Evaluation Measures of the Classification Performance of Imbalanced Data Sets

Qiong Gu[1,2], Li Zhu [2], and Zhihua Cai[2]

[1] Faculty of Mathematics & Computer Science, Xiangfan University, Xiangfan, Hubei, 441053, China
[2] School of Computer, China University of Geosciences, Wuhan, Hubei, 430074, China
gujone@163.com, cugzhuli@163.com, zhcai@cug.edu.cn

Abstract. Discriminant Measures for Classification Performance play a critical role in guiding the design of classifiers, assessment methods and evaluation measures are at least as important as algorithm and are the first key stage to a successful data mining. We systematically summarized the evaluation measures of Imbalanced Data Sets (IDS). Several different type measures, such as commonly performance evaluation measures and visualizing classifier performance measures have been analyzed and compared. The problems of these measures towards IDS may lead to misunderstanding of classification results and even wrong strategy decision. Beside that, a series of complex numerical evaluation measures were also investigated which can also serve for evaluating classification performance of IDS.

Keywords: Evaluation, classification performance, imbalanced data sets.

1 Introduction

The purpose of evaluation in Machine Learning is to determine the usefulness of our learned classifiers or of our learning algorithms on various collections of data sets. Most measures in use today focus on a classifier's ability to identify classes correctly. Assessment methods and evaluation measures of classification performance play a critical role in guiding the design of classifiers. Even the most widely used methods such as measuring accuracy or error rate on a test set has severe limitations. Thus the modification of classification algorithms in some extent equals the improvement of criterions. Many efforts have been conducted to design/develop more advanced algorithms to solve the classification problems. In fact, the assessment methods and evaluation measures are at least as important as algorithm and is the first key stage to a successful data mining.

The purpose of this paper is to give the reader an intuitive idea of what could go wrong with our commonly used evaluation methods. In particular, we show, through examples, that since evaluation metrics summarize the system's performance, they can, at times, obscure important behaviors of the hypotheses or algorithms under consideration. Since the purpose of evaluation is to offer simple and convenient ways to judge the performance of a learning system and/or to compare it to others, evaluation methods can be seen as summaries of the systems' performance.

Z. Cai et al. (Eds.): ISICA 2009, CCIS 51, pp. 461–471, 2009.

The outline of the paper is as follows. Several different types commonly performance evaluation measures, such as numeric measure and visualizing classifier performance measure, have been analyzed and compared in section 2, Section 3 focuses on the issue of performance metrics. More specifically, it demonstrates, through a number of examples the shortcomings of Accuracy, Precision/Recall and ROC. Beside that, a series of complex numerical evaluation measures were also investigated which can also serve for evaluating classification performance of IDS in section 4. Finally, the conclusion is drawn in Section 5.

2 Commonly Performance Evaluation Measures

Methods for evaluating the performance of classifiers fall into two broad categories: numerical and graphical. Numerical evaluations produce a single number summarizing a classifier's performance, whereas graphical methods depict performance in a plot that typically has just two or three dimensions so that it can be easily inspected by humans. Examples of numerical performance measures are accuracy, precision, recall^{+}, recall^{-} and AUC. Examples of graphical performance evaluations are Lift chart, ROC curve[1, 2], precision-recall curve[3], cost curve[4],et al.

2.1 Numerical Value Performance Measure

Most of the studies in IDS mainly concentrate on two-class problem as multi-class problem can be simplified to two-class problem. By convention, the class label of the minority class is positive, and the class label of the majority class is negative. Table 1 illustrates a confusion matrix of a two-class problem. The first column of the table is the actual class label of the examples, and the first row presents their predicted class label. TP and TN denote the number of positive and negative examples that are classified correctly, while FN and FP denote the number of misclassified positive and negative examples respectively.

