

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Applied Advanced Classifiers for Brain Computer Interface

José Luis Martínez, Antonio Barrientos  
*Politechnic University of Madrid  
 Spain*

## 1. Introduction

Since that Dr. Hans Berger discovered the electrical nature of the brain, it has been considered the possibility to communicate persons with external devices only through the use of the brain waves (Vidal, 1973).

Brain Computer Interface technology is aimed at communicating with persons using external computerised devices via the electroencephalographic signal as the primary command source (Wolpaw, J.R.; et al., 2000), (Birbaumer, N; et al., 2000). In the first international meeting for BCI technology it was established that BCI *"must not depend on the brain's normal output pathways of peripheral nerves and muscles"* (Wolpaw, J. R.; et al., 2002). The primary uses of this technology are to benefit persons with blocking diseases, such as: Amiotrophic Lateral Sclerosis (ALS), brainstem stroke, or cerebral palsy; or persons whom have suffered some kind of traumatic accident like for example paraplegic (E. Donchin and K. M. Spencer and R. Wijesinghe, 2000). Actually different types of classifications can be established for BCI technology, from the physiologic point of view BCI devices can be classified in exogenous and endogenous. The exogenous devices provide some kind of stimuli to the user and they analyse the user's responds to them, examples of this class are devices based on visual evoked potential or P300 (E. Donchin and K. M. Spencer and R. Wijesinghe, 2000). On the contrary, the endogenous devices do not depend on the user's respond to external stimuli, they base their operation in detecting and recognising brain-wave patterns controlled autonomously by the user, examples of this class are devices based on the desynchronisation and synchronisation of  $\mu$  and  $\beta$  rhythms (Wolpaw, J. R.; et al., 2002), (Pfurtscheller et al., 2000a), (Pineda, J.A. et al., 2003).

But in any case, independently of the classification criteria, an algorithm that detects, acquires, filters, learns and classifies the electroencephalographic signal is required in order to control an external device using thoughts, associating some mental patterns to device commands, as it is shown in the block diagram of Figure 1, (Kostov, A., 2000), (Pfurtscheller et al., 2000b).

The first block is in charge of acquiring and amplifying the brain signal, allocating the electrodes on specific places on the scalp in case of using superficial electrodes, or inside the brain in case of using intracortical ones; in the second block the signal is sampled, quantified and codified at periodic intervals of time in order to digitalize it, to simplify the following phases the digitalised signal may be filtered, for example to reduce the noise level obtaining a better SNR signal or identifying and processing artifacts. After this, in order to obtain a set

of parameters that represent the temporal window of the acquired brain signal the process of feature extraction is performed, because the main changes in brain activity are associated to changes in the power amplitude of band frequencies, spectrograms based on FFT are used to obtain initial feature vectors of six components (Obermaier et al., 2001) (Proakis & Manolakis, 1997).

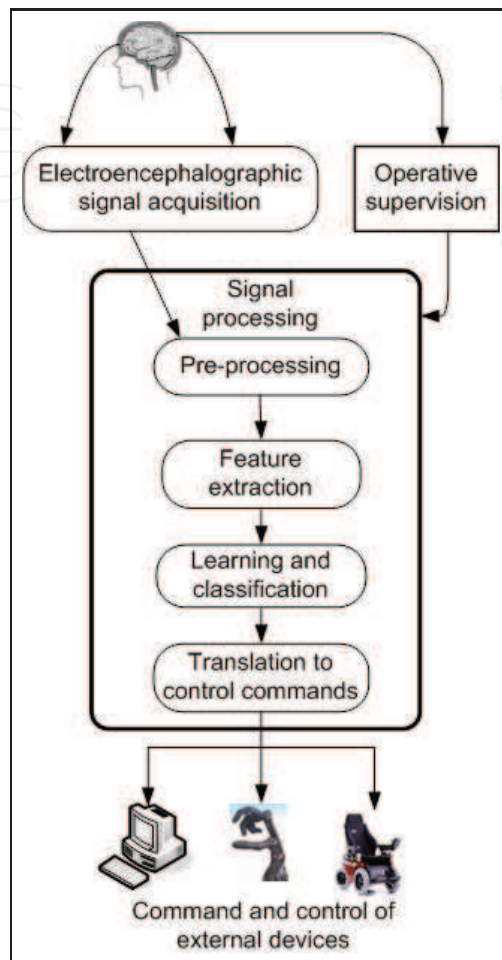


Fig. 1. Block diagram of a BCI device.

In the next block the features are processed in order to detect a specific event in the case of exogenous devices, or for identifying, learning, and recognising signal cerebral patterns, that are going to be used as inputs for the following block that translates them to control commands of the external device.

