Confidence Bands for ROC Curves

Sofus A. Macskassy, Foster J. Provost and Michael L. Littman

Department of Information, Operations and Management Sciences

Leonard N. Stern School of Business, New York University

44 West 4th Street, New York, NY 10012

smacskas@stern.nyu.edu, fprovost@stern.nyu.edu, mlittman@cs.rutgers.edu

Center for Digital Economy Research
Stern School of Business
Working Paper IS-03-04

Under Review --- International Conference of Machine Learning 2003

Confidence Bands for ROC Curves*

Sofus A. Macskassy SMACSKAS@STERN.NYU.EDU
Foster J. Provost FPROVOST@STERN.NYU.EDU
Department of Information, Operations, & Management Sciences
Leonard N. Stern School of Business, New York University
44 W. 4th Street, Suite 8-82, New York, NY 10012-1126

Michael L. Littman MLITTMAN@CS.RUTGERS.EDU
Department of Computer Science
Rutgers University
110 Frelinghuysen Rd, Piscataway, NJ 08854-8019

Abstract

We address the problem of comparing the performance of classifiers. In this paper we study techniques for generating and evaluating confidence bands on ROC curves. Historically this has been done using one-dimensional confidence intervals by freezing one variable—the false-positive rate, or threshold on the classification scoring function. We adapt two prior methods and introduce a new radial sweep method to generate confidence bands. We show, through empirical studies, that the bands are too tight and introduce a general optimization methodology for creating bands that better fit the data, as well as methods for evaluating confidence bands. We show empirically that the optimized confidence bands fit much better and that, using our new evaluation method, it is possible to gauge the relative fit of different confidence bands.

1. Introduction/Motivation

We address the problem of comparing the performance of classifiers. Receiver-Operator Characteristic (ROC) analysis is an evaluation technique used in signal detection theory, which in recent years has seen an increasing use for types of diagnostic, machine-learning, and information-retrieval systems (Swets, 1988; Provost & Fawcett, 1997; Ng & Kantor, 2000; Provost & Fawcett, 2001; Macskassy et al., 2001). ROC graphs plot false-positive (FP) rates on the x-axis and true-positive (TP) rates on the y-axis. ROC curves are generated in a similar fashion to precision/recall curves, by varying a threshold across the output range of a scoring model, and observing the corresponding classification performances. Although ROC curves are isomorphic to precision/recall curves, they have the added benefits that they are insensitive to changes in marginal class distribution. Often the comparison of two or more ROC curves consists of either looking at the Area Under the Curve (AUC) or focusing on a particular part of the curves and identifying which curve dominates the other in order to select the best-performing algorithm.

Much less attention has been given to robust statistical comparisons of ROC curves. This paper addresses the creation of confidence bands on ROC curves. Prior work has considered sweeping across thresholds on the classification scoring function, creating confidence intervals around the TP/FP points for various thresholds (Fawcett, 2003), or sweeping across the FP rates and creating vertical confidence intervals around averaged TP levels (Provost et al., 1998). Confidence bands could be created by connecting these confidence intervals (as we will show). We examine $1 - \delta$ confidence bands on a model's ROC curve. We ask whether, assuming test examples are drawn from the same, fixed distribution, one indeed should expect that the model's ROC curves will fall within the bands with probability $1 - \delta$.

Figure 1 shows an example of what such prototypical confidence bands should look like with $\delta = 0.05$. In the figure, any ROC curve that does not lie completely in the shaded area would be said to be different from the mean curve with a 95% confidence.

In this paper we examine methods for creating and evaluating such confidence bands for a given learned model. As we will show, the bands created by prior techniques are too tight. We introduce a new technique that creates more realistic bands based on an empirical distribution. To these ends, we describe a framework for evaluating the fit of ROC confidence bands.

The rest of the paper is organized as follows. The next section discusses related work on creating confidence

*Macskassy, S.A., Provost, F.J., and Littman, M.L., "Confidence Bands for ROC Curves," CeDER Working Paper IS-03-04, Stern School of Business, New York University, NY, NY 10012. Jan 2003.

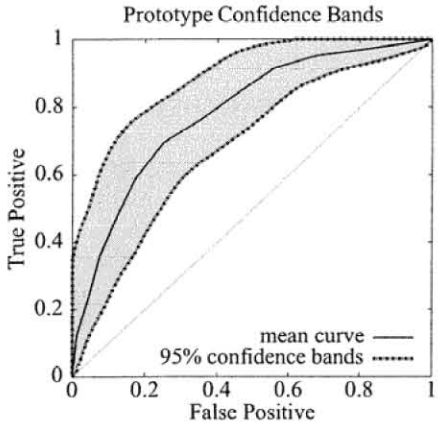

Figure 1. Prototype ROC Confidence Bands

intervals for ROC curves, followed by a section describing our methods for generating ROC confidence bands from confidence intervals. We then describe our evaluation methodology and a case study showing that our initial methods do not perform as well as expected. We then describe a general optimization-based methodology that can be applied to each of the band-generating techniques, and discuss a perhaps more reasonable evaluation measure and finally revisit the case study using the optimized method.

