Pattern Recognition 136 (2023) 109240

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

A proxy learning curve for the Bayes classifier

Addisson Salazar, Luis Vergara*, Enrique Vidal

Universitat Politècnica de València, Camino de Vera s/n, València 46022, Spain

ARTICLE INFO

Article history:
Received 22 April 2021
Revised 14 October 2022
Accepted 5 December 2022
Available online 9 December 2022

Keywords:
Classification
Parameter learning
Sample size
Training set size
Probability of error

ABSTRACT

In this paper, a theoretical learning curve is derived for the multi-class Bayes classifier. This curve fits general multivariate parametric models of the class-conditional probability density. The derivation uses a proxy approach based on analyzing the convergence of a statistic which is proportional to the posterior probability of the true class. By doing so, the curve depends only on the training set size and on the dimension of the feature vector; it does not depend on the model parameters. Essentially, the learning curve provides an estimate of the reduction in the excess of the probability of error that can be obtained by increasing the training set size. This makes it attractive in order to deal with the practical problems of defining appropriate training set sizes.

1. Introduction

1.1. Statement of the problem and previous related works

Predicting the sample size required for proper training of a classifier is a classic problem in pattern recognition. This is an important issue because obtaining labelled training data is one of the most typical limitations in real settings. For example, a diagnostic system based on biosignals implies experimentation with both healthy and sick patients. The necessary protocols and the nature of the experiment can greatly slow down the recording of signals, in addition to the stress often caused in the patient (e.g. the simultaneous acquisition of magnetic resonance images and electroencephalograms [1]). Also, the process of manually labelling signals can be tedious and require an expert with a high degree of specialization. One example of this is the classification of polysomnograms (PSG) [2]. PSGs are multimodal biomedical signalling recordings of sleeping patients, which are used to diagnose different sleep disorders such as sleep apnea. A typical 8 h PSG may require up to 2 h to be labelled by an expert. Another quite different example is the training of an automatic road surface classifier for automatic calibration of an Advanced Driver Assistance System [3]. This requires costly experiments to take into account the diversity of driving surfaces and conditions. A myriad of other examples can be given. All of them will greatly benefit from at least some guess as to how large the training sample size should be, i.e. what the minimum number of experiments should be to guarantee adequate training of the classifier. This could save on a lot of redundant experiments and their associated costs.

However, the minimum training sample size for a required performance depends on the (unknown) data distributions and on the classifier structure, so it is almost impossible to define a general criterion, even in the form of a rule of thumb. Thus, in the classic paper [4] some practical recommendations were given, though the main conclusion was the realization of the difficulty of the problem and the requirement for experimental verification of the limited theoretical guidelines. Since then, many experimental and theoretical contributions can be found. Some approaches have been purely experimental, considering some specific classifiers and datasets: Naïve Bayes, Support Vector Machine and Decision Tree to classify five classes of smoker status from excerpts of personal reports [5]; Partial Least Squares in combination with Linear Discriminant Analysis to classify five classes of cells from biospectroscopy signals considering small training sets [6]; Artificial Neural Networks to deduce a rule of thumb from a variety of simulated and real data sets relative to people preferences of transport mode [7]; Convolutional Neural Networks to classify six classes of Computer Tomography images [8]; Naïve Bayes to detect network intrusion in cybersecurity [9]; and Support Vector Machine with a linear kernel to classify a variety of medical data sets [10]. In most cases, a simple parametric model of convergence is fitted to the experimental learning curve. The hope is that these curves could be generalized to other classification setups. However, there is no evidence for such expectations. Moreover, the fully experimental approach needs realistic simulation models and/or large sets of real data.

* Corresponding author.
E-mail address: lvergara@dcom.upv.es (L. Vergara).

https://doi.org/10.1016/j.patcog.2022.109240

On the other hand, theoretical analysis is rather complex, requiring several of the following limiting assumptions: two-class problem, linear classifiers, known and/or equal covariance matrices, asymptotic expressions, and only lower and/or upper bounds obtained. The following are some representative works: [11] two-class, pseudo-Fisher linear classifier, focussing on very small sample sizes; [12] two-class, linear discriminant, common and known covariance matrices, the first moments of the error rate are obtained; [13] two-class, linear classifier, different but known covariance matrices, and bounds as well as an approximate expression of the error rate are derived; [14] two-class, linear discriminant, known and common covariance matrices, and asymptotic first moments and root mean-square of the error rate are obtained; [15] two-class, bounds of the error rate; [16] two-class, linear discriminant, common covariance, asymptotic error rate stochastic model derived from a stochastic model of the linear coefficients; [17] two-class, linear discriminant, and asymptotic expansions of the error rate are obtained; [18] two-class, different but known covariance matrices, and bounds of the error rate are given. A different approach is given in [19,20] for the general multi-class problem, not requiring prior knowledge of the model parameters, but considering only unidimensional discrete features.

The main difficulty in generalizing these theoretical approaches comes from trying a direct computation of the probability of error and/or the mean error of the classifier. Ref. [21] is a clear example of such a difficult approach, where convergence of the empirical error to the generalization error is considered, but only for unidimensional features and with a particular theoretical focus on the Support Vector Machine classifier. However, other guides for predicting the required training sample size can be obtained by measuring the convergence of other functions related to the final probability of error. We may call these methods "proxy approaches". Thus, in [22] the authors consider the analysis of the required sample size to estimate a covariance matrix, and in [23] the convergence of relative frequencies to probabilities is analyzed. From a practical viewpoint, these works lead to some rule of thumbs to fit the size of the training sample. The proposal in [24] is also interesting, where a given measure of complexity is related to the classifier performance. However, the objective is to extract the useful information in scenarios with an overabundance of data, rather than to predict the required sample size in situations of costly data acquisition.

1.2. New contributions and organization of the paper

The objective of this research is to get a theoretical learning curve for the Bayes classifier to be expressed as a function of only the training sample size and the feature space dimension. Therefore, the curve should be independent of the model parameters which are assumed to be unknown in a practical setting. To this end, we adopt a proxy approach. Instead of trying to compute the actual probability of error as a function of the training set size, we analyze the convergence of one specific statistic to its true value, as explained in the next section. The approach is given for the general multi-class problem. The analysis focuses on the Bayes classifier, initially assuming multivariate Gaussian models for every class, and then the results are extended to arbitrary models of parametric probability density function (pdf). In summary, in comparison with previous works, the learning curves obtained are valid for multidimensional features, any number of classes, arbitrary parametric pdf models, and they do not depend on the model parameters.

In Section 2, we explain the proxy approach for the Bayes classifier. Then, in Section 3 we derive the learning curve for the multivariate Gaussian model. In Section 4, the results are extended to general parametric models. Section 5 provides a set of experiments with simulated and real data to assess the usefulness of the theoretical learning curve in a variety of scenarios.

2. The proxy approach

Let us call the dimension of the feature vector M. This can later be considered a multivariate continuous random variable $\tilde{\mathbf{x}} = [\tilde{x}_1 ... \tilde{x}_M]^T$ with observed values $\mathbf{x} = [x_1 ... x_M]^T$.

Let us also assume classes $k = 1...K$. We consider that class k has a prior probability P_k and a class conditioned multivariate probability density $p_{\tilde{\mathbf{x}}}(\mathbf{x}/k, \boldsymbol{\theta}_k)$, which fits a model defined by the parameter vector $\boldsymbol{\theta}_k$. The posterior probability of class k conditioned to $\tilde{\mathbf{x}} = \mathbf{x}$can be written as:

$$P\left(k/\mathbf{x}, \boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\right) = \frac{p_{\tilde{\mathbf{x}}}\left(\mathbf{x}/k, \boldsymbol{\theta}_k\right) P_k}{p_{\tilde{\mathbf{x}}}\left(\mathbf{x}/\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K\right)}, \tag{1}$$

where $p_{\tilde{\mathbf{x}}}(\mathbf{x}/\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K) = \sum_{k=1}^{K} p_{\tilde{\mathbf{x}}}(\mathbf{x}/k, \boldsymbol{\theta}_k)$. The Bayes classifier selects the class with the maximum posterior. Let us define the set of values $z_k = g(p_{\tilde{\mathbf{x}}}(\mathbf{x}/k, \boldsymbol{\theta}_k) P_k) \quad k = 1...K$, where $g(\cdot)$ is any monotonic non-decreasing scalar function. Selecting the class with the maximum posterior is equivalent to selecting the class with maximumz_k. But z_k can be considered a realization of the random variable $\tilde{z}_k$ which depends on the random vector $\tilde{\mathbf{x}}$ and on the model parameters $\boldsymbol{\theta}_k$

$$\begin{array}{l} \tilde{z}_k = g\left(p_{\tilde{\mathbf{x}}}\left(\tilde{\mathbf{x}}/k, \boldsymbol{\theta}_k\right) P_k\right) \quad k = 1...K \\ \tilde{\mathbf{z}} = [\tilde{z}_1 ... \tilde{z}_K]^T \end{array}. \tag{2}$$

In practice, the model parameters are unknown, so they must be estimated from a limited number of training samples. Formally, this implies that (2) has to be modified to account for the randomness of the parameter estimates

$$\begin{array}{l} \tilde{\hat{z}}_k = g\left(p_{\tilde{\mathbf{x}}}\left(\tilde{\mathbf{x}}/k, \tilde{\hat{\theta}}_k\right) P_k\right) \quad k = 1...K \\ \tilde{\hat{\mathbf{z}}} = \left[\tilde{\hat{z}}_1 ... \tilde{\hat{z}}_K\right]^T \end{array}, \tag{3}$$

where $\tilde{\hat{\mathbf{z}}} = [\tilde{\hat{z}}_1 ... \tilde{\hat{z}}_K]^T$ is a random vector which, given $\mathbf{x}$ and the estimated parameters $\hat{\theta}_1, ..., \hat{\theta}_K$, yields particular realizations $\hat{\mathbf{z}} = [\hat{z}_1 ... \hat{z}_K]^T$.

Let us assume that a given observation $\mathbf{x}$ belongs to (true) class l and that we have perfect knowledge of all the model parameters. The true class will be selected if and only if$z_k < z_l, \ \forall k \neq l$. Hence, we can express the probability of error conditioned to l and to z_l in the form

$$P_{e/l,z_l} = 1 - \Pr\{\tilde{z}_k < z_l, \ \forall k \neq l\}, \tag{4}$$

where $\Pr\{\tilde{z}_k < z_l, \ \forall k \neq l\}$is the probability that the $K-1$ random variables $\tilde{z}_k \ k \neq l$ as defined in (2) are below the value z_l corresponding to the true class. Notice that $\Pr\{\tilde{z}_k < z_l, \ \forall k \neq l\}$is the joint cumulative distribution function of the K-1 random variables $\tilde{z}_k \ k \neq l$ at the K-1 points $z_l, ..., z_l$. To simplify the notation, we will define $\Pr\{\tilde{z}_k < z_l, \ \forall k \neq l\} = F_{\tilde{\mathbf{z}}_{k \neq l}}(z_l, ..., z_l)$where $\tilde{\mathbf{z}}_{k \neq l}$ is a random vector with elements $\tilde{z}_k \ k \neq l$. The overall probability of error can be computed by integrating $P_{e/l,z_l}$ over z_l and summing over l, thus:

$$\begin{array}{l} P_{e/l} = \int_{-\infty}^{\infty} P_{e/l,z_l} p_{\tilde{z}_l}(z_l) dz_l = 1 - \int_{-\infty}^{\infty} F_{\tilde{\mathbf{z}}_{k \neq l}}(z_l, ..., z_l) p_{\tilde{z}_l}(z_l) dz_l \\ P_e = \sum_{l=1}^{K} P_{e/l} P_l \end{array}, \tag{5}$$

where P_e is the Bayes error rate for the assumed parametric model, i.e. the probability of error corresponding to perfect knowledge of the model parameters. The actual probability of error will be obtained by replacing in (5) the distributions of the random variables $\tilde{z}_1, ..., \tilde{z}_K$with the corresponding distributions of the random

variables $\hat{\tilde{z}}_1, ..., \hat{\tilde{z}}_K$ (we assume for simplicity that $P_l \;\; l = 1...K$are known)

$$\begin{aligned} \hat{P}_{e/l} &= 1 - \int_{-\infty}^{\infty} F_{\tilde{\mathbf{z}}_{k\neq l}}(\hat{z}_l, ..., \hat{z}_l)\, p_{\tilde{z}_l}(\hat{z}_l) d\hat{z}_l \\ \hat{P}_e &= \sum_{l=1}^{K} \hat{P}_{e/l} P_l \end{aligned} \quad . \tag{6}$$

From (5) and (6), we can express the contribution of class l to the excess in probability of error due to the finite training sample size, in the form

$$\begin{aligned} \Delta P_{e/l} &= \hat{P}_{e/l} - P_{e/l} = E\left[\hat{f}\left(\hat{\tilde{z}}_l\right)\right] - E[f(\tilde{z}_l)] \\ \Delta P_e &= \sum_{l=1}^{K} \Delta P_{e/l} P_l \end{aligned} \quad , \tag{7}$$

where we have defined $\hat{f}(\hat{z}_l) = F_{\tilde{\mathbf{z}}_{k\neq l}}(\hat{z}_l, ..., \hat{z}_l) \;\; f(z_l) = F_{\tilde{\mathbf{z}}_{k\neq l}}(z_l, ..., z_l)$. Let us assume that the estimates $\hat{\theta}_1 ... \hat{\theta}_K$are consistent, i.e. $\hat{\theta}_k \to \boldsymbol{\theta}_k$ for an increasing size of the training set, so that$\hat{f}(\hat{z}_l) \to f(z_l)$and, from (7), $\Delta P_{e/l} \to 0$. Unfortunately, it is not possible to evaluate this convergence. This is because the analysis of convergence $\hat{f}(\hat{z}_l) \to f(z_l)$ is intractable. However, the analysis of the convergence $\hat{z}_l \to z_l$ is approachable if we know the sample size effects on the statistics $\hat{\tilde{z}}_l$. So, let us establish some connection between both convergences. Given that$\hat{f}(\hat{z}_l) \to f(z_l)$, by increasing the training set we will reach a size from which the following approximation holds:

$$\hat{f}\left(\hat{z}_l\right) \simeq \hat{f}(z_l) + \left(\hat{z}_l - z_l\right)\hat{f}'(z_l) \simeq f(z_l) + \left(\hat{z}_l - z_l\right) f'(z_l), \tag{8}$$

where $\hat{f}(\hat{z}_l)$ is expressed by the first two terms of the Taylor series expansion aroundz_l , and the joint cumulative distributions are close, i.e. $\hat{f}(\cdot) \simeq f(\cdot)$. This means that the learning curve obtained will be valid only after the training set size has increased so that (8) holds. We will show in all of the experiments in Section 5 that after an initial mismatch, the theoretical predictions of the learning curves reasonably fit the empirical estimates in concordance with (8).

