Metodološki zvezki, Vol. 3, No. 1, 2006, 89-108

ROC Curve, Lift Chart and Calibration Plot

Miha Vuk[1], Tomaž Curk[2]

Abstract

This paper presents ROC curve, lift chart and calibration plot, three well known graphical techniques that are useful for evaluating the quality of classification models used in data mining and machine learning. Each technique, normally used and studied separately, defines its own measure of classification quality and its visualization. Here, we give a brief survey of the methods and establish a common mathematical framework which adds some new aspects, explanations and interrelations between these techniques. We conclude with an empirical evaluation and a few examples on how to use the presented techniques to boost classification accuracy.

1 Introduction

In research result presentation of machine learning systems, we observe their performance under a specific setting. The way we observe their performance is tightly connected with the specific problem that they are solving. Classification problems are most common in machine learning and this paper will present three techniques for improving and evaluating classification models (called classifiers) used for automatic classification. ROC curve, lift chart and calibration plot are techniques for visualizing, organizing, improving and selecting classifiers based on their performance. They facilitate our conception of classifiers and are therefore useful in research and in result presentation.

This paper gives a quick introduction to all three techniques and practical guidelines for applying them in research. This part is already known from literature. The main contribution of this paper is a deeper theoretical background with some new explanations of areas under curves and a description of new interrelations between these three techniques and between derived measures of classification performance.

The paper is divided in two parts. The first part (Sections 3 to 6) covers the theory. In Section 3 we introduce the concept of a classifier and explain the difference between binary and probabilistic classifiers. In Section 4 we present ROC curve, area under the curve (AUC) and show how to use ROC curve to improve classification accuracy. In Section 5 we present lift chart and describe the interrelation between area under the ROC curve and lift chart curve. In Section 6 we introduce the calibration plot and show how ROC curve, lift chart and the area under both curves can be derived from the calibration

[1] Department of Knowledge Technologies, Jožef Stefan Institute, Slovenia; miha.vuk@ijs.si
[2] University of Ljubljana, Faculty of Computer and Information Science, Slovenia; tomaz.curk@fri.uni-lj.si
The two authors contributed equally to this work.

plot. In the second part (Section 7) of this paper we report on an empirical validation of the proposed method to improve classification accuracy using ROC analysis and give some practical examples. We show the presented techniques and approaches on different classifiers and data sets. The paper's main contributions can be found in Sections 4.1, 5.1 and 6.

2 Related work

Most books on data mining and machine learning (Witten, 2000; Phyle, 1999) dedicate relatively short sections to a description of ROC curves and lift charts. ROC curves [19, 20, 21] have long been used in signal detection theory to depict the tradeoff between hit rates and false alarm rates of classifiers (Egan, 1975; Centor, 1991). They are widely used by the decision making community and in medical diagnostic systems (Hanley and McNeil, 1982). A deeper explanation and implementation details for applying ROC analysis in practical research can be found in (Fawcett, 2003; with Provost, 2001, 1997).

Lift chart [14, 15, 16] is well know in the data mining community specialized in marketing and sales applications (Berry and Linoff, 1999). Apart from their primarily presentational purpose lift charts have not been much studied.

The term *calibration* and using graphs to present calibration quality is common in all scientific and engineering fields including statistics and data mining. There is not a single common name for calibration plots as they are often referenced as calibration map, calibration graph, calibration chart, etc. In this paper we will use the term *calibration plot*. Good references for calibration classifiers are Cohen, Goldszmidt (2004) and Zadrozny, Elkan (2002).

3 Classifiers

One of the important tasks in data mining and machine learning is classification. Given a set of examples that belong to different classes we want to construct a classification model (also called a classifier) that will classify examples to the correct class.

When constructing a classifier we usually assume that the test set of examples is not known, but there are some other previously known data that we can use to extract the knowledge. The phase of constructing the classifier is called training or learning and the data used in this phase are called training (learning) data or *training (example) set*. Afterwards we evaluate the classifier on some other data called test data or *test set*.

It is often hard or nearly impossible to construct a perfect classification model that would correctly classify all examples from the test set. Therefore we have to choose a suboptimal classification model that best suits our needs and works best on our problem domain. This paper presents different quality measures that can be used for such classifier selection. It also presents the techniques for visual comparison of different classification models.

An example: We want to develop a classification model to diagnose a specific illness. Each patient is described by several attributes on which decisions of our model are based.

Patients in the training set have an already known diagnosis (belong to either class *ill* or *healthy*) and data about these patients are used to learn a classifier. The classifier is then applied on the test set of patients where only attributes' values without class information are passed to the classifier. Finally, predictions are compared with the medically observed health status of patients in the test set, to assess the classifier's predictive quality.

In the example above we could use a classifier that makes a binary prediction (*i.e.* the patient is either ill or healthy) or a classifier that gives a probabilistic class prediction [3] to which class an example belongs. The fist is called *binary* classifier and the later is called *probabilistic* classifier.

3.1 Binary classifiers

When dealing with two class classification problems we can always label one class as a positive and the other one as a negative class. The test set consists of P positive and N negative examples. A classifier assigns a class to each of them, but some of the assignments are wrong. To assess the classification results we count the number of true positive (*TP*), true negative (*TN*), false positive (*FP*) (actually negative, but classified as positive) and false negative (*FN*) (actually positive, but classified as negative) examples.
It holds

$$TP + FN = P \tag{3.1}$$

and

$$TN + FP = N \tag{3.2}$$

The classifier assigned $TP + FP$ examples to the positive class and $TN + FN$ examples to the negative class.

Let us define a few well-known and widely used measures:

$$FPrate = \frac{FP}{N} \qquad TPrate = \frac{TP}{P} = Recall \qquad Yrate = \frac{TP + FP}{P + N} \tag{3.3}$$

$$Precision = \frac{TP}{TP + FP} \qquad Accuracy = \frac{TP + TN}{P + N} \tag{3.4}$$

Precision and *Accuracy* are often used to measure the classification quality of binary classifiers. Several other measures used for special purposes can also be defined. We describe them in the following sections.

3.2 Probabilistic classifiers

Probabilistic classifiers assign a score or a probability to each example. A probabilistic classifier is a function $f : X \to [0, 1]$ that maps each example x to a real number $f(x)$. Normally, a threshold t is selected for which the examples where $f(x) \geq t$ are considered positive and the others are considered negative.

[3] Some classifiers return a score between 0 and 1 instead of probability. For the sake of simplicity we shall call them also *probabilistic* classifiers, since an uncalibrated score function can be converted to a probability function. This will be the topic of Section 4.

This implies that each pair of a probabilistic classifier and threshold t defines a binary classifier. Measures defined in the section above can therefore also be used for probabilistic classifiers, but they are always a function of the threshold t.

