IDIAP COMMUNICATION

Feature Selection Methods on Distributed Linear Inverse Solutions for a Non-Invasive Brain-Machine Interface

Laurent Uldry [a] Pierre W. Ferrez [b]
José del R. Millán [b]

IDIAP–Com 07-04

March 2007

[a] EPFL student, MSc. internship performed at IDIAP
[b] IDIAP Research Institute

IDIAP Research Institute **www.idiap.ch**
Rue du Simplon 4 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

Contents

Acknowledgements

The elaboration of this Master thesis would not have been possible without the help of several persons. Firstly, I would like to thank Prof. José del R. Millán, who gave me the opportunity to work at IDIAP research institute during the period of my Master thesis. His leadership, availability and enthusiasm were an important source of motivation. I also would like to thank my two other advisors, Pierre Ferrez and Ferran Galan Moles, for their permanent help in machine learning processes, as well as the rest of the BCI group: it was a real pleasure to interact with my colleagues in a dynamic and positive atmosphere.
A special thanks goes to Febo Cincotti, for his invaluable contribution in my understanding of inverse models. His answers to my questions about the work of his group were so quick and precise that I did not even feel the distance between Rome and Martigny. In the same way, I would like to thank Rolando Grave de Peralta Menendez and Sara Gonzalez Andino for their precise answers about inverse solutions, when I needed some more explanations.
Finally, I would like to thank my parents for their precious support and encouragement throughout my undergraduate years of study.

1 Introduction

1.1 Brain Machine Interface Systems

Today's Brain-Machine Interfaces (BMIs) and Brain-Computer Interfaces (BCIs) represent a wide and active research field in the context of neurophysiology, neuroengineering, signal processing and machine learning. The main goal of Brain-Machine Interfaces is to allow severly handicapped persons to communicate with their environment and recover motor abilities, by means of an artificial interface controlled in real time by electrical brain activity. The art of interfacing the brain with artificial devices, such as computers or neuroprosthesis, has been described in several scientific articles: new readers could, for example, have a relatively complete review of invasive and non-invasive[1] BMI research with [28], [31] and [50]. Recently, it was proven that a mobile robot could be guided by a non-invasive BCI system in a realistic environment [33]; this shows great promise for future applications to real life, for example with the creation of an intelligent wheelchair or an artificial limb. In this work, exclusively non-invasive BCI systems are considered, and we will simply refer as "BCI systems" from now. This choice is motivated by the fact that such interfaces can be used directly on human beings, and therefore provide ethically correct solutions and applications in the short term.

The architecture of a standard BCI system illustrates the multidisciplinary characteristic of this scientific discipline. In order to extract meaningful commands and information from raw electrical activity of the brain, several important issues have to be addressed, as shown in figure 1.

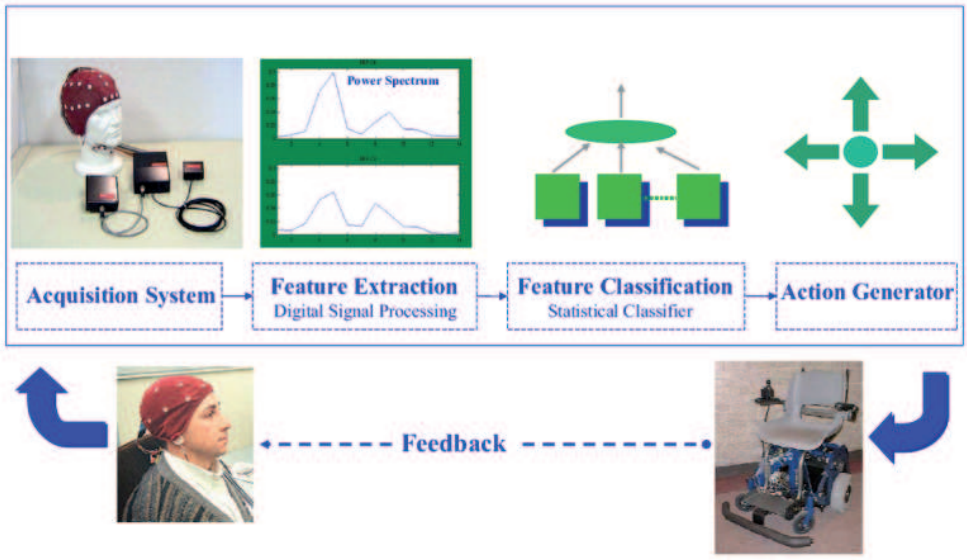

Figure 1: Architecture of a standard BCI system. (Source: Courtesy of J. del R. Millán)

- The first step, and maybe the most important, is the *acquisition* of the neurophysiological signal. The importance of having a clean signal from the beginning is crucial for later processing, and the acquisition system must be chosen and used with special care. In section 4, a system measuring *electroencephalogram*[2] (EEG) activity is described.
- The second part of the process represents the main point of this work, namely the selection of key features in the input signal, allowing to extract only the most relevant EEG components,

[1] Non-invasive means outside the skull, without affecting it surgically.
[2] See section 2.1 for a definition.

for which the performance of the system is the best. Special attention must be paid to this *feature selection* process, since it can improve the efficiency of the BCI system, both in terms of recognition rate and computational load.

- Once the features have been chosen, the BCI system must determine, for a given EEG activity, the most probable command meant by the user. This process is assessed by a *classification procedure*, and is directly related to machine learning algorithms.

- Finally, the extracted command must activate the BCI system according to the user's intention. The system output can be for example a motor command, or the choice of a letter in order to write a message. Moreover, a feedback has to be provided to the user in order to close the loop, in case of an error in the system detection. This final step is more related to robotics, and is beyond the scope of this work.

Recently, non-invasive methods estimating intracranial sources from scalp EEG have been developed by several scientific groups. These methods, often called *inverse solutions*, could be of high interest for BCI research, since estimated intracranial activity could provide a better spatial resolution than mere EEGs in order to decode brain activity, and thus user's intents. Moreover, these models allow a better understanding of dynamic neuronal processes in humans. Such methods are described in section 2 and are extensively used in the context of BCI throughout this study.

1.2 Outline of the Thesis

More precisely, this thesis focuses on the coupling of BCI feature selection methods with inverse solutions estimating intracranial activity. Indeed, approximating neuronal sources implies that the number of measurement points increases drastically. Therefore, feature selection methods become essential in order to minimize computational load. Moreover, we want to assess the potentialities of integrating inverse solutions in a BCI system in order to improve overall performances. The following points are presented:

- In chapter 2, the challenging issue of estimating non-invasively intracranial activity is addressed. The mathematical framework of the problem as well as three different models used during this work are described.

- In chapter 3, three feature selection methods that have been studied and developed during this work are described. The first method is the well-known Relief algorithm, and its modified versions; the second method is an improved version of a simple power discriminant method, and the third method is an algorithm based on linear discriminant analysis.

- Finally, all methods and models are compared and tested in a specific application related to BCI research: detection of error-related potentials. In chapter 5, the state of the art of error-related potentials is presented; on this basis, we can compare in chapter 6 the different feature selection methods in terms of classification accuracy. Chapter 7 compares the different inverse solutions, both in terms of localization of the selected features and in terms of classification of cognitive states. Finally, in chapter 8, we assess the potential improvements of integrating inverse solutions in a BCI system with respect to a standard BCI system based on EEG. All these investigations aim at providing an extension to a previous study made at IDIAP about error-related potentials.

Part I
Methods

2 Estimation of Intracranial Activity from Scalp EEG

The main goal of this thesis is to use estimated intracranial activity combined with feature selection methods in order to improve current BCI systems. In this chapter, we provide a theoretical framework of intracranial activity estimation and its underlying concepts. Besides, three inverse models that have been used during our studies are presented, with an emphasis on their respective specificities.

It is worth to note that *brain electromagnetic tomography*, i.e. the non-invasive three-dimensional reconstruction of the neuronal sources of the brain's electrical activity measured at the scalp, is a very wide and complex research field. During this work, we only considered these inverse solutions as available tools that we integrated in BCI systems; a description of the design and development of such models is beyond the scope of our work, but interested readers will find several references throughout this chapter. However, this chapter should provide all the necessary elements for a good understanding of our studies.

2.1 EEG Measurements

Before presenting the inverse models that we studied during this work, we shortly remind the basic principles of electroencephalogram (EEG) measurements; plenty of books, like for example [45], can give supplementary information on EEG signals, EEG processing and applications.

EEG measures the joint electrical activity of millions of active neurons in the brain. It can be measured with electrodes at the surface of the scalp, or intracranially; during this current study, only non-invasive scalp EEG has been investigated. The EEG activity mainly reflects the more or less synchronous activation of a large population of neurons, and more precisely their *postsynaptic activity*; the intracranial mean measure of this postsynaptic activity is called *local field potentials* (LFP). For more details about synaptic transmission and neuronal activation, refer to [6].

If in a large population, neurons are spatially aligned and have a synchronous activity, the resulting superimposed electrical field will be detected by electrodes at the scalp surface. This situation is often encountered for cortical pyramidal neurons, since they are oriented perpendicularly to the cortical surface, and their activity is thus most likely to be measured by EEG (see figure 2).

EEG can give neuronal information within a millisecond timescale: this very good temporal resolution allows to better understand neuronal dynamics and is the biggest advantage in using this technique, with respect to other imaging techniques such as magnetic resonance imaging (MRI) or positron-emission tomography (PET). However, the distance between the electrodes and the actual source of neuronal activity is an important drawback of EEG measurements, since it creates a low-pass filtering on the source signal. Thus, spatial resolution can become a problem in order to precisely describe neuronal processes; for this reason, estimation of intracranial activity from scalp EEG is a key challenge in neuronal data processing.

2.2 The Inverse Problem: A General Approach

Estimating the neuronal sources that generated a given potential map at the scalp surface requires the solution of an *inverse problem*. Such inverse problems are always initially undetermined, i.e. there is no unique solution. These problems require therefore supplementary a priori constraints in order to be univocally solved. The ultimate goal is then to un-mix the signals measured at the scalp, attributing to each brain area its own estimated temporal activity.

Historically, two different possible directions have been investigated in order to solve this inverse problem and find the generators of a given scalp activity; a global review can be found in [30]. On one hand, the so-called *dipole localization models* assume that only a limited number of generators are active over a period of time (e.g. [44], [11]); these generators are typically modeled as *equivalent current dipoles* (ECD). The number of generators that can be active at a given time is limited by the number of electrodes used for EEG measurements; thus, when in a given problem, the exact number of dipole sources cannot be determined a priori, this family of methods is not very appropriate. In such cases,

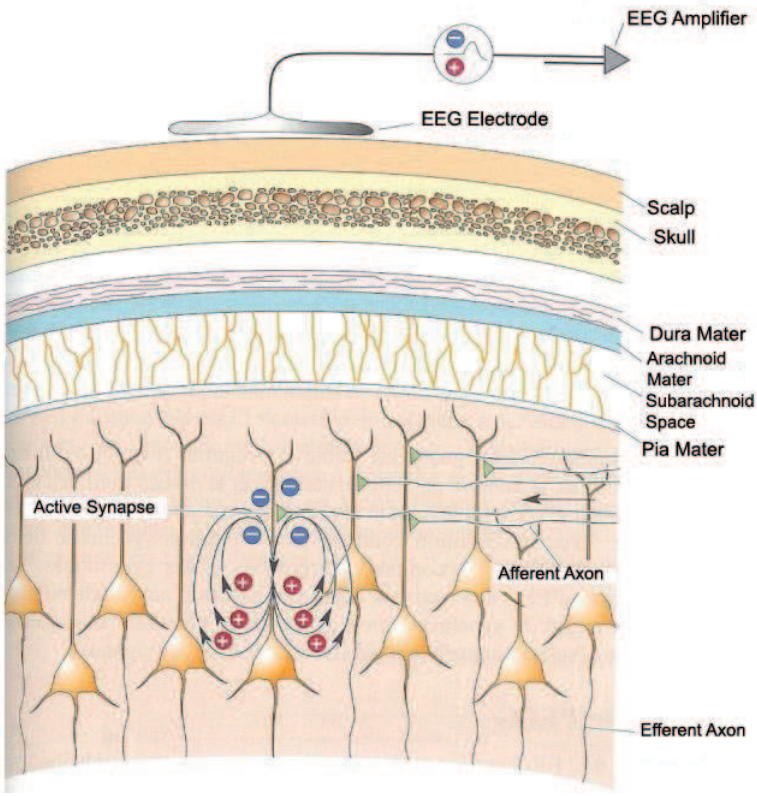

Figure 2: EEG principle: electrical fields generated by aligned pyramidal cells. (Source: Bear [6], 2001, p.637)

distributed models based on the linear theory in conjunction with mathematical and/or biophysical a priori constraints are more likely to be used (e.g. [10], [18], [1], [4], [3],[24]); these distributed models do not need a priori assumptions about the number of source generators, and estimate cortical current density by using sophisticated computational algorithms and detailed geometrical models of the head as volume conductor. With this approach, typically thousands of ECD covering evenly the cortical mantle are used, and their strength is estimated by using linear inverse procedures. In this work, only distributed models are considered.

2.2.1 Distributed Linear Inverse Estimation

For clarity purposes, we adopt the notation used in [9] for the formulation of inverse estimation, and we follow a similar reasoning to present the general form of a distributed linear inverse estimation. Assuming a measurement noise $\boldsymbol{n}$, an estimate of the dipole source configuration that generated a scalp potential $\boldsymbol{b}$ is obtained by solving the linear system:

$$\boldsymbol{A}\boldsymbol{x} + \boldsymbol{n} = \boldsymbol{b} \tag{1}$$

where $\boldsymbol{A}$ is a $m \times n$ matrix with m the number of sensors and n the number of modeled sources. The matrix $\boldsymbol{A}$ is called the *leadfield matrix*: the j^{th} column A_j represents the potential distribution over the m sensors due to each unitary j^{th} cortical dipole, and the collection of A_j describe how each dipole generates the potential distribution over the head model. The estimation of the cortical current density $\boldsymbol{x}$ is called the solution of the linear inverse problem, or inverse solution. In most

cases, the dimension of the vector $\boldsymbol{x}$ is greater than the number of measurements $\boldsymbol{b}$ of about one order of magnitude; thus, the linear system is strongly under-determined, and can have an infinite number of possible solutions. In order to solve this problem for a unique solution, assuming that $\boldsymbol{n}$ is normally distributed, a regularization scheme utilizing the Lagrange multiplier λ is applied, and the following functional has to be minimized:

$$\hat{\boldsymbol{x}} = \arg\min_{x}(\Phi), \qquad \Phi = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2_{\boldsymbol{M}} + \lambda^2 \|\boldsymbol{x}\|^2_{\boldsymbol{N}} \tag{2}$$

where the matrix $\boldsymbol{N}$ is the metric of the source space, i.e. the space of the current strength solutions $\boldsymbol{x}$, and the matrix $\boldsymbol{M}$ is the metric of the data space, namely the space in which $\boldsymbol{b}$ is considered. If no a priori information is added to equation (2), $\boldsymbol{M}$ and $\boldsymbol{N}$ are set to identity, and the estimation made is called *minimum norm estimation* (MN). Interpreting (2), it appears that on one hand, we try to minimize the energy of the error on the sensor data, given by the first term of Φ. On the other hand, a second term involving the energy of the source $\boldsymbol{x}$ regularizes the ill-posed problem: this term, modulated by λ, tends to minimize the overall intensity of the current distribution. At the end, a unique solution will be found, because only one combination of intracranial sources fit exactly the data, and has at the same time the lowest overall intensity. The problem is that the algorithm favors weak and localized activation patterns, instead of solutions with strong activation of a large number of solution points. Thus, the MN algorithm favors *superficial* sources, since less activity is required in superficial solution points to provide a certain surface voltage distribution: such models are not satisfying, because it means that deeper sources are incorrectly projected on the surface of the scalp. In order to cope with this problem, a well-known solution proposes to take into account a compensation factor for each dipole that equalizes the visibility of the dipoles from the sensors point of view. This so-called *column norm normalization* changes the source metric $\boldsymbol{N}$ as follows:

$$(\boldsymbol{N}^{-1})_{ii} = \|A_{\cdot i}\|^{-2} \tag{3}$$

with $(\boldsymbol{N}^{-1})_{ii}$ the i^{th} element of the inverse of the diagonal matrix $\boldsymbol{N}$ and $\|A_{\cdot i}\|$ the L_2 norm of the i^{th} column of the lead matrix $\boldsymbol{A}$. The use of this definition of the matrix $\boldsymbol{N}$ is known as *weighted minimal norm solution* (WMN), and penalizes dipoles close to the sensors in the solution of the inverse problem, since they have a large $\|A_{\cdot i}\|$. Thus, WMN solutions provide better estimates of intracranial activity, especially in the case of deep sources.
Equations (1), (2) and (3) set a general framework for distributed linear inverse models. From that point, a lot of free parameters have to be carefully chosen in order to converge to the best unique solution as possible. For example, the choice of additional constraints is crucial in terms of model specificity, and can drastically change the behaviour of the inverse solution. Additional constraints come from assumptions about likely current source distribution and statistics, sensor statistics, and information from other imaging techniques. In the next section, three inverse models with different assumptions are presented.

2.3 Inverse Solutions

2.3.1 CCD Inverse Model

The first presented model, that we will call CCD inverse model for "cortical current density inverse model", has been developed and provided by a research group working in the IRCCS Fundazione Santa Lucia, located in Rome[3]. References about this approach can be found in [4] and [5]. The model aims at providing an estimation of the activity of the cortical mantle. The procedure follows the reasoning of section 2.2.1 and includes:

1. a realistic magnetic resonance-constructed average head model.
2. multi-dipole cortical source model.

[3] http://www.hsantalucia.it/

3. regularised, weighted, minimum-norm linear inverse source estimate based on boundary element mathematics (WMN).

First, a geometrical reconstruction of the cortical surface is obtained from magnetic resonance imaging (MRI). In this model, the 152 subjects average brain of Montreal Neurological Institute[4] was used to have a realistic head model. At that point, an important anatomical constraint is considered: it is assumed that much of the observable EEG is produced by currents flowing in the apical dendrites of cortical pyramidal cells. The columnar organization of the cortex implies that the resulting local dipole moment is assumed to be oriented perpendicularly to the cortical surface. Thus, if the shape of the cortical mantle is known, we can divide it into patches that are sufficiently small so that a dipole in the center of a patch is representative of any dipole distribution within the patch. With the constraint of perpendicular orientation of the dipoles, the inverse problem reduces to estimating scalar distributions of dipole strength over the oriented patch.
In the case of the CCD inverse model, the MRI-based reconstruction of the head models the cortical mantle as a polyhedron with triangular faces, preserving the general features of the neocortical envelope; then, an orthogonal unitary ECD was placed in each node (or *vertex*) of the triangulated surface. On the whole, 3013 discrete current dipoles are chosen to represent the continuum current source distribution; see figure 3 for a view of the brain provided by the model.

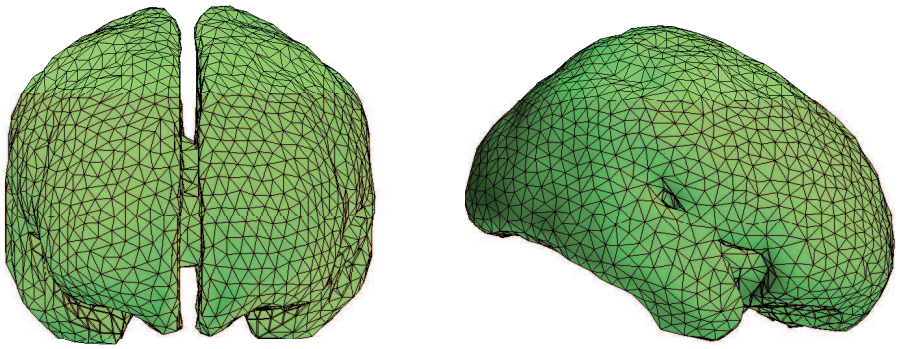

Figure 3: CCD inverse model: front and lateral views of the cortical mantle modeled with 3013 vertices of a polyhedron with triangular faces.

The second constraint of the CCD inverse model is based on WMN estimates, and forces the dipoles to explain the recorded data with a minimum or a low amount of energy, whitout penalizing too much deeper sources, as explained in section 2.2.1. During this thesis work, the CCD inverse model has been used extensively, both for localization studies and for BCI-oriented classification procedures, and showed impressive results.