Then, from (7) and (8), we can write:

$$\Delta P_{e/l} \simeq E\left[\left(\hat{\tilde{z}}_l - \tilde{z}_l\right) f'(\tilde{z}_l)\right] \leq E^{\frac{1}{2}}\left[\left(\hat{\tilde{z}}_l - \tilde{z}_l\right)^2\right] E^{\frac{1}{2}}\left[f'(\tilde{z}_l)^2\right] = \Delta P_{e/l}^{UB}, \tag{9}$$

where we have made use of the Schwartz-Cauchy inequality to establish an upper bound $\Delta P_{e/l}^{UB}$ for the excess of the probability of error. Notice that the statistics of $\tilde{z}_l$do not depend on the training set size, hence $E^{\frac{1}{2}}[f'(\tilde{z}_l)^2]$in (9) will simply be an unknown constant when analysing the convergence of $\Delta P_{e/l}^{UB}$for increasing training set size. Also remember that for consistent estimates of the model parameters, $\hat{z}_l \to z_l$, hence $E^{\frac{1}{2}}[(\hat{\tilde{z}}_l - \tilde{z}_l)^2] \to 0$ so $\Delta P_{e/l}^{UB} \to 0$ and then $\Delta P_{e/l} \to 0$. Analysis of the convergence $E^{\frac{1}{2}}[(\hat{\tilde{z}}_l - \tilde{z}_l)^2] \to 0$will lead us to a proxy learning curve of the classifier. Thus, in the next section we are going to calculate the mean-square error $(MSE)\; E[(\hat{\tilde{z}}_l - \tilde{z}_l)^2]$ assuming that$p_{\tilde{\mathbf{x}}}(\mathbf{x}/k, \boldsymbol{\theta}_k) \;\; k = 1...K$are multivariate Gaussian pdfs with arbitrary means and covariances. Then, the results will be extended to arbitrary parametric pdfs in Section 4. We will see that the theoretical learning curve only depends on the dimension of the feature vector and on the size of the training set, but not on the model parameters. This, in conjunction with (9), will provide practical interest in the learning curve obtained, as we will show in Section 5 via some simulated and real data experiments.

3. Derivation of the learning curve for the multivariate Gaussian model

Let us consider the multivariate Gaussian model

$$\ln p_{\tilde{\mathbf{x}}}(\mathbf{x}/k, \mathbf{b}_k, \mathbf{C}_k) = -\frac{M}{2}\ln 2\pi - \frac{1}{2}\ln|\mathbf{C}_k| - \frac{1}{2}(\mathbf{x} - \mathbf{b}_k)^T \mathbf{C}_k^{-1}(\mathbf{x} - \mathbf{b}_k), \tag{10}$$

where $\mathbf{b}_k$, the mean vector, and $\mathbf{C}_k$, the covariance matrix, are the model parameters of class k. Let us define

$$z_k = -\frac{1}{2}\ln|\mathbf{C}_k| - \frac{1}{2}(\mathbf{x} - \mathbf{b}_k)^T \mathbf{C}_k^{-1}(\mathbf{x} - \mathbf{b}_k) + \ln P_k. \tag{11}$$

It is clear from (1) that $\underbrace{\arg\max}_{k} P_{\tilde{\mathbf{x}}}(k/\mathbf{x}, \mathbf{b}_k, \mathbf{C}_k) = \underbrace{\arg\max}_{k}(z_k)$, so, given $\mathbf{x}$we have to compute (11) for $k = 1...K$ and select the class with maximumz_k. In (11) we have assumed perfect knowledge of the class model parameters, but in practice we have to estimate these parameters from a labelled training set of independent instances$\mathbf{x}_k^{(n)} \;\; n = 1...N_k$. Thus, maximum likelihood estimates are obtained from

$$\begin{aligned} \hat{\mathbf{b}}_k &= \tfrac{1}{N_k}\sum_{n=1}^{N_k} \mathbf{x}_n^{(k)} \\ \hat{\mathbf{C}}_k &= \tfrac{1}{N_k - 1}\sum_{n=1}^{N_k}\left(\mathbf{x}_n^{(k)} - \hat{\mathbf{b}}_k\right)\left(\mathbf{x}_n^{(k)} - \hat{\mathbf{b}}_k\right)^T \end{aligned} . \tag{12}$$

Substituting in (11), we have

$$\hat{z}_k = -\frac{1}{2}\ln\left|\hat{\mathbf{C}}_k\right| - \frac{1}{2}\left(\mathbf{x} - \hat{\mathbf{b}}_k\right)^T \hat{\mathbf{C}}_k^{-1}\left(\mathbf{x} - \hat{\mathbf{b}}_k\right) + \ln P_k. \tag{13}$$

Notice that $z_k$in (11) is a realization of the random variable $\tilde{z}_k$, which is a function of the random variable $\tilde{\mathbf{x}}$. However, $\hat{z}_k$ in (13) is a realization of the random variable $\hat{\tilde{z}}_k$, which is not only a function of the random variable $\tilde{\mathbf{x}}$, but also of the random variables $\hat{\tilde{\mathbf{b}}}_k$and $\hat{\tilde{\mathbf{C}}}_k$. Now, considering the previous section, we will concentrate on the values $z_l$and $\hat{z}_l$corresponding to the true class$l$, i.e., $\tilde{\mathbf{x}} \sim N(\mathbf{b}_l, \mathbf{C}_l)$.

3.1. Bias term of the learning curve

Firstly, we are going to derive the convergence of the mean $E[\hat{\tilde{z}}_l] \to E[\tilde{z}_l]$. In Appendix A, we have derived:

$$\begin{aligned} E\left[\hat{\tilde{z}}_l\right] = &-\frac{1}{2}\psi_M\left(\frac{N_l - 1}{2}\right) - \frac{1}{2}M\ln\frac{2}{N_l - 1} - \frac{1}{2}\ln|\mathbf{C}_l| \\ &-\frac{1}{2}\frac{N_l - 1}{N_l - M - 2}\left(1 + \frac{1}{N_l}\right)M + \ln P_l, \end{aligned} \tag{14}$$

provided that $N_l > M + 2$.

On the other hand:

$$\begin{aligned} E[\tilde{z}_l] &= -\tfrac{1}{2}\ln|\mathbf{C}_l| - \tfrac{1}{2}E\left[(\tilde{\mathbf{x}} - \mathbf{b}_l)^T \mathbf{C}_l^{-1}(\tilde{\mathbf{x}} - \mathbf{b}_l)\right] + \ln P_l = \\ &= -\tfrac{1}{2}\ln|\mathbf{C}_l| - \tfrac{1}{2}trace\left[\mathbf{C}_k^{-1}\mathbf{C}_l\right] + \ln P_l = -\tfrac{1}{2}\ln|\mathbf{C}_l| - \tfrac{1}{2}M + \ln P_l \end{aligned} . \tag{15}$$

Therefore, we have found the bias term of the learning curve, which is:

$$\begin{aligned} \Delta_B(N_l, M) = E[\tilde{z}_l] - E\left[\hat{\tilde{z}}_l\right] = &\frac{1}{2}\psi_M\left(\frac{N_l - 1}{2}\right) + \frac{1}{2}M\ln\frac{2}{N_l - 1} \\ &+\frac{1}{2}\frac{N_l - 1}{N_l - M - 2}\left(1 + \frac{1}{N_l}\right)M - \frac{1}{2}M, \end{aligned} \tag{16}$$

provided that $N_l > M + 2$.

Notice that the multivariate digamma function can be expressed as

$$\psi_M\left(\frac{N_l - 1}{2}\right) = \sum_{m=1}^{M}\psi\left(\frac{N_l - 1}{2} + \frac{1 - m}{2}\right) = \sum_{m=1}^{M}\psi\left(\frac{N_l - m}{2}\right), \tag{17}$$

where $\psi(x)$is the univariate digamma function. But:

$$\lim_{x\to\infty}\psi(x)=\ln x \quad\Rightarrow\quad \lim_{N_l\to\infty}\psi_M\Big(\frac{N_l-1}{2}\Big)=M\ln\frac{N_l-1}{2}. \tag{18}$$

So:

$$\lim_{N_l\to\infty}\Delta_B(N_l,M)=0. \tag{19}$$

Thus,$\hat{z}_l$ is an asymptotically unbiased estimate of z_l.

3.2. MSE learning curve

We need to compute the mean square error (*MSE*) $E[(\tilde{z}_l-\tilde{\hat{z}}_l)^2]$, which can be expressed as:

$$E\Big[\big(\tilde{z}_l-\tilde{\hat{z}}_l\big)^2\Big]=\mathrm{var}\big[\tilde{z}_l-\tilde{\hat{z}}_l\big]+E^2\big[\tilde{z}_l-\tilde{\hat{z}}_l\big]=\mathrm{var}\big[\tilde{z}_l-\tilde{\hat{z}}_l\big]+\Delta_B^2(N_l,M). \tag{20}$$

So in the following, we concentrate on the computation of $\mathrm{var}[\tilde{z}_l-\tilde{\hat{z}}_l]$. This can be expressed as:

$$\mathrm{var}\big[\tilde{z}_l-\tilde{\hat{z}}_l\big]=\mathrm{var}[\tilde{z}_l]+\mathrm{var}\big[\tilde{\hat{z}}_l\big]-2\mathrm{cov}\big[\tilde{z}_l\cdot\tilde{\hat{z}}_l\big]. \tag{21}$$

But [29]:

$$\mathrm{var}[\tilde{z}_l]=\mathrm{var}\big[-\tfrac{1}{2}\ln|\mathbf{C}_l|-\tfrac{1}{2}(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l)+\ln P_l\big]= \\ =\tfrac{1}{4}\mathrm{var}\big[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l)\big]=\tfrac{1}{2}trace\big[\mathbf{C}_l^{-1}\mathbf{C}_l\mathbf{C}_l^{-1}\mathbf{C}_l\big]=\tfrac{M}{2} \tag{22}$$

Moreover:

$$\mathrm{var}\big[\tilde{\hat{z}}_l\big]=\mathrm{var}\left[-\frac{1}{2}\ln\left|\tilde{\hat{C}}_l\right|-\frac{1}{2}\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)^T\tilde{\hat{C}}_{ll}^{-1}\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)+\ln P_l\right] \\ =\frac{1}{4}\mathrm{var}\left[\ln\left|\tilde{\hat{C}}_l\right|\right]+\frac{1}{4}\mathrm{var}\left[\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)^T\tilde{\hat{C}}_{ll}^{-1}\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)\right] \\ -\frac{1}{2}\mathrm{cov}\left[\ln\left|\tilde{\hat{C}}_l\right|,\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)^T\tilde{\hat{C}}_{ll}^{-1}\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)\right]. \tag{23}$$

Considering again that $(N_l-1)\tilde{\hat{\mathbf{C}}}_l\sim W_M(\mathbf{C}_l,N_l-1)$, we can write [25]:

$$\mathrm{var}\left[\ln\left|\tilde{\hat{\mathbf{C}}}_l\right|\right]=\psi'_M\Big(\frac{N-1}{2}\Big). \tag{24}$$

We have demonstrated in Appendix B that:

$$\mathrm{var}\left[\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)^T\tilde{\hat{C}}_{ll}^{-1}\Big(\tilde{x}-\tilde{\hat{b}}_l\Big)\right] \\ =\frac{(N_l-1)^2}{(N_l-M-2)^2(N_l-M-4)}\Big(M\big(1+\tfrac{1}{N_l}\big)+2M\big(1+\tfrac{1}{N_l}\big)^2\Big), \\ +\Big(\tfrac{N_l-1}{N_l-M-2}\Big)^2 2M\big(1+\tfrac{1}{N_l}\big) \tag{25}$$

provided that $N_l>M+4$.

We have also demonstrated in Appendix C that: $\mathrm{cov}[\ln|\tilde{\hat{C}}_l|,(\tilde{x}-\tilde{\hat{b}}_l)^T\tilde{\hat{C}}_{ll}^{-1}(\tilde{x}-\tilde{\hat{b}}_l)]\simeq 0$ for practical values of N_l.

Finally, we have demonstrated in Appendix D that:

$$\mathrm{cov}\big[\tilde{z}_l\cdot\tilde{\hat{z}}_l\big]=\frac{1}{4}\frac{N_l-1}{N_l-M-2}2M, \tag{26}$$

provided that $N_l>M+2$.

Considering (21) and the results obtained in (22), (24)–(26), we can define a variance learning curve $\Delta_V(N_l,M)$ in the form:

$$\Delta_V(N_l,M)=\mathrm{var}\Big[\tilde{z}_l-\tilde{\hat{z}}_l\Big]=\mathrm{var}[\tilde{z}_l]+\mathrm{var}\Big[\tilde{\hat{z}}_l\Big]-2\mathrm{cov}\Big[\tilde{z}_l\cdot\tilde{\hat{z}}_l\Big]= \\ =\frac{M}{2}+\frac{1}{4}\psi'_M\Big(\frac{N-1}{2}\Big)+ \\ +\frac{1}{4}\frac{(N_l-1)^2}{(N_l-M-2)^2(N_l-M-4)}\Big(M\big(1+\frac{1}{N_l}\big) \\ +2M\Big(1+\frac{1}{N_l}\Big)^2\Big)+ \\ +\Big(\frac{N_l-1}{N_l-M-2}\Big)^2\frac{M}{2}\Big(1+\frac{1}{N_l}\Big)-M\frac{N_l-1}{N_l-M-2}, \tag{27}$$

provided that: $N_l>M+4$.