2. Related Work

Prior work on creating confidence intervals for ROC curves has for the most part been in the context of creating one-dimensional confidence intervals.

Pooling is a technique in which the i-th points from all the ROC curves in the sample are averaged (Bradley, 1997). This makes a strong assumption that the i-th points from all these curves are actually estimating the same point in ROC space, which is at best a doubtful assumption.

Vertical averaging looks at successive FP rates and averages the TPs of multiple ROC curves at that FP rate (Provost et al., 1998). By freezing the FP rate, it is possible to generate a (parametric) confidence interval for the TP rate based on the mean and variance; multiple curves are generated using cross-validation or other sampling techniques. A potential weakness of this method is the practical lack of independent control over a model's false-positive rates (Fawcett, 2003). (We also show that the distributional assumptions typically used with this technique are violated in our case study.)

Threshold averaging seeks to overcome the potential weakness of the vertical averaging by freezing the thresholds of the scoring model rather than the FP rate (Fawcett, 2003). It chooses a uniformly distributed subset of thresholds among the sorted set of all thresholds seen across the set of ROC curves in the sample. For each of these thresholds, it identifies the set of ROC points that would be generated using that threshold on each of the ROC curves. From these ROC points, the mean and standard deviations are generated for the FP and TP rates, giving the mean ROC point as well as vertical and horizontal confidence intervals.

Medical researchers also have examined the use of ROC curves and have introduced perhaps the most comprehensive techniques for creating confidence boundaries. One such technique is similar to that of threshold averages in that it creates a confidence boundary around each of the N ROC points associated with N discrete events in an underlying model (Tilbury et al., 2000). It does this by considering each axis as independent and considering an N-dimensional vector along each axis, where the i-th element in the vectors represent the i-th point in the ROC curve. Discretizing the values and assuming a binomial distribution, it then generates a probability distribution of the likelihood that the j-th value lies in each discretized cell. It map this probability density back into ROC space thereby generating confidence boundaries for each point in the ROC curve. These models are very complex and are not tractable for a large set of ROC points as is typically found in the ROC curves common in machine learning studies.

Others have looked the simpler problem of comparing an ROC curve to that of the expected performance of a random model (Macskassy, 2003). As the true theoretical bands can be generated under the assumption of a random predictor, this method was used to generate an ROC confidence band around the expected random performance given a specific test set.

Use of the bootstrap (Efron & Tibshirani, 1993) as a more robust way to evaluate expected performance has previously been used for evaluating cost-sensitive classifiers (Margineantu & Dietterich, 2000). In this work, bootstrapping was used to repeatedly draw predictions $p(i, j)$, where $p(i, j)$ is the probability that an instance of class j was predicted to be in class i. Using these sample predictions, it was possible to generate a final cost based on a cost-matrix. They did this repeatedly to generate a set of estimated costs, which they then used to generate confidence bounds on expected cost.

3. Generating Confidence Bands

In this section we describe our methodology for generating confidence bands for a classification model or modeling algorithm. The main assumption we make for being able to generate these confidence bands is that we can generate (or are given) a set of ROC curves. These can be generated by running a learning

algorithm on multiple training sets, testing on multiple testing sets, or resampling the same data. These ROC curves will be used to generate confidence bands about an average curve. We adapt two existing methods: vertical averaging and threshold averaging for generating confidence intervals. We also introduce a new radial-sweep method, which generates bands based on a radial sweep of the curves as we describe below.

Our methodology comprises the following steps.

1. Creating a distribution of ROC Curves
2. Generating 1-dimensional confidence intervals
   - Choosing a distribution
   - Sweeping across the ROC curves
3. Creating confidence bands from the confidence intervals

3.1. Creating the Distribution of ROC Curves

There exist various ways of generating a distribution of instances from which to generate a confidence interval. The most common methods, including Cross-validation (Kohavi, 1995), repeatedly split a data set into training and test sets. Each such split gives rise to a learned model, which can be evaluated against the test set—thereby generating one ROC curve per split. Although to our knowledge it has not been used before to generate multiple ROC curves, bootstrapping (Efron & Tibshirani, 1993) is a standard statistical technique that creates multiple samples by randomly drawing instances, with replacement, from a host sample (the host sample is a surrogate for the true population). We will describe how we use bootstrapping in Section 5.3.

