3 Performance Measures for Speaker Verification

Every time a researcher proposes a new idea or a new model to solve a given task, he needs to validate his approach using empirical data. In order to estimate the quality of a system, empirical measures such as numbers or curves are often used. They can be used for instance to estimate the expected performance on a new dataset coming from the same distribution as the one used to estimate the model, or to compare two different approaches.

In person authentication, several measures are commonly used as performance measures, such as equal error rate, half total error rate or detection cost functions. Even if the community made large efforts to make these measures standard in the speaker verification domain, for example during NIST evaluation (Martin and Przybocki, 2000), the published results in the scientific literature are most of the time optimistically biased. Too often, models are compared with some parameters estimated on the same examples as those used to estimate the performance measure. The estimation of these parameters are not trivial and the robustness of the models to the decision threshold for example, can be very variable. The machine learning framework proposes several tools to provide unbiased results, such as k-fold cross-validation or train - development - test set approaches. We will see in this chapter that this framework can be applied directly to performance measures such as half total error rate and also to new proposed curves called "expected performance curves".

Moreover, a single error value is difficult to assess without some form of confidence interval. In fact, as the quantity of available data to estimate the quality of a system is limited, the measures can vary depending on the size of the chosen dataset. It is thus important to give an interval around a given error,

REF A. Martin and M. Przybocki. The NIST 1999 speaker recognition evaluation - an overview. *Digital Signal Processing*, 10:1–18, 2000.

or a confidence value based on the hypothesis that two models are different, for example. Statistics provides tools such as proportion tests that can be used to compute these intervals. Strangely enough, almost nobody use this kind of tests in their research papers or if they do, the tests are often not correctly used. We thus provide a solution to apply a proportion test to the speaker verification domain.

The outline of this chapter goes as follows. In Section 3.1, we present the common measures in speaker verification and show their limitations. In Section 3.2, we present a new family of curves designed to compare systems. Section 3.3 is dedicated to the adaptation of the proportion test for speaker verification systems. Finally, in Section 3.4, we summarize the performance measures and the methodology used in this thesis.

3.1 Common Measures

A verification system has to deal with two kinds of events: either the person claiming a given identity is the one who he claims to be (in which case, he is called a *client*), or he is not (in which case, he is called an *impostor*). Moreover, the system may generally take two decisions: either *accept* the *client* or *reject* him and decide he is an *impostor*. From a machine learning point of view a client access can be labelled as 1 and an impostor as -1.

Let us thus 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 $\in \{-1, 1\}$, we are searching for a function $f(\cdot)$ and a threshold Δ such that

$$f(x_i) > \Delta \text{ when } y_i = 1 \text{ and } f(x_i) <= \Delta \text{ when } y_i = -1, \quad \forall i \; . \tag{3.1}$$

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

Table 3.1. Types of errors in a 2-class classification problem.

The obtained function $f(\cdot)$ (and associated threshold Δ) 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 3.1, in 4 different categories: *true positives* (where both the desired and the obtained classes are 1), *true negatives* (where both the desired and the obtained classes is 1), *false positives* (where the desired class is -1 and the obtained class is 1), and *false negatives* (where the desired class is 1 and the obtained class is -1). Let TP, TN, FP and FN represent respectively the *number of utterances* of each of the corresponding outcomes 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 Δ. In the following, we will sometimes refer to, say, FP by FP(Δ) in order to specifically show the dependency with the associated threshold.

In speaker verification, false positives and false negatives are respectively referred as *false acceptance* and *false rejection*.

Note that in most benchmark databases used in the authentication literature, there is a significant unbalance between the number of client accesses and the number of impostor accesses. This is probably due to the relatively higher cost of obtaining the former with respect to the latter. In order to be independent on the specific dataset distribution, the performance of the system is often measured in terms of rates of these two different errors, as follows:

$$\text{FAR} = \frac{\text{FP}}{\text{FP+TN}} = \frac{\text{FP}}{\text{NN}}\,, \quad \text{FRR} = \frac{\text{FN}}{\text{FN+TP}} = \frac{\text{FN}}{\text{NP}} \tag{3.2}$$

where NP is the number of true client (positive) examples, NN is the number of impostors (negative) examples, FAR is the false acceptance rate and FRR the false rejection rate. Based on these two kinds of errors, we need to define some measures to estimate the performance of a given system on unseen client and impostor accesses. These measures will be denoted hereafter "a posteriori" measures, when the decision threshold is set using the already seen examples and "a priori" measures when the decision threshold is set using unseen examples. The "a posteriori" measures should be used only for analysis purposes and not for comparison purposes.

A often used unique measure combines these two ratios into the so-called *detection cost function* (DCF) (Martin and Przybocki, 2000) as follows:

$$\text{DCF} = \begin{cases} \text{Cost(FN)} \cdot P(\text{client}) \cdot \text{FRR} \\ +\text{Cost(FP)} \cdot P(\text{impostor}) \cdot \text{FAR} \end{cases} \tag{3.3}$$

where P(client) is the prior probability that a client will use the system, P(impostor) is the prior probability that an impostor will use the system,

REF A. Martin and M. Przybocki. The NIST 1999 speaker recognition evaluation - an overview. *Digital Signal Processing*, 10:1–18, 2000.

Cost(FR) is the cost of a false rejection, and Cost(FA) is the cost of a false acceptance. These two costs depend on the application at hand.

A particular case of the DCF is known as the *half total error rate* (HTER) where the costs are equal to 1 and the probabilities are 0.5 each:

$$\text{HTER} = \frac{\text{FAR} + \text{FRR}}{2} \,. \tag{3.4}$$

Most authentication systems are measured and compared using HTER or variations of it.

In the literature, we also often encounter a measure called equal error rate (EER) which corresponds to the threshold nearest to a solution such that FAR = FRR, often estimated as follows:

$$\Delta^\star = \arg\min_{\Delta} |\text{FAR}(\Delta) - \text{FRR}(\Delta)| \text{ and } \text{EER} = \text{FAR}(\Delta) = \text{FRR}(\Delta). \tag{3.5}$$

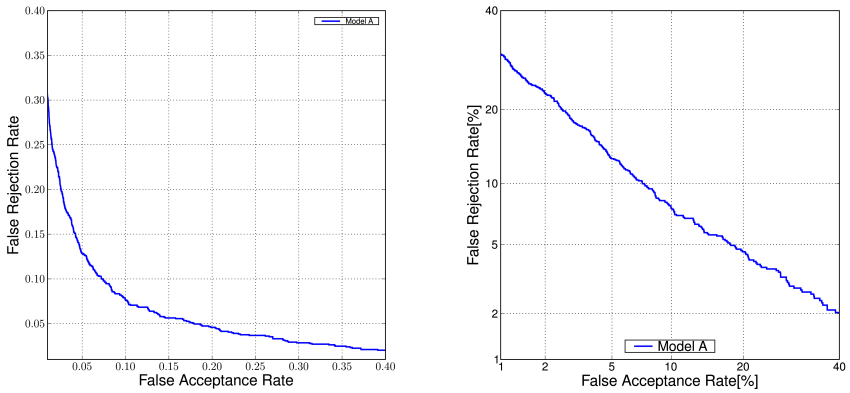
One has to note that this measure is an "a posteriori" measure and should only be used as a criterion to select a decision threshold and not to compare systems, because the exact decision threshold value that reaches the equal error rate in test (unseen) data cannot be known in advance. Only an estimation of it can be found and $\text{FAR}_{test} \neq \text{FRR}_{test}$. Often HTER and EER are similar and both measures are often used as criterion to select the threshold. However, as HTER can fall in a local minimum, EER seems to be more robust and will thus be used in the following.