2.3.2 sLORETA Inverse Model

The second inverse model is a *standardized low resolution brain electromagnetic tomography* method (sLORETA): this software, known for its zero localization error, is freely provided by the KEY Institute for Brain-Mind Research[5] in Zürich. We used this software only as a localization tool throughout the studies, but a description of the method for localizing sources is useful here. The volume conductor model is a three-shell spherical head model registered to the Talairach human brain atlas [46],

[4] http://www.bic.mni.mcgill.ca/
[5] http://www.unizh.ch/keyinst/index.html

available as a digitzed MRI from the Montreal Neurological Institute imaging center; the solution points are placed on a 3D regular grid covering the whole brain.
Historically, a first method called low resolution electromagnetic tomography (LORETA) was introduced by Pascual-Marqui in [38] and [37]. In this method, an additional constraint called *Laplacian Weighted Minimum Norm* was added to the typical WMN depth weighting. This method selects the solution having the smoothest spatial distribution by minimizing the Laplacian of the weighted sources, a measure of spatial roughness. A physiological assumption is hidden behind this method: the model assumes that neighboring grid points, i.e. neighboring neurons, are more likely to be synchronized (similar orientation and strength) than grid points that are far from each other. Thus, this maximization of smoothness is applied to find a unique distribution of electrical activity in the brain. The characteristic feature of this solution is its low spatial resolution, which is a direct consequence of the smoothness constraint: LORETA provides rather blurred images of a point source, conserving the location of the maximal activity with a certain degree of dispersion, as shown in figure 4, generated by sLORETA. Furthermore, the assumption that two neighboring areas are correlated has to be considered with caution; indeed, functionally distinct areas can be anatomically very close. However, the localizations made by LORETA are satisfying in most cases.

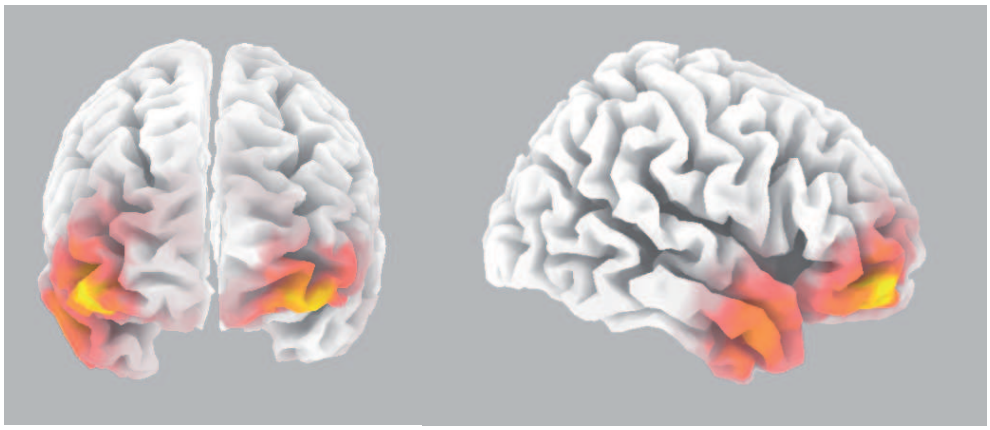

Figure 4: sLORETA model: front and lateral views of the brain during a localization study.

Recently, a new version of the method, called standardized low resolution brain electromagnetic tomography (sLORETA) has been developed, and yields images of standardized current density with zero localization error. The difference with the previous algorithm is that sLORETA employs the current density estimate given by the minimum norm solution, and localization inference is based on standardized values of the current density estimates, as explained in [36]. Only by itself, the solution of the MN inverse solution is incapable of correct localization of deep sources. With this standardization process, sLORETA reaches zero localization error, even if the sources are deep. However, the drawback of this method is that because of this standardization process, sLORETA *is not* an authentic solution to the inverse problem; according to the KEY Institute website, it seems that a new version of the software, called eLORETA (for "exact low resolution brain electromagnetic tomography") will soon be released, and will provide a formal solution providing exact localization to test point sources.

2.3.3 ELECTRA-LAURA Inverse Model

The third inverse solution presented here is slightly different from the previous models; this is a distributed source model called ELECTRA (for electrical analysis), developed by Grave de Peralta Menendez and colleagues ([23], [24]) in Geneva University Hospital (HUG). In conjunction with this linear distributed model, a regularization strategy called LAURA (local autoregressive averages) is

applied on the inverse solution.
The difference of ELECTRA-LAURA model is that the source model is changed with respect to the previous models, based on the following considerations: the microscopic current flowing in biological tissue can be decomposed into two terms: a *primary current* (or *active current*) and a *secondary current* (or *volume current*). Primary currents are induced by ionic flow between intra- and extra-cellular space in activated neurons, whereas volume currents are passive currents representing the electrical response of the media to compensate charge accumulation driven by primary currents, according to electrochemical gradient. It has been shown in [39] that only volume currents are measured by EEG, and not active currents: this observation is crucial, since the mathematical implication is that the currents measured by EEG are ohmic and can be modeled as *irrotational* currents. Thus, the ELECTRA source model only estimates ohmic currents; it is not an inverse solution, but rather a source model in which the generators of the scalp maps are the intracranial potentials instead of the usual 3D current densities.
In order to reach a unique solution, a regularization strategy called LAURA (for "local auto-regressive averages") incorporates biophysical laws as constraints in the MN algorithm ([24], [22]). According to Maxwell equations, the strength of the sources fall off with the inverse of the cubic distance for vector fields, and with the inverse of the squared distance for potential fields. LAURA integrates these laws in terms of a local autoregressive average with coefficients depending on the distances between solution points.
The model provided by Geneva's group is composed of a solution space formed by 4024 nodes (referred to as *voxels*) homogeneously distributed within the inner compartment of a realistic head model: once again, the head model is the average brain of Montreal Neurological Institute. The voxels are restricted to the grey matter and form an isotropic grid of 6 *mm* resolution. A view of the solution space of this model is presented in figure 5.

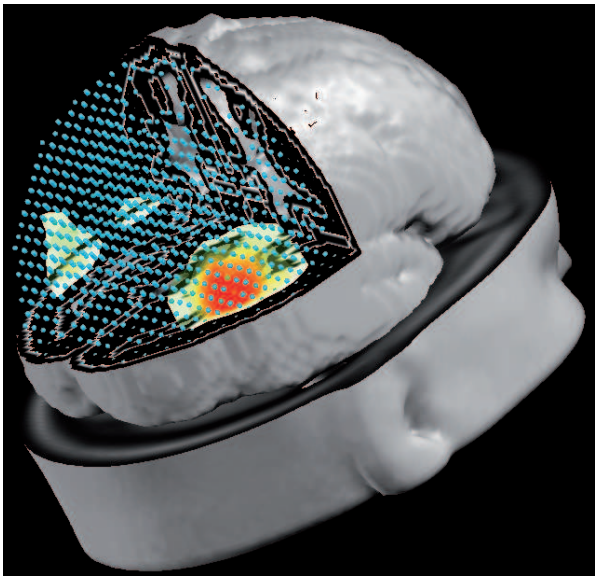

Figure 5: Isotropic 3D grid of voxels provided by ELECTRA-LAURA inverse model.

The most interesting point with ELECTRA-LAURA inverse solution is that the model allows an estimation of the 3D distribution of electrical potentials in the brain as if they were recorded with intracranial electrodes. As we mentioned above, LFPs arise largely from dendritic activity over large brain regions and thus provide a measure of the input to a given area, and of the local processing within this area. Recently, LFPs revealed themselves to be of crucial interest for providing meaningful

information about neuronal processes related to motor actions (e.g. [13], [43], [40]), and more generally about brain dynamics (e.g. [49], [19]). Hence, estimating the LFP activity from the scalp EEG represent a very challenging and exciting issue, since it can provide a non-invasive way to investigate neuronal processes in humans with a highly relevant physiological meaning.

3 Feature Selection Methods: Filter Methods

In a BCI classification context, data describing a given mental task are fed to a classifier for decision making. Ideally, the classifier should be able to use whichever features are necessary, and discard the irrelevant features. However, it is known that the complexity of many learning algorithms depends on the number of input dimensions, as well as on the size of the data sample. Mostly for this reason, researchers are interested in reducing the dimensionality of the problem. Therefore, feature selection methods are crucial in order to choose a smaller number of features that will describe the data at best. In a classification task, the features have to be selected according to their ability to discriminate between the different classes of a given problem. Thus, a good feature selection should provide accurate discrimination and reduce the computational load. Therefore, feature selection has become a very active field of research in the context of BCI (see [32]).
This chapter describes three feature selection methods that have been implemented and tested during this work. Before presenting our methods in details, a brief introduction about feature selection modalities justifies our choice of methods. Then, section 3.2 introduces the so-called Relief and ReliefF algorithms. Section 3.3 describes a modified version of a simple power discriminant function. Finally, section 3.4 presents an algorithm based on linear disriminant analysis.

3.1 Filter Methods vs. Wrapper Methods

Feature selection aims at finding those relevant components for which the performance of the learned classifier is the best. From this idea, we can differentiate two processes, that we will consider as separated: feature selection on one hand, and induction, i.e., the process of learning the appropriate classifier, on the other hand. Depending on the relationship between these processes, it is possible to distinguish two important families of methods:

1. *Filter* methods - the feature selection is done before induction algorithm.
2. *Wrapper* methods - the feature selection process *uses* the induction algorithm.

Filter methods are applied on the entire dataset, and *before* the induction algorithm, as shown in figure 6. The name given to these methods is meaningful, since irrelevant attributes in the initial dataset are filtered, creating a simplified dataset for the induction algorithm. The main disadvantage of filter methods is that feature selection is completely independent of the induction algorithm, and the former cannot be guided by the classifier error rate. Indeed, the criteria used to decide if a feature is relevant or not vary from a filter method to another, and in most of cases, these criteria are not exactly the same as those of the induction algorithm. Thus, the best features selected by the filter method are not necessarily the best features according to the criteria of the induction algorithm.

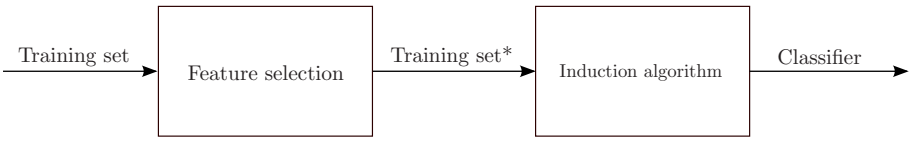

Figure 6: Schematic process of a filter method.

On the contrary, wrapper methods use the induction algorithm to make the selection, as shown in figure 7. Following a given strategy, the feature selection process explores the *state space* of each subset of features in the entire training set. For each state, i.e. each subset of features, the evaluation of the quality of the subset is done by an appropriate function executing the induction algorithm. The latter builds a classifier based on the simplified training set containing the features of the current subset, and estimates the performance of the classifier. From this estimation, the feature selection process decides to keep this subset of features, or to try another one. Going from one state to another

is done by using operators that delete or add features from or to the current set: starting from an empty set and adding features is called *forward selection*, whereas starting with the full set of features and deleting them is called *backward elimination*. At the end of the feature selection process, the selected relevant features are used to build the final classifier, which is tested on a testing set totally independent of the training set.
In the case of wrapper methods, there is a strong interaction between selection and induction algorithms. The estimated performance of the classifier is the quality criteria for the selection of features: in some sense, these features are specifically chosen for the final classifier. Thus, wrapper methods find more relevant features than filter methods in most of the cases. However, the major disadvantage of wrapper methods is the high computational complexity of the process. Exploring all the subset of features for a given training set implies a lot of iterations, and building a classifier for each iteration is highly time consuming.

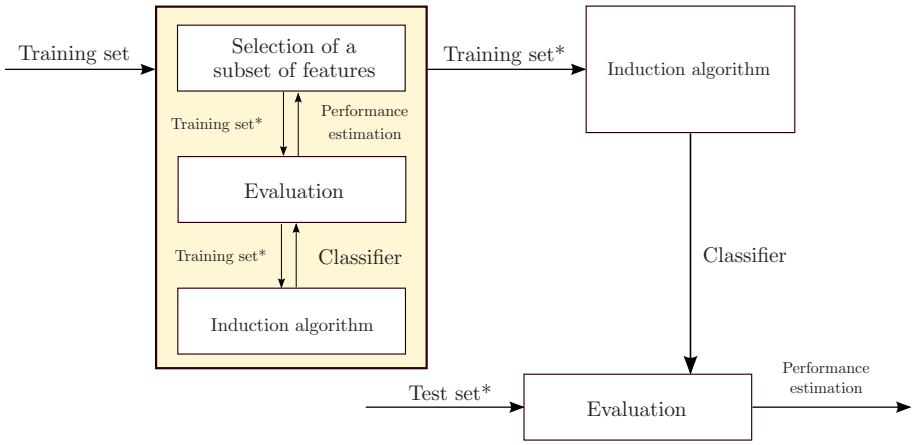

Figure 7: Schematic process of a wrapper method.

For BCI applications, it seems that wrapper methods are reasonably not appropriate, because of the slowness of the feature selection process. Indeed, the goal of a BCI system is to decode brain activity in real time; it is clear that wrapper methods don't allow this kind of quick processes. Therefore, this chapter focuses on filter methods only, so that if one of the methods shows great promise during offline[6] analysis, it will be possible to integrate it in a real BCI system.

3.2 Relief and ReliefF Algorithms

The Relief family of algorithms, firstly described in 1992 [26], is a group of general and successful attribute estimators, well described by Robnik-Šikonja and Kononenko [41]. This section provides a theoretical description of these methods, as well as a presentation of the practical implementation of the methods.

3.2.1 Theory of Relief and ReliefF Algorithms

In this section, we describe the Relief algorithms implemented during this work, and their theoretical properties. We assume that examples $I_1, I_2, \ldots, I_n$ in the instance space are described by a vector of attributes $A_i, i = 1, \ldots, a$, where a is the number of explanatory attributes, and are labelled with the target value τ_j. The examples are therefore points in the a dimensional space. We will first describe the original Relief algorithm, limited to classification problems with two classes; then we will discuss its ReliefF extension for multiclass problems.

[6]"offline" means "not in real time".

Relief Algorithm The original Relief algorithm [26] deals exclusively with two class problem. The main idea of the method is to estimate the quality of attributes (or features), by determining how well their values distinguish between instances that are near to each other. More precisely, Relief acts iteratively: in each iteration, Relief selects a random instance R_i, and then searches for two nearest neighbors: one from the same class, called *nearest hit* H, and the other from the other class, called *nearest miss* M. The quality estimation $W[A]$ is updated for all attributes A depending on their values for R_i, M and H; at the end, the weight assigned to every feature is a real value in the range $[-1; 1]$. A pseudo-code of the algorithm is given in figure 8.

Algorithm Relief
Input: for each training instance a vector of attribute values and the class value
Output: the vector W of estimations of the qualities of attributes

1. set all weights $W[A] := 0.0$;
2. **for** i := 1 **to** m **do begin**
3. randomly select an instance R_i;
4. find nearest hit H and nearest miss M;
5. **for** A := 1 **to** a **do**
6. $W[A] := W[A] - \text{diff}(A,R_i,H)/m + \text{diff}(A,R_i,M)/m$;
7. **end**;

Figure 8: Pseudo code of the basic Relief algorithm.

Function diff(A,I_1,I_2) is used to calculate the difference between the values of the attribute A for two instances I_1 and I_2. For numerical attributes, it is defined as:

$$\text{diff}(A,I_1,I_2) = \frac{|value(A, I_1) - value(A, I_2)|}{max(A) - min(A)} \tag{4}$$

This function is also used to compute the distance between instances to find nearest neighbors. In this process, the total distance can be simply assumed to be the sum of distances over all attributes (Manhattan distance).

The idea behind the process of Relief algorithm can be intuitivey understood. On one hand, different values of the attribute A between R_i and M means that this attribute A tends to separate two instances with different class labeling; in this case, the attribute has a desired discriminative effect, and the quality estimation $W[A]$ is thus increased. On the other hand, if A shows different values for R_i and H, $W[A]$ will be decreased, because attribute A tends to separate instances of the same class. The whole process is repeated m times, where m can be defined by the user. At the end of the iterative process, the vector W will give for each feature a score representing the ability of the feature to separate instances of different classes and keep instances of the same class near to each other.

ReliefF algorithm An extended version of Relief, called ReliefF algorithm, was developed in 1994 [27]. This algorithm is not limited to problems with two classes and is known to be more robust to noise. The difference with the original Relief algorithm is that after having randomly selected an instance R_i, ReliefF searches for k of its nearest neighbors from the same class, called nearest hits H_j, and k nearest neighbors from each of the different classes, called nearest misses $M_j(C)$. The quality estimation $W[A]$ for all attributes is updated depending on their values for R_i, H_j and $M_j(C)$, as shown in the pseudo-code of figure 9. The contribution of all the hits and all the misses are averaged in the update formula, and the contribution for each class of the misses is weighted with the prior

probability of that class $P(C)$. In this work, all classes are assumed to have the same prior probability. The factor $1 - P(class(R_i))$ dividing each probability weight ensures that misses' probability weights sum to 1, thus providing symmetric contributions of hits and misses. The whole process is repeated for m times.

The most important difference of ReliefF algorithm is the user-defined parameter k, that has several advantages:

- Selection of k nearest hits and misses provides greater robustness of the algorithm concerning noise. To illustrate this assumption, let us consider a situation where two instances of a class are outliers, namely far from the mean of the class, but somehow near to each other. In this case, if one of the outliers is selected as R_i, the nearest hit H will surely be the other outlier. If only one neighbor is considered, most of the attributes will be very similar between the selected outlier R_i and his neighbor H, although they are not representative of the mean behaviour of the class, and the corresponding quality estimates will be increased. On the contrary, if several neighbors are observed, the other nearest hits H_i, $i = 2, ..., k$, will have different values of attributes from those of R_i. The weighted contribution of the k neighbor will thus update the vector W in a more appropriate way. In that sense, taking several neighbors applies a filtering on noisy data.

- The parameter k is also useful in order to control the locality of the estimates. When k is small, the quality estimation $W[A]$ of attribute A is based on the similarity of attribute A between instances that are near to each other, in a very local domain. When k increases, the weighted sum contributing to the update of $W[A]$ contains instances that are more distant from each other; the locality of the estimates is less restricted.

Algorithm ReliefF
Input: for each training instance a vector of attribute values and the class value
Output: the vector W of estimations of the qualities of attributes

1. set all weights $W[A] := 0.0$;
2. **for** i := 1 **to** m **do begin**
3. randomly select an instance R_i;
4. find k nearest hits H_j;
5. **for** each class $C \neq class(R_i)$ **do**
6. from class C find k nearest misses $M_j(C)$;
7. **for** $A := 1$ **to** a **do**
8. $W[A] := W[A] - \sum_{j=1}^{k} \text{diff}(A, R_i, H_j)/(m \cdot k) +$
9. $\sum_{C \neq class(R_i)} [\frac{P(C)}{1-P(class(R_i))} \sum_{j=1}^{k} \text{diff}(A, R_i, M_j(C))]/(m \cdot k)$;
10. **end**;

Figure 9: Pseudo code of the extended ReliefF algorithm.

3.2.2 Implementation and Validation under Matlab

Relief and ReliefF algorithms have both been implemented under Matlab 7.1[7]. Several versions of the algorithms have been implemented: one version is the basic Relief algorithm for two class problems, as described in section 3.2.1. ReliefF algorithm for multiclass problems has been coded as well, and a version of ReliefF for two class problems has been developed, in order to take advantage of the parameter k even for two class problems.

In order to verify that the algorithms have been implemented correctly, we reproduced an example given in the reference [41]. In this example, a Boolean problem is presented, where the class value is defined as $\tau = (A_1 \wedge A_2) \vee (A_1 \wedge A_3)$. Table 1 gives a schematic description of the problem.

Line	A_1	A_2	A_3	τ	Responsible attributes
1	1	1	1	1	A_1
2	1	1	0	1	A_1 or A_2
3	1	0	1	1	A_1 or A_3
4	1	0	0	0	A_2 or A_2
5	0	1	1	0	A_1
6	0	1	0	0	A_1
7	0	0	1	0	A_1
8	0	0	0	0	(A_1, A_2) or (A_2, A_3)

Table 1: Schematic description of the concept $\tau = (A_1 \wedge A_2) \vee (A_1 \wedge A_3)$ and the responsibility of the attributes for the change of the predicted value.

