Volume Under the ROC Surface for Multi-class Problems

C. Ferri J. Hernández-Orallo M.A. Salido

Dep. Sistemes Informàtics i Computació, Univ. Politècnica de València (Spain)
{cferri, jorallo, msalido}@dsic.upv.es

Abstract. Receiver Operating Characteristic (ROC) analysis has been successfully applied to classifier problems with two classes. The Area Under the ROC Curve (AUC) has been elected as a better way to evaluate classifiers than predictive accuracy or error and has also recently used for evaluating probability estimators. However, the extension of the Area Under the ROC Curve for more than two classes has not been addressed to date, because of the complexity and elusiveness of its precise definition. Some approximations to the real AUC are used without an exact appraisal of their quality. In this paper, we present the real extension to the Area Under the ROC Curve in the form of the Volume Under the ROC Surface (VUS), showing how to compute the polytope that corresponds to the absence of classifiers (given only by the trivial classifiers), to the best classifier and to whatever set of classifiers. We compare the real VUS with "approximations" or "extensions" of the AUC for more than two classes.

1 Introduction

In general, classifiers are used to make predictions for decision support. Since predictions can be wrong, it is important to know what the effect is when the predictions are incorrect. In many situations not every error has the same consequences. Some errors have greater cost than others, especially in diagnosis. For instance, a wrong diagnosis or treatment can have different cost and dangers depending on which kind of mistake has been done. In fact, it is usually the case that misclassifications of minority classes into majority classes (e.g. predicting that a system is safe when it is not) have greater costs than misclassifications of majority classes into minority classes (e.g. predicting that a system is not safe when it actually is). Obviously, the costs of each misclassification are problem dependent, but it is almost never the case that they would be uniform for a single problem. Consequently, accuracy is not generally the best way to evaluate the quality of a classifier or a learning algorithm.

Cost-sensitive learning [14] is a more realistic generalisation of predictive learning, and cost-sensitive models allow for a better and wiser decision making. The quality of a model is measured in terms of cost minimisation rather than error minimisation. When cost matrices are provided a priori, i.e. before learning takes place, the matrices have to be fully exploited to obtain models that minimise cost.

However, in many circumstances, costs are not known a priori or the models are just there to be evaluated or chosen. Receiver Operating Characteristic (ROC) analysis [5][9][13] has been proven to be very useful for evaluating given classifiers in these cases, when the cost matrix was not known at the moment the classifiers were constructed. ROC analysis provides tools to select a set of classifiers that will behave optimally and reject some other useless classifiers. In order to do this, the convex hull of all the classifiers is constructed, giving a "curve" (a convex polygon).

In the simplest case, a single 2-class classifier forms a 4-segment ROC curve (a polygon in a strict sense) with the point given by the classifier, two trivial classifiers (the classifier that always predicts class 0 and the classifier that always predicts class 1) and the origin, whose area can be computed. This area is called the Area Under the ROC Curve (AUC) and has become a better alternative than accuracy (or error), for evaluating classifiers. AUC is also used for probabilistic estimators, where these estimations are used where ranking prediction is important [10].

ROC analysis and the AUC measure have been extensively used in the area of medical decision making [7][15], in the field of knowledge discovery, data mining, pattern recognition [1] and science in general [13]. However, the applicability of ROC analysis and the AUC has only been shown for problems with two classes. Although ROC analysis can be extended in theory for multi-dimensional problems [12], practical issues (computational complexity and representational comprehensibility, especially) preclude its use in practice. The major hindrance is the high dimensionality. A confusion matrix obtained from a problem of c classes has c^2 positions, and $(c \cdot (c\text{-}1))$ dimensions (d), i.e. all possible misclassification combinations are needed.

Nonetheless, although difficult, it is possible to perform ROC analysis for more than two classes and to compute the AUC (more precisely, the Volume Under the ROC Surface, VUS). However, the trivial classifiers for more than two classes, the minimum and maximum volume have not been identified to date in the literature.

In this paper, we present the trivial classifiers, the equations, the maximum and minimum VUS, for classifiers of more than 2 classes. We use this to compute the real VUS for any classifier by the use of a Hyperpolyhedron Search Algorithm (HSA) [11]. We then compare experimentally the real VUS with several other (and new) approximations, showing which approximation is best.

2 ROC Analysis

The Receiver Operating Characteristic (ROC) analysis [5][9][13] allows the evaluation of classifier performance in a more independent and complete way than just using accuracy. ROC analysis has usually been presented for two classes, because it is easy to define, to interpret and it is computationally feasible.

ROC analysis for two classes is based on plotting the true-positive rate (TPR) on the y-axis and the false-positive rate (FPR) on the x-axis, giving a point for each classifier. A "curve" is obtained because we can obtain infinitely many derived classifiers along the segment that connects two classifiers just by voting them with different weights. Hence, any point *below* that segment will have greater cost for any class distribution and cost matrix, because it has lower TPR and/or higher FPR. According to this, given several classifiers, one can discard the classifiers that fall under the convex hull formed by the points representing the classifiers and the points (0,0) and (1,1), which represent the default classifiers always predicting negative and positive, respectively. A detailed description of ROC analysis can be found in [5][9].