3.2. Generating 1-Dimensional Confidence Intervals

3.2.1. Distribution Assumption

Most methodologies assume a normal distribution, but it may be that ROC points are not distributed normally. For example, for a given x-value (FP rate) the y-value (TP rate) is a proportion. So a binomial distribution may be appropriate. We consider three distributions for creating confidence intervals: normal, binomial, and empirical. Let us assume that we are given a sample distribution $\mathcal{D}$ of points along some dimension and a confidence threshold of δ.

We generate confidence intervals under the assumption of a *normal distribution* by calculating the mean μ and standard deviation σ of $\mathcal{D}$. We then look up the statistical constant, z, for a two-sided bound of δ confidence on a distribution size of $|\mathcal{D}|$ giving us a confidence interval of $\mu \pm \cdot z \cdot \sigma$.

For the *binomial distribution*, we calculate the variance as $V = \mu \cdot (1 - \mu)$, thus giving confidence interval $\mu \pm z \cdot \sqrt{\frac{V}{|\mathcal{D}|}}$.

For an *empirical distribution* we sort the values of $\mathcal{D}$ and choose v_l and v_u, such that v_l is the value is smaller than $\frac{1-\delta}{2}$ of all values and v_u is larger than $\frac{1-\delta}{2}$ of all values, thus $1 - \delta$ of all values lie between v_l and v_u.

We will examine these three techniques for calculating 1-dimensional intervals (*i.e.*, given a sample distribution of values for one variable). If not stated otherwise, results presented will be based on the empirical distribution.

3.2.2. Sweep Methods

So what are these dimensions along which the confidence intervals will be created? These are defined by how one "sweeps" across the collection of ROC curves. A sweep samples the set of points that define a point on the average ROC curve and the confidence interval about it. We use three different sweep orientations to sample ROC points. The first two are adaptations from existing methods and the last, the *radial sweep*, is a method we introduce in this paper.

The *vertical sweep* method sweeps a vertical line from FP = 0 to FP = 1, sampling the distribution of TPs from the collection of ROC curves at regular points along the sweep. For each such sampling at a fixed FP, TP confidence intervals can be created using any of the distribution assumptions mentioned above.

The *threshold sweep* method works a little differently than the vertical sweep. It sweeps along the thresholds on the model scores from $-\infty$ to $+\infty$, sampling the distribution of ROC points generated with each threshold. It then generates the mean (FP,TP) point for each sampled threshold and finds the confidence intervals of the FPs and TPs, using any of the distribution assumptions mentioned above.

Both of these consider only the x or y axis as the axes for orienting the confidence intervals. The drawback with both of these is that they do not take the curvature into account. For example, vertical intervals will tend to be much wider for smaller FP rates than for larger FP rates (due to the slopes of the curves). In fact, for cost-sensitive classification corresponding points on different ROC curves are points where the tangent lines to the curves have the same slope (Provost & Fawcett, 2001). Thus, one might argue that it is proper to have confidence intervals that are normal to an average curve. Producing intervals normal to an average curve is not easy (nor even well defined); for this paper we introduce a straightforward, intuitive approximation.

For the *radial sweep* method, rather than freezing the threshold or the FP rate, we instead do a radial sweep of the given curves by affixing one end of a vector to the lower right corner (at position (1,0)) and sweeping it radially from (0,0) to (1,1). At fixed angular intervals, we sample the points where all the given ROC

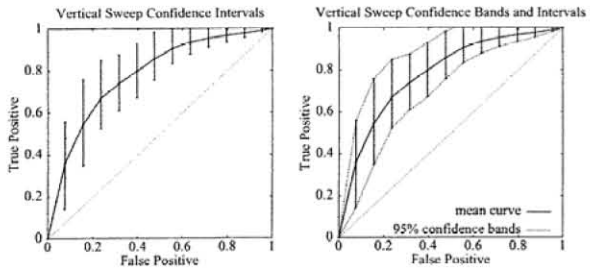

Figure 2. Transforming vertical sweep into confidence bands.

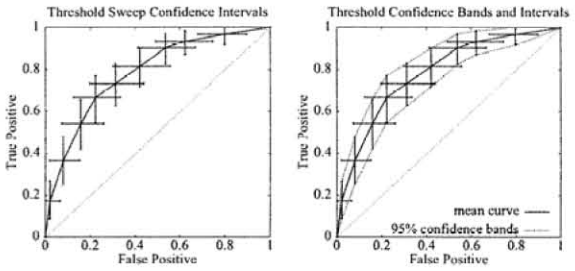

Figure 3. Transforming threshold sweep into confidence bands.

