Toward a tight upper bound for the error probability of the binary Gaussian classification problem

Moataz M.H. El Ayadi*, Mohamed S. Kamel, Fakhri Karray

Pattern Analysis and Machine Intelligence Lab, Electrical and Computer Engineering, University of Waterloo, Ont., Canada N2L 3G1

Received 22 March 2007; received in revised form 23 October 2007; accepted 26 October 2007

Abstract

It is well known that the error probability, of the binary Gaussian classification problem with different class covariance matrices, cannot be generally evaluated exactly because of the lack of closed-form expression. This fact pointed out the need to find a tight upper bound for the error probability. This issue has been for more than 50 years ago and is still of interest. All derived upper-bounds are not free of flaws. They might be loose, computationally inefficient particularly in highly dimensional situations, or excessively time consuming if high degree of accuracy is desired. In this paper, a new technique is developed to estimate a tight upper bound for the error probability of the well-known binary Gaussian classification problem with different covariance matrices. The basic idea of the proposed technique is to replace the optimal Bayes decision boundary with suboptimal boundaries which provide an easy-to-calculate upper bound for the error probability. In particular, three types of decision boundaries are investigated: planes, elliptic cylinders, and cones. The new decision boundaries are selected in such a way as to provide the tightest possible upper bound. The proposed technique is found to provide an upper bound, tighter than many of the often used bounds such as the Chernoff bound and the Bayesian-distance bound. In addition, the computation time of the proposed bound is much less than that required by the Monte-Carlo simulation technique. When applied to real world classification problems, obtained from the UCI repository [H. Chernoff, A measure for asymptotic efficiency of a hypothesis based on a sum of observations, Ann. Math. Statist. 23 (1952) 493–507.], the proposed bound was found to provide a tight bound for the analytical error probability of the quadratic discriminant analysis (QDA) classifier and a good approximation to its empirical error probability.

Keywords: Binary classification; Bayesian decision rule; Decision boundary; Error probability; Monte-Carlo simulations; Multivariate normal distribution; Quadratic surfaces

1. Introduction

It is well known that there is no closed-form expression for the error probability of the binary Gaussian classification problem when the covariance matrices of the two class are different. Though the attempts of finding a tight upper bound for this error probability dates back to more than 50 years ago [1], it has been found that this problem is still of interest in many applications in different research areas. For example, in pattern recognition applications, the quadratic discriminant analysis (QDA) classifier has been found to provide powerful classification performances in some pattern recognition applications [2] despite its simplicity in comparison with other classification techniques. Since the QDA is based on the assumption that the class conditional densities are Gaussian, there have been some attempts to give a tight upper bound for the error probability of the QDA classifier [3]. In addition, estimates of this error probability has been used recently as a feature selection criterion [4]. In digital communication, there have been many attempts to estimate an upper bound for the error probability of different communication systems such as non-coherent coded modulation [5] and code division multiple access [6,7] when the communication channel suffers from an additive white Gaussian noise (AWGN), slow fading, or rapid fading. It can be shown that estimating the detection error probability in all these cases is equivalent to the considered Gaussian classification problem. In addition, in some radio astronomy and sonar applications, both the signal and the noise are best modelled as Gaussian
random processes since signals are perturbed by propagation through turbulent media [8]. Hence, the problem of evaluating the detection-error probability is also equivalent the considered Gaussian classification problem.

Because of the popularity of this problem, there have been many attempts to upper bound this error probability. The oldest and most known bounds for this error probability are the Chernoff bound [1] and the Bhattacharyya bound [9]. Though these bounds are easy to calculate, they are found to be significantly loose in many problems. Tighter error bounds have also been proposed: the equivocation bound [10]; the Bayesian distance bound [11]; the sinusoidal bound [3]; and the exponential bound [12]. However, again, there is no closed-form expressions for these bounds and numerical integrations may be necessary. Thus, evaluation of these bounds may well be inefficient for problems with high dimensionality because they require a great deal of computations.

A third approach to approximate the error probability is through the generation of a relatively large number of random instances that follow the distribution of each class. The generated samples are then classified according to the Bayesian decision rule. The error probability is simply estimated as the ratio of the number of misclassified samples to the total number of samples. This technique in the literature is known as the Monte-Carlo simulation technique [13]. Amazingly, the complexity of this technique does not grow with increasing the dimensionality of the data. However, the accuracy of the obtained estimate is exponentially proportional to the number of the generated points. In particular, in order to increase the accuracy of the obtained estimate to one digit of precision, the number of generated samples should be increased by two orders of magnitude. Therefore, the estimation of the error probability using this technique is time consuming due to the large number of samples required for a sufficient degree of accuracy. In order to overcome with this deficiency, some improvement has been proposed to the conventional Monte-Carlo simulation technique such as importance sampling [13]. Though these methods have been applied successfully in some problems, it is difficult to apply them in some complex problems such as Viterbi decoding [14].

In this paper, a new method is proposed for the estimation of a tight upper bound for the error probability of the binary Gaussian classification problem. This method has the advantages of providing good approximation to the error probability with a relatively small computation time. The basic idea is to replace the Bayesian boundary with sub-optimal decision boundaries. In particular, three types of decision boundaries are considered: planes, elliptic cylinder, and cones. The main motivation behind this replacement is the relative easiness of calculating the error probability when these surfaces are used as classification boundaries. At the same time, these boundaries are suboptimal in the sense that classification using them must result in inferior classification performance than that provided by the optimal Bayesian decision rule. Hence, these boundaries provide an upper bound for the true error probability. However, in order not to obtain a loose upper bound, their parameters should be optimized in such a way to obtain the least possible upper bound.

The rest of this paper is organized as follows. In Section 2, the binary Gaussian classification problem is briefly reviewed with the introduction of known error bounds. The possible shapes of the optimal Bayesian decision boundary are discussed in Section 3. The proposed error upper bound is described in Section 4. Comparative performance evaluation of proposed bounding technique with other selected techniques is provided in Section 5. Finally, concluding remarks are mentioned in Section 6.

2. The binary Gaussian classification problem: A brief review

The binary Gaussian classification problem is generally formulated as follows. Given a vector, \( x \in \mathbb{R}^d \), which belongs to one of either two possible classes: \( C_1 \) or \( C_2 \). It is required to determine which one of the following two hypotheses is more likely to occur:

\[ H_1 : x \text{ comes from the first class, } C_1. \]
\[ H_2 : x \text{ comes from the second class, } C_2. \]

It is usually assumed that data vectors belonging to each class \( C_i \), \( i = 1, 2 \) follow a certain distribution \( p(x|C_i) \). Usually, these distributions are incompletely known and the unknown parameters are estimated based on a set of training data. In this paper, it will be assumed that the amount of the training data is enough to estimate the unknown parameters with a sufficient degree of accuracy. Moreover, it will be assumed that the distributions take the form a multivariate normal distribution, i.e.,

\[
p(x|C_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2} (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i) \right),
\]

(1)

where \( \mu_i \) and \( \Sigma_i \) are the mean vector and the covariance matrix of \( C_i \). According to the Bayesian decision theory, the optimal decision rule is given as

\[
P(\bar{C}_i) p(x|\bar{C}_i) \leq P(\bar{C}_2) p(x|\bar{C}_2).
\]

(2)

Substituting Eq. (1) into Eq. (2) and performing some manipulations, the optimal Bayes decision rule for the Gaussian case is given by

\[
x^T Ax - 2b^T x + c \geq 0,
\]

(3)

where

\[
A = \Sigma_1^{-1} - \Sigma_2^{-1},
\]
\[
b = \Sigma_1^{-1} \mu_1 - \Sigma_2^{-1} \mu_2,
\]
\[
c = \mu_1^T \Sigma_1^{-1} \mu_1 - \mu_2^T \Sigma_2^{-1} \mu_2 + \log |\Sigma_1| - 2 \log \frac{P(\bar{C}_1)}{P(\bar{C}_2)}.
\]