In most cases, the system can be tuned using a decision threshold in order to obtain a compromise between either a small FAR or a small FRR. There is thus a trade-off which depends on the application: it might sometimes be more important to have a system with a very small FAR, for high security systems, while in other situations it might be more important to have a system with a small FRR, for domestic applications such as games for example. In order to see the performance of a system with respect to this trade-off, we usually plot the so-called Receiver Operating Characteristic (ROC) curve, which represents the FRR as a function of the FAR (Van Trees, 1968) (hence, the curve which is nearer the $(0, 0)$ coordinate is the best ROC curve). Figure 3.1(a) shows an example of a typical ROC. Other researchers have also proposed the DET curve (Martin et al., 1997), which is a non-linear transformation of the ROC

REF H. L. Van Trees. *Detection, Estimation and Modulation Theory, vol. 1.* Wiley, New York, 1968.

REF A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki. The DET curve in assessment of detection task performance. In *Proceedings of Eurospeech'97, Rhodes, Greece*, pages 1895–1898, 1997.

curve in order to make results easier to be compared. The non-linearity is in fact a normal deviate, coming from the hypothesis that the scores of client accesses and impostor accesses follow a Gaussian distribution. If this hypothesis is true, the DET curve should be a line. Figure 3.1(b) shows an example of typical DET curve. Note that Figures 3.1(a) and 3.1(b) are computed for the same system. As we will see in the following, these curves make the implicit assumption that the decision threshold estimation is perfect. We can say that these curves are somehow "a posteriori" curves and thus cannot be use to compare two systems; we thus propose instead a new kind of curve, called expected performance curves.



(a) A typical ROC curve.

(b) A typical DET curve.

Figure 3.1. Comparison between DET and ROC curve for the same system.

3.2 Expected Performance Curve

ROC curves are used in several domains, such as text categorization, biometric authentication, medical studies, etc. To be domain independent we need to redefine in a general framework the measures used in these domains.

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 as defined in Table 3.1. Moreover, a unique measure (V) often combines $V1$ and $V2$. For instance,

- in the domain of person authentication (Verlinde et al., 2000) as we have already seen, the chosen measures are

$$V1 = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \text{ and } V2 = \frac{\mathrm{FN}}{\mathrm{FN} + \mathrm{TP}}. \tag{3.6}$$

  Several aggregate measures have been proposed, the simplest being the (HTER)

$$V = \frac{V1 + V2}{2} = \frac{\mathrm{FAR} + \mathrm{FRR}}{2} = \mathrm{HTER}\ ; \tag{3.7}$$

- in the domain of text categorization (Sebastiani, 2002),

$$V1 = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \text{ and } V2 = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3.8}$$

  and are called *precision* and *recall* respectively. Again several aggregate measures exist, such as the *F1* measure

$$V = \frac{2 \cdot V1 \cdot V2}{V1 + V2} = \frac{2 \cdot \mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} = F1\ ; \tag{3.9}$$

- in medical studies,

$$V1 = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \text{ and } V2 = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{3.10}$$

  and are called *sensitivity* and *specificity* respectively (Zweig and Campbell, 1993).

In all the cases, in order to use the system effectively, one has to select the threshold Δ according to some criterion which is in general of the following generic form

$$\Delta^\star = \arg\min_{\Delta} g(V1(\Delta), V2(\Delta))\ . \tag{3.11}$$

Examples of $g(\cdot,\cdot)$ are the HTER and *F1* functions already defined in equations (3.7) and (3.9) respectively. However, the most used criterion is called the *break even point* (BEP) also sometimes called equal error rate (EER) when $V1$ and $V2$ are error rates and corresponds to the threshold nearest to a solution such that $V1 = V2$, often estimated as follows:

$$\Delta^\star = \arg\min_{\Delta} |\mathrm{V1}(\Delta) - \mathrm{V2}(\Delta)|\ . \tag{3.12}$$

REF P. Verlinde, G. Chollet, and M. Acheroy. Multi-modal identity verification using expert fusion. *Information Fusion*, 1:17–33, 2000.

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

REF M.H. Zweig and G. Campbell. ROC plots: a fundamental evaluation tool in clinical medicine. *Clinical Chemistry*, 39(4):561–577, 1993.

Note that the choice of the threshold can have a significant impact in the resulting system: in general Δ represents a trade-off between giving importance to $V1$ or $V2$. Hence, instead of committing to a single operating point, an alternative method is to present results by using ROCs. 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 3.2 shows an example of two ROC curves. 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 3.2, the best curve corresponds to the one nearest to the bottom left corner (corresponding to simultaneous small values of $V1$ and $V2$).

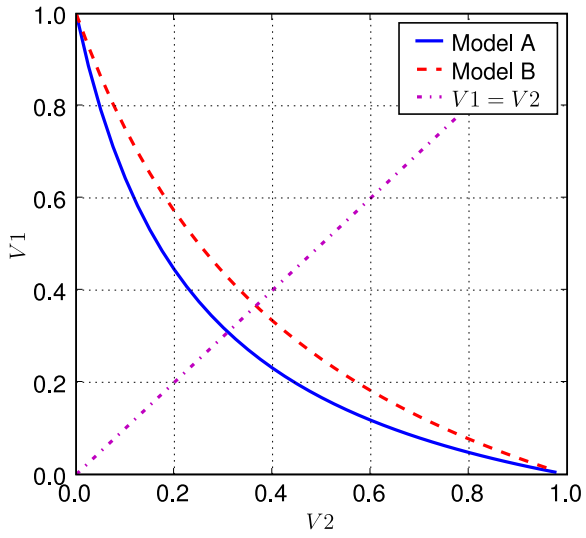


Figure 3.2. Example of two ROC curves with the BEP line.

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 defined in equation (3.12), which produces a single value of Δ and to produce some aggregate value $V(\Delta)$ (such as *F1* or HTER). On Figure 3.2, the line intersecting the two ROCs is the BEP line and the intersections with each ROC correspond to their respective BEP point.

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, 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 3.2, 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 Δ. However, in real applications, Δ needs to be decided prior to seeing the test set. This is in general done using some criterion of the form of equation (3.11) such as searching for the BEP, equation (3.12), using some development data (obviously different from the test set).

Hence, assuming for instance that one decided to select the threshold according to (3.12) 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 (3.12) on one set is not guaranteed to be the same as the estimate on the other set.

Let us call $\Delta_A^\star$ the threshold estimated on the development set using model A and similarly for $\Delta_B^\star$. 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 3.3 shows such an example, where indeed,

$$V1(\Delta_B^\star) + V2(\Delta_B^\star) < V1(\Delta_A^\star) + V2(\Delta_A^\star) \tag{3.13}$$

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

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

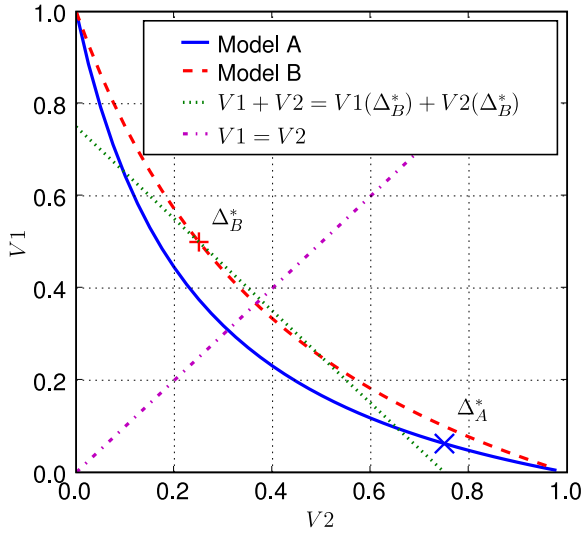
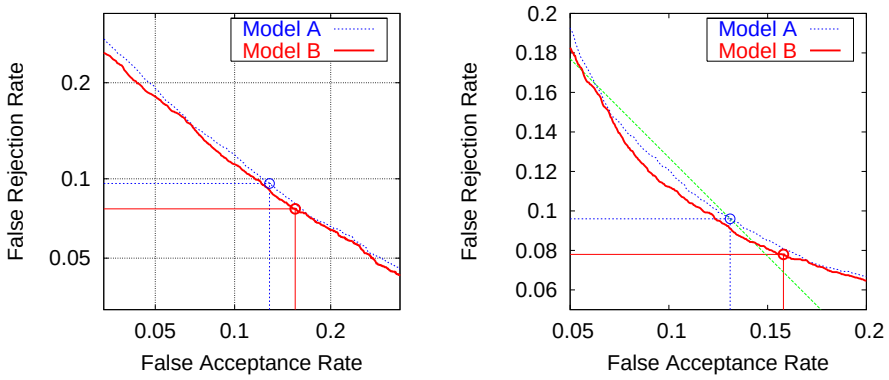


Figure 3.3. Two ROC curves of two different models with their own decision threshold learnt by minimizing the BEP.

even though the ROC of model A is always below that of model B, including at the intersection with the BEP line. 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 and Mariéthoz, 2004). We replicate in the right side of Figure 3.4 the ROCs and in the left side, the DETs 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 world life situation), the HTER of model A (0.111) becomes lower than that of model B (0.112), 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). One can expect that it reflects the expected generalization error, but this is true when the size of the data is huge, and false in the general case. Furthermore, real applications often suffer from an addi-

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



(a) DET curves. (b) ROCs curves.

