The Expected Performance Curve

Samy Bengio BENGIO@IDIAP.CH
Johnny Mariéthoz MARIETHO@IDIAP.CH
Mikaela Keller MKELLER@IDIAP.CH
IDIAP Research Institute, CP 592, rue du Simplon 4, 1920 Martigny, Switzerland

Abstract

In several research domains concerned with classification tasks, curves like ROC are often used to assess the quality of a particular model or to compare two or more models with respect to various operating points. Researchers also often publish some statistics coming from the ROC, such as the so-called *break-even point* or *equal error rate*. The purpose of this paper is to first argue that these measures can be misleading in a machine learning context and should be used with care. Instead, we propose to use the *Expected Performance Curves* (EPC) which provide unbiased estimates of performance at various operating points. Furthermore, we show how to use adequately a non-parametric statistical test in order to produce EPCs with confidence intervals or assess the statistical significant difference between two models under various settings.

1. Introduction

Two-class classification problems are common in machine learning. In several domains, on top of selecting the appropriate discriminant function, practitioners also modify the corresponding threshold in order to better suit an independent cost function. Moreover, they compare models with respect to the whole range of possible values this threshold could take, generating curves such as ROCs. In order to also provide quantitative comparisons, they often select one particular point on this curve (such as the so-called *break-even point* or *equal error rate*).

The main purpose of this paper is to argue that such curves, as well as particular points on it like *break-even point* or *equal error rate* can be misleading when used to compare two or more models, or to obtain a realistic estimate of the expected performance of a given model.

We thus propose instead the use of a new set of curves, called *Expected Performance Curves* (EPC), which really reflect the expected (and reachable) performance of systems. While EPCs are presented here for general machine learning tasks, they were first presented specifically in the context of person authentication in (Bengio & Mariéthoz, 2004).

Furthermore, we propose here the use of a simple non-parametric technique to show a confidence interval along the EPCs or to show regions where two models are statistically significantly different from each other with a given level of confidence.

In Section 2, we review the various performance measures used in several research domains in front of 2-class classification tasks, such as person authentication and text categorization. In Section 3, we explain why some of these measures can be misleading. In Section 4, we present the family of EPCs, that really reflects the expected performance of a given model, hence enabling a fair comparison between models. Finally, in Section 5, we present a technique to compute confidence intervals and statistical significance tests together with EPCs. Section 6 concludes the paper.

2. Performance Measures for 2-Class Classification Tasks

Let us consider two-class classification problems defined as follows: given a training set of examples (x_i, y_i) where x_i represents the input and y_i is the target class $\in \{0, 1\}$, we are searching for a function $f(\cdot)$ and a threshold θ such that

$$f(x_i) > \theta \text{ when } y_i = 1 \text{ and } f(x_i) <= \theta \text{ when } y_i = 0, \ \forall i \,. \tag{1}$$

The obtained function $f(\cdot)$ (and associated threshold

Appearing in *Proceedings of the ICML 2005 workshop on ROC Analysis in Machine Learning*, Bonn, Germany, 2005.
Copyright 2005 by the author(s)/owner(s).

		Desired Class	
		1	0
Obtained Class	1	TP	FP
	0	FN	TN

Table 1. Types of errors in a 2-class classification problem.

θ) can then be tested on a separate test data set and one can count the number of utterances of each possible outcome: either the obtained class corresponds to the desired class, or not. In fact, one can decompose these outcomes further, as exposed in Table 1, in 4 different categories: *true positives* (where both the desired and the obtained class is 1), *true negatives* (where both the desired and the obtained class is 0), *false positives* (where the desired class is 0 and the obtained class is 1), and *false negatives* (where the desired class is 1 and the obtained class is 0). Let TP, TN, FP and FN represent respectively the *number of utterances* of each of the corresponding outcome in the data set.

Note once again that TP, TN, FP, FN and all other measures derived from them are in fact dependent both on the obtained function $f(\cdot)$ and the threshold θ. In the following, we will sometimes refer to, say, FP by $\text{FP}(\theta)$ in order to specifically show the dependency with the associated threshold.