Note that $TP(t)$ and $FP(t)$ are always monotonic descending functions. For a finite example set they are stepwise, not continuous.

By varying t we get a family of binary classifiers. The rest of this paper will focus on evaluating such families of binary classifiers (usually derived from probabilistic classifier). The three techniques we mentioned in the introduction each offer its own way to visualize the classification "quality" of the whole family. They are used to compare different families and to choose an optimal binary classifier from the family.

4 ROC curve

Suppose we have developed a classifier that will be used in an alarm system. Usually we are especially interested in portion of alarms caused by positive events (that should really fire an alarm) and portion of alarms caused by negative events. The ratio between positive and negative events can vary during time, so we want to measure the quality of our alarm system independently of this ratio. In such cases the ROC curve (*receiver operating characteristic*) (Fawcett (2003), [19, 20, 21]) is the right tool to use.

ROC graph is defined by a parametric definition

$$x = FPrate(t), \quad y = TPrate(t). \tag{4.1}$$

Each binary classifier (for a given test set of examples) is represented by a point $(\mathit{FPrate}, \mathit{TPrate})$ on the graph. By varying the threshold of the probabilistic classifier, we get a set of binary classifiers, represented with a set of points on the graph. The ROC curve is independent of the $P : N$ ratio and is therefore suitable for comparing classifiers when this ratio may vary.

An example of a probabilistic classifier and its results on a given test set are shown in Table 1. Figure 1 shows the ROC curve for this classifier.

ROC graph in the above example is composed of a discrete set of points. There are several ways to make a curve out of these points. The most common is using the *convex hull* that is shown in Figure 2.

Such representation also has a practical meaning, since we are able to construct a binary classifier for each point on the convex hull. Each straight segment of a convex hull is defined with two endpoints that correspond to two classifiers. We will label the first one A and the other one B. A new (combined) classifier C can be defined. For a given value of parameter $\alpha \in (0, 1)$ we can combine the predictions of classifiers A and B. We take the prediction of A with probability α and the prediction of B with probability $1 - \alpha$. The combined classifier C corresponds to the point on the straight segment and by varying the parameter α we cover the whole straight segment between A and B. If the original ROC graph corresponds to probabilistic classifier, its convex hull also corresponds to a probabilistic classifier that is always at least as good as the original one.

Convex hull is just one approach for constructing a ROC curve from a given set of points. Other approaches are presented in the next section.

Table 1: Probabilistic classifier. The table shows the assigned scores and the real classes of the examples in the given test set.

Inst#	Class	Score	Inst#	Class	Score
1	p	.9	11	p	.4
2	p	.8	12	n	.39
3	n	.7	13	p	.38
4	p	.6	14	p	.37
5	p	.55	15	n	.36
6	p	.54	16	n	.35
7	n	.53	17	p	.34
8	n	.52	18	n	.33
9	p	.51	19	p	.30
10	n	.505	10	n	.1

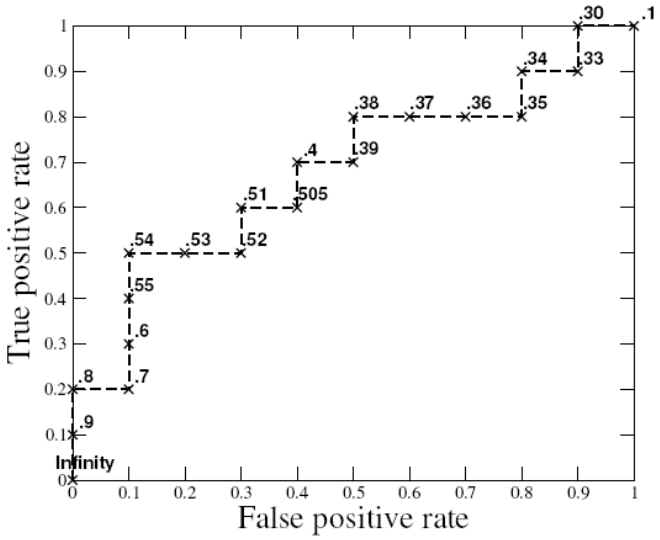

Figure 1: ROC graph of the probabilistic classifier from Table 1. Thresholds are also marked on the graph.

4.1 Area under curve — AUC

Area under ROC curve is often used as a measure of quality of a probabilistic classifier. As we will show later it is close to the perception of classification quality that most people have. AUC is computed with the following formula:

$$A_{ROC} = \int_0^1 \frac{TP}{P} \quad d\frac{FP}{N} = \frac{1}{P\,N} \int_0^N TP \;\; dFP \tag{4.2}$$

A random classifier (*e.g.* classifying by tossing up a coin) has an area under curve 0.5, while a perfect classifier has 1. Classifiers used in practice should therefore be somewhere in between, preferably close to 1.

Now take a look at what A_{ROC} really expresses. We will use the basic stepwise ROC

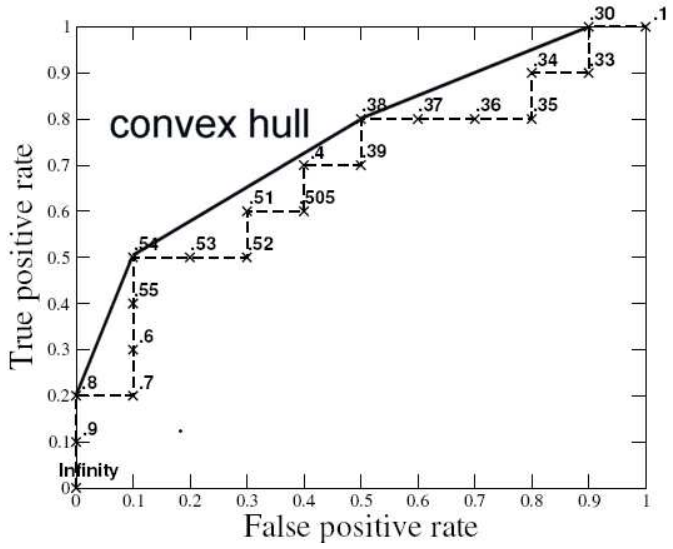

Figure 2: ROC graph with a convex hull of the probabilistic classifier from Table 1.

curve (*e.g.* the dashed line in Figure 1) obtained from a given set of points and we will also assume that our probabilistic classifier assigns a different score to each example. Then the above formula instructs us: for each negative example count the number of positive examples with a higher assigned scores than the negative example, sum it up and divide everything with $P\,N$. This is exactly the same procedure as used to compute the probability that a random positive example has a higher assigned score than random negative example.

$$A_{ROC} = P(\text{random positive example} > \text{random negative example}) \tag{4.3}$$

Remark $P(X)$ denotes the probability of event X and has no connection with P which denotes the number of positive examples in test set.