Notice that:

$$\psi'_M\big(\tfrac{N-1}{2}\big)=\sum_{m=1}^{M}\psi'\big(\tfrac{N_l-m}{2}\big)\ , \\ \lim_{x\to\infty}\psi'(x)=0\ \Rightarrow\ \lim_{N_l\to\infty}\psi'_M\big(\tfrac{N_l-1}{2}\big)=0 \tag{28}$$

Hence, it is straightforward to verify that:

$$\lim_{N_l\to\infty}\Delta_V(N_l,M)=0. \tag{29}$$

Finally, the *MSE* convergence of $\hat{z}_l$towards $z_l$will be determined by the *MSE* learning curve. From (16) to (27):

$$E\left[\Big(\tilde{z}_l-\tilde{\hat{z}}_l\Big)^2\right]=\Delta_{MSE}(N_l,M)=\Delta_V(N_l,M)+\Delta_B^2(N_l,M), \tag{30}$$

provided that $N_l>M+4$.

From (19) to (29):

$$\lim_{N_l\to\infty}\Delta_{MSE}(N_l,M)=0. \tag{31}$$

Thus, $\hat{z}_l$ is a consistent estimate of z_l.

3.3. Overall learning

It is assumed in the above derivation that the true class is l, i.e. we have obtained the contribution $\Delta P_{e/l}$to the whole excess of probability of error as expressed in (7). Then, considering (9) in (7), the total excess of the error probability will be bounded by:

$$\Delta P_e=\sum_{l=1}^{K}\Delta P_{e/l}P_l\le\sum_{l=1}^{K}\Delta P_{e/l}^{UB}P_l=\sum_{l=1}^{K}\Delta_{MSE}^{1/2}(N_l,M)\alpha_l(M)P_l, \tag{32}$$

where $\alpha_l(M)=E^{\frac{1}{2}}[f'(\tilde{z}_l)^2]$ is a constant that is not dependent on N_l. Let us define $N=\min(N_l)\quad l=1...K$, and express $\Delta_{MSE}^{1/2}(N_l,M)=\Delta_{MSE}^{1/2}(N,M)-\delta_l$; then we can write (32) as

$$\Delta P_e\le\sum_{l=1}^{K}\big(\Delta_{MSE}^{1/2}(N,M)-\delta_l\big)\alpha_l(M)P_l\le\beta(M)\cdot\Delta_{MSE}^{1/2}(N,M), \tag{33}$$

where $\beta(M)=\sum_{l=1}^{K}\alpha_l(M)P_l$and we have taken into account that δ_l, $\alpha_l(M)$ and P_l are positive numbers. Notice that $\beta(M)$depends on the specific data distribution model, through $\alpha_l(M)$ (which depends on the joint distributions $f(z_l)=F_{\tilde{\mathbf{z}}_{k\neq l}}(z_1,...,z_l)$) and the priors $P_l\quad l=1...K$, but $\Delta_{MSE}^{1/2}(N,M)$ is a function of only N, the minimum learning set size per class, and M, the feature space dimension. This will not prevent the use of $\Delta_{MSE}^{1/2}(N,M)$as a proxy learning curve because it may provide the relative reduction of ΔP_e for increasing N, not requiring knowledge of $\beta(M)$. This will be shown in the experimental Section 5.

Meanwhile, let us gain some insights into $\Delta_{MSE}^{1/2}(N,M)$. Fig. 1a shows the function $10\log\Delta_{MSE}^{1/2}(N,M)$ for increasing N and M (we have represented the square root in consonance with (9)). Accordingly with the validity of (30), the initial value considered for $N$is $M+5$, and the final value is $N=200$ in all cases. As expected, $\Delta_{MSE}(N,M)$ decreases monotonically with N and increases monotonically with M. A fast descent can be observed at the beginning of the curves, which suggests a large reduction of $\Delta_{MSE}(N,M)$

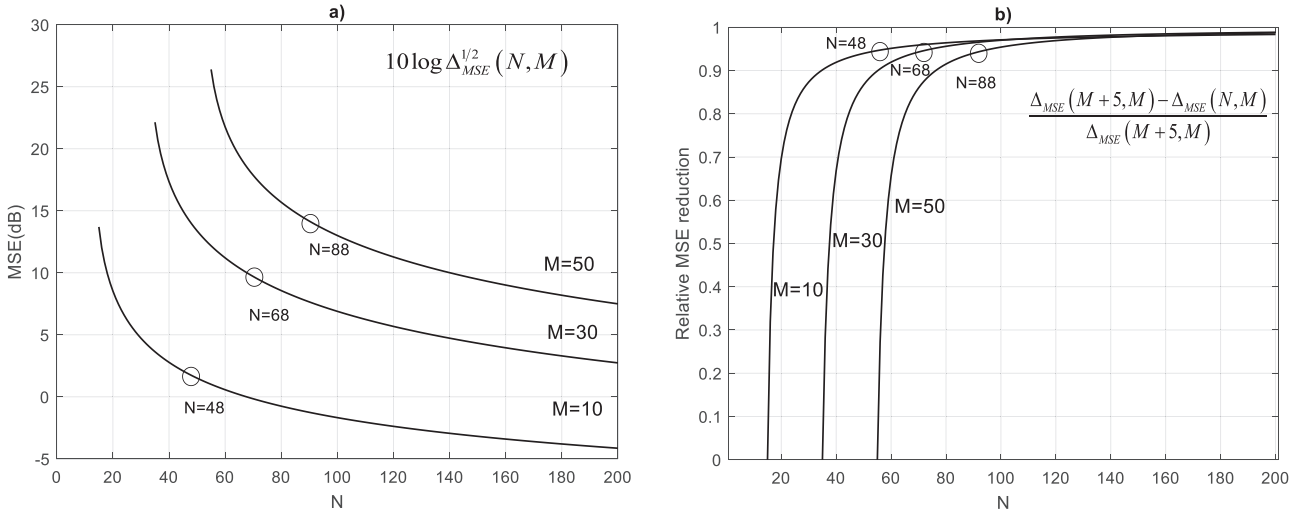

Fig. 1. Learning curves from Eq. (30): (a) *MSE* in dB scale. (b) *MSE* relative reduction.

for a relatively small increase in N. After that initial fast decrease, all of the curves fall slowly, which means that significantly greater improvement will require significantly greater values of$N$. On the other hand, notice that the total possible *MSE* reduction achieved by increasing N is given by $\Delta_{MSE}(M+5,M) - \Delta_{MSE}(\infty,M)$, but $\Delta_{MSE}(\infty,M)=0$ (see (31)). Hence, the relative reduction achieved at N with respect to the total possible reduction will be $(\Delta_{MSE}(M+5,M) - \Delta_{MSE}(N,M))/\Delta_{MSE}(M+5,M)$. We have represented this relative reduction in linear scale in Fig. 1b, so that the saturation effect for increasing N can be clearly seen.

In addition to its theoretical interest, the derived learning curve may have a direct practical application. Thus, we have marked 'o' at the points of the theoretical curves of Fig. 1a corresponding to a 12 dBs *MSE* fall, which correspond to around 94% of *MSE* relative reduction in Fig. 1b as defined in the paragraph above. This threshold seems a good (qualitative) trade-off between a significant reduction of the initial *MSE* and a small value$N$. Although this is somewhat arbitrary, a 12 dBs threshold could be selected, for example, to define a tentative value for N, which can be refined in the context of a particular application. Thus, the tedious and costly search for an appropriate training set size is alleviated. This will be shown in the real data application of Section 5.3, while the general validity of the theoretical curve will be assessed in Section 5.2 by Montecarlo simulations.

4. Extension to arbitrary parametric models

In this section, we consider the generalization of the previous results to arbitrary parametric models. First of all, the learning curves obtained can be applied to classifiers which assume multivariate Gaussian models for the class-conditional pdfs. As the learning curves are independent of the model parameters, the results are valid for the Gaussian Naïve Bayes (the particular case where $\mathbf{C}_l$ is assumed to be diagonal), linear discriminant ($\mathbf{C}_l$is assumed to be the same for all classes) or quadratic discriminant ($\mathbf{C}_l$can be different for every class).

On the other hand, as demonstrated in [30], arbitrary pdfs can be approximated by a weighted mixture of Gaussians. So let us express the class-conditional pdfs in the form:

$$p_{\tilde{\mathbf{x}}}\left(\mathbf{x}/k,\boldsymbol{\theta}_k\right)=\sum_{i=1}^{l_k}\pi_{ki}\mathrm{N}(\mathbf{b}_{ki},\mathbf{C}_{ki}) \qquad k=1...K, \tag{34}$$

where $\mathbf{b}_{ki}$and $\mathbf{C}_{ki}$ are respectively the mean and covariance of the i-th normal pdf component $\mathrm{N}(\mathbf{b}_{ki},\mathbf{C}_{ki})$ of the Gaussian mixture corresponding to class k, being weighted by π_{ki}. Every component in (34) defines a subclass i inside a class k. This is an extension of the model of Eq. (10), where only one Gaussian component is assumed for every class. Then, we can compute the following statistic for every subclass, as we did in (11) for every class:

$$z_{ki}=-\frac{1}{2}\ln|\mathbf{C}_{ki}|-\frac{1}{2}(\mathbf{x}-\mathbf{b}_{ki})^T\mathbf{C}_{ki}^{-1}(\mathbf{x}-\mathbf{b}_{ki})]+\ln\pi_{ki}. \tag{35}$$

Every given observation $\mathbf{x}$ belongs to a true subclass j of a true class l. So, in a similar manner to our derivation in Section 3, let us consider the convergence $E[(\tilde{\hat{z}}_{lj}-\tilde{z}_{lj})^2]\rightarrow 0$ to get a proxy learning curve for the general model in (34). First, notice that the maximum likelihood estimates of parameters $\mathbf{b}_{lj}$ and $\mathbf{C}_{lj}$can be obtained from (12) in a supervised setting where labelled training data are available for every subclass. We can also consider unsupervised training by using the Expectation Maximization (EM) algorithm. Notice that EM essentially makes an iterative use of (12) until maximum likelihood estimates of the parameters of every subclass are obtained. Thus, the expression of $\Delta_{MSE}(N_l,M)$ in (30), which was applicable to the theoretical learning of every class in the Gaussian model, is valid for the learning of every subclass of the Gaussian mixture. Considering the minimum training set size $N$for all the subclasses as we did in Section 3.3 for all classes, the minimum required training set size will be N multiplied by the total number of subclasses, while in the Gaussian single-component model of Section 3 it was N multiplied by the number of classes. The number of subclasses may be assumed to be known in advance or can be estimated using a variety of methods [31,32].

Certainly, a good Gaussian mixture approximation of arbitrary parametric pdfs as in (34) may in some cases require a large number of components and so a large number of parameters. In the event that a more parsimonious modelling of $p_{\tilde{\mathbf{x}}}(\mathbf{x}/k,\boldsymbol{\theta}_k)$ is possible, we have to estimate a minor number of parameters than the ones corresponding to the Gaussian mixture model of (34). Thus, for a given total training set size, the model estimation error will be smaller for the more parsimonious model (for example, see [33] for a detailed analysis of the importance of reducing the parameter dimension which describes the data). Thus, convergence will actually be faster than indicated by the theoretical curve obtained. In that case, $\Delta_{MSE}(N,M)$ provides us with an upper bound for convergence as it is derived under the fitting of a less parsimonious model than it may actually be.

Finally, nonparametric methods using Gaussian kernels also define a mixture of Gaussians to estimate the class-conditional pdfs. However, this cannot be approached using the framework presented. This is because every mixture component corresponds to

a Gaussian kernel centered in every member of the training set. Thus, the number of components increases with N. In general, nonparametric methods require much greater training sets than parametric approaches, so they are not very appropriate for scenarios of data scarcity. Nonparametric methods are outside the scope of this work; they have been largely considered in other works like [23].

5. Experiments

5.1. Preliminary considerations

In this section, we assess the theoretical results in some experiments using both simulated and real data. In all cases, we have considered two-class problems. Both classes will be equally probable and equally distributed except that means will have different signs. This is also applicable to subclasses when Gaussian mixtures are considered. All subclasses will be equally probable but the means of the components belonging to one class will have different signs from the means of the components belonging to the other class. We have also considered for simplicity that the training set size is the same for all classes or subclasses. Thus, the training set size per class (Gaussian model) or per subclass (Gaussian Mixture Models) will be indicated by N.

We use the Bayes classifier (Eq. (1)) in all cases. In the case of simulated Gaussian models, a multivariate Gaussian is assumed for the class conditioned multivariate probability density, and the model parameters were estimated using Eq. (12). In the case of the simulated Gaussian mixtures, a Gaussian mixture model is assumed for the class conditioned multivariate probability density, and the model parameters were estimated using EM method. In the real data experiment, a multivariate Gaussian is assumed for the class conditioned multivariate probability density and the model parameters were estimated using Eq. (12).