Several tasks are in fact specific incarnations of 2-class classification problems. However, often for historical reasons, researchers specialized in these tasks have chosen different methods to measure the quality of their systems. In general the selected measures come by pair, which we will call generically here $V1$ and $V2$, and are simple antagonist combinations of TP, TN, FP and FN. Moreover, a unique measure (V) often combines $V1$ and $V2$. For instance,

- in the domain of person authentication (Verlinde et al., 2000), the chosen measures are
$$V1 = \frac{\text{FP}}{\text{FP} + \text{TN}} \text{ and } V2 = \frac{\text{FN}}{\text{FN} + \text{TP}} \quad (2)$$
and are called *false acceptance rate* (FAR) and *false rejection rate* (FRR) respectively. Several aggregate measures have been proposed, the simplest being the *half total error rate* (HTER)
$$V = \frac{V1 + V2}{2} = \frac{\text{FAR} + \text{FRR}}{2} = \text{HTER} \ , \quad (3)$$
- in the domain of text categorization (Sebastiani, 2002),
$$V1 = \frac{\text{TP}}{\text{TP} + \text{FP}} \text{ and } V2 = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (4)$$
and are called *precision* and *recall* respectively. Again several aggregate measures exist, such as the *F1* measure
$$V = \frac{2 \cdot V1 \cdot V2}{V1 + V2} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = F1 \ , \quad (5)$$
- in medical studies,
$$V1 = \frac{\text{TP}}{\text{TP} + \text{FN}} \text{ and } V2 = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (6)$$
and are called *sensitivity* and *specificity* respectively (Zweig & Campbell, 1993).

In all the cases, in order to use the system effectively, one has to select the threshold θ according to some criterion which is in general of the following generic form
$$\theta^\star = \arg\min_\theta g(V1(\theta), V2(\theta)) \ . \quad (7)$$
Examples of $g(\cdot,\cdot)$ are the HTER and *F1* functions already defined in equations (3) and (5) respectively. However, the most used criterion is called the *break even point* (BEP) or *equal error rate* (EER) and corresponds to the threshold nearest to a solution such that $V1 = V2$, often estimated as follows:
$$\theta^\star = \arg\min_\theta |\text{V1}(\theta) - \text{V2}(\theta)| \ . \quad (8)$$

Note that the choice of the threshold can have a significant impact in the resulting system: in general θ represents a trade-off between giving importance to $V1$ or $V2$. Hence, instead of committing to a single operating point, an alternative method to present results often used in the research community is to produce a graph that presents $V1$ with respect to $V2$ for all possible values of θ. Such a graph is called the *Receiver Operating Characteristic* (ROC) (Green & Swets, 1964)[1]. Figure 1 shows an example of two typical ROCs. Note that depending on the precise definition of $V1$ and $V2$, the best curve would tend to one of the four corners of the graph. In Figure 1, the best curve corresponds to the one nearest to the bottom left corner (corresponding to simultaneous small values of $V1$ and $V2$).

Instead of providing the whole ROC, researchers often summarize it by some typical values taken from it; the most common summary measure is computed by using the BEP, already described in equation (8), which produces a single value of θ and to produce some

[1]Note that the original ROC plots the true positive rate with respect to the false positive rate, but several researchers use the name *ROC* with various other definitions of $V1$ and $V2$.

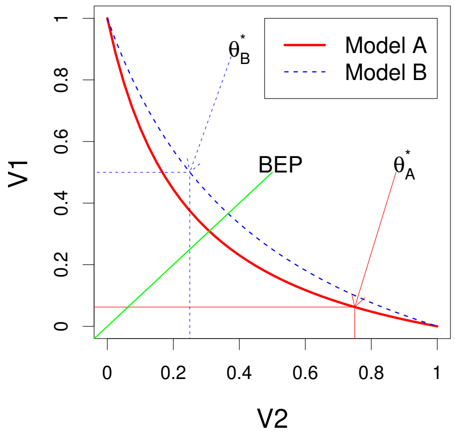

Figure 1. Example of two typical ROCs.

aggregate value $V(\theta)$ (such as *F1* or HTER). On Figure 1, the line intersecting the two ROCs is the BEP line and the intersections with each ROC correspond to their respective BEP point.