Figure 3.4. Curves of two real models for a Text-Independent Speaker Verification task with their corresponding "a priori"operating points.

tional mismatch between training and test conditions which should be reflected in the used measure.

Expected Performance Curve: an "a priori" Performance Curve

We have seen in Section 3.1 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 that ROCs can be misleading since selecting a threshold prior to seeing the test set (as it should be done) may end up in obtaining 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).

General Framework

The general framework of EPCs is the following. Let us define some parametric performance measure $\mathcal{C}(V1(\Delta, D), V2(\Delta, D); \gamma)$ which depends on a trade-off parameter γ as well as $V1$ and $V2$ computed on some data D for a particular value of the decision threshold Δ. Examples of $\mathcal{C}(\cdot, \cdot; \gamma)$ are the following:

- in person authentication, one could use for instance

$$\begin{aligned}\mathcal{C}(V1(\Delta, D), V2(\Delta, D); \gamma) \\ = \mathcal{C}(\text{FAR}(\Delta, D), \text{FRR}(\Delta, D); \gamma) \\ = \gamma \cdot \text{FAR}(\Delta, D) + (1 - \gamma) \cdot \text{FRR}(\Delta, D)\end{aligned} \tag{3.14}$$

  which basically varies the relative importance of $V1$ (FAR) with respect to $V2$ (FRR); in fact, setting $\gamma = 0.5$ yields the HTER cost (3.7);

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

$$\mathcal{C}(V1(\Delta, D), V2(\Delta, D); \gamma) \tag{3.15}$$

$$= \mathcal{C}(\text{Precision}(\Delta, D), \text{Recall}(\Delta, D); \gamma)$$

$$= -(\gamma \cdot \text{Precision}(\Delta, D) + (1 - \gamma) \cdot \text{Recall}(\Delta, D)) \tag{3.16}$$

  where $V1$ is the precision and $V2$ is the recall; notice the negative sign in 3.16 as precision and recall are penalty measures and instead of costs.

- 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

$$\mathcal{C}(V1(\Delta, D), V2(\Delta, D); \gamma) = |\gamma - V1(\Delta, D)| \tag{3.17}$$

  or

$$\mathcal{C}(V1(\Delta, D), V2(\Delta, D); \gamma) = |\gamma - V2(\Delta, D)| \,. \tag{3.18}$$

Having defined $\mathcal{C}(\cdot, \cdot; \gamma)$, the main procedure to generate the EPC is to vary γ inside a reasonable range (say, from 0 to 1), and for each value of γ, to estimate Δ that minimizes $\mathcal{C}(\cdot, \cdot; \gamma)$ on a development set, and then use the obtained Δ to compute some aggregate value (say, V), on the test set. Algorithm 3.1 details the procedure, while Figure 3.5 shows an artificial example of comparing the EPCs of two models. Looking at this figure, we can now state that for specific values of γ (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 3.5 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 (3.14). In that case, γ varies from 0 to 1,

Algorithm 3.1 Method to generate the Expected Performance Curve

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

and when $\gamma = 0.5$ 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 $\gamma < 0.5$ it corresponds to settings where we gave more importance to false rejection errors and when $\gamma > 0.5$ we gave more importance to false acceptance errors.

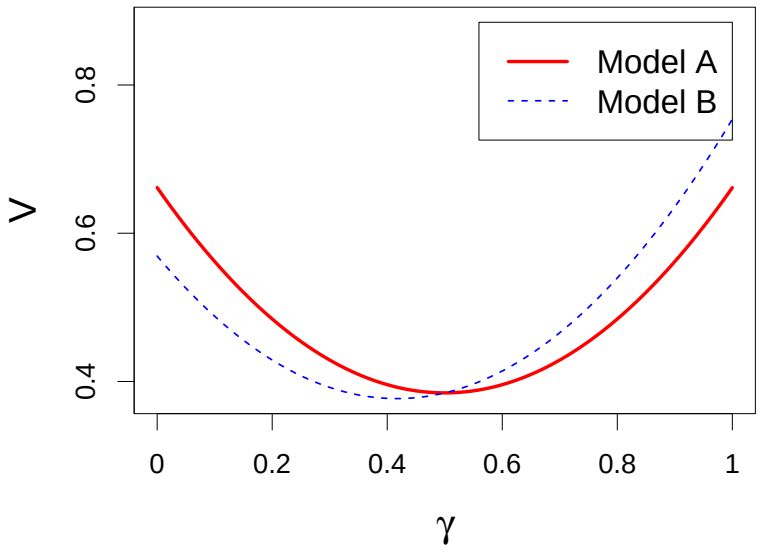


Figure 3.5. Example of two theoretical EPCs.

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

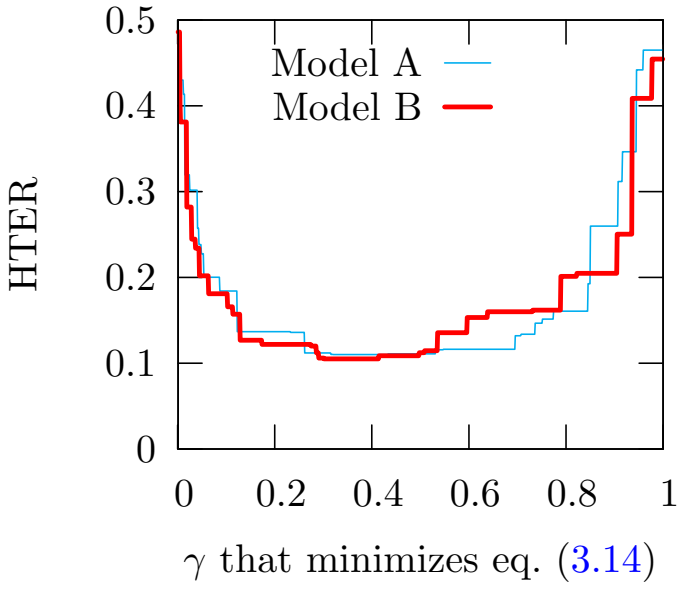


Figure 3.6. Expected Performance Curves for person authentication, where one wants to trade-off false acceptance rates with false rejection rates.

