Generalized Sparse Discriminant Analysis for Event-Related Potential Classification

Victoria Peterson[a,*], Hugo Leonardo Rufiner[a,b], Ruben Daniel Spies[c]

[a] *Instituto de Investigación en Señales, Sistemas e Inteligencia Computacional, UNL, CONICET, FICH, Ruta Nac. 168, km 472.4, 3000, Santa Fe, Argentina.*
[b] *Facultad de Ingeniería, Universidad Nacional de Entre Ríos, Ruta Prov. 11, km 10, 3100, Oro Verde, Argentina.*
[c] *Instituto de Matemática Aplicada del Litoral, UNL, CONICET, FIQ, Predio Dr. Alberto Cassano del CCT-CONICET-Santa Fe, Ruta Nac. 168, km 0, 3000, Santa Fe, Argentina.*

Abstract

A brain computer interface (BCI) is a system which provides direct communication between the mind of a person and the outside world by using only brain activity (EEG). The event-related potential (ERP)-based BCI problem consists of a binary pattern recognition. Linear discriminant analysis (LDA) is widely used to solve this type of classification problems, but it fails when the number of features is large relative to the number of observations. In this work we propose a penalized version of the sparse discriminant analysis (SDA), called generalized sparse discriminant analysis (GSDA), for binary classification. This method inherits both the discriminative feature selection and classification properties of SDA and it also improves SDA performance through the addition of Kullback-Leibler class discrepancy information. The

[*] Corresponding author

Email addresses: vpeterson@sinc.unl.edu.ar (Victoria Peterson), lrufiner@sinc.unl.edu.ar (Hugo Leonardo Rufiner), rspies@santafe-conicet.gov.ar (Ruben Daniel Spies)

GSDA method is designed to automatically select the optimal regularization parameters. Numerical experiments with two real ERP-EEG datasets show that, on one hand, GSDA outperforms standard SDA in the sense of classification performance, sparsity and required computing time, and, on the other hand, it also yields better overall performances, compared to well-known ERP classification algorithms, for single-trial ERP classification when insufficient training samples are available. Hence, GSDA constitute a potential useful method for reducing the calibration times in ERP-based BCI systems.

Keywords: Brain-Computer Interface, Event-Related Potential, Kullback-Leibler Divergence, Penalization, Sparse Discriminant Analysis

1. Introduction

A brain computer interface (BCI) is a system that measures brain activity and converts it into an artificial output which is able to replace, restore or improve any normal output (neuromuscular or hormonal) used by a person to communicate and control his/her external or internal environment. Thus, BCI can significantly improve the quality of life of people with severe neuromuscular disabilities [1].

Communication between the brain of a person and the outside world can be appropriately established by means of a BCI system based on event-related potentials (ERPs), which are manifestations of neural activity as a consequence of certain infrequent or relevant stimuli. The main reason for using ERP-based BCI are: it is non-invasive, it requires minimal user training and it is quite robust (in the sense that it can be used by more than 90% of people) [2]. One of the main components of such ERPs is the

P300 wave, which is a positive deflection occurring in the scalp-recorded EEG approximately 300 ms after the stimulus has been applied. The P300 wave is unconsciously generated and its latency and amplitude vary between different EEG records of the same person, and even more, between EEG records of different persons [3]. By using the "oddball" paradigm [4] the ERP-based BCI can decode desired commands from the subject by detecting those ERPs in the background EEG. From a pattern recognition point of view, the ERP-based BCI classification problem, in which two classes are involved (EEG with ERP or target class and EEG without ERP or non-target class), is highly complex. This is so mainly for two reasons: the presence of the high inter-trial variability and the unfavourable signal-to-noise ratio.

It is well-know that in any BCI classification scheme two main difficulties must be dealt with: the curse-of-dimensionality and the bias-variance trade-off [5]. While the former is a consequence of working with a concatenation of multiple time points from multiple channels, the latter refers to the generalization capability of the classifier. Several works have proposed different feature extraction methods for reducing the dimension of the feature space and capturing the most discriminative information in a single-trial ERP [6–8]. For instance, the common spatial patterns (CSP) method introduced in [9] is a supervised feature extraction technique which is widely used in motor imagery BCI ([10–12]). A Fisher's criterion (FC)-based on spatial filtering for ERP classification, which has shown stronger denoising capability than CSP for ERP-based BCI, was presented in [13].

The feature extraction step is usually follow by the design of an appropriate classification technique. In this regard, although many classification strategies have been proposed, it is widely accepted that linear discriminant analysis (LDA) is a very good classification scheme, resulting most of the times in optimal performances while keeping the solution simple [14]. As a drawback, effective training of a LDA classifier usually requires a number of samples between five and ten times the dimensionality of the patterns [15], resulting in very long system calibration times. Several regularized LDA schemes within the BCI context have been proposed [4, 14, 16, 17]. It has been shown that a regularized version of LDA can significantly increase the classification performance obtained by standard LDA. This improvement is due to the fact that regularization helps avoiding: i) the influence of outliers and strong noise, ii) the complexity of the classifier and iii) the raggedness of the decision surface [16].

One of the main disadvantages of current BCI systems is the fact that they require long calibration times to achieve a reliable and stable communication. Hence, the design of a scheme capable of providing good classification performance in small sample scenarios is highly desirable in order to enhance the practicability of an ERP-based BCI system. As an effort in this direction, for the case of high dimensional data with small training samples, the shrinkage LDA (SKLDA) method presented in [14] seeks to improve the usual estimation of the ill-conditioned covariance matrix used in LDA by a shrinkage covariance estimator.

Also, it has been claimed in [18] that data preprocessing, feature extraction and classification should not be regarded as isolated processes, since attacking each of these tasks separately and ignoring the inter-relationship between them might result in sub-optimum performances. Other works

([19, 20]) also suggest that an unified discriminative approach might provide a better overall performance. In line with the above philosophy, in this article we propose a method in which feature selection and classification are made in an interleaved and integrated process. A well-known and widely used method in which classification and feature selection are jointly made is the so-called stepwise LDA (SWLDA), originally introduced in ERP classification problems by Farwell and Donchin in [4]. The SWLDA method is a combination of forward and backward stepwise regression with statistical testing in which features are automatically selected by adding the most significant variables and removing the least significant ones. This process is iterated until a predetermined number of coefficients are included, or until no additional coefficients satisfy the given entry nor the removal criteria.

More recent classification schemes ([17, 21–24]) make use of ℓ_1-regularized least squares regression techniques which induce sparse solutions and therefore result in very robust classifiers.

Following the above research direction, in this work we propose a model which combines and makes simultaneous use of regularization, sparse feature selection and a-priori discriminative information. More precisely, we develop a new penalized version of the sparse discriminant analysis (SDA) [25], which we call generalized sparse discriminant analysis (GSDA), with the main objective of solving the binary ERP classification problem. As far as we know SDA has never been used before in ERP-based BCI classification problems. The performance of the GSDA method will first be compared with that of SDA and then, in small training sample scenarios, with those of LDA, SWLDA, SKLDA and FC+LDA. These comparison results will clearly show that our GSDA method has a high potential for reducing calibration times in BCI systems.

The organization of this article is as follows. In Section 2 we make a brief review on discriminant analysis from the statistical literature. Our proposed new approach is presented in Section 3. In Section 4 the two ERP-EEG databases used in the experiments are described. Section 5 contains details on all the experiments and results. Discussions are given in Section 6. Finally, concluding remarks and future works are presented in Section 7.

2. Discriminant Analysis: a brief review

The LDA criterion is a well-known dimensionality reduction tool in the context of supervised classification. Its popularity is mainly due to its simplicity and robustness which lead to very high classification performances in many applications [26].

Let $\mathbf{W}_1, \ldots, \mathbf{W}_{\mathrm{K}}$ be p-dimensional random vectors whose distributions uniquely characterize each one of the K classes of a given classification problem. In addition, let $\mathbf{X}$ be an $n \times p$ data matrix such that each one of its rows, $\mathbf{x}_i$, is a realization of one and only one of the aforementioned random vectors, and let $\mathbf{z} \in \{1, 2, \ldots, K\}^n$ be a categorical variable accounting for class membership, i.e. such that if pattern $\mathbf{x}_i$ is a realization of $\mathbf{W}_{\mathrm{k}}$, then $z_i = k$.