3. Cautious Interpretation of ROC and BEP

As explained above, researchers often use ROC and BEP to present and compare their results; for example, all results presented in (Sebastiani, 2002), which is a very good survey of text categorization, are presented using the BEP; a recent and complete tutorial on text independent speaker verification (Bimbot et al., 2004) proposes to measure performance through the use of DET curves, which are non-linear versions of ROCs, as well as the error corresponding to equal error rate, hence the BEP. We would like here to draw the attention of the reader to some potential risk of using ROC or BEP for comparing two systems, as it is done for instance in Figure 1, where we compare the test performance of models A and B. As can be seen on this Figure, and reminding that in this case $V1$ and $V2$ must be minimized, the best model appears to always be model A, since its curve is always below that of model B. Moreover, computing the BEP of models A and B yields the same conclusion.

Let us now remind that each point of the ROC corresponds to a particular setting of the threshold θ. However, in real applications, θ needs to be decided prior to seeing the test set. This is in general done using some criterion of the form of equation (7) such as searching for the BEP, equation (8), using some development data (obviously different from the test set).

Hence, assuming for instance that one decided to select the threshold according to (8) on a development set, the obtained threshold may not correspond to the BEP on the test set. There are many reasons that could yield such mismatch, the simplest being that assuming the test and development sets to come from the same distribution but be of fixed (non-infinite) size, the estimate of (8) on one set is not guaranteed to be the same as the estimate on the other set.

Let us call $\theta^\star_A$ the threshold estimated on the development set using model A and similarly for $\theta^\star_B$. While the hope is that both of them should be aligned, on the test set, with the BEP line, there is nothing, in theory, that prevents them to be slightly or even largely far from it. Figure 1 shows such an example, where indeed,

$$V1(\theta^\star_B) + V2(\theta^\star_B) < V1(\theta^\star_A) + V2(\theta^\star_A) \qquad (9)$$

even though the ROC of model A is always below that of model B, including at the intersection with the BEP line[2]. One might argue that this may only rarely happen, but we have indeed observed this scenario several times in person authentication and text categorization tasks, including a text independent speaker verification application where the problem is described in more details in (Bengio & Mariéthoz, 2004). We replicate in Figure 2 the ROCs obtained on this task using two different models, with model B apparently always better than model A. However, when selecting the threshold on a separate validation set (hence simulating a real life situation), the HTER of model A becomes lower than of model B (the graph shows the operating points selected for the two models).

In summary, showing ROCs has potentially the same drawbacks and risks as showing the training error (indeed, one parameter, the threshold, has been implicitly tuned on the test data) and expect that it reflects the expected generalization error: this is true when the size of the data is huge, but false in the general case. Furthermore, real applications often suffer from an additional mismatch between training and test conditions which should be reflected in the procedure.

4. The Expected Performance Curve

We have seen in Section 2 that given the trade-off between $V1$ and $V2$, researchers often prefer to provide a curve that assesses the performance of their model for all possible values of the threshold. On the other hand, we have seen in Section 3 that ROCs can be misleading since selecting a threshold prior to seeing the test set (as it should be done) may end up in ob-

[2] Note that at BEP, $V1 = V2 = \frac{1}{2}(V1 + V2)$.

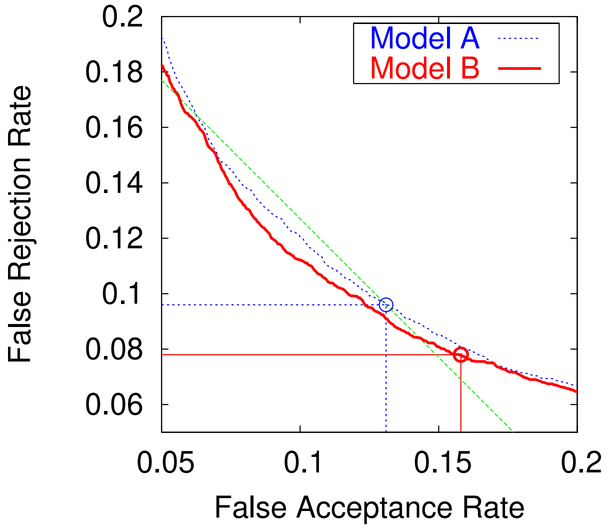

Figure 2. ROCs of two real models for a Text-Independent Speaker Verification task.

taining a different trade-off in the test set. Hence, we would like here to propose the use of new curves which would let the user select a threshold according to some criterion, in an unbiased way, and still present a range of possible expected performances on the test set. We shall call these curves Expected Performance Curves (EPC).