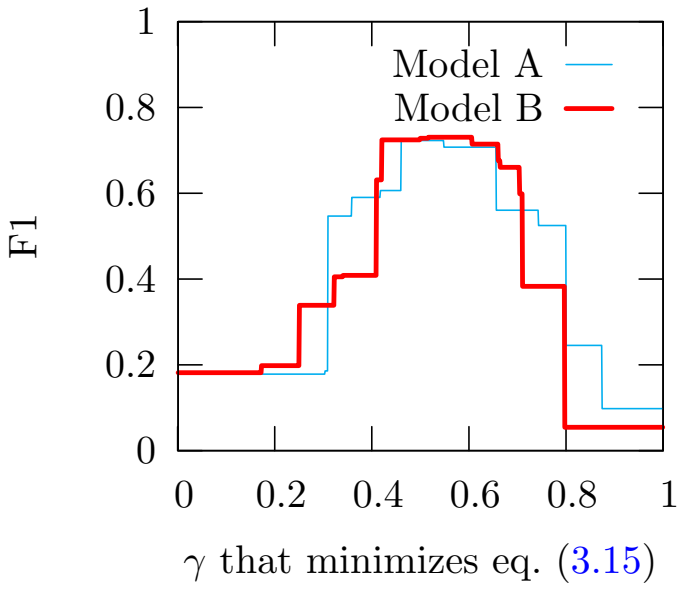


Figure 3.7. Expected Performance Curves for text categorization, where one wants to trade-off precision and recall and print the $F1$ measure.

curves can be seen in Figures 3.6 and 3.7. 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 3.6 corresponds to the ROC and DET of Figure 3.4. Finally, note that we kindly provide a C++ tool that generates such EPCs. An EPC generator is available at `http://www.Torch.ch/extras/epc` as a package of the Torch machine learning library.

To compare the performance of two systems, we can use either numbers such as HTER with a decision threshold estimated "a priori" or curves such as EPC. Unfortunately, this might not be enough; as an error may be meaningless if no confidence interval is given. In "biometric authentication", measures such as HTER are used instead of the classification error, thus, as will be shown in the next section, usual techniques to estimate the confidence interval cannot be used as is. We thus propose an adaptation of the z-test for speaker verification systems that can be applied to numbers such as HTER, DCF and also to EPCs.

3.3 Statistical Tests

Whenever one researcher wants to compare a novel model to an existing solution, using either one value such as HTER or using a curve such as EPC, a quick review of the current literature in person authentication shows that either no statistical test is used to assess the difference between models, or, worse, statistical tests are used incorrectly, which often ends up in over-optimistic results, tending to show, for instance, that the new model is statistically significantly better than the state-of-the-art while it might not be the case in fact.

In this section, we present a proper method to compute a simple statistical test, known as the *test of two proportions*, or *z-test*, adapted to the problem of aggregate measures such as HTER and DCF.

The Z-Test on Proportions

Several statistical tests are available in the literature. For standard classification tasks, a simple yet often used test is known as the *z-test*, or *test between two proportions*. The rationale of this test is the following: given a set of n examples, each drawn independently and identically distributed (i.i.d.) from an unknown distribution, a given system is going to take a decision for each example, and this decision will be correct or not. Let us now look at the distribution of the number of errors that will be made by the classification system. Since each decision is independent from the others and is binary, it is reasonable to

assume that the random variable $\mathbf{X}$ representing the number of errors should follow a *Binomial* distribution $\mathcal{B}(n, p)$ where n is the number of examples and p is the percentage of errors. In this section we use the following notation: bold letters such as **FA** represent random variables, while normal letters such as FA represent a particular value of the underlying random variable.

Moreover, it is known that a Binomial $\mathcal{B}(n, p)$ can be approximated by a Normal distribution $\mathcal{N}(\mu, \sigma^2)$ with

$$\mu = np \quad \text{and} \quad \sigma^2 = np(1-p)$$

when n is large enough. A rule of thumb often used is to have $np(1-p)$ larger than 10.

Finally, if $\mathbf{X} \sim \mathcal{N}(np, np(1-p))$, then the distribution of the proportion of errors $\mathbf{Y} = \frac{\mathbf{X}}{n} \sim \mathcal{N}\left(p, \frac{p(1-p)}{n}\right)$.

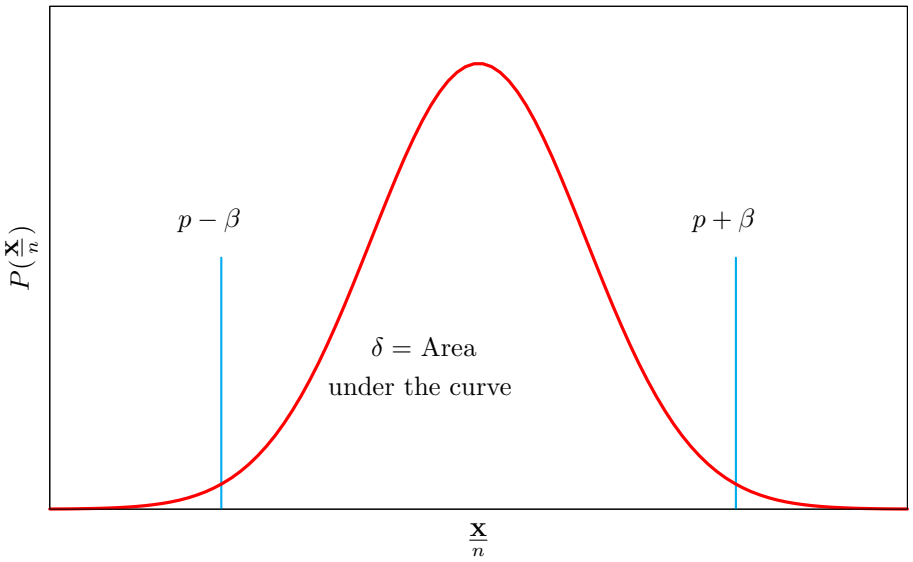


Figure 3.8. Confidence intervals are computed using the area under the Normal curve.

Confidence Intervals

In order to compute a confidence interval around p, we can search for bounds $\{p - \beta, p + \beta\}$ such that

$$P\left(p - \beta < \mathbf{Y} < p + \beta\right) = \delta \tag{3.19}$$

where δ represents our confidence. This is called a *two-sided* test since we are searching for two bounds around p. Fortunately, finding β in (3.19) for a given δ can be done efficiently for the Normal distribution. Figure 3.8 illustrates graphically the problem.

Difference Between Proportions

Alternatively, if one wants to verify whether a given proportion of errors p_A is statistically significantly different from another proportion p_B, a similar test can be performed. In the case where we already know that p_A cannot be lower than p_B, a *one-sided* test is used, otherwise we use a *two-sided* test. Noting respectively $\mathbf{Y}_A$ and $\mathbf{Y}_B$ the random variables representing the distribution of p_A and p_B, the *one-sided* test is based on

$$P(\mathbf{Y}_A - \mathbf{Y}_B < p_A - p_B) = \delta \tag{3.20}$$

while the *two-sided* test is based on

$$P(|\mathbf{Y}_A - \mathbf{Y}_B| < |p_A - p_B|) = \delta \tag{3.21}$$

which can be solved using the fact that the difference between two independent Normal distributions is a Normal distribution where the mean is the difference between the two Normal means and the variance is the sum of the two Normal variances, hence, if $\mathbf{Y}_A$ is not statistically different from $\mathbf{Y}_B$, then

$$\mathbf{Y}_A - \mathbf{Y}_B \sim \mathcal{N}\left(p_A - p_B, \frac{p_A(1-p_A) + p_B(1-p_B)}{n}\right) \tag{3.22}$$

and if δ is higher than a predefined value (such as 95%), then one can state that p_A is significantly different from p_B. Note that a better estimate of the variance of (3.22) can be obtained when assuming $p_A = p_B$ (which should be the case if they are not significantly different). In that case, equation (3.22) becomes

$$\mathbf{Y}_A - \mathbf{Y}_B \sim \mathcal{N}\left(0, \frac{2p(1-p)}{n}\right) \tag{3.23}$$

with

$$p = \frac{p_A + p_B}{2} \,.$$

Note however that using this test to verify whether two models give statistically significantly different results on the same test database makes a wrong hypothesis, since $\mathbf{Y}_A$ and $\mathbf{Y}_B$ are not really independent as they correspond to decisions taken on *the same test set*.