Notice that the probability of error is a function of N and M; thus, in the different experiments we have computed the empirical probability of error $\hat{P}_e(N,M)$. Moreover, assuming consistent convergence, the Bayes error rate can be estimated as $\hat{P}_e(M) = \hat{P}_e(N_\infty, M)$, where N_∞ is a large value from which no further reduction of $\hat{P}_e(N,M)$ is observed by increasing N. According to (7), the empirical estimate of the excess of probability of error will be computed as $\hat{\Delta}P_e(N,M) = \hat{P}_e(N,M) - \hat{P}_e(M)$. Notice that different data distributions will have different Bayes error rates. Therefore, for a better comparison among different models we compute the empirical relative excess of probability of error defined as:

$$\hat{r}(N,M) = \frac{\hat{\Delta}P_e(N,M)}{\hat{P}_e(M)}, \tag{36}$$

Let us apply (33) in (36)

$$\hat{r}(N,M) = \frac{\hat{\Delta}P_e(N,M)}{\hat{P}_e(M)} \leq \frac{\beta(M)}{\hat{P}_e(M)} \cdot \Delta^{1/2}_{MSE}(N,M) = \lambda(M) \cdot \Delta^{1/2}_{MSE}(N,M). \tag{37}$$

Hence, we see that $\hat{r}(N,M)$ is upper bounded by $\lambda(M) \cdot \Delta^{1/2}_{MSE}(N,M)$. Notice that, for a given M, this upper bound is linearly related to the proxy learning curve. Let us assume that this linear relationship also holds for the bounded variable $\hat{r}(N,M)$ for some $\gamma(M) \leq \lambda(M)$, i.e., $\hat{r}(N,M) = \gamma(M) \cdot \Delta^{1/2}_{MSE}(N,M)$. Then we could compute the reduction of $\hat{r}(N,M)$ for increasing$N$, from the corresponding reduction of the proxy learning curve, namely

$$\frac{\hat{r}(N,M)}{\hat{r}(N+m,M)} = \frac{\gamma(M)\Delta^{1/2}_{MSE}(N,M)}{\gamma(M)\Delta^{1/2}_{MSE}(N+m,M)} = \frac{\Delta^{1/2}_{MSE}(N,M)}{\Delta^{1/2}_{MSE}(N+m,M)}. \tag{38}$$

To verify that this is a reasonable assumption, in the experiments in the next section the theoretical curve $10\log_{10}\Delta^{1/2}_{MSE}(N,M)$ has been shifted (linear proportionality implies a shift in logarithmic scale) to get an optimal superposition with $10\log_{10}\hat{r}(N,M)$. Note that knowledge of the shifting value is not required in the practical application of the proxy curve; it is simply considered to assess the linear proportionality between $\hat{r}(N,M)$ and $\Delta^{1/2}_{MSE}(N,M)$.

5.2. Simulated data

First, we have considered multivariate Gaussian distributions, where

$class\,1 \quad \tilde{\mathbf{x}} \sim \mathrm{N}(a\mathbf{1},\ \mathbf{C})$
$class\,2 \quad \tilde{\mathbf{x}} \sim \mathrm{N}(-a\mathbf{1},\ \mathbf{C})$

We have defined four cases, depending on the signal-to-noise ratio (*SNR*) and the correlation. *SNR* is defined as the quotient between the magnitude of the mean value and the standard deviation. Therefore in this case $SNR = a$. The four cases are:

- Low *SNR*, no corr.: $a = 0.2 \quad \mathbf{C} = \mathbf{I}$
- Low *SNR*, corr.: $a = 0.2 \quad \mathbf{C}(i,j) = 0.3^{i-j}$
- High *SNR*, no corr.: $a = 0.4 \quad \mathbf{C} = \mathbf{I}$
- High *SNR*, corr.: $a = 0.4 \quad \mathbf{C}(i,j) = 0.3^{i-j}$

In all four cases, covariance matrices and means have been estimated using (12) from labelled training samples.

We have also considered cases where the class-conditional pdfs are mixtures of two Gaussian components, namely:

$class\,1 \quad \tilde{\mathbf{x}} \sim 0.5\mathrm{N}(a\mathbf{1},\ \mathbf{C}) + 0.5\mathrm{N}(b\mathbf{1},\ \mathbf{C})$
$class\,2 \quad \tilde{\mathbf{x}} \sim 0.5\mathrm{N}(-a\mathbf{1},\ \mathbf{C}) + 0.5\mathrm{N}(-b\mathbf{1},\ \mathbf{C})$

We have also defined four cases, depending on the signal-to-noise ratio (SNR) and the correlation. Now, *SNR* is defined as the quotient between the magnitude of the mean value of the second component and the standard deviation. Therefore in this case $SNR = b$. The four cases are:

- Low *SNR*, no corr.: $a = 0.2 \quad b = 0.1 \quad \mathbf{C} = \mathbf{I}$
- Low *SNR*, corr.: $a = 0.2 \quad b = 0.1 \quad \mathbf{C}(i,j) = 0.3^{i-j}$
- High *SNR*, no corr.: $a = 0.2 \quad b = 0.4 \quad \mathbf{C} = \mathbf{I}$
- High *SNR*, corr.: $a = 0.2 \quad b = 0.4 \quad \mathbf{C}(i,j) = 0.3^{i-j}$

Among the two options mentioned in Section 4, we have considered the most difficult, i.e., unsupervised learning. Then, the EM algorithm was applied to estimate the parameters of the Gaussian mixture. This is a well-known algorithm, routinely used with Gaussian mixture models. Starting from some initial estimates, the parameters of every Gaussian subcomponent i corresponding to every class k (Eq. (34)) are iteratively updated. This is done by first computing the conditioned probability of every training instance $\mathbf{x}_n^{(k)}$ to every Gaussian subcomponent considering the current parameters. Then, Eq. (12) is used to update the parameters, but weighting every contribution of $\mathbf{x}_n^{(k)}$ by the current estimate of the subcomponent conditioned probability. This later is updated with the new parameters. The algorithm ends when no significant variation is observed between two consecutive iterations.

In this experiment the number of components per class was assumed to be known and equal to 2. Fig. 2 shows the results corresponding to the eight cases. The upper half shows the four Gaussian cases, and the lower half the four non-Gaussian cases. We have considered the three feature vector sizes of Fig. 1, M=10, 30, 50. All the curves are functions of the training set size N, varying from N=M+5 (the smallest value for which the theoretical learning curve can be defined) to N=200. Remember that N is the training set size per class (one Gaussian component per class) or per subclass (two Gaussian components per class). Thus, the total size of the training set will be $2N$ in the Gaussian cases and $4N$ in the Gaussian mixture cases.

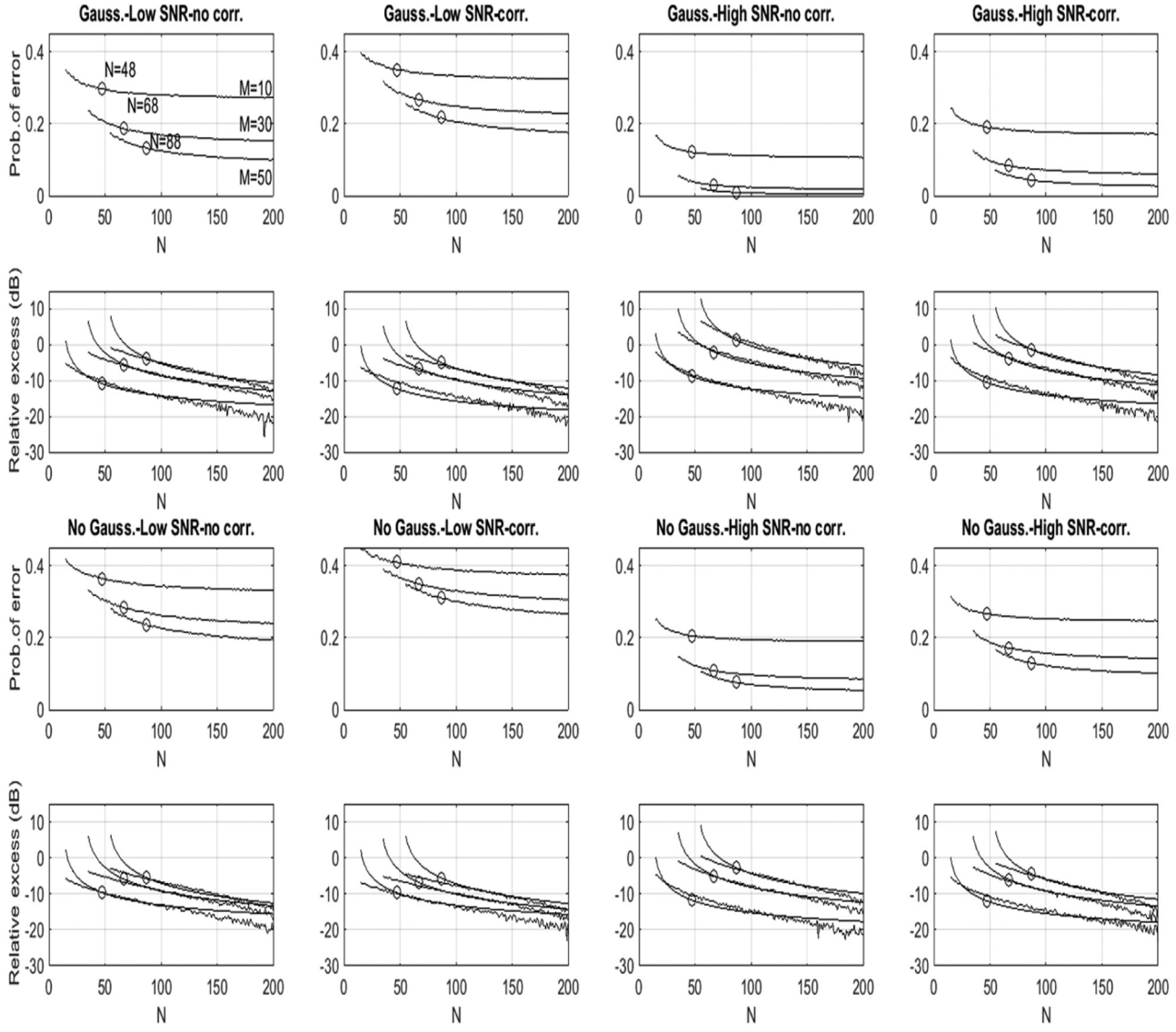

Fig. 2. Estimated probability of error, relative excess of probability of error and theoretical curves for the 8 simulated cases.

For every case, we have two sets of curves corresponding to the figures of merit explained in the foregoing Section 5.1. The upper set shows the empirical probabilities of error$\hat{P}_e(N, M)$. These probabilities have been estimated by averaging the estimates of 200 runs. Every run counted the number of errors of the trained classifier on a set of 500 testing samples per class.

As expected, we see that the probability of error is lower for high *SNR* in comparison with low *SNR*, as well as for no-correlation in comparison with correlation and also for Gaussianity in comparison with non-Gaussianity. The lower set of curves shows the corresponding shifted theoretical curves $10\log_{10}\Delta_{MSE}^{1/2}(N, M)$ superimposed, and the empirical relative excess of probability of error $10\log_{10}\hat{r}(N, M)$ as defined in (36). In all cases, we have selected$N_\infty = 201$, which, looking at Fig. 2 is an adequate value for $\hat{P}_e(N, M)$ to converge to a stable minimum (the estimated Bayes error rate). Superposition has been achieved by shifting $10\log_{10}\Delta_{MSE}^{1/2}(N, M)$ to get the minimum mean-square-error fit to $10\log_{10}\hat{r}(N, M)$. We can see that, after an initial mismatch, the theoretical curves fit the relative excess of probability of error quite well. The mismatch is in accordance with the first-order approximation of (8): the learning curve obtained will be valid only after the training set size has increased so that (8) holds. There is also a small deviation at the end of some of the curves. This is due to the empirical computations: we are estimating the Bayes error rate as $\hat{P}_e(201, M)$. However, $\hat{r}(N, M)$ approaches very fast to zero as N approaches 200, then $r(N, M)$ is slightly underestimated in some cases. A small difference between two values close to zero is enhanced in logarithmic scale, as we can see in Fig. 2.

Notice that the probability of error curves as well as the Bayes error rates change significantly from low to high *SNR*, from no-correlation to correlation, and from Gaussianity to non-Gaussianity. However, the relative excess curves cover a similar range of values in logarithmic scale, and fit the theoretical learning curves similarly. This suggests that in general we could use the theoretical curves to select an appropriate value for N, so that the initial relative excess of probability of error is conveniently reduced.

Moreover, we have computed the correlation coefficient between every pair of superimposed curves in Fig. 2. This is a classical measure of the possible linear relation between two variables. We have obtained a range of correlation coefficients between 0.89 and 0.95, with a mean value of 0.93. This indicates strong linear dependence between the empirical relative excess of error probability and the theoretical learning curves, as claimed after Eq. (37).

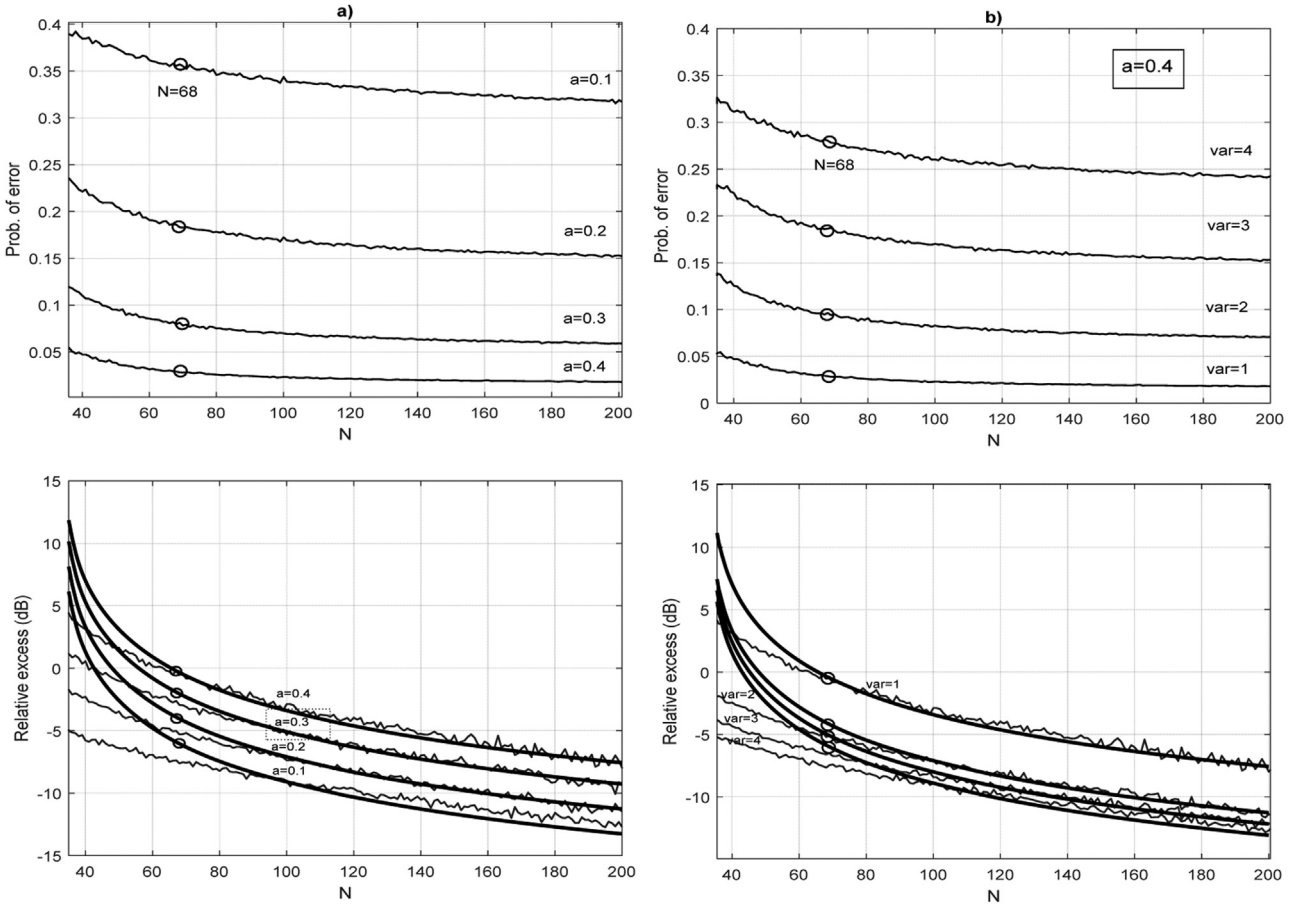

Fig. 3. Empirical probability of error (top), empirical relative excess and theoretical curves (bottom) for varying class separability, Gaussian case: a) Increasing mean b) Decreasing variance. M=30 in all cases.