A classification error occurs if a data vector \( x \) belongs to one class but falls in the decision region of the other class.
Therefore, the error probability is given by

\[ P_e = \int P(\text{error}|x)p(x)\, dx \]

\[ = \int \min (P(\mathcal{C}_1)p(x|\mathcal{C}_1), P(\mathcal{C}_2)p(x|\mathcal{C}_2)) \, dx \]

\[ = P(\mathcal{C}_1) \int_{R_1} p(x|\mathcal{C}_1) \, dx + P(\mathcal{C}_2) \int_{R_2} p(x|\mathcal{C}_2) \, dx, \]

(5)

where \( R_1 \) and \( R_2 \) are the decision region for \( \mathcal{C}_1 \) and \( \mathcal{C}_2 \), respectively. Unfortunately, the integrals in Eq. (5) do not have a closed form expression when \( \Sigma_1 \neq \Sigma_2 \). The main reason is that the decision boundaries are generally quadratic in nature which complicates the integrations in Eq. (5). Therefore, many upper bounds have been suggested to approximate the error probability. Almost all of these bounds depend on the approximation of the \( \min() \) function in Eq. (4) by a continuous differentiable function. Putting \( p = P(\mathcal{C}_1|x) \), the following approximations have been proposed:

- **Chernoff** [1]:
  \[ \min(p, 1-p) \leq p^s(1-p)^{1-s}, \quad 0 \leq s \leq 1. \]

- **Bhattacharyya** [9]:
  \[ \min(p, 1-p) \leq \sqrt{p(1-p)}. \]

- **Equivocation** [10]:
  \[ \min(p, 1-p) \leq -\frac{1}{2} \left[ p \log_2 p + (1-p) \log_2(1-p) \right]. \]

- **Bayesian distance** [11]:
  \[ \min(p, 1-p) \leq 2p(1-p). \]

- **Sinusoidal bound** [3]:
  \[ \min(p, 1-p) \leq 0.5 \sin(\pi p) \exp(-1.8063(p - 0.5)^2). \]

- **Exponential bound** [12]:
  \[ \min(p, 1-p) \leq g_t(p) + [1 - 2g_t(0.5)]g_e(p), \]
  where \( g_e(p) \) can be any of the above mentioned upper bounds and
  \[ g_t(p) = \frac{1}{x} \ln \left( \frac{\cosh(x/2)}{\cosh(x(p - 0.5))} \right). \]

Applying the Chernoff and Battahcharyya approximation to the \( \min() \) function in Eq. (4), the following two bounds are obtained:

- **Chernoff bound**:
  \[ P_e \leq \min_{0 \leq s \leq 1} \left\{ \left[ P(\mathcal{C}_1)p(\mathcal{C}_2)^{1-s} \right]^{\frac{1}{s(1-s)}} \right\} \times \exp \left( -\frac{s(1-s)}{2} (\mu_1 - \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) \right), \]
  (6)

where

\[ \Sigma = s\Sigma_1 + (1-s)\Sigma_2. \]

- **Bhattacharyya bound**:
  \[ P_e \leq \sqrt{P(\mathcal{C}_1)P(\mathcal{C}_2) - \frac{\sqrt{||\Sigma_1|| \Sigma_2||}}{\sqrt{||\Sigma_1 + \Sigma_2||/2}}} \times \exp \left( -\frac{1}{8} (\mu_1 - \mu_2)^T \left( \frac{\Sigma_1 + \Sigma_2}{2} \right)^{-1} (\mu_1 - \mu_2) \right). \]

(7)

Chernoff bound and Bhattacharyya bound are easy to compute. However, they are found to be much larger than the true value of the error probability obtained by the numerical integration in Eq. (5). This statement will be demonstrated further in Section 5. On the other hand, the other bounds are claimed to be tighter than the Chernoff and Battahcharyya bounds. However, there are no known closed-form expressions for these bounds and the integrals should be evaluated numerically [15]. Although the newly obtained integrals are easier to calculate than the original integral in Eq. (4) since they do not require the determination of the decision boundary, it is still infeasible to perform such integrations for problems with high dimensionality.

In order to develop an upper bound for the error probability that is tight and at the same time computationally efficient, it is beneficial to study the possible shapes of the optimal Bayesian decision boundaries. This is done in the next section. This can be useful in determining the cases at which it is possible to calculate the exact value of the error probability, and hence, no upper bound is needed.

3. Possible shapes of the optimal Bayesian decision boundary

Before proceeding to the details of the proposed bounding technique, it is worth figuring out the possible shapes of the decision boundary in Eq. (3). In order to simplify the analysis, the following linear transformation is made:

\[ y = U^T x, \]

(8)

where \( U \) is the eigenvector matrix for \( \Sigma_1^{-1}\Sigma_2 \) that satisfies

\[ U^T \Sigma_1 U = I, \]

\[ U^T \Sigma_2 U = A, \]

\( I \) is the identity matrix and \( A \) is a diagonal matrix: \( A = \text{diag}(\lambda_1, \ldots, \lambda_d). \) Applying the transformation (8), the decision rule (3) becomes

\[ y^T (I - A^{-1}) y - 2b^T y + c \geq 0, \]

(9)

where \( b = U^{-1} b. \) The conditional class densities of \( y \) are

\[ y|\mathcal{C}_1 \sim N(\mu_1, I), \]

\[ y|\mathcal{C}_2 \sim N(\mu_2, A), \]
where \( \tilde{\mu}_i = U^T \mu_i, i = 1, 2 \). The decision boundary in Eq. (9) (and Eq. (3)) can be either

1. A plane if \( A = I \),
2. An ellipsoid if \( I - A^{-1} \) is positive definite or negative definite, i.e., \( \lambda_i > 1 \) for all \( i \), or \( \lambda_i < 1 \) for all \( i \),
3. A cylinder (with a generally quadratic cross section of lower dimensionality) if there exist at least one index \( i \) such that \( \lambda_i = 1 \) and \( \tilde{b}_i = 0 \),
4. A paraboloid if \( \tilde{b}_i \neq 0 \) whenever \( \lambda_i = 1 \), or
5. A hyperboloid if some of the eigenvalues are greater than one while the others are less than one. A cone is a special case of the hyper-hyperboloid in which

\[
c = \tilde{b}^T (I - A^{-1})^{-1} \tilde{b}.
\]

It is straightforward to calculate the error probability for the first and the second cases as we shall show shortly. Under some suitable non-invertible linear transformation, the third case can be transformed to another binary Gaussian classification problem with lower dimensionality. The fourth and fifth cases are the hardest cases in which it may be necessary to replace these complex decision boundaries with simpler ones, namely either a plane, an elliptic cylinder, or a cone. Each of the five cases will be treated separately.

### 3.1. Linear decision boundary

In this case, the quadratic term in Eq. (9) will vanish. The error probability in this case is given by

\[
P_e = P(\xi_1) \int_{-\infty}^{\infty} N(y; \tilde{\mu}_1, I) \, dy
+ P(\xi_2) \int_{-\infty}^{\infty} N(y; \tilde{\mu}_2, I) \, dy.
\]

Making the change of variables \( z = y - \tilde{\mu}_1 \) in the first integral and \( w = y - \tilde{\mu}_2 \) in the second one, it is not hard to show that the above integral reduces to [2]

\[
P_e = P(\xi_1)Q \left( \frac{\tilde{b}^T \mu_1 - c/2}{\sqrt{\tilde{b}^T \tilde{b}}} \right) + P(\xi_2)Q \left( \frac{\tilde{b}^T \mu_2 + c/2}{\sqrt{\tilde{b}^T \tilde{b}}} \right),
\]

where

\[
Q(z) = \int_{z}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} \, du.
\]

It should be mentioned that there are efficient algorithms for the computation of the \( Q(.) \) function (see Ref. [16] for example). Moreover, there are some good approximations to this function such as [17]

\[
\frac{2e^{-z^2/2}}{(z + \sqrt{z^2 + 4}) \sqrt{2\pi}} \leq Q(z) \leq \frac{2e^{-z^2/2}}{(z + \sqrt{z^2 + 2}) \sqrt{2\pi}}.
\]