4.1. General Framework

The general framework of EPCs is the following. Let us define some parametric performance measure $C(V1(\theta, D), V2(\theta, D); \alpha)$ which depends on the parameter α as well as $V1$ and $V2$ computed on some data D for a particular value of θ. Examples of $C(\cdot, \cdot; \alpha)$ are the following:

- in person authentication, one could use for instance

$$\begin{aligned} & C(V1(\theta, D), V2(\theta, D); \alpha) \quad (10) \\ & \quad = C(\text{FAR}(\theta, D), \text{FRR}(\theta, D); \alpha) \\ & \quad = \alpha \cdot \text{FAR}(\theta, D) + (1 - \alpha) \cdot \text{FRR}(\theta, D) \end{aligned}$$

  which basically varies the relative importance of $V1$ (FAR) with respect to $V2$ (FRR); in fact, setting $\alpha = 0.5$ yields the HTER cost (3);

- in text categorization, since the goal is to maximize precision and recall, one could use

$$\begin{aligned} & C(V1(\theta, D), V2(\theta, D); \alpha) \quad (11) \\ & \quad = C(\text{Precision}(\theta, D), \text{Recall}(\theta, D); \alpha) \\ & \quad = -(\alpha \cdot \text{Precision}(\theta, D) + (1 - \alpha) \cdot \text{Recall}(\theta, D)) \end{aligned}$$

  where $V1$ is the precision and $V2$ is the recall;

- in general, one could also be interested in trying to reach a particular relative value of $V1$ (or $V2$), such as *I am searching for a solution with as close as possible to 10% false acceptance rate*; in that case, one could use

$$C(V1(\theta, D), V2(\theta, D); \alpha) = |\alpha - V1(\theta, D)| \quad (12)$$

  or

$$C(V1(\theta, D), V2(\theta, D); \alpha) = |\alpha - V2(\theta, D)| \; . \quad (13)$$

Having defined $C(\cdot, \cdot; \alpha)$, the main procedure to generate the EPC is to vary α inside a reasonable range (say, from 0 to 1), and for each value of α, to estimate θ that minimizes $C(\cdot, \cdot; \alpha)$ on a development set, and then use the obtained θ to compute some aggregate value (say, V), on the test set. Algorithm 1 details the procedure, while Figure 3 shows an artificial example of comparing the EPCs of two models. Looking at this figure, we can now state that for specific values of α (say, between 0 and 0.5), the underlying obtained thresholds are such that model B is better than model A, while for other values, this is the converse. This assessment is unbiased in the sense that it takes into account the possible mismatch one can face while estimating the desired threshold.

Let us suppose that Figure 3 was produced for a person authentication task, where V is the HTER, $V1$ is the FAR, and $V2$ is the FRR. Furthermore let us define the criterion as in (10). In that case, α varies from 0 to 1, and when $\alpha = 0.5$ this corresponds to the setting where we tried to obtain a BEP (or Equal Error Rate, as it is called in this domain), while when $\alpha < 0.5$ it corresponds to settings where we gave more importance to false rejection errors and when $\alpha > 0.5$ we gave more importance to false acceptance errors.

In order to illustrate EPCs in real applications, we have generated them for both a person authentication task and a text categorization task. The resulting curves can be seen in Figures 4 and 5. Note that the graph reporting $F1$ seems inverted with respect to the one reporting HTER, but this is because we are searching for low HTERs in person authentication but high $F1$ in text categorization. Note also that the EPC of Figure 4 corresponds to the ROC of Figure 2. Finally, note that we kindly provide a C++ tool that generates such EPCs[3].

[3] An EPC generator is available at `http://www.Torch.ch/extras/epc` as a package of the Torch machine learning library.