The usual way to represent the ROC space is not, in our opinion, a very coherent way, since the true class is represented incrementally for correct predictions and the false class is represented incrementally for incorrect predictions. Moreover this choice is not easily extensible for more than two classes. Instead, we propose to represent the false-negative-rate (FNR) and the FPR. Now, the points (0,1) and (1,0) represent, respectively, the classifier that classifies anything as negative and the classifier that classifies anything as positive. The curve is now computed with points (0,1), (1,0) and (1,1).

Obviously, with this new diagram, instead of looking for the maximisation of the Area Under the ROC Curve (AUC) we have to look for its minimisation. A better option is to compute the Area Above the ROC Curve (AAC). In order to maintain accordance with classical terminology, we will refer to the AAC also as AUC.

3 Multi-class ROC Analysis

Srinivasan has shown in [12] that, theoretically, the ROC analysis extends to more than two classes "directly". For c classes, and assuming a normalised cost matrix, we have to construct a vector of $d = c(c\text{-}1)$ dimensions for each classifier. In general the cost of a classifier for c classes is:

$$Cost = \sum_{i,j,i\neq j} p(i)\cdot C(i,j)R(i,j)$$

where R is the confusion ratio matrix (each column normalised to sum 1), C is the cost matrix, and $p(i)$ is the absolute frequency of class i. From the previous formula, two classifiers 1 and 2 will have the same cost when they are on the same iso-performance hyperplane. However, the d-1 values of the hyperplane are not so straightforward and easy to obtain and understand as the slope value of the bi-dimensional case.

In the same way as the bi-dimensional case, the convex hull can be constructed, forming a polytope. To know whether a classifier can be rejected, it has to be seen whether the intersection of the current polytope with the polytope of the new classifier gives the new polytope, i.e., the new polytope is included in the first polytope [8].

Provided this direct theoretical extension, there are some problems.

- In two dimensions, doubling the probability of one class has a direct counterpart in costs. This is not so for $d > 2$, because there are many more degrees of freedom.
- The best algorithm for the convex hull of N points is $O(N \log N + N^{d/2})$ [8][3].
- In the 2-d case, it is relatively straightforward how to detect the trivial classifiers and the points for the minimum and maximum cases.

However, not only there are computational limitations but also representational ones. ROC analysis in two dimensions has a very nice and understandable representation, but it cannot be directly extended to more than two classes, because even for 3 classes we have a 6D space, quite difficult to be represented. In what follows, we illustrate the extension for 3 classes, although the expressions can be generalised easily.

3.1 Extending ROC Analysis for 3 classes

In this part we consider the extension of ROC analysis for 3-class problems. In this context we consider the following cost ratio matrix for three-class classifiers:

			Actual	
		a	*b*	*c*
Predicted	*a*	h_a	x_1	x_2
	b	x_3	h_b	x_4
	c	x_5	x_6	h_c

This gives a 6-dimensional point (x_1, x_2, x_3, x_4, x_5, x_6). The values h_a, h_b and h_c are dependent and do not need to be represented, because:

$$h_a + x_3 + x_5 = 1, \quad h_b + x_1 + x_6 = 1, \quad h_c + x_2 + x_4 = 1$$

3.1.1 Maximum VUS for 3 classes

Let us begin by considering the maximum volume. The maximum volume should represent the volume containing all the possible classifiers. A point in the 1-long hypercube is a classifier if and only if:

$$x_3 + x_5 \leq 1, \quad x_1 + x_6 \leq 1, x_2 + x_4 \leq 1$$

It is easy to obtain the volume of the space determined by these equations, just by using the probability that 6 random numbers under a uniform distribution U(0,1) would follow the previous conditions. More precisely:

$$\mathrm{VUS}_3^{\max} = \mathrm{P}(\mathrm{U}(0,1) + \mathrm{U}(0,1) \leq 1) \cdot \mathrm{P}(\mathrm{U}(0,1) + \mathrm{U}(0,1) \leq 1) \cdot \mathrm{P}(\mathrm{U}(0,1) + \mathrm{U}(0,1) \leq 1)$$
$$= [\mathrm{P}(\mathrm{U}(0,1) + \mathrm{U}(0,1) \leq 1)]^3.$$

It is easy to see that the probability that the sum of two random numbers under the distribution U(0,1) is less than 1 is exactly ½ , i.e:

$$\mathrm{P}(\mathrm{U}(0,1) + \mathrm{U}(0,1) \leq 1) = \tfrac{1}{2}, \text{ consequently } \mathrm{VUS}_3^{\max} = (\tfrac{1}{2})^3 = 1/8$$