### 3.2. Ellipsoidal decision boundary

Without any loss of generality, it can be assumed that \( I - A^{-1} \) is positive definite; otherwise, the roles of classes \( \xi_1 \) and \( \xi_2 \) are switched. It will be convenient to rearrange the decision rule (9) to be in the form

\[
(y - \Theta^{-1} \tilde{b})^T \Theta(y - \Theta^{-1} \tilde{b}) \geq \tilde{b}^T \Theta^{-1} \tilde{b} - c,
\]

where \( \Theta = (I - A^{-1}) \). It should be noted that \( \Theta \) is also diagonal. Thus, after applying the transformation (8), the error probabilities are given by

\[
P_e = P(\xi_1) \int_{\mathcal{H}_1} N(y; \tilde{\mu}_1, I) \, dy + P(\xi_2) \int_{\mathcal{H}_2} N(y; \tilde{\mu}_2, I) \, dy,
\]

where

\[
\mathcal{H}_1 = \{ y : (y - \Theta^{-1} \tilde{b})^T \Theta(y - \Theta^{-1} \tilde{b}) < \tilde{b}^T \Theta^{-1} \tilde{b} - c \},
\]

and

\[
\mathcal{H}_2 = \{ y : (y - \Theta^{-1} \tilde{b})^T \Theta(y - \Theta^{-1} \tilde{b}) > \tilde{b}^T \Theta^{-1} \tilde{b} - c \}.
\]

Making the change of variables \( z = y - \tilde{\mu}_1 \) in the first integral and \( w = A^{-0.5} (y - \tilde{\mu}_2) \) in the second one and defining

\[
\Psi(\Phi, m, t) \equiv \int_{(y-m)^T \Phi (y-m) < t} N(y; 0, I) \, dy,
\]

the error probability can be expressed as

\[
P_e = P(\xi_1) \left[ 1 - \Psi \left( \Theta, \Theta^{-1} \tilde{b} - U^T \tilde{\mu}_1, \tilde{b}^T \Theta^{-1} \tilde{b} - c \right) \right]
+ P(\xi_2) \Psi (A \Theta, A^{-1/2} (\Theta^{-1} \tilde{b} - U^T \tilde{\mu}_2), \tilde{b}^T \Theta^{-1} \tilde{b} - c).
\]

Thus, what remains is to find an algorithm to calculate the function \( \Psi(\Theta, c, t) \). Fortunately, this problem was solved by Ruben [18]. He established an efficient algorithm for calculating \( \Psi(\Theta, c, t) \) stated in the following theorem.

**Theorem 1.** If \( y \sim N(0, I) \), the distribution of a positive definite quadratic expression of \( y \) is

\[
\Psi(\Phi, m, t) \equiv P \left( \sum_{i=1}^{d} \phi_i (y_i - m_i)^2 \leq t \right)
= \sum_{j=0}^{\infty} \alpha_j F_{d+2j}(t/p),
\]

where \( F_n(x) \) is the cumulative distribution function of a (central) \( \chi^2 \) random variable with \( n \) degrees of freedom, \( p \) is an

\footnote{In fact, there are many other algorithms for calculating \( \Psi(\Theta, c, t) \) such as Ref. [19]. In the authors’ opinion, the algorithm developed by Ruben is the most efficient algorithm found so far.}
arbitrary positive constant satisfying
\[|1 - p/\theta_i| < 1 \quad \forall i = 1, 2, \ldots, d,\]
and \(a_j\)'s are positive constants that sum to unity and are given by the following recurrence relations:
\[a_0 = \exp \left(-\frac{1}{2} \sum_{i=1}^{d} m_i^2 \right), \quad a_j = \prod_{i=1}^{d} p/\phi_i, \quad j = 1, 2, \ldots\]
\[a_j = \frac{1}{2j} \sum_{r=0}^{d} a_r g_{j-r}, \quad j = 1, 2, \ldots\]
\[g_n = \frac{d}{i=1} (1 - p/\phi_i)^n + np \sum_{i=1}^{d} m_i^2 (1 - p/\phi_i)^{n-1}.\]

Moreover, the error after summing \(k\) terms in the above expansion is bounded by
\[0 \leq \Psi(\Phi, e, t) - \sum_{j=0}^{k-1} a_j F_{d+2j}(t/p) \leq \left(1 - \sum_{j=0}^{k-1} a_j\right) F_{d+2k}(t/p).\]

**Proof.** The proof is omitted but can be found in Ref. [18]. □

It should be mentioned that the above series expansion can converge very quickly after summing few terms by a proper choice of \(p\). Moreover, for moderate and small values of \(t/p\), the following recurrence relation is useful.
\[F_r(x) = F_{r-2}(x) - \frac{e^{-x^2/(2r)}}{\Gamma(r/2)}, \quad r = d + 2, d + 4, \ldots,\]
where \(\Gamma()\) is the gamma special function.

### 3.3. Cylindrical decision boundary

Let \(I\) denotes the set of indices \(i\) for which \(\lambda_i = 1\) and \(\tilde{b}_i = 0\). It can be proven that the problem in this case can be reduced to another binary Gaussian classification problem with a smaller dimensionality by simply removing the components whose indices belongs to the set \(I\). This is done as follows. It is not difficult to show that if \(\tilde{b}_i = 0\) whenever \(i \in I\), the decision boundary in Eq. (9) will not contain \(\{y_i\}, i \in I\). Assume it takes the form
\[h(y_1) \geq 0, \quad h(y_1) \leq 0,\]
where \(y_1\) is a vector containing the elements of \(y\) whose indices do not belong to \(I\) and \(h(y_1)\) is some quadratic function of \(y_1\). Let \(y_2\) be a vector that contains the remaining elements of \(y\). The conditional error probability given that \(\mathcal{H}_1\) is true is thus given by
\[P(error|\mathcal{H}_1) = \int_{h(y_1) > 0} \int_{y_2} \mathbb{N}(y; \tilde{\mu}_1, I) dy\]
\[= \int_{h(y_1) > 0} \mathbb{N}(y_1; \tilde{\mu}_{11}; I_{d_1}) dy_1 \times \int_{y_2} \mathbb{N}(y_2; \tilde{\mu}_{12}; I_{d_2}) dy_2\]
\[= \int_{h(y_1) > 0} \mathbb{N}(y_1; \tilde{\mu}_{11}; I_{d_1}) dy_1, \quad (16)\]
where \(\tilde{\mu}_{11}\) is a vector containing elements of \(\tilde{\mu}_1\) whose indices does not belong to \(I\) and \(\tilde{\mu}_{12}\) contains the remaining items of \(\tilde{\mu}_1\). By the same way, it can be proven that
\[Pr(error|\mathcal{H}_2) = \int_{h(y_1) < 0} \mathbb{N}(y_1; \tilde{\mu}_{21}; A_1) dy_1, \quad (17)\]
where \(\tilde{\mu}_{21}\) is a vector containing elements of \(\tilde{\mu}_2\) whose of indices does not belong to \(I\) and \(A_1\) is a diagonal matrix that contains \(\lambda_i, i \notin I\). Thus, the error probability is finally found by integrating over \(y_1\) after removing out the elements in \(\tilde{\mu}_1, \tilde{\mu}_2\), and \(A\) whose indices belong to \(I\).

### 3.4. Paraboloidal decision boundary

Generally a hyperboloid may be considered as the limiting case of a hyper-hyperboloid in which one or more eigenvalues tense to one. Therefore, by replacing the unit eigenvalue(s) by \(1+\varepsilon\), where \(\varepsilon\) is a small value, the problem may be approximated to another classification problem with a hyperboloid decision boundary.

### 3.5. Hyperboloid decision boundary

This case is the most frequent case in problems with high dimensionality. The reason is simply that it is unlikely that all the eigenvalues of \(A\) are less than one or all of them are greater than one. Unfortunately, this is the hardest case for which to calculate the error probability. Similar to the case of an ellipsoidal decision boundary, it can be easily shown that the expression for the error probability is similar to Eq. (15) with the main difference that the first argument of \(\Psi\) is indefinite. Thus, it is required to find an expression for (or an upper bound to) the distribution of an indefinite quadratic expression of a standard multivariate normal variable.