Dependent Case

One possible solution proposed in (Snedecor and Cochran, 1989) is to only take into account the examples for which the two models disagree. Let p_{AB} be the proportion of examples correctly classified by model A and incorrectly classified by model B, and similarly p_{BA} be the proportion of examples correctly classified by model B and incorrectly classified by model A. In that case, the distribution $\mathbf{Y}_{|A-B|}$ of the difference between the proportions of errors committed by each model is still Normally distributed and, assuming the two models are not different from each other, should follow

$$\mathbf{Y}_{|A-B|} \sim \mathcal{N}\left(0, \frac{p_{AB} + p_{BA}}{n}\right) \tag{3.24}$$

with the corresponding two-sided test

$$P(\mathbf{Y}_{|A-B|} < |p_{AB} - p_{BA}|) = \delta \; . \tag{3.25}$$

This test is in fact very similar to the well-known McNemar test, based on a χ^2 distribution.

In the literature, most people adopt equation (3.23) and some adopt equation (3.24); remember that in order to use equation (3.24), one needs to have access to all the scores of both models, and not just the numbers of errors. When possible, we will look at both solutions here, for the case of person authentication.

$\mathbf{Z_{HTER}}$-Test: a Statistical Test for HTERs

HTERs are not proportions, but they are an average of two well-defined proportions (FAR and FRR). In the following, we propose to extend the test between two proportions for the case of HTERs. We assume the distributions of FAR and FRR independent. This may look false since they are both linked by the same model and threshold, but in fact, *given a model and associated threshold* these two quantities are indeed most probably independent since they are computed on separate data (the client accesses and the impostor accesses), assuming the model was estimated on a separate training set, as it should be.

Confidence Intervals

Let the random variable **FP** represent the number of false positive. We can model it by a Binomial, and hence by a Normal, as follows:

REF G. W. Snedecor and W. G. Cochran. *Statistical Methods*. Iowa State University Press, 1989.

$$\begin{aligned}\mathbf{FP} &\sim \mathcal{B}\left(\mathrm{NN}, \frac{\mathrm{FP}}{\mathrm{NN}}\right) \\ &\sim \mathcal{N}\left(\mathrm{NN}\cdot\frac{\mathrm{FP}}{\mathrm{NN}}, \mathrm{NN}\cdot\frac{\mathrm{FP}}{\mathrm{NN}}\cdot\left(1-\frac{\mathrm{FP}}{\mathrm{NN}}\right)\right) \\ &\sim \mathcal{N}\left(\mathrm{FP}, \mathrm{FP}\cdot(1-\mathrm{FAR})\right) \ . \end{aligned} \tag{3.26}$$

The random variable **FN** representing the number of false negative can be modeled accordingly:

$$\begin{aligned}\mathbf{FN} &\sim \mathcal{B}\left(\mathrm{NP}, \frac{\mathrm{FN}}{\mathrm{NP}}\right) \\ &\sim \mathcal{N}\left(\mathrm{NP}\cdot\frac{\mathrm{FN}}{\mathrm{NP}}, \mathrm{NP}\cdot\frac{\mathrm{FN}}{\mathrm{NP}}\cdot\left(1-\frac{\mathrm{FN}}{\mathrm{NP}}\right)\right) \\ &\sim \mathcal{N}\left(\mathrm{FN}, \mathrm{FN}\cdot(1-\mathrm{FRR})\right) \ . \end{aligned} \tag{3.27}$$

We can now write the distribution of the random variable **FAR** representing the ratio of false acceptances:

$$\begin{aligned}\mathbf{FAR} &\sim \mathcal{N}\left(\frac{\mathrm{FP}}{\mathrm{NN}}, \frac{\mathrm{FP}\,(1-\mathrm{FAR})}{\mathrm{NN}\cdot\mathrm{NN}}\right) \\ &\sim \mathcal{N}\left(\mathrm{FAR}, \frac{\mathrm{FAR}\,(1-\mathrm{FAR})}{\mathrm{NN}}\right) \end{aligned} \tag{3.28}$$

and similarly for the random variable **FRR**:

$$\begin{aligned}\mathbf{FRR} &\sim \mathcal{N}\left(\frac{\mathrm{FN}}{\mathrm{NP}}, \frac{\mathrm{FN}\,(1-\mathrm{FRR})}{\mathrm{NP}\cdot\mathrm{NP}}\right) \\ &\sim \mathcal{N}\left(\mathrm{FRR}, \frac{\mathrm{FRR}\,(1-\mathrm{FRR})}{\mathrm{NP}}\right) \end{aligned} \tag{3.29}$$

Given the distribution of **FAR** and **FRR**, we can estimate the distribution of the random variable **HTER** as follows:

$$\begin{aligned}\mathbf{FAR{+}FRR} &\sim \mathcal{N}\left(\mathrm{FAR{+}FRR}, \frac{\mathrm{FAR}\,(1-\mathrm{FAR})}{\mathrm{NN}} + \frac{\mathrm{FRR}\,(1-\mathrm{FRR})}{\mathrm{NP}}\right) \\ \frac{\mathbf{FAR{+}FRR}}{2} &\sim \mathcal{N}\left(\frac{\mathrm{FAR{+}FRR}}{2}, \frac{\mathrm{FAR}\,(1-\mathrm{FAR})}{4\cdot\mathrm{NN}} + \frac{\mathrm{FRR}\,(1-\mathrm{FRR})}{4\cdot\mathrm{NP}}\right) \\ \mathbf{HTER} &\sim \mathcal{N}\left(\mathrm{HTER}, \frac{\mathrm{FAR}(1-\mathrm{FAR})}{4\cdot\mathrm{NN}} + \frac{\mathrm{FRR}(1-\mathrm{FRR})}{4\cdot\mathrm{NP}}\right) \end{aligned} \tag{3.30}$$

Using this last definition, we can now compute easily confidence intervals around HTERs using the methodology summarized in Figure 3.9 for classical confidence values used in the scientific literature.

Moreover, the test can be easily extended to variations of HTER, such as the DCF in (3.3). For instance, in the case of the well-known NIST evaluations performed yearly to compare speaker verification systems, and which use the DCF measure described by equation (3.3) with Cost(FR) = 10, P(client) = 0.01, Cost(FA) = 1 and P(impostor) = 0.99, the underlying Normal becomes:

$$\mathbf{DCF} \sim \mathcal{N}\left(\mathrm{DCF}, \frac{\mathrm{FAR}\,(1-\mathrm{FAR})}{0.99^{-2}\cdot \mathrm{NN}} + \frac{\mathrm{FRR}\,(1-\mathrm{FRR})}{100\cdot \mathrm{NP}}\right) . \tag{3.31}$$

Difference Between HTERs

The distribution of the difference between two HTERs assuming *independence* between the two underlying distributions is

$$\mathbf{HTER}_A - \mathbf{HTER}_B \sim \mathcal{N}\left(0, \sigma^2_{\mathbf{INDEP}}\right) \tag{3.32}$$

with

$$\sigma^2_{\mathbf{INDEP}} = \begin{cases} \dfrac{\mathrm{FAR}_A\,(1-\mathrm{FAR}_A) + \mathrm{FAR}_B\,(1-\mathrm{FAR}_B)}{4\cdot \mathrm{NN}} \\ + \dfrac{\mathrm{FRR}_A\,(1-\mathrm{FRR}_A) + \mathrm{FRR}_B\,(1-\mathrm{FRR}_B)}{4\cdot \mathrm{NP}} \end{cases}$$

while the distribution of the difference between two HTERs assuming *dependence* between the two underlying distributions becomes

$$\mathbf{HTER}_A - \mathbf{HTER}_B \sim \mathcal{N}\left(0, \sigma^2_{\mathbf{DEP}}\right) \tag{3.33}$$

with

$$\sigma^2_{\mathbf{DEP}} = \frac{\mathrm{FAR}_{AB} + \mathrm{FAR}_{BA}}{4\cdot \mathrm{NN}} + \frac{\mathrm{FRR}_{AB} + \mathrm{FRR}_{BA}}{4\cdot \mathrm{NP}}$$

where $\mathrm{FAR}_{AB} = \frac{\mathrm{NN}_{AB}}{\mathrm{NN}}$ and NN_{AB} is the number of impostor accesses correctly rejected by model A and incorrectly accepted by model B, with similar definitions for FAR_{BA}, FRR_{AB}, and FRR_{BA}.