The right most column of the table shows the attributes responsible for the change of the predicted value. For example, in line 1, A_1 is responsible for the class assignment because changing its value to 0 would change τ to 0, while changing only one of A_2 or A_3 would leave τ unchanged. The other lines can be explained similarly. It is then possible to give an estimate of the importance of each feature: A_1 will get the estimate $\frac{4+2\cdot\frac{1}{2}+2\cdot\frac{1}{2}}{8} = \frac{3}{4} = 0.75$, since it is alone responsible for lines 1,5,6,7, shares the credit for lines 2 and 3, and cooperates in both credits for line 8. Similarly, A_2 and A_3 both get estimates $\frac{2\cdot\frac{1}{2}+\frac{1}{2}}{8} = \frac{3}{16} = 0.1875$. In order to scatter the concept and make the problem more difficult to solve, five random binary attributes, A_4 to A_8, were added besides the relevant features A_1, A_2 and A_3. ReliefF was then applied on this problem, and the results for the values of A_1, A_2 and A_3 are shown in figure 10. The X-axis of the figure represents the number of trials for one class; it means that the dataset will then contain $2 \cdot N_{trials}$ instances. We can observe that when the number of trials increases, the estimate for A_1 converges to 0.75, and the estimates for A_2 and A_3 approach the expected value 0.1875; of course, the convergence is not exact because we are observing a practical case of the theoretical concept, involving randomly generated data. The estimates of the random features A_4 to A_8 are set to 0 relatively quickly during the iterations.

3.2.3 Convergence of Relief and ReliefF Algorithms

Since the algorithms of Relief family are iterative, the issue of the convergence and the stability of the process have to be investigated, in order to assess the parameters that will provide a correct estimate of the quality of the features. Two concepts are crucial for iterative algorithms: the convergence of the algorithm on one hand, and the stability of the convergence on the other hand.

- Figure 10 shows that when the number of trials increases above a certain threshold, the algorithm will converge accurately. It is therefore likely to have big datasets containing a lot of single trials, so that the algorithm can approximate correctly the importance of each feature. However, even

[7] http://www.mathworks.com/

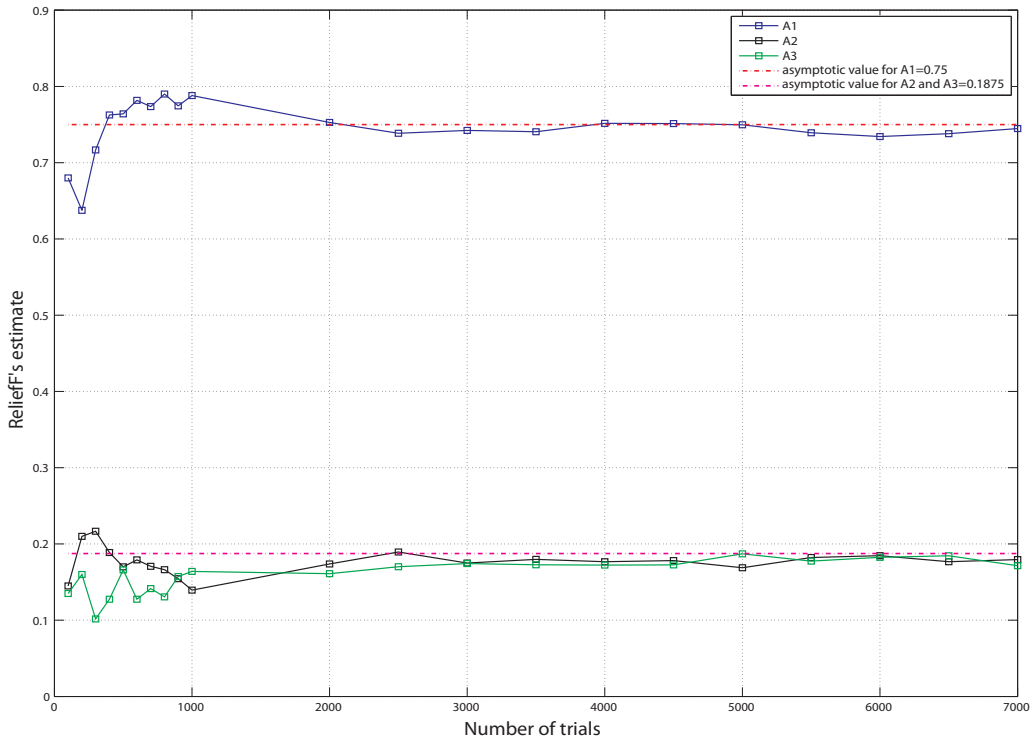

Figure 10: Estimates of the attributes made by ReliefF for the Boolean problem $\tau = (A_1 \wedge A_2) \vee (A_1 \wedge A_3)$.

if the number of trials is small, Relief(F) algorithm can provide a pretty good estimate of the features quality, as shown in the same figure.

- The second important free parameter of Relief(F) algorithms is the number of iterations made by the algorithm. This user-defined parameter seems to have a significative effect on the stability of the estimates. In order to show this dependency, we repeated the Boolean problem experiment of section 3.2.2 for different numbers of iterations, as shown in figure 11. For consistency purpose, the parameter N_{iter} is defined as a fraction of the current N_{trials} value, in order to apply always the same relative number of iterations with respect to the dataset, when N_{trials} increases. Figure 11 shows that the estimates of the features are more stable when $N_{iter} \geq 0.5 \cdot N_{trials}$; we can keep this value in mind as a reference, but each application could need a specific fine-tuning of N_{iter} in order to ensure the best convergence.

3.3 Modified Discriminant Power Method

In this section, we present a very simple but efficient feature selection method that we called *discriminant power function* (DP) [21]. More precisely, we implemented a modified version of the DP function that can deal with noisy data. The basic method will be first introduced, and then the modified method implemented during this work will be described.

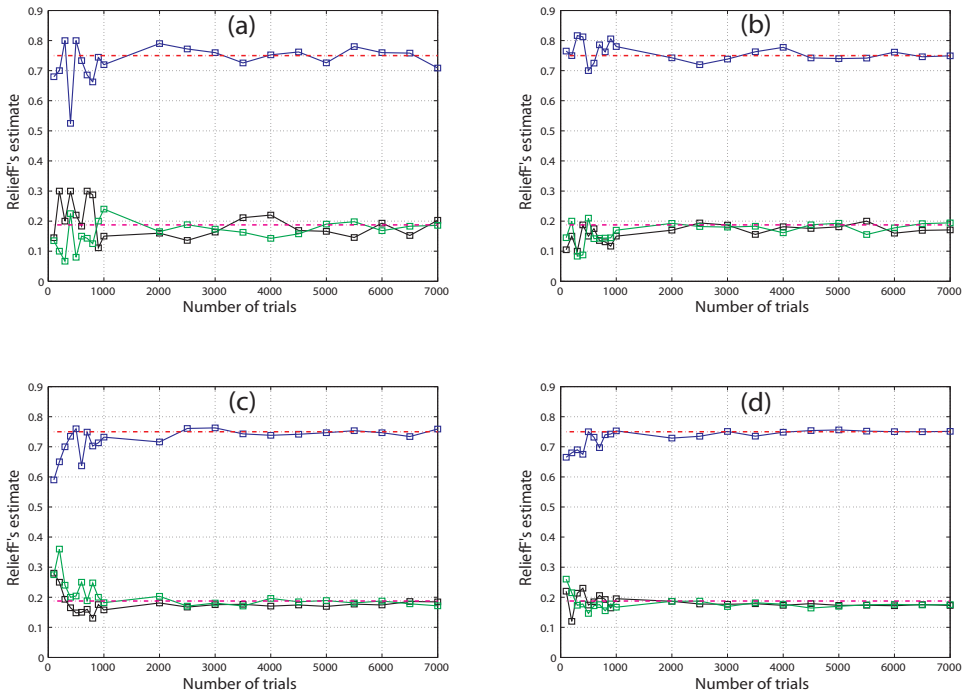

Figure 11: Comparison of convergence for different numbers of iterations: (a) $N_{iter} = 0.1 \cdot N_{trials}$; (b) $N_{iter} = 0.2 \cdot N_{trials}$; (c) $N_{iter} = 0.5 \cdot N_{trials}$; (d) $N_{iter} = 1 \cdot N_{trials}$.

3.3.1 Basic DP Function

A basic DP function estimates the quality of a given feature following a very simple principle: if the distribution of the feature, namely its *probability density function* (pdf), is different for each class, then the feature is a good candidate to discriminate between these classes. More precisely, let us take the example of a two class problem. If the distribution of the feature f for class 1, $pdf_1(f)$, is well separated from the distribution of the same feature for class 2, $pdf_2(f)$, then the feature f has a high discriminant power; otherwise if $pdf_1(f)$ and $pdf_2(f)$ are strongly overlapping, the discriminant power of feature f is low.

Actually, the basic version of DP function doesn't make an estimation of the pdf of each class for a given feature f, but simply looks for the maximum and minimum sample values of feature f for each class over all trials. With these boundaries $max(s_{fk})$ and $min(s_{fk})$ for the k^{th} class and for feature f, the algorithm can then calculate the proportion of samples of feature f lying in the non-overlapping zones between boundaries of each class. For a two class problem, the formula of the discriminant power of feature f would be:

$$ND_{f1} = \sum_{i=1}^{N_{t1}} \Big(1\big(s_{f1}(i) > max(s_{f2})\big) + 1\big((s_{f1}(i) < min(s_{f2})\big) \Big) \tag{5}$$

$$ND_{f2} = \sum_{j=1}^{N_{t2}} \Big(1\big(s_{f2}(j) > max(s_{f1})\big) + 1\big((s_{f2}(j) < min(s_{f1})\big) \Big) \tag{6}$$

$$\mathrm{DP}(f) = \frac{ND_{f1} + ND_{f2}}{N_{t1} + N_{t2}} \tag{7}$$

where N_{t1} and N_{t2} are the respective number of samples (or trials) for each class, s_{f1} and s_{f2} are vectors containing the samples of class 1 and 2 for feature f, ND_{f1} and ND_{f2} are the number of discriminant samples of each class located in non-overlapping zones, and $1(x)$ is a function defined by:

$$1(x) = \begin{cases} 1 & \text{if } x \text{ is true} \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

The basic DP algorithm returns a value DP(f) between 0 and 1 for each feature. This value can be thought of as the discriminant power of the feature, since it is the percentage of discriminant samples over all trials. A graphical representation of the process of the basic DP function can be found in figure 12. The score returned by the DP algorithm for the example of this figure is the number of samples lying out of the grey shaded area divided by the total number of samples of both classes.

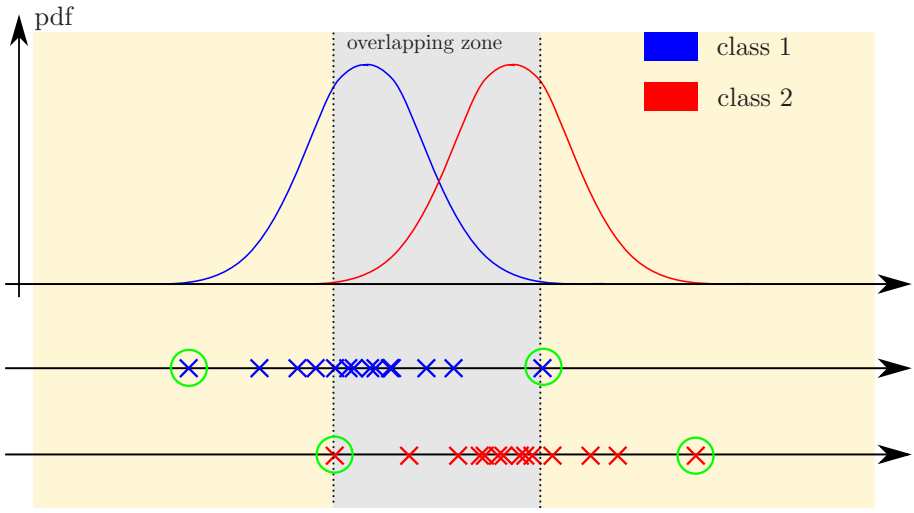

Figure 12: Schematic representation of basic DP algorithm.

The advantage of this basic method is that the mathematic operations involved in the process are very simple, and thus quickly computed: this feature selection method is indeed very fast and practical for online[8] applications of a learning process. However, the basic DP function has a major drawback: it is highly sensitive to noisy data. An example of this weakness is shown in figure 13. The distributions shown in figure 13 are relatively well separated, and should give a good result in terms of disriminant power. The only difference with figure 12 is that one of the samples of class 1 is corrupted by noise, and can be considered as an outlier. The basic DP function will assume this outlier sample to be the maximum of class 1 distribution, and the resulting "overlapping" zone will entirely encompass the distribution of class 2, since the grey shaded area of figure 13 is only defined by the extrema of class 2. Comparing this situation with figure 12, it is straightforward to conclude that in this case, the resulting score of DP algorithm will not be representative of the non-overlapping property of the observed classes, even though the distributions of class 1 and 2 are not totally overlapping.

3.3.2 Modified DP Function

In order to cope with noisy data, a little preprocessing step has been added to the original DP algorithm: the distributions of both classes are first truncated, in order to keep only a given percentage of the data around the mean of each distribution. This pruning step will discard outlier data if the

[8] "online" means "in real time".

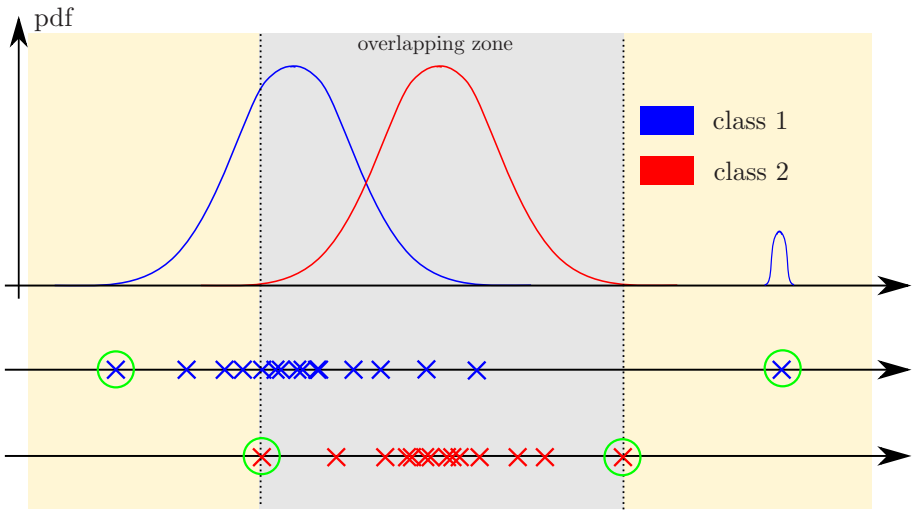

Figure 13: Sensitivity to noise of basic DP algorithm.

percentage of truncated data is well chosen. It is known that under the assumption of a normal distribution, and for a given integer k, a certain percentage of the values are within k standard deviations from the mean μ. Table 2 gives several different values of confidence intervals and the corresponding proportion of data within the interval, for normal distributions. If we are not sure of

Confidence Interval	% of data in the interval
$[\mu - \sigma; \mu + \sigma]$	68%
$[\mu - 1.177 \cdot \sigma; \mu + 1.177 \cdot \sigma]$	76%
$[\mu - 2 \cdot \sigma; \mu + 2 \cdot \sigma]$	95%
$[\mu - 3 \cdot \sigma; \mu + 3 \cdot \sigma]$	99%

Table 2: Confidence intervals and their corresponding proportion of data within the interval for normal distributions.

the normality of the distribution, a more general formula is provided by *Chebyshev's inequality*:

$$\Pr(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2} \tag{9}$$

Note that only the case $k > 1$ gives useful information. Thus, even if the distributions are not normal, at least $100 \cdot (1 - \frac{1}{k^2})\%$ of the values are within k standard deviations from the mean μ.

Keeping a too high percentage of the original data could maintain some outlier data in the pruned dataset, and the DP score would remain meaningless. The issue is thus to assess which is the percentage of noisy data in a given application. In this sense, the *full width at half maximum* (FWHM) value seems to be a good choice for an upper limit in the pruning process. The FWHM value is an expression of the extent of a function, given by the difference between the two extreme values of the independent variable at which the dependent variable is equal to half of its maximum value. An illustration of FWHM value is given in figure 14 for a normal distribution; in this specific case, the relationship between FWHM and the standard deviation is:

$$\text{FWHM} = 2 \cdot \sqrt{2 \cdot \ln(2)} \cdot \sigma \approx 2.354 \cdot \sigma \tag{10}$$

and FWHM is the interval $\mu \pm \sqrt{2 \cdot \ln(2)} \cdot \sigma$, which means that we keep 76% of the samples after having truncated the distributions, according to normal distribution knowledge. In the implementation of the

modified version of DP algorithm, we decided to approximate the distributions by normal distributions, and simply keep data in the interval $\mu \pm \sqrt{2 \cdot \ln(2)} \cdot \sigma = \mu \pm 1.177 \cdot \sigma$, assuming that the maximum value is near from the mean μ. Of course, taking the interval $[\mu - 1.177 \cdot \sigma; \mu + 1.177 \cdot \sigma]$ as the threshold of our algorithm is arbitrary and gives a comfortable margin; in any specific application, if the percentage of noisy data is precisely known, the pruning threshold can be tuned so that only outlier data are discarded. Nevertheless, it is crucial to keep in mind that if the amount of noisy data is big, the distributions will not be strictly normal anymore, since the mean will be shifted away from the maximum value of the distribution: in this case, the percentages given in table 2 are too optimistic, and the pruning threshold will have to be chosen with care. But in any case, we believe that the approximation of nearly normal distributions is acceptable for most applications, and our choice of FWHM as truncated interval provides a good trade-off in order to remove noise without losing too much information about the actual distribution.

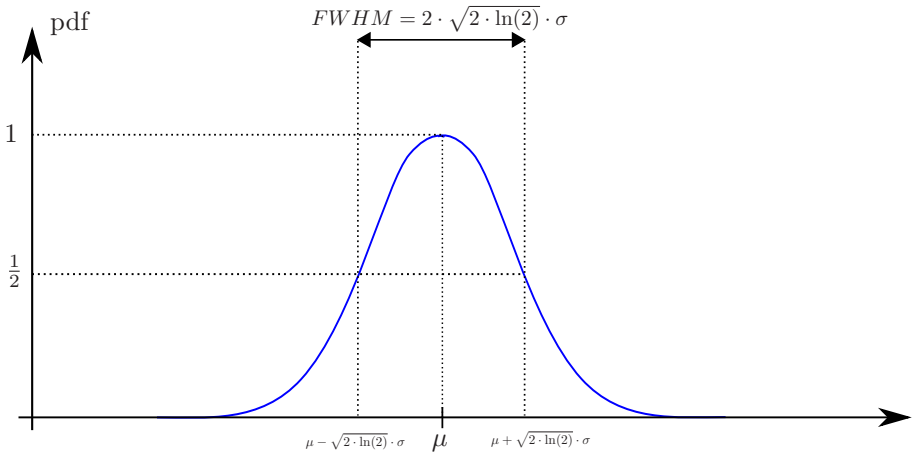

Figure 14: Illustration of full width at half maximum value.

With this modified DP algorithm, the problematic situation of figure 13 can be solved, as shown in figure 15. If the percentage of noisy data is not too big, the outliers of class 1 will be discarded by the preprocessing step, and the remaining truncated distributions will reflect the real non-overlapping property of the classes. Indeed, we can see that the grey shaded overlapping zone is defined by extreme values of both classes, and the resulting score given by the modified DP algorithm is thus meaningful in terms of discriminant power.

3.3.3 Validation with Synthetic Data

In order to compare the basic version of DP algorithm with the modified version implemented during this work, a typical example with synthetic data has been generated and analysed. In this two class problem, both class have initially normal distributions; class 1 has a mean $\mu_1 = 0$ and a standard deviation $\sigma_1 = 0.7$, while class 2 has a mean $\mu_2 = 3$ and a standard deviation $\sigma_2 = 0.7$. In order to make the situation more problematic, we added outlier samples to the first class, with mean $\mu_{noise} = 10$ and a standard deviation $\sigma_{noise} = 0.3$. It is important to note that the population of outlier data is relatively big, since it was set to 17% of the total population of class 1. Thus, the distribution of class 1 is surely not normal anymore: the new mean of class 1 after noise addition is $\mu'_1 = 1.69$ and its standard deviation is $\sigma'_1 = 3.814$. The resulting classes are shown in figure 16.