While much attention in research was dedicated to finding the distribution of positive-definite or semi-definite quadratic expressions of a normal random variable, relatively few researchers such as [19–21] considered the case of indefinite quadratic forms. Moreover, the distribution is a series that is too complex to calculate. Thus, it is difficult in general to calculate the error probability when the decision boundary is hyperbolic. For this particular case, an upper bound for the error probability may be necessary.
4. The proposed error bound

Since it is difficult to calculate the Gaussian probability content over regions bounded by a hyperboloid, one may replace this boundary with another simpler linear or quadratic surface provided that the latter provides an upper bound for the error probability. Fortunately, this condition is always satisfied as long as the old decision boundary is the optimal Bayesian decision boundary. This fact is well known in the literature of pattern recognition and formally proven in the following theorem.

Theorem 2. The error probability due to any non-Bayesian decision regions are greater than or equal to that obtained by the optimal Bayesian decision regions.

Proof. Let \( \mathcal{R}_1 \) and \( \mathcal{R}_2 \) be the optimal Bayesian decision regions for class \( \mathcal{C}_1 \) and \( \mathcal{C}_2 \), respectively, and \( \mathcal{R}_1' \) and \( \mathcal{R}_2' \) be alternative decision regions for \( \mathcal{C}_1 \) and \( \mathcal{C}_2 \), respectively. From Eq. (5), the error probability is given by

\[
P_e = \int_{\mathcal{R}_1 \cap \mathcal{R}_1'} P(C_1) p(x|C_1) \, dx + \int_{\mathcal{R}_2 \cap \mathcal{R}_2'} P(C_1) p(x|C_1) \, dx 
+ \int_{\mathcal{R}_1 \cap \mathcal{R}_2} P(C_2) p(x|C_2) \, dx + \int_{\mathcal{R}_1 \cap \mathcal{R}_2'} P(C_2) p(x|C_2) \, dx.
\]

(18)

According to the decision rule (3), the first integral in Eq. (18) is bounded by

\[
\int_{\mathcal{R}_1 \cap \mathcal{R}_1'} P(C_1) p(x|C_1) \, dx \leq \int_{\mathcal{R}_2 \cap \mathcal{R}_2'} P(C_2) p(x|C_2) \, dx.
\]

(19)

since \( \mathcal{R}_2 \cap \mathcal{R}_1' \subset \mathcal{R}_2 \). Similarly, the fourth integral is bounded by

\[
\int_{\mathcal{R}_1 \cap \mathcal{R}_2'} P(C_2) p(x|C_2) \, dx \leq \int_{\mathcal{R}_1 \cap \mathcal{R}_1'} P(C_1) p(x|C_1) \, dx.
\]

(20)

Substituting inequalities (19) and (20) into Eq. (18), and arranging terms, the total error probability is bounded by

\[
P_e \leq \int_{\mathcal{R}_1 \cap \mathcal{R}_1'} P(C_1) p(x|C_1) \, dx 
+ \int_{\mathcal{R}_2 \cap \mathcal{R}_2'} P(C_2) p(x|C_2) \, dx 
= P(C_1) \int_{\mathcal{R}_1'} p(x|C_1) \, dx + P(C_2) \int_{\mathcal{R}_2} p(x|C_2) \, dx. \quad \square
\]

(21)

In this paper, three types of decision boundaries are investigated: planes, ellipsoids, and cones.

4.1. Alternative linear decision boundaries

As noted from Section 2, the easiest case for which to calculate the error probability is found when the decision boundaries are hyper-planes. However, the parameters of the decision boundary should be optimized in such a way to provide the tightest possible classification and hence the tightest possible classification. This problem was first investigated by Anderson and Bahadur [22]. They derived the following theorem, which we state here without proof.

Theorem 3. Consider the binary classification problem with class densities \( \mathcal{N}(\mu_i, \Sigma_i) \), \( i = 1, 2 \). Let \( t \) be found by the following optimization problem:

\[
t = \arg \min_{0 \leq x \leq 1} \left( P(C_1) Q\left( \sqrt{b^T \Sigma_1 b} \right) + P(C_2) Q\left( \sqrt{(1-x)b^T \Sigma_1 b} \right) \right).
\]

(22)

The best linear discriminant is given in the form

\[
b^T x - c \leq 0,
\]

with \( b \) being unique to a multiplicative constant and found as a solution to

\[
(t \Sigma_1 + (1-t) \Sigma_2)b = \mu_2 - \mu_1,
\]

and

\[
c = b^T \mu_1 + t b^T \Sigma_1 b = b^T \mu_2 - (1-t) b^T \Sigma_2 b.
\]

However, minimizing the objective function in Eq. (22) is not straightforward and usually done by trial and error. This may be undesirable for problems with high dimensionality because the computation of the inverse of \( t \Sigma_1 + (1-t) \Sigma_2 \) may be expensive. We shall refer to the bound obtained by this linear decision boundary as the ‘linear bound’.

4.2. Alternative elliptic cylinder and conical decision boundaries

In Section 5, it will be shown that the best linear discriminator is tight for some problems but loose for some others. Other possible alternative decision boundaries are ellipsoids, elliptic cylinders, and cones. However, it is very difficult to optimize the parameters of either an ellipsoidal or conical decision boundaries to obtain the tightest possible error bound. There are no closed expression for the optimum parameters of either the ellipsoidal or the conical decision boundary. At the same time, applying any optimization procedure will require much computations and may well be inferior to the Monte-Carlo simulation technique. As a suboptimal procedure, a good decision boundary is the one that best approximates the ideal Bayesian decision boundary geometrically.

For obtaining such approximate bounds, we derive the conditional error probabilities for the optimal Bayesian decision boundary and then investigate the possible simplification to the above problem. Define \( \tilde{P}(\cdot) \) in a way similar to Eq. (14) with the only difference that \( \Phi \) is in general indefinite. Thus, it is required to find an expression for (or an upper bound to) \( \tilde{P}(\Phi, \mathbf{m}, t) \) with \( t > 0 \). Without a loss of generality, the matrix \( \Phi \) can be assumed to be in the form

\[
\Phi = \text{diag}(\phi_1, \ldots, \phi_d, -\phi_{d+1}, \ldots, -\phi_d).
\]
Hence, $\Psi(\Phi, \mathbf{m}, t)$ can be expressed as

$$\Psi(\Phi, \mathbf{m}, t) = P \left( \sum_{i=1}^{d_1} \phi_i(y_i - m_i)^2 - \sum_{j=d_1+1}^{d} \phi_j(y_j - m_j)^2 < t \right)$$

$$= P \left( \sum_{i=1}^{d_1} \phi_i(y_i - m_i)^2 < t + \sum_{j=d_1+1}^{d} \phi_j(y_j - m_j)^2 \right)$$

Defining $u = \sum_{j=d_1+1}^{d} \phi_j(y_j - m_j)^2$, the above equation can be written in the following form:

$$\Psi(\Phi, \mathbf{m}, t) = E_u\{\Psi(\Phi_1, \mathbf{m}_1, t + u)\},$$

where the subscript ‘1’ denotes the first $d_1$ components of the vector and the first $d_1$ rows and $d_1$ columns of a matrix and $E_u\{\}$ denotes the expectation with respect to $u$. Applying Theorem 1 to the term inside the expectation, there exists some constants $e_k$ that depend only on $b_i$ and $m_i$, $i = 1, 2, \ldots, d_1$ and satisfy

$$\Psi(\Phi_1, \mathbf{m}_1, t + u) = \sum_{k=0}^{\infty} e_k F_{d_1+2k} \left( \frac{t + u}{p_1} \right),$$

for some constant $p_1$ chosen properly to make the above series converge quickly after the summation of few terms. Hence, Eq. (23) reduces to

$$\hat{\Psi}(\Phi, \mathbf{m}, t) = \sum_{k=0}^{\infty} e_k F_{d_1+2k} \left( \frac{t + u}{p_1} \right).$$