If we allow that several positive and several negative examples can have the same assigned score, then there are several equivalent approaches to construct a ROC curve, each resulting in different AUC. Table 2 shows an example of such classifier and Figure 3 shows ROC curves constructed using approaches defined below.

The equation 4.3 still holds true if two adjacent points are connected with lower sides of a right-angled triangle (see Figure 3). We will label the AUC computed using this approach as A_{ROC1}.

$$A_{ROC1} = P(\text{random positive example} > \text{random negative example}) \tag{4.4}$$

A more natural approach is connecting the two adjacent points with a straight line. AUC computed using this approach will labeled as A_{ROC2}. To compute A_{ROC2} we must define the Wilcoxon test result W (see Hanley, McNeil (1982)).

$$S(x_p, x_n) = \begin{cases} 1, & \text{if } x_p > x_n \\ \frac{1}{2}, & \text{if } x_p = x_n \\ 0, & \text{if } x_p < x_n \end{cases} \tag{4.5}$$

$$W = \frac{1}{PN} \sum_{x_p \in pos.} \sum_{x_n \in neg.} S(x_p, x_n) \tag{4.6}$$

Then the following equality holds true

$$A_{ROC2} = W = P(\text{random pos.} > \text{random neg.}) + \frac{1}{2} P(\text{random pos.} = \text{random neg.}) \tag{4.7}$$

To better comprehend the difference between A_{ROC1} and A_{ROC2} please see Figure 3. Note that none of the areas defined in this section correspond to the area under the convex hull.

Table 2: Probabilistic classifier that assigns the same score (0.4) to two examples in a given test set.

Example Number	Class	Score
1	p	0.9
2	p	0.6
3	n	0.4
4	p	0.4
5	n	0.2

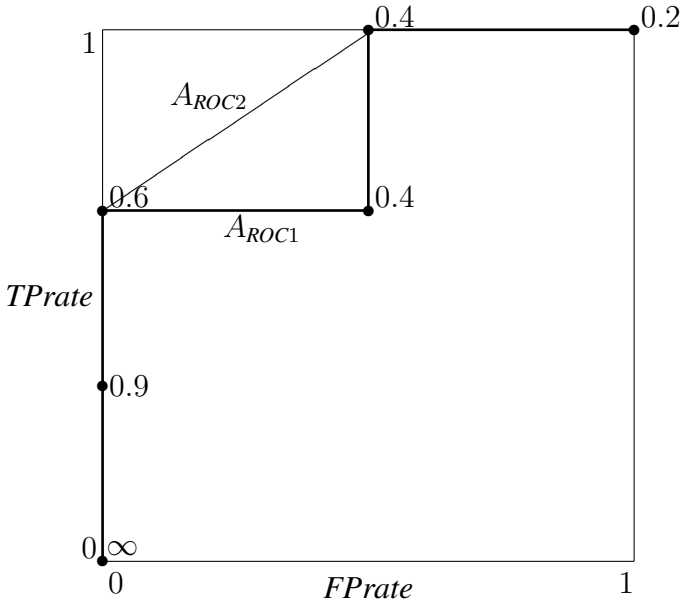

Figure 3: Comparison of methods A_{ROC1} and A_{ROC2}.

Using the formulae above we can compute A_{ROC1} and A_{ROC2} directly (without a graph) and they both have a mathematical meaning and interpretation. In general, both measures tell us how well a classifier distinguishes between positive and negative examples. While it is an important aspect of a classification, it is not a guarantee of good classification accuracy nor of other classifier's qualities.

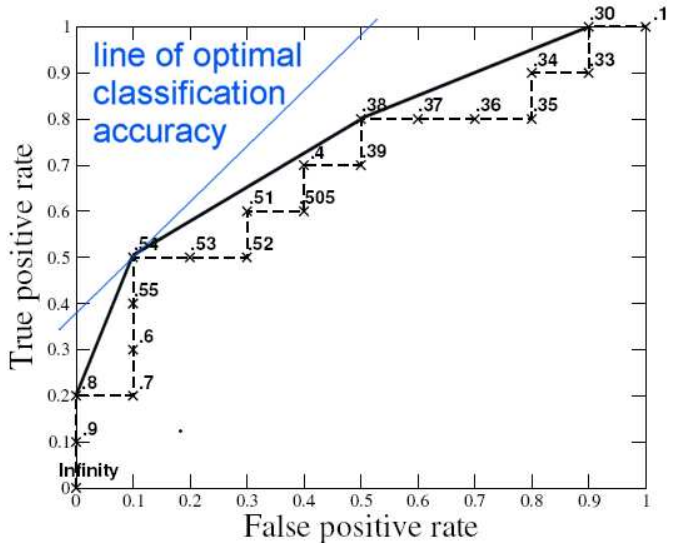

Figure 4: Tangent on a ROC curve when $\frac{N}{P} = \frac{6}{5}$ and the point of optimal classification accuracy.

If the proportion of examples with the same assigned value between all examples is small, then the difference between A_{ROC1} and A_{ROC2} becomes negligible.

4.2 Optimal classification accuracy

Every point on a ROC curve corresponds to a binary classifier, for which we can calculate the classification accuracy and other quality measures. To compute the classification accuracy we need to know the $P : N$ ratio. Knowing this we can find a point on the graph with optimal classification accuracy. Even the probabilistic classifier with a perfect ROC curve ($A_{ROC} = 1$), has 100% classification accuracy in only one point (upper left corner).

An algorithm for finding the point of optimal classification accuracy is given next. Using the formula for accuracy

$$Accuracy = \frac{TP + TN}{P + N} = \frac{TPrate\ P + (1 - FPrate)N}{P + N} \tag{4.8}$$

we can get a definition of a straight line connecting the points with equal classification accuracy (iso-performance or iso-parametric line) (Fawcett (2003)).

$$TPrate = \frac{Accuracy(P + N) - (1 - FPrate)N}{P} = \frac{N}{P}FPrate + \frac{Accuracy(P + N) - N}{P} \tag{4.9}$$

Using this formula we get a set of parallel lines each representing different classification accuracy. The best one goes through the upper left corner and the worst one goes through lower right one. The point of optimal classification accuracy of a probabilistic classifier is the intersection of the iso-performance tangent and the ROC curve. A graphical example is given in Figure 4.

Note that other points on a curve can have considerably lower classification accuracy. Also, it is easy to imagine two ROC curves with the same A_{ROC}, but with considerably different classification accuracy (for the same $P : N$ ratio).

We can define other iso-performance lines for other measures of classification quality (*e.g.* error cost). Another important aspect of ROC performance analysis is the ability to assign weights to positive and negative errors. Weights influence just the angle of the tangential line and thus influence the selection of the optimal binary classifier.