Even with the presence of noise in the first dataset, it is obvious that class 1 and 2 are well separated, and could give very good classification performances in a machine learning context. However, the basic DP method will give a low score for the observed situation, because the overlapping zone will be totally defined by samples of class 1, encompassing class 2 distribution. This configuration is similar to the

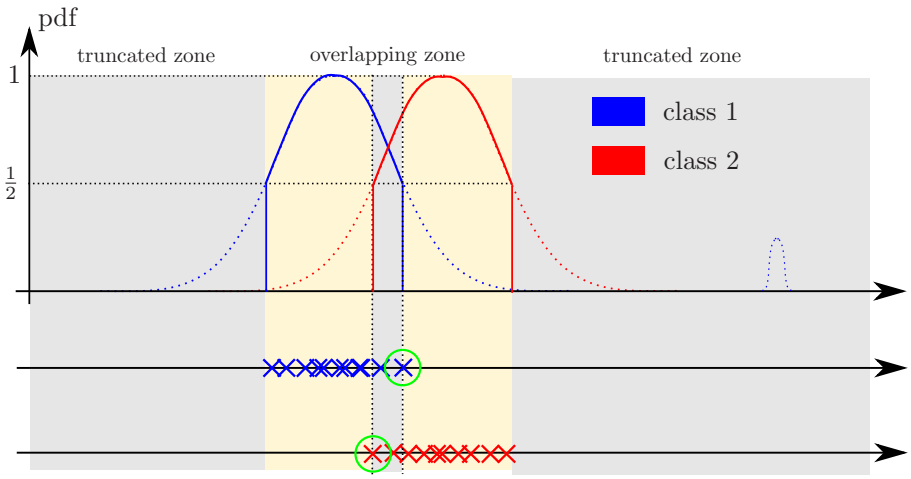

Figure 15: Schematic representation of modified DP algorithm.

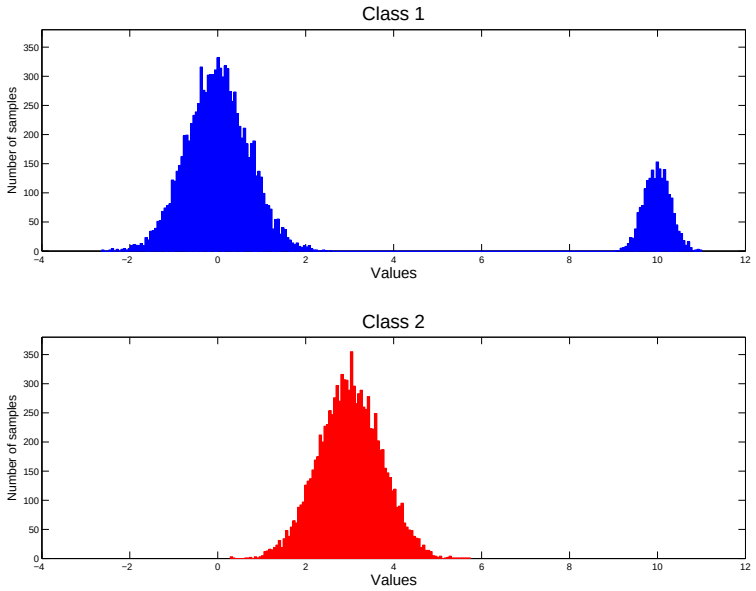

Figure 16: Comparison of DP algorithms: Initial distributions for class 1 and 2.

case presented in figure 13, and provides a score that does not illustrate the separability of the classes, since only one class contributes to the score. Table 3 shows th DP values of each method for the current problem.

On the contrary, the modified DP method can deal with this kind of situation, as shown in figure 17. By truncating both distributions and keeping only the interval $[\mu - 1.177 \cdot \sigma; \mu + 1.177 \cdot \sigma]$ of the FWHM value, the algorithm ignores all the noisy samples, and returns its score only based on relevant samples of both distributions. Even if the modified DP value is a little bit erroneous with respect to the actual situation because both distributions have been firstly truncated, this score is

Method	Truncated Interval	DP score (%)
Basic DP method	-	39.56%
Modified DP method	$[\mu - 1.177 \cdot \sigma; \mu + 1.177 \cdot \sigma]$	91.34%
Modified DP method	$[\mu - 2 \cdot \sigma; \mu + 2 \cdot \sigma]$	50.57%

Table 3: Scores of different DP methods.

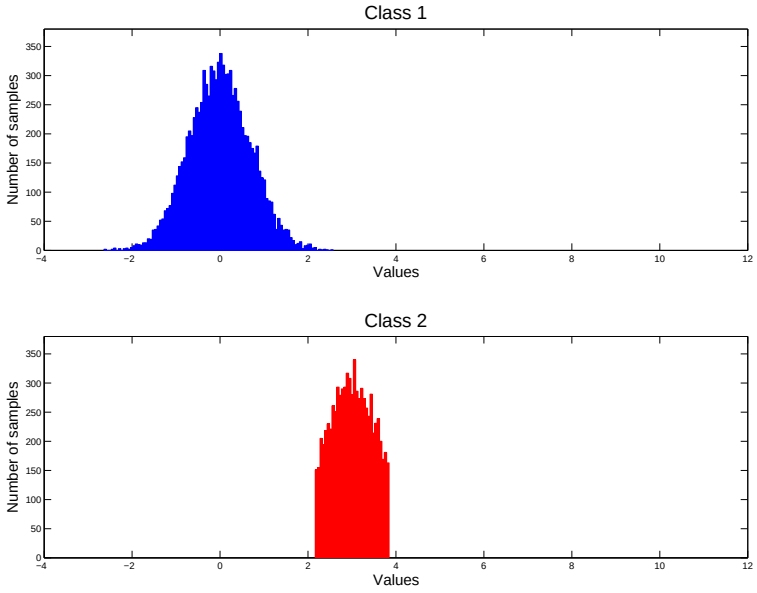

Figure 17: Comparison of DP algorithms: distributions for class 1 and 2 after modified DP method. Truncated interval: $[\mu - 1.177 \cdot \sigma; \mu + 1.177 \cdot \sigma]$.

now representative of the separability of the classes, since the boundaries of the overlapping zone are defined by both classes. Moreover, even if the red distribution of the second class looks a bit too much truncated in figure 17, we have to keep in mind that in real applications, and especially for applications dealing with bioelectrical signals, acquired data are always noisy, and follow normal law only approximately. Thus, after the preprocessing step of modified DP algorithm, truncated data will look more like the blue distribution of figure 17 than like the red one.
Finally, we show in figure 18 that if we choose the interval $[\mu - 2 \cdot \sigma; \mu + 2 \cdot \sigma]$ to truncate the samples, the DP score of the same example drops again to a low value of 50.57%, because some outliers samples, indicated by the black arrow, are kept despite the preprocessing step of our modified algorithm.

An important observation can be done at this point: the fact that, for this application, the interval $[\mu \pm 2\sigma]$ is too big and keeps noisy samples, can give us information about the "normality" of class 1. When we introduce $k = 2$ in Chebyshev's inequality, we obtain that 75% of the values should be within the interval if the distribution was not normal. Knowing that noisy data in class 1 represent 17% of the population, we can infer that in our example, the interval $[\mu \pm 2\sigma]$ contain at least 83% of the values. This means that class 1, as expected, still has some similarities with a purely normal distribution, which, by the way, would contain 95% of the population, as shown in table 2.
The conclusion is that taking FWHM value as truncating interval is a reasonable margin, if the distributions are assumed to be approximately normal, since a maximum of 76% of the values would

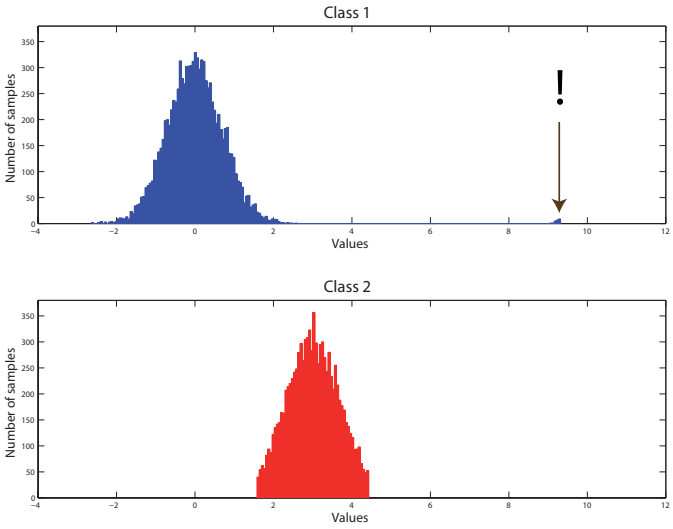

Figure 18: Comparison of DP algorithms: distributions for class 1 and 2 after modified DP method. Truncated interval: $[\mu - 2 \cdot \sigma; \mu + 2 \cdot \sigma]$.

be kept, in the case of a quasi-normal distribution; in this situation, noise removal will be done successfully in most cases, whereas bigger intervals can fail for some situations.

3.4 LDA-based Feature Selection Method

In this section, we propose a feature selection method based on a linear discriminant classifier. After a brief survey about linear discriminant analysis (LDA), the principle of this algorithm is described.

3.4.1 Linear Discriminant Analysis

Linear discriminant analysis (LDA) [2] is a supervised method for dimensionality reduction for classification problem. We present here the case where there are two classes; generalization to $K > 2$ classes is straightforward.
Given samples from two classes C_1 and C_2, we want to find the direction, as defined by a vector $\boldsymbol{w}$, such that when the data are projected onto $\boldsymbol{w}$, the examples from the two classes are as well separated as possible. the projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is

$$z = \boldsymbol{w}^T \boldsymbol{x} \tag{11}$$

Thus, if d is the number of dimensions of the input space, equation (11) is a dimensionality reduction from d to 1.
$\boldsymbol{\mu_1}$ and μ_1 are the means of samples from C_1 before and after projection, respectively. Note that $\boldsymbol{\mu_1} \in \Re^d$ and $\mu_1 \in \Re$. Given a sample $X = \{\boldsymbol{x}^t, r^t\}$ such that $r^t = 1$ if $\boldsymbol{x}^t \in C_1$ and $r^t = 0$ if $\boldsymbol{x}^t \in C_2$,

$$\mu_1 = \frac{\sum_t \boldsymbol{w}^T \boldsymbol{x}^t r^t}{\sum_t r^t} = \boldsymbol{w}^T \boldsymbol{\mu}_1 \tag{12}$$

$$\mu_2 = \frac{\sum_t \boldsymbol{w}^T \boldsymbol{x}^t (1 - r^t)}{\sum_t r^t} = \boldsymbol{w}^T \boldsymbol{\mu}_2 \tag{13}$$

The *scatter* of samples for C_1 and C_2 after projection are

$$\sigma_1^2 = \sum_t (\boldsymbol{w}^T \boldsymbol{x}^t - \mu_1)^2 r^t \tag{14}$$

$$\sigma_2^2 = \sum_t (\boldsymbol{w}^T \boldsymbol{x}^t - \mu_2)^2 (1 - r^t) \tag{15}$$

After projection, for the classes to be well separated, we would like the means to be as far as possible

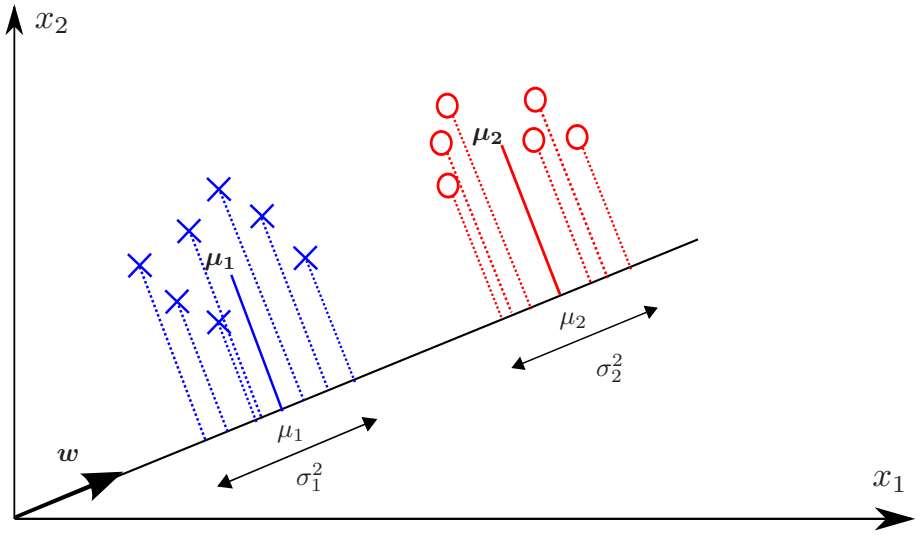

Figure 19: 2D two class data projected on $\boldsymbol{w}$.

and the examples of classes be scattered in a region that is as small as possible. So we want $|m_1 - m_2|$ to be large and $s_1^2 + s_2^2$ to be small (see figure 19). *Fisher's linear discriminant* is $\boldsymbol{w}$ that maximizes

$$J(\boldsymbol{w}) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \tag{16}$$

Rewriting the numerator, we get

$$\begin{aligned}(\mu_1 - \mu_2)^2 &= (\boldsymbol{w}^T \boldsymbol{\mu}_1 - \boldsymbol{w}^T \boldsymbol{\mu}_2)^2 \\ &= \boldsymbol{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{w} \\ &= \boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}\end{aligned} \tag{17}$$

where

$$\boldsymbol{S}_B = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \tag{18}$$

is the *between-class scatter matrix*. The denominator is the sum of the scatter of examples around their means after projection and can be rewritten as

$$\begin{aligned}\sigma_1^2 &= \sum_t (\boldsymbol{w}^T \boldsymbol{x}^t - \mu_1)^2 r^t \\ &= \sum_t \boldsymbol{w}^T (\boldsymbol{x}^t - \boldsymbol{\mu}_1)(\boldsymbol{x}^t - \boldsymbol{\mu}_1)^T \boldsymbol{w} r^t \\ &= \boldsymbol{w}^T \boldsymbol{S}_1 \boldsymbol{w}\end{aligned} \tag{19}$$

where

$$\boldsymbol{S}_1 = \sum_t r^t (\boldsymbol{x}^t - \boldsymbol{\mu}_1)(\boldsymbol{x}^t - \boldsymbol{\mu}_1)^T \tag{20}$$

is the *within-class scatter matrix* for C_1. $\frac{\boldsymbol{S}_1}{\sum_t r^t}$ is the estimator of the covariance matrix Σ_1. Similarly, $\sigma_2^2 = \boldsymbol{w}^T \boldsymbol{S}_2 \boldsymbol{w}$ with $\boldsymbol{S}_2 = \sum_t (1 - r^t)(\boldsymbol{x}^t - \boldsymbol{\mu}_2)(\boldsymbol{x}^t - \boldsymbol{\mu}_2)^T$, and we get

$$\sigma_1^2 + \sigma_2^2 = \boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w} \tag{21}$$

where $\boldsymbol{S}_W = \boldsymbol{S}_1 + \boldsymbol{S}_2$ is the total within-class scatter matrix. Note that $\sigma_1^2 + \sigma_2^2$ divided by the total number of samples is the variance of the pooled data. Equation (16) can be rewritten as

$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w}} = \frac{|\boldsymbol{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)|^2}{\boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w}} \tag{22}$$

Taking the derivative of J with respect to $\boldsymbol{w}$ and setting it equal to 0, we get

$$\frac{\boldsymbol{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}{\boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w}} \cdot \left(2(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) - \frac{\boldsymbol{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}{\boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w}} \boldsymbol{S}_W \boldsymbol{w} \right) = 0 \tag{23}$$

Given that $\frac{\boldsymbol{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)}{\boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w}}$ is a constant, we have

$$\boldsymbol{w} = c \boldsymbol{S}_W^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \tag{24}$$

where c is some constant. Since we are more interested in the direction than in the magnitude, we can just take $c = 1$ and find $\boldsymbol{w}$. To summarize, we have projected the samples from d dimensions to one, and any classification method can be used afterward. Fisher's linear discriminant is optimal when the classes are normally distributed, but it can be used even when the classes are not normal.

3.4.2 Proposed Algorithm

We developed a feature selection based on LDA classification, in order to establish a ranking of the features in a given data set. A very basic LDA classifier was implemented on Matlab; the advantage is that such classifiers are quick, and can be used in an iterative procedure. The Matlab function takes two labeled matrices as inputs: the first matrix is the training set used to tune the classifier, and the second is a test set allowing to return a score for the performance of the classifier. The process is the following, for the case of the selection of N_{sel} features among N_f initial features:

- Each feature is considered independently, and N_f matrices are built with all the labeled trials of the data set for each feature.
- A LDA classification procedure is applied on each of these matrices, giving a list of N_f scores. The ranking of these scores gives the ranking of the corresponding features.
- The N_{sel} best features are kept from this list.

LDA classification is a relatively quick process, and that is the reason why we chose an LDA classifier in our method. It is clear that we could have chosen any other classifier, such as a gaussian classifier or an artificial neural networks, in order to insert it in our algorithm. But it is important to understand that in this feature selection procedure, the absolute scores of classification are not really important in order to create the ranking of the features: the interesting point is the relative scores between features. Thus, the choice of the classifier is not crucial, and we decided to take the fastest one.

Part II
Experimental Results: Error-Related Potentials Study

4 EEG Acquisition System

4.1 Experimental Setup

The EEG measurement system of IDIAP BCI group comes from Biosemi Instrumentation Company, Amsterdam, Netherland [9]. The model is an Active Two System, with 64 EEG electrodes following the 10-20 international electrode layout (see figure 20). Moreover, 8 supplementary electrode channels can be used for complementary measurement, such as EMG. The digital resolution is very good (31.25nV) and the input range is 524mVpp.

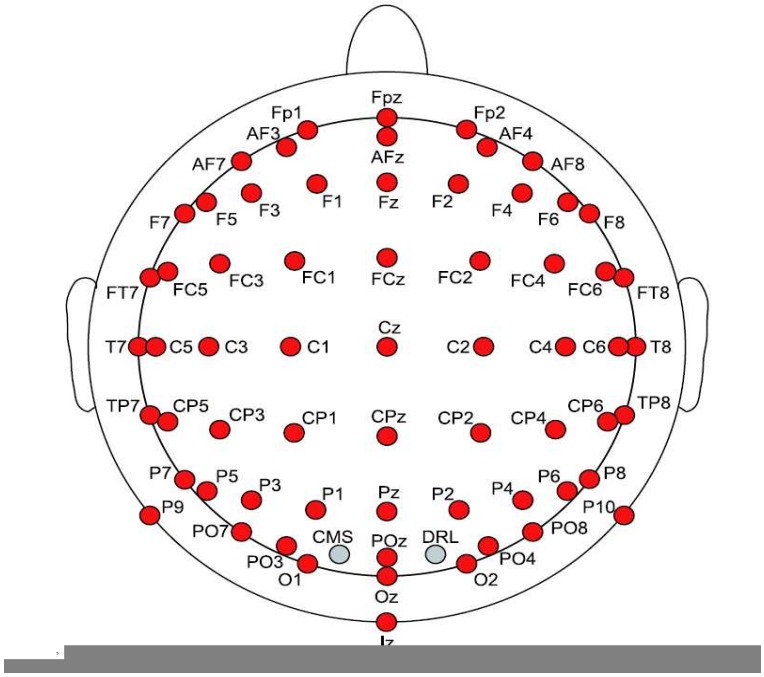

Figure 20: 64 EEG electrodes following the 10-20 international electrode layout.

In order to efficiently connect the system to the ground, the Biosemi system uses two electrodes: a Common Mode Sense (CMS) electrode and a Driven Right Leg electrode (DRL). The CMS electrode measures the potential of the patient, while the DRL electrode closes the loop between the patient and the A/D converter. The DRL electrode is directly connected to a Driven Right Leg circuit [29], in order to reduce the common-mode voltage and protect the patient by limiting the output current.

The Active Two system can assume any electrode or combination of electrodes to be the reference of measurement, providing a sufficient Common Mode Rejection Ratio of 80dB.