Table 1. A confusion matrix for a two-class classification

Recognized / Actually Class	Predicted as Positive Class	Predicted as Negative Class
Actually Positive class	True Positive(TP)	False Negative(FN)
Actually Negative class	False Positive(FP)	True Negative(TN)

Based on Table 1, the performance metrics are defined as:

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{True Positive Rate}(Acc^{+})=\frac{TP}{TP+FN}=Recall^{+}=Sensitivity$$

$$\text{True Negative Rate}(Acc-)=\frac{TN}{TN+FP}=Recall-=Specificity$$

$$\text{Positive Predictive Value}=\frac{TP}{TP+FP}=Precision$$

Traditionally, accuracy is the most commonly used measure for these purposes. However, for classification with the class imbalance problem, accuracy is no longer a proper measure since the rare class has very little impact on accuracy as compared to the prevalent class[5]. this measurement is meaningless to some applications where the learning concern is the identification of the rare cases. Accuracy does not distinguish between the numbers of correct labels of different classes. For any classifier, there is always a trade off between true positive rate and true negative rate; and the same applies for recall and precision. In the case of learning extremely imbalanced data, quite often the rare class is of great interest. In many cases, it is desirable to have a classifier that gives high prediction accuracy over the minority class (*Acc*+), while maintaining reasonable accuracy for the majority class (*Acc*−).

2.2 Graphical Performance Analysis with Probabilistic Classifiers

Graphical methods are especially useful when there is uncertainty about the misclassification costs or the class distribution that will occur when the classifier is deployed. In this setting, graphical measures can present a classifier's actual performance for a wide variety of different operating points (combinations of costs and class distributions), whereas the best a numerical measure can do is to represent the average performance across a set of operating points.

2.2.1 Lift Chart

The lift chart is a standard detection evaluation method to validate machine learning algorithms. The lift chart represents an effective measure for the validation of the detection process and on whether a given attack classification is valid or not. The *x*-axis represents the number of examples of the test set that were selected according to the probabilistic ranking generated by the classifier. The *y*-axis represents the percentage of positive examples in the subset of selected examples. This percentage is calculated over the total number of examples in the test set.

$$x = \mathrm{Yrate}(t) = \frac{TP(t) + FP(t)}{P + N},\ y = \mathrm{TP}(t) \tag{1}$$

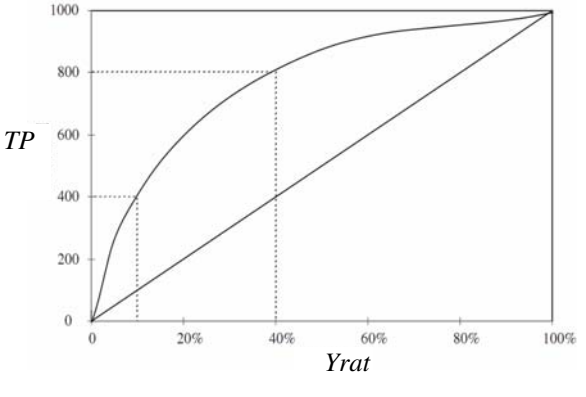


Fig. 1. A hypothetical lift chart

Figure 1 shows the lift chart for the attack type parameter upsweep. Normally, we'd like to be in a lift chart is near the upper left-hand corner, at the very best, the further to the northwest the better. The upper lift point (0,1000) denotes the ideal case for accurate detection with minimum cost. Lift curve also indicates how far the detector model is effective from the point of view of reducing the false alarms.

2.2.2 ROC Curves

ROC curve[6] is one of the popular metrics to evaluate the learners for IDS. It is a two-dimensional graph in which TP_{rate} is plotted on the *y*-axis and FP_{rate} is plotted on the *x*-axis. ROC curve depicts relative trade-offs between benefits (TP_{rate}) and costs (FP_{rate}). Consider that the minority class, whose performance will be analyzed, is the positive class. Some classifiers have parameter for which different settings produce different ROC points. Figure 2 shows a ROC curve, Typically this is a discrete set of points, including (0,0) and (1,1), which are connected by line segments. The lower left point (0,0) represents a strategy that classifies every example as belonging to the negative class. The upper right point represents a strategy that classifies every example as belonging to the positive class. The point (0,1) represents the perfect classification, and the line x = y represents the strategy of random guessing the class. The ideal model is one that obtains 1 True Positive Rate and 0 False Positive Rate (1,0). A ROC curve gives a good summary of the performance of a classification model. To compare several classification models by comparing ROC curves, it is hard to claim a winner unless one curve clearly dominates the others over the entire space[7].