At the same time, $u$ is a positive definite quadratic expression of the random vector $\mathbf{y}_2$ that contains the last $d - d_1$ components of the vector $\mathbf{y}$. Therefore, according to Theorem 1 again, the distribution of $u$ can take the form of a convex combination of $\chi^2$ distributions:

$$P(U < u) = \sum_{l=0}^{\infty} h_l F_{d_2+2l} \left( \frac{u}{p_2} \right),$$

where $h_l$ are some positive constants obtained by Theorem 1 and $d_2 = d - d_1$. Hence, the probability density function of $u$ is given by

$$\frac{d}{du} P(U < u) = \sum_{l=0}^{\infty} \frac{h_l}{p_2} f_{d_2+2l} \left( \frac{u}{p_2} \right),$$

where $f_n(x)$ is the probability density function of a $\chi^2$ random variable with $n$ degrees of freedom, i.e.,

$$f_n(x) = \frac{e^{-x/2} x^{(n/2) - 1}}{2^n/\Gamma(n/2)}.$$  

Combining Eqs. (24) and (26)

$$\hat{\Psi}(\Phi, \mathbf{m}, t) = \sum_{k=0}^{\infty} e_k h_l \int_0^{u/p_2} \int_0^{(t+u)/p_1} f_{2k+d_1}(v) f_{2l+d_2} \left( \frac{u}{p_2} \right) dv \frac{du}{p_2}.$$  

Fig. 1. Optimal and alternative decision boundaries for the Gaussian Classification problem. Solid curve denotes the optimal Bayesian decision boundary. Dashed curve and dotted curve denote the conical and the cylindrical approximation to the Bayesian decision boundary. (a) An elliptical cylinder is better than a cone to approximate the optimal Bayesian decision region when $t$ is small. (b) A cone is better than an elliptical cylinder to approximate the optimal Bayesian decision region when $t$ is large.

The above integration is difficult to evaluate. Depending on the value of $t$, there are two possible approximations to this integral. When $t$ is sufficiently large, the $u$ term in the upper limit of the inner integral can be neglected. In this case, the above equation simplifies to

$$\hat{\Psi}(\Phi, \mathbf{m}, t) = \sum_{k=0}^{\infty} e_k F_{d_1+2k} \left( \frac{t}{p_1} \right).$$

Geometrically, this approximation corresponds to the replacement of the optimal Bayesian decision boundary with a cylindrical ellipsoid whose axis is the subspace spanned by the axes $y_j, j = d_1 + 1, \ldots, d$ and whose cross section at the hyperplane defined by $y_j = 0, j = d_1 + 1, \ldots, d$ is the ellipsoid

$$\sum_{i=1}^{d_1} \phi_i(y_i - m_i)^2 = t.$$  

This case is illustrated in Fig. 1(a). This bound will be referred to as the ‘cylindrical bound’.

On the other hand, if $t$ is sufficiently small, it can be neglected in the upper limit of the inner integral in Eq. (27). This approximation corresponds to a conical decision region as shown in Fig. 1(b). This bound will be called ‘conical bound’. To the author knowledge, very little research have been made to derive the probability content of a non-centralized cone. In this paper, an efficient algorithm is provided to calculate this value. In the simulation section, it will be demonstrated through an example this approximation is reasonably tight and the time of the proposed algorithm is relatively small.

Thus, $t$ will be set to zero in the above equation so as to derive the probability content of a cone. Substituting $u = 2z p_2 \cos^2(\zeta)$, and $v = 2z \sin^2(\zeta)$ in the above equation and after some
The amount of computations of the above double summation is obtained

\[
\hat{\psi}(\Phi, m, t) = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} e_k h_l \frac{J_{2k+d_1-1, 2l+d_2-1}}{\beta \left( \frac{2k+d_1}{2}, \frac{2l+d_2}{2} \right)},
\]

(29)

where \( \beta(\ldots) \) is the (complete) Beta function and \( J_{m,n} \) is given by

\[
J_{m,n} = 2 \int_0^{\zeta_0} \sin^m(\zeta) \cos^n(\zeta) \, d\zeta,
\]

\[
\zeta_0 = \tan^{-1} \left( \frac{p_2}{p_1} \right).
\]

The amount of computations of the above double summation can be greatly reduced by making use of the following recurrence relations:

\[
J_{m,n} = \frac{1}{m+n} \left( \frac{1}{m-n+1} - \frac{1}{m-n} \beta(m, n) \right) + (m-1) J_{m-2,n},
\]

\[
= \frac{1}{m+n} \left( \frac{1}{m-n+1} - \frac{1}{m-n} \beta(m, n) \right) + (n-1) J_{m,n-2},
\]

and

\[
\beta(m, n) = \frac{m+n-1}{m-1} \beta(m-1, n) = \frac{m+n-1}{n-1} \beta(m, n-1).
\]

Defining

\[
T_{k,l} = \frac{J_{2k+d_1-1, 2l+d_2-1}}{\beta \left( \frac{2k+d_1}{2}, \frac{2l+d_2}{2} \right)},
\]

it is not difficult to prove the following relations

\[
\lim_{l \to \infty} T_{k,l} = 1,
\]

(30)

\[
\lim_{k \to \infty} T_{k,l} = 0,
\]

(31)

\[
T_{k,l} = T_{k,l-1} + \frac{2}{(2l+d_2-2) \beta \left( \frac{2k+d_1}{2}, \frac{2l+d_2-2}{2} \right)} \times \frac{p_2^{(2k+d_1)/2} p_1^{(2l+d_2-2)/2}}{(p_1 + p_2)^{(2k+d_1-2)/2}},
\]

(32)

\[
T_{k,l} = T_{k-1,l} - \frac{2}{(2k+d_1-2) \beta \left( \frac{2k+d_1-2}{2}, \frac{2l+d_2}{2} \right)} \times \frac{p_2^{(2k+d_1-2)/2} p_1^{(2l+d_2)/2}}{(p_1 + p_2)^{(2k+d_1-2)/2}}.
\]

(33)

That is, for a fixed \( k \), the terms \( T_{k,l} \) increase with the increase of \( l \) but does not exceed one and decreases with the increase of \( k \). At the same time each of the \( T_{k,l} \) is between 0 and 1, and hence, the above expression can be calculated in a stable way. In addition, simple recurrence relations can be derived to calculate quickly second terms in the last two equations.

Based on the above relations, it is important to obtain an upper bound to the error resulting from summing up finite number of terms in Eq. (29). Let \( K \) and \( L \) be the number of terms calculated from the sequences \( \{e_k\} \) and \( \{h_l\} \), respectively. Eq. (29) is bounded as follows.

\[
\hat{\psi}(\Phi, m, t) = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} e_k h_l T_{k,l},
\]

\[
= \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} e_k h_l T_{k,l} + \sum_{k=K}^{\infty} \sum_{l=0}^{L-1} e_k h_l T_{k,l}
\]

\[
+ \sum_{k=0}^{\infty} \sum_{l=L}^{\infty} e_k h_l T_{k,l}.
\]

(34)

Using Eq. (32), it is easy to show that \( T_{k,l} \leq T_{K,L} \) for all terms in the second summation. Setting \( T_{k,l} \leq 1 \) in the third summation, we get

\[
\hat{\psi}(\Phi, m, t) \leq \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} e_k h_l T_{k,l} + \sum_{k=K}^{\infty} \sum_{l=0}^{L-1} e_k h_l T_{K,L}
\]

\[
+ \sum_{k=0}^{\infty} \sum_{l=L}^{\infty} e_k h_l T_{k,l}
\]

\[
= \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} e_k h_l T_{k,l} + \left( 1 - \sum_{k=0}^{K-1} e_k \right) \left( \sum_{l=0}^{L-1} h_l \right) T_{K,L}
\]

\[
+ 1 - \sum_{l=0}^{L-1} h_l.
\]

(35)

Hence for finite \( L \) and \( K \), the right-hand side of Eq. (35) can be calculated and then used as an upper bound for the conical and, consequently, the true error probabilities. It is also clear that the values of \( L \) and \( K \) should be large such that the first term in the right-hand side of Eq. (35) dominates other terms.

