Volume Under the ROC Surface for Multi-class Problems

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

Departament de Sistemes Informàtics i Computació

Universitat Politècnica de València,

Valencia, Spain

{cferri, jorallo, msalido}@dsic.upv.es

ECML’03, Cavtat-Dubrovnik, September 22-26, 2003.

Outline

- Motivation
- ROC Analysis
- Multi-class ROC Analysis
- HSA for Computing the ROC Polytopes
- Evaluation of Approximations
- Conclusions and Future Work

Motivation

- Cost-sensitive Learning is a more realistic generalisation of predictive learning:
  - Costs are not the same for all kinds of misclassifications.
  - Class distributions are usually unbalanced.
- ROC Analysis:
  - Useful for choosing classifiers when costs are not known in advance.
- AUC (Area Under the ROC Curve):
  - A simple measure for each classifier, which estimates:
    - The quality of the classifier for a range of class distributions.
    - A measure of how well the classifier ranks examples (equivalent to the Wilcoxon statistic)

Motivation

- Applications:
  - ROC analysis and AUC-related measures have been used in many areas: medical decision making, marketing campaign design, probability estimation, etc.
- Problems:
  - ROC Analysis has not been extended for more than two classes, because of a difficult definition and complexity.
  - There are approximations of the AUC measure for more than two classes, but:
    - No acquaintance about the quality of these approximations.
- Goal:

Extend ROC analysis to more than 2 classes and evaluate approximations.

ROC Analysis

- Receiver Operating Characteristic (ROC) Analysis is useful when we don't know:
  - The proportion of examples of each class in application time (class distribution)
  - The cost matrix in application time

- ROC Analysis can be applied in these situations. Provides tools to:
  - Distinguish classifiers that can be discarded under any circumstance (class distribution or cost matrix).
  - Select the optimal classifier once the cost matrix is known.

ROC Analysis

- Given a confusion matrix:

Real

Predicted	Yes	No
Yes	30	20
No	10	40

- We can normalise each column

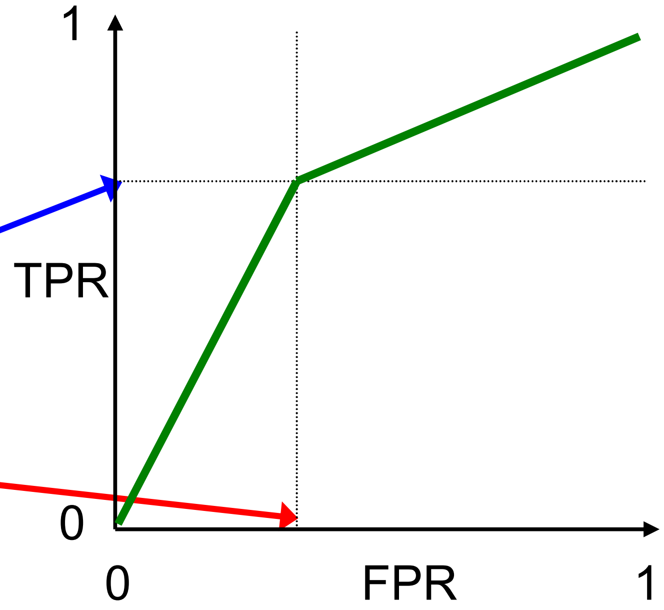
Real

Predicted	Yes	No
Yes	0.75	0.33
No	0.25	0.67

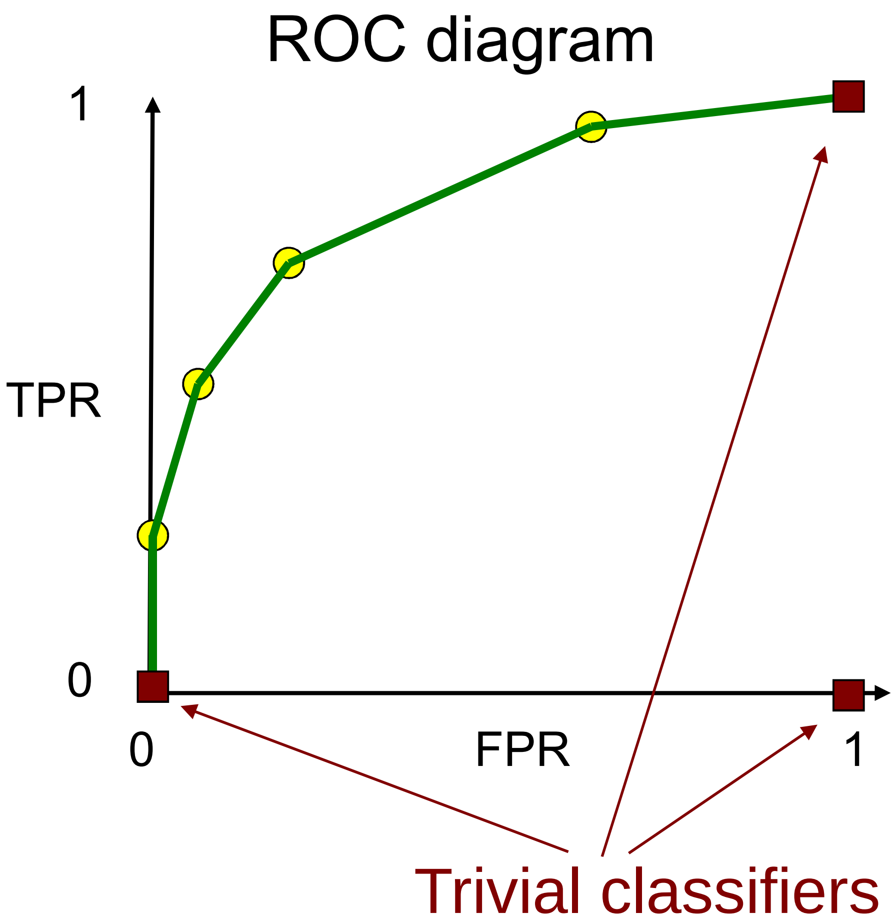
TPR

FPR

ROC diagram



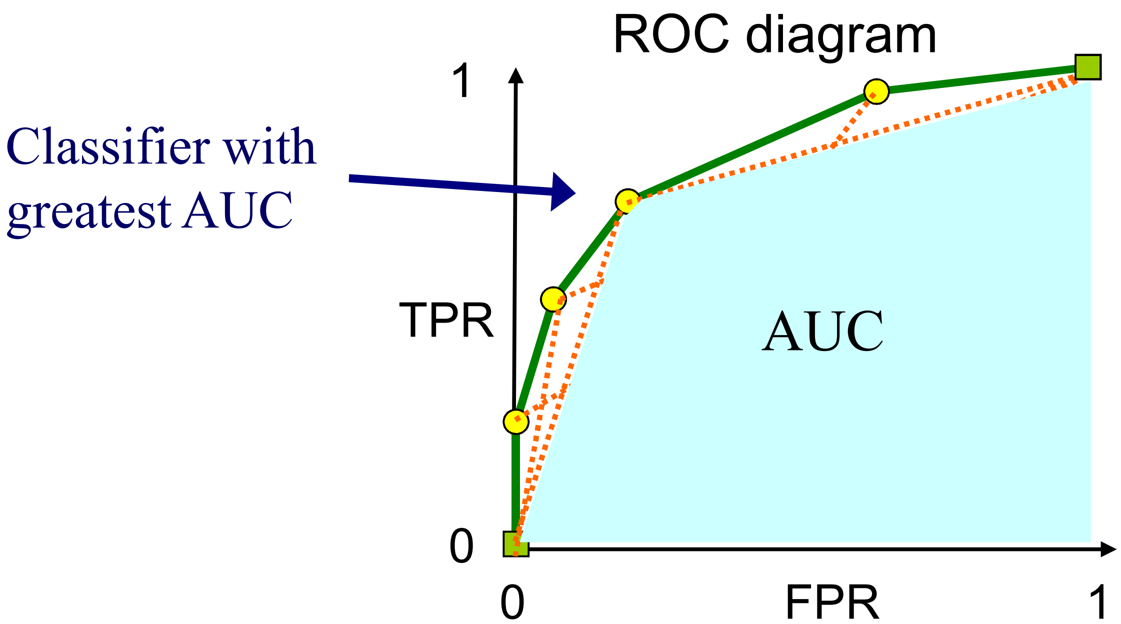
ROC Analysis



- Given several classifiers:
  - We can construct the convex hull of their points (FPR,TPR) and the trivial classifiers (0,0), (1,1), (1,0).
  - The classifiers falling under the ROC curve can be discarded.
  - The best classifier of the remaining classifiers can be chosen in application time.

ROC Analysis

- If we want to select *one* classifier:



  - We calculate the Area Under the ROC Curve (AUC) of all the classifiers and choose the one with greatest AUC.

Multi-class ROC Analysis

- Classes and Dimensions
  - For 2 classes, there is a 2×2 matrix, and there are 2 degrees of freedom. Hence 2 dimensions.
  - For 3 classes, there is a 3×3 matrix, and there are 6 degrees of freedom. Hence 6 dimensions.
  - For n classes … $d = n \times (n-1)$ dimensions.

- Problems:
  - Representation of 6 or more dimensions difficult.
  - The identification of the trivial classifiers is not clear.
  - The computation of the convex hull of N points in a d-dimensional space is in $O(N \log N + N^{d/2})$.