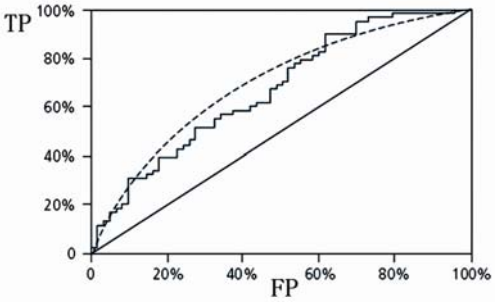


Fig. 2. A sample ROC curve

2.2.3 Recall-Precision Curves

Some researchers define recall and precision, and a list of yes's and no's represent a ranking of retrieved documents.In just the same way as ROC curves and lift charts, except that the axes are different, the PR curves are hyperbolic in shape,the desired operating point is toward the upper right.

Figure 3 shows a Recall-Precision curve. An important difference between ROC space and PR space is the visual representation of the curves. PR curves can expose differences between algorithms that are not apparent in ROC space. These curves, taken from the same learned models on a highly-skewed dataset, highlight the visual difference between these spaces. The goal in ROC space is to be in the upper-left-hand corner, and when one looks at the ROC curves they appear to be fairly close to

optimal. In PR space the goal is to be in the upper-right-hand corner and the PR curves show that there is still vast room for improvement. Each dataset contains a fixed number of positive and negative examples. It is revealed in the study that there exists a sound relationship between ROC and PR spaces. For a given dataset of positive and negative examples, there exists a one-to-one correspondence between a curve in ROC space and a curve in PR space, such that the curves contain exactly the same confusion matrices, if Recall ≠ 0. For a fixed number of positive and negative examples, one curve dominates a second curve in ROC space if and only if the first dominates the second in PR space.

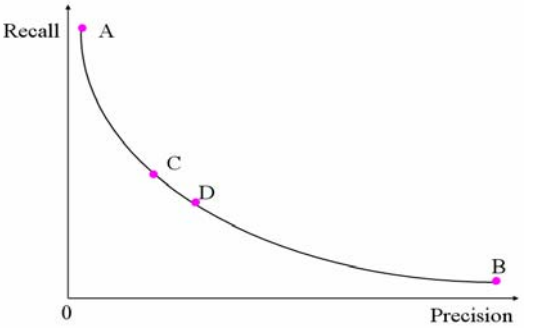
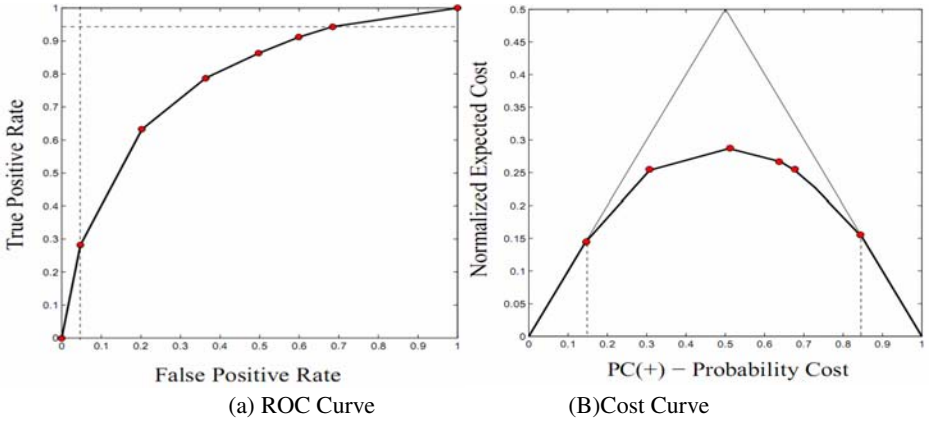


Fig. 3. Recall-Precision curve

2.2.4 Cost Curves

Cost curves are a different kind of display on which a single classifier corresponds to a straight line that shows how the performance varies as the class distribution changes. Cost curves are perhaps the ideal graphical method in this setting because they directly show performance as a function of the misclassification costs and class distribution. Figure 4 shows ROC Curve and corresponding Cost Curve.



(a) ROC Curve (B)Cost Curve

Fig. 4. ROC Curve and corresponding Cost Curve

In particular, the *x*-axis and *y*-axis of a cost curve plot are defined as follows. The *x*-axis of a cost curve plot is defined by combining the two misclassification costs and the class distribution-represented by p(+), the probability that a given instance is positive-into a single value, PC(+), using the following formula:

$$PCF(+) = \frac{p(+) * C(-|+)}{p(+) * C(-|+) + p(-) * C(+|-)} \tag{2}$$

where C(-|+)is the cost of a false negative and C(+|-)is the cost of a false positive. Classifier performance, the y-axis of a cost curve plot, is normalized expected cost(NEC), NEC ranges between 0 and 1. Cost curves directly show performance on their y-axis, whereas ROC curves do not explicitly depict performance. This means performance and performance differences can be easily seen in cost curves but not in ROC curves.