Algorithm 1 Method to generate the Expected Performance Curve

Let *devel* be the development set
Let *test* be the test set
Let $V(\theta, D)$ be the value of V obtained on the data set D for threshold θ
Let $C(V1(\theta, D), V2(\theta, D); \alpha)$ be the value of a criterion C that depends on α, and is computed on the data set D
for values $\alpha \in [a, b]$ where a and b are reasonable bounds **do**
  $\theta^\star = \arg\min_\theta C(V1(\theta, devel), V2(\theta, devel); \alpha)$
  compute $V(\theta^\star, test)$
  plot $V(\theta^\star, test)$ with respect to α
end for

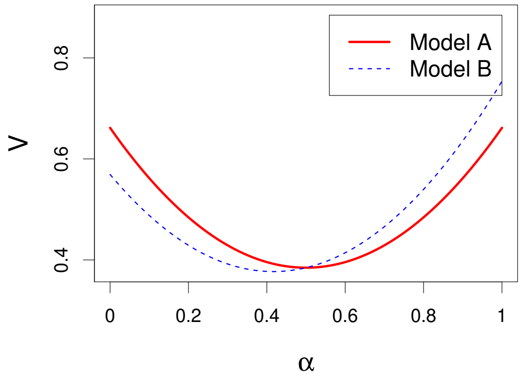

Figure 3. Example of two typical EPCs.

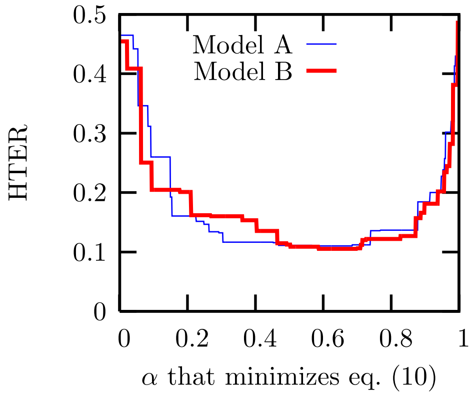

Figure 4. Expected Performance Curves for person authentication, where one wants to trade-off false acceptance rates with false rejection rates.

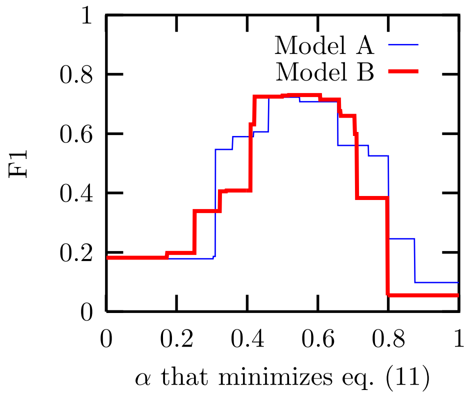

Figure 5. Expected Performance Curves for text categorization, where one wants to trade-off precision and recall and print the $F1$ measure.

4.2. Areas Under the Expected Performance Curves

In general, people often prefer to compare their models according to a unique quantitative performance measure, rather than through the use of curves which can be difficult to interpret. One solution proposed by several researchers is to summarize the ROC by some approximation of the area under it.

Knowing that the ROC may in fact be a misleading measure of the expected performance of a system, the corresponding area under it may also be misleading. Would it be possible to obtain a measure of the expected performance over a given range of operating points? We propose here to compute $E[\bar{V}]$, the expected value of $\bar{V}$, which would be defined as the average between two antagonist measures $V1$ and $V2$ given a criterion $C(\cdot, \cdot; \alpha)$. We will show that this is in fact related to the area under the ROC curve (AUC), although it now concerns an area under a curve of *reachable* solutions instead of theoretical solutions. Note that there are several theoretical properties of the AUC measure which makes it appealing, such as the fact that, when $V1$ and $V2$ are respectively defined as the true and false positive rates, it corresponds to the well-known Mann-Whitney statistical test which can be used to compare two independent groups of sampled data (Hanley & McNeil, 1982).