What remains is to determine when each of the ‘conical’ and ‘cylindrical’ approximations is better than the other. Fortunately, there is no need to answer this question as long as the computation time is small. The reason is simply that both approximations are upper bounds to the true error probability thanks to Theorem 2. Hence, one can calculate both of them and then take their minimum as a new bound for the error probability. This composite bound can be called the ‘conical-cylindrical’ bound.

5. Comparative performance evaluation of some selected bounding techniques

In this section, we compare the performance of the proposed bounding technique to several bounds already known in the literature by working on four examples. The first two
The curve corresponding to the Monte-Carlo bound is adopted to Gaussian distributions, $N_{\text{data}}$.

5.1. Example 1: Classification of two-dimensional Gaussian data

Consider the binary classification problem with the following Gaussian distributions, $N(\mu_i, \Sigma_i)$, $i = 1, 2$, where

$$\mu_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mu_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix},$$

$$\Sigma_1 = \alpha \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix}, \quad \Sigma_2 = \alpha \begin{bmatrix} 5 & -2 \\ -2 & 1 \end{bmatrix},$$

and $\alpha$ is a parameter controlling the interference between classes. For each value of $\alpha$ the following bounds are used to estimate the error probability: the Monte-Carlo simulation, the Chernoff bound, the Bayesian-distance bound, the linear bound, the cylindrical bound, and the conical bound. In Monte-Carlo simulations, $2.4 \times 10^6$ random samples are generated in order to ensure an accuracy of three significant digits with a 95% confidence level. The value of $\alpha$ was varied from 1 to 5 steps of 0.5. The above-mentioned bounds are plotted in Fig. 2. The curve corresponding to the Monte-Carlo bound is adopted as a reference curve because estimates measured using this method are the most accurate ones, (almost error free).

As expected, the error probability increases with the increase of $\alpha$. In the pattern classification problem, increasing $\alpha$ corresponds to a greater interference between classes.

It is also clear from Fig. 2 that, over the range $0 \leq \alpha \leq 1$, the curve representing the cylindrical bound is the closest one to the reference curve of Monte-Carlo technique. The average calculation time of different methods is also shown in Table 1.

5.2. Example 2: Classification of 12-dimensional Gaussian data

This example is similar to example 1 with the only different that data dimensionality was increased from 2 to 12. Two sets of 12-dimensional pseudo-random data vectors were generated according to the multivariate normal distribution, $N(\mu_i, \Sigma_i)$, $i = 1, 2$, where

$$\mu_1 = [3 \ 0 \ldots 0]^T, \quad \mu_2 = [-3 \ 0 \ldots 0]^T,$$

$$\Sigma_1 = \alpha \begin{bmatrix} 3 & -1 & 0 & \ldots & 0 \\ -1 & 3 & -1 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & -1 & 3 \\ 0 & \ldots & 0 & -1 & 3 \end{bmatrix},$$

$$\Sigma_2 = \alpha \begin{bmatrix} 4 & -2 & 0 & \ldots & 0 \\ -2 & 4 & -2 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & -2 & 4 \end{bmatrix}.$$
Fig. 3. Variation of different bounds values to the error probability for the two dimensional Gaussian distributions in Example 1.

Table 2
Average calculation times for different bounds of the error probability in Example 2

<table>
<thead>
<tr>
<th>Estimation method</th>
<th>Average calculation time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monte-Carlo simulation</td>
<td>20.1720</td>
</tr>
<tr>
<td>Chernoff bound</td>
<td>0.0232</td>
</tr>
<tr>
<td>Linear bound</td>
<td>0.1025</td>
</tr>
<tr>
<td>Conical bound</td>
<td>0.0350</td>
</tr>
<tr>
<td>Cylindrical bound</td>
<td>0.1666</td>
</tr>
</tbody>
</table>

as a reference curve. Averages calculation times for different bounds are reported in Table 2.

The value of the cylindrical bound was about 0.5 for all values of $\alpha$ (i.e. very loose) and hence not plotted in Fig. 3. As illustrated in the figure, the conical bound provided a very tight bound to the true value of the error probability. This means that the optimal Bayesian decision boundary is much closer in shape to a cone than to a cylinder. It follows from Table 2 that average calculation time of Monte-Carlo bound is 576 times as great as that of the conical bound. While the Chernoff bound is calculated in a very small computation time, it is indeed a loose upper bound. On the other hand, the linear bound provides a better upper bound and at the same time requires relatively small calculation time.

From Examples (1) and (2), it can be concluded that by calculating both the cylindrical and conical bound and then taking the minimum of the two obtained estimates, one can get a bound that achieves the best compromise between desired accuracy of the estimated error probability and the small calculation time. It seems also that the linear bound is tight for a special set of problems but not tighter than the composite conical-cylindrical bound.

5.3. Example 3: Classification of many randomly generated problems

In this example, we applied all the estimation methods except the linear bound to many randomly generated problems. In particular, the values of the class means were fixed to

$$\mu_1 = [1 \ 0 \ 0 \ 0 \ 0]^T,$$
$$\mu_2 = [-1 \ 0 \ 0 \ 0 \ 0]^T$$

while the covariance are generated randomly according to the formula

$$\Sigma_i = \alpha A^T A, \quad i = 1, 2$$

where $\alpha$ is a control parameter similar to those in the previous examples and $A$ is a $5 \times 5$ matrix whose all entries are generated according to the standard uniform distribution. The following five different values were substituted for $\alpha$: $10$, $100$, $1000$, $10^4$, $10^5$.

The following procedure was applied for each value of $\alpha$. 100 different Gaussian classification problems were generated. For each problem, all the above-mentioned estimation methods were employed. It was counted how many times the proposed bound (minimum of ellipsoidal and conical) was closer to the Monte-Carlo estimate than the Chernoff bound. The sum of the absolute of the differences between the proposed bound and the Monte-Carlo estimate as well as the sum of the absolute of the differences between the Chernoff bound and the Monte-Carlo estimate were also reported. Results for different values of $\alpha$ are summarized in Table 3. It is clear that the proposed bound is tighter than the Chernoff bound at about 78% of the cases. It is also obvious from Table 3 that, on the average, the proposed ellipsoidal–conical bound is much closer to the real Monte-Carlo estimate than Chernoff bound. However, the price we have to pay is, of course, a larger computation time but it is still significantly smaller than that required by the conventional Monte-Carlo estimation method.

5.4. Example 4: Application to real-world classification problems

The proposed bounding technique is applied to real world classification problems selected from the UCI machine learning repository [23]. For each selected database, a pair of classes is selected to apply our bounding technique. For each binary classification problem, the prior class probabilities, the mean
Table 4
A comparison between the analytical error probability of different bounds when applied to real world databases

