$ while tracing a curve through ROC space, as shown in the right figure



How a ROC “curve” is constructed from a test set

- An illustration of how a ROC “curve” (really, a stepwise graph) is constructed from a test set.
- The example set, at left, consists of 100 positives and 100 negatives. The model assigns a score to each instance and the instances are ordered decreasing from bottom to top.
- To construct the curve, start at the bottom with an initial confusion matrix where everything is classified as N. Moving upward, every instance moves a count of 1 from the N row to the Y row, resulting in a new confusion matrix.
- Each confusion matrix maps to a (fp rate, tp rate) pair in ROC space.

The Area Under the ROC Curve (AUC)

- An important summary statistic is the area under the ROC curve (AUC). As the name implies, this is simply the area under a classifier's curve expressed as a fraction of the unit square.
- Its value ranges from zero to one. Though a ROC curve provides more information than its area, the AUC is useful when a single number is needed to summarize performance, or when nothing is known about the operating conditions.

Reference

- Provost, Foster, and Tom Fawcett. Data Science for Business: What you need to know about data mining and data-analytic thinking. O'Reilly Media, Inc., 2013.

Data Science

Week 14

Representing and Mining Text