Let $\theta_{f=\alpha}$ be the threshold such that

$$\theta_{f=\alpha} = \arg\min_\theta |\alpha - f(\theta)|, \tag{14}$$

we can write the expected value of $\bar{V} = \frac{V1+V2}{2}$ using $V1$ as a threshold selection criterion, as follows:

$$E_{V1}[\bar{V}] = \frac{1}{2}\int_{\alpha\in[0,1]} [V1(\theta_{V1=\alpha}) + V2(\theta_{V1=\alpha})]\, d\alpha, \tag{15}$$

and using $V2$ as criterion,

$$E_{V2}[\bar{V}] = \frac{1}{2} \int_{\beta \in [0,1]} \left[V1(\theta_{V2=\beta}) + V2(\theta_{V2=\beta}) \right] d\beta. \tag{16}$$

Note that if we select the thresholds θ on the test set then,

$$\begin{aligned} & V1(\theta_{V1=\alpha}) = \alpha, \quad V2(\theta_{V2=\beta}) = \beta, \\ & \int_{\alpha \in [0,1]} V2(\theta_{V1=\alpha}) d\alpha = \text{AUC} \quad \text{and} \\ & \int_{\beta \in [0,1]} V1(\theta_{V2=\beta}) d\beta = \text{AUC} \;. \end{aligned} \tag{17}$$

Thus, using the fact that $\int_0^1 \gamma d\gamma = \frac{1}{2}$, we can obtain the relation between the expected $\bar{V}$ when the thresholds are computed on the test set (which we will call $E[\bar{V}]_{post}$) and the area under the ROC, by computing the average of equations (15) and (16) when the threshold is chosen on the test set:

$$\begin{aligned} G(\bar{V})_{post} &= \frac{1}{2} \left\{ E_{V1}[\bar{V}]_{post} + E_{V2}[\bar{V}]_{post} \right\} \\ &= \frac{1}{2} \left\{ \text{AUC} + \frac{1}{2} \right\}. \end{aligned} \tag{18}$$

Of course, if we select the thresholds using a separate development set, the result obtained in (18) is not true anymore. However, in this case the average $G(\bar{V})$ remains interesting since it can be interpreted as a measure summarizing two EPCs. Indeed, the two components of the average, (15) and (16), are the area under an EPC computed using respectively criteria (12) and (13), hence it integrates two antagonist performance measures over a large range of operating points.

Note that in equations (15) and (16), we integrate $\bar{V}$ over expected values of $V1$ and $V2$ from 0 to 1. However in some cases, a value of $V1$ around, say 0, may be reachable but of no interest. In the field of person authentication it is common to only pay attention to "reasonable" values of FAR and FRR (hence it is not useful to take into account values of FAR greater than 0.5 for instance). The values of "reasonable" bounds are task dependent, but their choice can be decisive, and should be taken into account when computing the expected performance of the system.

5. Confidence Intervals and Statistical Difference Tests

While producing unbiased quantitative measures is important when assessing the performance of a model, it is also important to take into account some of the possible variability induced by the training or testing procedure involved in the process. Several statistical techniques do exist to estimate confidence intervals around an obtained performance or to assess the statistical significantness of the difference between the performance of two models (see for instance (Dietterich, 1998) for a good comparison of some of the available tests used in machine learning).

However, in most cases, these tests involve several hypotheses that cannot be met in the general case where the reported measure is some arbitrary combination of TP, TN, FP and FN (for instance the $F1$ measure used in text categorization cannot be considered as following a Normal distribution for which one could easily estimate the variance; moreover, the difference of two $F1$ measures cannot be decomposed into a sum of independent variables, since the numerator and the denominator are both non-constant sums of independent variables).

Hence, we would like to propose here the use of a non-parametric test based on the Bootstrap Percentile Test (Efron & Tibshirani, 1993) which has recently been applied to compute confidence intervals around ROCs (Bolle et al., 2004). We here suggest its use for the practical case of EPCs. The aim of this test is to estimate a given distribution using bootstrap replicates of the available data. Given the estimated distribution, it is then easy to compute the probability that the random variable of interest is higher than a given threshold, which is the basis of most statistical tests.

Let us here give a generic example where the goal is to verify whether a given model A is statistically significantly better than a second model B, when their respective performance is measured with V_A and V_B, which are aggregates of $V1_A$, $V1_B$, $V2_A$ and $V2_B$, which are themselves defined using TP, TN, FP and FN, as explained in Section 2. Let us further assume that these measures were computed on some test set containing N examples, for a given threshold (normally selected on a separate development set, as explained in Section 4).