5 Lift chart

Let us start with an example. A marketing agency is planning to send advertisements to selected households with the goal to boost sales of a product. The agency has a list of all households where each household is described by a set of attributes. Each advertisement sent costs a few pennies, but it is well paid off if the customer buys the product. Therefore an agency wants to minimize the number of advertisements sent, while at the same time maximize the number of sold products by reaching only the consumers that will actually buy the product.

Therefore it develops a classifier that predicts the probability that a household is a potential customer. To fit this classifier and to express the dependency between the costs and the expected benefit the lift chart can be used. The number off all potential customers P is often unknown, therefore *TPrate* cannot be computed and the ROC curve cannot used, but the lift chart is useful in such settings. Also the TP is often hard to measure in practice; one might have just a few measurements from a sales analysis. Even in such cases lift chart can help the agency select the amount of most promising households to which an advertisement should be sent. Of course, lift charts are also useful for many other similar problems.

Although developed for other purposes, lift chart (Witten, Frank (2000), [14, 16, 15]) is quite similar to the ROC curve. We will therefore focus on the differences between them. The reader can find the missing details in Section 4.

Lift chart is a graph with a parametric definition

$$x = \mathit{Yrate}(t) = \frac{\mathit{TP}(t) + \mathit{FP}(t)}{P + N}, \quad y = TP(t). \tag{5.1}$$

Similarly as explained for ROC curve (Section 4), each binary classifier corresponds to a point on a graph in this parametric space. By varying the threshold of a probabilistic classifier we get a set of points, *i.e.* set of binary classifiers. The curve we get by drawing a convex hull of the given (binary) points is called a *lift chart*. Again it holds true that each point on the convex hull corresponds to a combined classifier and the probabilistic classifier that corresponds to the convex hull is always at least as good as the original classifier from which the hull was derived.

Figure 5 shows an example lift chart for the marketing operation described above. Unlike the ROC curve, lift chart depends on the $P : N$ ratio.

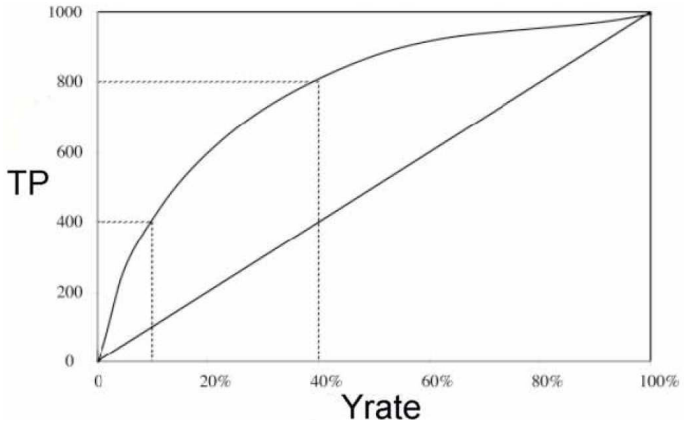

Figure 5: A typical marketing lift char for sending advertisements to 1000 households.

5.1 Area under curve - AUC

Area under lift chart A_{lift} can be used as a measure of classification quality of a probabilistic classifier. It is computed with the following formula.

$$A_{lift} = \int_0^1 TP \quad d\frac{TP+FP}{P+N} = \frac{1}{P+N}\left(\int_0^P TP \;\; dTP + \int_0^N TP \;\; dFP\right) \tag{5.2}$$

$$A_{lift} = \frac{1}{P+N}\;\left(\frac{P^2}{2} + P\,N \cdot A_{ROC}\right) \tag{5.3}$$

Random classifier (coin flips) has an area under the curve $\frac{1}{P+N}\;(\frac{P^2}{2} + P\,N \cdot A_{ROC}) = \frac{P}{2}$, while a perfect classifier has an area P. Classifiers used in practice should therefore be somewhere in between. As one can see from equation 5.2, A_{lift} always depends on $P : N$ ratio. If P is much smaller that N then we can use the approximation $A_{lift} \approx A_{ROC} \cdot P$.

Like A_{ROC}, A_{lift} also has a nice probabilistic (statistical) explanation. For this purpose we will use a stepwise lift chart, where each point on the graph defines a column (or a step) with the point in its upper left corner. The sum of areas of all columns gives us A_{lift}. This definition is consistent with the formula 5.2 that instructs us: For each example count the number of positive examples with a higher assigned score than the chosen example, sum it up and divide by $P + N$. This is exactly the same procedure used to compute the average number of positive examples with a higher assigned score than a random example.

$$A_{lift1} = P \cdot P(\text{random positive example} > \text{random example}) \tag{5.4}$$

We could say that A_{lift1} shows how good the classifier distinguishes positive examples from all examples, but it does not seem to have more practical meaning.

In practice we usually do not want a stepwise lift chart, but a smooth curve, where the adjacent points are connected with straight lines. Area under such curve can be expressed with the following less elegant equation.

$$\begin{aligned} A_{lift2} &= \int_0^1 (TP + \frac{dTP}{2}) \quad d\frac{TP+FP}{P+N} \;\; = \\ &= \frac{1}{P+N}\left(\int_0^P (TP + \frac{dTP}{2}) \;\; dTP + \int_0^N (TP + \frac{dTP}{2}) \;\; dFP\right) \end{aligned} \tag{5.5}$$

We use the integral notation even when $d\frac{TP+FP}{P+N}$ is not arbitrary small. In our case this corresponds to the difference in $Yrate$ between two adjacent points on the graph and TP corresponds to the TP value of the leftmost point of such an adjacent point pair. (Imagine the stepwise lift chart from definition of A_{lift1}.)

The graphical example of A_{lift1} and A_{lift2} is shown in Figure 6. Note that both formulae (5.4 and 5.4) hold true when several different examples have the same assigned score.

Table 3: Probabilistic classifier that assigns the same score (0.4) to two examples in the given test set.

Num	Class	Score
1	p	0.9
2	p	0.6
3	n	0.5
4	n	0.4
5	p	0.4
6	n	0.2

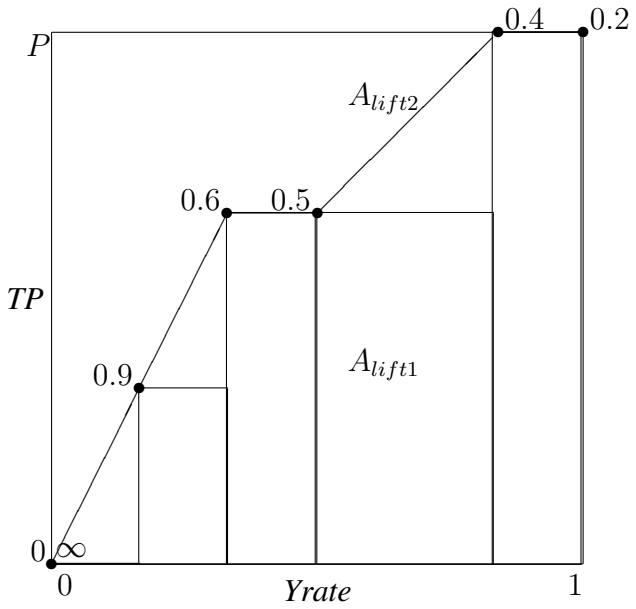

Figure 6: Comparison of methods A_{lift1} and A_{lift2} on data from Table 3.