Below, we present some additional experiments with simulated data to further assess the general validity of the approach for different data distribution models. In particular, we have focused on a relevant data model property that refers to class separability. It is clear that classes whose distributions are more separated in the feature space will lead to a smaller probability of error for a given training set size, as well as to a smaller Bayes error rate. Actually, this has been verified in the previous experiments under the concept of *SNR*. Notice that *SNR* was determined by the parameter a in the Gaussian case, and by the parameter b in the non-Gaussian case. Hence, a larger *SNR* implies a larger class separability and vice versa. In Figs. 3 and 4, we present the results of some additional simulations focussing on the separability matter. In all cases, we have considered $M = 30$, which is the intermediate value of the feature vector size of the three values considered in the previous experiment. Thus, similarly to Fig. 2, we show the empirical probability of error in the upper side of Figs. 3 and 4, and the empirical relative excess of probability of error superimposed with the shifted theoretical curves in the lower side. In Fig. 3 we have considered one Gaussian component per class (Gaussian case) for increasing class separability achieved by varying the mean: $a = \pm 0.1, \pm 0.2, \pm 0.3, \pm 0.4$ $\mathbf{C} = \mathbf{I}$ (Fig 3a), or by reducing the variance: $a = \pm 0.4$ $\mathbf{C} = 4\mathbf{I}$, $3\mathbf{I}$, $2\mathbf{I}$, $1\mathbf{I}$ (Fig. 3b). In Fig. 4 we have considered Gaussian mixtures for increasing class separability achieved by successively suppressing one component to the four-component model$a = \pm 0.4$, $b = \pm 0.3$, $c = \pm 0.2$, $d = \pm 0.1$ $\mathbf{C} = \mathbf{I}$, or by reducing the variance to the four-component model: $a = \pm 0.4$, $b = \pm 0.3$, $c = \pm 0.2$, $d = \pm 0.1$ $\mathbf{C} = 2\mathbf{I}$, $\mathbf{C} = 1\mathbf{I}$, $\mathbf{C} = 0.6\mathbf{I}$, $\mathbf{C} = 0.4\mathbf{I}$.

As expected, the results of Figs. 3 and 4 confirm that class separability has a definite impact on the probability of error and so on the Bayes error rate. However, the good fit of the shifted theoretical curve to the relative excess remains similar to the one observed in the eight cases of Fig. 2. Finally, as we did in Fig. 1, we have marked 'o' at points of the theoretical curves of Figs. 2–4 corresponding to the 12 dBs fall from the initial *MSE*. The selected points are also marked in the curves of probability of error. In all cases, it can be observed that a significant reduction of the error probability is achieved using this threshold. Then, as already suggested in Section 3.3, the corresponding value N can be an initial selection for the training set size. Let us show this practical issue in the next section with a real data experiment.

5.3. Real data

In this section, we present a real data example where features are extracted from Electroencephalographic (EEG) signals recorded at the Hospital Universitari i Politècnic La Fe, Valencia (Spain). The aim is to implement an automatic classifier of neurological activity (see [34] Section V, [35] Section 5, and [36] Section 4, for more details of the application). The subject under analysis performs an abbreviated subtest of the "Barcelona test" (BT) suite [37]. A total of 10 trials were carried out with increasing difficulty. In every trial, the subject was shown an item on the computer monitor screen for 3 s (stimuli), and after a 2 s retention interval they were asked to recognize the previously seen item from among a set of four similar items (response). Once recognized, the subject pressed

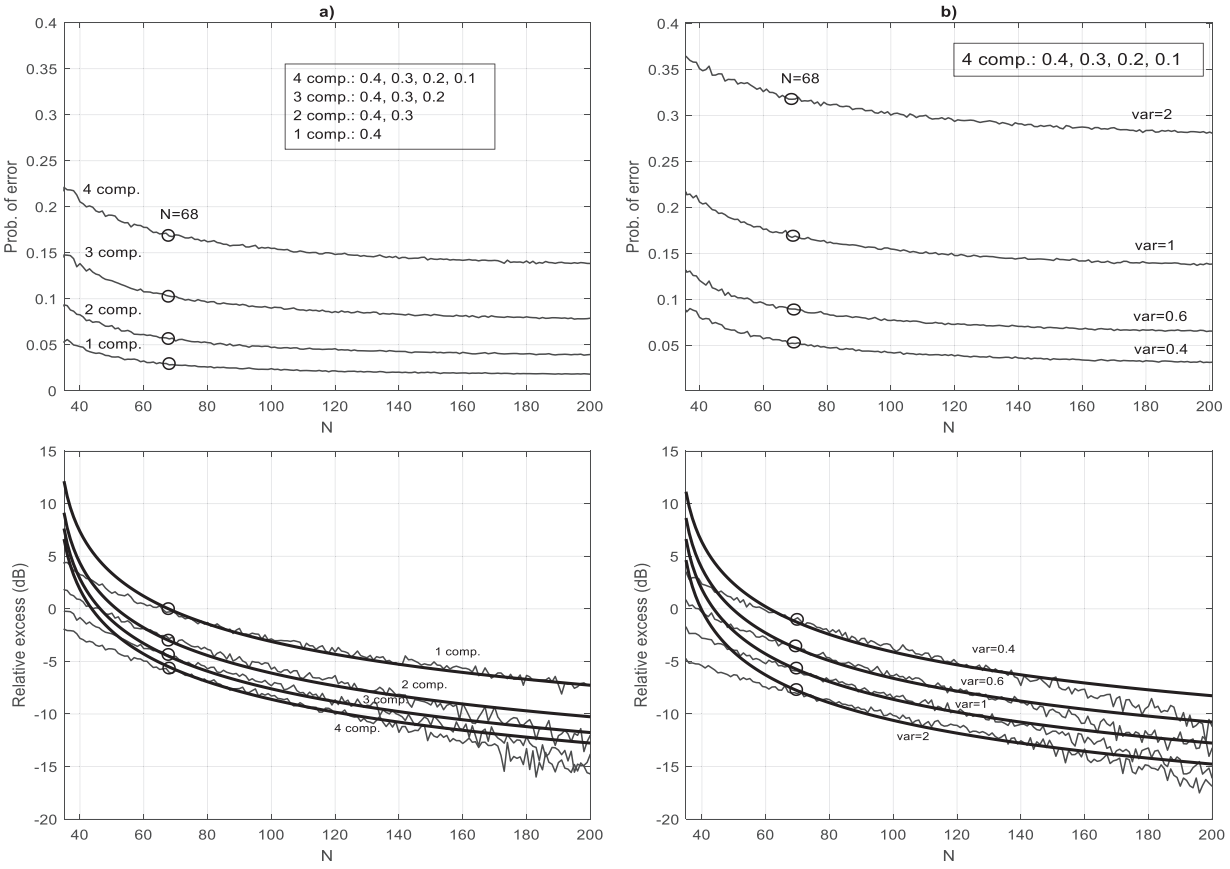

Fig. 4. Empirical probability of error (top), empirical relative excess and theoretical curves (bottom) for varying class separability, Gaussian mixture case: (a) Decreasing number of components (b) Four components, decreasing variance. M=30 in all cases.

a key and a new trial started. During the test, a given number of EEG channels was recorded. Every channel was band-pass filtered between 0.5 and 30 Hz and sampled at a sampling frequency of 500 Hz. The objective was to implement an automatic two-class classifier, where Class 1 corresponds to the "stimuli + retention" state and Class 2 to the "response" state. The correct performance of the classifier on a healthy subject will demonstrate that in normal conditions the EEG signals provide information about the commutation between the two different neurological activities of the BT. Hence, the particular performance of an automatic classifier trained on a subject may be an additional element to diagnose possible neurological diseases.

In this experiment, we selected one of the EEG signals recorded, which was divided into non-overlapped epochs of 0.25 s. We extracted seven features in every epoch: sample mean, sample mean absolute value, centroid frequency, and powers in the delta (0.5–4 Hz), theta (4–8 Hz), alpha (8–13 Hz) and beta (13–30 Hz) frequency bands. From these features, we made epoch feature vectors of dimension 7. Then, we obtained feature class vectors by averaging all the epoch feature vectors included within the same class interval (we know the initial and final instants of every class). Thus, we obtained one labelled feature vector of every class for each trial, for a total of 10 labelled feature vectors of each class for every implementation of the BT. To get more labelled feature vectors for both training and testing, we can repeat the BT on the same subject with different monitoring images as many times as we need. However, note that the subject will become progressively tired, so it is of great relevance to estimate what a reasonable training and testing sample size should be.

In Fig. 5 we show similar curves to the ones shown in the previous figures from simulated data. We have assumed a single-component Gaussian model in both classes. Covariance matrices and means have been estimated using (12). Two different subjects have been tested. Every subject has run the BT 10 times so that we got 100 labelled feature vectors for every class. Then, we made 250 partitions of N and 100-N feature vectors for training and testing respectively, with N ranging from 7+5 to 82 in steps of 5. In Fig. 3 (top) we show the estimated probabilities of error for a training set size ranging from N=7+5 to 77. This has been computed by averaging over the 250 partitions. The value $\hat{P}_e(82,7)$ is not shown because it has been used as an estimate of the Bayes error rate required to compute $10\log_{10}\hat{r}(N,7)$. These later curves are shown in Fig. 3 (bottom) superimposed with the shifted theoretical curves $10\log_{10}\Delta_{MSE}^{1/2}(N,7)$. In Fig. 3 we have also marked with 'o' the points corresponding to a 12 dB drop of the theoretical curves, which corresponds to a training set size of N=47. However, in the context of this application, it is more important to determine if the classifier shows a learning performance, i.e. if the empirical probability of error reasonably decreases for increasing training size, rather than to achieve an error probability as small as possible. However, the theoretical curve indicates that a smaller threshold than 12 dB could be enough to assess the learning capability from the subject's recorded EEG signals. Thus, for example, a 10 dB drop achieves about 90% of MSE relative reduction as defined in Section 3.3 and Fig. 1. This corresponds to a training set size of N=32 as indicated in Fig. 5. We can see that the empirical probability of error is significantly greater in subject 1, but the learning capability can be sufficiently deduced in both subjects by only considering the interval from N=12 to N=32. Thus, three BT implementations could be enough to train the classifier (remember that every BT is formed by 10 trials, every one providing a feature vector per class), instead of the five BT implementations if the ten-