<table>
<thead>
<tr>
<th>Database</th>
<th>Classes</th>
<th>( P(C_1) )</th>
<th>( P_{e_F} )</th>
<th>( P_{e_l} )</th>
<th>( P_{e_Q} )</th>
<th>( P_{eCher} )</th>
<th>( P_{e_{\text{ub}}Q} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bupa</td>
<td>1, 2</td>
<td>0.4203</td>
<td>0.3452</td>
<td>0.3408</td>
<td>0.2538</td>
<td>0.3907</td>
<td>0.3110</td>
</tr>
<tr>
<td>CMC</td>
<td>1, 2</td>
<td>0.6538</td>
<td>0.2823</td>
<td>0.2789</td>
<td>0.1925</td>
<td>0.3385</td>
<td>0.2804</td>
</tr>
<tr>
<td>CMC</td>
<td>2, 3</td>
<td>0.3945</td>
<td>0.5305</td>
<td>0.3286</td>
<td>0.2785</td>
<td>0.4212</td>
<td>0.2807</td>
</tr>
<tr>
<td>CMC</td>
<td>1, 3</td>
<td>0.5518</td>
<td>0.3519</td>
<td>0.3367</td>
<td>0.2690</td>
<td>0.4005</td>
<td>0.3332</td>
</tr>
<tr>
<td>Iris</td>
<td>Setosa, versicolor</td>
<td>0.5</td>
<td>2.3257 \times 10^{-6}</td>
<td>6.166 \times 10^{-8}</td>
<td>\approx 6 \times 10^{-8}</td>
<td>3.0442 \times 10^{-7}</td>
<td>9.478 \times 10^{-5}</td>
</tr>
<tr>
<td>Iris</td>
<td>Versicolor, virginica</td>
<td>0.5</td>
<td>0.0285</td>
<td>0.0283</td>
<td>0.0236</td>
<td>0.0671</td>
<td>0.0459</td>
</tr>
<tr>
<td>Magic04</td>
<td>1, 2</td>
<td>0.6484</td>
<td>0.1927</td>
<td>0.1635</td>
<td>0.0375</td>
<td>0.1056</td>
<td>0.0399</td>
</tr>
<tr>
<td>PIMA</td>
<td>0, 1</td>
<td>0.6510</td>
<td>0.2232</td>
<td>0.2203</td>
<td>0.1759</td>
<td>0.3386</td>
<td>0.3368</td>
</tr>
<tr>
<td>Yeast</td>
<td>MIT, NUC</td>
<td>0.3626</td>
<td>0.1790</td>
<td>0.1705</td>
<td>0.0038</td>
<td>0.0088</td>
<td>0.0131</td>
</tr>
<tr>
<td>Yeast</td>
<td>NUC, CYT</td>
<td>0.4890</td>
<td>0.3514</td>
<td>0.3115</td>
<td>0.0462</td>
<td>0.0817</td>
<td>0.0640</td>
</tr>
<tr>
<td>Yeast</td>
<td>MIT, CYT</td>
<td>0.3451</td>
<td>0.1871</td>
<td>0.1806</td>
<td>0.0191</td>
<td>0.0337</td>
<td>0.0208</td>
</tr>
<tr>
<td>Yeast</td>
<td>MIT, ME1</td>
<td>0.8472</td>
<td>0.0217</td>
<td>0.0171</td>
<td>0.0024</td>
<td>0.0080</td>
<td>0.0029</td>
</tr>
<tr>
<td>Yeast</td>
<td>NUC, ME1</td>
<td>0.9070</td>
<td>0.0067</td>
<td>0.0043</td>
<td>0.0034</td>
<td>0.0159</td>
<td>0.0282</td>
</tr>
<tr>
<td>Yeast</td>
<td>CYT, ME1</td>
<td>0.9132</td>
<td>0.0083</td>
<td>0.0061</td>
<td>0.0021</td>
<td>0.0096</td>
<td>0.0051</td>
</tr>
<tr>
<td>Waveform</td>
<td>0, 2</td>
<td>0.4942</td>
<td>0.0717</td>
<td>0.0665</td>
<td>0.0524</td>
<td>0.0950</td>
<td>0.0559</td>
</tr>
<tr>
<td>Waveform</td>
<td>1, 2</td>
<td>0.4927</td>
<td>0.0549</td>
<td>0.0541</td>
<td>0.0395</td>
<td>0.0709</td>
<td>0.0408</td>
</tr>
<tr>
<td>Wine</td>
<td>1, 2</td>
<td>0.4538</td>
<td>0.0054</td>
<td>0.0060</td>
<td>0.0018</td>
<td>0.0065</td>
<td>0.0056</td>
</tr>
<tr>
<td>Wine</td>
<td>2, 3</td>
<td>0.5966</td>
<td>0.0044</td>
<td>0.0030</td>
<td>0.000452</td>
<td>0.0014</td>
<td>0.0052</td>
</tr>
<tr>
<td>WPBC</td>
<td>Recur, non-recur</td>
<td>0.2371</td>
<td>0.1284</td>
<td>0.1284</td>
<td>3.2378 \times 10^{-4}</td>
<td>0.0012</td>
<td>0.0215</td>
</tr>
</tbody>
</table>

Table 5
A comparison between the empirical error probability of different bounds when applied to real world databases

<table>
<thead>
<tr>
<th>Database</th>
<th>Classes</th>
<th>( P(C_1) )</th>
<th>( P_{e_{(emp)}} )</th>
<th>( P_{e_{(emp)}} )</th>
<th>( P_{e_{(emp)}} )</th>
<th>( P_{e_{(emp)}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bupa</td>
<td>1, 2</td>
<td>0.4203</td>
<td>0.2928</td>
<td>0.3594</td>
<td>0.3623</td>
<td>0.3110</td>
</tr>
<tr>
<td>CMC</td>
<td>1, 2</td>
<td>0.6538</td>
<td>0.2661</td>
<td>0.2744</td>
<td>0.3015</td>
<td>0.2804</td>
</tr>
<tr>
<td>CMC</td>
<td>2, 3</td>
<td>0.3945</td>
<td>0.3602</td>
<td>0.3472</td>
<td>0.3389</td>
<td>0.2807</td>
</tr>
<tr>
<td>CMC</td>
<td>1, 3</td>
<td>0.5518</td>
<td>0.3579</td>
<td>0.3640</td>
<td>0.3351</td>
<td>0.3332</td>
</tr>
<tr>
<td>Iris</td>
<td>Setosa, versicolor</td>
<td>0.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>9.478 \times 10^{-5}</td>
</tr>
<tr>
<td>Iris</td>
<td>Versicolor, virginica</td>
<td>0.5</td>
<td>0.0300</td>
<td>0.0300</td>
<td>0.0300</td>
<td>0.0459</td>
</tr>
<tr>
<td>Magic04</td>
<td>1, 2</td>
<td>0.6484</td>
<td>0.2230</td>
<td>0.2335</td>
<td>0.2154</td>
<td>0.0399</td>
</tr>
<tr>
<td>PIMA</td>
<td>0, 1</td>
<td>0.6510</td>
<td>0.2214</td>
<td>0.2266</td>
<td>0.2344</td>
<td>0.3368</td>
</tr>
<tr>
<td>Yeast</td>
<td>MIT, NUC</td>
<td>0.3626</td>
<td>0.1575</td>
<td>0.1634</td>
<td>0.0059</td>
<td>0.0131</td>
</tr>
<tr>
<td>Yeast</td>
<td>NUC, CYT</td>
<td>0.4890</td>
<td>0.3498</td>
<td>0.3307</td>
<td>0.0034</td>
<td>0.0640</td>
</tr>
<tr>
<td>Yeast</td>
<td>MIT, CYT</td>
<td>0.3451</td>
<td>0.1825</td>
<td>0.1839</td>
<td>0.0071</td>
<td>0.0208</td>
</tr>
<tr>
<td>Yeast</td>
<td>MIT, ME1</td>
<td>0.8472</td>
<td>0.0382</td>
<td>0.0208</td>
<td>0.0486</td>
<td>0.0029</td>
</tr>
<tr>
<td>Yeast</td>
<td>NUC, ME1</td>
<td>0.9070</td>
<td>0.0085</td>
<td>0.0021</td>
<td>0</td>
<td>0.0282</td>
</tr>
<tr>
<td>Yeast</td>
<td>CYT, ME1</td>
<td>0.9132</td>
<td>0.0059</td>
<td>0.0039</td>
<td>0.0039</td>
<td>0.0051</td>
</tr>
<tr>
<td>Waveform</td>
<td>0, 2</td>
<td>0.4942</td>
<td>0.0731</td>
<td>0.0761</td>
<td>0.0671</td>
<td>0.0559</td>
</tr>
<tr>
<td>Waveform</td>
<td>1, 2</td>
<td>0.4927</td>
<td>0.0443</td>
<td>0.0491</td>
<td>0.0479</td>
<td>0.0408</td>
</tr>
<tr>
<td>Wine</td>
<td>1, 2</td>
<td>0.4538</td>
<td>0</td>
<td>0</td>
<td>0.0077</td>
<td>0.0056</td>
</tr>
<tr>
<td>Wine</td>
<td>2, 3</td>
<td>0.5966</td>
<td>0.0084</td>
<td>0</td>
<td>0</td>
<td>0.0052</td>
</tr>
<tr>
<td>WPBC</td>
<td>Recur, non-recur</td>
<td>0.2371</td>
<td>0.1082</td>
<td>0.1134</td>
<td>0.0101</td>
<td>0.0215</td>
</tr>
</tbody>
</table>

vectors, and the covariance matrices for each class are estimated from the training data using the ML criterion. These estimates are used to compute the following error probabilities:

1. \( P_{e_F} \): The analytical error probability of the Fisher discriminant classifier [17].
2. \( P_{e_l} \): The linear bound for the error probability.
3. \( P_{e_Q} \): an estimate of the analytical error probability of the QDA classifier, obtained by \( 5 \times 10^5 \) Monte-Carlo trials.
4. \( P_{e_{Cher}} \): the Chernoff bound for the error probability.
5. \( P_{e_{ubQ}} \): the proposed upper bound for the error probability of the obtained QDA classifier.