Finally, but not less important, is the Operative Supervision block which sets the operative mode of the BCI device under the user's supervision, this is if the device is operating in on-line / off-line mode, or if it is modifying its internal parameters during the learning phase in order to adjust to the user's cerebral activity.

In the experiments considered for this paper only two electroencephalographic channels "C3 and C4" have been considered to capture the endogenous electroencephalographic signal from the subject. In order to facilitate the use of this technology it is important to make it easy to use, the "cosmosis" or how the user's looks like wearing the BCI device is also important, this is the reason that the number of electrodes employed in these devices is a global key feature, as the fewer of electrodes used, the higher the comfort (Wolpaw, 2007).

This chapter deals with the application of these concepts for developing BCI devices, focusing in the classification of the user's cerebral activity.

The contents of this chapter are distributed in the following sections:

- The first section contemplates this introduction.
- The second section briefly describes the signal processing phase and the selection of the features that describe the user's cerebral activity.
- In the third section is analysed the discrimination capability between the feature vector populations sampled when the user develop three different cognitive tasks.
- Afterwards, in the fourth section, it is assessed the best component combination of the feature vector in order to reduce the feature space dimensionality improving the discrimination capability.
- The fifth section describes different types of advanced classifiers based on: Neural Networks, Hidden Markov Models, and Support Vector Machines .
- The experiments, carried on with signal sampled from real users, are described in the sixth section. The different experimental paradigms, results, and analysis, are explained in it.
- Finally the seventh section is devoted to conclusions.

## 2. Signal processing and feature selection

The tests described below were carried out on five male healthy subjects, one of them has been trained before, but the other four were novice in the use of the system.

In order to facilitate the mental concentration on the proposed activities, the experiments were carried on in a room with low level of noise and under controlled environmental conditions, all electronic equipments external to the experiment around the subject were switched off to avoid electromagnetic artifacts. The subjects were sat-down in front of the acquisition system monitor, at 50 cm from the screen, their hands were in a visible position, the supervisor of the experiment controlled the correct development of it.

Two different types of experimental procedures had been considered for the acquisition of the user's cerebral signal. In the first one, the user concentrates on the proposed cognitive tasks meanwhile the system registers the cerebral activity but without communicating any feedback about the signal classification.

In the second type of experiments the user receives the classification feedback from a simple classifier based on artificial neural networks. These neural networks have been trained with registers associated to each cognitive task obtained from the previous kind of experiments.

Because in the first type of experiments there is not any kind of feedback they are named Off-line experimental procedures, in contrast to the second class called On-line experimental procedures.

The flow of activities for each experimental procedure are described in the following subsections.

### 2.1 Flow of the activities for the Off-line experimental procedure

The experimental process is shown on Figure 2.

- *Test of system devices.* It checks the correct level of battery, and state of the electrodes.
- *System assembly.* Device connections: superficial electrodes (Au-Cu), battery, bio-amplifier (g.BSamp by g.tec), acquisition signal card (PCI-MIO-16/E-4 by National Instrument), computer.

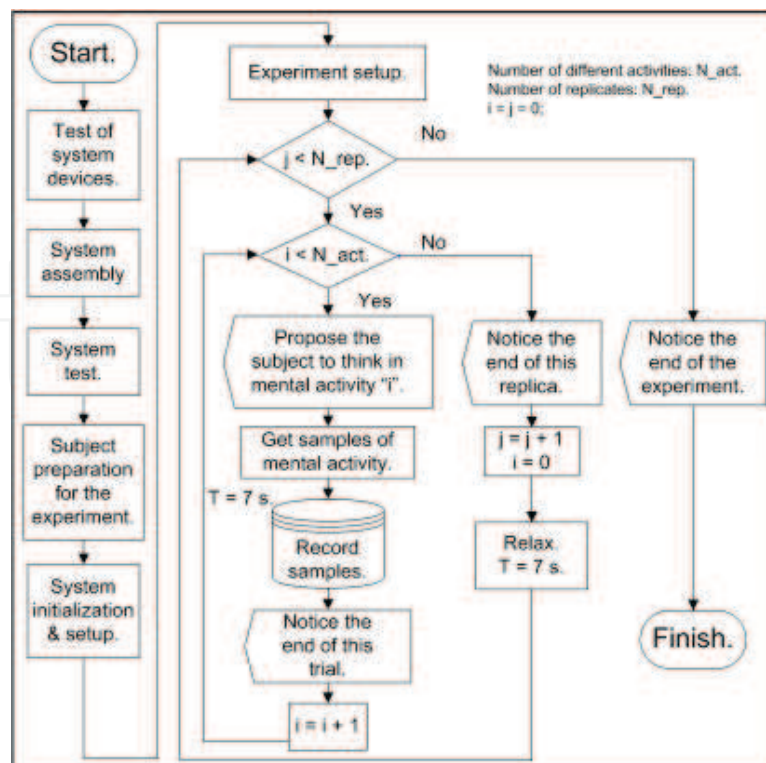


Fig. 2. Diagram of the experiment realization.