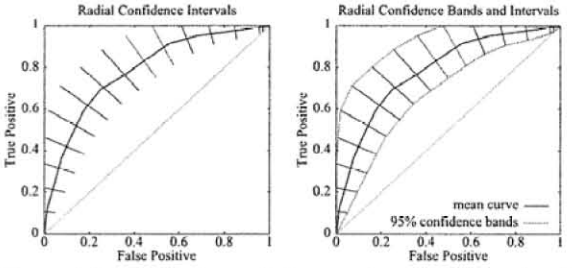

Figure 4. Transforming radial sweep into confidence bands.

curves intersect the vector. For each such sampling at angle θ—which ranges from 0 at (0,0) to $\frac{\pi}{2}$ at (1,1)—and for each ROC curve, we get a polar coordinate (θ,length) where the curve intersects the sweep vector. The length in the polar coordinates (the distance of the point from the lower right corner) is the variable for which we will compute the confidence interval—again using any of the distribution assumptions mentioned above. Although the sweep vector rarely is truly orthogonal to the ROC curve tangent at any given intersection, the sweep method does provide us with a straightforward approximation.

All of our sweep methods require three parameters:

1. The confidence δ, which we set to 0.05 for a 95% confidence bound throughout this paper. We did test with other δ's (0.10 and 0.01) with similar results as those presented below.
2. The distribution assumption under which the confidence intervals are generated. We test under all three distribution assumptions mentioned above: normal, binomial, and empirical.
3. The set of points to sample along the sweep, which we set to a uniformly distributed 100 points. This number can be changed depending on how fine-grained a curve is needed.[1]

3.3. Creating Confidence Bands from Confidence Intervals

3.3.1. Vertical Sweep

Vertical sweep can be adapted directly to generate confidence bands rather than a set of distinct confidence intervals. What we do is to consider all the upper (lower) interval points as the points making up the upper (lower) band. Figure 2 illustrates this methodology. For each FP (0.00 through 0.99—1.0 always has a TP of 1.00), we generate a distribution of possible TPs across all the sampled ROC curves and generate the bands based on this distribution.

3.3.2. Threshold Sweep

This method is a little more problematic to adapt to our framework as there are various ways to deal with two confidence intervals. In this paper we chose the simplest approach: discount the confidence interval for FP and only use the confidence interval for TP. Because of this, the bands we generate turn out to be somewhat conservative and containment probably is underestimated. Figure 3 illustrates the transformation as well as the drawback. In the figure, we clearly see that some FP intervals reach outside the confidence bands (opposite to the vertical intervals, the horizontal intervals will tend to be larger for higher FP rates). We are currently investigating more robust and better-performing ways to generate confidence bands from threshold sweeps.

3.3.3. Radial Sweep

As with the vertical sweep method, generating the confidence bands from this method is straightforward. For each sampled vector at angle θ, we can generate the far (near) point from the polar confidence intervals which we then map back into ROC space to generate the points for the upper (lower) confidence band. Figure 4 illustrates how this method is applied.

4. Evaluation

The key question we ask in this paper is how good are these bands? As with confidence intervals on a single variable, we would like to be able to say that given a δ, the bands generated can be expected to fully contain the curve from a given model with a probability of $1-\delta$ (assuming that new test instances come from the same distribution). As we will show, for none of the methods proposed above does this hold. Later, we will introduce an optimization method below for generating better bands, as well as new evaluation measures that give a sense of how well the bands do fit.

[1] While this is a free variable that will have some effect on the overall fit of the bands, we do not investigate its effect in this paper.

5. Case Study

5.1. Data

We now present a case study using the Covertype data set from the UCI repository (Blake & Merz, 1998). We chose this data set because its large size enabled us to do more in-depth testing, across a wide range of training- and test-set sizes. The Covertype data set consists of 581,012 instances having 54 features, 10 being numerical and the rest being ordinal or binary. While it has seven classes, there is a large variation in class membership sizes. To study the ROC curves, we chose examples of the two classes with the most instances, giving us a data set of 495,141 instances (57.2% base error rate).