The LDA method consists of finding $q < K$ discriminant vectors (directions), $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_q$ such that by projecting the data matrix $\mathbf{X}$ over those directions, the "classes" will be well separated one from each other. It is assumed that the random vectors $\mathbf{W}_1, \ldots, \mathbf{W}_{\mathrm{K}}$ are independently and normally distributed with a common covariance matrix $\mathbf{\Sigma}_t$. The procedure for finding the vectors $\boldsymbol{\beta}_j$ requires of estimates of the within-class, the between-class and the total covariance matrices, $\mathbf{\Sigma}_w$, $\mathbf{\Sigma}_b$ and $\mathbf{\Sigma}_t$, respectively. These estimates are given by:

$$\hat{\mathbf{\Sigma}}_w = \frac{1}{n}\sum_{k=1}^{K}\sum_{i\in I_k}(\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T,$$

$$\hat{\mathbf{\Sigma}}_b = \frac{1}{n}\sum_{k=1}^{K} n_k(\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T,$$

$$\hat{\mathbf{\Sigma}}_t = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T,$$

where I_k and n_k are the set of indices and the number of patterns belonging to class k, respectively, $\boldsymbol{\mu}_k \doteq \frac{1}{n_k}\sum_{i\in I_k}\mathbf{x}_i$ is the k-class sample mean and $\boldsymbol{\mu} \doteq \frac{1}{n}\sum_{k=1}^{K}\boldsymbol{\mu}_k$ is the common sample mean. Note that $\hat{\mathbf{\Sigma}}_t = \hat{\mathbf{\Sigma}}_w + \hat{\mathbf{\Sigma}}_b$.

The LDA method seeks to find the vectors $\boldsymbol{\beta}_j$ in such a way that they maximize separability between classes, which is achieved by simultaneously maximizing $\hat{\mathbf{\Sigma}}_b$ and minimizing $\hat{\mathbf{\Sigma}}_w$, or equivalently, by simultaneously maximizing $\hat{\mathbf{\Sigma}}_b$ and minimizing $\hat{\mathbf{\Sigma}}_t$. Since the rank of $\hat{\mathbf{\Sigma}}_b$ is at most $K-1$, there are at most $K-1$ non-trivial solutions $\boldsymbol{\beta}_j^*$. Usually $q = K-1$.

In the particular case $K = 2$ (and therefore $q = 1$), the solution to the LDA problem has the following explicit formulation:

$$\boldsymbol{\beta}^* = \hat{\mathbf{\Sigma}}_t^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2). \quad (1)$$

This special case is known as Fisher linear discriminant analysis (FLDA) [27]. The FLDA approach can be formulated as a linear regression model [26, 27]. Let $\mathbf{X}$ be as before and let $\mathbf{y}$ be a n-dimensional vector such that $y_i = \frac{n_2}{n}$ or

$y_i = -\frac{n_1}{n}$, depending on whether the i^{th} observation belongs to class 1 or to class 2, respectively, and let us consider the following ordinary least squares problem (OLS):

$$\boldsymbol{\alpha}^* = \underset{\boldsymbol{\alpha}\in\mathbb{R}^p}{\arg\min} \|\mathbf{y} - \boldsymbol{X}\boldsymbol{\alpha}\|_2^2, \tag{2}$$

whose solutions are all the vectors in the set $\mathcal{N}(\mathbf{X}^T\mathbf{X}) + (\mathbf{X}^T\mathbf{X})^\dagger\mathbf{X}^T\mathbf{y}$, where "†" denotes the Moore-Penrose generalized inverse and $\mathcal{N}(\mathbf{X}^T\mathbf{X})$ denotes the null space of $\mathbf{X}^T\mathbf{X}$. If $\mathbf{X}^T\mathbf{X}$ is invertible, then (2) has a unique solution given by $\boldsymbol{\alpha}^* = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$. For convenience it is assumed that $\boldsymbol{\mu} = 0$, and therefore $\mathbf{X}^T\mathbf{X} = n\boldsymbol{\Sigma}_t$ and $\mathbf{X}^T\mathbf{y} = \frac{n_1 n_2}{n}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$. Hence $\boldsymbol{\alpha}^* = \frac{n_1 n_2}{n^2}\boldsymbol{\beta}^*$, where $\boldsymbol{\beta}^*$ is given by (1). Since the direction of the solution is independent of the proportionality constant $\frac{n_1 n_2}{n^2}$, this proves that OLS (2) is equivalent to the FLDA method (1).

Several works ([28–30], to cite a few) have extended the above OLS-LDA formulation to multi-class problems. It has been shown that the LDA solution can be obtained from a multivariate regression fit. In particular, Hastie et al. in [28], introduced a richer and more flexible classification scheme into LDA, called *optimal scoring*, which we briefly describe below.

Let $\mathbf{X}$ be as before and $\mathbf{Y}$ be a $n \times K$ matrix of binary variables such that y_{ij} is an indicator variable of whether the i^{th} observation belongs to the j^{th} class. Let us define $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_q] \in \mathbb{R}^{K\times q}$, where the vectors $\boldsymbol{\theta}_j$ are recursively obtained, for $j = 1, 2, \ldots, q$, as the solution of the following constrained least squares problem which resumes the optimal scoring method:

$$\begin{aligned}(\boldsymbol{\beta}_j, \boldsymbol{\theta}_j) &= \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\theta}\in\mathbb{R}^K}{\arg\min} \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_2^2, \\ &\quad s.t.\ \frac{1}{n}\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} = 1,\ \boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta}_l = 0 \quad \forall l = 1, 2, \ldots, j-1.\end{aligned} \tag{3}$$

Note that when $j = 1$ the orthogonality condition in (3), which is imposed to avoid trivial solutions, is vacuous and hence it is not enforced. Details about the computational implementation to solve (3) can be found in [28].

In the sequel we shall refer to $\boldsymbol{\theta}_j$ as the "score vector". Observe that $\boldsymbol{\theta}_j$ is the vector in $\mathbb{R}^K$ for which the mapping from $\mathbb{R}^{n\times K}$ to $\mathbb{R}^n$ defined by $\mathbf{Y} \to \mathbf{Y}\boldsymbol{\theta}_j$, results optimal for the constrained least squares problem defined by (3). This mapping is precisely what introduces more flexibility into the LDA framework since it transforms binary variables into real ones.

Clemmensen et al. [25] introduced a regularized version of the optimal scoring problem by adding two penalization terms to the functional in (3). These penalization terms on one hand induce sparsity and on the other hand they allow correlated variables to be included in the solution. This regularized LDA formulation, named SDA, consists on recursively solving for $j = 1, 2, \ldots, q$, the following problem:

$$\begin{aligned}\left(\boldsymbol{\beta}_j, \boldsymbol{\theta}_j\right) = & \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\theta}\in\mathbb{R}^K}{\arg\min} \{\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_1\|\boldsymbol{\beta}\|_1 + \lambda_2\|\boldsymbol{\beta}\|_2^2\}, \\ & s.t. \quad \frac{1}{n}\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} = 1,\ \boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta}_l = 0 \quad \forall l = 1, 2, \ldots, j-1, \end{aligned} \tag{4}$$

where λ_1 and λ_2 are predefined positive constants, called regularization parameters, which balance the amount of sparsity and the number of correlated variables, respectively.

Problem (4) is alternately and iteratively solved as follows. At first $\boldsymbol{\theta}_j$ is hold fixed and optimization is performed with respect to $\boldsymbol{\beta}_j$. Then $\boldsymbol{\beta}_j$ is hold fixed and optimization is performed with respect to $\boldsymbol{\theta}_j$. The following two steps are iterated:

1. For given (fixed) $\boldsymbol{\theta}_j$, solve:

$$\boldsymbol{\beta}_j = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min}\{\|\mathbf{Y}\boldsymbol{\theta}_j - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_1\|\boldsymbol{\beta}\|_1 + \lambda_2\|\boldsymbol{\beta}\|_2^2\}. \tag{5}$$

2. For given (fixed) $\boldsymbol{\beta}_j$, solve:

$$\begin{aligned}\boldsymbol{\theta}_j = {} & \underset{\boldsymbol{\theta}\in\mathbb{R}^K}{\arg\min}\,\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}_j\|_2^2 \\ & s.t.\ \frac{1}{n}\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} = 1,\ \boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta}_l = 0 \quad \forall l = 1, 2, \ldots, j-1.\end{aligned}$$

For computational implementation details of the above steps we refer the reader to [25] and [31].