When applied to a set of cost curves the natural way of averaging two-dimensional curves produces a cost curve that represents the average of the performances represented by the given curves. By contrast, there is no agreed upon way to average ROC curves, and none of the proposed averaging methods produces an ROC curve representing average performance. Cost curves allow confidence intervals to be estimated for a classifier's performance, and allow the statistical significance of performance differences to be assessed. The confidence interval and statistical significance testing methods for ROC curves do not relate directly to classifier performance.

Table 2 summarizes the four different ways utilized in evaluating the same basic trade off. Either the proportion can be increased by using a smaller coverage, or the coverage can be increased at the expense of the proportion. Different techniques can be plotted as different lines on any of these graphical charts. Each point on a lift chart, ROC curve, or recall–precision curve represents a classifier, typically obtained using different threshold values for a method. Cost curves represent each classifier using a straight line, and a suite of classifiers will sweep out a curved envelope whose lower limit shows how well that type of classifier

Table 2. Different Measures Used to Evaluate the False Positive versus the False Negative Trade Off

Technique	Domain	Axes	Explanation of axes	at the very best place
Lift chart	marketing	*TP* subset size	Number of *TP* $\frac{TP+FP}{TP+FN+TN+FN}\times 100\%$	near the upper left-hand corner
ROC curve (the jagged line)	communications	*TP* rate *FP* rate	$tp=\frac{TP}{TP+FN}\times 100\%$ $fp=\frac{FP}{FP+TN}\times 100\%$	the northwest corner
PR Curve (hyperbolic in shape)	Information retrieval	Recall Precision	$tp=\frac{TP}{TP+FN}\times 100\%$ $\frac{TP}{TP+FP}\times 100\%$	toward the upper right hand corner
Cost curve	Classification with known error costs.	Normalized Expected Cost Probability Cost Function	$PC(+)=\frac{FN\times Pc(+)+FP\times(1-Pc(+))}{p(+)*C(+\|-)+p(-)*C(-\|+)}$	at the bottom of graphics

can do if the parameter is well chosen. Although such measures may be useful if costs and class distributions are unknown, one method must be chosen to handle all situations. ROC curves are a very useful tool for visualizing and evaluating classifiers.

3 Shortcomings of Some Performance Metrics

In this section, we consider the three most commonly metrics in Machine Learning: Accuracy, Precision/Recall and ROC Analysis. In each case, we begin by stating the advantages of these methods, continue by explaining the shortcomings they each have.

3.1 Shortcomings of Accuracy

Accuracy is the simplest, most intuitive evaluation measure for classifiers, but in learning extremely imbalanced data; the accuracy is often not an appropriate measure of performance. It is worth noting that Accuracy does not distinguish between the types of errors it makes..

We illustrate the problem more specifically with the following example: Consider two classifiers represented by the two confusion matrices of Table 3. These two classifiers behave quite differently. The one symbolized by the confusion matrix on left does not classify positive examples very well, getting only 200 out of 600 right. On the other hand, it does not do a terrible job on the negative data, getting 500 out of 600 well classified. The classifier represented by the confusion matrix on the right does the exact opposite, classifying the positive class better than the negative class with 500 out of 600 versus 200 out of 600. It is clear that these classifiers exhibit quite different strengths and weaknesses and shouldn't be used blindly on a data set. Yet, both classifiers exhibit the same accuracy of 58.3%.

Table 3. The trouble with Accuracy: Two confusion matrices yielding the same accuracy despite serious differences

Prediction Class	Algorithm A		Algorithm B	
True class	Positive	Negative	Positive	Negative
Positive P=600	200	400	500	100
Negative N=600	100	500	400	200

3.2 Shortcomings of Precision/Recall

Precision and Recall still have a relatively straightforward interpretation,Precision assesses to what extent the classifier was correct in classifying examples as positives, while Recall assesses to what extent all the examples that needed to be classified as positive were so. Precision and Recall have the advantage of not falling into the problem encountered by Accuracy. Indeed, considering, again, the two confusion matrices of Table 3, we can compute the values for Precision and Recall and obtain the following results:

Precision = 66.7% and Recall = 33.3% in the left case, and
Precision = 55.6% and Recall = 83.3% in the right

These results, indeed, reflect the strength of the right classifier on the positive data, with respect to the left classifier. This is a great advantage over accuracy.