Hence, in summary, and using the standard confidence values used in the scientific literature, we obtain the simple methodology described in Figure 3.9 in order to compute statistical tests for person authentication tasks. Figure 3.9 represents a two-sided test and we thus use $Z_\alpha/2$ instead of Z_α. While this summary concerns HTERs, it should now be obvious to extend it to the general DCF function.

The confidence interval (CI) around an HTER is $\text{HTER} \pm \sigma \cdot Z_{\alpha/2}$ with

$$\sigma = \sqrt{\frac{\text{FAR}(1-\text{FAR})}{4 \cdot \text{NN}} + \frac{\text{FRR}(1-\text{FRR})}{4 \cdot \text{NP}}}$$

$$Z_{\alpha/2} = \begin{cases} 1.645 & \text{for a 90\% CI} \\ 1.960 & \text{for a 95\% CI} \\ 2.576 & \text{for a 99\% CI} \end{cases}$$

and similarly, HTER_A and HTER_B are statistically significantly different if $z > Z_{\alpha/2}$ with

$$z = \frac{|\text{HTER}_A - \text{HTER}_B|}{\sqrt{\frac{\text{FAR}_A(1-\text{FAR}_A) + \text{FAR}_B(1-\text{FAR}_B)}{4 \cdot \text{NN}} + \frac{\text{FRR}_A(1-\text{FRR}_A) + \text{FRR}_B(1-\text{FRR}_B)}{4 \cdot \text{NP}}}}$$

in the independent case, and

$$z = \frac{|\text{FAR}_{AB} - \text{FAR}_{BA} + \text{FRR}_{AB} - \text{FRR}_{BA}|}{\sqrt{\frac{\text{FAR}_{AB} + \text{FAR}_{BA}}{4 \cdot \text{NN}} + \frac{\text{FRR}_{AB} + \text{FRR}_{BA}}{4 \cdot \text{NP}}}}$$

in the dependent case.

Figure 3.9. Methodology for statistical tests around HTERs for a two-sided test.

Other Statistical Tests

While several researchers have pointed out the use of the *z-test* to compute statistical tests around values such as FAR or FRR, see for instance (Wayman, 1999), we are not aware, to the best of our knowledge, of any similar attempt for aggregate measures such as HTERs (or EER, or DCF). However, most people publishing results in verification use HTERs or DCF to assess the quality of their methods.

One simple solution could be to consider the classification error instead of

REF J.L. Wayman. Confidence interval and test size estimation for biometric data. In *Proceedings of the IEEE AutoID Conference*, 1999.

the HTER and compute statistical tests around it. Since the classification error is a well-defined proportion, we can apply the *z-test* as well; Let **CLASS** be defined as the following random variable:

$$\mathbf{CLASS} = \frac{\mathbf{FP{+}FN}}{\mathrm{NP{+}NN}}$$

then, the corresponding underlying Normal becomes:

$$\mathbf{CLASS} \sim \mathcal{N}\left(\frac{\mathrm{FP{+}FN}}{\mathrm{NP{+}NN}}, \frac{\mathrm{FP{+}FN}}{(\mathrm{NP{+}NN})^2}\left(1 - \frac{\mathrm{FP{+}FN}}{\mathrm{NP{+}NN}}\right)\right) \tag{3.34}$$

but remember that while this test is correct to assess models according to their respective classification error, it does not say anything on the confidence one has over the corresponding HTER, which is the measure of interest in person authentication. In fact, we will show in the next section that, under reasonable assumptions, the variance of **CLASS** in equation (3.34) is always smaller than the variance of **HTER** in equation (3.30), hence confidence tests using (3.34) will always result in over-confident statistical significance (or smaller confidence intervals). This will be explored further in the following section.

Another possible solution is to consider the HTER itself as a proportion (which it is not directly) and compute the statistical test on it. Let **NAIVE** be the random variable of this value; the underlying Normal becomes:

$$\mathbf{NAIVE} \sim \mathcal{N}\left(\mathrm{HTER}, \frac{\mathrm{HTER}(1 - \mathrm{HTER})}{\mathrm{NP{+}NN}}\right) \tag{3.35}$$

Again, we will show in next section that under reasonable assumptions, the variance of **NAIVE** in equation (3.35) is always smaller than the variance of **HTER** in equation (3.30), hence confidence tests using (3.34) should always result in over-confident statistical significance (or smaller confidence intervals).

Yet another solution that has been proposed by some researchers, see for instance (Koolwaaij, 2000), is to compute a statistical test for FAR and FRR separately and then combine the results. The well-known NIST evaluation campaigns have also apparently recently investigated the use of the McNemar test to assess speaker verification methods, but have considered separately FARs and FRRs (Martin, 2004). For instance, in order to compute a confidence interval for HTER, one would average both upper bounds and both lower bounds found separately by the FAR and FRR tests. On top of the fact that there is

REF J. Koolwaaij. *Automatic Speaker Verification in Telephony: a probabilitic approach.* PrintPartners Ipskamp B.V., Enschede, 2000.

REF A Martin. Personal communication. http://www.nist.gov/speech/staff/martinal.htm, 2004.

no theoretical ground to justify such an approach, there is an evident problem with all approaches that consider separately FARs and FRRs. Two models could yield very similar HTERs but for some reason (linked to the choice of the threshold, which should be selected on a separate data set) one could be slightly biased toward FRRs and the other one slightly biased toward FARs. In such a case, these tests would consider them statistically significantly different while they would not be when considering globally their respective HTER instead. For this reason, we will not consider this solution further here.

Analysis

We would like to compare in this section the use of the $\mathbf{Z_{HTER}}$-test with respect to the two other Class and Naive tests presented in the previous section. We will first show that under some reasonable conditions, increasing the ratio between NN and NP will increase the difference between the variance of the Normal of the $\mathbf{Z_{HTER}}$-test and the variance of the Normal of the other tests. Afterwards, we present two real case studies where the use of the $\mathbf{Z_{HTER}}$-test would have yielded a different conclusion with regard to the confidence intervals and the difference between the compared models.

Theoretical Analysis

Let us first look in which conditions $\sigma^2(3.30)$, the variance of **HTER** as written in equation (3.30) is higher than $\sigma^2(3.35)$, the variance of **NAIVE** as written in equation (3.35):

$$\sigma^2(3.30) > \sigma^2(3.35) \tag{3.36}$$

implies that

$$\frac{\mathrm{FAR}\,(1-\mathrm{FAR})}{4\,\mathrm{NN}} + \frac{\mathrm{FRR}\,(1-\mathrm{FRR})}{4\,\mathrm{NP}} > \frac{\mathrm{HTER}(1-\mathrm{HTER})}{\mathrm{NP}+\mathrm{NN}} \tag{3.37}$$

and assuming FAR is similar than FRR (again, when the threshold is chosen such that we have equal error rate (EER) on a separate validation set, as it is often done, this is reasonable), which can be simplified and yields

$$1 > \frac{1}{\mathrm{NP}+\mathrm{NN}} \tag{3.38}$$

which means that inequation (3.36) is always true under the assumption that FAR = FRR.

Let us now look in which conditions $\sigma^2(3.30)$ is higher than $\sigma^2(3.34)$, the variance of **CLASS**, representing the classification error:

$$\sigma^2(3.30) > \sigma^2(3.34) \tag{3.39}$$

implies that

$$\frac{\text{FAR}(1-\text{FAR})}{4 \cdot \text{NN}} + \frac{\text{FRR}(1-\text{FRR})}{4 \cdot \text{NP}} > \frac{\text{FP+FN}}{(\text{NP+NN})^2} \cdot (1 - \frac{\text{FP+FN}}{\text{NP+NN}})$$

and assuming FAR is similar to FRR, it can be simplified into

$$1 > \frac{1}{\text{NP+NN}} \tag{3.40}$$

which is true as long as FAR = FRR. Note that (3.38) is equal to 3.40, because $\sigma^2(3.35) = \sigma^2(3.34)$ when FAR = FRR.