The solution of problem (4), just like in LDA, provides q discriminant directions, $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_q$, over which the classes of the projected data matrix $\left(\boldsymbol{X}\boldsymbol{\beta}_1\ \boldsymbol{X}\boldsymbol{\beta}_2\ \ldots\ \boldsymbol{X}\boldsymbol{\beta}_q\right) \in \mathbb{R}^{n\times q}$ cab be well-separated by a simply linear classifier.

Solving (5) involves the well-known elastic-net problem (e-net) [32]. Besides performing sparse variable selection like LASSO (least absolute shrinkage and selection operator [33]), e-net tends to overcome one of LASSO's main limitations, which, as stated by several authors (e.g. [32, 34, 35]) is the fact that from a group of correlated variables, it always chooses only one of them.

In this work we propose a generalized version of the SDA method to efficiently solve the binary classification problem appearing in BCI systems based on ERPs. This new method seeks to increase classification performance by taking into account information about the difference between classes by means of the inclusion of appropriate anisotropy matrices into the penalizing terms. The use of adaptive penalizers and, in particular of anisotropy matrices in regularization method for inverse ill-posed problems is a new approach that has shown to produce significantly better results than those obtained with the corresponding non-adaptive or isotropic penalizers [36, 37].

3. A new approach: Generalized Sparse Discriminant Analysis

An ERP-based BCI system implies solving a binary classification problem ($K = 2$). In this case, our GSDA scheme consists of solving the following regularized constrained least squares problem:

$$\begin{aligned} \left(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}}\right) &= \underset{\boldsymbol{\beta}\in\mathbb{R}^p, \boldsymbol{\theta}\in\mathbb{R}^K}{\arg\min} \{\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_1\|\mathbf{D}_1\boldsymbol{\beta}\|_1 + \lambda_2\|\mathbf{D}_2\boldsymbol{\beta}\|_2^2\}, \\ s.t. &\quad \frac{1}{n}\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} = 1, \end{aligned} \tag{6}$$

where $\mathbf{X}$ is a data matrix, $\mathbf{Y}$ is a $n \times 2$ matrix of binary variables accounting for class-membership as before, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ are generic p and 2-dimensional vectors, λ_1 and λ_2 are positive regularization parameters, and $\mathbf{D}_1$ and $\mathbf{D}_2$ are appropriately defined $p \times p$ positive definite matrices.

Note that since $K = 2$, the orthogonality condition in (4) is vacuous. As in the SDA case, the solution to problem (6) is approximated by alternatively iterating the following two steps (with an adequate initialization):

1. Given $\boldsymbol{\theta}$, solution of (8), solve:

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min}\{\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_1\|\mathbf{D}_1\boldsymbol{\beta}\|_1 + \lambda_2\|\mathbf{D}_2\boldsymbol{\beta}\|_2^2\}. \tag{7}$$

2. Given, $\boldsymbol{\beta}$ solution of (7), solve:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\mathbb{R}^2}{\arg\min}\, \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \quad s.t. \quad \frac{1}{n}\boldsymbol{\theta}^T\mathbf{Y}^T\mathbf{Y}\boldsymbol{\theta} = 1. \tag{8}$$

The vector $\hat{\boldsymbol{\beta}}$, solution of (6), not only inherits both the correlated variables selection and sparsity properties of SDA, but it also contains in each one of its components appropriate discriminative information which is suitable for improving separability between classes. As before, the classification rule is constructed based upon the $n \times 1$ projected matrix $\boldsymbol{X}\hat{\boldsymbol{\beta}}$. In the following subsection we show how the Kullback-Leibler divergence can be used for constructing the anisotropy matrices $\mathbf{D}_1$ and $\mathbf{D}_2$ in such a way that they appropriately incorporate discriminative information into GSDA.

3.1. Kullback-Leibler discriminant information

Discriminative information can be incorporated into GSDA by appropriately quantifying the "distances" between classes, or more precisely, between their probability distributions. Although there is a wide variety of "metrics" for comparing probability distributions [38], we shall use here the well-known Kullback-Leibler divergence [39]. The decision to use this particular "metric" is due not only to its nice mathematical properties, but also to the fact that it was already successfully applied in many classification problems [40–42].

Let $\mathbf{N}$ be a discrete random variable defined on a discrete outcome space $\mathcal{N}$ and consider two probability functions $f_1(n)$ and $f_2(n)$, $n \in \mathcal{N}$. Then, the Kullback-Leibler "distance" (KLD) of f_1 relative to f_2 is defined as:

$$D_{\mathrm{KL}}(f_1||f_2) \doteq \sum_{n \in \mathcal{N}} f_1(n) \log\left(\frac{f_1(n)}{f_2(n)}\right),$$

with the convention that $0.\log 0 \doteq 0$. Although $D_{\mathrm{KL}}(f_1||f_2)$ quantifies the discrepancy between f_1 and f_2, it is not a metric in the rigorous mathematical sense, because it is not symmetric and it does not satisfy the triangle inequality. If, for any reason, symmetry is desired then a modified KLD, called

J-divergence, can be defined as follows:

$$J_{\mathrm{KL}}(f_1, f_2) \doteq \frac{D_{KL}(f_1||f_2) + D_{KL}(f_2||f_1)}{2}.$$

Let $f_j^i(\cdot)$ be the probability function of the j^{th} class in the i^{th} feature, with $j = 1, 2$ and $i = 1, 2, \ldots, p$. We define the J-divergence at feature i as

$$J_{KL}(i) \doteq J_{KL}\left(\left\{f_j^i\right\}_{j=1}^{K=2}\right). \tag{9}$$

This function quantifies the discrepancy between the two classes at feature i. A value of $J_{KL}(i)$ close to zero means that there is very little discriminative information at feature i, while a large value of $J_{KL}(i)$ means that feature i contains a significant amount of discriminative information which we definitely want to take into account in the construction of the solution vector $\hat{\boldsymbol{\beta}}$. In Section 6.1 we show in detail how the J-divergence is able to highlight the most discriminative features.

As mentioned before, the available a-priori discriminative information can be incorporated into the GSDA formulation (6) by means of appropriately constructed anisotropy matrices $\mathbf{D}_1$ and $\mathbf{D}_2$. Since we wish to spotlight those features containing significant amount of discriminative information, the matrices $\mathbf{D}_1$ and $\mathbf{D}_2$ must be constructed so as to strongly penalize those features where there is little or none discriminative information while avoiding penalization at the remaining ones (see Section 5).

3.2. Computational implementation

Our computational implementation of GSDA bellow is made by appropriately modifying the original SDA algorithm [31]. Thus GSDA is mainly solved in two steps. In the first step we solve equation (7), which is a generalized version of the e-net problem [43]. The second step consists of updating the optimal score vector $\boldsymbol{\theta}$ by solving (8). It is shown in [25] that the solution of (8) is given by $\hat{\boldsymbol{\theta}} = s(\mathbf{I} - \boldsymbol{\theta}\boldsymbol{\theta}^T\boldsymbol{\pi})\boldsymbol{\pi}^{-1}\mathbf{Y}^T\mathbf{X}\hat{\boldsymbol{\beta}}$, where $\boldsymbol{\pi} \doteq \frac{1}{n}\mathbf{Y}^T\mathbf{Y}$ and s is a proportionality constant such that $\hat{\boldsymbol{\theta}}\boldsymbol{\pi}\hat{\boldsymbol{\theta}} = 1$.