We have also considered the maximum VUS for *c* classes. It is easy to see that the volume of the space determined by valid equations for *c* classes is:

$$\mathrm{VUS}^{\max} = \textstyle\prod_c [\ \mathrm{P}(\Sigma_{c\text{-}1}\, \mathrm{U}(0,1) \leq 1)\ ] = [\mathrm{P}(\Sigma_{c\text{-}1}\, \mathrm{U}(0,1) \leq 1)]^{c}\ .$$

However, the probability that the sum of $c-1$ random numbers under the distribution U(0,1) is less than 1 is difficult to be obtained. In particular, the probability density function of the sum of *n* uniform variables on the interval [0,1] can be obtained using the characteristic function of the uniform distribution.

$$d_n(x) = F^{-1}\left[\left(\frac{i-\cos t+\sin t}{t}\right)^n\right] = \frac{1}{2(n-1)!}\sum_{k=0}^{n}(-1)^k\binom{n}{k}(x-k)^{n-1}\operatorname{sgn}(x-k)$$

Using the cumulative distribution function $D_n(x)$, we have that the probability that the sum of *n* random numbers with U(0,1) is less than 1 is:

$$D_n(1) = \lim_{x\to 1}\int_{-\infty}^{x} d_n(x)dx = \frac{1}{2(n-1)!}\int_{-\infty}^{1^-}\sum_{k=0}^{n}(-1)^k\binom{n}{k}(x-k)^{n-1}\operatorname{sgn}(x-k)dx$$

For n=1 we have $D_1(1)$= 1, for n=2 we have $D_2(1)$= ½, for n=3, $D_3(1)$= 1/6 and:

$$\mathrm{VUS}_c^{\max} = (D_{c-1}(1))^c$$

And hence we have, $\mathrm{VUS}_2^{\max}$= 1, $\mathrm{VUS}_3^{\max}$= 1/8 and $\mathrm{VUS}_4^{\max}$= 1/1296. However, for n>3, D_n is complex. For such cases, we can approximate the sum of *n* random numbers under the distribution U(0,1) with a single variable (*Y*) under the normal distribution with μ=n/2 and σ= n/12 using the central limit theorem. Then:

$$P(Y \le 1) \approx P\left(\frac{Y - \frac{n}{2}}{\sqrt{\frac{n}{12}}} \le \frac{1 - \frac{n}{2}}{\sqrt{\frac{n}{12}}} \right) = P\left(Z \le \frac{1 - \frac{n}{2}}{\sqrt{\frac{n}{12}}} \right) = \Phi\left(\frac{1 - \frac{n}{2}}{\sqrt{\frac{n}{12}}} \right)$$

Where Z is a standard normal distribution variable. Therefore, when c>3:

$$\mathrm{VUS}_{\mathrm{c}}^{\max} \approx \left[\Phi\left(\frac{1 - \frac{c-1}{2}}{\sqrt{\frac{c-1}{12}}} \right) \right]^{C}$$

3.1.2 Minimum VUS for 3 classes

Now let us try to derive the minimum VUS. Without any knowledge we can construct trivial classifiers by giving more or less probability to each class, as follows:

		Actual		
		a	*B*	*c*
	a	h_a	h_a	h_a
Predicted	*b*	h_b	h_b	h_b
	c	h_c	h_c	h_c

where $h_a + h_b + h_c = 1$. These obviously include the three extreme trivial classifiers “everything is a”, “everything is b” and “everything is c”. Given a classifier:

		Actual		
		A	*B*	*c*
	a	v_{aa}	v_{ba}	v_{ca}
Predicted	*b*	v_{ab}	v_{bb}	v_{cb}
	c	v_{ac}	v_{bc}	v_{cc}

we can discard this classifier if and only if it is above a trivial classifier, formally:

$$\exists h_a, h_b, h_c \in \mathrm{R}^{+} \text{ where } (h_a + h_b + h_c = 1) \text{ such that:}$$

$$v_{ba} \ge h_a, v_{ca} \ge h_a, v_{ab} \ge h_b, v_{cb} \ge h_b, v_{ac} \ge h_c, v_{bc} \ge h_c$$

From here, we can derive the following theorem (see [4] for the proof):

Theorem 1: Without any knowledge, a classifier $(x_1, x_2, x_3, x_4, x_5, x_6)$ can be discarded iff: $r_1 + r_2 + r_3 \ge 1$ where $r_1 = \min(x_1, x_2)$, $r_2 = \min(x_3, x_4)$ and $r_3 = \min(x_5, x_6)$.

Given the previous property, we only have to compute the space of classifiers that follow the condition that $r_1 + r_2 + r_3 \ge 1$ where $r_1 = \min(x_1, x_2)$, $r_2 = \min(x_3, x_4)$ and $r_3 = \min(x_5, x_6)$ to obtain the minimum volume corresponding to total absence of information. More precisely, we have to compute the volume formed by this condition jointly with the valid classifier conditions, i.e.:

$$x_3 + x_5 \le 1,\ x_1 + x_6 \le 1,\ x_2 + x_4 \le 1 \text{ and } r_1 + r_2 + r_3 \ge 1$$

$$\text{where } r_1 = \min(x_1, x_2),\ r_2 = \min(x_3, x_4) \text{ and } r_3 = \min(x_5, x_6)$$

This volume is more difficult to be obtained by a probability estimation, due to the min function and especially because the first conditions and the last one are dependent. Let us compute this volume using a Monte Carlo method.