In order to verify these relations graphically, we have fixed some variables to reasonable values (FAR = 0.1, FRR = 0.2, NP = 100) and have varied NN, the number of impostor accesses. Figure 3.10 shows the relation between the standard deviation of the underlying Normal distributions and the ratio between NN and NP.

As expected, the higher the ratio $\frac{\text{NN}}{\text{NP}}$, the bigger the difference between the standard deviation of the Normal distributions related to the three statistical tests. Moreover, we see that the standard deviation of the $\mathbf{Z_{HTER}}$-test distribution stays close to the one of the **FRR** distribution, which is mostly influenced by NP, the number of client accesses, and does not decrease with the increase of NN, contrary to the two other solutions. Since the size of the confidence interval is directly related to the standard deviation, this figure essentially shows that the confidence interval computed using the $\mathbf{Z_{HTER}}$-test will always be larger than that of the two other techniques. Hence two verification methods yielding two different HTERs could easily be considered statistically significantly different using one of the Class or Naive methods, while they would not be considered statistically significantly different using the $\mathbf{Z_{HTER}}$-test technique. In fact, the figure shows that the confidence interval is directly influenced by the minimum of NP and NN and not their sum.

In the next two subsections, we present two real case studies where the use of the $\mathbf{Z_{HTER}}$ statistical test would have yielded a different conclusion.

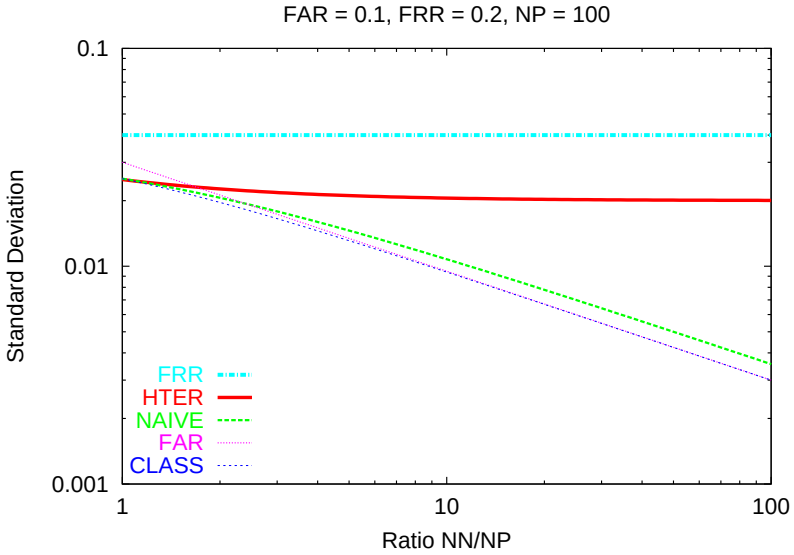


Figure 3.10. Standard deviation of the Normal distributions underlying the three different choices of distributions for a statistical test on HTERs. Also shown: standard deviations of both the **FAR** and **FRR** distributions. All curves are in log-log scale. The order in the legend corresponds to the order of the curves at the right of the figure.

Empirical Analysis on XM2VTS

In the first case, the well-known text-independent audio-visual verification database XM2VTS (Lüttin, 1998) was used. In this database, the test set consists of up to 112000 impostor accesses and only 400 client accesses, for a total of 112400 accesses. In a recent competition (Messer et al., 2003), several models were compared on a face verification task and we will look here at the results of the best model, hereafter called *model A*, and the third best model, hereafter called *model B*, apparently significantly worse. Table 3.2 shows the difference of performance in terms of HTER between models A and B. Having up to 112400 examples, one could indeed expect the difference between the two

REF J Lüttin. Evaluation protocol for the the XM2FDB database (lausanne protocol). IDIAP-COM 05, IDIAP, 1998.

REF K. Messer, J. Kittler, M. Sadeghi, S. Marcel, C. Marcel, S. Bengio, F. Cardinaux, C. Sanderson, J. Czyz, L. Vandendorpe, S. Srisuk, M. Petrou, W. Kurutach, A. Kadyrov, R. Paredes, B. Kepenekci, F. B. Tek, G. B. Akar, F. Deravi, and N. Mavity. Face verification competition on the XM2VTS database. In *4th International Conference on Audio- and Video-Based Biometric Person Authentication, AVBPA*. Springer-Verlag, 2003.

models to be statistically significant.

While this is not the topic of this section (since it should apply to any data/model), people interested in knowing more about the problem tackled in this case study are referred to (Messer et al., 2003); we used results of the models of IDIAP and UniS-NC on the automatic registration task, using Lausanne Protocol I. Furthermore, note that the results of UniS-NC are slightly different from those published by Messer et al. (2003), but correspond to the list of scores provided by one of the authors of the method.

Method	FAR (%)	FRR (%)	HTER (%)
Model A	1.15	2.50	1.82
Model B	1.95	2.75	2.35

Table 3.2. HTER Performance comparison on the test set between models A and B when the threshold was selected according to the Equal Error Rate criterion (EER) on a separate validation set.

δ	HTER eq (3.30)	NAIVE eq (3.35)	CLASS eq (3.34)
90%	1.285%	0.131%	0.105%
95%	1.531%	0.156%	0.125%
99%	2.013%	0.206%	0.164%

Table 3.3. Confidence intervals around results of model A, computed using three different hypotheses (and their respective equation).

Table 3.3 shows the size of the confidence intervals computed around the result (using HTER or the classification error) obtained by model A for the three methods for three different values of δ (90%, 95% and 99%). As we can see, for all values of δ, the size of the interval is about one order of magnitude larger for the $\mathbf{Z_{HTER}}$-test based method than for the two other methods.

Table 3.4 verifies whether the HTER obtained by model A gives statistically significantly different results than the one obtained by model B, using the

REF K. Messer, J. Kittler, M. Sadeghi, S. Marcel, C. Marcel, S. Bengio, F. Cardinaux, C. Sanderson, J. Czyz, L. Vandendorpe, S. Srisuk, M. Petrou, W. Kurutach, A. Kadyrov, R. Paredes, B. Kepenekci, F. B. Tek, G. B. Akar, F. Deravi, and N. Mavity. Face verification competition on the XM2VTS database. In *4th International Conference on Audio- and Video-Based Biometric Person Authentication, AVBPA*. Springer-Verlag, 2003.

	HTER DEP, eq (3.33)	HTER INDEP, eq (3.32)	NAIVE eq (3.35)	CLASS eq (3.34)
δ	69.2%	64.7%	100.0%	100.0%
σ	0.0052	0.0057	0.0006	0.0005

Table 3.4. Confidence value δ on the fact that model A is statistically significantly different from model B, according to their respective performance (HTER or classification error), and computed using four different hypotheses (and their respective equation). For each method, we also give σ, the standard deviation of the corresponding statistical test.

two-sided test of equation (3.21) for the independent cases and (3.25) for the dependent case. According to both proposed $\mathbf{Z_{HTER}}$-test based methods (independent and dependent cases), both models are equivalent (the confidence on their difference, δ is much less than, say, 90%), while according to both other methods, the models would be different (with 100% confidence!). Remember that there was only 400 client accesses during the test, hence it is reasonable that only one error on these accesses makes a visible difference in HTER while it cannot seriously be considered statistically significant. This is well captured by our technique, but not by the other ones. Moreover, in this case, the dependence/independence assumption did not have any impact on the final decision.

Empirical Analysis on NIST'2000

In the second case, the well-known text-independent speaker verification benchmark database NIST'2000 was used. Here, the test set consists of 57748 impostor accesses and 5825 client accesses, for a total of 63573 accesses. We compared the performance of two models hereafter called *models C* and *D*. Note that, while on XM2VTS the ratio between the number of impostor and client accesses was very high (280 times more), for the NIST database, the ratio is more reasonable, but still high (around 10). Once again, while this is not the topic of this section, people interested in knowing more about the problem tackled in this case study are referred to (Mariéthoz and Bengio, 2003).