Let T be a table of two columns and N rows containing, for each of the N test examples, whether it was considered as a true positive, a true negative, a false positive or a false negative, for both models A (first column) and B (second column). It should be clear that with such a table, one can compute again any aggregate value V based on the numbers TP, TN, FP, and FN gathered from the table.

Let us now create M (where M should be a big integer, the bigger the better, say 10000) bootstrap replicates of table T. The i^{th} bootstrap replicate of T is done by creating a new table T_i also containing N rows of two columns, and where each row is a copy of one of the row of T, selected randomly *with replacement*. Thus, replicate T_i may contain some of the original rows of T in more than one copy, and may not contain some other rows of T. Interestingly, it can be shown that the T_i are drawn from the same distribution as T which is an empirical, unbiased and exhaustive estimate of the true distribution (Efron & Tibshirani, 1993).

Using each bootstrap replicate T_i, we now compute an estimate of our aggregate measure of interest, normally based on V_A and V_B: it could be the signed difference of V_A and V_B if we are interested in estimating whether model A is better than model B, or it could be only based on V_A if we want to estimate a confidence interval around V_A. This yields M estimates of our statistics of interests.

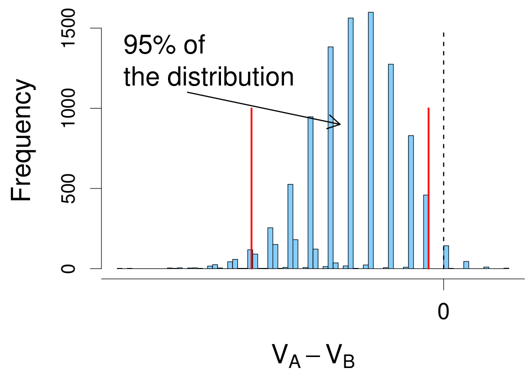

Figure 6. Example of an obtained histogram distribution of $V_A - V_B$, where the frequency of particular values of $V_A - V_B$ is plotted with respect to the values of $V_A - V_B$. The two vertical thick lines show the bounds of the 95% confidence interval and the vertical dashed line shows the value of 0, which in that case is not inside the interval.

Figure 6 shows an example of a histogram plot of M estimates of $V_A - V_B$. Using this histogram, one can for instance verify whether 0 (which corresponds to the point where $V_A = V_B$) is inside or outside a 95% confidence interval centered at the empirical mean; if it is outside (as it is the case in Figure 6), then one can assert with 95% confidence that V_A is statistically significantly different from V_B (in the case of Figure 6, V_B is higher than V_A more than 95% of the times); on the other hand, if 0 lies inside the bounds, then we cannot assert any statistical difference with 95% confidence.

The same technique could be used to compute a confidence interval around a single measure (say, V_A) by generating a histogram of V_A and looking at the points in the histogram corresponding to the bounds of the interval centered at the empirical mean of V_A and comprising 95% of the distribution.

Note that in (Bolle et al., 2004), the authors further modify the procedure to take into account possible dependencies between examples. Note furthermore that this technique has also been used recently in other research areas such as in automatic speech recognition where the measure of interest is the *word error rate* and is an aggregate of word insertions, deletions and substitutions between the target sentence and the obtained sentence (Bisani & Ney, 2004).

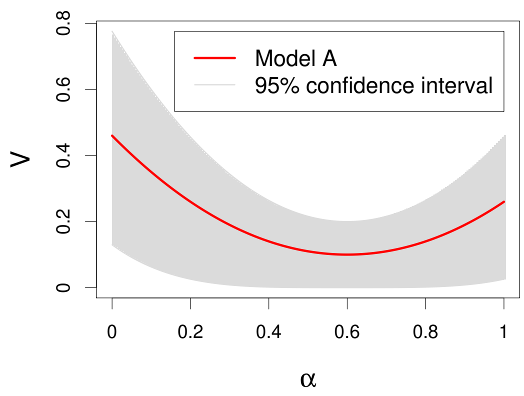

Figure 7. Example of an EPC and its corresponding 95% confidence interval

Going back to EPCs, one can now combine any EPC with a confidence interval by simply computing the interval for all possible values of α using the above technique, and the result is depicted in Figure 7. Note that the width of the interval will vary with respect to α, showing the importance of such graph. Alternatively, one can compute the statistical significance level of the difference between two models over a range of possible values of α, as shown in Figure 8 where we highlighted in gray the range of α for which the two models were statistically significantly different with 95% confidence.