3.1.3 Monte Carlo Method for obtaining Max and Min VUS

Monte Carlo methods are used to randomly generate a subset of cases from a problem space and estimate the probability that a random case follows a set of conditions. These methods are particularly interesting to approximate volumes, such as the volume under the ROC curve we are dealing with.

For this purpose, we generate an increasing number of points in the 6D hypercube of length 1 (i.e., we generate six variables x_1, x_2, x_3, x_4, x_5, x_6 using a uniform distribution between 0 and 1) and then check whether or not they follow the previous maximum or minimum conditions. Since we are working with a 1-length hypercube, the proportion of cases following the conditions is exactly the volume we are looking for.

In particular, we have obtained the following results:

- Maximum: 0.12483 for 1,000,000 cases, matching our theoretical $\mathrm{VUS}_3^{\max} = 1/8$.
- Minimum: 0.00555523 for 1,000,000 cases, which is approximately 1/180.

However, although we have obtained the exact maximum, we have not obtained the exact minimum (although 1/180 is conjectured). In the next section we introduce a method to compute the real $\mathrm{VUS}^{\min}$, and, more importantly, to obtain the ROC polytopes that form these volumes.

4 A Constraint Satisfaction Algorithm for the ROC Polytopes

In the previous section we have developed the conditions for the maximum and minimum VUS, given, respectively, when the best classifier is known (0, 0, 0, 0, 0, 0) and when no classifier is given (absence of information). However, we are interested in a way to obtain the border points of each space, i.e., the polytopes that represent both cases. What we need is a way to compute these polytopes given the set of conditions. A general system able to do this is HSA.

4.1 Hyperpolyhedron Search Algorithm (HSA)

In the constraint satisfaction literature, researchers have focussed on discrete and binary Constraint Satisfaction Problems (CSPs). However, many real problems (as the ROC surface problem) can be naturally modelled using non-binary constraints over continuous variables. Hyperpolyhedron Search Algorithm (HSA) [11] is a CSP solver that manages non-binary and continuous problems. HSA carries out the search through a hyperpolyhedron that maintains in its vertices those solutions that satisfy all non-binary constraints. The handling of the non-binary constraints (linear inequations) can be seen as the handling of global hyperpolyhedron constraints. Initially, the hyperpolyhedron is created by the Cartesian product of the variable domain bounds. For each constraint, HSA checks the consistency, updating the hyperpolyhedron through linear programming techniques. Each constraint is a hyperplane that is intersected to obtain the new hyperpolyhedron vertices. The resulting hyperpolyhedron is a convex set of solutions to the CSP. A solution is an assignment of a value from its domain to every variable where all constraints are satisfied. HSA can determine: whether a solution exists (consistency), several solutions or the extreme solutions.

In the ROC surface problem, we will use HSA to determine the extreme solutions in order to calculate the convex hull of the resulting hyperpolyhedron. HSA does not compute the volume of the hyperpolyhedron. For this purpose, we are using QHull [2]. QHull is, among other things, an algorithm that implements a quick method for computing the convex hull of a set of points and the volume of the hull.

4.2 Maximum VUS points for 3 classes

Let us recover the equations for the maximum volume (valid classifier conditions):

$$x_3 + x_5 \leq 1, x_1 + x_6 \leq 1, x_2 + x_4 \leq 1$$

We introduce these equations to HSA and look for solutions for these six variables. We obtain 41 points (which can be simplified into just 27 points, see [4]) whose volume is, as expected, 0.125 (1/8).

4.3 Minimum VUS for 3 classes

From Theorem 1, in order to compute the minimum VUS, we only have to compute the space of classifiers following that $r_1 + r_2 + r_3 \geq 1$ where $r_1 = \min(x_1, x_2)$, $r_2 = \min(x_3, x_4)$ and $r_3 = \min(x_5, x_6)$ to obtain the minimum volume corresponding to total absence of information. Using this condition and the hyper-cube conditions, we have:

$$x_3 + x_5 \leq 1,\ x_1 + x_6 \leq 1, x_2 + x_4 \leq 1,\ r_1 + r_2 + r_3 \geq 1$$

where $r_1 = \min(x_1, x_2)$, $r_2 = \min(x_3, x_4)$ and $r_3 = \min(x_5, x_6)$. Since the min function is not handled by HSA, we convert the last condition into eight equivalent inequations:

$$x_1 + x_3 + x_5 \geq 1, x_1 + x_3 + x_6 \geq 1, x_1 + x_4 + x_5 \geq 1, x_1 + x_4 + x_6 \geq 1,$$
$$x_2 + x_3 + x_5 \geq 1, x_2 + x_3 + x_6 \geq 1, x_2 + x_4 + x_5 \geq 1, x_2 + x_4 + x_6 \geq 1$$

and now we obtain a set of 25 points whose volume is 0.0055, which is approximately 1/180 and matches the volume obtained by the Monte Carlo method.