As we have seen for ROC curves, it is possible to compute A_{lift1} and A_{lift2} directly (without a graph) using formulae 5.2 and 5.5. If the number of all examples is big and the proportion of examples with the same assigned score between all examples is small, then the difference between A_{lift1} and A_{lift2} becomes negligible.

5.2 Optimal classification accuracy

Even though lift chart is not meant for this purpose, we can use a similar approach used for ROC curve to get the point of optimal classification accuracy.

A more interesting problem for lift charts is finding the point of maximal profit which is tightly connected to the weighted classification accuracy. For this purpose we assume profit consists of fixed benefit for every correctly classified example, reduced for a fixed cost for every misclassified example. The point of optimal profit is where the (statistically) expected benefit of the next positive example is equal to its expected cost.

Each point on a lift chart corresponds to a binary classifier, for which we can define classification accuracy and other quality measures. To compute classification accuracy we need to know P and N, but just to find the optimal point (with optimal classification accuracy or profit) the $P : N$ ratio will suffice.

From the equation

$$\mathit{Accuracy} = \frac{\mathit{TP} + \mathit{TN}}{P + N} = \frac{\mathit{TP} + N - \mathit{Yrate}(P + N) + \mathit{TP}}{P + N} \tag{5.6}$$

we get a definition of an iso-performance line of constant classification accuracy.

$$\mathit{TP} = \frac{\mathit{Accuracy}(P + N) - N + \mathit{Yrate}(P + N)}{2} = \frac{P + N}{2}\mathit{Yrate} + \frac{\mathit{Accuracy}(P + N) - N}{2} \tag{5.7}$$

In both described cases (accuracy and profit) the iso-performance lines are straight and parallel, so it is easy to find a tangent to the curve. A graphical example is shown in Figure 7.

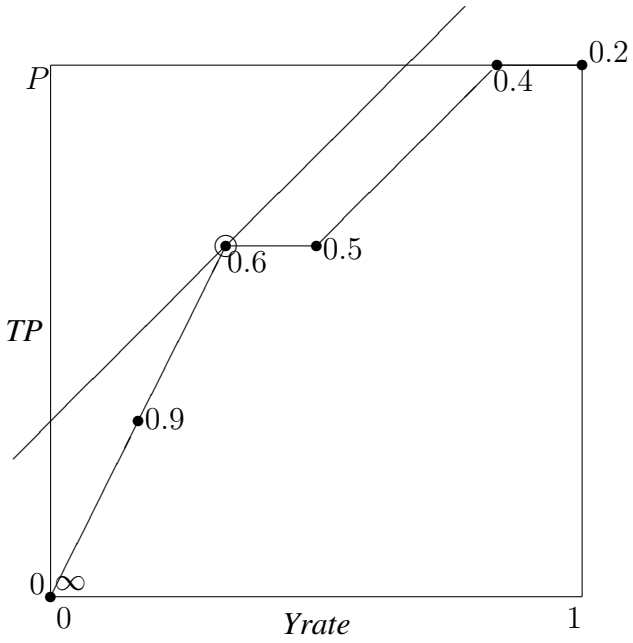

Figure 7: Tangent and point of optimal performance on lift chart, for $P = N$.

Note that other points on the curve can have considerably lower classification accuracy. There is also no direct connection between classification accuracy and area under lift chart.

We can define other iso-performance lines for other measures of classification quality (*e.g.* profit or error cost). Adding weights to positive and negative errors impacts only the angle of these lines.

6 Calibration plot

Calibration plot (Cohen, Goldszmidt, 2004) is quite different from the two previous curves. In Section 3.2 we introduced probabilistic classifiers. Such classifiers assign each example a score (from range $[0, 1]$) or probability that should express the true probability that an example belongs to the positive class. One of the signs that a suitable classification model has been found is also that predicted probabilities (scores) are *well calibrated*, that is that a fraction of about p of events with predicted probability p actually occurs.

Calibration plot is a method that shows us how well the classifier is calibrated and allows us to calibrate it perfectly (Zadrozny, Elkan (2002)). Nevertheless, even after perfect calibration of a classifier, its ROC and lift chart are not affected and its classification ability remains unchanged.

Calibration plot is a graph with a parametric definition

$$x = true\,probability, \quad y = predicted\,probability. \tag{6.1}$$

True probabilities are calculated for (sub)sets of examples with the same score. If there are not enough examples with the same score, examples with similar score are grouped by partitioning the range of possible predictions into subsegments (or bins). In each subsegment the number of positive and negative examples is counted and their ratio defines the true probability. When working on a small test set the points are often spread out, therefore a LOESS [17] method is used in such cases to get a smooth curve. Additionally, true example distribution in the test set is presented by showing positive examples above the graph area (on the x-axis) and negative example below the graph area, in what is called "a rag" (see Figure 8). A good classifier (with good classification accuracy) gathers positive examples near the upper right corner (near 1) and negative examples near lower left corner (near 0).
The $P : N$ ratio influences the true probabilities, so it also influences the plot. Calibration plot only shows the bias of a classifier and has no connection with the classification quality (accuracy). Nonetheless, if a classifier turns out to be very biased, it is probably better to find a different one.