4.2 Data Preprocessing

EEG signals were acquired continuously during relatively long sessions; the trials had to be segmented afterward. The sampling frequency was 512Hz for the specific applications of this work. The preprocessing made on raw EEG data was applied systematically. These preprocessing steps are the following:

[9] http://www.biosemi.com/

- EEG signals are first saved in 24 bits binarized integers BDF format by the LabVIEW[10] software provided by Biosemi. The first operation is to convert these data in ASCII files with physiological amplitudes, i.e. μV.
- We remove the mean activity of each electrode independently. This DC Removal procedure is crucial in order to set all electrodes to the same order of amplitude, and thus avoid biases among electrodes.
- If needed, filters can be applied on the signals, in order to keep a specific band frequency.
- We consider the mean of all the connected electrodes as the reference, in order to remove the background EEG activity. This reference is called Common Average Reference (CAR), and is simply the mean activity of all electrodes; we remove it at each time sample.
- If needed, a specific matrix applying a linear inverse solution and transforming scalp EEG data into estimated intracranial activity is applied at this step.

[10] http://www.ni.com/labview/

5 Error-Related Potentials

5.1 State of the Art

In contrast to other interaction modalities, a unique feature of the "brain channel" is that it conveys both information from which we can derive mental control commands to operate a brain-actuated device as well as information about cognitive states that are crucial for a purposeful interaction, all this on the millisecond range. One of these states is the awareness of erroneous responses, which a number of groups have recently started to explore as a way to improve the performance of BCIs: see for instance [42], [7], [35], [16] and [15].
Since the late 1980s, different physiological studies have shown the presence of error-related potentials (ErrP) in the EEG recorded right after people get aware they have made an error ([8],[14], [25]). Apart from Schalk et al. (2000) who investigated ErrP in real BCI feedback, most of these studies show the presence of ErrP in typical choice reaction tasks ([7],[35],[8],[14]). In this kind of tasks, the subject is asked to respond as quickly as possible to a stimulus and ErrP (sometimes referred to as "response ErrP") arise following errors due to the subject's incorrect motor action. The main components here are a negative potential showing up 80 ms after the incorrect response followed by a larger positive peak showing up between 200 and 500 ms after the incorrect response. More recently, other studies have also shown the presence of ErrP in typical reinforcement learning tasks where the subject is asked to make a choice and ErrP (sometimes referred to as "feedback ErrP") arise following the presentation of a stimulus that indicates incorrect performance [25]. The main component here is a negative deflection observed 250 ms after presentation of the feedback indicating incorrect performance. Finally, other studies reported the presence of ErrP (that we will refer to as "observation ErrP") following observation of errors made by an operator during choice reaction tasks [47] where the operator needs to respond to stimuli. As in the feedback ErrP, the main component here is a negative potential showing up 250 ms after the incorrect response of the operator performing the task. ErrP are most probably generated in a brain area called anterior cingulate cortex (ACC), which is crucial for regulating emotional responses [25].
An important aspect of the first two described ErrP is that they always follow an error made by the subject himself. First, the subject makes a selection, and then ErrP arise either simply after the occurrence of an error (choice reaction task) or after a feedback indicating the error (reinforcement learning task). However, in the context of a BCI or human-computer interaction in general, the central question is to know if ErrP are also elicited when the error is made *by the interface* during the recognition of the subject's intent. Investigations have been made at IDIAP about this precise issue.

5.2 IDIAP Research: Previous Study

Very recently, Ferrez and Millán investigated in [15] how ErrP could be used to improve the performance of a BCI. Especially, if ErrP are also elicited when the error is made by the interface, then it could be integrated in a BCI in the following way as shown in figure 21: after translating the subject's intention into a control command, the BCI provides a feedback of that command, which will be actually executed only if no ErrP follows the feedback. This should greatly increase the reliability of the BCI system, as shown in the paper.

5.2.1 Experimental Setup

To test the presence of ErrP after a feedback indicating errors made by the interface in the recognition of the subject's intent, a human-robot interaction task where the subject has to bring the robot to targets 2 or 3 steps either to the left or to the right was simulated. To isolate the issue of the recognition of ErrP out of the more difficult and general problem of a whole BCI where erroneous feedback can be due to non-optimal performance of both the interface (i.e., the classifier embedded into the interface) and the user himself, in the following experiments the subject delivers commands manually and not mentally. Five volunteer healthy subjects participated in these experiments. The

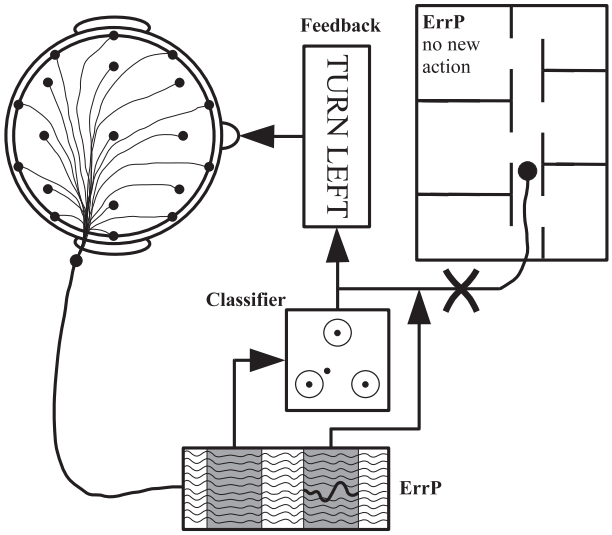

Figure 21: Exploiting error-related potentials (ErrP) in a brain-controlled mobile robot. The subject receives visual feedback indicating the output of the classifier before the actual execution of the associated command (e.g., "TURN LEFT"). If the feedback generates an ErrP (left), this command is simply ignored and the robot will stay executing the previous command. Otherwise, the command is sent to the robot (right).

system moved the cursor with an error rate of 20%; i.e., at each step, there was a 20% probability that the cursor moved in the opposite direction. Subjects performed 10 sessions of 3 minutes on 2 different days, corresponding to ∼75 single trials per session; it means that in each sessions, about 60 correct trials and 15 error trials were recorded. The delay between the two days of measurements was about 3 months. The sampling rate was 512 Hz and signals were measured at full DC. Raw EEG potentials were first spatially filtered by subtracting from each electrode the common average reference at each time step. Then, a 1-10 Hz bandpass filter was applied, as ErrP are known to be a relatively slow cortical potential.
Only interesting part of the recorded signal was kept as follows: half-second windows starting 150 ms after the feedback and ending 650 ms after the feedback were extracted. EEG signals were then subsampled from 512 Hz to 64 Hz (i.e., one point out of 8 was taken) before classification, which was entirely based on temporal features. The two different classes are recognized by a Gaussian classifier.

5.2.2 Results

Figure 22 shows the difference error-minus-correct for channel FCz for the five subjects plus the grand average of the five subjects for the two days of recordings. A first sharp negative peak (Ne) can be clearly seen 250 ms after the feedback. A later positive peak (Pe) appears between 320 ms after the feedback. Finally a second negative peak occurs about 450 ms after the feedback. Figure 3 also shows the scalp potentials topographies, for the grand average EEG of the five subjects, at the occurrence of the maximum of the Ne, the Pe and the additional negative peak: a first fronto-central negativity appears after 250 ms, followed by a fronto-central positivity at 320 ms and followed by a fronto-central negativity at 450 ms. Moreover, the feasibility of detecting single-trial erroneous responses was explored, by means of a 10-fold cross-validation study where the testing set consists of one of the recorded sessions. In this way, testing is always done on a different recording session to those used for training the model. To summarize, the existence of a new kind of error-related potentials, called "interaction ErrP", was confirmed; the feasibility of detecting single-trial erroneous responses is very

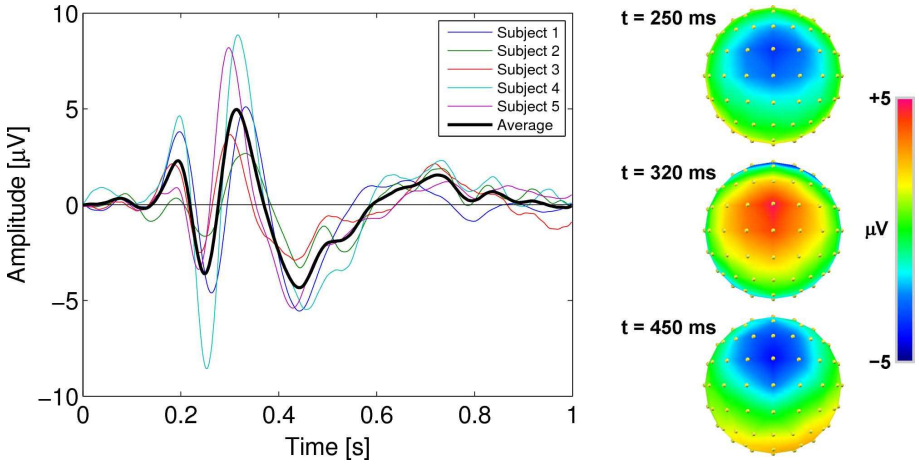

Figure 22: *Left.* Average EEG for the difference error-minus-correct at channel FCz for the five subjects plus the grand average of them for the first day. *Right.* Scalp potential topographies, for the grand average EEG of the five subjects, at the occurrence of the peaks. Small filled circles indicate positions of the electrodes (frontal on top), Cz being in the middle.

important, since it could improve the performance and reliability of a BCI system.

5.3 Objectives: Extending the Study

From that point, our goal in next three sections is two-fold. On one hand, comparisons of all the methods and models presented in the previous sections are done in the framework of error-related potentials study; advantages and drawbacks of each method are pointed out, and inverse solutions are compared based on different quality criteria.
But most importantly, all these comparisons are done with the same underlying objective: providing an extension to the article presented in section 5.2. Indeed, we aim at extending the study to the use of inverse solutions, as well as new feature selection methods. In order to do it, we took the same data sets with the five subjects, and reproduced the experiments reported in [15] with only slight differences allowing us to do a consistent comparison between EEG and inverse solutions. The process was the following:

- The same preprocessing as described in section 5.2.1 was applied on data before classification.
- A 10-fold cross-validation was done on the first day of recording for the five subjects, and for both EEG signals and intracranial signals obtained by applying the CCD inverse model. In each fold, a feature selection was done with our LDA-based method: a given number of features was kept for the classification procedure. For the inverse model, we repeated the whole process varying the number of kept features, in order to look for an optimal number of features.
- Instead of keeping always the same two electrodes (FCz and Cz) for all the folds as in the previous study, we decided to take the two best electrodes after feature selection independently for each fold. Thus, a direct comparison could be made with CCD inverse model, since we did exactly the same with the vertices of the model.
- Instead of a gaussian classifier, we used a simple LDA classifier for the classification procedure of each fold. We justify this choice by noting that this classifier does not need any tuning, as the statistical gaussian classifier does. Thus, we could repeat our experiences easily and vary the number of selected features for the inverse solutions. Moreover, we are more interested in

the comparison between EEG and inverse models, or the comparison between feature selection methods, than in the absolute scores of classification. Even if the scores given by our LDA classifier are expected to be lower than in [15], we are interested in relative differences between scores.

- For one of the subject (subject 4), we extended our experiments to other feature selection methods and inverse solutions, in order to allow comparisons. Thus, the same cross-validation process was applied, taking successively modified DP function and ReliefF algorithm for the feature selection step. In order to compare inverse models, 10-fold cross-validation was also applied on ELECTRA-LAURA inverse solution.

- Finally, for both EEG and CCD inverse solution, we classified each of the 10 sessions of the second day of recording with selected features based on all the sessions of the first day. By doing this generalization over extended periods of time, we could have a clear idea of the properties of stability of EEG and CCD inverse model over time.

6 Comparing Feature Selection Methods

Error-related potentials are very convenient for feature selection method testing, since we know approximately which part of the brain should be more involved in the process of error detection. Indeed, several articles, including studies in which EEG and functional MRI were measured simultaneously [12], have shown strong relationship between the error negativity (ERN) and error-related anterior cingulate cortex (ACC) activation[11]. Thus, we can reasonably expect that our methods select channels that are located near fronto-central areas. In this chapter, the feature selection methods presented in section 3 are validated on EEG data, and then applied on the CCD inverse model in the framework of a 10-fold cross-validation on one of the subjects.

6.1 Method of Comparison

The ranking of the channels, i.e. the electrodes or the cortical vertices, was established for each fold of the cross-validation as follows:

- For our LDA-based feature selection method, we built for each of the N_f channels a LDA classifier and measured its classification score on the training set of the current fold. The training set is a matrix containing only the activity of the corresponding channels and of dimension $N_{trials} \times 32$ (because we subsampled at $64Hz$ and took 0.5 time windows). We take the classification score as the quality estimation of the channel. At the end, a list of N_f scores gave the ranking of the channels for a given fold.
- For ReliefF algorithm and the modified DP function, the estimation of the quality of a given channel was done in two steps. First, the methods were applied on each of the N_f matrices described above, in order to get an estimation of the discriminant power of each of the 32 time samples. Then, these 32 scores were simply sumed to have a unique score for a given channel. This quality estimation is meaningful, because if the 32 time samples are more discriminative in average for a given channel than for the other features, the resulting sum will be bigger, giving a good score to the channel.

6.2 Selection of Relevant Scalp Channels

As a validation of our feature selection methods, we first applied them on EEG signals, in order to select relevant electrodes that could better discriminate between correct trials and error trials. We analyzed the data set made of all the trials of the first day of recording of subject 4. The number of iterations of ReliefF algorithm was set equal to the number of correct trials of the data set. The results are shown in table 4.

Rank	1	2	3	4	5	6	7	8	9	10
LDA-based method	Fz	FC1	FCz	FC2	F1	F2	P8	P6	FC3	C1
Modified DP	FCz	FC1	Fz	FC2	F2	F1	Cz	C1	P8	PO7
ReliefF	FCz	FC1	FC2	Fz	F2	Cz	C1	F1	P8	PO7

Table 4: Rankings of electrodes for subject 4, day I, for the three feature selection methods.

As expected, the best electrodes are all located in fronto-central areas (see figure 20 for EEG cap layout). Moreover, the reliability of our methods is demonstrated, since the four best electrodes are the same for all methods. Beside the strong activation of fronto-central areas, we note that electrode P8 is part of the ten best electrodes for the three methods: activation of parietal areas has already been reported by other studies [48]. A possible interpretation could be that this parietal activation is related to associative areas activated by the occurrence of an error.

[11] see [46] for a detailed atlas of the brain.

6.3 Selection of Relevant Cortical Areas

The second step was to compare the methods by using the CCD inverse solution. Here, we assessed the quality of each feature selection both in terms of classification accuracy and localization of the features, by means of a 10-fold cross-validation on the first day of recording of subject 4.

6.3.1 Cross-Validation

The results of the cross-validation procedure are shown in figure 23 for each feature selection method, and for different numbers of selected features. We can do the following observations:

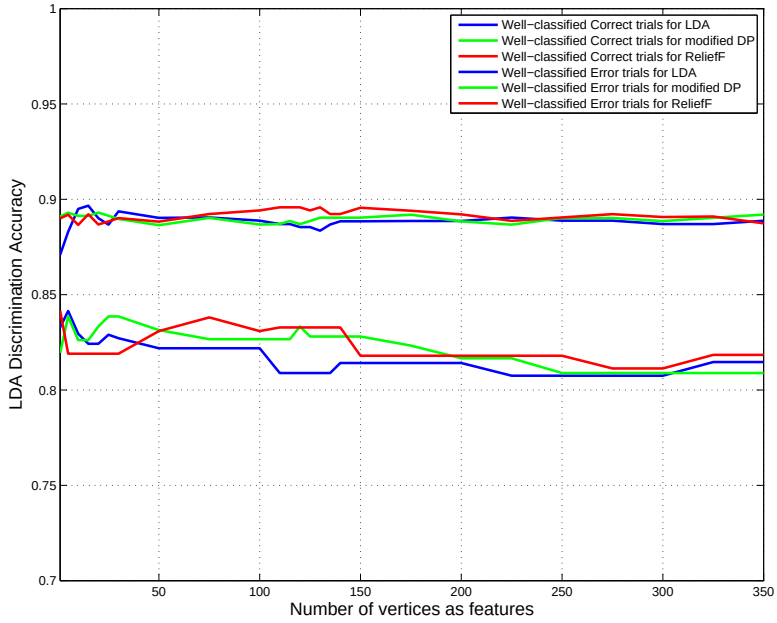

Figure 23: Results of 10-fold cross-validation for different feature selection methods, and different numbers of selected features.

- All three methods provide roughly the same results in terms of classification accuracy. We could have expected the LDA-based method to be the best of the three methods, since the criteria of selection of features for this method is based on the LDA classifier applied in the classification process of the cross-validation. But we see that ReliefF and the modified DP function are as efficient as the LDA-based method. An explanation could be that with ReliefF and the modified DP function, we sum the discriminative powers of the 32 time samples. Thus, this way of computing the scores of the channels gives a different, and maybe better idea of the global discriminative behaviour of a given channel over time.
- Interestingly, we note that the best classification accuracies are often obtained with a small number of selected channels. Indeed, the best accuracy for the classification of correct trials (89.66%) is obtained by the LDA-based method for 15 selected vertices, whereas for the classification of error trials, the LDA-based method reaches 84.14% for 5 selected vertices. For ReliefF algorithm, the optimal number of selected features is a little bit bigger (between 50 and 120 vertices), but still acceptable. These observations are of the highest importance, since the

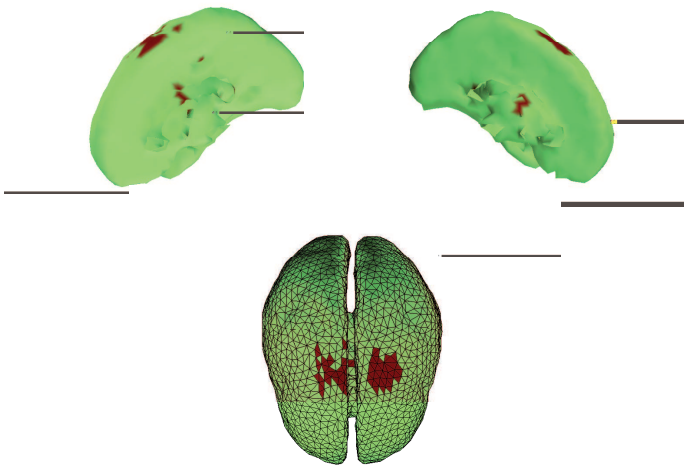

Figure 24: Localization of features made by the LDA-based method on fold 1 of the cross-validation procedure.

dimension of the input space would not explode with only 15 to 100 selected features, allowing online implementation.

- Comparing the methods in terms of computational speed, it appears that ReliefF algorithm is by far the worst of the methods. Indeed, ReliefF is an iterative methods, and a certain number of iterations must be done in order to converge to a good estimation of the quality of each feature. In these experiments, the number of iterations was set equal to the number of trials of one class, which is an acceptable value (see section 3.2.3). But these iterations have to be done on *each* feature of the initial input space, which means a lot of times when we deal with inverse solutions (for instance, 3013 times for CCD inverse model). Thus, the feature selection made by ReliefF was a very long process; online implementations of such methods would be impossible. As opposed to ReliefF algorithm, the LDA-based method and the modified DP function were much faster; the modified DP function was the fastest method, selecting features almost instantaneously.