- *System test.* Verifies the correct operation of the whole system. To minimise noise from the electrical network the Notch filter (50Hz) of the bio-amplifier is switched on.
- *Subject preparation for the experiment.* Application of electrodes on subject's head. Impedance  $\leq 4\text{KOhms}$ .
- *System initialisation and setup.* Verification of data register. It is monitored the signal evolution, in the spectrogram should appear a very low component of 50 Hz.
- *Experiment setup.* The supervisor of the experiment sets-up the number of replications,  $N_{rep} = 10$ , and the quantity of different mental activities. The duration of each trial is  $t = 7\text{s}$ , the acquisition frequency is  $f_s = 384\text{Hz}$ . The system suggests to the subject to think about the proposed mental activity. A short relax is allowed at the end of each trial.

## 2.2 Flow of the activities for the On-line experimental procedure

In these tests, a cursor in the centre of the screen and a square goal are shown to the subject, the square goal appears half the trials on the left of the screen and the other half on the right. The subject shall try to move the cursor towards the goal thinking in the cognitive tasks proposed in the Off-line experiments. The experimental On-line process is shown on Figure 3.

- *Experiment set-up.* This phase determines the cognitive tasks used to move the cursor to the left and to the right, the number of trials and the time for each trial.
- *Display initialisation.* It initialises the display, for even trials the goal is on the right, for odds on the left.
- *Data acquisition.* In this phase 128 samples per electroencephalographic channel are acquired at  $f_s = 384\text{Hz}$ .

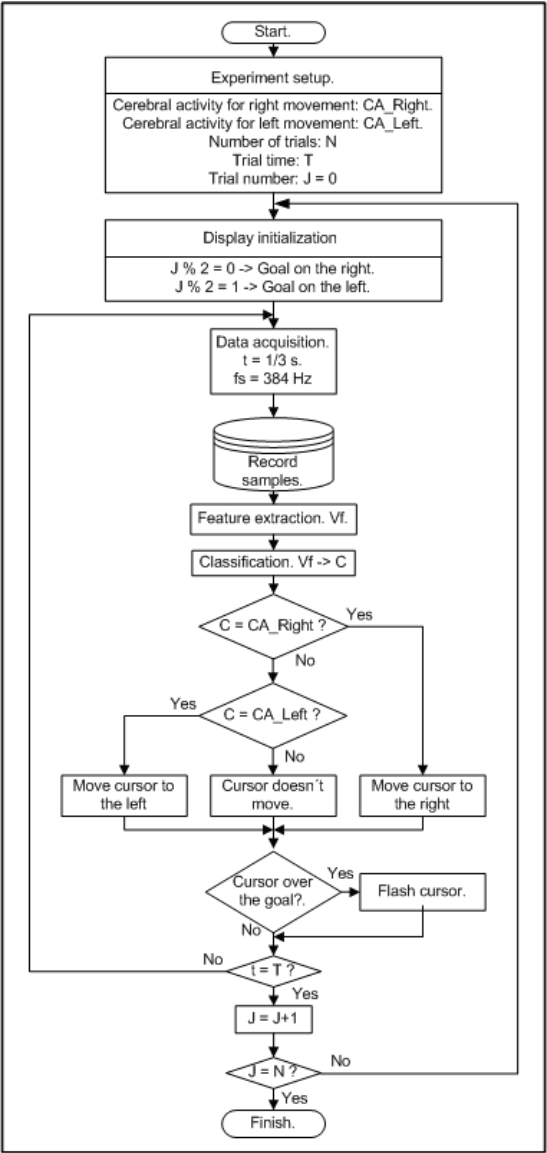


Fig. 3. Diagram of the On-line experiment realization.

- *Record samples.* The previous samples are recorded for a posterior analysis.
- *Feature extraction.* A vector of features is extracted from the acquired samples.
- *Classification.* The vector of features is classified as belonging to one of the previous mental tasks, and the associated movement is performed; if the vector can't be classified in any of the cerebral activities, the cursor doesn't move. If the trial time is exceeded a new trial is carried out until the  $N$  trials had been performed.

2.3 Position of electrodes

Electrodes were placed in the central zone of the skull, next to C3 and C4 (Penny, W. D.; et al., 2000), two pair of electrodes were placed in front of and behind of Rolandic sulcus, this zone is one with the highest discriminant power, it takes signal from motor and sensory areas of the brain (Birbaumer, N; et al., 2000), (Neuper, C.; et al., 2001).

Reference electrode was placed on the right mastoid, two more electrode were placed near to the corner of the eyes to register blinking.





Fig. 4. Electrode placement.