Some of these points are exactly on the surface of the volume and can be removed without modifying the volume in a simplified set of 9 points (see [4]).

4.4 Computing the VUS of Any Classifier

Now it seems that we can obtain the VUS of any classifier just be adding the coordinates of the point it represents and adding them as a new point to the minimum and then computing the convex hull. However, this would be a hasty step. The surprise would come up if we take the minimum (9 points, 1/180) and add the origin (the best classifier with no error at all). In this case, we obtain 10 points and 1/120 volume, which is a greater volume but it is not the maximum. This seems contradictory, because if we have the best classifier, we should obtain the maximum volume. The reason is the following. When we have the perfect classifier, represented by the point (0, 0, 0, 0, 0, 0), any classifier that has a value equal or greater than 0 in any coordinate is discardable and, logically, this should give 1/8, not 1/120. The issue is that whenever we add a new classifier we have to consider the conditions it produces, which are polytopes, not just points.

In other words, the perfect classifier generates the following discard equations:

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0, x_6 \geq 0$$

These inequations are null, because all the values should be positive, and, hence, we only have the valid classifier conditions, and then we have the maximum volume 1/8.

Now let us consider the same thing for any arbitrary classifier C1:

		Actual		
		a	*b*	*c*
	a	z_{aa}	z_{ba}	z_{ca}
Predicted	*b*	z_{ab}	z_{bb}	z_{cb}
	c	z_{ac}	z_{bc}	z_{cc}

What can be discarded? The answer is that any classifier such that is worse than the classifier C1 (combined with the trivial classifiers), i.e., any classifier that would have greater values for the 6 dimensions. Consequently, given a new classifier C2:

		Actual		
		a	*b*	*c*
	a	v_{aa}	v_{ba}	v_{ca}
Predicted	*b*	v_{ab}	v_{bb}	v_{cb}
	c	v_{ac}	v_{bc}	v_{cc}

We have to look at all the classifiers constructed as a linear combination of the three trivial classifiers and the classifier C1, and see whether C2 is worse than any of the constructed classifiers. Formally, the linear combination is defined as:

$h_a \cdot (1, 1, 0, 0, 0, 0) + h_b \cdot (0, 0, 1, 1, 0, 0) + h_c \cdot (0, 0, 0, 0, 1, 1) + h_d \cdot (z_{ba}, z_{ca}, z_{ab}, z_{cb}, z_{ac}, z_{bc})$

And we can discard C2 when

$$\exists h_a, h_b, h_c, h_d \in \mathrm{R}^+ \text{ where } (h_a + h_b + h_c + h_d = 1) \text{ such that:}$$
$$v_{ba} \geq h_a + 0 + 0 + h_d \cdot z_{ba,}\ v_{ca} \geq h_a + 0 + 0 + h_d \cdot z_{ca,}\ v_{ab} \geq 0 + h_b + 0 + h_d \cdot z_{ab,}$$
$$v_{cb} \geq 0 + h_b + 0 + h_d \cdot z_{cb,}\ v_{ac} \geq 0 + 0 + h_c + h_d \cdot z_{ac,}\ v_{bc} \geq 0 + 0 + h_c\ + h_d \cdot z_{bc}$$

This gives a system of inequations with 10 variables (z_{ij} are constants given by C1), that can be input to HSA, and then we obtain the edge six dimensions points from v_{ij}.

4.5 Real VUS for More than One Classifier

In the same way as before, given a set of classifiers, we can compute the true VUS of the set, just generalising the previous formula. Let us illustrate it for 4 classifiers *Z, W, Y* and *X*. In fact, what we have to do is to consider the linear combination of the three trivial classifiers with the four given classifiers, i.e.:

$$h_a \cdot (1, 1, 0, 0, 0, 0) + h_b \cdot (0, 0, 1, 1, 0, 0) + h_c \cdot (0, 0, 0, 0, 1, 1) + h_1 \cdot (z_{ba}, z_{ca}, z_{ab}, z_{cb}, z_{ac}, z_{bc})$$
$$+ h_2 \cdot (w_{ba}, w_{ca}, w_{ab}, w_{cb}, w_{ac}, w_{bc}) + h_3 \cdot (x_{ba}, x_{ca}, x_{ab}, x_{cb}, x_{ac}, x_{bc}) + h_4 \cdot (y_{ba}, y_{ca}, y_{ab}, y_{cb}, y_{ac}, y_{bc})$$