Multi-class ROC Analysis

- Example. 3 classes.

		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

- The x_i give a 6-dimensional point. The values h_a, h_b, h_c are dependent since:
  - $h_a+x_3+x_5=1$
  - $h_b+x_1+x_6=1$
  - $h_c+x_2+x_4=1$

- We can't represent a ROC diagram, but still we could obtain the AUC.
  - called in this case VUS (Volume Under the ROC Surface)

Multi-class ROC Analysis

- Maximum VUS for 3 classes.
  - A point is a classifier if and only if:

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

  - The space determined by these equations can be easily obtained:
    - It is equal to the probability that 6 random numbers under a uniform distribution U(0,1) follow these conditions

$$VUS_3^{max} = P(U(0,1) + U(0,1) \leq 1) \cdot P(U(0,1) + U(0,1) \leq 1) \cdot P(U(0,1) + U(0,1) \leq 1) = [P(U(0,1) + U(0,1) \leq 1)]^3 = (½)^3 = 1/8$$

- The previous expression can be approximated for more than 3 classes.

Multi-class ROC Analysis

- Minimum VUS for 3 classes.

  - Trivial classifiers:

    » Where $h_a + h_b + h_c = 1$

Predicted \ Actual	a	b	c
a	h_a	h_a	h_a
b	h_b	h_b	h_b
c	h_c	h_c	h_c

  - We can discard a classifier if and only if it is above a trivial classifier:

$$\exists h_a, h_b, h_c \in R^+ \text{ where } (h_a + h_b + h_c = 1) \text{ such that: }$$
$$x_1 \geq h_a,\ x_2 \geq h_a,\ x_3 \geq h_b,\ x_4 \geq h_b,\ x_5 \geq h_c,\ x_6 \geq h_c$$

  - This can be simplified into:

- Theorem 1:
  - A classifier $(x_1, x_2, x_3, x_4, x_5, x_6)$ can be discarded iff:

$$r_1 + r_2 + r_3 \geq 1$$

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

- By a Montecarlo method, the minimum is approximated to 1/180.

HSA for Computing the ROC Polytopes

- The inequations (constraints) for max and min make it very difficult to obtain the exact values analytically.

How can we obtain the maximum and minimum VUS values *exactly?*

- And, more importantly,

How can we obtain the VUS of any classifier *exactly?*

- HSA (Hyperpolyhedron Search Algorithm):
  - A Constraint Satisfaction Problem (CSP) Solver.
  - Manages non-binary and continuous problems.
  - Uses linear programming techniques.
  - We will use HSA to determine the extreme solutions (hyperpolyhedron)

HSA for Computing the ROC Polytopes

- Minimum and Maximum VUS with HSA.
  - Maximum. We solve the constraints:

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

    - We have 1/8, as expected.
  - Minimum. Given the equations:

$$r_1 + r_2 + r_3 \geq 1$$

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

    - We transform them into:

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

    - Which can be solved by HSA, giving 1/180, as expected.

HSA for Computing the ROC Polytopes

- Computing the VUS of any classifier with HSA.
  - Basic Idea: Given a classifier, we combine it with the trivial classifier in order to know the volume of the classifiers it discards.
    - The linear combination of one classifier *z* and the trivial classifiers is given by:

$$ha \cdot (1, 1, 0, 0, 0, 0) + hb \cdot (0, 0, 1, 1, 0, 0) + hc \cdot (0, 0, 0, 0, 1, 1) + hd \cdot (zba, zca, zab, zcb, zac, zbc)$$

  - We can discard a classifier *v* iff:

$$\exists ha, hb, hc, hd \in R^+ \text{ where } (ha + hb + hc + hd = 1) \text{ such that:}$$
$$vba \geq ha + hd \cdot zba,\ vca \geq ha + hd \cdot zca,\ vab \geq hb + hd \cdot zab,$$
$$vcb \geq hb + hd \cdot zcb,\ vac \geq hc + hd \cdot zac,\ vbc \geq hc + hd \cdot zbc$$

  - This sums up to a system of inequations with 10 variables that HAS can solve.

HSA for Computing the ROC Polytopes

- Computing the VUS of a set of classifiers with HSA.
  - The idea can be extended to a set of classifiers.
  - E.g. given four classifiers, we can calculate the VUS of the convex hull of the four classifiers.
    - The linear combination of four classifier *z, w, x* and *y,* and the trivial classifiers is given by:

$$ha \cdot (1, 1, 0, 0, 0, 0) + hb \cdot (0, 0, 1, 1, 0, 0) + hc \cdot (0, 0, 0, 0, 1, 1) + h1 \cdot (zba, zca, zab, zcb, zac, zbc)$$

$$+ h2 \cdot (wba, wca, wab, wcb, wac, wbc) + h3 \cdot (xba, xca, xab, xcb, xac, xbc) + h4 \cdot (yba, yca, yab, ycb, yac, ybc)$$

  - In the same way as before, we have a system with 9+4 variables, which can be solved by HAS.