A perfectly calibrated classifier is represented by a diagonal on the graph, which can be quite misleading for the previously mentioned fact that It is possible to calibrate almost every classifier to express the diagonal on the graph without improving its quality of classification. The calibration preserves the "ordering" of examples that the classifier makes by assigning scores (which is tightly connected to the classifier's ability to distinguish between the two classes). If the original classifier assigned the same value to two examples, the same will hold true also after calibration.

$$\text{perfectly calibrated (unbiased) classifier} \neq \text{perfect classifier}$$

Transforming a calibration plot into a ROC, lift chart or calculating accuracy requires knowledge about the distribution density ϱ of the classifier's predictions, and the values of P and N. Let p denote the true probability that an example with a given prediction (score) is positive, o denotes a classifier's prediction and T denotes the classification threshold. It holds true

$$\int_0^1 \varrho(o)\; do = 1. \tag{6.2}$$

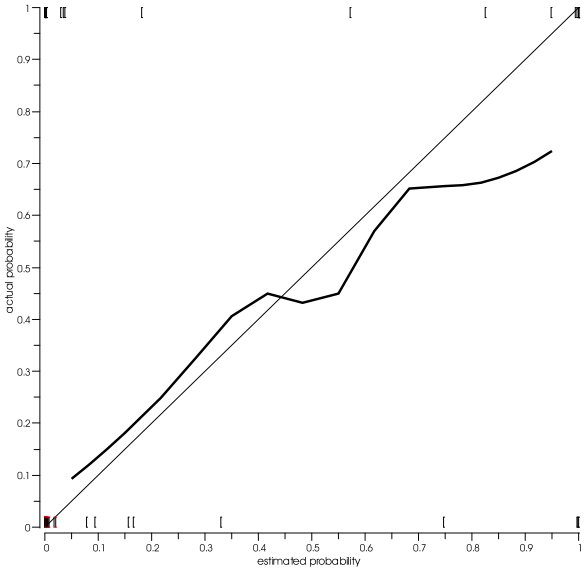

Figure 8: Calibration plot.

For a given threshold T we get the following equations:

$$\frac{TP}{P+N} = \int_T^1 p(o)\varrho(o)\ do, \quad \frac{FP}{P+N} = \int_T^1 (1-p(o))\varrho(o)\ do, \tag{6.3}$$

$$\frac{FN}{P+N} = \int_0^T p(o)\varrho(o)\ do, \quad \frac{TN}{P+N} = \int_0^T (1-p(o))\varrho(o)\ do. \tag{6.4}$$

From these we can derive the ROC and lift curve and all the derived measures.
For example we will derive A_{ROC}:

$$\begin{aligned} A_{ROC} &= \int_0^1 \frac{TP}{P}\ d\frac{FP}{N} && (6.5)\\ &= \frac{(P+N)^2}{P\cdot N}\int_0^1 \frac{TP}{P+N}\ d\frac{FP}{P+N} && (6.6)\\ &= \frac{(P+N)^2}{P\cdot N}\int_0^1 \Big(\int_T^1 p(o)\varrho(o)\ do\Big)\quad (-(1-p(T))\varrho(T))dT && (6.7)\\ &= \frac{(P+N)^2}{P\cdot N}\int_0^1 (p(T)-1)\varrho(T)\int_T^1 p(o)\varrho(o)\ do\ dT && (6.8) \end{aligned}$$

Similar equations can be derived for A_{lift}, classification accuracy and other measures of classification quality. None of these derivations seem to be obviously useful, but it is good to know that they do exist. For example, if we are given a calibration plot and we know the prediction distribution ϱ, we can calculate the classification results (*TP*, *FP*, etc.) In such case the above transformation might be useful, but besides that the primary and the only purpose of the calibration plot is classifier calibration. To the best of our knowledge, we are not aware of any other mathematical meaning and purpose of the calibration plot.

7 Experimental validation and some examples

ROC curve performance analysis is the most appropriate and widely used method for a typical machine learning application setting where choosing a classifier with best classification accuracy for a selected problem domain is of crucial importance. In this section we focus on ROC performance analysis and present an empirical validation of how such analysis can be used to boost classification accuracy. We also show some example graphs of the three curves on selected data sets. The experimental validation was done with the open-source machine learning system Orange (Demar, Zupan, Leban (2004)). We implemented and extended Orange with a set of graphical components (called widgets) for ROC curve and ROC performance analysis, lift chart and calibration plot. We used the implemented widgets to generate graphs and process the data presented in this part of empirical research.

In Section 4.2 we described a method to determine the classifier's score threshold for optimal classification accuracy (for a given $P : N$ rate). To assess the success of this method we used ten-fold cross-validation (10xCV) on fourteen data sets with binary class which are included in Orange and come from various sources (for details about the data sets see Orange web site). We inferred naive Bayesian and decision tree classifiers and measured the increase in classification accuracy after optimal threshold selection. The decision tree classifiers were inferred with the parameter m for pruning set to 2, all other were default parameters set by Orange.

To perform ten-fold cross-validation we randomly divided the data into ten parts (or folds) of equal sizes and with similar class distribution as the entire data set. Nine parts of the data (the learning set) were used to select the optimal threshold and to infer classifier(s). The remaining part (test set) of the data was then used to evaluate the inferred classifier(s) with the selected optimal threshold. We repeated this ten times, each time selecting different nine parts as learning set and the remaining part as test set. Inside each step of the main cross-validation loop an internal ten-fold cross-validation was performed to select the classifiers's optimal threshold on the learning set.

Results on the ten folds from internal cross-validation were merged (for details see Fawcett, 2003; with Provost, 1997) and a single ROC performance analysis was performed to determine the optimal threshold. The $P : N$ rate needed by ROC performance analysis was calculated using class distribution in the entire learning set from the main cross-validation loop. Then, the classifier was inferred on the entire learning set and its threshold (*i.e.* the threshold above which a positive class is predicted) was set to the optimal threshold determined for the current learning set. The classifier was then tested on the main cross-validation test set.

Predictions from the main cross-validation loop were used to calculate the fold-average classification accuracy of classifiers with selected optimal thresholds. To compare the change in classification accuracy after optimal threshold (t_o) selection we also performed a ten-fold cross-validation of classifiers with the default threshold value (*i.e.* threshold 0.5) on same data folds. These two classification accuracies were then compared (see Table 4). Change (Δ) is calculated by subtracting the classification accuracy of a classifier with default threshold (CA_d) from the classification accuracy of that same classifier but with optimal threshold selected (CA_o).

Table 4: Change in classification accuracy (CA) after optimal threshold selection. Classification accuracy for classifiers using default threshold 0.5 is shown in second column (CA_d). Third column (CA_o) shows classification accuracy for classifiers using optimal threshold (t_o) determined by ROC analysis. Fourth column shows the change in classification accuracy ($\Delta = CA_o - CA_d$). Classifier 'bayes' indicates a naive Bayesian classifier and 'tree' indicates a decision tree classifier. Rows are sorted by decreasing change in classification accuracy (Δ).