And now we can discard when:

$$\exists h_a, h_b, h_c, h_d \in \mathrm{R}^+ \text{ where } (h_a + h_b + h_c + h_1 + h_2 + h_3 + h_4 = 1) \text{ such that:}$$
$$v_{ba} \geq h_a + 0 + 0 + h_1 \cdot z_{ba} + \mathrm{h}_2 \cdot w_{ba} + h_3 \cdot x_{ba}\ + h_4 \cdot y_{ba,}$$
$$v_{ca} \geq h_a + 0 + 0 + h_1 \cdot z_{ca} + h_2 \cdot w_{ca} + h_3 \cdot x_{ca} + h_4 \cdot y_{ca},$$
$$v_{ab} \geq 0 + h_b + 0 + h_1 \cdot z_{ab} + h_2 \cdot w_{ab} + h_3 \cdot x_{ab} + h_4 \cdot y_{ab},$$
$$v_{cb} \geq 0 + h_b + 0 + h_1 \cdot z_{cb} + h_2 \cdot w_{cb} + h_3 \cdot x_{cb} + h_4 \cdot y_{cb},$$
$$v_{ac} \geq 0 + 0 + h_c + h_1 \cdot z_{ac} + h_2 \cdot w_{ac} + h_3 \cdot x_{ac} + h_4 \cdot y_{ac},$$
$$v_{bc} \geq 0 + 0 + h_c\ + h_1 \cdot z_{bc} + h_2 \cdot w_{bc} + h_3 \cdot x_{bc} + h_4 \cdot y_{bc}$$

This gives a system with 9+4 variables that can be solved by HSA, from which we again retain just 6 (v_{ij}) variables to obtain the polytope.

5 Evaluation of Multi-class Approximations to the VUS

In the previous section we have developed a method (conditions + HSA) to obtain the real VUS of any classifier for an arbitrary number of classes (the extension for more than 3 classes is trivial). However, this exact computation, although quite efficient for 3 and 4 classes, must be impractical for a higher number of classes or classifiers.

In the literature, there have been several approximations for the extension of the AUC measure for multi-class problems, either based on the interpretation of the AUC as distribution separability [6] or the meaning of the equivalent (for two classes) Wilcoxon statistic or GINI coefficient. However, there is no appraisal or estimation, either theoretical or practical, of how good they are.

In this section we gather and remind the approximations for the AUC for more than two classes known to date: macro-average, 1-point trivial AUC extension and some Hand & Till [6] variants. We are going to make a comparison with the real measure we have presented in this work: the exact VUS (through the HSA method).

We will give the definitions for three classes, although this can be easily extended to more classes. For the following definitions consider a classifier C2 as before.

5.1 Macro-average

The macro-average is just defined as the average of the class accuracies, i.e.:

$$\text{MAVG}_3 = (v_{aa} + v_{bb} + v_{cc}) / 3$$

This measure has been used as a very simple way to handle more appropriately unbalanced datasets (without using ROC analysis).

5.2 Macro-average Modified

We modify the original definition of macro-average because this does not consider the standard deviation between the points. For instance, using two classes, the point (0.2, 0.2) has greater AUC than the point (0.1, 0.3) although both points have identical macro-average. Thereby, we will employ the generalised mean instead of average:

$$\text{MAVG}_3\text{-MOD} = \left(\frac{1}{n}\sum_{k=1}^{n} a_k^t\right)^{1/t}$$

The best value for t between the arithmetic mean (t=1) and the geometric mean ($t \rightarrow 0$) has been estimated experimentally. The value t=0.76 obtains the best performance.

5.4 1-point Trivial AUC Extension

Going back to two classes, the area for one point (v_{ba}, v_{ab}) (in our representation) is:

$$\text{AUC}_2 = \max(1/2, 1 - v_{ba}/2 - v_{ab}/2)$$

Extending trivially the previous formula, we have this extension for 1-point:

$$\text{AUC-1PT}_3 = \max(1/3, 1 - (v_{ba} + v_{ca} + v_{ab} + v_{cb} + v_{ac} + v_{bc}) / 3$$

This extension turns to be equal to the macro-average since the columns of the matrix sum to 1. The only difference is that the 1PT_3 measure is never lower than 1/3.

5.5 1-point Hand and Till Extension

Hand and Till have presented a generalisation of the AUC measure [6] for soft classifiers, i.e., classifiers that assign a different score, reliability or probability with each prediction. Although we will deal with soft classifiers later, let us adapt Hand and Till's formulation for crisp classifiers, i.e., classifiers that predict one of the possible classes, without giving any additional information about the reliability or probability of the predicted class, or the other classes.