6. Conclusion

In this paper, we have explained why the current use of ROCs in machine learning, as well as measures such as EER and BEP, used regularly in several publications related to domains such as person authentication, text categorization, or medical applications, can

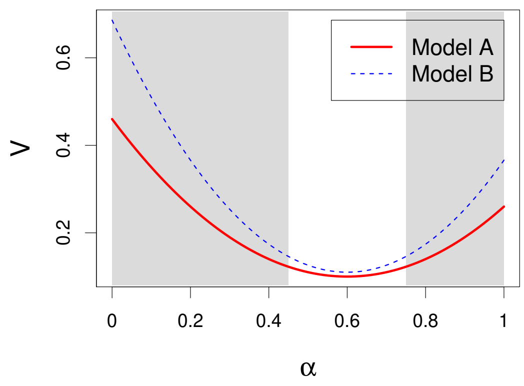

Figure 8. Example of two EPCs where we show in gray the regions where the difference between the two models is statistically significant with 95% confidence.

be misleading when used to compare performance between models or to assess the expected performance of a given model.

We have thus proposed the use of new curves called Expected Performance Curves (EPC), which reflect more precisely the criteria underlying the real application and therefore enable a more realistic comparison between models as well as a better analysis of their respective expected performance. From these curves, several single measures can also be obtained, and all of them should reflect a realistic performance comparison for a particular (and reachable) operating point of the system. Moreover, a summary measure, similar to the AUC, reflecting the expected performance of the system under a large range of reachable conditions, has also been proposed. Note that a free software is available to compute these curves and statistics (`http://www.torch.ch/extras/epc`).

Finally, we have proposed to link such EPCs with a non-parametric statistical test in order to show confidence intervals or statistical significant differences along a range of operating points.

It might be argued that one weakness of this new set of measures is the need for a separate development set. While this is true and necessary in order to obtain realistic expected performances, one could always rely on cross-validation techniques to solve this problem of a lack of training data.

ROCs can certainly still be used when the goal is to understand the behavior of a model without taking into account the selection of the threshold, however this should be done with caution, since it does not correspond to a real application setting.

Acknowledgements

This research has been partially carried out in the framework of the European PASCAL project, funded by the Swiss OFES. It was also partially funded by the Swiss NCCR project (IM)2.

References

Bengio, S., & Mariéthoz, J. (2004). The expected performance curve: a new assessment measure for person authentication. *Proceedings of Odyssey 2004: The Speaker and Language Recognition Workshop.*

Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-Garcia, J., Petrovsk-Delacrétaz, D., & Reynolds, D. (2004). A tutorial on text-independent speaker verification. *EURASIP Journal on Applied Signal Processing*, *4*, 430–451.

Bisani, M., & Ney, H. (2004). Bootstrap estimates for confidence intervals in ASR performance evaluation. *IEEE International Conference on Acoustic, Speech, and Signal Processing, ICASSP.*

Bolle, R., Ratha, N., & Pankanti, S. (2004). Error analysis of pattern recognition systems - the subsets bootstrap. *Computer Vision and Image Understanding*, *93*, 1–33.

Dietterich, T. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*, 1895–1924.

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

Green, D., & Swets, J. (1964). *Signal detection theory and psychophysics.* John Wiley & Sons.

Hanley, J., & McNeil, B. (1982). The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, *143*, 29–36.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, *34*, 1–47.

Verlinde, P., Chollet, G., & Acheroy, M. (2000). Multimodal identity verification using expert fusion. *Information Fusion*, *1*, 17–33.

Zweig, & Campbell (1993). ROC plots: a fundamental evaluation tool in clinical medicine. *Clinical Chemistry*, *39*, 561–577.