Evaluation of Approximations

- Now we are able to obtain the real VUS.
  - The calculation is expensive, especially for 4 or more classes (12 or more dimensions).
  - However, it can be used as a reference for evaluating current or new approximations.
- Approximations to the VUS for crisp classifiers:
  - Since, to date, the real AUC (VUS) could not be calculated, there have been many approximations:
    - Macro-average
    - Macro-average Modified
    - 1-point trivial AUC extension
    - 1-point Hand and Till Extension

Evaluation of Approximations

- Macro-average
  - Given a classifier:

Actual

Predicted

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

  - The macro-average is just the average of the partial class accuracies.

$$MAVG_3 = (v_{aa} + v_{bb} + v_{cc}) / 3$$

  - Since the matrix is normalised, for points, this is equivalent to accuracy.

Evaluation of Approximations

- Macro-average Modified
  - Macro-average does not take into account that extreme partial accuracies are not good for AUC.
    - Example: (0.2, 0.2) has more AUC than (0.1, 0.3), although macro-average is the same.
  - One solution is a geometric mean, a macro-geomean, but this can be too much.
  - A more general solution is the generalised mean:

$$\text{MAVG3-MOD} = \left(\frac{1}{n}\sum_{k=1}^{n} a_k^t\right)^{1/t} .$$

    - With *t* being a factor to be estimated.

Evaluation of Approximations

- 1-point trivial AUC extension
  - We know that the AUC for two classes is:

$$\text{AUC2} = \max(1/2,\ 1 - vba\ /2 - vab\ /2)$$

  - Extending it trivially to 3 classes we have:

$$\text{AUC-1PT3} = \max(1/3,\ 1 - (vba + vca + vab + vcb\ + vac + vbc)\ /3$$

    – Quite similar to macro-average, but different in some situations.

Evaluation of Approximations

- 1-point Hand and Till Extension
  - Hand and Till presented an extension of the AUC measure for more than 2 classes as a one-to-one weighting of all combinations.

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

  - We consider three different variants for crisp classifiers:

$$HT1b = (\max(1/2\,,\ 1-(vba+vab)/2) + \max(1/2,\ 1-(vca+vac)/2) + \max(1/2,\ 1-(vcb+vbc)/2))\ /\ 3$$

$$HT2 = (\max(1/2\,,\ 1-(vba/(vba+vbb)+vab/(vaa+vab))/2) + \max(1/2,\ 1-(vca/(vca+vcc) + vac/(vaa+vac))/2) + \max(1/2,\ 1-(vcb/(vcb+vcc)+vbc/(vbb+vbc))/2))\ /\ 3$$

$$HT3 = (AUC\,a,rest + AUC\,a,rest + AUC\,a,rest)/\ 3$$

  - being:

$$AUCa,rest = \max(1/2, 1 - [(vab+vac)/(vaa+vab+vac)]/2 - [(vba+vca)/(vba+vca+vbb+vbc+vcb+vcc)]/2$$

Evaluation of Approximations

- Evaluation:
  - It is based on how well the approximations “rank” the classifiers, in comparison to the ranking, given to the real VUS.
  - We define a measure of discrepancy.
  - The results are:

Accuracy	Macro-avg	Mod-avg (0.76)	1-p trivial	HT1B	HT2	HT3
0.0871	0.0871	0.0588	0.0913	0.104	0.141	0.0968

  - The best results are obtained by the modified macro-average.
  - More importantly, it is the only measure that is better than accuracy for evaluating crisps classifiers for ranking!

Conclusions and Future Work

- Conclusions:
  - The extension of ROC analysis, and related measures (AUC → VUS) has been addressed.
    - We have identified the maximum VUS and the minimum VUS, and the general inequations.
    - We can solve these inequations through the HSA algorithm and hence obtain the VUS of any classifier and any set of classifiers.
  - We have compared the approximations for VUS with the real VUS obtained by HAS.
    - We have shown that only a modification of the macro-average is better than accuracy for evaluating crisp classifiers, if we want to use them for ranking.

Conclusions and Future Work

- Ongoing Work:
  - The evaluation of approximations of VUS for soft classifiers is our main immediate goal.
  - We are evaluating approximations for *soft* classifiers (probability estimators). In this case,
    - Hand and Till's approximation (1vs1) seems to be better than accuracy.
    - Fawcett's approximation (1vsAll) performs still better.

- Future Work
  - Development of new approximations of VUS for soft classifiers.
    - Much more accurate than current approximations.
    - Much more efficient than HAS (able to cope with 5, 6 or more classes).