In regard to the first step, it is known that the e-net problem can be reformulated by means of LASSO. In fact by defining the following augmented variables:

$$\tilde{\mathbf{X}} \doteq \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\,\mathbf{D}_2 \end{pmatrix}_{(n+p)\times p} , \quad \tilde{\mathbf{Y}} \doteq \begin{pmatrix} \mathbf{Y}\boldsymbol{\theta} \\ \mathbf{0}_{p\times 1} \end{pmatrix}_{(n+p)\times 1} ,$$

the generalized e-net problem (7) can be re-written as:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}\in\mathbb{R}^p}\{\|\tilde{\mathbf{Y}} - \tilde{\mathbf{X}}\boldsymbol{\beta}\|_2^2 + \lambda_1\|\mathbf{D}_1\boldsymbol{\beta}\|_1\}, \tag{10}$$

which is known as generalized LASSO [44]. If $\mathbf{D}_1$ is invertible the solution of (10) can be explicitly found as $\hat{\boldsymbol{\beta}} = \mathbf{D}_1^{-1}\hat{\boldsymbol{\alpha}}$, where $\hat{\boldsymbol{\alpha}}$ is the solution of:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^p}\{\|\tilde{\mathbf{Y}} - \tilde{\mathbf{X}}\mathbf{D}_1^{-1}\boldsymbol{\alpha}\|_2^2 + \lambda_1\|\boldsymbol{\alpha}\|_1\}. \tag{11}$$

Thus, given this relationship (11), the first GSDA step (7) can be implemented by using the modified LARS-EN algorithm presented in [31], in which e-net is performed with early stopping. This criterion consists of introducing a parameter called *stop*, in such a way that if stop is negative, its absolute value corresponds to the desired number of variables, and while if stop is non-negative, it corresponds to an upper bound for $\|\boldsymbol{\beta}\|_1$.

3.3. Regularization parameters

It is well-known that in every regularization method the choice of the regularization parameters is crucial. For Tikhonov-type functionals a popular and widely used method for approximating the optimal parameters is the so called L-curve criterion [45]. One of the main advantages of this selection criterion is the fact that it does not require of any prior knowledge about the noise. Roughly speaking, the method finds an optimal compromise between the norm of the residual and the norm of the regularized solution by selecting the point of maximal curvature in the curve described by those two quantities, parametrized by the corresponding regularization parameter (for details see [45]).

Despite its popularity, the L-curve method cannot be directly applied to multi-parameter penalization functionals like (7). In 1998, Belge et al. proposed and extension of the L-curve technique, called L-hypersurface, for approximating the optimal regularization parameters in those cases [46]. The authors show that a good approximation to the optimal regularization parameter is given by the minimizer of the residual norm.

Within the GSDA context formalized by (6) or, more precisely, in the context of the generalized e-net problem (7), the L-hypersurface is defined as $S(\boldsymbol{\lambda}) \doteq \{(x_1(\boldsymbol{\lambda}), x_2(\boldsymbol{\lambda}),\ z(\boldsymbol{\lambda})) : \boldsymbol{\lambda} \in \mathbb{R}^2_+\}$, where $x_1(\boldsymbol{\lambda}) \doteq \log \|\mathbf{D}_1\boldsymbol{\beta}(\boldsymbol{\lambda})\|_1$, $x_2(\boldsymbol{\lambda}) \doteq \log \|\mathbf{D}_2\boldsymbol{\beta}(\boldsymbol{\lambda})\|_2^2$ and $z(\boldsymbol{\lambda}) \doteq \log \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}(\boldsymbol{\lambda})\|_2^2$. Then, the optimal regularized parameter vector ends up being defined by $\hat{\boldsymbol{\lambda}} \doteq \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}^M_+} z(\boldsymbol{\lambda})$.

Although generalized e-net is defined in terms of λ_1 and λ_2, there are other possible choices for tuning parameters [32]. For example, the ℓ_1-norm of the coefficients (t) can be chosen instead of λ_1. In fact, this can be achieved by re-writing the LASSO version (11) of our generalized e-net as a constrained optimization problem with an upper bound on $\|\boldsymbol{\alpha}\|_1$. Similarly, since the LARS-EN algorithm is a forward stage-wise additive fitting procedure, the number of steps (κ) of the algorithm can also be used as a tuning parameter replacing λ_1. This is so because, for each fixed λ_2, LARS-EN produces a finite number of vectors $\hat{\boldsymbol{\beta}} \in \mathbb{R}^p$ which are approximations of the true solution at each step. In our GSDA implementation we adopted λ_2 and κ as tuning parameters, i.e. $\boldsymbol{\lambda} \doteq (\lambda_2,\ \kappa)$. By defining $z(\boldsymbol{\lambda}) \doteq \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}(\boldsymbol{\lambda})\|_2^2$, and, in accordance to Belge's remark described above, the best parameter vector $\boldsymbol{\lambda}^*$ was selected as that minimizing the residual norm. The steps for solving GSDA with this proposal (together with automatic parameter selection) are presented in Algorithm 1.

4. P300 speller databases

Two real ERP-EEG databases were used to evaluate the classification performance of our GSDA method. In both databases the P300 speller paradigm was used [4].

4.1. Dataset-1

Dataset-1 is an open-access P300 speller database from the "Laboratorio de Investigación en Neuroimagenología de la Universidad Autónoma Metropolitana", Mexico D.F., described in [47]. This database consists of EEG records acquired from 25 healthy subjects, recorded by 10 channels (Fz, C3, Cz, C4, P3, Pz, P4, PO7, PO8, Oz) at 256 Hz sampling rate using a gUSBamp (g.tec, Austria). A 6-by-6 matrix containing letters and numbers was presented to each subject on a computer screen. Each row/column was highlighted for a period of 62.5 ms with inter-stimuli intervals of 125 ms. For each character to be spelled the stimulating block (12 consecutive flashings) was repeated 15 times.

Algorithm 1 GSDA with automatic parameter selection

```
Inputs: $\mathbf{X}$, $\mathbf{Y}$ , $\mathbf{D}_1$, $\mathbf{D}_2$, $\Lambda_2 = \left\{\lambda_2^{(1)}, \ldots, \lambda_2^{(d)}\right\}$.
Define $\boldsymbol{\pi} \doteq \frac{1}{n}\mathbf{Y}^T\mathbf{Y},\ K = 2$
Initialize: $\boldsymbol{\theta} = eye(K, 1)$.
while the sparse discriminative direction $\boldsymbol{\beta}$ has not converged do
    for $i = 1, \ldots, d$ do
        Re-define the variables:
            $\tilde{\mathbf{X}} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2^{(i)}}\,\mathbf{D}_2 \end{pmatrix}_{(n+p)\times p}, \quad \tilde{\mathbf{Y}} = \begin{pmatrix} \mathbf{Y}\boldsymbol{\theta} \\ \mathbf{0}_{p\times 1} \end{pmatrix}_{(n+p)\times 1}$
        Solve the generalized e-net problem and save the solution path:
            $(\mathbf{A}, \kappa) = \text{LARSEN}(\tilde{\mathbf{X}}\mathbf{D}_1^{-1}, \tilde{\mathbf{Y}}), \quad \mathbf{B} = \mathbf{D}_1^{-1}\mathbf{A}$
        Find the residual:
            $\mathbf{R}(\lambda_2, 1:\kappa) = \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\mathbf{B}_j\|_2^2$
        Save the solutions:
            $\mathbf{B}_{\text{all}}(\lambda_2, 1:\kappa, :) = \mathbf{B}$
    end for
    Select the optimal direction:
        $(\hat{\lambda_2}, \hat{\kappa}) = \arg\min_{\lambda_2, \kappa} \mathbf{R}(\lambda_2, \kappa)$
        $\boldsymbol{\beta} = \mathbf{B}_{\text{all}}(\hat{\lambda_2}, \hat{\kappa}, :)$
    Update $\boldsymbol{\theta}$:
        $\tilde{\boldsymbol{\theta}} = (\mathbf{I} - \boldsymbol{\theta}\boldsymbol{\theta}^T\boldsymbol{\pi})\boldsymbol{\pi}^{-1}\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta}, \quad \boldsymbol{\theta} = \frac{\tilde{\boldsymbol{\theta}}}{\sqrt{\tilde{\boldsymbol{\theta}}^T\boldsymbol{\pi}\tilde{\boldsymbol{\theta}}}}$
end while
Outputs: : $\boldsymbol{\theta}$, $\boldsymbol{\beta}$
```

Each subject participated in 4 sessions, the first two of which were copy-spelling runs, i.e. they contained the true label data vector. For this reason, in this work we used those two copy-spelling sessions as our dataset. Each subject had to spell 21 characters and the stimulating block was repeated 15 times. In the preprocessing stage, the EEG records were filtered from 0.1 Hz to 12 Hz by a 4^{th} order forward-backward Butterworth band-pass filter. A 1000 ms data segment (trial) was extracted (windowed) from the EEG records at the beginning of each stimulus. A total of 3780 EEG trials (630 of them being target) of dimension of $10 \times 256 = 2560$, conforms each subject's database.