2.4 Description of cognitive tasks

The supervisor of the experiment suggests the subject to figure out the following mental activities:

*Activity A. Mathematical task.* Recursive subtraction of a prime number, i.e. 7, from a big quantity, i.e. 3.000.000.

*Activity B. Movement task.* The subject imagines moving their limbs or hands, but without the materialisation of it.

*Activity C. Relax.* The subject is relaxed.

These tasks will be the cerebral patterns to differentiate among them (Neuper, C.; et al., 2001).

2.5 Computational process

This section describes the procedure applied to recorded signal just before its classification.

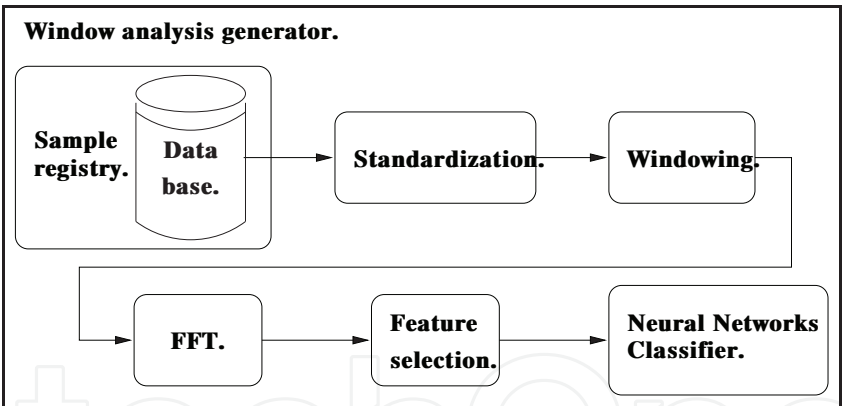


Fig. 5. Computational process flow.

2.5.1 Window analysis generator

In this block the registered signal is chopped in packages of samples, similar to the bundles of samples obtained from an acquisition card in an on-line BCI application. The number of samples in each package is a compromise between the goodness of the classification and the amount of time taken by this classification. An algorithm with very good classification and low number of mistakes will take a very big package, so the time between classifications will be also very big, it will do the algorithm useless for a real on-line BCI system, neither a very fast algorithm with small packages of samples but with a high number of mistakes will be useful.

In this work we have considered packages of 128 samples, the sample frequency is  $F_s = 384\text{Hz}$ , so it is possible to obtain a classification latency of  $t = 1/3\text{s}$ .

The duration of each activity is 7s, so there will be 21 classifications obtained from each register, no overlap between windows have been considered.

2.5.2 Standardisation

To compare the signal of different sessions is necessary to standardise the samples, avoiding for example that variations in the impedance of the electrodes changes the classification result. The standardisation of each analysis window consists in the subtraction of the average value and the division by the standard deviation, eqs. 1 to 3.

$$\mu = \frac{\sum_{i=1}^N x_i}{N} \tag{1}$$

$$\sigma^2 = \frac{\sum_{i=1}^N (x - \mu)^2}{N} \tag{2}$$

$$x' = \frac{x - \mu}{\sigma} \tag{3}$$

2.5.3 Windowing

The frequency leakage effect occurs when signals with low frequency components are chopped or processed with temporal windows with sharp edges, in this case in the spectrogram appears high frequency components as it is shown in Figure 6, (Harris, 1978).

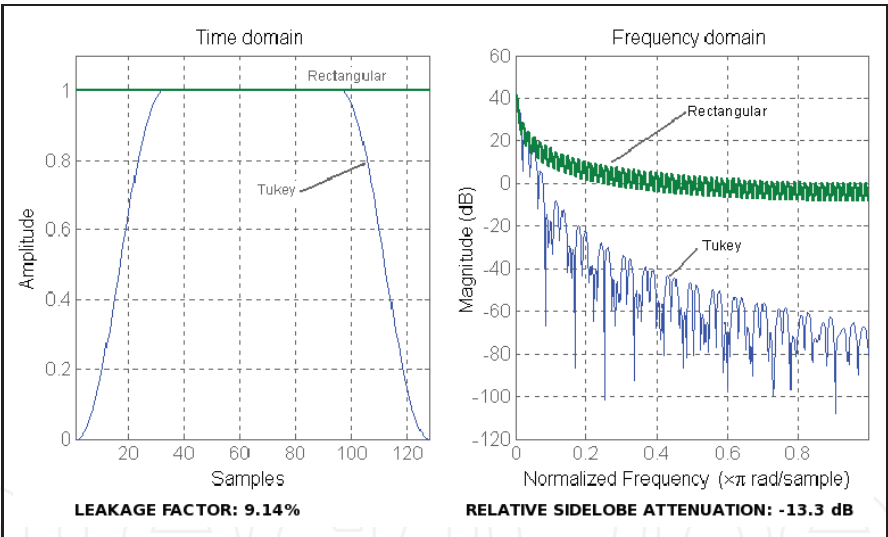


Fig. 6. Example of leakage effect.