6.3.2 Localization of Features

In order to assess the quality of the feature selection made by our methods, we observed the locations of the selected features in randomly selected folds of the cross-validation. We chose the 1^{st}, 5^{th} and 9^{th} folds, and looked at the 50 best features in each folds, for each method. Results are shown in figures 24 to 28. Once again, our methods provide features with physiological meaning, confirming the first investigations with EEG signals. Here, with the help of inverse solutions, we can even go further in the details of the localization, and look at the Brodmann areas[12] (BA) involved in the process of error discrimination. In figures 24, 25 and 26, we see that our LDA-based method selected features in two relevant areas of the brain: first, a focus is found in fronto-central area, at the surface of the cortex. This area is the 6^{th} Brodmann area, which encompasses pre-motor and supplementary motor cortex (pre-SMA). The second cluster of selected features is focused on the rostral and caudal cingulate zone (BA 24 and 32). In terms of neurophysiology, both selected areas are well-known to be involved in

[12]For an online atlas: http://spot.colorado.edu/∼dubin/talks/brodmann/brodmann.html

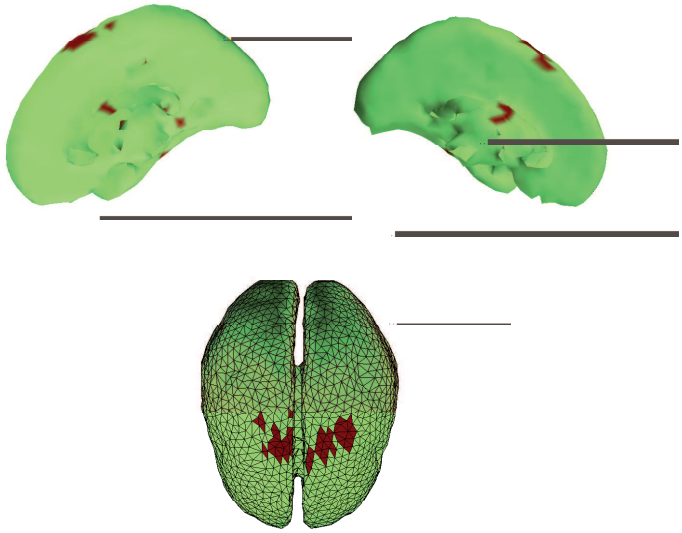

Figure 25: Localization of features made by the LDA-based method on fold 5 of the cross-validation procedure.

error detection and correction processes [17]. Since similar foci (BA 24 and 32) were found for the modified DP method and for the ReliefF algorithm, we only displayed top views of the localizations of features for these methods. Thus, it proves that our methods are able to select relevant features in terms of physiology.

Moreover, we tried to assess the stability of the selections by comparing, for each method independently, the selections made in the different folds of the cross-validation. First, we took each possible pair of folds (45) and calculated the percentage of identical selected vertices within the 20 best features; we restricted the lists to the 20 best features, since we saw that a small number of them already provides good classification accuracy. Then, we averaged these 45 percentages to have a global idea of the stability of the feature selection between two different folds. The second step was to look for the relevant vertices that were selected in *all* folds; once again, we considered the 20 best features, and looked for those that were present in all of the 10 folds. Results are shown in table 5.

Method	Averaged % of identical selected vertices over all possible pairs of folds	# of identical selected vertices in all folds
LDA-based	45.89%	1
Modified DP	91.67%	14
ReliefF	59.89%	3

Table 5: Stability of feature selections depending on the method. The 20 best selected features are considered.

We see that the modified DP method selects its features with an impressive regularity. Besides, it is possible to confirm that statement simply by looking at figure 27. Indeed, the clusters located on pre-SMA are very similar between the folds; for instance, 98% of the selected vertices (49 over 50)

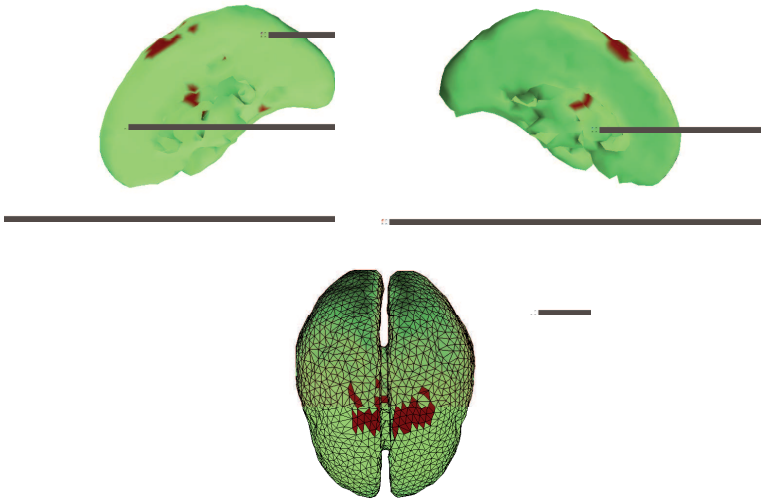

Figure 26: Localization of features made by the LDA-based method on fold 9 of the cross-validation procedure.

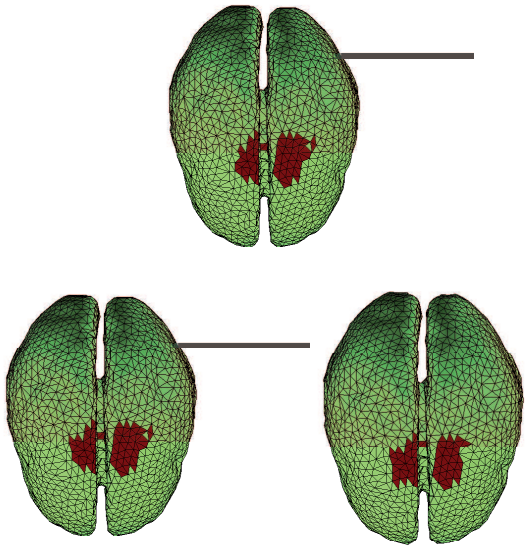

Figure 27: Localization of features made by modified DP method on fold 1 (top), 5 (bottom left) and 9 (bottom right) of the cross-validation procedure.

are similar between fold 0 and 4! The LDA-based method performs less stable selections than the other methods, but this results must be considered with care. Indeed, even if the patterns of selected vertices were slightly different from one fold to the other, the global clusters were always located on

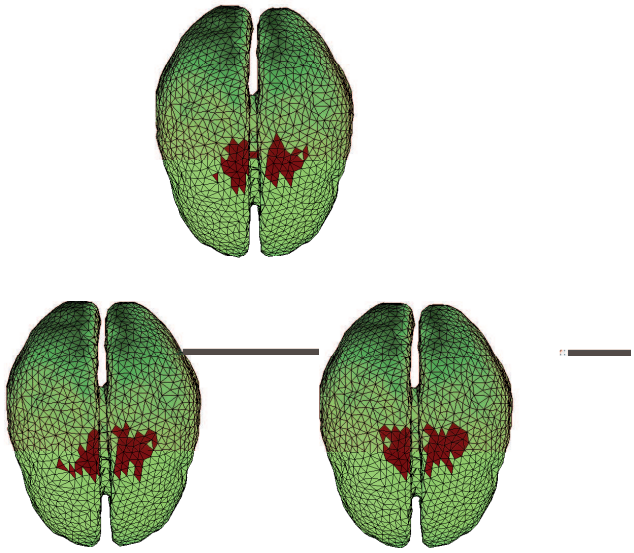

Figure 28: Localization of features made by ReliefF method on fold 1 (top), 5 (bottom left) and 9 (bottom right) of the cross-validation procedure.

BA6 and BA24-32, as shown in the related figures. Thus, we believe that the property of stability of a feature selection cannot be completely defined only by looking at the number of constantly selected features, as table 5 does; we should rather look at the stability of the global shape of the clusters of features over time. The proof is that the LDA-based method was the best one in terms of classification accuracy after cross-validation (see section 6.3.1); our interpretation is that the algorithm chose for each fold the best combination of vertices *within the same physiologically relevant area*. These selected vertices may be different from one fold to the other, but the combination is always the most efficient in terms of discrimination. Thus, table 5 tells us about the strict stability of the feature selections, but we should not do any conclusions based on these values about the global efficiency of the methods in the context of a BCI system, since our ultimate goal is to reach the highest classification accuracy. More investigations about stability of the selected features are done in next sections, and validate our statements.

7 Comparing Inverse Solutions

We provide in this chapter a comparison of the inverse solutions presented in section 2.3. This comparison will be based on different criteria. For one of the models, namely the sLORETA inverse solution, we did not have access to the transformation matrix; thus, the software was used as a "black box", and only localization studies are reported here, in order to give an idea about the results provided by this inverse model. For the LAURA-ELECTRA and CCD inverse solutions, a more complete comparison was made, following the process of chapter 6.

7.1 sLORETA : Localization of Relevant Cortical Areas

sLORETA inverse solution is known to be the only localization tool with zero localization error [36]. The software is available on sLORETA website[13] as a "blackbox" software: the user doesn't really know about the details of the localization process and the creation of the transformation matrix, except the theoretical elements of chapter 2.3.2. Even if the software is very easy to install and to use, it is impossible to apply our machine learning methods without the transformation matrix. Thus, we restricted our study of sLORETA model to a localization study aiming at showing the advantages and drawbacks of this inverse method. The process was the following:

- We computed the matrix of the averaged EEG activity of subject 4 over day I. The dimensions of the matrix are 64×512, since we used 64 electrodes and considered a time window of 1 second right after the occurrence of an error, with a sampling frequency of 512Hz. For instance, the averaged waveform of error trials of electrode FCz is shown in figure 29.
- We loaded the matrix in sLORETA software as well as a map of the 64 used electrodes and the sampling frequency. From that point, everything was generated internally, so that the localization could be immediately visualized in the main display window of the software.

The advantage of sLORETA is that the localization can be observed over time. Indeed, a very efficient and practical visualization tool is provided in the software, allowing to see the *evolution* of neuronal patterns of a given process over time. Thus, sLORETA visualization tool is useful to understand the underlying psychophysiological processes of error detection.
To illustrate that, we analyzed cerebral activations at four well chosen instants of the error process; these instants are marked with red circles in figure 29.

Figure 30 to 33 show 3D cortical views as well as Talairach slices of localized activity related to each of the red circles of figure 29.

We make the following observations:

- As expected, typical areas involved in error potentials such as pre-SMA (BA6) or ACC (BA24-32) are activated. Interestingly, we note that activation of areas BA6 and BA24-32 is stronger at instants corresponding to the peaks of the averaged error potential, namely the negative peak (230ms after error occurrence) and the subsequent positive peak (300ms after error occurrence); between the peaks, activation patterns of lateral prefrontal cortex (PFC) and orbito-frontal cortex similar to those of figure 30 are observed (not shown).
- Such activation of PFC in the context of error detection has already been extensively commented in the literature. Particularly, Gehring and Knight [20] reported an interaction of the lateral prefrontal cortex with the anterior cingulate cortex in monitoring behavior and in guiding compensatory system: PFC could be related to monitoring processes, whereas ACC and pre-SMA are more involved in error detection. This hypothesis seems to be in agreement with the patterns of activation we observed on our BCI signals.

[13] http://www.unizh.ch/keyinst/NewLORETA/LORETA01.htm

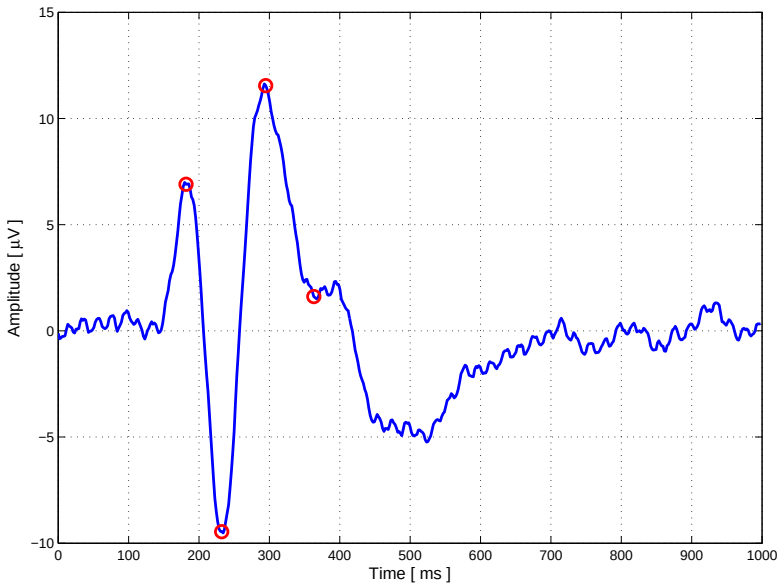

Figure 29: Mean activity of electrode FCz.

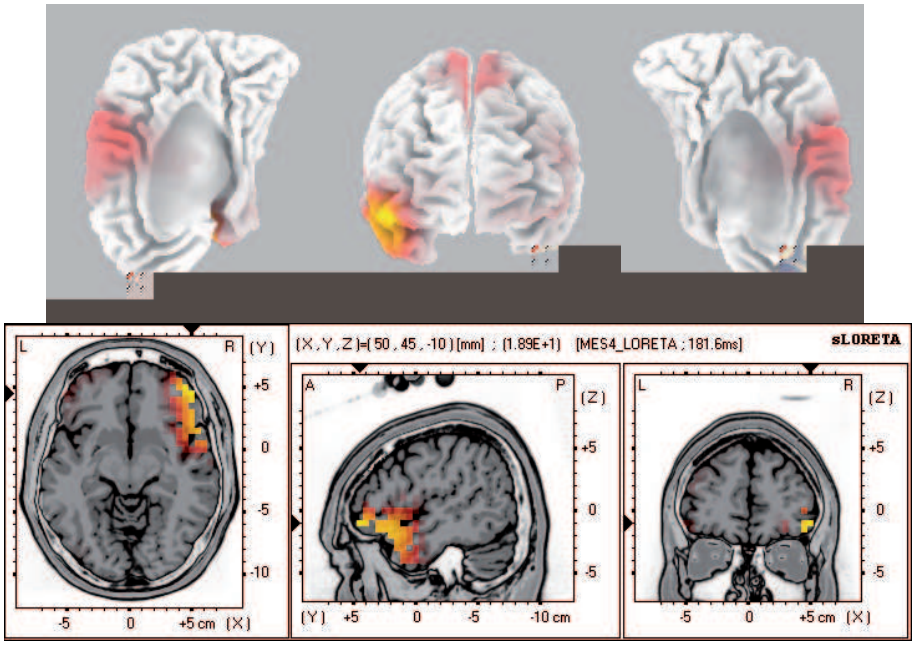

Figure 30: sLORETA localized activity 181.6ms after error occurrence.

- Finally, 50ms after the second positive peak, activations of parietal areas are observed: these associative areas could be related to the fact that the subject becomes aware of the error. Indeed, it has been proposed that the positive peak of an error potential was associated with conscious

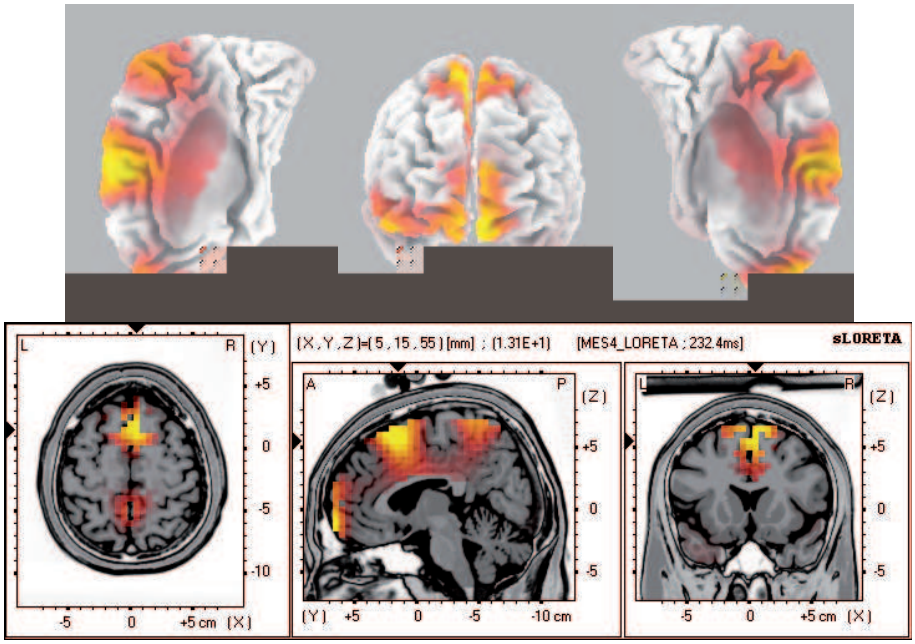

Figure 31: sLORETA localized activity 232.4ms after error occurrence.

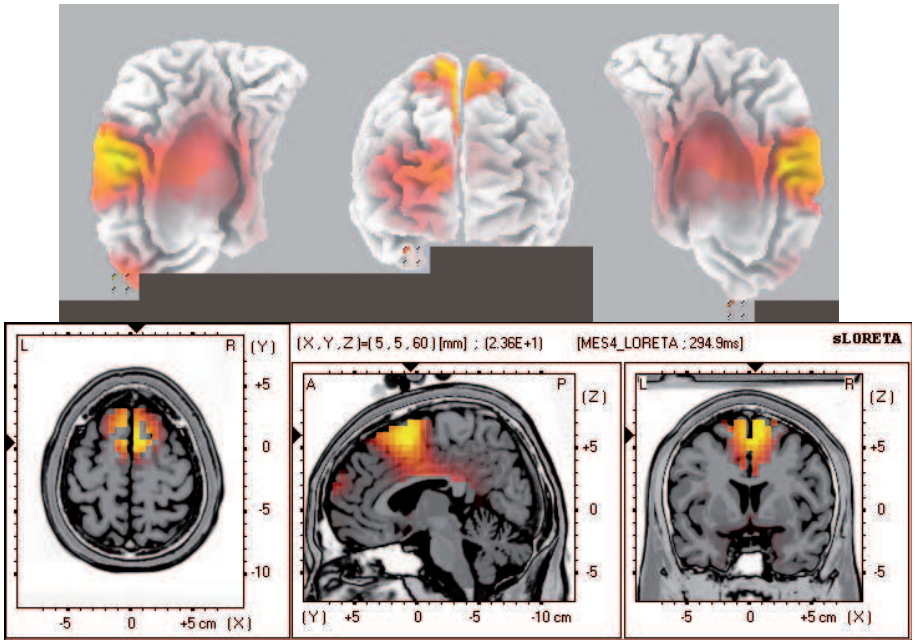

Figure 32: sLORETA localized activity 294.9ms after error occurrence.

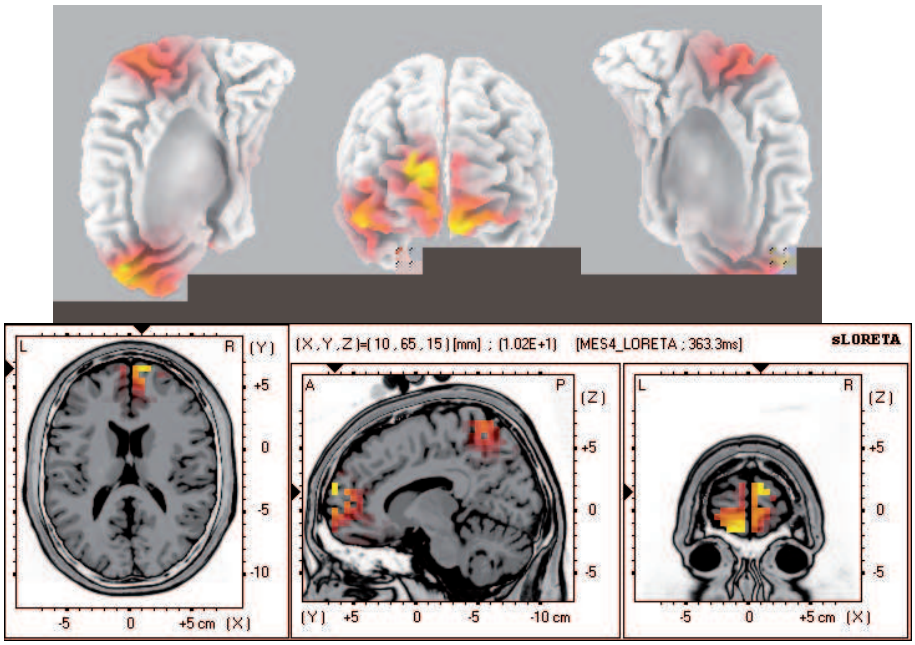

Figure 33: sLORETA localized activity 363.2ms after error occurrence.

error recognition [34]; in our case, the activation of associative areas only 50ms after the positive peak agrees with that hypothesis.

By means of sLORETA visualization software, a rough idea of the underlying psychophysiological process of error monitoring related to BCI can be described. However, a major drawback of sLORETA has to be emphasized: the foci of activation are very blurred, with respect to the clusters of relevant vertices provided by the CCD inverse model in figures 24, 25 and 26, for instance. The reason of that oversmoothing is the use of the Laplacian Weighted Minimum Norm constraint (see section 2.3.2). Thus, the lack of precision in the definition of the foci of activation could be a limitation for the use of sLORETA as classification tool in the context of BCI.