4.2. Dataset-2

Dataset-2 corresponds to dataset II of the BCI competition III[1]. The dataset consists of EEG records from two subjects (A and B) recorded at 240 Hz sampling rate with 64 channels, divided into train and test datasets, containing 85 and 100 characters, respectively, with 15 repetitions of the stimulating block for each character. In this case each row/column of the P300 speller was intensified for 100 ms with inter-stimuli intervals of 75 ms. For more information we refer the reader to [48].

In the present work only the train data was used to test our method, since it contains the true labels and a large number (15300 of which 2550 are target) of EEG trials. The same pre-processing stage for Dataset-1 was implemented. We used the EEG patterns from 16 channels (F3, Fz, F4, T7, C3, Cz, C4, T8, P7, P3, Pz, P4, P8, PO7, PO8 and Oz) as selected by the authors in

[1] http://www.bbci.de/competition/iii/

[17, 20]. Therefore, the dimension of each pattern is $16 \times 240 = 3840$.

5. Experiments and Results

5.1. GSDA vs. SDA

In this section we compare the performance of our GSDA method with the one obtained with the standard SDA in the context of the aforementioned real ERP-based BCI classification problem using both datasets described in Section 4.

The symmetric J_{KL} version of KLD 9 was used as measure of discrepancy. To compute J_{KL}, the probability distribution of each class was estimated by using appropriate discrete sample distribution constructed from the available training data. The KLD information was then used to construct the followings two $p \times p$ diagonal anisotropy matrices:

$$\mathbf{D_1} \doteq diag\left(1 - \alpha_i + \alpha_i c_i\right),$$
$$\mathbf{D_2} \doteq diag(c_i),$$

where

$$c_i \doteq \frac{\left(\prod_{j=1}^{p} J_{KL}(j)\right)^{1/p}}{J_{KL}(i)}, \quad \alpha_i \doteq \frac{\max\{c_j\}_{j=1}^{p} - c_i}{\max\{c_j\}_{j=1}^{p} - \min\{c_j\}_{j=1}^{p}}, \; i = 1, \ldots, p.$$

Note that with $\mathbf{D_1}$ and $\mathbf{D_2}$ so defined, c_i is large where $J_{KL}(i)$ is small, and vice-versa. The parameter α_i (observe that $0 \leq \alpha_i \leq 1$, $\forall i = 1, \ldots, p$) weights the KLD information proportionally to its relevance. Thus $\alpha_i = 1$ if $J_{\mathrm{KL}}(i) = \max\{J_{\mathrm{KL}}(j)\}_{j=1}^{p}$ and $\alpha_i = 0$ if $J_{\mathrm{KL}}(i) = \min\{J_{\mathrm{KL}}(j)\}_{j=1}^{p}$. Hence, with this choice of $\mathbf{D}_1$ the KLD based information is used with preference (as measured by α_i) in the ℓ_1-norm over those features having large discriminant information, while with the introduction of $\mathbf{D}_2$ in the ℓ_2-norm we avoid penalization where KLD information is large. Clearly, if there exist $i_0, 1 \leq i_0 \leq p$, such that $J_{KL}(i_0) = 0$ then the matrices $\mathbf{D}_1$ and $\mathbf{D}_2$ cannot be formally defined as above. This case, however, can be overcome by simply replacing $J_{KL}(i_0)$ by $J_{KL}(i_0) + \epsilon$, with $\epsilon > 0$ very small.

Both SDA (corresponding to $\mathbf{D}_1 = \mathbf{D}_2 = \mathbf{I}$) and our GSDA methods were implemented with automatic parameter selection as described in Algorithm 1, in which the parameter λ_2 was allowed to vary between 10^{-6} and 10^{-1} in a log-scale. In order to compare SDA and GSDA under the same stopping condition we set the upper bound of the ℓ_1-norm of the coefficients equal to 10% ans 20% of the dimension of the patterns (p) for Dataset-1 and Dataset-2, respectively. The decision of using all sample features with no downsampling was made in order to work with high dimensional data scenarios in which LASSO and e-net have already been largely applied. All codes were run in MATLAB® on an Intel® Core$^{\text{TM}}$ i7-6700K CPU @ 4.00GHz × 8 with 32GB of memory.

Several measures exist for evaluating the classification performance of a BCI classification method [49]. The receiver operator characteristics (ROC) curve is a powerful tool for evaluating a two-class unbalanced problem [50]. In the present work the Area Under the ROC Curve [51], denoted by AUC, was used as the classification performance measure. For avoiding classification bias, a 3-fold cross-validation procedure was implemented. In each fold, KLD is estimated with the available training data. A one-way anova with a level of significance $\alpha = 0.05$ was performed to statistically analyze difference between the performance reached by both methods.

The GSDA and SDA classification results obtained with Dataset-1 and Dataset-2 are shown in Figure 1 and Table 1, respectively.

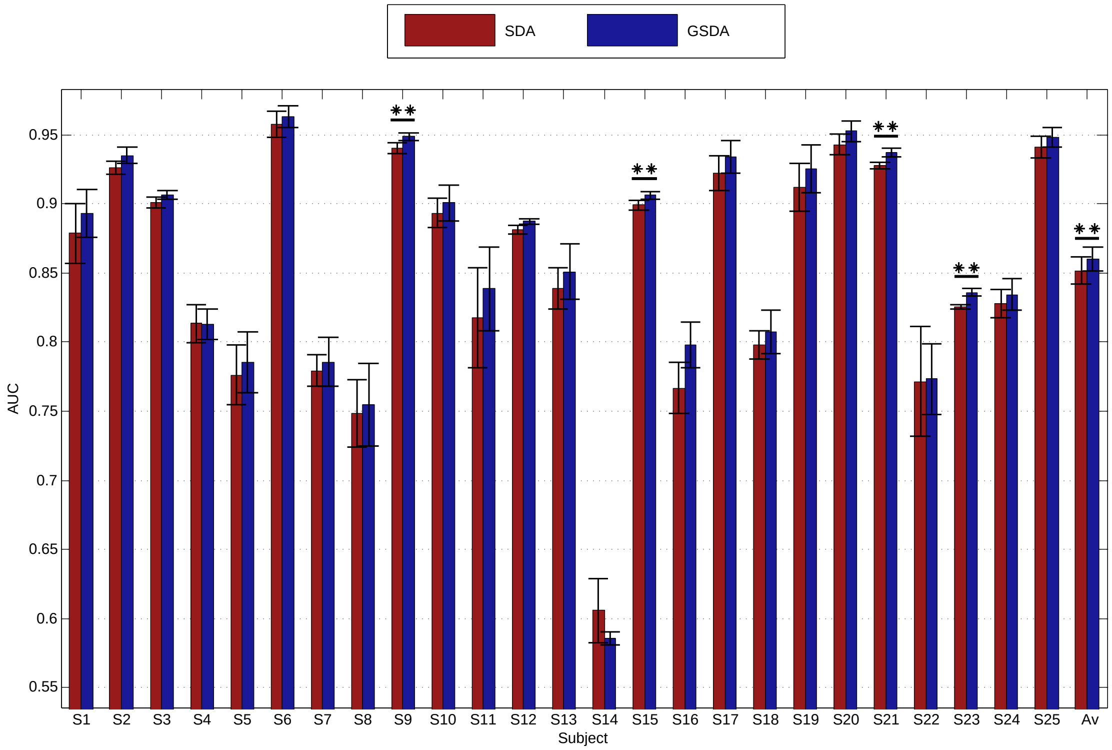


Figure 1: Area under the ROC curve (AUC) on test data from Dataset-1 evaluated by 3-fold cross-validation derived by SDA and GSDA. The errorbars for each subject correspond to the AUC standard deviation on the 3-fold. The errorbars of the average correspond to the standard deviation on all subjects. Here "$**$" indicates GSDA>SDA with p-value< 0.05.