Hand and Till's extension for more than two classes is based on the idea that if we can compute the AUC for two classes *i,j* (let us denote this by *A*(*i,j*)), then we can compute an extension of AUC for any arbitrary number of classes by choosing all the possible pairs (1 vs. 1). Since *A(i,j) = A(j,i)*, this can be simplified as shown in the following Hand and Till's *M* function:

$$M = \frac{1}{c(c-1)}\sum_{i\neq j}\hat{A}(i,j) = \frac{2}{c(c-1)}\sum_{i<j}\hat{A}(i,j)$$

Pursuing this idea we are going to introduce three variants. The first variant is given if we consider the macro-average extension. Then we have:

$$\mathrm{HT1a} = (\max(1/2, (v_{aa} + v_{bb})/2) + \max\ (1/2, (v_{aa} + v_{cc})/2) + \max(1/2, (v_{bb} + v_{cc})/2)\ )\ /\ 3$$

This is equal to the 1PT. But if we take failures into account instead of hits, we have:

$$\mathrm{HT1b}= (\max(1/2\ ,\ 1–(v_{ba}+v_{ab})/2) + \max(1/2,\ 1–(v_{ca}+v_{ac})\ /2) + \max(1/2,\ 1–(v_{cb}+v_{bc})\ /2))\ /\ 3$$

This measure is slightly different from the previous ones and we will use this one. Another different way is normalisation, e.g., if we normalise only for classes *a* and *b*:

		Actual	
		a	*b*
Predicted	*a*	v_{aa} / (v_{aa} + v_{ab})	x= v_{ba} / (v_{ba} + v_{bb})
	b	y= v_{ab} / (v_{aa} + v_{ab})	v_{bb} / (v_{ba} + v_{bb})

We have max(1/2, (x+y)/2) and the same for the rest of combinations. Namely:

$$\mathrm{HT2}= (\max(1/2\ ,\ 1 – (v_{ba}\ /\ (v_{ba} + v_{bb}) + v_{ab}\ /\ (v_{aa} + v_{ab}))/2)\ + \max(1/2,\ 1 – (v_{ca}\ /\ (v_{ca} + v_{cc}) + v_{ac}\ /\ (v_{aa} + v_{ac}))\ /2) + \max(1/2,\ 1 – (v_{cb}\ /\ (v_{cb} + v_{cc}) + v_{bc}\ /\ (v_{bb} + v_{bc}))\ /2))\ \ /\ 3$$

Finally, we can define a third variant that instead of computing partial AUCs of pairs of classes, computes the AUC of each class against the rest (1 vs. rest) and then average the results. For instance, the AUC of class *a* and the rest (*b* and *c* joined) will be obtained from a condensed 2x2 matrix:

		Actual	
		a	*rest*
Predicted	*a*	v_{aa} / (v_{aa} + v_{ab}+ v_{ac})	(v_{ba} + v_{ca}) / (v_{ba} + v_{ca} + v_{bb} + v_{bc} + v_{cb} + v_{cc})
	rest	(v_{ab}+ v_{ac}) / (v_{aa} + v_{ab}+ v_{ac})	(v_{bb} + v_{bc} + v_{cb} + v_{cc}) / (v_{ba} + v_{ca} + v_{bb} + v_{bc} + v_{cb} + v_{cc})

Using cells (a,rest) and (rest,a) we have:

$$\mathrm{AUC}_{\mathrm{a,rest}} = \max(1/2, 1 - [(v_{ab}+ v_{ac})\ /\ (v_{aa} + v_{ab}+ v_{ac})]/2 - [\ (v_{ba} + v_{ca})\ /\ (v_{ba} + v_{ca} + v_{bb} + v_{bc} + v_{cb} + v_{cc})]/2$$

In the same way we can obtain $\mathrm{AUC}_{a,rest}$ and $\mathrm{AUC}_{a,rest}$. This allows us to define HT3:

$$\mathrm{HT3}= (\mathrm{AUC}_{a,rest} + \mathrm{AUC}_{a,rest}\ + \mathrm{AUC}_{a,rest}\ )/\ 3$$

5.6 Experimental Evaluation

Once the previous approximations are presented, we are ready to evaluate them in comparison to the exact computation given by the HSA method.

We are interested in how well the approximations "rank" the classifiers. To evaluate which approximation is best, we generate a set of 100 random classifiers (more precisely, we randomly generate 100 normalised confusion matrices).

Then, we compute the value of each classifier for each approximation *a* (exact VUS, accuracy, macro-avg, mod-avb, 1-p trivial, HT1B, HT2, HT3). Next, we make a one-to-one comparison (a ranking) for each approximation *a* and fill a matrix M_a, which tells whether *i* is ranked above *j.* Done all this (for a detailed description of the methodology of this process, see [4]), we are ready to compare approximations.

For instance, given the matrices M_1 and M_2 of two different methods, we compare the discrepancy of the matrices in the following way:

$$disc = \frac{2\sum_{i=1}^{n} \sum_{j=1,i<j}^{n} \left| M_1(i,j) - M_2(i,j) \right|}{n(n-1)}$$

With this formula we can evaluate the discrepancy of the methods for 3 class problems with respect the real VUS computed with the HAS method. The results are:

Accuracy	Macro-avg	Mod-avg (0.76)	1-p trivial	HT1B	HT2	HT3
0.08707	0,087071	0.0587879	0.09131	0.10404	0.14081	0.09677

According to these results, the best approximation is the modified macro-average (generalised mean). Note that this is the only one that is better than accuracy. Note also that for 2 classes, AUC = geomean(TPR, TNR), while for 3 classes, the best result is obtained somehow in the middle between the arithmetic mean and the geometric mean. This modified mean obtains the lower discrepancy among the studied approximations and hence could be used as an alternative to accuracy and macro-avg.