A_{ROC}	CA_d	CA_o	(Δ)	classifier	t_o	data set
0.741	0.695	0.752	0.056	bayes	0.40	tic_tac_toe
0.999	0.933	0.980	0.047	bayes	0.76	shuttle-landing-control
0.879	0.804	0.844	0.041	bayes	0.87	adult_sample
0.538	0.622	0.657	0.035	bayes	0.43	monks-2
0.960	0.868	0.895	0.027	bayes	0.55	promoters
0.987	0.968	0.984	0.016	tree	0.36	monks-1
0.937	0.889	0.897	0.008	bayes	0.01	ionosphere
0.869	0.862	0.871	0.008	tree	0.20	tic_tac_toe
0.983	0.964	0.971	0.007	bayes	0.76	monks-3
0.973	0.901	0.908	0.007	bayes	0.22	voting
0.906	0.829	0.835	0.006	bayes	0.40	heart_disease
0.676	0.642	0.647	0.005	bayes	0.49	bupa
0.959	0.961	0.966	0.005	tree	0.42	voting
0.923	0.861	0.862	0.001	bayes	0.46	crx
0.726	0.789	0.791	0.001	tree	0.83	titanic
0.715	0.779	0.779	0.000	bayes	0.56	titanic
0.991	0.989	0.989	0.000	tree	0.92	monks-3
0.982	0.980	0.980	0.000	tree	0.45	shuttle-landing-control
0.913	0.930	0.930	0.000	tree	1.00	wdbc
0.912	0.914	0.914	0.000	tree	0.99	ionosphere
0.843	0.846	0.846	0.000	tree	0.78	crx
0.726	0.746	0.746	0.000	bayes	0.53	monks-1
0.983	0.953	0.951	-0.002	bayes	0.93	wdbc
0.655	0.780	0.778	-0.002	tree	0.74	adult_sample
0.653	0.667	0.664	-0.003	tree	0.43	bupa
0.622	0.682	0.679	-0.003	tree	0.41	heart_disease
0.736	0.749	0.736	-0.013	tree	0.61	monks-2
0.818	0.830	0.811	-0.019	tree	0.56	promoters

Classification accuracy increased in 15 cases out of 28 total cases, with maximum increase of 5.6% (*i.e.* 54 more examples correctly classified out of all 958 examples) for the tic_tac_toe data set and naive Bayesian classifier. Only six cases resulted in worse classification accuracy, with maximum decrease of 1.9% (*i.e.* two additional examples misclassified out of all 109 examples by the decision tree classifier on the "promoters" data set). Four of them had low starting A_{ROC} (below 0.75) which is an indication of the learning algorithm inability to correctly model the problem domain. In such cases little can be done to boost classification accuracy but to select a different learning algorithm.

Classification accuracy remained same in seven cases; these are all cases where classifiers achieved high A_{ROC} and classification accuracy, and where further improvements are hard to achieve. Overall the threshold selection method is performing as expected - generally increasing the classification accuracy where possible.

We will now focus on two extreme examples with maximum increase and maximum decrease in classification accuracy and explain the reasons for it. Naive Bayesian classifier has the highest increase on the tic_tac_toe data set. The classifier has a relatively low AUC of only 0.741 and the default classification accuracy 69.5%. Looking at the ROC curve and optimal threshold analysis in Figure 9 one can see that the optimal threshold for predicting positive class "p" is $t_o = 0.403$. The default threshold 0.5 is laying on the ROC curve below the curve's convex hull (point of default threshold 0.5 is shown in Figure 9) which is the reason for lower classification accuracy when using the default threshold.

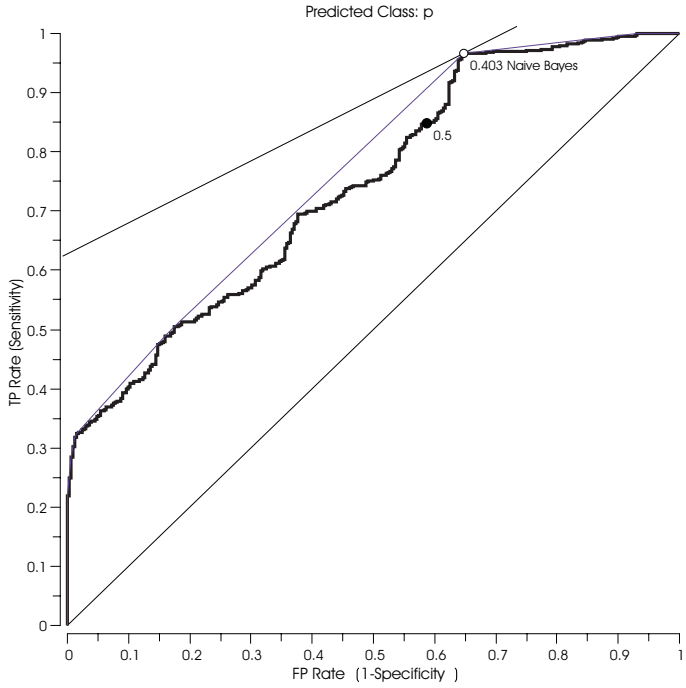

Figure 9: ROC curve and its convex hull for the naive Bayesian classifier on the tic_tac_toe data set. The point with optimal threshold ($t_o = 0.403$) when predicting positive class "p" is where the iso-performance tangent intersects the ROC curve. Slope of the tangent was calculated using the *a priori* class distribution. Point for the classifier with default threhold (0.5) is shown on ROC curve.

A second example is the classification tree classifier used on the promoters data set. Comparing the calibration plots in Figure 10 of both classifiers used (naive Bayes and classification tree) on the promoters data set one can see that the tree returns extremely uncalibrated probabilities. This is a clear sign of the difficulty for the tree learning method to deal with the promoters problem domain. Bad calibration is also partly the result of the tree's tendency to return a relatively small number of different probabilistic scores when classifying examples (now shown). We observed this in the classification plot rug for the tree classifier and it can be also observed in the shape of the ROC curve shown in Figure 11. The ROC curve of the Bayesian classifier has many more "steps" than the

tree's curve, which is a direct indication of a more diverse set of scores it can return on the given data set. The main reason for the decrease in classification accuracy is the tree's model instability. Looking at Table 4 we can see that the average optimal threshold is $t_o = 0.56$ which could be an indication of good calibration, what we saw is not the case here. Standard deviation of the optimal threshold across the ten folds is 0.316 (not shown for the others) which only confirms the tree's instability and explains why small changes in the threshold could have great consequences on the final classification accuracy.