5.2. Learning Method

We use a modified C4.5R8 (Quinlan, 1993) that generates a Probability Estimation Tree (PET) (Provost & Domingos, 2002). PETs are generated by considering the predictions made for each leaf in a decision tree. If a leaf matches p positive examples and n negative examples, the probability of class membership in the positive example is $\frac{p}{p+n}$. Further, to produce a better class-probability estimate, we apply a simple Laplace correction (Niblett, 1987) under the assumption of uniform class distribution $\frac{1}{C}$ for C classes—giving us a final probability estimate of $\frac{p+1}{p+n+2}$, as we have 2 classes. Further, we do no pruning of the tree, as standard pruning does not consider differences in scores that do not affect 0/1 loss (but may deflate the ROC curve) (Provost & Domingos, 2002).

5.3. Bootstrap-based Evaluation

To generate and evaluate confidence bands, we use the following method based on a bootstrapped empirical sampling distribution.

1. Randomly split the complete data set into a training set of 256,000 instances and a test set of 125,000 instances, keeping these two sets disjoint.
2. Sample with replacement from each of these two sets to generate a training set, multiple "fitting" sets, and multiple test sets:
   (a) Fix the training size, sample a training set of that size, and learn a classifier.
   (b) Fix the test size and repeatedly generate "fitting" sets of that size. For each fitting set, generate an ROC curve for the model. The result is a set of ROC curves, one per fitting set.
   (c) Generate confidence bands based on the ROC curves generated in the fitting step (b).
   (d) Do 1000 sampling runs. For each run we pick a test set using the same size as in (b), from which we then generate an ROC curve. We then calculate how many of the resulting 1000 ROC curves fall completely within the generated confidence bands.

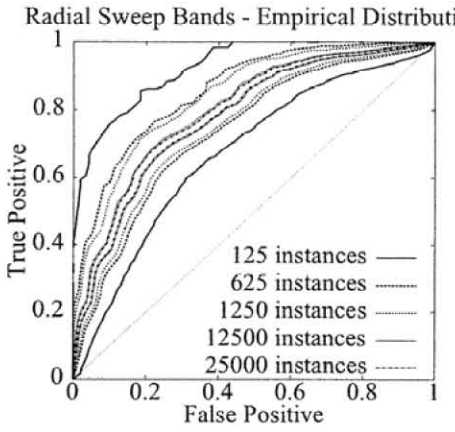

Figure 5. ROC Bands using various test sizes.

This methodology has three parameters: the training size, the test size, and the number of sampling runs used in step (b) to generate the confidence curves. We examine the sensitivity to each of these parameters in the next section. Note that for this paper, we do not consider variance in curves due to the training set—only confidence bands on the ROC curve of a particular (learned) classifier. However, a similar methodology would apply to the generation of confidence bands for a learning algorithm.

5.4. Trends in Confidence Bands

In this section we examine the experimental parameters identified above, and choose values for our evaluation. Unless stated otherwise, we will use the radial sweep method under the empirical distribution assumption for the figures presented. All other sweeps and distributions had similar performances, though this combination is the best performer among the methods described thus far.

5.4.1. Training Size

This parameter is the least interesting for this particular case study. As the training size increases, the ROC curves become higher as would be expected. However, while this has some effect on the width of the confidence bands, it is more a matter of considering different learned models than of how to generate good bands for a given model. As such, we do not consider this to be an important dimension for further discussion here and fix the training size to 1000 instances.

5.4.2. Test size

Test-set size should have an obvious effect on the bands generated. We fixed the test size to 125, 625, 1250, 6250, 12500 and 25000 instances (0.1%, 0.5%, 1%, 5%, 10% and 20%, respectively, of the complete test set). As the test-set size increases, the approximate confidence intervals generated by any of our sweep methods become narrower and therefore so do our confidence

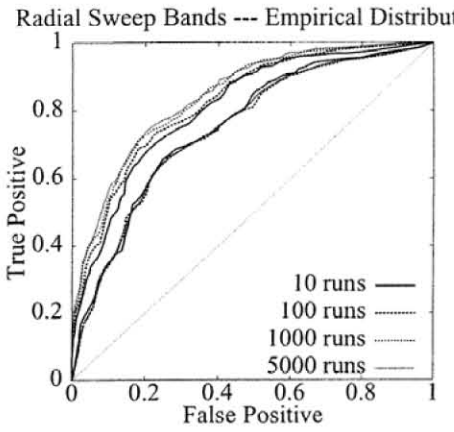

Figure 6. ROC Bands using varying number of sampling runs.

bands. This is a general statistical property—with too few samples, the estimate of the confidence interval tends to be inaccurate and biased to be too wide. The same thing is happening in the ROC space. Figure 5 illustrates this effect clearly.

To limit our presentation for this paper, we fix the test size to 12500, though the results hold for other sizes as well.