In order to minimise this effect, seven different types of preprocessing windows have been applied to the standardised signal. The following types of windows have been considered:

- Rectangular window.  $h(n) = 1.$
- Triangular or Bartlett’s window.  $h(n) = 1 - \frac{2|n - \frac{M-1}{2}|}{M-1}.$
- Blackman’s window.  $h(n) = 0.42 - 0.5cos(\frac{2\pi n}{M-1}) + 0.08cos(\frac{4\pi n}{M-1}).$
- Hamming’s window.  $h(n) = \frac{1}{2}(1 - cos(\frac{2\pi n}{M-1})).$
- Hanning’s window.  $h(n) = 0.54 - 0.46cos(\frac{2\pi n}{M-1}).$



- Kaiser's window. 
$$h(n) = \frac{I_0 \left[ \alpha \sqrt{\left(\frac{M-1}{2}\right)^2 - \left(n - \frac{M-1}{2}\right)^2} \right]}{I_0 \left[ \alpha \left(\frac{M-1}{2}\right) \right]}.$$
- Tukey's window. 
$$h(n) = \frac{1}{2} \left[ 1 + \cos \left( \frac{n - (1+\alpha)(M-1)/2}{(1-\alpha)(M-1)/2} \pi \right) \right]$$
  

$$\frac{\alpha(M-1)}{2} \leq \left| n - \frac{M-1}{2} \right| \leq \frac{M-1}{2}.$$
- Time domain sequence: 
$$h(n), 0 \leq n \leq M-1.^1$$

With the use of a window which gets good separability between mental patterns, the classifier will be easier, faster and the results more reliable.

#### 2.5.4 FFT

The cerebral activity becomes apparent mainly through the frequency components of the electroencephalographic signal. Different kind of mental activities have different frequency components, (Harris, 1978), (Neuper, C.; et al., 2001), (Penny, W. D.; et al., 2000). For this reason it is necessary to transform the sampled time domain signal to frequency domain, so a Fast Fourier Transform is applied to each block of  $2^7$  sampled data.

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad 0 \leq k \leq N-1 \quad (4)$$

$$W_N = e^{-\frac{2\pi j}{N}} \quad (5)$$

Having in mind that the sample frequency is 384Hz, the frequency resolution is:

$$\Delta f = \frac{384\text{Hz}}{128} = 3\text{Hz}. \quad (6)$$

In this application the useful information is in the amplitude of the frequency components, so the phases are discarded, we focus our attention on the spectrograms of each of the analysis windows. Considering the properties of the Fourier Transform and that the signal in the time domain only have real components, in the Nyquist frequency is produced the reflection effect, so the signal information is in the first half of the components, (Harris, 1978).

#### 2.5.5 Feature selection

A vector of six features is extracted from each signal analysis window. This vector, table 1, is made up as the mean of the amplitudes of the frequency bands. Because the frequency of normal human brain is under 40-50Hz, only frequencies between 6 and 38Hz have been considered.

### 3. Statistical analysis procedure

In order to assess if it is possible to discriminate between the samples acquired when the user was performing the proposed cognitive tasks, the statistical technique of bilateral contrast test is applied to each population pair of features obtained from each cognitive activity. Each component of the vector is considered to determine its significance and separability power. Bilateral contrast makes use of population variance, if the equality of both population variances is rejected it is necessary to apply a correction factor in the degrees of freedom. These

<sup>1</sup>  $M$  = length of the filtering window

FFT index.	Frequency.	Denomination.
1 - 2	0 - 5	Not considered
3	6 - 8	$\theta$ .
4	9 - 11	$\alpha_1$ .
5	12 - 14	$\alpha_2$ .
6 - 7	15 - 20	$\beta_1$ .
8 - 10	21 - 29	$\beta_2$ .
11 - 13	30 - 38	$\beta_3$ .
14 - 64	39 - 192	Not considered

Table 1. Feature vector.

contrasts were applied to samples of both electroencephalographic channels preprocessed with each type of filtering window.