As it can be seen in Figure 1 and Table 1, GSDA outperforms SDA for both subjects of Dataset-2 and for all but one of the 25 subjects of Dataset-1. These results reinforce our original belief that the appropriate inclusion of a-

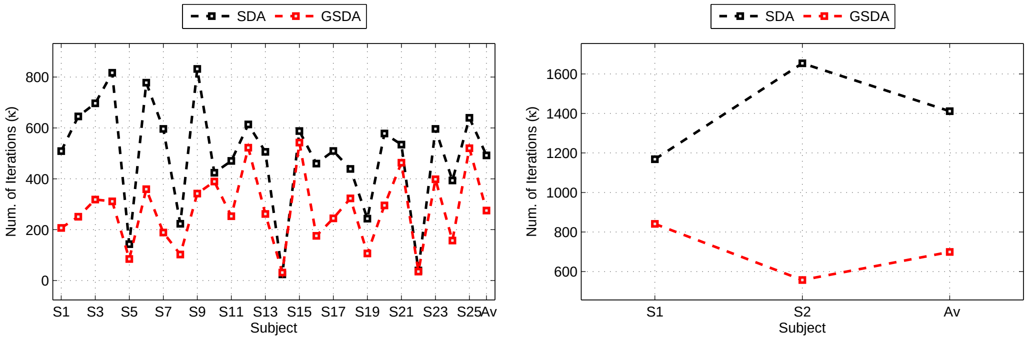
Table 1: Mean and standard deviation of the area under the ROC curve (AUC) on test data from Dataset-2 evaluated by 3-fold cross-validation derived by SDA and GSDA.

	SDA	**GSDA**
Subject A	0.7547 (± 0.0087)	**0.7551** (± 0.0098)
Subject B	0.8418 (± 0.0041)	**0.8490** (± 0.0024)
Average	0.7982 (± 0.0033)	**0.8020** (± 0.0053)

priori discriminant information into the model may significantly influence the classification results. In this regard, we have first performed a naive approach in which the KLD information is used as a dimentionality reduction tool by selecting those features with larger KLD values. We selected the N features ($N = 0.1 \times p$) associated to the N samples at which KLD is larger, and afterwards, simple LDA was performed. The average classification results over the 3-fold for Dataset-1 and Dataset-2 were found to be 0.5529 (± 0.0858) and 0.5207 (± 0.0345). These results clearly indicate both the importance of appropriate usage of the a-priori available information and the advantage of tackling feature selection and classification together. In the case of Subject 14 of Dataset-1, the inclusion of KLD information reduces the classification performance. Although it is not clear why this happens, for reasons beyond our understanding, for this particular subject there seems to be no clear relation between the information provided by KLD and the P300 wave (as it can be seen in Figure 4a).

Since solution sparsity was desired, the mean of the number of non-zero values for each method was analyzed. For the SDA and GSDA the mean of the percentage of non-zero values with respect to the number of sample points for Dataset-1 were found to be around 14% and 6%, respectively, while for Dataset-2 those values were found to be around 24% and 5%, respectively. It is timely to highlight that with less than 6% of the features high classification performances are achieved for both datasets. In light of the sparsity degree of the solutions and the classification performances obtained, the results indicate that the implemented automatic parameter selection procedure is adequate.

We have also analyzed the number of iterations required for both methods to fin the discriminant vector $\hat{\boldsymbol{\beta}}$. This is depicted in Figure 2 for both datasets.



(a) Dataset-1

(b) Dataset-2

Figure 2: Number of iterations needed for SDA and GSDA to find the solution vector $\hat{\boldsymbol{\beta}}$ for Dataset-1 and Dataset-2.

It is highly important to note that our GSDA method reaches convergence in a lower number of iterations than those needed by SDA. In turn, this is translated into a reduction in the computing time needed for GSDA. In our cases the elapsed time was cutback in average by more than 2.9% and 9.4%

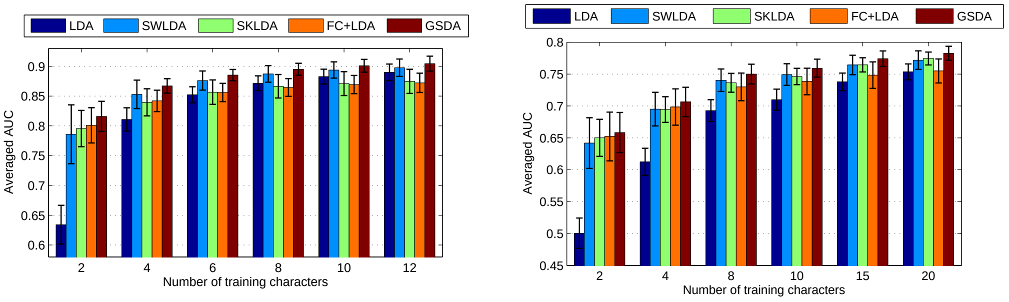
for Datasets 1 and 2, respectively.

5.2. Small training size scenarios

Now with the main objective of reducing as much as possible the calibration time required by any ERP-based BCI system, the GSDA classification performance is compared in different hard training scenarios by using a small number of samples to train the classifier. In this case, GSDA is compared with the LDA, SWLDA, SKLDA and FC+LDA methods in small training size scenarios by randomly selecting patterns for spelling different given number of characters (2, 4, 6, 8, 10 y 12 out of 21 for Dataset-1 and 2, 4, 8, 10, 15 and 20 out of 85 for Dataset-2). This selection procedure was repeated 100 times. For reducing the pattern dimension a proper downsampling step was made for both datasets. The EEG segments for both datasets were downsampled by selecting each 8^{th} point from the filtered data resulting in 32 and 30 sample points for Dataset-1 and Dataset-2, respectively. It is important to mention here that each spelling character generates 180 patterns ($1\times12\times15$).

The SWLDA method was implemented by setting the add-feature parameter to 0.1, the remove-feature parameter to 0.15 and the maximal number of feature to 60, as suggested in [15]. The SKLDA method was implemented by common diagonalization as presented in [14]. This two algorithms were implemented by using the BBCI toolbox [52]. In regard to the FC method, the first two spatial filter were used to reduce the dimension of the EEG data (320 to 64 features) and then ordinary LDA was applied, as in [20]. For our GSDA method, in order to speed up convergence, LARS-EN with early negative stopping criterion was implemented. After analyzing several different stopping criteria, for Dataset-1 we decided to start with a number of non-zero features equal to 40% (128 features) of the dimension of the patterns with increments of 5% for each scenario (reaching a 65% (208 features) in the last one) while for Dataset-2 we found those values from 25% (120 features) to 50% (240 features) in increments of 5%. This choice seems to be a reasonable compromise between optimality and generalizability of the solution. For statistical analysis a one-way anova with $\alpha = 0.05$ and multi-comparison test were implemented in each scenario.

Comparison results are shown in Figure 3. Several remarks are in order. Firstly, GSDA is always significantly better (p-value<0.05) than the other methods for Dataset-1, and in all but two cases (SKLDA and FC+LDA in 2 character training scenario) for Dataset-2. Secondly, the smallest error variance among all methods most of the times corresponds to GSDA, what indicates that this constitutes a highly stable classifier method. Thirdly and finally, for both datasets increasing the number of training characters beyond 10 seems to have very little effect on classification results.



(a) Dataset-1

(b) Dataset-2

Figure 3: Averaged AUC on test data delivered by LDA, SWLDA, SKLDA, FC+LDA and GSDA, respectively, using different number of training characters for Dataset-1 and Dataset-2. The errorbar denotes the standard deviation of AUC on the 100 repetitions.