5.4.3. Number of Sampling Runs

The number of sampling runs used to create the empirical distribution (step 2(b) in Section 5.3) is the last free parameter that we consider. In order to generate the ROC bands, we need to have a sample of ROC curves from which to generate these bands. The question to answer is how many such ROC curves—the number of sampling runs—are needed to generate reasonable bands. While the effect of this variable is not as intuitive as the test or training size, it still does have an effect as can be seen in Figure 6. While the lower band is fairly stable we see that the upper band widens with more sampling runs. (This would be expected from a distribution with a long tail.)

As we observe from Figure 6, the upper bands between using 1000 and 5000 sampling runs were very similar. Based on this observation, we fix the number of sampling runs to 1000, though our results hold for other values as well.

5.5. How Good Are The Bands?

Having fixed our experimental parameters, let us now ask our main question: do the $1-\delta$ confidence bands actually contain $1-\delta$ of the empirical distribution? Our mechanism allows us to ask two variations on this question: do the bands contain $1-\delta$ of the "fitting" distribution? Do the bands contain $1-\delta$ of the "test" distribution?

As per our bootstrap-based methodology, we randomly

	distribution assumption					
	empirical		normal		binomial	
Method	$\hat{\delta}_{\text{fitting}}$	$\hat{\delta}_{\text{test}}$	$\hat{\delta}_{\text{fitting}}$	$\hat{\delta}_{\text{test}}$	$\hat{\delta}_{\text{fitting}}$	$\hat{\delta}_{\text{test}}$
radial	73.5	63.9	51.9	41.5	00.0	00.0
vertical	31.6	01.7	42.7	00.0	00.0	00.0
threshold	00.9	00.0	00.8	00.0	00.0	00.0

Table 1. How many ROC curves fall within the bands of each method using a given distribution for generating bands? $\hat{\delta}_{\text{fitting}}$ is the percentage of samples used to generate the bands and $\hat{\delta}_{\text{test}}$ is the percentage of samples drawn afterwards.

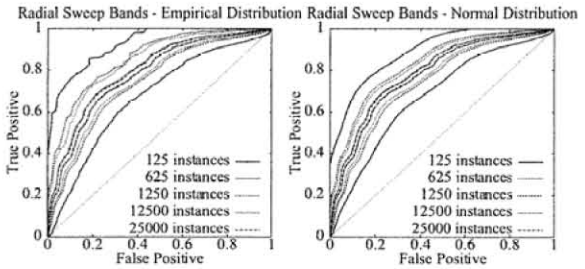

Figure 7. Comparison of bands generated under the empirical and normal distribution assumptions.

sampled test sets of size 12,500 with replacement from the original test set of 125,000 and counted how many of the 1000 ROC curves fell within each band. We did this for each of our three methods using each of the three distribution assumptions. Table 1 shows how many ROC curves fall within the bands of each method using a given distribution assumption for generating the bands. $\hat{\delta}_{\text{fitting}}$ is the percentage based on the "fitting" samples that were used to generate the bands and $\hat{\delta}_{\text{test}}$ is the percentage of ROC curves based on samples drawn after the bands had been generated.

Surprisingly, none of the bands get anywhere near the 95% that we would expect. In particular, we see that the binomial distribution assumption generates very bad bands and that neither the vertical sweep nor threshold sweep methods perform as well as the radial sweep method.[2] Interestingly, bands generated under the normal distribution assumption did not perform as well as the bands generated under the empirical distribution. Figure 7 shows the bands generated under these two distribution assumptions side by side. Note that they are very similar in shape, though the empirical distribution bands are much more jagged. The empirical bands are noticeably wider (on the "high" side). Would one expect ROC curves to be distributed normally with respect to the vertical, threshold, or radial dimensions? We do not have a good answer, but the empirical bands do seem to fit better.

What remains to be addressed is the poor containment

[2] Recall that the bands generated by the threshold sweep method are overly conservative and that better bands may be found with a better connecting method.

Method	$\hat{\delta}_{\text{fitting}}$	$\hat{\delta}_{\text{test}}$
opt-radial	96.8%	86.2%

Table 2. Percentage of curves contained within the bands generated by the optimized radial sweep method.

of the bands. While the radial sweep method produced the best fit, it still fell far short of the expected containment of the empirical distribution of ROC curves. Is it possible to produce better bands? Is there a better way to evaluate ROC bands? The rest of the paper presents first steps toward answering these questions.