More specifically, consider, as an extreme situation, the confusion matrices of table 4.The matrix on left is the same as the left matrix of Table 3, whereas the one on the right represents a new classifier tested on a different data set. Although both classifiers have the same Precision and Recall of 66.7% and 33.3%, respectively, it is clear that the classifier represented by the confusion matrix on the right presents a much more severe shortcoming than the one on left since it is incapable of classifying true negative examples as negative. This suggests that Precision and Recall are quite blind, in a certain respect, and might be more useful when combined with accuracy or when applied to both the positive and the negative class.

Table 4. The trouble with Precision and Recall: Two confusion matrices with the same values of precision and recall, but very different behaviors

Prediction Class	Data set A		Data set B	
True class	Positive	Negative	Positive	Negative
Positive	200	400	200	400
Negative	100	500	100	0

3.3 Shortcomings of ROC

ROC Analysis has intuitive appeal. We only consider ROC Analysis: the performance measure it uses. The great advantage of this performance measure is that it separates the algorithm's performance over the positive class from its performance over the negative class. As a result, it does not suffer from either of the two problems we considered before. Indeed, in the case of Table 3, the left classifier is represented by ROC graph point (0.167, 0.333) while the right classifier is represented by point (0.667, 0.833). This clearly shows the tradeoff between the two approaches: although the left classifier makes fewer errors on the negative class than the right one, the right one achieves much greater performance on the positive class than the left one. It is, thus clear that ROC Analysis has great advantages over Accuracy, Precision and Recall. Nonetheless, there are reasons why ROC analysis is not an end in itself, either. This is illustrated numerically in the following example. Consider the two confusion matrices of Table 5. The classifier represented by the confusion matrix on the right generates a point in ROC space that is on the same vertical line as the point generated by the classifier represented by the confusion matrix on left (x = FP rate = 0.25%), but that is substantially higher (by 22.25%, [recall_left= 40%; recall_right = 62.5%]) than the one on left. This suggests that the classifier on the right is a better choice than the one on the left, yet, when viewed in terms of precision, we see that the classifier on left is much more precise, with a precision of 95.24% than the one on the right, with a precision of 33.3%. Ironically, this problem is caused by the fact that ROC Analysis nicely separates the performance of the two classes, thus staying away from the previous problems encountered by Accuracy and Precision/Recall.

Table 5. Two confusion matrices representing the same point in ROC space, but with very different precisions

Prediction Class / True class	Algorithm A		Algorithm B	
	Positive	Negative	Positive	Negative
Positive	200	300	500	300
Negative	10	4000	1000	400000

4 Complex Numerical Evaluation Measures

Some measures that caught our attention have been used in medical diagnosis to analyze tests. They combine sensitivity and specificity and their complements.

4.1 F-Measure

F-measure is a popular evaluation metric for imbalance problem[8]. It is a kind of combination of recall and precision, which are effective metrics for information retrieval community where the imbalance problem exists. F-measure also depends on the β factor, which is a parameter that takes values from 0 to infinity and is used to control the influence of recall and precision separately. It can be shown that when β=0 then F-measure reduces to precision and conversely when $\beta \rightarrow \infty$ then F-measure approaches recall.

$$\text{F-measure} = \frac{(1+\beta) * \text{Precision} * \text{Recall}}{\beta * \text{Precision} + \text{Recall}} \tag{3}$$

When β = 1 then F-measure is suggested to integrate these two measures as an average, In principle, F-measure represents a harmonic mean between recall and precision.

$$\text{F-measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

The harmonic mean of two numbers tends to be closer to the smaller of the two. Hence, a high F-measure value ensures that both recall and precision are reasonably high.

4.2 G-Mean

When the performance of both classes is concerned, both True Positive Rate (TP_{rate}) and True Negative Rate (TN_{rate}) are expected to be high simultaneously. Kubat et al[9] suggested the G-mean defined as:

$$\text{G-mean} = \sqrt{\text{TP}_{\text{rate}} \cdot \text{TN}_{\text{rate}}} \tag{5}$$

G-mean measures the balanced performance of a learning algorithm between these two classes. The comparison among harmonic, geometric, and arithmetic means are illustrated in[8].This measure tries to maximize accuracy in order to balance both classes at the same time. It is an evaluation measure that allows to simultaneously maximizing the accuracy in positive and negative examples with a good trade-off.