7.2 LAURA-ELECTRA and CCD Inverse Models

For ELECTRA-LAURA and CCD inverse solutions, a more complete comparison is allowed, since the transformation matrices are provided. Thus, we reproduce the same comparison process as in chapter 6; the classification accuracy is first compared for both methods after 10-fold cross-validation on the first day of recording of subject 4. Then, the localization of the relevant selected features is analyzed. As a feature selection method, we chose the LDA-based method for both models, since it was the method providing the best classification performances.

7.2.1 Cross-Validation

The results of the 10-fold cross-validation procedure are shown in figure 34, in which we compare the CCD inverse model and the ELECTRA-LAURA inverse solution for different numbers of selected channels.

Observing this figure, we can do the following remarks:

- Both inverse solutions provide their best performances with a small number of selected channels. Indeed, the CCD inverse model reaches the best classification of correct trials with $N_{channels} =$

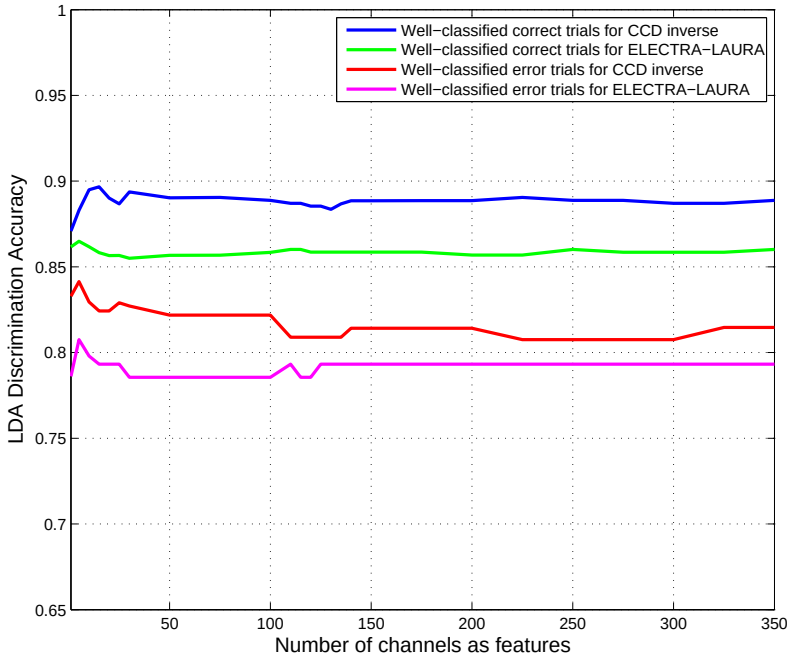

Figure 34: Results of 10-fold cross-validation of subject 4, day I, for ELECTRA-LAURA and CCD inverse model.

15, and the best classification of erroneous trials with $N_{channels} = 5$, whereas ELECTRA-LAURA model uses $N_{channels} = 5$ to get the best performances in the classification of both correct and erroneous trials. We see that in these cases, the dimensionality of the input space is relatively small, and feature selection algorithms, for instance, can be quickly applied on it. This observation confirms our belief that inverse solutions could be of high interest in the context of BCI research, since it is possible to take advantage of the spatial resolution of such methods without being limited by computational resources.

- It seems that the CCD inverse model always provides better performances in term of classification accuracy, independently of the number of selected channels. In order to confirm our observation, we performed a Wilcoxon rank sum test on our samples following this procedure: first, we considered for each model, the best cross-validation results (see previous remark for the respective optimal numbers of selected channels for each model). Then, we calculated for each inverse solution what we call the *accuracy value* of each fold of the cross-validation, defined by:

$$\text{accuracy} = \frac{C_{cc} + E_{cc}}{C_{tot} + E_{tot}} \tag{25}$$

where C_{cc} and E_{cc} are the number of correctly classified correct trials and error trials, respectively, and C_{tot} and E_{tot} are the total number of correct and error trials. This accuracy value takes into account that there are more correct trials than error trials in each fold, and then the contribution of the correct trials classification accuracy will have a slightly larger contribution to the final value. Thus, we obtained 10 values for each inverse solution. Then, we applied the Wilcoxon test with the null hypothesis H_0 being that the means of the samples are equal; we performed it at the 0.05 significance level. We obtained that H_0 was rejected with a p-value of

$p = 0.028$. Thus, there is a significant difference between the models in terms of classification performances.

7.2.2 Localization of Features

The second part of the comparison of the inverse models consists in observing the location of the selected channels in the brain. Figures 24, 25 and 26 already show the localization of the 50 best vertices selected by the CCD inverse model in folds number 1, 5 and 9, with the LDA-based feature selection. Figures 35 to 37 show the localization of the features made by the ELECTRA-LAURA model in the same conditions: the red points are the 50 selected voxels, whereas the blue crosses represent the location of anterior cingulate cortex. In addition, table 6 reports the same analysis described in section 6.3.2 aiming at characterizing the regularity in the selection of the channels for each method.

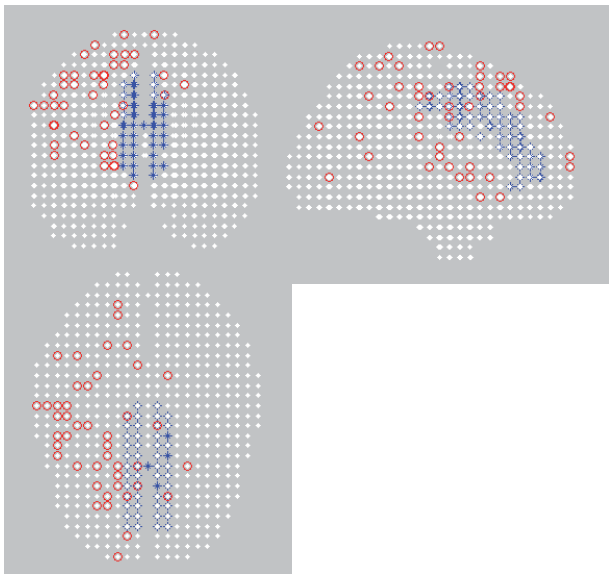

Figure 35: Localization of ELECTRA-LAURA features made by the LDA-based method on fold 1 of the cross-validation procedure.

Inverse model	Averaged % of identical selected channels over all possible pairs of folds	# of identical selected channels in all folds
CCD inverse	45.89%	1
ELECTRA-LAURA	63.89%	5

Table 6: Stability of channel selections depending on the inverse solution. The 20 best selected channels are considered, and LDA-based feature selection is applied.

We note that:

- While the CCD inverse model selects channels located in physiologically relevant areas like pre-SMA and ACC, it appears that the 50 best voxels selected by ELECTRA-LAURA do not show

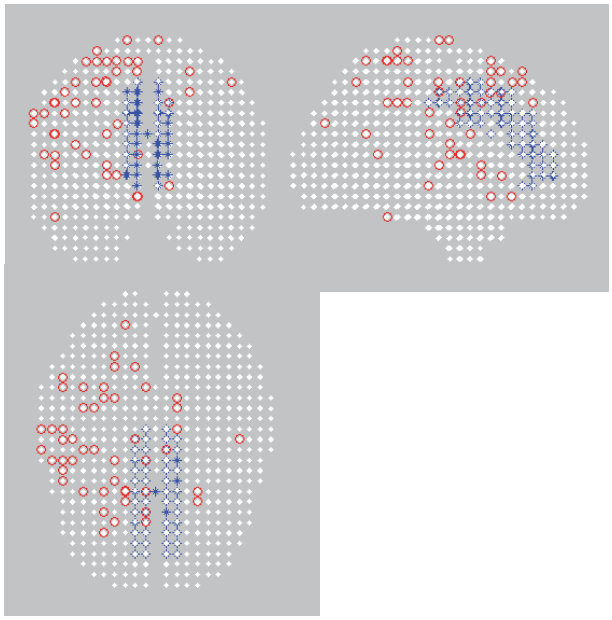

Figure 36: Localization of ELECTRA-LAURA features made by the LDA-based method on fold 5 of the cross-validation procedure.

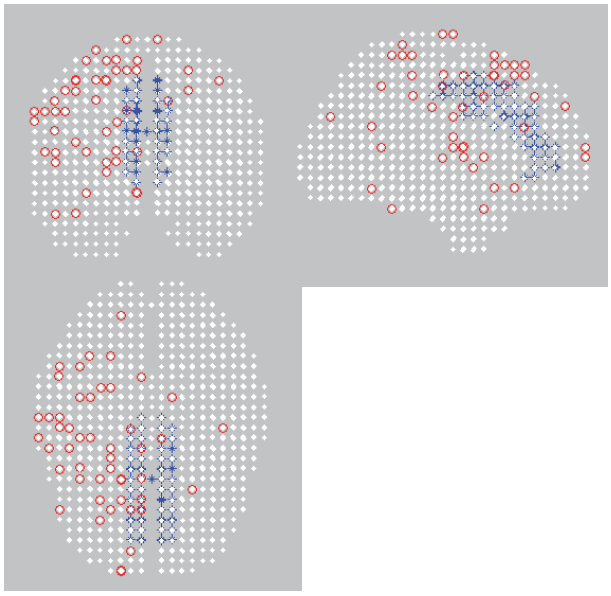

Figure 37: Localization of ELECTRA-LAURA features made by the LDA-based method on fold 9 of the cross-validation procedure.

similar properties; there is no clear cluster in the configuration of the selected voxels, even though most of the selected voxels are in the right hemisphere. Moreover, none of the 50 best voxels are located in the ACC (figures can be misleading, since these are 2D views), and the 5 voxels that are present in all folds have no physiological meaning, since they are located on the cortical

surface of right hemisphere. Therefore, a physiological interpretation of the localization of the features is difficult in this case.

- Interestingly, we see in table 6 that the ELECTRA-LAURA method selects its channels with a bigger regularity than the CCD inverse model. However, we saw in the previous section that the performances of the CCD inverse model in terms of classification accuracy were significatively better than those of LAURA-ELECTRA inverse solution, even though the difference was not so big. This somehow paradoxical result proves that the most important property for a good classification performance is not the regularity of the selection itself, but more the regularity in the *clustering tendency* of the selection.

8 Comparing EEG and CCD Inverse Model

In order to complete our study on inverse solutions and their integration into BCI systems, we compare in this chapter the performances of the CCD inverse model with those of EEG signals in the case of the cross-validation procedure described in section 5.3. This comparison aims at extending IDIAP studies on BCI error-related potentials, since we would like to assess the improvement due to the use of inverse solutions in this specific application.

8.1 Cross-Validation

The cross-validation procedure was done for the five subjects, on the 1^{st} day of recording. The only differences with respect to the process adopted in [15] is that we selected in each fold the 2 best EEG electrodes before classification. Doing this, we can do a better comparison with the inverse models, since we select the best channels in each fold as well. In addition, we use a LDA-based classifier instead of the gaussian classifier used in [15]; thus, we expect our classification scores to be a little bit lower than in the article. But as we already mentioned before, we are more interested in the relative differences between scalp EEG and CCD inverse model. Figures 38 to 42 show the results of the cross-validation for the 5 subjects. Well classified correct and erroneous trials are shown for the CCD inverse model for different number of selected vertices. In addition, superimposed dashed lines represent the corresponding EEG classification scores for correct and erroneous trials, obtained by selecting the 2 best electrodes in each fold. Results are summarized in the upper part of table 7 for the five subjects and the average of them; for each subject, the classification score for the opimal number of selected channels is displayed. We can do the following observations:

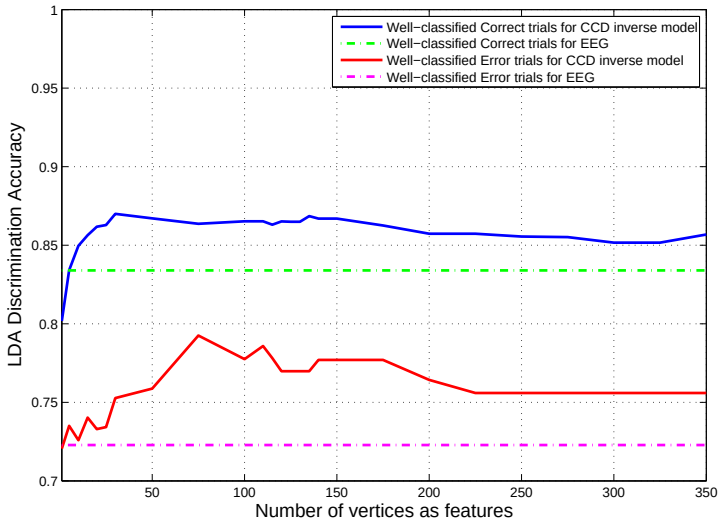

Figure 38: Results of 10-fold cross-validation for day I, subject 1. Results of CCD inverse model are shown for different number of selected features; superimposed dashed lines represent the results of EEG by taking the 2 best electrodes in each fold.

- By looking at figures 38 to 42, we see that the classification scores provided by the inverse solution is better than the results provided by EEG in most cases. Here, it is important to note that for the five subjects, the best classification scores are often obtained with a small number of selected channels. Even if for subject 3 and 5, the maximal scores are obtained for a

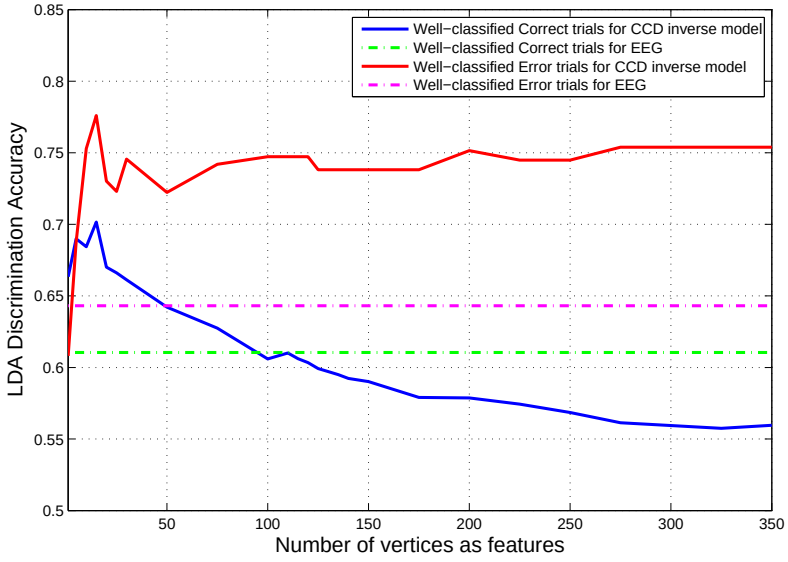

Figure 39: Same as figure 38 for subject 2.

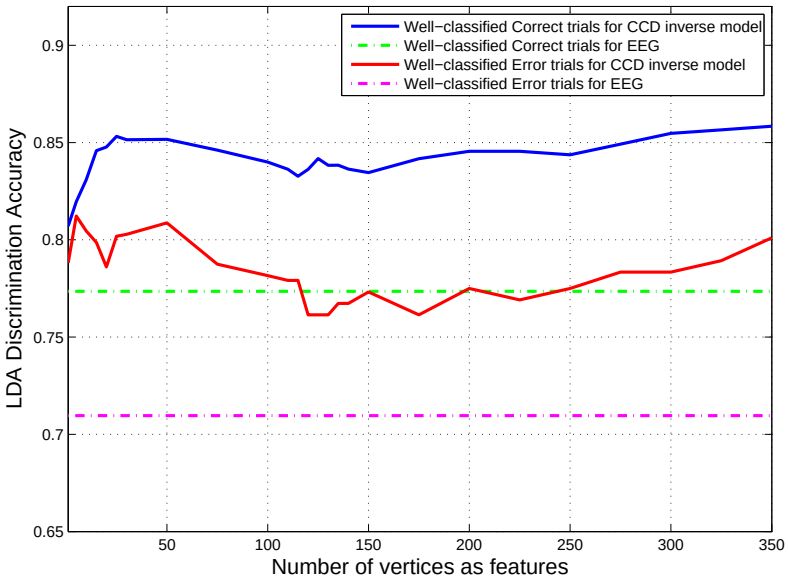

Figure 40: Same as figure 38 for subject 3.

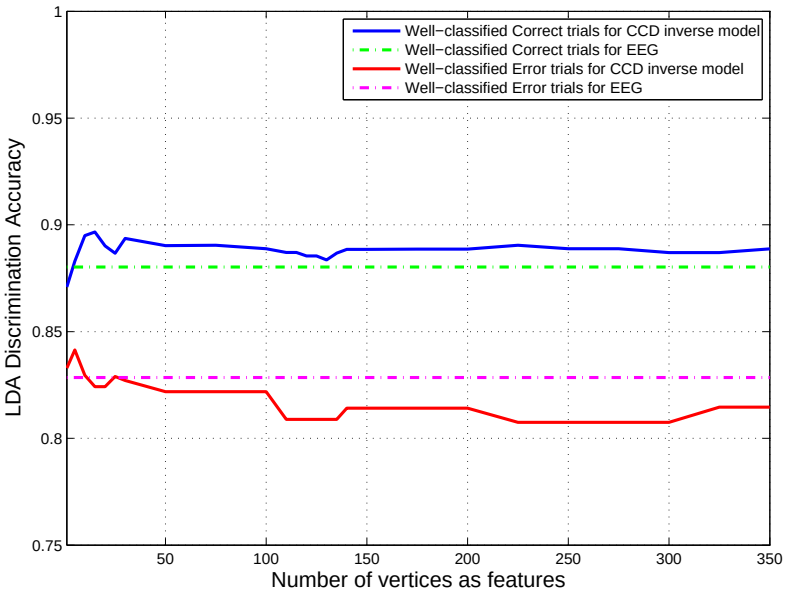

Figure 41: Same as figure 38 for subject 4.

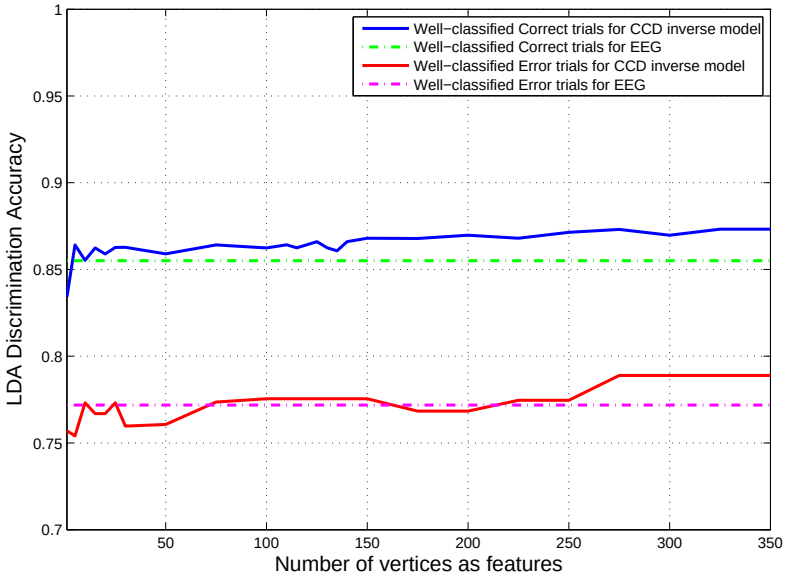

Figure 42: Same as figure 38 for subject 5.

bigger number of selected channels, we see on the profiles that with a small number of features, the scores are already in the same range as the maximal score. Moreover, for subject 2, the classification scores of well-classified correct trials decrease drastically as the number of selected vertices increases.

- Table 7 illustrates well the improvement due to the use of the CCD inverse solution. Indeed, for all subjects, the classification scores are significantly higher for the CCD inverse solution than for EEG, and the standard deviations of CCD inverse results are smaller in average.

- In order to assess the significance of this improvement, we performed a Wilcoxon test similarly as we did in section 7.2.1 on the accuracy values of each subject for EEG and CCD inverse model, and on the averaged accuracy values. The results are shown in table 8: only subjects 4 and 5 do not show significant improvement when we use the inverse model. More importantly, the test on the grand average rejected the null hypothesis, proving that in average, applying an inverse model provides significantly better results.