6. Optimized ROC Bands

None of the methods performed as expected, even on the ROC curves (the fitting curves) that were used to generate the bands in the first place. We propose to revisit the way in which these bands were generated and optimize them such that they fit the empirical distribution of curves better. To do so, we use the following optimization methodology:

1. Generate an empirical distribution using a method appropriate for the problem domain (e.g., our bootstrap mechanism).
2. Select a method for generating bands (*e.g.*, radial sweep) based on some underlying distribution (*e.g.*, the empirical distribution).
3. Optimize the bands with respect to an objective function that is suitable for the problem domain.

We instantiate this methodology by generating the sampling distribution as given before. Because the radial sweep method performed well using the empirical distribution, we choose these as the baseline from which we will optimize. For the optimization step, for this paper we adopt a very simple method:

1. For each sampling in the radial sweep generate a set of polar coordinates. Let θ_α be the angle of the vector used to draw this sample, and let N be the number of ROC curves in the distribution.
2. Sort the values by length, giving us the sorted set $l_{\theta_\alpha,1} < \ldots < l_{\theta_\alpha,N}$.
3. Starting at the outermost bands ($L = 1$ and $U = N$), we define the candidate lower band as the set of points $l_{\theta_i,L}$ for $i = 1 \ldots N$ and the candidate upper band as the set $l_{\theta_i,U}$ for $i = 1 \ldots N$. Set W to the number of curves in our sample that fall completely within (or lie on) these bands.
4. Increase L by 1 and decrease U by one and recalculate W.
5. Continue until the candidate bands contain fewer than $1 - \delta$ of the "fitting" curves and use $U + 1$ and $L - 1$ to generate the final bands.

Table 2 shows the performance of this Optimized Radial Sweep method, *opt-radial*, using the same evaluation as before with same parameter settings. As we can see, this method was able to generate bands that had a better containment than the non-optimized methods. However, it still did not fit the test set as well as expected.

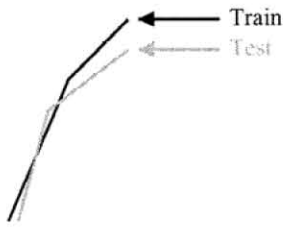

Figure 8. Example of point outside the curve.

7. Evaluation Revisited

One possible explanation for the below-expected containment even of the optimized method is that maybe there is no good way to generate bands that fit well due to the chaotic behavior often found in ROC curves where they crisscross many times (as seen in Figure 6). With curves such as these it may be unlikely to be able to do any better than the convex hull in order to get the expected containment. Looking more closely, the convex hull of the fitting samples used to generate the bands might still not be enough. If even one point falls outside the convex hull as shown in Figure 8, the complete curve is not contained. If the fitting samples are chaotic and crisscross many times, why would new samples behave differently? They may be very likely have at least one point outside the bands found in the original samples. Maybe we should not require the bands fully contain an ROC curve, but instead to contain "almost all" of the ROC curve. If we can quantify "almost all" then we can evaluate how well the bands fit the data with respect to this measure.

The measure we use for this evaluation is based the percentage ϵ of the points of an ROC curve that falls outside the bands. For a set of confidence bands, we calculate ϵ for each of the ROC curves in the empirical distribution, and identify $\hat{\epsilon}$ such that $1 - \delta$ of all the curves have $\epsilon \leq \hat{\epsilon}$. To use such δ, ϵ confidence bands, a new ROC curve would be considered statistically different if more than $\hat{\epsilon}$ of its points fall outside the bands. We can then evaluate the fitness of a type of band by assessing its $\hat{\epsilon}$.

8. Case Study Revisited

We now revisit our case study and compute the $\hat{\epsilon}$'s for each method. Figure 9 graphs, for our four sweep methods using the empirical distribution, the percent of curves contained as we increase ϵ. The vertical line is 95% $(1 - \delta)$ containment. As is clear from the graph, the optimized radial sweep outperformed all the other methods though all methods were able to achieve 95% containment at varying ϵs. Table 3 shows the $\hat{\epsilon}$'s needed by each method using the normal and empiri-

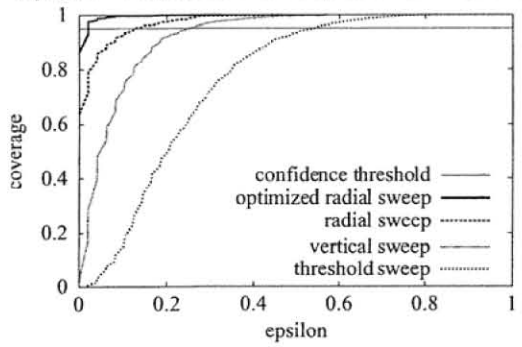

Figure 9. ϵ coverage for different sweep methods.

	distribution assumption			
	empirical		normal	
Method	$\hat{\epsilon}$train	$\hat{\epsilon}_{\text{test}}$	$\hat{\epsilon}$train	$\hat{\epsilon}_{\text{test}}$
opt-radial	0.0000	0.0208	—	—
radial	0.1765	0.1250	0.2353	0.2083
vertical	0.2843	0.2500	0.2451	0.2245
threshold	0.5588	0.5417	0.5588	0.5306

Table 3. What ϵ's are needed to achieve a 95% containment.

cal distributions.[3] For example, the optimized sweep completely contained (by construction) the 95% of the fitting curves, and required $\epsilon = 0.02$ to contain 95% of the test curves. The other methods required considerably higher ϵ values to achieve 95% containment.