- *Bilateral contrast to the variance ratio.* The equality of variances is obtained with  $R = 1$ .  
 $n_1$  : sample size of the first population.  
 $n_2$  : sample size of the second population.  
 $\sigma_1$  : variance of the first population.  
 $\sigma_2$  : variance of the second population.  
 $\hat{S}_1$  : variance estimation of the first population.  
 $\hat{S}_2$  : variance estimation of the second population.  
 $F$  = Fisher distribution.  
 $T$  = Student distribution.  
Null hypothesis  $H_o$  vs. alternative hypothesis  $H_1$ .

$$H_o : \frac{\sigma_1}{\sigma_2} = R \quad vs. \quad H_1 : \frac{\sigma_1}{\sigma_2} \neq R \tag{7}$$

Considering that:

$$\frac{(n_1 - 1)\hat{S}_1}{\sigma_1} \rightsquigarrow \chi^2_{n_1-1} \qquad \frac{(n_2 - 1)\hat{S}_2}{\sigma_2} \rightsquigarrow \chi^2_{n_2-1} \tag{8}$$

$$\frac{\frac{1}{n_1-1} \frac{(n_1-1)\hat{S}_1}{\sigma_1}}{\frac{1}{n_2-1} \frac{(n_2-1)\hat{S}_2}{\sigma_2}} = \frac{\sigma_2}{\sigma_1} \frac{\hat{S}_1}{\hat{S}_2} \rightsquigarrow F_{n_1-1,n_2-1} \tag{9}$$

Therefore under the fulfillment of the null hypothesis:

$$F_{Exp} = \frac{1}{R} \frac{\hat{S}_1}{\hat{S}_2} \rightsquigarrow F_{n_1-1,n_2-1} \tag{10}$$

The acceptance zone of  $H_o$  is:

$$a_{teo} = F_{(n_1-1,n_2-1,1-\frac{\alpha}{2})} \tag{11}$$

$$b_{teo} = F_{(n_1-1,n_2-1,1-\frac{\alpha}{2})} \tag{12}$$

$$a_{teo} \leq F_{Exp} \leq b_{teo} \tag{13}$$

- *Bilateral contrast of two independent normal and homocedastic populations.* Null hypothesis  $H_o$  vs. alternative hypothesis  $H_1$ .  
$$H_o : \mu_1 - \mu_2 = \Delta \quad vs. \quad H_1 : \mu_1 - \mu_2 \neq \Delta \tag{14}$$

The variances of the both population are equal but unknown.

$$T_{Exp} = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\hat{S}(\frac{1}{n_1} + \frac{1}{n_2})}} \quad (15)$$

In which  $\hat{S}$  is the pseudo-variance of  $\hat{S}_1$  and  $\hat{S}_2$

$$\hat{S} = \frac{(n_1 - 1) * \hat{S}_1 + (n_2 - 1) * \hat{S}_2}{n_1 + n_2 - 2} \quad (16)$$

The zone of  $H_o$  acceptance is:  $T_{Teo} = t_{(n_1+n_2-2, 1-\frac{\alpha}{2})}$  (17)

If  $|T_{Exp}| \leq T_{Teo}$  then  $H_o$  is accepted, on the contrary  $H_1$  is accepted and  $H_o$  is rejected.

- *Bilateral contrast of two independent normal and heterocedastic populations.* The null hypothesis  $H_o$  and alternative hypothesis are similar to the previous ones, the statistical measure is:

$$T_{Exp} = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{\hat{S}_1}{n_1} + \frac{\hat{S}_2}{n_2}}} \rightsquigarrow t_f \quad (18)$$

In which  $f$  is the number of degrees of freedom calculated with the Welch's formula:

$$f = \frac{(\frac{\hat{S}_1}{n_1} + \frac{\hat{S}_2}{n_2})^2}{\frac{1}{n_1+1}(\frac{\hat{S}_1}{n_1})^2 + \frac{1}{n_2+1}(\frac{\hat{S}_2}{n_2})^2} - 2 \quad (19)$$

In this case the zone of  $H_o$  acceptance is:

$$T_{Teo} = t_{(f, 1-\frac{\alpha}{2})} \quad (20)$$

If  $|T_{Exp}| \leq T_{Teo}$  then  $H_o$  is accepted, on the contrary it is assumed that the populations are different.

The results of these analyses are graphically shown in the subsection 6.2.1.

#### 4. Reduction of the feature space dimensionality

Linear Discriminant Analysis is a preprocessing technique used in machine learning, its objective is to find the best combination of features that separate two or more types of objects or events. The result can be used as linear classifier or as a technique to reduce the feature space dimension before the classification process.

Under the consideration that it is possible to discriminate between electroencephalographic samples acquired when the user was performing the suggested cognitive tasks, the next phase is to find the best combination of features that separates in an optimal way the registers of these mental tasks.

In machine learning a preprocessing technique called Linear Discriminant Analysis finds automatically this combination of features. The result can be used as linear classifier or as a technique to reduce the feature space dimension before the classification process.