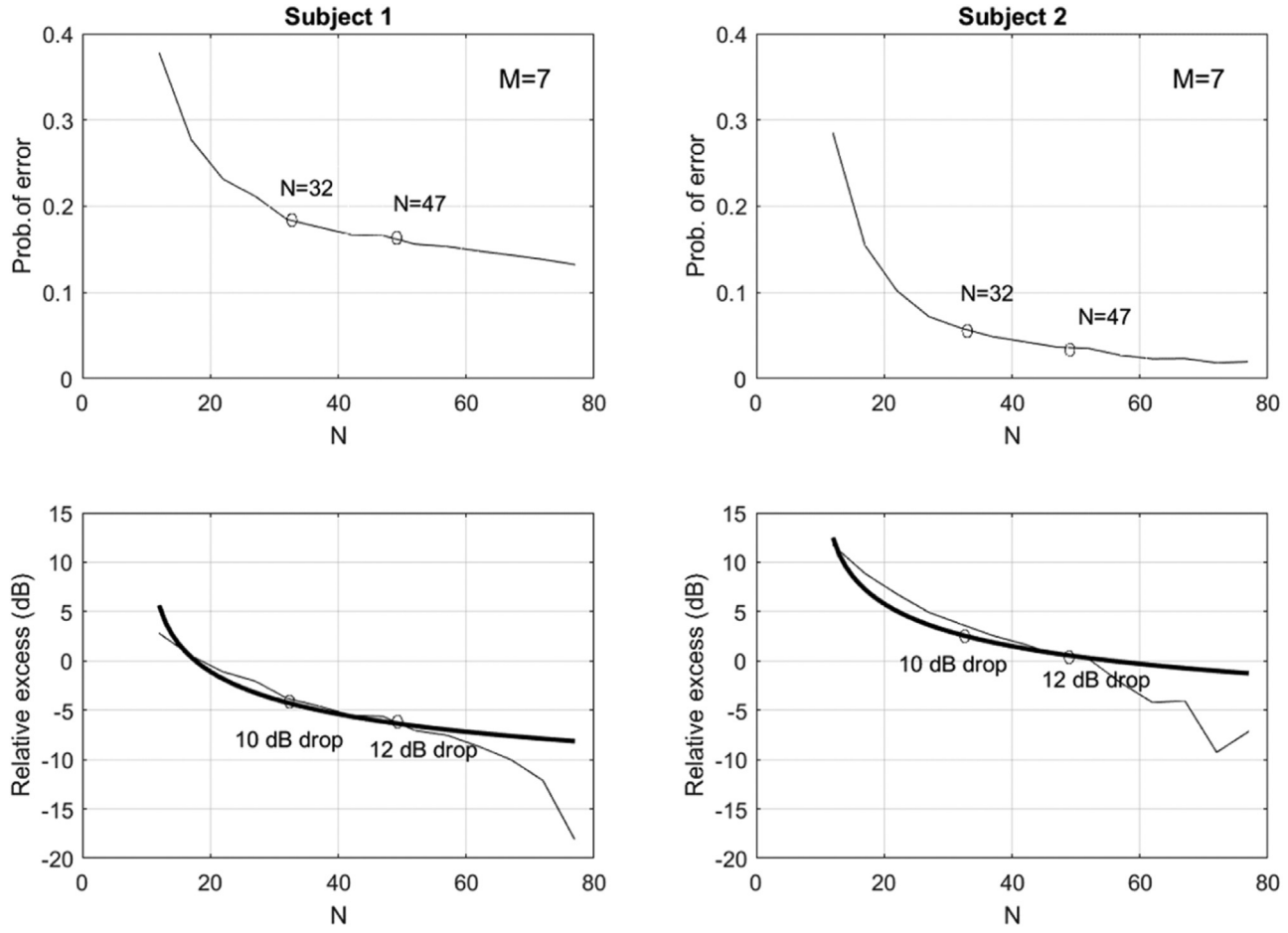

Fig. 5. Estimated probability of error (top), relative excess of error probability and theoretical curves (bottom) for two different subjects of the real data experiments.

tative threshold of 12 dB were considered. This is very significant both from the point of view of time consumption and of the subject's fatigue. In summary, the theoretical curve provides a guide to alleviate the task of defining an appropriate training set size in a practical setting.

6. Conclusion

We have derived a theoretical curve which can help to fit the appropriate value of the training set size. The derivation has been based on an indirect (proxy) approach where the *MSE* convergence of the statistic corresponding to the true class or subclass has been analyzed instead of the direct (intractable) analysis of convergence of the error probability. First, a multivariate Gaussian model has been assumed and then extended to arbitrary parametric models.

Given a particular model of the feature data distribution, the Bayes error rate is the lowest possible error rate for any classifier applied to those data. It can be reached only for consistent estimates, i.e. perfect model estimation when the training set size tends to infinity. While the Bayes error rate fully depends on the model, e.g. lower separable classes mean highest Bayes error rates, the excess of the error probability with respect to the Bayes error rate has been demonstrated to be proportional to the derived proxy curve. Thus, given the feature space dimension, the theoretical curve provides an estimate of the reduction in the excess of error probability as the training set size increases. In a practical setting, this may be useful to define a tentative value for the training set size. This value can be refined by considering the context of the particular application, as we have illustrated in the real data experiment of the section above. Thus, the tedious and costly search for an appropriate training set size is alleviated. In the experiments, we have shown the general validity of the theoretical curve in a variety of simulated models as well as in a real data example. This is consistent with the fact that the proxy curve depends only on the training set size and the feature space dimension, but not on the data model distribution.

Several matters could be considered for improvement in future research. Firstly, some possible knowledge about the model parameters could be incorporated into the analysis to obtain theoretical curves that better match those particular models. For example, some parameters could be assumed known, e.g. correlation matrices are diagonals, and/or equal for all classes, or some knowledge about prior probabilities could be considered [38]. Also, other models different from strictly Gaussian ones, e.g. imprecise Gaussian [39], or different from Gaussian mixtures, e.g. Independent Component Analysis mixtures [35] may be assumed, though they will probably be intractable in most cases. Moreover, we have considered that features are continuous random variables, however the discrete case [38,40] may be of interest in some application domains, but the analysis should be substantially different. Finally, note that in this work, the model parameters are estimated from the training set using closed expressions like (12). The extension of the results to heuristic optimization methods such as bioinspired ones [41,42] is not obvious. However, this work could be useful to define an appropriate training set size to achieve a good starting point for the algorithms if a parametric model is assumed to obtain the initial estimates of the parameters.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

Data will be made available on request.

Acknowledgments

Grant TEC2017-84743-P funded by MCIN/AEI/10.13039/501100011033 and by the European Union.

Appendix A

From (13) we may write:

$$E\left[\tilde{\hat{z}}_l\right] = -\frac{1}{2}E\left[\ln\left|\tilde{\hat{\mathbf{C}}}_l\right|\right] - \frac{1}{2}E\left[\left(\tilde{\mathbf{x}} - \tilde{\hat{\mathbf{b}}}_l\right)^{T\sim -1}\hat{\mathbf{C}}_l\left(\tilde{\mathbf{x}} - \tilde{\hat{\mathbf{b}}}_l\right)\right] + \ln P_l. \tag{A1}$$

For $N_l > M$ the random matrix, $(N_l-1)\tilde{\tilde{\mathbf{C}}}_l$ follows a Wishart distribution $(N_l-1)\tilde{\tilde{\mathbf{C}}}_l \sim W_M(\mathbf{C}_l, N_l-1)$[25], then [26]:

$$\begin{aligned} &E\left[\ln\left|(N_l-1)\tilde{\tilde{\mathbf{C}}}_l\right|\right] = \psi_M\left(\tfrac{N_l-1}{2}\right) + M\ln 2 + \ln|\mathbf{C}_l| \quad \Rightarrow \\ &\Rightarrow E\left[\ln\left|\tilde{\tilde{\mathbf{C}}}_l\right|\right] = \psi_M\left(\tfrac{N_l-1}{2}\right) + M\ln\tfrac{2}{N_l-1} + \ln|\mathbf{C}_l| \end{aligned}, \tag{A2}$$

where $\psi_M(\cdot)$is the multivariate digamma function. On the other hand:

$$E_{\tilde{\hat{\mathbf{C}}}_l}\left[\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)^T \tilde{\tilde{\mathbf{C}}}_l^{-1}\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)/\mathbf{x}, \hat{\mathbf{b}}_l\right] = \left(\mathbf{x} - \hat{\mathbf{b}}_l\right)^T E_{\tilde{\hat{\mathbf{C}}}_l}\left[\tilde{\tilde{\mathbf{C}}}_l^{-1}\right]\left(\mathbf{x} - \hat{\mathbf{b}}_l\right), \tag{A3}$$

(we have used a subindex to indicate over which random variable the expectation is taken, and we will keep this notation where required throughout the paper). Notice that $\hat{\mathbf{b}}_l$and $\tilde{\tilde{\mathbf{C}}}_l$are independent because the covariance of a linear form with a quadratic form of a multivariate Gaussian variable is zero due to the cancellation of third order moments ([27], page 201). So, conditional to $\hat{\mathbf{b}}_l$, matrix $\frac{1}{N_l-1}\tilde{\tilde{\mathbf{C}}}_l^{-1}$still follows an inverse Wishart distribution $\frac{1}{N_l-1}\tilde{\tilde{\mathbf{C}}}_l^{-1} \sim W_M^{-1}(\mathbf{C}_l^{-1}, N_l-1)$ [28], then for $N_l > M+2$:

$$E_{\tilde{\hat{\mathbf{C}}}_l}\left[\tilde{\tilde{\mathbf{C}}}_l^{-1}\right] = \frac{N_l-1}{N_l-M-2}\mathbf{C}_l^{-1}. \tag{A4}$$

On the other hand, $\tilde{\tilde{\mathbf{b}}}_l \sim N(\mathbf{b}_l, \frac{1}{N_l}\mathbf{C}_l)$ [25], and $\tilde{\mathbf{x}} \sim N(\mathbf{b}_l, \mathbf{C}_l)$. But $\tilde{\tilde{\mathbf{b}}}_l$and $\tilde{\mathbf{x}}$ are independent because the instance being tested will not be included in the training set to avoid overfitting, so $(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l) \sim N(\mathbf{0},\ \mathbf{C}_l + \frac{1}{N_l}\mathbf{C}_l)$. Then, from (A3), (A4), the law of total expectation and [29]:

$$\begin{aligned} &E\left[\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)^T \tilde{\tilde{\mathbf{C}}}_l^{-1}\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)\right] = E_{\tilde{\mathbf{x}},\tilde{\tilde{\mathbf{b}}}_l}\left[\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)^T E_{\tilde{\hat{\mathbf{C}}}_l}\left[\tilde{\tilde{\mathbf{C}}}_l^{-1}\right]\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)\right] = \\ &= trace\left[E_{\tilde{\hat{\mathbf{C}}}_l^{-1}}\left[\tilde{\tilde{\mathbf{C}}}_l^{-1}\right]\left(\mathbf{C}_l + \tfrac{1}{N_l}\mathbf{C}_l\right)\right] = trace\left[\tfrac{N_l-1}{N_l-M-2}\mathbf{C}_l^{-1}\left(\mathbf{C}_l + \tfrac{1}{N_l}\mathbf{C}_l\right)\right] = \\ &= \tfrac{N_l-1}{N_l-M-2}\left(1+\tfrac{1}{N_l}\right)trace[\mathbf{I}] = \tfrac{N_l-1}{N_l-M-2}\left(1+\tfrac{1}{N_l}\right)M \end{aligned} \tag{A5}$$

Finally,

$$\begin{aligned} E\left[\tilde{\hat{z}}_l\right] = &-\frac{1}{2}\psi_M\left(\frac{N_l-1}{2}\right) - \frac{1}{2}M\ln\frac{2}{N_l-1} - \frac{1}{2}\ln|\mathbf{C}_l| \\ &-\frac{1}{2}\frac{N_l-1}{N_l-M-2}\left(1+\frac{1}{N_l}\right)M + \ln P_l \end{aligned} \tag{A6}$$

provided that $N_l > M+2$.

Appendix B

The law of total variance allows us to write:

$$\begin{aligned} &\mathrm{var}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)\right] \\ &= E_{\tilde{x},\tilde{\hat{b}}_l}\left[\mathrm{var}_{\tilde{\hat{C}}_l}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)/x, \hat{b}_l\right]\right] \\ &+\mathrm{var}_{\tilde{x},\tilde{\hat{b}}_l}\left[E_{\tilde{\hat{C}}_l}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)/x, \hat{b}_l\right]\right] \end{aligned}. \tag{B1}$$

But the quadratic form of an inverse Wishart matrix, properly normalized, follows an inverse chi-square distribution $\tilde{\mathbf{M}}^{-1} \sim W_Q^{-1}(\mathbf{M}_l^{-1}, P) \Rightarrow \mathbf{a}^T\tilde{\mathbf{M}}^{-1}\mathbf{a}/\mathbf{a}^T\mathbf{M}^{-1}\mathbf{a} \sim inv\chi^2_{P-Q+1}$ [27].

Considering that $x \sim inv\chi^2_\nu \Rightarrow \mathrm{var}(x) = 2/(\nu-2)^2(\nu-4) \quad \nu > 4$, we can write:

$$\begin{aligned} &\tfrac{1}{N_l-1}\tilde{\hat{C}}_l^{-1} \sim W_M^{-1}\left(C_l^{-1}, N_l-1\right) \\ &\Rightarrow \mathrm{var}_{\tilde{\hat{C}}_l}\left[\frac{\frac{1}{N_l-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)}{\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T C_l^{-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)}/x, \hat{b}_l\right] = \tfrac{2}{(N_l-M-2)^2(N_l-M-4)} \end{aligned}. \tag{B2}$$

provided that$N_l > M+4$

Therefore,

$$\begin{aligned} &E_{\tilde{x},\tilde{\hat{b}}_l}\left[\mathrm{var}_{\tilde{\hat{C}}_l}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)/x, \hat{b}_l\right]\right] \\ &= \tfrac{(N_l-1)^2}{(N_l-M-2)^2(N_l-M-4)}E_{\tilde{x},\tilde{\hat{b}}_l}\left[\left(\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T C_l^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)\right)^2\right] \end{aligned}. \tag{B3}$$

But,

$$\begin{aligned} &E_{\tilde{\mathbf{x}},\tilde{\tilde{\mathbf{b}}}_l}\left[\left(\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)^T \mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)\right)^2\right] = \\ &= E^2_{\tilde{\mathbf{x}},\tilde{\tilde{\mathbf{b}}}_l}\left[\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)^T \mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)\right] + \mathrm{var}_{\tilde{\mathbf{x}},\tilde{\tilde{\mathbf{b}}}_l}\left[\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)^T \mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}} - \tilde{\tilde{\mathbf{b}}}_l\right)\right] = \\ &= trace\left[\mathbf{C}_l^{-1}\left(\mathbf{C}_l\left(1+\tfrac{1}{N_l}\right)\right)\right] + 2trace\left[\mathbf{C}_l^{-1}\left(\mathbf{C}_l\left(1+\tfrac{1}{N_l}\right)\right)\mathbf{C}_l^{-1}\left(\mathbf{C}_l\left(1+\tfrac{1}{N_l}\right)\right)\right] = \\ &= M\left(1+\tfrac{1}{N_l}\right) + 2M\left(1+\tfrac{1}{N_l}\right)^2 \end{aligned} \tag{B4}$$

Therefore,

$$\begin{aligned} &E_{\tilde{x},\tilde{\hat{b}}_l}\left[\mathrm{var}_{\tilde{\hat{C}}_l}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)/x, \hat{b}_l\right]\right] \\ &= \tfrac{(N_l-1)^2}{(N_l-M-2)^2(N_l-M-4)}\left(M\left(1+\tfrac{1}{N_l}\right) + 2M\left(1+\tfrac{1}{N_l}\right)^2\right) \end{aligned}. \tag{B5}$$

We still have to compute the second term of (B1), but considering (A4), we can write

$$\begin{aligned} &\mathrm{var}_{\tilde{x},\tilde{\hat{b}}_l}\left[E_{\tilde{\hat{C}}_{ll}^{-1}}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)/x, \hat{b}_l\right]\right] \\ &= \mathrm{var}_{\tilde{x},\tilde{\hat{b}}_l}\left[\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T \tfrac{N_l-1}{N_l-M-2}C_l^{-1}\left(\tilde{x} - \tilde{\hat{b}}_l\right)^T\right] \\ &= \left(\tfrac{N_l-1}{N_l-M-2}\right)^2\left(trace\left[C_l^{-1}\left(C_l\left(1+\tfrac{1}{N_l}\right)\right)\right]\right) = \left(\tfrac{N_l-1}{N_l-M-2}\right)^2 2M\left(1+\tfrac{1}{N_l}\right) \end{aligned} \tag{B6}$$

Adding (B5) and (B6), we can write

$$\begin{aligned}&\mathrm{var}\left[\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)\right]\\&=\frac{(N_l-1)^2}{(N_l-M-2)^2(N_l-M-4)}\left(M\left(1+\tfrac{1}{N_l}\right)+2M\left(1+\tfrac{1}{N_l}\right)^2\right)\cdot\\&\quad+\left(\tfrac{N_l-1}{N_l-M-2}\right)^2 2M\left(1+\tfrac{1}{N_l}\right)\end{aligned}\tag{B7}$$

provided that$N_l > M+4$.