Table 4 shows the above-mentioned error probabilities for different databases. It is noted that the following inequality holds for all the databases: \( P_{e_Q} \leq P_{e_l} \leq P_{e_F} \). The first part of the inequality is consistent with our knowledge that the QDA classifier has a better classification ability than any linear classifier when the two classes can be modelled by Gaussian distributions. For some databases such as Yeast and WDBC, there is a significant difference between the two error probabilities \( P_{e_Q}, P_{e_l} \). The second part of the inequality naturally follows since \( P_{e_l} \) is the error probability of the best possible linear classifier. However, the difference between \( P_{e_F} \) and \( P_{e_l} \) is insignificant for almost all the databases. It is also noted that the...
proposed bound $P_{\text{emp}}(\text{ub})$ is less than the Chernoff bound in 14 cases, while the Chernoff bound is better in five cases only. This indicates the superiority of our proposed bound to the Chernoff bound in real world problems.

Table 5 shows the empirical estimates of the above-mentioned error probabilities. It is observed from Tables 4 and 5 that, only for few databases, there is a notable difference between the analytical error probability and the corresponding empirical estimate. This difference may be attributed to three reasons: the distributions of classes are not Gaussian, the number of the training data points in some databases are too small to reliably estimate the classifier parameters, and too few data points are used to estimate the empirical error probabilities. With only five exceptions, the following inequality is satisfied: $P_{\text{emp}}(\text{ub}) \leq P_{\text{eb}}(\text{emp})$. That is, in real world classification problems, the QDA classifier is expected to perform better than the best possible linear discriminant even when the mean vectors and the covariance matrices are estimated from the data. It can be also noted from Tables 4 and 5 that the average difference between $P_{\text{ub}}(\text{Q})$ and $P_{\text{eb}}(\text{Q})$ is 0.025727 and the average of $|P_{\text{ub}}(\text{Q}) - P_{\text{eb}}(\text{Q})|$ is 0.032631. This means that, on average, the proposed bound is close to the analytical error probability and provides a reasonable approximation to the empirical training error probability. Recalling that the expected value of the empirical error probability is equal to the true error probability [2], it is expected that the proposed upper bound (or any other upper bound) is greater than the empirical error probability when the number of data points for each class is large enough and the distribution of the data can be reasonably approximated as a Gaussian density.

6. Conclusions

In this paper, a new technique has been proposed for estimating an easy-to-calculate tight error bound for the error probability of the binary Gaussian classification problem. The basic idea is to replace the quadratic decision boundary by sub-optimal decision boundaries. The parameters of the linear decision boundary were optimized such that it provides the least possible upper bound for the error probability. On the other hand, the other two types of decision boundaries are obtained by a geometrical approximations to the optimal Bayesian decision boundary.

Experimental results show that the linear bound provides reasonably tight bound for some examples but loose bounds for others. It has also been shown that taking the minimum of the conical bound and the cylindrical bound results in a good bound in terms of closeness to the true value of the error probability and the small computation time it requires.

Comparing the proposed bounding technique to existing bounds such as Chernoff bound and the Bayesian-distance bound, it has been found that the proposed bounding technique was the tightest of them most of the time. At the same time, its computation requirements are not significantly different than that of Chernoff bound and are much less than that required by the Bayesian-distance bound.

For future research, we believe that the approach in Section 4.2 can be modified to exactly estimate the error probability when the decision boundary is hyper-hyperboloid using a series whose terms can be efficiently calculated like the one in Eq. (29). The only obstacle we have is the evaluation of the integral in Eq. (27).

References


About the Author—MOATAZ M.H. EL AYADI received his B.Sc. degree (Distinction with Hons) in Electronics and Communication Engineering from Faculty of Engineering, Cairo University, Egypt in 2000. He received his M.Sc. degree from the Dept. of Engineering Mathematics and Physics, Faculty of Engineering, Cairo University in 2003. He is currently a Ph.D. student in the Electrical & Computer Engineering dept., Faculty of Engineering, University of Waterloo. He is also a member of the Pattern Analysis and Machine Intelligence (PAMI) group. His research interests include statistical pattern recognition, and speech processing. His master work was in enhancing the performance of text independent speaker identification systems that uses Gaussian Mixture Models as the core statistical classifier. The main contribution was in developing a new model order selection technique based on the goodness of fit statistical test. He is expected to follow the same line of research in his Ph.D.

About the Author—MOHAMED S. KAMEL received his B.Sc. (Hons) EE (Alexandria University), M.A.Sc (McMaster University), Ph.D. (University of Toronto). He joined the University of Waterloo, Canada in 1985 where he is at present Professor and Director of the Pattern Analysis and Machine Intelligence Laboratory at the Department of Electrical and Computer Engineering. Professor Kamel holds Canada Research Chair in Cooperative Intelligent Systems. Dr. Kamel’s research interests are in Computational Intelligence, Pattern Recognition, Machine Learning and Cooperative Intelligent Systems. He has authored and co-authored over 350 papers in journals and conference proceedings, 7 edited volumes, 2 patents and numerous technical and industrial project reports. Under his supervision, 72 Ph.D. and M.A.Sc students have completed their degrees. He is the Editor-in-Chief of the International Journal of Robotics and Automation, Associate Editor of the IEEE SMC, Part A, Pattern Recognition Letters, Cognitive Neurodynamics Journal. He is also member of the editorial advisory board of the International Journal of Image and Graphics and the Intelligent Automation and Soft Computing Journal. He also served as Associate Editor of Simulation, the Journal of The Society for Computer Simulation. Based on his work at the NCR, he received the NCR Inventor Award. He is also a recipient of the Systems Research Foundation Award for outstanding presentation in 1985 and the ISRAM best paper award in 1992. In 1994 he has been awarded the IEEE Computer Society Press outstanding referee award. He was also a coauthor of the best paper in the 2000 IEEE Canadian Conference on Electrical and Computer Engineering. Dr. Kamel is recipient of the University of Waterloo outstanding performance award, the Faculty of Engineering distinguished performance award.

About the Author—FAKHRI KARRAY (M’98–SM’01) received his Ing. Degree from the University of Tunis, Tunisia, in 1984 and his Ph.D. degree from the University of Illinois, Urbana-Champaign, in 1989. He is a Professor of electrical and computer engineering at the University of Waterloo, Waterloo, ON, Canada, and Associate Director of the Pattern Analysis and Machine Intelligence Laboratory. His current research interests are in the areas of autonomous systems and intelligent man–machine interfacing design. He is the coauthor of Soft Computing and Intelligent Intelligent Systems Design (Reading, MA: Addison-Wesley, 2004). He is Associate Editor of the International Journal of Robotics and Automation and the Journal of Control and Intelligent Systems. He is a member of the Editorial Advisory Board of Inhenierya. Prof. Karray is an Associate Editor of the IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics. He was Program Chair of the 2002 IEEE International Symposium on Intelligent Control. He is KW Chapter Chair of the IEEE Control Systems Society.