##### 4.1 Linear discriminant analysis

Supposed  $C$  classes of observations, Linear Discriminant Analysis is a preprocess technique that finds the transformation matrix  $W$  which separates in an optimal way two or more classes. LDA considers maximising the following objective:

$$J(W) = \frac{W^T S_B W}{W^T S_W W} \tag{21}$$

where  $S_B$  is the between classes scatter matrix, and  $S_w$  is the within classes scatter matrix, the definitions of the both matrices are:

$$S_B = \sum_c N_c (\mu_c - \bar{x})(\mu_c - \bar{x})^T \tag{22}$$

$$S_W = \sum_c \sum_{i \in c} (x_i - \mu_c)(x_i - \mu_c)^T \tag{23}$$

$$\mu_c = \frac{1}{N_c} \sum_{i \in c} x_i \tag{24}$$

$$\bar{x} = \frac{1}{N} \sum_i x_i = \frac{1}{N} \sum_c N_c \mu_c \tag{25}$$

and  $N_c$  is the number of samples in class  $c$ .  
Because  $J$  is invariant to rescaling of the vectors  $W \rightarrow \alpha W$ , hence it is possible to choose  $W$  such that the denominator is  $W^T S_W W = 1$ . So the problem of maximising  $J$  can be transformed to the following constrained optimisation problem,

$$\min_W \quad -\frac{1}{2} W^T S_B W \tag{26}$$

$$s.t. \quad W^T S_W W = 1 \tag{27}$$

corresponding to the Lagrangian,

$$L_P = -\frac{1}{2} W^T S_B W + \frac{1}{2} \lambda (W^T S_W W - 1) \tag{28}$$

With solution (the halves are added for convenience):

$$S_B W = \lambda S_W W \Rightarrow S_W^{-1} S_B W = \lambda W \tag{29}$$

This is a generalised eigen-problem, and using the fact that  $S_B$  is symmetric positive definite and can hence be written as  $S_B^{\frac{1}{2}} S_B^{\frac{1}{2}}$ , where  $S_B^{\frac{1}{2}}$  is constructed from its eigenvalue decomposition as  $S_B = U \Lambda U^T \rightarrow S_B^{\frac{1}{2}} = U \Lambda^{\frac{1}{2}} U^T$ . Defining  $V = S_B^{\frac{1}{2}} W$  it is get

$$S_B^{\frac{1}{2}} S_W^{-1} S_B^{\frac{1}{2}} V = \lambda V \tag{30}$$

this is a regular eigenvalue problem for a symmetric positive definite matrix, with solutions  $\lambda_k$  as eigen-values and  $V_k$  as eigen-vectors, which leads to solution:

$$W = S_B^{-\frac{1}{2}} V \tag{31}$$

Plugging the solution back into the objective  $J(W)$ , it is found that the desired solution which maximise the objective is the one with largest eigenvalues.

## 4.2 Operational procedure

1. Samples from each mental tasks are obtained.

$X_a$  Mathematical Activity.  
 $X_b$  Movement imagination.  
 $X_c$  Relax.

2. Statistical definition of all populations:

$$\bar{\mu}_a = E[x_a] \quad S_a = E[(x_a - \bar{\mu}_a)(x_a - \bar{\mu}_a)^T] \quad (32)$$

$$\bar{\mu}_b = E[x_b] \quad S_b = E[(x_b - \bar{\mu}_b)(x_b - \bar{\mu}_b)^T] \quad (33)$$

$$\bar{\mu}_c = E[x_c] \quad S_c = E[(x_c - \bar{\mu}_c)(x_c - \bar{\mu}_c)^T] \quad (34)$$

3. Calculation of the scattering matrices (eq.22 & 3).
4. Application of LDA optimising criterion (eq.30).
5. Calculation of the transformation matrix,  $W$  (eq.31), formed by the eigen-vectors,  $V_k$ , which eigen-values are bigger than  $1 * 10^{-4}$  ordered from high to low magnitudes.
6. Once the transformation matrices have been obtained, the data sets are transformed using LDA transform. The decision region in the transformed space is a hyperplane of lower dimension than the feature space.

$$X_a \Rightarrow X'_a = W^T * X_a \quad (35)$$

$$X_b \Rightarrow X'_b = W^T * X_b \quad (36)$$

$$X_c \Rightarrow X'_c = W^T * X_c \quad (37)$$

7. For classification problems once the LDA transformations are completed, Euclidean or Mahalanobis distances to the centre of each class could be used to classify new vectors. The smallest value among the  $c$  distances classifies the new vector as belonging to class  $c$ .

In order to evaluate the feedback effect over the discrimination capability the Off-line and On-line experimental procedures described respectively on subsections 2.1 and 2.2 were carried out on five healthy male subjects obtaining the results shown in subsection 6.2.2

## 5. Classifier description

This section briefly presents the different types of classifiers used in the experimental procedures.

### 5.1 Neural networks classifiers for BCI devices

Once that the discrimination capability of the electroencephalographic signals has been assessed and analysed the possibility for the reduction of the original feature space without affecting the discrimination capability, the next step is the application of different families of supervised classifiers to the electroencephalographic signal and analysing the results.