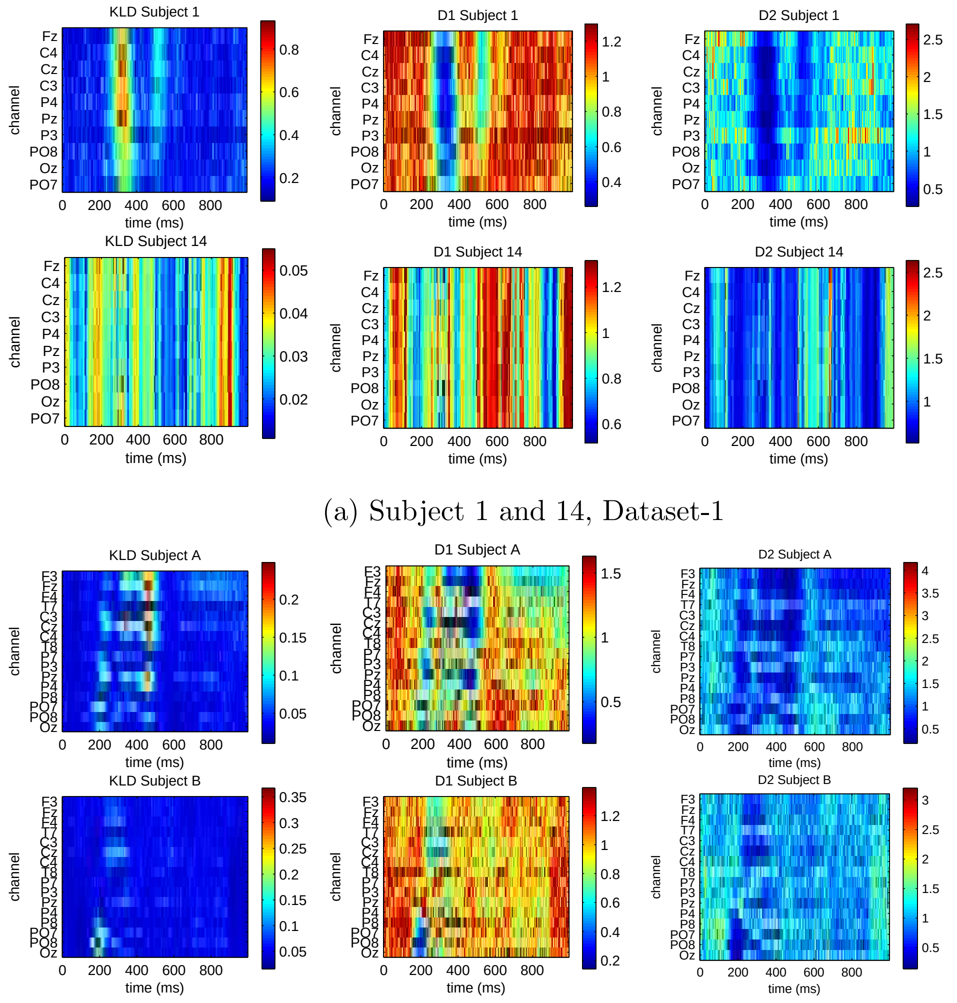
6. Discussion

6.1. KLD and anisotropy matrices

An analysis of the J-divergence as a function of channel and time allowed us to detect the most discriminative features. The KLD discriminant information was introduced into the GSDA formulation through the anisotropy matrices $\mathbf{D}_1$ and $\mathbf{D}_2$. Figure 4 shows three plots on the time-channel plane, for subjects 1 and 14 of Dataset-1 and subjects A and B of Dataset-2. These plots depict the KLD information and the matrices $\mathbf{D}_1$ and $\mathbf{D}_2$. Several observation are in order. First, the KLD plots indicate that most of the discriminant information is located in the 250-500 ms time window, in accordance with the well-known latency window of the P300 wave. Secondly, note that some channels seem to have no contribution at all to class separation. In the case of Subject 14 of Dataset-1 (the peculiar case previously mentioned in Section 5), the KLD plot does not seem to point out to any preferable discriminative region. The plots corresponding to $\mathbf{D}_1$ and $\mathbf{D}_2$ show how these anisotropy matrices avoid penalization at places where KLD values are large. Figure 4 also shows the high variability of the ERP morphology between subjects, as pointed out in Section 1. Finally, it is reasonable to think that KLD is an appropriate measure for enhancing the impact of the P300 wave in the GSDA solution, by selecting both the most discriminative channels and the most discriminative time samples.

6.2. Practicability of GSDA

Better classification performance in small training sample size scenarios suggest that our GSDA method could be effective to reduce calibration times



(a) Subject 1 and 14, Dataset-1

(b) Subject A and B, Dataset-2

Figure 4: J-divergence (KLD) and anisotropy matrices ($\mathbf{D}_1$ and $\mathbf{D}_2$) for different subject of Dataset-1 and Dataset-2

for ERP-based BCI systems. Indeed, from the results shown in Figure 3 for Dataset-1, we see that GSDA needs of only 4 training characters (720 training samples) to achieve averaged AUC around 86% in single-trial classification while reaching close to 90% in the case of 10 training characters (1800 training samples). A similar analysis for Dataset-2 reveals that those percentages are about 72 and 77, respectively. Thus, if an ERP-based BCI system is to be calibrated with 10 training characters, we estimates that only 5 minutes are required for data collection (assuming 100 ms flashing stimulus time and 75 ms inter-stimulus time with 12 flashing per block and 15 repetition of each block). In our experiments we found that the complete GSDA training procedure by using 1800 training samples requires only between 5 and 10 seconds. Hence, GSDA seem to be very suitable for daily calibration setting, by better adopting to the changes in the users' physiological pre-condition. Due to the high classification performance and the small variance obtain for the 100 iterations on the 10 character training scenario, we expected that our GSDA will yield a very good performance in an on-line context.

7. Conclusions

In the present work we briefly reviewed different LDA approaches for classification purposes from both statistical signal processing and BCI literature and developed a new penalized sparse discriminant analysis method, called Generalized Sparse Discriminant Analysis. This new method not only inherits the good properties of SDA, but it also allows for the inclusion of appropriate a-priori discriminative information. Our GSDA implementation also incorporates automatic tuning parameter selection.

We compared the new GSDA approach with the standard SDA. An analysis of the results showed that GSDA outperforms SDA not only in the sense of classification results, but also in the sense of the degree of sparsity and in required the number of iterations. Comparison results of GSDA with several other well-known state of the art methods in small training size scenarios seem to indicate that GSDA is a potential tool for reducing calibration times while keeping high classification performances.

Although these results are quite encouraging and they indicate that GSDA could be a valuable alternative for the ERP-EEG classification problem, and for many other applications, there is certainly much room for improvement. Further research is currently under-way in several directions. In particular, different discrepancy measures, anisotropy matrices and penalizing terms are being considered. Moreover we plan to extend the binary GSDA framework introduced in this article to a multi-class classification problem and to test it in well-known databases.

Acknowledgments

This work was supported in part by Consejo Nacional de Investigaciones Científicas y Técnicas, CONICET, through PIP 2014-2016 No. 11220130100216-CO, the Air Force Office of Scientific Research, AFOSR/SOARD, through Grant FA9550-14-1-0130 and and by Universidad Nacional del Litoral, UNL, through CAID-UNL 2011 Project No.525 within PACT "Señales, Sistemas e Inteligencia Computacional".

References

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, T. M. Vaughan, Brain-computer interfaces for communication and control, Clinical neurophysiology 113 (6) (2002) 767–791.

[2] J. Wolpaw, E. W. Wolpaw, Brain-Computer Interfaces: principles and practice, Oxford University Press, USA, 2012.

[3] S. A. Hillyard, M. Kutas, Electrophysiology of cognitive processing, Annual review of psychology 34 (1) (1983) 33–61.

[4] L. A. Farwell, E. Donchin, Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials, Electroencephalography and clinical Neurophysiology 70 (6) (1988) 510–523.

[5] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, B. Arnaldi, A review of classification algorithms for EEG-based brain-computer interfaces, Journal of neural engineering 4 (2) (2007) R1.

[6] V. Peterson, R. Acevedo, H. L. Rufiner, R. Spies, Local discriminant wavelet packet basis for signal classification in brain computer interface, in: VI Latin American Congress on Biomedical Engineering CLAIB 2014, Paraná, Argentina 29, 30 & 31 October 2014, Springer, 2015, pp. 584–587.

[7] N. Xu, X. Gao, B. Hong, X. Miao, S. Gao, F. Yang, Bci competition 2003-data set IIb: enhancing P300 wave detection using ICA-based subspace projections for bci applications, IEEE transactions on biomedical engineering 51 (6) (2004) 1067–1072.

[8] L. Ke, R. Li, Classification of EEG signals by multi-scale filtering and PCA, in: Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on, Vol. 1, IEEE, 2009, pp. 362–366.

[9] K. Fukunaga, W. L. Koontz, Application of the karhunen-loeve expansion to feature selection and ordering, IEEE Transactions on Computers 19 (4) (1970) 311–318.

[10] Y. Li, X. Gao, H. Liu, S. Gao, Classification of single-trial electroencephalogram during finger movement, IEEE Transactions on biomedical engineering 51 (6) (2004) 1019–1025.