	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5	**Average**
	10-fold cross-validation Day I: EEG					
C	83.4 ± 7.3	61.1 ± 13.8	77.4 ± 7.5	88.0 ± 3.4	85.5 ± 7.3	$\mathbf{79.1 \pm 10.8}$
E	72.3 ± 14.0	64.3 ± 20.1	71.0 ± 15.7	82.9 ± 11.3	77.2 ± 13.1	$\mathbf{73.5 \pm 7.0}$
	10-fold cross-validation Day I: CCD inverse model					
C	87.0 ± 2.3	70.2 ± 13.8	85.8 ± 4.3	89.7 ± 3.5	87.3 ± 5.9	$\mathbf{83.6 \pm 7.7}$
E	79.3 ± 6.5	77.6 ± 19.5	81.2 ± 19.5	84.1 ± 10.7	78.9 ± 10.4	$\mathbf{79.9 \pm 2.8}$
	Day II classified with Day I: EEG					
C	77.8 ± 7.3	82.6 ± 3.9	75.7 ± 7.5	90.8 ± 3.1	77.8 ± 7.3	$\mathbf{80.9 \pm 6.1}$
E	73.8 ± 10.3	41.5 ± 17.9	65.3 ± 14.6	68.5 ± 13.9	73.8 ± 10.3	$\mathbf{64.6 \pm 13.4}$
	Day II classified with Day I: CCD inverse model					
C	88.0 ± 5.9	82.5 ± 4.1	86.9 ± 5.6	91.4 ± 4.1	88.5 ± 6.9	$\mathbf{87.5 \pm 3.2}$
E	79.3 ± 9.8	45.2 ± 17.5	74.9 ± 9.7	81.4 ± 10.1	77.6 ± 9.8	$\mathbf{71.7 \pm 15.0}$

Table 7: Percentages (mean and standard deviations) of correctly recognized correct trials (C) and error trials (E) for the five subjects and the average of them, performing a 10-fold cross-validation on day I, and using data of day I to classify day II. Results are shown for both EEG and CCD inverse model.

10-fold cross-validation Day I		
Subject 1	Subject 2	Subject 3
H_0 rejected	H_0 rejected	H_0 rejected
$p = 0.0101$	$p = 0.0191$	$p = 0.0065$
Subject 4	Subject 5	**Average**
H_0 not rejected	H_0 not rejected	$\boldsymbol{H_0}$ **rejected**
$p = 0.1974$	$p = 0.4723$	$\boldsymbol{p = 0.001}$

Table 8: Results of Wilcoxon tests applied on the 5 subjects and the average of them for the cross-validation on day I, to assess significant differences between EEG and CCD inverse model. Tests are performed at the 0.05 significance level; the null hypothesis H_0 states that the means are equal.

8.2 Generalization

The second objective of our analysis was to quantify the improvement due to the use of our CCD inverse solution when generalizing over extended periods of time. We built a LDA classifier based on the whole data set of the 1^{st} day of recording, and we classified the 10 sessions of the 2^{nd} day of recording with this classifier. We remind that the delay between the two days of recordings was about 3 months. The results are shown in the lower part of table 7. Once again, the classification scores increase significantly when using the inverse solution. Maximal improvements of more than 10% were obtained for the classification of correct trial for subjects 1, 3 and 5, and for the classification of erroneous trials for subject 4. The average improvement was about +7% for both correct and erroneous trial classification. In table 9, the results of Wilcoxon tests performed on each subject and on the average of the subjects confirm the global significance of this improvement.

Day II classified with Day I		
Subject 1	Subject 2	Subject 3
H_0 rejected $p = 0.0036$	H_0 not rejected $p = 0.79$	H_0 rejected $p = 0.001$
Subject 4	Subject 5	**Average**
H_0 not rejected $p = 0.42$	H_0 rejected $p = 0.0002$	$\boldsymbol{H_0}$ **rejected** $\boldsymbol{p = 0.0002}$

Table 9: Results of Wilcoxon tests applied on the 5 subjects and the average of them for the classification of Day II using a classifier built on data of Day I, to assess significant differences between EEG and CCD inverse model. Tests are performed at the 0.05 significance level; the null hypothesis H_0 states that the means are equal.

9 Discussions and Conclusions

In this last chapter, a synoptic view of this master thesis is provided as well as propositions of future investigations. The different issues that we addressed during this study, namely feature selection methods and inverse solutions, will be disscussed successively, in order to summarize the different findings of this long-term work.

9.1 Discussions

9.1.1 Feature Selection Methods

During this work, we implemented three feature selection methods, namely ReliefF algorithm, a modified DP function and an LDA-based filter method, and we applied them in the context of a specific BCI application, namely error-related potentials. We analyzed their respective performances both in terms of localization of the features and classification accuracy. Our conclusions are the following:

- All our methods were able to select relevant scalp electrodes as well as relevant intracranial channels in the context of an error-potential study conducted at IDIAP and extended in this work.

- In terms of localization of the features, the modified DP method showed an impressive regularity in the selection of its features. From one fold of the cross-validation to another, most of the selected features were identical with the modified DP function, whereas the features changed slightly from one fold to the other for the two other methods. However, the best classification performance was obtained by the LDA-based method, which was the worst method in terms of regularity in the selection. These results are not contradictory, since the features selected by the LDA-based method were always located in the same physiologically meaningful area as the modified DP function, namely pre-SMA and ACC. Thus, it seems that the strict regularity of the selection is not very important in order to achieve good classification results, as long as the configuration of the selected features is representative of the underlying physiological process.

- In terms of computational time, the ReliefF algorithm is by far the worst method. When dealing with high dimensional input spaces, which is the case with inverse solutions, this iterative algorithm takes a lot of time in order to converge and return its result. In addition, since the number of iterations is not known for a given application, its choice is arbitrary and we cannot always be sure that the algorithm converged correctly. On the contrary, the LDA-based feature selection method is quite fast, even when we applied it on inverse solution data, and the modified DP function is almost instantaneous. Thus, for future investigations, and in case of online implementation, we suggest to keep the LDA-based method and the modified DP function only.

9.1.2 Inverse Solutions

This thesis was the opportunity to have a first contact with the captivating research field of inverse models. It is clear that the goal of this work was not to go into the details of the complex theories of inverse solutions, but rather to apply them in practical BCI applications, and assess their abilities to provide new results for BCI research.

- It appeared that the so-called CCD inverse model provided impressive results, both in terms of localization of cortical activity and classification of single trials in the context of BCI error-related potentials. All the foci of activity observed with this model during our experiments were in agreement with neurophysiological evidences in the field of error potentials. Further, it was statistically proven that this inverse solution can improve the performances of an error detection system for BCI with respect to a system based on EEG.

- We encountered some problems in localizing correctly clusters of activity related to error detection with the ELECTRA-LAURA model. Since our work is only the beginning of a longer process aiming at combining inverse solutions like ELECTRA-LAURA with standard BCI methods, we can only conclude that further investigations are needed in order to understand the reasons of these unexpected results. Indeed, estimated local field potentials have a crucial physiological meaning, and integrating this model into BCI systems could be useful to better understand the neuronal processes that we try to decode.

- During our experiments on inverse solutions, we found that in average, only a small number of channels had to be selected in order to achieve very good results in terms of single trial classification of error-related potentials signals. This finding is probably the most important of this thesis, since the main problem with inverse solutions is the dramatic increase of the initial input space. Thus, by selecting a small number of relevant intracranial channels by means of appropriate feature selections, we can reduce the problem to the size of a standard EEG problem, and benefit from the better spatial resolution of inverse solutions without being limited by computational problems. Moreover, online implementations including inverse solutions can be allowed if the number of channels is not too big. These results show great promise for future investigations about inverse solutions and their integration into BCI systems.

- sLORETA inverse model has been used in this study as a visualization tool only. Its abilities to describe neurophysiological processes over time are impressive, as well as its simplicity of use. sLORETA allowed us to begin a pyschophysiological description of the underlying process of BCI error-related potentials. An interesting study would consist in pointing out the differences between "standard" error potentials elicited by the subject himself, and BCI-driven error-potentials. The first elements of such a study are provided in this work. Moreover, we could even extend the investigations to the comparison of different experimental protocols related to BCI error-related potentials. In our study, commands are delivered manually by the users; it would be interesting to see the differences in the psychophysiological process when the subjects deliver the same commands, but mentally, by means of a brain-computer interface.

In this thesis, we considered the application of error-related potentials in order to apply our methods and inverse models. The choice of this particular application was based on this simple idea: since error potentials are temporally well defined and focused on precise areas of the brain, it would facilitate the estimation of the quality of the methods and of the inverse solutions.
However, now that inverse solutions have proven to be useful for BCI, we have to define precisely which applications of BCI research really need the contribution of inverse models. Especially, we think that inverse solutions are really useful for decoding neuronal processes that are not clearly localized, such as movement imagination for instance. Indeed, this kind of mental task involves different processes synchronizing at different instants and different locations in the brain. Scalp EEG electrodes will have difficulties in order to decode precisely such neuronal patterns, and the enhanced spatial resolution provided by inverse solutions becomes crucial in such cases. Thus, future investigations will have to be done in order to assess the potentialities of inverse solutions in the context of motor imagery.

9.2 Conclusion

MAIA Project The current study is related to an European project called MAIA[14] (Mental Augmentation through Determination of Intended Action – Non Invasive Brain Interaction with Robots). The goal of this project is to develop non-invasive prosthesis driven by a BCI system. Particularly, one of the major objectives is to perform recognition of the subject's motor intent from the analysis of high resolution brain maps, which estimates intracranial potentials from scalp EEG. In addition, recognition of cognitive states such as error-related potentials detection will be integrated in the system. Final applications will be an intelligent BCI-driven wheelchair, or the control of a robot arm.

[14] http://www.maia-project.org/

Thanks to this project, I had the opportunity to attend the 2006 MAIA Workshop[15], which took place in Rome in November 2006.

Finally, the results of this study are satisfying, since the main goals of the thesis are achieved: we developed and compared feature selection methods, and we assessed the potentiality of inverse models in the context of BCI research. Of course, this master thesis is only the beginning of a much harder and longer work, aiming at integrating inverse models in a real BCI system. However, the results reported in this thesis give new insights into how such models can be processed, and future investigations promise to be exciting.

[15]More infos on MAIA website.

References

[1] P. Ahammad, R. Bajcsy, and S. S. Sastry. A framework for characterization and comparison of event related neuronal activity. (UCB/EECS-2006-128), October 2006.

[2] E. Alpaydin. *Introduction to Machine Learning*. MIT Press, 2004.

[3] F. Babiloni, C. Babiloni, F. Carducci, G. L. Romani, P. M. Rossini, L. M. Angelone, and F. Cincotti. Multimodal integration of high resolution EEG and functional magnetic resonance: A simulation study. *Neuroimage*, 19:1–15, 2003.

[4] F. Babiloni, C. Babiloni, L. Locche, F. Cincotti, P. M. Rossini, and F. Carducci. High-resolution electro-encephalogram: Source estimates of laplacian-transformed somatosensory-evoked potentials using a realistic subject head model constructed from magnetic resonance imaging. *Med. Biol. Eng. Comput.*, 38:512–519, 2000.

[5] F. Babiloni, F. Cincotti, C. Babiloni, F. Carducci, D. Mattia, L. Astolfi, A. Basilico, P. M. Rossini, L. Ding, Y. Ni, J. Cheng, K. Christine, J. Sweeney, and B. He. Estimation of the cortical functional connectivity with the multimodal integration of high-resolution EEG and fMRI data directed by transfer function. *Neuroimage*, 24:118–131, 2005.

[6] M. F. Bear, B. W. Connors, and M. A. Paradiso. *Neuroscience: Exploring the Brain*. Baltimore: Lippincott, 2001.

[7] B. Blankertz, G. Dornhege, C. Schäfer, R. Krepki, J. Kohlmorgen, K.-R. Müller, V. Kunzmann, F. Losch, and G. Curio. Boosting bit rates and error detection for the classification of fast-paced motor commands based on single-trial EEG analysis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):127–131, 2003.

[8] C.S. Carter, T.S. Braver, D.M. Barch, M.M Botvinick andD. Noll, and J.D. Cohen. Anterior cingulate cortex, error detection, and the online monitoring of performance. *Science*, 280:747–749, 1998.

[9] F. Cincotti. *Multimodal Integration of Neuroelectromagnetic and fMRI Data: the Role of Different Metrics in the Solution of the Linear Inverse Problem*. PhD thesis, Università degli Studi di Roma La Sapienza, 2002.

[10] A. M. Dale and M. I. Sereno. Improved localization of cortical activity by combining EEG and MEG with MRI cortical surface reconstruction: A linear approach. *J. Cogn. Neurosci.*, 5:162–176, 1993.

[11] F. Darvas, D. Pantazis, E. Kucukaltun-Yildirim, and R. M. Leahy. Mapping human brain function with MEG and EEG: Methods and validation. *Neuroimage*, 23:S289–S299, 2004.

[12] S. Debener, M. Ullsperger, M. Siegel, K. Fiehler, D. Y. von Cramon, and A. K. Engel. Trial-by-trial coupling of concurrent electroencephalogram and functional magnetic resonance imaging identifies the dynamics of performance monitoring. *The Journal of Neurosciences*, 25(50):11730–11737, 2005.

[13] O. Donchin, A. Gribova, O. Steinberg, H. Bergman, S. Cardoso de Oliveira, and E. Vaadia. Local field potentials related to bimanual movements in the primary and supplementary motor cortices. *Experimental Brain Research*, 140:46–55, 2001.

[14] M. Falkenstein, J. Hoormann, S. Christ, and J. Hohnsbein. ERP components on reaction errors and their functional significance: a tutorial. *Biological Psychology*, 51:87–107, 2000.

[15] P. W. Ferrez and J. del R. Millán. Error-related EEG potentials generated during brain-computer interaction. Number 69, Martigny, Switzerland, 2006. IDIAP. MAIA-Conference Paper.

[16] P.W. Ferrez and J. del R. Millán. You are wrong!—Automatic detection of interaction errors from brain waves. In *Proc. 19th Int. Joint Conf. Artificial Intelligence*, 2005.

[17] K. Fiehler, M. Ullsperger, and D. Y. von Cramon. Neural correlates of error detection and error correction: is there a common neuroanatomical substrate? *European Journal of Neurosciences*, 19:3081–3087, 2004.

[18] M. Fuchs, M. Wagner, T. Kohler, and H.-A. Wischmann. Linear and nonlinear current density reconstructions. *J Clin Neurophysiol.*, 16(3):267–295, 1999.

[19] V. Galva, J. Chen, and N. M. Weinberger. Long-term frequency tuning of local field potentials in the auditory cortex of the waking guinea pig. *J. Assoc. Res. Otolaryngol.*, 2(3):199–215, 2001.

[20] W. J. Gehring and R. T. Knight. Prefrontal-cingulate interactions in action monitoring. *Nature Neurosciences*, 3(5):516–520, 2000.

[21] S. L. Gonzalez Andino, R. Grave de Peralta Menendez, G. Thut, J. del R. Millán, P. Morier, and T. Landis. Very high frequency oscillations (VHFO) as a predictor of movement intentions. *NeuroImage*, 32(1):170–179, 2006.

[22] R. Grave de Peralta Menendez, S. L. Gonzalez Andino, G. Lantz, C. M. Michel, and T. Landis. Noninvasive localization of electromagnetic epileptic activity. i. method descriptions and simulations. *Brain Topography*, 14(2):131–137, 2001.

[23] R. Grave de Peralta Menendez, S. L. Gonzalez Andino, S. Morand, C. M. Michel, and T. Landis. Imaging the electrical activity of the brain: ELECTRA. *Hum Brain Mapp*, 91(1):1–12, 2000.

[24] R. Grave de Peralta Menendez et al. Electrical neuroimaging based on biophysical constraints. *Neuroimage*, 21(2):527–539, 2004.

[25] C.B. Holroyd and M.G.H. Coles. The neural basis of human error processing: Reinforcement learning, dopamine and the error-related negativity. *Psychological Review*, 109:679–709, 2002.

[26] K. Kira and L. A. Rendell. A practical approach to feature selection. In *ML92: Proceedings of the Ninth International Workshop on Machine Learning*, pages 249–256, San Francisco, CA, USA, 1992. Morgan Kaufmann Publishers Inc.

[27] I. Kononenko. Estimating attributes: Analysis and extensions of RELIEF. In *European Conference on Machine Learning*, pages 171–182, 1994.

[28] M. Lebedev and M. Nicolelis. Brain machine interfaces: Past, present and future. *Trends Neurosci.*, 29:530–546, 2006.

[29] A. C. Metting van Rijn, A. Peper, and C. A. Grimbergen. High quality recording of bioelectric events. Interference reduction, theory and practice. *Med. & Biol. Eng. & Comput.*, 28:389–397, 1990.

[30] C. M. Michel, M. Murray, G. Lantz, S. L. Gonzalez Andino, L. Spinelli, and R. Grave de Peralta Menendez. EEG source imaging. *Clin Neurophysiol*, 115(10):2195–2222, 2004.

[31] J. del R. Millán. Brain-Computer Interfaces. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 178–181. MIT Press, 2nd edition, 2002.

[32] J. del R. Millán, M. Franzé, J. Mouriño, F. Cincotti, and F. Babiloni. Relevant EEG features for the classification of spontaneous motor-related tasks. *Biological cybernetics*, 86(2):89–95, 2002.

[33] J. del R. Millán, F. Renkens, J. Mouriño, and W. Gerstner. Non-invasive brain-actuated control of a mobile robot by human EEG. *IEEE Trans. on Biomedical Engineering, Special Issue on Brain-Machine Interfaces*, 2004.

[34] S. Nieuwenhuis, K. R. Ridderrinkhof, J. Blom, G. P. H. Band, and A. Kok. Error-related brain potentials are differentially related to awareness of response errors: Evidence from antisaccade task. *Psychophysiology*, 38:752–760, 2001.

[35] L.C. Parra, C.D. Spence, A.D. Gerson, and P. Sajda. Response error correction - a demonstration of improved human-machine performance using real-time EEG monitoring. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):173–177, 2003.

[36] R. D. Pascual-Marqui. Standardized low resolution brain electromagnetic tomography (sLORETA): technical details. *Methods & Findings in Experimental & Clinical Pharmacology*, 24D:5–12, 2002.

[37] R. D. Pascual-Marqui, M. Esslen, K. Kochi, and D. Lehmann. Functional imaging with low resolution brain electromagnetic tomography (LORETA): a review. *Methods & Findings in Experimental & Clinical Pharmacology*, 24C:91–95, 2002.

[38] R. D. Pascual-Marqui, C. M. Michel, and D. Lehmann. Low resolution electromagnetic tomography: a new method for localizing electrical activity in the brain. *International Journal of Psychophysiology*, 18:49–65, 1994.

[39] R. Plonsey. The nature of sources of bioelectric and biomagnetic fields. *Biophys. J.*, 39:309–312, 1982.

[40] J. Rickert, S. C. de Oliveira, E. Vaadia, A. Aertsen, S. Rotter, and C. Mehring. Encoding of movement direction in different frequency ranges of motor cortical local field potentials. *The Journal of Neuroscience*, 25(39):8815–8824, 2005.

[41] M. Robnik-Sikonja and I. Kononenko. Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning*, 53:23–69, 2003.

[42] G. Schalk, J.R. Wolpaw, D.J. McFarland, and G. Pfurtscheller. EEG-based communication: presence of an error potential. *Clinical Neurophysiology*, 111:2138–2144, 2000.

[43] H. Scherberger and M. R. Jarvis. Cortical local field potential encodes movement intentions in the posterior parietal cortex. *Neuron*, 46:347–354, 2005.

[44] M. Scherg. Fundamentals of dipole source potential analysis. *Advances in audiology (Adv. audiol.)*, 6:40–69, 1990.

[45] L. Sörnmo and P. Laguna. *Bioelectrical Signal Processing in Cardiac and Neurological Applications*. Elsevier Academic Press, 2005.

[46] J. Talairach and P. Tournoux. *Co-Planar Stereotaxic Atlas of the Human Brain*. Thieme, 1988.

[47] H.T. van Schie, R.B. Mars, M.G.H. Coles, and H. Bekkering. Modulation of activity in medial frontal and motor cortices during error observation. *Nature Neuroscience*, 7(5):549–554, 2004.

[48] V. van Veen and C. S. Carter. The timing of action-monitoring processes in the anterior cingulate cortex. *The Journal of Cognitive Neurosciences*, 14(4):593–602, 2002.

[49] J. D. Victor, K. Purpura, E. Katz, and B. Mao. Population encoding of spatial frequency, orientation, and color in macaque V1. *Journal of Neurophysiology*, 72:2151–2166, 1994.

[50] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113:767–791, 2002.