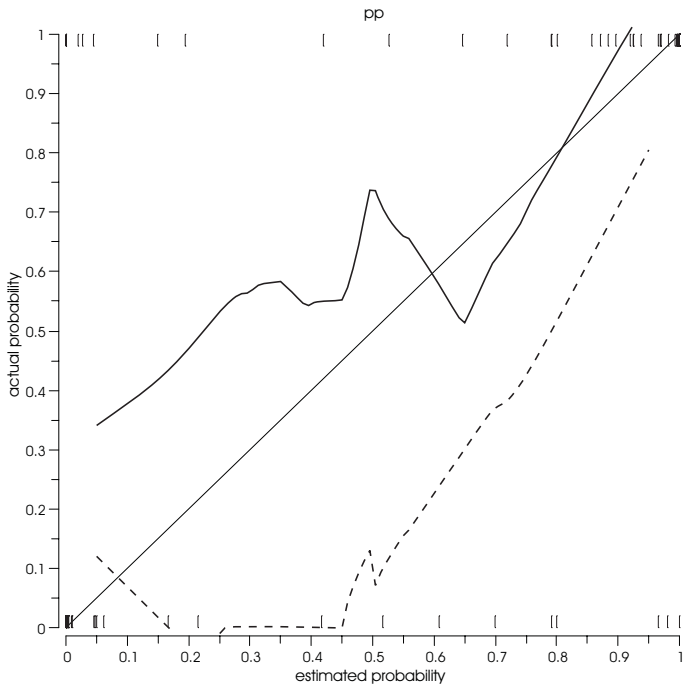

Figure 10: Calibration plots for naive Bayesian classifier (solid line), and decision tree classifier (dashed line) when predicting positive class "pp."

8 Conclusion

Three different graphical techniques (ROC curve, lift chart and calibration plot) used to evaluate the quality of classification models were presented. Basic facts about each technique are well known from literature, but this paper presents a common framework, stresses and derives the similarities, differences and interrelations between them. Their mathematical meaning and interpretation was given (more precisely than in related papers), with special focus on different measures of classification quality that are closely related to these three techniques. The relations and possible transformations between the curves and between some derived measures were also presented. These extensions are, to the best of our knowledge, the main novelty and contribution of this paper.

9 Acknowledgement

The authors would like to thank prof. dr. Blaž Zupan for the initial idea and help. The work done by T.C. was supported by Program and Project grants from the Slovenian Research Agency.

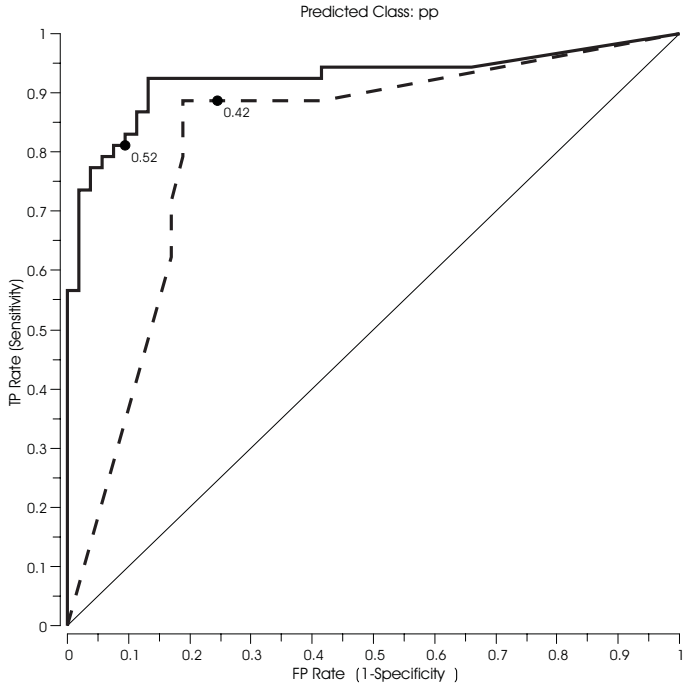

Figure 11: ROC curve for naive Bayesian classifier (solid line), and decision tree classifier (dashed line) when predicting positive class "pp." Points on ROC curve closest to default threshold (0.5) are shown for the two classifiers.

References

[1] Berry, M.J.A. and Linoff, G. (1999): *Data Mining Techniques: For Marketing, Sales, and Customer Support*. Morgan Kaufmann Publishers.

[2] Centor, R.M. (1991): Signal detectability: The use of ROC curves and their analyses. *Medical Decision Making*.

[3] Cohen, I. and Goldszmidt, M. (2004): Properties and benefits of calibrated classifiers, In *Proceedings of ECML 2004*.
http://www.ifp.uiuc.edu/ iracohen/publications/CalibrationECML2004.pdf.

[4] Demšar, J., Zupan, B. and Leban, G. (2004): Orange: From Experimental Machine Learning to Interactive Data Mining, *White Paper (www.ailab.si/orange)*, Faculty of Computer and Information Science, University of Ljubljana.

[5] Egan, J.P. (1975): *Signal Detection Theory and ROC Analysis*, New York: Academic Press.

[6] Fawcett, T. (2003): ROC Graphs: Notes and Practical Considerations for Data Mining Researchers. HP Laboratories.
http://home.comcast.net/˜tom.fawcett/public_html/papers/ROC101.pdf.

[7] Fayyad, U.M. and Irani, K.B. (1993): Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of 13th International Joint Conference on Artificial Intelligence (IJCAI-93)*, 1022–1027.

[8] Hanley, J.A. and McNeil, B.J. (1982): The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Diagnostic Radiology*, **143**, 29–36.

[9] Pyle, D. (1999): *Data Preparation for Data Mining*. Morgan Kaufmann Publishers.

[10] Provost, F. and Fawcett, T. (1997): *Analysis and Visualization of Classifier Performance: Comparison under Imprecise Class and Cost Distributions*. *KDD-97*.

[11] Provost, F., Fawcett, T. (2001): Robust classification for imprecise environments. *Machine Learning*, **42**, 203–231.

[12] Zadrozny, B., Elkan, C. (2002), Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth International Conference on Knowledge Discovery and Data Mining (KDD'02)*. http://www-cse.ucsd.edu/ zadrozny/kdd2002-Transf.pdf.

[13] Witten, I. H., Frank, E. (2000): *Data Mining, Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers.

[14] Cumulative gains and lift charts. http://www2.cs.uregina.ca/˜dbd/cs831/notes/lift_chart/lift_chart.html.

[15] Data modeling and mining: Why lift? http://www.dmreview.com/article_sub.cfm?articleId=5329.

[16] Lift chart, profit chart, confusion matrix. http://www.kdkeys.net/forums/51/ShowForum.aspx.

[17] Loess curve: http://en.wikipedia.org/wiki/Loess_curve.

[18] Lowess and Loess: Local regression smoothing. http://www.mathworks.com/access/helpdesk/help/toolbox/curvefit/ch_data7.html.

[19] Receiver operating characteristic. http://en.wikipedia.org/wiki/Receiver_operator_characteristic.

[20] Receiver operating characteristic curves. http://www.anaesthetist.com/mnm/stats/roc.

[21] Receiver operating characteristics (ROC). http://www.cs.ucl.ac.uk/staff/W.Langdon/roc.

[22] Receiver operating characteristic (ROC) literature research. http://splweb.bwh.harvard.edu:8000/pages/ppl/zou/roc.html.