[11] G. Blanchard, B. Blankertz, Bci competition 2003-data set iia: spatial patterns of self-controlled brain rhythm modulations, IEEE Transactions on Biomedical Engineering 51 (6) (2004) 1062–1066.

[12] J. Müller-Gerking, G. Pfurtscheller, H. Flyvbjerg, Designing optimal spatial filters for single-trial eeg classification in a movement task, Clinical neurophysiology 110 (5) (1999) 787–798.

[13] G. Pires, U. Nunes, M. Castelo-Branco, Statistical spatial filtering for a p300-based bci: tests in able-bodied, and patients with cerebral palsy and amyotrophic lateral sclerosis, Journal of neuroscience methods 195 (2) (2011) 270–281.

[14] B. Blankertz, S. Lemm, M. Treder, S. Hauf, K. R. M*ü*ler, Single-trial analysis and classification of ERP component- a tutorial, Neuroimage 56 (2011) 814–825.

[15] D. J. Krusienski, E. W. Sellers, F. Cabestaing, S. Bayoudh, D. J. McFarland, T. M. Vaughan, J. R. Wolpaw, A comparison of classification techniques for the P300 speller, Journal of neural engineering 3 (4) (2006) 299.

[16] K.-R. Müller, M. Krauledat, G. Dornhege, G. Curio, B. Blankertz, Machine learning techniques for brain-computer interfaces, Biomed. Tech 49 (1) (2004) 11–22.

[17] Y. Zhang, G. Zhou, J. Jin, Q. Zhao, X. Wang, A. Cichocki, Aggregation of sparse linear discriminant analyses for event-related potential classification in brain-computer interface, International journal of neural systems 24 (01) (2014) 1450003.

[18] J. Mak, Y. Arbel, J. Minett, L. McCane, B. Yuksel, D. Ryan, D. Thompson, L. Bianchi, D. Erdogmus, Optimizing the P300-based brain-computer interface: current status, limitations and future directions, Journal of neural engineering 8 (2) (2011) 025003.

[19] R. Tomioka, K.-R. Müller, A regularized discriminative framework for EEG analysis with application to brain–computer interface, NeuroImage 49 (1) (2010) 415–432.

[20] Y. Zhang, G. Zhou, Q. Zhao, J. Jin, X. Wang, A. Cichocki, Spatial-temporal discriminant analysis for ERP-based brain-computer interface, IEEE Transactions on Neural Systems and Rehabilitation Engineering 21 (2) (2013) 233–243.

[21] G. Rodríguez-Bermúdez, P. J. García-Laencina, J. Roca-González, J. Roca-Dorda, Efficient feature selection and linear discrimination of EEG signals, Neurocomputing 115 (2013) 161–165.

[22] J. Fruitet, D. J. McFarland, J. R. Wolpaw, A comparison of regres-

sion techniques for a two-dimensional sensorimotor rhythm-based brain-computer interface, Journal of Neural engineering 7 (1) (2010) 016003.

[23] Y. Shin, S. Lee, J. Lee, H.-N. Lee, Sparse representation-based classification scheme for motor imagery-based brain–computer interface systems, Journal of neural engineering 9 (5) (2012) 056002.

[24] U. Hoffmann, A. Yazdani, J.-M. Vesin, T. Ebrahimi, Bayesian feature selection applied in a P300 brain-computer interface, in: Signal Processing Conference, 2008 16th European, IEEE, 2008, pp. 1–5.

[25] L. Clemmensen, T. Hastie, D. Witten, B. Ersbøll, Sparse discriminant analysis, Technometrics 53 (2012) 406–413.

[26] R. O. Duda, P. E. Hart, D. G. Stork, Pattern classification, John Wiley & Sons, 2012.

[27] R. A. Fisher, The use of multiple measurements in taxonomic problems, Annals of eugenics 7 (2) (1936) 179–188.

[28] T. Hastie, R. Tibshirani, A. Buja, Flexible discriminant analysis by optimal scoring, Journal of the American statistical association 89 (428) (1994) 1255–1270.

[29] J. Ye, Leaste squares linear discriminant analysis, in: Proceedings of the 24th international conference on Machine learning, ACM, 2007, pp. 1087–1093.

[30] T. Hastie, R. Tibshirani, J. Friedman, The elements of statistical learn-

ing: data mining, inference and prediction, Springer Series in Statistics, 2009.

[31] K. Sjöstrand, L. H. Clemmensen, R. Larsen, B. Ersbøll, SpaSM: A matlab toolbox for sparse statistical modeling, Journal of Statistical Software Accepted for publication.

[32] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67 (2) (2005) 301–320.

[33] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society. Series B (Methodological) (1996) 267–288.

[34] J. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent, Journal of statistical software 33 (1) (2010) 1.

[35] A. Majumdar, R. K. Ward, Classification via group sparsity promoting regularization, in: Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on, IEEE, 2009, pp. 861–864.

[36] D. Calvetti, F. Sgallari, E. Somersalo, Image inpainting with structural bootstrap priors, Image and Vision Computing 24 (7) (2006) 782–793.

[37] F. J. Ibarrola, G. L. Mazzieri, R. D. Spies, K. G. Temperini, Anisotropic BV-L2 regularization of linear inverse ill-posed problems, Journal of mathematical Analysis and Applications.

[38] M. Basseville, Distance measures for signal processing and pattern recognition, Signal Processing 18 (1989) 349–369.

[39] S. Kullback, R. A. Leibler, On information and sufficiency, The annals of mathematical statistics 22 (1) (1951) 79–86.

[40] W. Gersch, F. Martinelli, J. Yonemoto, M. Low, J. Mc Ewan, Automatic classification of electroencephalograms: Kullback-leibler nearest neighbor rules, Science 205 (4402) (1979) 193–195.

[41] P. J. Moreno, P. P. Ho, N. Vasconcelos, A kullback-leibler divergence based kernel for SVM classification in multimedia applications, in: Advances in neural information processing systems, 2003, p. None.

[42] A. Gupta, S. Parameswaran, C.-H. Lee, Classification of electroencephalography (EEG) signals for different mental activities using kullback leibler (KL) divergence, in: Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on, IEEE, 2009, pp. 1697–1700.

[43] G. Mouret, J.-J. Brault, V. Partovi Nia, Generalized elastic net regression, in: JSM Proceedings, 2013, pp. 3457–3464.

[44] R. J. Tibshirani, J. E. Taylor, The solution path of the generalized lasso, The Annals of Statistics 39 (2011) 1335–1371.

[45] P. C. Hansen, Analysis of discrete ill-posed problems by means of the L-curve, SIAM review 34 (4) (1992) 561–580.

[46] M. Belge, M. E. Kilmer, E. L. Miller, Simultaneous multiple regularization parameter selection by means of the L-hypersurface with applications to linear inverse problems posed in the wavelet transform domain, in: SPIE's International Symposium on Optical Science, Engineering, and Instrumentation, International Society for Optics and Photonics, 1998, pp. 328–336.

[47] C. Ledesma-Ramirez, E. Bojorges-Valdez, O. Yáñez-Suarez, C. Saavedra, L. Bougrain, G. G. Gentiletti, An open-access P300 speller database, in: Fourth International Brain-Computer Interface Meeting, 2010.

[48] B. Blankertz, K.-R. Muller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlogl, G. Pfurtscheller, J. R. Millan, M. Schroder, N. Birbaumer, The bci competition III: Validating alternative approaches to actual BCI problems, IEEE transactions on neural systems and rehabilitation engineering 14 (2) (2006) 153–159.

[49] M. Billinger, I. Daly, V. Kaiser, J. Jin, B. Z. Allison, G. R. Müller-Putz, C. Brunner, Is it significant? guidelines for reporting BCI performance, in: Towards Practical Brain-Computer Interfaces, Springer, 2012, pp. 333–354.

[50] T. Fawcett, An introduction to ROC analysis, Pattern recognition letters 27 (8) (2006) 861–874.

[51] A. P. Bradley, The use of the area under the ROC curve in the evaluation

of machine learning algorithms, Pattern recognition 30 (7) (1997) 1145–1159.

[52] B. Blankertz, M. Tangermann, C. Vidaurre, S. Fazli, C. Sannelli, S. Haufe, C. Maeder, L. Ramsey, I. Sturm, G. Curio, et al., The berlin brain-computer interface: non-medical uses of BCI technology, Frontiers in neuroscience 4.