6 Conclusions

In this paper we have addressed the extension of ROC analysis for multi-class problems. We have identified the trivial classifiers and then derived the discard conditions, identified the maximum and minimum VUS and their polytopes, as well as the VUS for any arbitrary set of crisp classifiers. This is computed through the HSA algorithm. We have then compared experimentally the real VUS with several other approximations for crisp classifiers, showing which approximation is best. The best approximation seems to be a modification of the macro-average for one classifier.

For soft classifiers (i.e., classifiers that accompany each prediction with the reliability or, even better, with the estimated probabilities of each class), we have performed some preliminary experiments (see [4]) which show that the best approximation for soft classifiers is HT3. It is precisely for soft classifiers where the results can be more directly applicable to real-world problems.

For the moment, the results of this work dissuade the use of Hand and Till's and related measures as an extension of AUC for more than two classes for one crisp classifier. We propose an alternative approximation (mod-average). Nonetheless, for the case of soft classifiers the preliminary results in [4] are good for Hand and Till's extension (1 vs. 1, i.e. HT2) but especially for Fawcett's extension (1 vs. rest, i.e. HT3) already used in [9][10] for sets of classifiers or soft classifiers. Pursuing the work initiated here will bring a more justified use of AUC extensions as evaluation measure for classifiers. As future work, we would like to work further on soft classifiers, deriving accurate approximations of the real VUS in a reasonable time.

Acknowledgements

We thank Peter Flach for introducing us in the area of ROC analysis, and Tom Fawcett for some discussions on the multi-class extension. This work has been partially supported by CICYT under grant TIC2001-2705-C03-01, by the project DPI2001-2094-C03-03 from the Spanish Government, and by Universitat Politècnica de València under grant ref. 20020651. M.A. Salido enjoyed a stay as visiting research fellow (PPI-02-03) also funded by Universitat Politècnica de València.

References

1. Adams, N.M., Hand, D.J. "Comparing classifiers when the misallocation costs are uncertain, Pattern Recognition, Vol. 32 (7) (1999) pp. 1139-1147.
2. Barber, C.B.; Huhdanpaa, H. "QHull", The Geometry Center, University of Minnesota, http://www.geom.umn.edu/software/qhull/.
3. Boissonat, J.D.; Yvinec, M. "Algorithmic Geometry" Cambridge University Press, 1998.
4. Ferri C., Hernández-Orallo J., Salido M. A., "Volume Under the ROC Surface for Multiclass Problems. Exact Computation and Evaluation of Approximations" Technical Report DSIC. Univ. Politèc. València. 2003. http://www.dsic.upv.es/users/elp/cferri/VUS.pdf.
5. Flach P., Blockeel H., Ferri C., Hernández-Orallo J., Struyf J. "Decision support for data mining; Introduction to ROC analysis and its applications". In Data Mining and Decision Support: Integration and Collaboration. Kluwer Publishers. To appear. 2003.
6. Hand, D.J.; Till, R.J. "A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems" Machine Learning, 45, 171-186, 2001.
7. Hanley, J.A.; McNeil, B.J. "The meaning and use of the area under a receiver operating characteristic (ROC) curve" Radiology. 1982: 143:29-36.
8. Lane, T. "Extensions of ROC Analysis to Multi-Class Domains", ICML-2000 Workshop on cost-sensitive learning, 2000.
9. Provost, F., Fawcett, T. "Analysis and visualization of classifier performance: Comparison under imprecise class and cost distribution" in Proc. of The Third International Conference on Knowledge Discovery and Data Mining (KDD-97), pp. 43-48, AAAI Press, 1997.
10. Provost, F., Domingos P. "Tree Induction for Probability-based Ranking" Machine Learning 52:3, 199-215 , 2003.
11. Salido, M.A.; Giret, A. Barber, F. "Constraint Satisfaction by means of Dynamic Polyhedra" in Operations Research Proceedings 2001. Springer Verlag, pp: 405-412, 2002.
12. Srinivasan, A. "Note on the Location of Optimal Classifiers in N-dimensional ROC Space" Technical Report PRG-TR-2-99, Oxford University Computing Laboratory,
13. Swets, J., Dawes, R., Monahan, J. "Better decisions through science" Scientific American, October 2000, 82-87.
14. Turney P. "Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm", Journal of Artificial Intelligence Research, 2, 369-409, 1995.
15. Zweig, M.H.; Campbell, G. "Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine", Clin. Chem, 1993; 39: 561-77.