Appendix C

Let us consider de eigendecomposition of the sample covariance matrix

$$\tilde{\hat{\mathbf{C}}}_l=\sum_{m=1}^{M}\tilde{\hat{\lambda}}_m\tilde{\hat{\mathbf{u}}}_m\tilde{\hat{\mathbf{u}}}_m^T\qquad\tilde{\hat{\lambda}}_m>0\quad\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{u}}}_{m'}=\begin{cases}1 & m=m'\\0 & m\neq m'\end{cases},\tag{C1}$$

where $\{\tilde{\hat{\mathbf{u}}}_m\}$ are the eigenvectors of matrix $\tilde{\hat{\mathbf{C}}}_l$and $\{\tilde{\hat{\lambda}}_m\}$the corresponding eigenvalues. Then we can write:

$$\begin{aligned}&\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)\\&=\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\left(\sum_{m=1}^{M}\tilde{\hat{\lambda}}_m^{-1}\hat{u}_m\hat{u}_m^T\right)\left(\tilde{x}-\tilde{\hat{b}}_l\right)\\&=\sum_{m=1}^{M}\tilde{\hat{\lambda}}_m^{-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\hat{u}_m\hat{u}_m^T\left(\tilde{x}-\tilde{\hat{b}}_l\right)\\&\ln\left|\tilde{\hat{C}}_l\right|=\sum_{m=1}^{M}\ln\tilde{\hat{\lambda}}_m\end{aligned}\tag{C2}$$

And so

$$\begin{aligned}&\mathrm{cov}\left[\ln\left|\tilde{\hat{C}}_l\right|,\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\hat{b}}_l\right)\right]\\&=\sum_{m=1}^{M}\sum_{m'=1}^{M}\mathrm{cov}\left[\ln\tilde{\hat{\lambda}}_{m'},\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{u}}_m^T\left(\tilde{x}-\tilde{\hat{b}}_l\right)\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{u}}_m\right].\end{aligned}\tag{C3}$$

Notice that $\mathrm{cov}[\ln\tilde{\hat{\lambda}}_{m'},\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l)(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l)^T\tilde{\hat{\mathbf{u}}}_m]$ will be zero if $\mathrm{cov}[\tilde{\hat{\lambda}}_{m'},\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l)(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l)^T\tilde{\hat{\mathbf{u}}}_m]$ is zero. But:

$$\begin{aligned}&\mathrm{cov}\left[\tilde{\hat{\lambda}}_{m'},\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]=\\&=E\left[\tilde{\hat{\lambda}}_{m'}\cdot\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]\\&\quad-E\left[\tilde{\hat{\lambda}}_{m'}\right]\cdot E\left[\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]\end{aligned}.\tag{C4}$$

Now, let us take into account that $\tilde{\hat{\lambda}}_m=\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m$, so that we can write

$$E\left[\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]=E\left[\frac{\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}\right].\tag{C5}$$

Then, we can apply the law of total expectation to compute (C5). Previously, notice that a realization $\tilde{\hat{\mathbf{C}}}_l$of the sample matrix implies a realization of all its eigenvectors. However, we will assume the approximation that given a realization $\hat{\mathbf{u}}_m$of the m-th eigenvector, $\tilde{\hat{\mathbf{C}}}_l$ is still a random Wishart matrix $(N_l-1)\tilde{\hat{\mathbf{C}}}_l\sim W_M(\mathbf{C}_l,N_l-1)$. This is a reasonable approximation for large M, since this is the total number of eigenvalues and eigenvectors defining the matrix eigendecomposition in (C1). Then we can write

$$E_{\tilde{\mathbf{x}},\tilde{\hat{\mathbf{b}}}_l}\left[\frac{\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}/\tilde{\hat{\mathbf{C}}}_l,\tilde{\hat{\mathbf{u}}}_m\right]=\frac{\tilde{\hat{\mathbf{u}}}_m^TE_{\tilde{\mathbf{x}},\tilde{\hat{\mathbf{b}}}_l}\left[\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\right]\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}=\left(1+\tfrac{1}{N}\right)\frac{\tilde{\hat{\mathbf{u}}}_m^T\mathbf{C}_l\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}.\tag{C6}$$

But

$$\left[\frac{\tilde{\hat{\mathbf{u}}}_m^T\mathbf{C}_l\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T(N-1)\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}/\hat{\mathbf{u}}_m\right]\sim inv\chi^2_{N-1}\;\Rightarrow\;E_{\tilde{\hat{\mathbf{C}}}_l}\left[\frac{\tilde{\hat{\mathbf{u}}}_m^T\mathbf{C}_l\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}/\hat{\mathbf{u}}_m\right]=\frac{N-1}{N-3}.\tag{C7}$$

Therefore

$$\begin{aligned}&E\left[\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]\\&=E_{\tilde{\hat{\mathbf{u}}}_m}\left[\left(1+\frac{1}{N}\right)\frac{N-1}{N-3}\right]=\left(1+\frac{1}{N}\right)\frac{N-1}{N-3}\end{aligned}.\tag{C8}$$

Let us now consider the first term in (C4). We can proceed in a similar manner, in this case assuming that given both a realization $\hat{\mathbf{u}}_m$of the mth eigenvector, and a realization$\hat{\lambda}_{m'}$ of the m'-th eigenvalue $\tilde{\hat{\mathbf{C}}}_l$is still a random Wishart matrix. Therefore for $m\neq m'$ we may write:

$$\begin{aligned}&E\left[\tilde{\hat{\lambda}}_{m'}\cdot\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]=E\left[\tilde{\hat{\lambda}}_{m'}\frac{\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}\right]\\&E_{\tilde{\mathbf{x}},\tilde{\hat{\mathbf{b}}}_l}\left[\tilde{\hat{\lambda}}_{m'}\frac{\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}/\tilde{\hat{\mathbf{C}}}_l,\tilde{\hat{\mathbf{u}}}_m,\tilde{\hat{\lambda}}_{m'}\right]=\tilde{\hat{\lambda}}_{m'}\left(1+\tfrac{1}{N}\right)\frac{\tilde{\hat{\mathbf{u}}}_m^T\mathbf{C}_l\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}\end{aligned}$$

$$\begin{aligned}&E_{\tilde{\hat{\mathbf{C}}}_l}\left[\tilde{\hat{\lambda}}_{m'}\left(1+\tfrac{1}{N}\right)\frac{\tilde{\hat{\mathbf{u}}}_m^T\mathbf{C}_l\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}/\hat{\mathbf{u}}_m,\tilde{\hat{\lambda}}_{m'}\right]=\tilde{\hat{\lambda}}_{m'}\left(1+\tfrac{1}{N}\right)\tfrac{N-1}{N-3}\\&E\left[\tilde{\hat{\lambda}}_{m'}\cdot\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]\\&\quad=E_{\hat{\mathbf{u}}_m,\tilde{\hat{\lambda}}_{m'}}\left[\tilde{\hat{\lambda}}_{m'}\left(1+\tfrac{1}{N}\right)\tfrac{N-1}{N-3}\right]=E\left[\tilde{\hat{\lambda}}_{m'}\right]\left(1+\tfrac{1}{N}\right)\tfrac{N-1}{N-3}\end{aligned}.\tag{C9}$$

And both terms in (C4) cancel out so that$\mathrm{cov}[\tilde{\hat{\lambda}}_{m'},\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l)(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l)^T\tilde{\hat{\mathbf{u}}}_m]=0\quad m\neq m'$. For $m=m'$, the second term in (C3) is simply $E[\tilde{\hat{\lambda}}_m](1+\frac{1}{N})\frac{N-1}{N-3}$ and the second term becomes

$$\begin{aligned}&E\left[\tilde{\hat{\lambda}}_{m'}\cdot\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{u}}_m^T\left(\tilde{x}-\tilde{\hat{b}}_l\right)\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{u}}_m\right]=E\left[\tilde{\hat{u}}_m^T\left(\tilde{x}-\tilde{\hat{b}}_l\right)\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{u}}_m\right]=\\&=E_{\tilde{\hat{u}}_m}\left[E_{\tilde{x},\tilde{\hat{b}}_l}\left[\tilde{\hat{u}}_m^T\left(\tilde{x}-\tilde{\hat{b}}_l\right)\left(\tilde{x}-\tilde{\hat{b}}_l\right)^T\tilde{\hat{u}}_m/\hat{u}_m\right]\right]=\left(1+\tfrac{1}{N}\right)E_{\tilde{\hat{u}}_m}\left[\tilde{\hat{u}}_m^TC_l\tilde{\hat{u}}_m/\hat{u}_m\right]\end{aligned}.\tag{C10}$$

But,

$$\left[\frac{\tilde{\hat{\mathbf{u}}}_m^T(N-1)\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m}{\tilde{\hat{\mathbf{u}}}_m^T\mathbf{C}_l\tilde{\hat{\mathbf{u}}}_m}/\hat{\mathbf{u}}_m\right]\sim\chi^2_{N-1}\;\Rightarrow\;E_{\tilde{\hat{\mathbf{C}}}_l}\left[\tilde{\hat{\mathbf{u}}}_m^T\tilde{\hat{\mathbf{C}}}_l\tilde{\hat{\mathbf{u}}}_m/\hat{\mathbf{u}}_m\right]=\hat{\mathbf{u}}_m^T\mathbf{C}_l\hat{\mathbf{u}}_m.\tag{C11}$$

Then, $E_{\tilde{\hat{u}}_m}[\tilde{\hat{u}}_m^TC_l\tilde{\hat{u}}_m/\hat{u}_m]=E_{\tilde{\hat{u}}_m}[E_{\tilde{\hat{C}}_l}[\tilde{\hat{u}}_m^T\tilde{\hat{C}}_l\tilde{\hat{u}}_m/\hat{u}_m]]=E[\tilde{\hat{\lambda}}_m]$ and

$$\begin{aligned}&\mathrm{cov}\left[\tilde{\hat{\lambda}}_m,\tilde{\hat{\lambda}}_m^{-1}\tilde{\hat{\mathbf{u}}}_m^T\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)\left(\tilde{\mathbf{x}}-\tilde{\hat{\mathbf{b}}}_l\right)^T\tilde{\hat{\mathbf{u}}}_m\right]\\&=\left(1+\tfrac{1}{N}\right)E\left[\tilde{\hat{\lambda}}_m\right]\left(1-\tfrac{N-1}{N-3}\right),\end{aligned}\tag{C12}$$

which for practical values of N will be close to zero.

Appendix D

Taking into account that $\mathrm{cov}[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l),\ \ln|\tilde{\tilde{\mathbf{C}}}_l|]=0$, the law of total covariance can be applied in a similar form to the law of total variance in (B1), namely:

$$\begin{aligned}\mathrm{cov}\left[\tilde{z}_l\cdot\tilde{\tilde{z}}_l\right] &= \tfrac{1}{4}\mathrm{cov}\left[(\tilde{x}-b_l)^TC_l^{-1}(\tilde{x}-b_l),\left(\tilde{x}-\tilde{\tilde{b}}_l\right)^T\tilde{\tilde{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\tilde{b}}_l\right)\right]\\ &= \tfrac{1}{4}E_{\tilde{x},\tilde{\tilde{b}}_l}\left[\mathrm{cov}_{\tilde{\tilde{C}}_l}\left[(\tilde{x}-b_l)^TC_l^{-1}(\tilde{x}-b_l),\left(\tilde{x}-\tilde{\tilde{b}}_l\right)^T\tilde{\tilde{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\tilde{b}}_l\right)/x,\hat{b}_l\right]\right]\\ &+\tfrac{1}{4}\mathrm{cov}_{\tilde{x},\tilde{\tilde{b}}_l}\left[E_{\tilde{\tilde{C}}_l}\left[(\tilde{x}-b_l)^TC_l^{-1}(\tilde{x}-b_l)/x,\hat{b}_l\right],\ E_{\tilde{\tilde{C}}_l}\left[\left(\tilde{x}-\tilde{\tilde{b}}_l\right)^T\tilde{\tilde{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\tilde{b}}_l\right)/x,\hat{b}_l\right]\right]\end{aligned} \tag{D1}$$

The first term in (D1) vanishes because, conditional to $\tilde{\mathbf{x}}=\mathbf{x}$, $(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\tilde{\tilde{\mathbf{C}}}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l)$does not depend on $\tilde{\tilde{\mathbf{C}}}_l$. Regarding the second term, we have (remember (A4)):

$$\begin{aligned}E_{\tilde{\tilde{C}}_l}\left[(\tilde{x}-b_l)^TC_l^{-1}(\tilde{x}-b_l)/x,\hat{b}_l\right] &= (x-b_l)^TC_l^{-1}(x-b_l)\\ E_{\tilde{\tilde{C}}_l}\left[\left(\tilde{x}-\tilde{\tilde{b}}_l\right)^T\tilde{\tilde{C}}_{ll}^{-1}\left(\tilde{x}-\tilde{\tilde{b}}_l\right)/x,\hat{b}_l\right] &= \left(x-\hat{b}_l\right)^T\tfrac{N_l-1}{N_l-M-2}C_l^{-1}\left(x-\hat{b}_l\right)^T.\end{aligned} \tag{D2}$$

So [29],

$$\begin{aligned}\mathrm{cov}\left[\tilde{z}_l\cdot\hat{\tilde{z}}_l\right] &= \tfrac{1}{4}\tfrac{N_l-1}{N_l-M-2}\cdot\mathrm{cov}_{\tilde{\mathbf{x}},\tilde{\tilde{\mathbf{b}}}_l}\\ &\times\left[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l),\ \left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\right].\end{aligned} \tag{D3}$$

provided that$N_l > M+2$.