4.3 Youden's Index

The avoidance of failure complements accuracy, or the ability to correctly label examples. Youden's index γ[10]evaluates the algorithm's ability to avoid failure-equally weights its performance on positive and negative examples:

$$\gamma = \text{sensitivity} + \text{specificity} - 1 \tag{6}$$

Youden's index has been traditionally used to compare diagnostic abilities of two tests[11]. It summarizes sensitivity and specificity and has linear correspondence balanced accuracy (a higher value of γ means better ability to avoid failure):

$$\gamma = 2\text{AUC}_b - 1 \tag{7}$$

4.4 Likelihoods

If a measure accommodates both sensitivity and specificity, but treats them separately, then we can evaluate the classifier's performance to finer degree with respect to both classes. The following measure combining positive and negative likelihoods allows us to do just that[11]:

$$\text{Positive Likelihood Ratio, } LR^+ = TPR/FPR = Sensitivity/(1 - Specificity) \tag{8}$$

$$\text{Negative Likelihood Ratio, } LR- = (1 - TPR)/(1 - FPR) = (1 - Sensitivity)/Specificity \tag{9}$$

A higher positive and a lower negative likelihood mean better performance on positive and negative classes respectively. If an algorithm does not satisfy this condition, then "positive" and "negative" likelihood values should be swapped.Relations depicted show that the likelihoods are an easy-to-understand measure that gives a comprehensive evaluation of the algorithm's performance.

4.5 Discriminatory Power

Another measure summarizes sensitivity and specificity is Discriminatory power (DP) [12]:

$$DP = \frac{\sqrt{3}}{\pi}(\log(Sensitivity/(1 - Sensitivity)) + \log(Specificity/(1 - Specificity))) \tag{10}$$

To the best of our knowledge, until now DP has been mostly used in ML for feature selection. The algorithm is a poor discriminate if DP< 1, limited if DP< 2, fair if DP< 3, good – in other cases.

5 Conclusions

In general, there is no a generalized evaluation measure for various kind of classification problems. A good strategy to identify a proper evaluation measure should largely depend upon specific application requirement. Choose appropriate evaluation measure according to different background can help people make correct judgment to the algorithm classification performance. We hope that this very simple review of some of the

problems surrounding evaluation will sensitize Machine Learning and Data Mining researchers to the issue and encourage us to think twice, prior to selecting and applying an evaluation method.

References

1. Pepe, M.S.: Receiver Operating Characteristic Methodology. Journal of the American Statistical Association 95, 308–311 (2000)
2. Fawcett, T.: ROC graphs: Notes and practical considerations for researchers. Machine learning 31 (2004)
3. Davis, J., Goadrich, M.: The relationship between precision-recall and ROC curves. In: The 23rd International Conference on Machine Learning (ICML 2006), pp. 233–240. ACM, New York (2006)
4. Drummond, C., Holte, R.C.: Cost curves: An improved method for visualizing classifier performance. Machine learning 65, 95–130 (2006)
5. Weiss, G.M.: Mining with rarity: a unifying framework. Newsletter of the ACM Special Interest Group on Knowledge Discovery and Data Mining 6, 7–19 (2004)
6. Bradley, A.P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition 30, 1145–1159 (1997)
7. Provost, F., Fawcett, T.: Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions. In: The 3rd International Conference on Knowledge Discovery and Data Mining, pp. 43–48 (1997)
8. van Rijsbergen, C.J.: Information Retrieval. Butterworths, London (1979)
9. Kubat, M., Holte, R.C., Matwin, S.: Machine Learning for the Detection of Oil Spills in Satellite Radar Images. Machine Learning 30, 195–215 (1998)
10. Youden, W.J.: Index for rating diagnostic tests. Cancer 3, 32–35 (1950)
11. Biggersta, B.J.: Comparing diagnostic tests: a simple graphic using likelihood ratios. Statistics in Medicine 19, 649–663 (2000)
12. Blakeley, D.D., Oddone, E.Z., Hasselblad, V., Simel, D.L., Matchar, D.B.: Noninvasive carotid artery testing: a meta-analytic review. Am. Coll. Physicians 122, 360–367 (1995)