We now present the same kinds of results as for the XM2VTS case. Table 3.5 shows the difference of performance in terms of HTER between models C and

REF J. Mariéthoz and S. Bengio. An alternative to silence removal for text-independent speaker verification. IDIAP-RR 51, IDIAP, Martigny, Switzerland, 2003.

Method	FAR (%)	FRR (%)	HTER (%)
Model C	13.1	9.6	11.4
Model D	15.8	7.8	11.8

Table 3.5. HTER Performance comparison on the test set between models C and D when the threshold was selected according to the Equal Error Rate criterion (EER) on a separate validation set.

δ	HTER eq (3.30)	NAIVE eq (3.35)	CLASS eq (3.34)
90%	0.676%	0.414%	0.436%
95%	0.805%	0.493%	0.519%
99%	1.058%	0.648%	0.682%

Table 3.6. Confidence intervals around results of model C, computed using three different hypotheses (and their respective equation).

D; Table 3.6 shows the size of the confidence intervals computed around the result obtained by model C; as we can see, given a ratio of impostor and client accesses around 10 instead of 280, the difference between all the confidence intervals is less drastic but still exists; Table 3.7 verifies whether the HTER obtained by model C gives statistically significantly different results than the one obtained by model D. For each test, we show both the confidence value δ and the standard deviation σ of the corresponding statistical test.

As it can be seen, in the DEP case, σ is very small, even smaller than

	HTER DEP, eq (3.33)	HTER INDEP, eq (3.32)	NAIVE eq (3.35)	CLASS eq (3.34)
δ	98.8%	89.1%	98.9%	100.0%
σ^2	0.0016	0.0028	0.0018	0.0019

Table 3.7. Confidence value δ on the fact that model C is statistically significantly different from model D, according to their respective performance (HTER or classification error), and computed using four different hypotheses (and their respective equation). For each method, we also give σ, the standard deviation of the corresponding statistical test.

the NAIVE and CLASS solutions, hence obtaining a very high confidence that the two models are different. In order to explain this unexpected result, note than none of the tests takes into account the possible dependence existing between the compared *models*. Indeed, if the two models are based on the same technique (which is often the case; for instance, in speaker verification, most systems are often based on Gaussian Mixture Models, but trained with slightly different assumptions), then both systems will have a natural tendency to answer very correlated scores on the same example. In the case of the two models trained on the XM2VTS database, they were very different (one was based on a Gaussian Mixture Model, while the other one was based on Linear Discriminant Analysis and Normalized Correlation); while for the models trained on the NIST database, both were in fact variations of Gaussian Mixture Models, hence are probably very correlated. Unfortunately, there exist no test that take this dependency into account. Hence, for instance, the variance $\frac{p_{AB}+p_{BA}}{n}$ of equation (3.24) will be quickly very small simply because the models are correlated (and not just because the examples are the same). Using this equation will thus result in an underestimate of the true variance when models are very correlated, as empirically shown in Table 3.7.

On the other hand, the INDEP case does not take into account the dependency between the data, but somehow it is reasonable to expect that the effect of this error may be balanced by the fact that it does not take into account the dependency between the models neither. The correct solution probably lies somewhere between these two solutions, hence, one should probably favor the most difficult test so as to only assess statistical differences when both tests agree on this fact (hence, here, with only 89.1% confidence).

As we have seen, two tests can be used: the independent case and the dependent case. In the following, we will use the independent case because it is very simple to compute, only FAR and FRR are needed, and we make sure that its outcome is not optimistically biased. As we have defined several new concepts such as EPC and z-test for speaker verification systems, we now present a summary of the way we tend to present results in the rest of this document.

3.4 Methodology and Presentation of Results

In this thesis, we present results using numbers and curves. We chose to present HTER as number measure by setting the threshold with a criterion that minimizes the EER on some separate validation set. We also add a confidence interval using the algorithm described in Figure 3.9 using the independent case.

Table 3.8 shows examples of results:

Table 3.8. Sample of Results.

	Model A	Model B
HTER [%]	4.9	4.58
95% Confidence	±0.33	±0.33

DET curves will be used only for analysis purpose, as we have seen in Section 3.2, that EPC are more appropriate to presents final results. Different kinds of curves can be used. We propose to use a linear combination of FAR and FRR in abscissa representing the variation of γ. In ordinate, we would like to present a combination of FAR and FRR; two choices are possible, the DCF or HTER. The DCF has the advantage to plot what we are optimizing: a linear combination of FAR and FRR. The main drawback of this measure is that each point of the same curve cannot be compared. We can use HTER instead and in this case all points are comparable between curves which can be useful to choose a good operation point for a specific application. Figure 3.11 shows a typical EPC curve as presented later in order to compare systems. The best curve has its own confidence interval, but we need to have a confidence of how two models are different. This is thus presented in the second part of the figure. Each time that the blue line is greater that 95%, we can consider the two models as different with 95% confidence.

3.5 Conclusion

In this chapter we have presented the common measures used in speaker verification. We pointed out some problems of the use of theses measures found in the literature. First, we reminded that measures such that EER, ROC and DET curves are "a posteriori" measures and should thus not be used to compare systems. As no previously defined curve, to the best of our knowledge, was taking into account the decision threshold estimation problem, we have proposed new kinds of curves called EPCs. This work has been published in:

CONTRIB S. Bengio, J. Mariéthoz, and M. Keller. The expected performance curve. In *International Conference on Machine Learning, ICML, Workshop on ROC Analysis in Machine Learning*, 2005

and more specifically for speaker verification in:

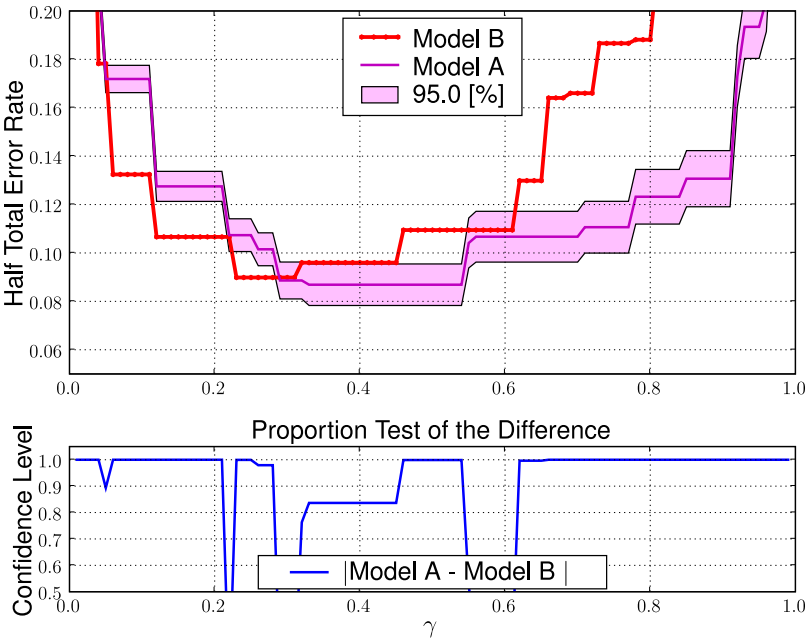


Figure 3.11. EPC curves using HTER with Confidence Intervals.

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

Moreover as no statistical test, such as the Z-test, was applicable to the speaker verification problem, we proposed an adapted Z-test to give a confidence interval for speaker verification systems such as HTER and DCF. This work has been published in:

CONTRIB S. Bengio and J. Mariéthoz. A statistical significance test for person authentication. In *Proceedings of Odyssey 2004: The Speaker and Language Recognition Workshop*, pages 237–240, 2004

Finally, we have presented a typical example of results as presented later in this thesis.

Once we have defined the measures, we need data to estimate the quality of

our new models. In the next chapter, we have chosen three well-known datasets and we have defined a new methodology to use them with discriminant models. Moreover, we present a new database called Banca with its own protocols and show that it is not easy to design a protocol to obtain unbiased results.