We can apply the law of total covariance again to compute the above covariance:

$$\begin{aligned}&\mathrm{cov}_{\tilde{\mathbf{x}},\tilde{\tilde{\mathbf{b}}}_l}\left[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l),\ \left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\right] =\\ &= E_{\tilde{\tilde{\mathbf{b}}}_l}\left[\mathrm{cov}_{\tilde{\mathbf{x}}}\left[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l),\ \left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T/\hat{\mathbf{b}}_l\right]\right]+\\ &+\mathrm{cov}_{\tilde{\tilde{\mathbf{b}}}_l}\left[E_{\tilde{\mathbf{x}}}\left[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l)/\hat{\mathbf{b}}_l\right],\ E_{\tilde{\mathbf{x}}}\left[\left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T/\hat{\mathbf{b}}_l\right]\right]\end{aligned} \tag{D4}$$

Let us define $\tilde{\mathbf{w}}=\tilde{\mathbf{x}}-\mathbf{b}_land\hat{\mathbf{v}}=\mathbf{b}_l-\hat{\mathbf{b}}_l$ so that $\tilde{\mathbf{x}}-\hat{\mathbf{b}}_l=\tilde{\mathbf{w}}+\hat{\mathbf{v}}$. Then we can operate with the first term in (D4):

$$\begin{aligned}&\mathrm{cov}_{\tilde{\mathbf{x}}}\left[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l),\ \left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T\mathbf{C}_l^{-1}\left(\tilde{\mathbf{x}}-\tilde{\tilde{\mathbf{b}}}_l\right)^T/\hat{\mathbf{b}}_l\right]\\ &= \mathrm{cov}_{\tilde{\mathbf{w}}}\left[\tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}},\ \left(\tilde{\mathbf{w}}+\hat{\mathbf{v}}\right)^T\mathbf{C}_l^{-1}\left(\tilde{\mathbf{w}}+\hat{\mathbf{v}}\right)^T\right] =\\ &= \mathrm{cov}_{\tilde{\mathbf{w}}}\left[\tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}},\ \tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}}^T\right]+\mathrm{cov}_{\tilde{\mathbf{w}}}\left[\tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}},\ \hat{\mathbf{v}}^T\mathbf{C}_l^{-1}\hat{\mathbf{v}}^T\right]\\ &\quad+2\mathrm{cov}_{\tilde{\mathbf{w}}}\left[\tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}},\ \tilde{\mathbf{w}}\mathbf{C}_l^{-1}\hat{\mathbf{v}}^T\right]\end{aligned} \tag{D5}$$

The second term is zero because $\hat{\mathbf{v}}^T\mathbf{C}_l^{-1}\hat{\mathbf{v}}^T$does not depend on$\tilde{\mathbf{w}}$. The third term also vanishes because it is the covariance of a quadratic form with a linear form of a multivariate Gaussian variable ([27], page 201). So, the first term in (D4) is given by:

$$E_{\tilde{\tilde{\mathbf{b}}}_l}\left[\mathrm{cov}_{\tilde{\mathbf{w}}}\left[\tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}},\ \tilde{\mathbf{w}}^T\mathbf{C}_l^{-1}\tilde{\mathbf{w}}^T\right]\right] = E_{\tilde{\tilde{\mathbf{b}}}_l}\left[2trace\left[\mathbf{C}_l^{-1}\mathbf{C}_l\mathbf{C}_l^{-1}\mathbf{C}_l\right]\right] = 2M. \tag{D6}$$

Moreover, the second term in (D4) is zero because $E_{\tilde{\mathbf{x}}}[(\tilde{\mathbf{x}}-\mathbf{b}_l)^T\mathbf{C}_l^{-1}(\tilde{\mathbf{x}}-\mathbf{b}_l)/\hat{\mathbf{b}}_l] = trace[\mathbf{C}_l^{-1}\mathbf{C}_l] = M$is constant. Therefore, returning to (D3) and considering (D4)–(D6), we may write:

$$\mathrm{cov}\left[\tilde{z}_l\cdot\hat{\tilde{z}}_l\right] = \frac{1}{4}\frac{N_l-1}{N_l-M-2}2M. \tag{D7}$$

provided that $N_l > M+2$.

References

[1] J. Jorge, W. van der Zwaag, P. Figueiredo, EEG–fMRI integration for the study of human brain function, NeuroImage 102 (2014) 24–34.
[2] A. Malhotra, M. Younes, S.T. Kuna, R. Benca, Performance of an automated polysomnography scoring system versus computer-assisted manual scoring, Sleep 36 (4) (2013) 573–582.
[3] A. Salazar, A. Rodríguez, N. Vargas, L. Vergara, On training road surface classifiers by data augmentation Special Issue in Novel Methods and Technologies for Intelligent Vehicles, Appl. Sci. 12 (7) (2022) 3423, doi:10.3390/app12073423.
[4] S.J. Raudys, A.K. Jain, Small sample size effects in statistical pattern recognition: recommendations for practitioners, IEEE Trans. Pattern Anal. Mach. Intell. 13 (1991) 252–264.
[5] M. Sordo, Q. Zeng, J.L. Oliveira, V. Maojo, F. Martín-Sánchez, A. Sousa, On sample size and classification accuracy: a performance comparison, in: Biological and Medical Data Analysis, Springer, Berlín, 2005, pp. 193–201.
[6] C. Beleites, U. Neugebauer, T. Bocklitz, C. Krafft, J. Popp, Sample size planning for classification models, Anal. Chim. Acta 760 (2013) 25–33.
[7] A. Alwosheel, S. van Cranenburgh, C.G. Chorus, Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis, J. Choice Model. 28 (2018) 167–182.
[8] J. Cho, K. Lee, E. Shin, G. Choy, S. Do, How much data is needed to train a medical image deep learning system to achieve necessary high accuracy?. https://doi.org/10.48550/arXiv.1511.06348.
[9] Y. Wahba, E. ElSalamouny, G. ElTawel, Estimating the sample size for training intrusion detection systems, Int. J. Comput. Netw. Inf. Secur. 12 (2017) 1–10.
[10] R.L. Figueroa, Q. Zeng-Treitler, S. Kandula, L.H. Ngo, Predicting sample size required for classification performance, BMC Med. Inform. Decis. Mak. 12 (2012) 1–10.
[11] S. Raudys, R.P.W. Duin, Expected classification error of the fisher linear classifier with pseudo-inverse covariance matrix, Pattern Recognit. Lett. 19 (1998) 385–392.
[12] A. Zollanvari, M.G. Genton, On Kolmogorov asymptotics of estimators of the misclassification error rate in linear discriminant analysis, Sankhya Ser. A 75 (2013) 30–326.
[13] L. Rueda, A one-dimensional analysis for the probability of error of linear classifiers for normally distributed classes, Pattern Recognit. 38 (2005) 1197–1207.
[14] A. Zollanvari, E.R. Dougherty, Moments and root-mean-square error of the Bayesian MMSE estimator of classification error in the Gaussian model, Pattern Recognit. 47 (2014) 2178–2192.
[15] F. Nielsen, Generalized Bhattacharyya and Chernoff upper bounds on Bayes error using quasi-arithmetic means, Pattern. Recognit. Lett. 42 (2014) 25–34.
[16] T. Bodnar, S. Mazur, E. Ngailo, N. Parolya, Discriminant analysis in small and large dimensions, Theory Probab. Math. Stat. 100 (2020) 21–41.
[17] F. Wyman, D. Young, D. Turner, A comparison of asymptotic error rate expansions for the sample linear discriminant function, Pattern Recognit. 23 (1990) 775–783.
[18] M.M.H. El Ayadi, M.S. Kamel, F. Karray, Toward a tight upper bound for the error probability of the binary gaussian classification problem, Pattern Recognit. 41 (2008) 2120–2132.
[19] V.B. Berikov, An approach to the evaluation of the performance of a discrete classifier, Pattern Recognit. Lett. 23 (2002) 227–233.
[20] V.B. Berikov, A. Litvinenko, The influence of prior knowledge on the expected performance of a classifier, Pattern Recognit. Lett. 24 (2003) 2537–2548.
[21] O. Bousquet, A. Elisseeff, Stability and generalization, J. Mach. Learn. Res. 2 (2002) 499–526.
[22] H.M. Kalayeh, D.A. Landgrebe, Predicting the required number of training samples, IEEE Trans. Pattern Anal. Mach. Intell. 5 (1983) 664–667.
[23] V.N. Vapnik, A.Y. Chervonenkis, V. Vovk, H. Papadopoulos, A. Gammerman, On the uniform convergence of relative frequencies of events to their probabilities, in: Measures of Complexity, Springer, Cham, 2015, pp. 11–30.
[24] J. Zubek, D.M. Plewczynski, Complexity curve: a graphical measure of data complexity and classifier performance, Peer J. Comput. Sci. 2 (2016) e76.
[25] C. Chatfield, A.J. Collins, Introduction to Multivariate Analysis, Chapman and Hall, London, 1980.
[26] C.M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag, Berlin, Heidelberg, 2006.
[27] D.A. Harville, Linear Models and the Relevant Distributions and Matrix Algebra, Chapman and Hall, CRC, London, 2018.
[28] V. Kanti, J.T. Mardia, B.J.M. Kent, Multivariate Analysis, Academic Press, London, 1979.
[29] A.C. Rencher, G.B. Schaalje, Linear Models in Statistics, 2nd ed., John Wiley & Sons, Hoboken, N.J, 2008.
[30] J.Q. Li, A.R. Barron, Mixture density estimation, Adv. Neural Inf. Process. Syst. 12 (2000) 279–285.
[31] D. Kim, B. Seo, Assessment of the number of components in Gaussian mixture models in the presence of multiple local maximizers, J. Multivar. Anal. 125 (2014) 100–120.
[32] G.J. McLachlan, S. Suren, On the number of components in a Gaussian mixture model, WIREs Data Min. Knowl. Discov. 4 (2014) 341–355.
[33] P. Campadelli, E. Casiraghi, C. Ceruti, A. Rozza, Intrinsic dimension estimation: relevant techniques and a benchmark framework, Math. Probl. Eng. (2015) 1–201.

[34] G. Safont, A. Salazar, L. Vergara, E. Gómez, V. Villanueva, Probabilistic distance for mixtures of independent component analyzers, IEEE Trans. Neural Netw. Learn. Syst. 29 (2018) 1161–1173.
[35] G. Safont, A. Salazar, L. Vergara, E. Gómez, V. Villanueva, Multichannel dynamic modeling of non-Gaussian mixtures, Pattern Recognit. 93 (2019) 312–323.
[36] A. Salazar, L. Vergara, G. Safont, Generative adversarial networks and markov random fields for oversampling very small training sets, Expert Syst.Appl. 163 (2021) 113819.
[37] M. Quintana, Spanish multicenter normative studies (Neuronorma project): norms for the abbreviated Barcelona Test, Arch. Clin. Neuropsychol. 26 (2010) 144–157.
[38] A. Broumand, M. Shahrokh, E. Byung-Jun, Y. Edward, E.R. Dougherty, Discrete optimal Bayesian classification with error-conditioned sequential sampling, Pattern Recognit. 48 (2015) 3766–3782.
[39] Y.C. Carranza, S. Destercke, Imprecise Gaussian discriminant classification, Pattern Recognit. 112 (2021) 107739.
[40] S. Sharmin, M. Shoyaib, A. Ahsan, M.A. Hossain, O. Chaed, Simultaneous feature selection and discretization based on mutual information, Pattern Recognit. 91 (2019) 162–174.
[41] J.O. Agushaka, A.E. Ezugwu, L. Abualigah, Dwarf mongoose optimization algorithm, Comput. Methods Appl. Mech. Eng. 391 (2022) 114570.
[42] L. Abualigah, M.A. Elaziz, P. Sumari, Z.W. Geem, A.H. Gandomi, Reptile search algorithm (RSA): a nature-inspired meta-heuristic optimizer, Expert Syst. Appl. 191 (2022) 116158.

Addisson Salazar was awarded a Ph.D. in electrical engineering from the Universitat Politècnica de València in 2011. He is currently a senior researcher at the UPV's Institute of Telecommunications and Multimedia Applications. He has more than 100 papers in statistical signal processing, machine learning, decision fusion and pattern recognition.

Luis Vergara earned a Ph.D. in electrical engineering from Universidad Politécnica de Madrid in 1983. He is a full professor of telecommunications, signal and data processing at the Universitat Politècnica de València. He has more than 250 publications in theoretical and applied problems of signal and data processing and has led many important projects in these fields.

Enrique Vidal is an emeritus professor of computer science at the Universitat Politècnica de València (Spain). He has published more than 250 research papers in the fields of Pattern Recognition, Multimodal Interaction and applications to Language, Speech and Image Processing and has led many important projects in these fields. Dr. Vidal is a member of the IEEE and a fellow of the International Association for Pattern Recognition (IAPR).