9. Discussion and Limitations

In this paper we evaluated various methods for generating confidence bands for ROC curves. We introduced a new radial sweep method for generating confidence bands around the ROC curve and developed a general framework for optimizing such bands using bootstrapping techniques. We showed that methods based on existing techniques produced bands that were far too narrow. The optimized method performed considerably better, but still was too narrow. We then introduced a new measure to evaluate the containment of ROC confidence bands and showed how our optimized radial sweep method required relatively little leeway to achieve proper containment.

However, although we introduced the radial sweep method to approximate confidence bands that are normal to an ROC curve at any given point, a better technique might yield improved results. One question that we did not investigate here was how sensitive the bands are to the number of points sampled along the sweep. Further, although we introduced the notion of optimizing the bands, we only considered a straightforward and simplistic optimization in this paper. Finally, it is still an open question whether the bands found are too loose in certain regions of the curve and too tight in others. These are all issues that we hope to investigate further.

We hope this work takes a significant step toward more robust comparisons of machine learning methods using ROC analysis.

[3]Note that we have dropped the comparison to the binomial distribution as it performed so badly in the previous evaluation.

Acknowledgments

We would like to thank Tom Fawcett for his pointers to related work and for many discussions about ROC curves, Haym Hirsh for his feedback early in the design stages and Matthew Stone who initially suggested using the bootstrap for evaluating ROC curves.

This work is sponsored in part by the Defense Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-01-2-585. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency (DARPA), the Air Force Research Laboratory, or the U.S. Government.

References

Blake, C. L., & Merz, C. J. (1998). UCI Repository of machine learning databases.

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition, 7*, 1145–1159.

Efron, B., & Tibshirani, R. (1993). *An introduction to the bootstrap.* Chapman & Hall.

Fawcett, T. (2003). *Roc graphs: Notes and practical considerations for data mining researchers*Technical Report HPL-2003-4). HP Labs.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the 14th International Joint Conference on Artificial Intelligence* (pp. 1137–1143). San Francisco, CA.

Macskassy, S. A. (2003). *New techniques in intelligent information filtering.* Doctoral dissertation, Rutgers University.

Macskassy, S. A., Hirsh, H., Provost, F. J., Sankaranarayanan, R., & Dhar, V. (2001). Intelligent information triage. *The 24th Annoual Internationl Conference on Research and Development in Information Retrieval (SIGIR).* New Orleans, LA.

Margineantu, D. D., & Dietterich, T. G. (2000). Bootstrap methods for the cost-sensitive evaluation of classifiers. *International Conference on Machine Learning, ICML-2000* (pp. 582–590).

Ng, K.-B., & Kantor, P. (2000). Predicting the effectiveness of naive data fusion on the basis of system characteristics. *Journal of American Society for Information Science, 51*, 1177–1189.

Niblett, T. (1987). Constructing decision trees in noisy domains. *Proceedings of the Second European Working Session on Learning* (pp. 67–78). Sigma, Bled, Yugoslavia.

Provost, F. J., & Domingos, P. (2002). Tree induction for probability-based rankings. *Machine Learning.* to appear.

Provost, F. J., & Fawcett, T. (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining* (pp. 445–453).

Provost, F. J., & Fawcett, T. (2001). Robust classification for imprecise environments. *Machine Learning, 42*, 203–231.

Provost, F. J., Fawcett, T., & Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms. *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 445–453). San Francisco, CA: Morgan Kaufman.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning.* San Mateo, CA: Morgan Kaufmann.

Swets, J. A. (1988). Measuring the accuracy of diagnostic systems. *Science, 240*, 1285–1293.

Tilbury, J., Eetvelt, P. V., Garibaldi, J., Curnow, J., & Ifeachor, E. (2000). Receiver operating characteristic analysis for intelligent medical systems – a new approach for finding non-parametric confidence intervals. *IEEE Transactions on Biomedical Engineering, 47*, 952–963.