One of these family of classifiers is based on different types of artificial neural networks. This section describes the architecture of three types of classifiers based on: Radial Basis Functions (RBF), Probabilistic Neural Networks (PNN), and Multi-Layer Perceptrons (MLP) (Bishop, 1995), (Ripley, 1996).

For each type of neural network two architectures of classifiers were implemented (refer to Figure 7).

Each classifier applies the following procedure to the vector of features extracted previously:

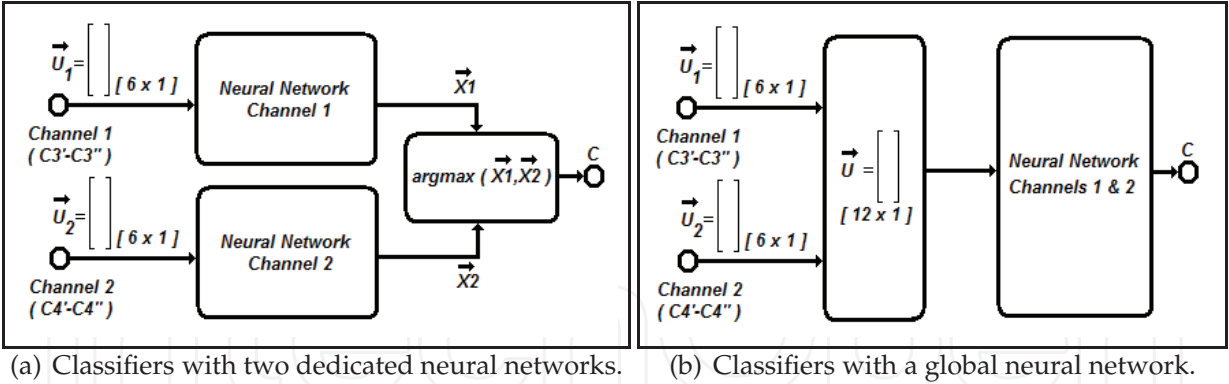


Fig. 7. Architecture of classifiers.

1. Determination of the learning (50%), test (25%) and validation (25%) data sets.
2. Calculation of the normalisation matrix for the learning data set.
3. Application of Principal Component Analysis to the learning data set in order to reduce the dimensionality of the data input space.
4. Learning of the input data set by the neural network.
5. Application of the neural network to the test data set. If the error test is less than the goal error ( $1e^{-5}$ ), then the learning process is stopped. Otherwise, the network is trained again.
6. Estimation of the network performance error.
7. Application of the neural net to the whole data set and result registration.
8. Calculation of the confusion matrices for each experiment.

5.1.1 Multi-Layer Perceptron Classifier

The setup parameters used in this classifier are:

Parameter	Value
Learning algorithm	Levenberg-Marquardt (Backpropagation)
Number of output neurons	3
Goal error	$1e^{-5}$
Epochs	400
Max. fail	5
Mem. reduc.	1
Min. grad.	$1e^{-10}$
$\mu$	$1e^{-3}$
$\mu_{dec}$	0.1
$\mu_{inc}$	10
$\mu_{max}$	$1e^{-5}$

Table 2. Parameters for MLP Classifiers.



### 5.1.2 Radial Basis Function Classifier

The setup parameters used in this classifier are:

- Number of hidden neurons: The learning algorithm used by this type of neural networks determines the number of neurons that are in the hidden layer through an iterative process (Horward Demuth, 2006). That is, it starts with a reduced number of hidden neurons, which are increased as long as the goal error is not achieved or a maximum number of neurons is reached.
- Spread constant : 0.25 (Determine the zone of influence of each neuron).

$$a = e^{-\left(\frac{\|w-p\|^2}{b}\right)} \quad (38)$$

In which:

- a: Output of the neuron.
- w: Weight vector.
- p: Input vector.
- b: Spread constant.
- Number of output neurons : 3. One for each cognitive activity.

### 5.1.3 Probabilistic Neural Network Classifier

The setup parameters used in this classifier are:

- Number of hidden neurons: The learning algorithm used as many hidden neurons as pairs of input vector - target vectors were in the learning data set.
- Spread constant : 0.25 (Determines the zone of influence of each neuron, same expression as eq.38).
- Number of output neurons : 3. One for each cognitive activity.

## 5.2 Adaptive bi-stage classifier based on RBF-HMM

In this section it is described an adaptive bi-stage classifier. The first stage is based on Radial Basis Function neural networks, which provides sequences of pre-assignments to the second stage, that it is based on three different Hidden Markov Models, each one trained with pre-assignment sequences from the cognitive activities between classifying. The segment of EEG signal is assigned to the HMM with the highest probability of generating the pre-assignment sequence.

The algorithm is tested with real samples of electroencephalographic signal, from five healthy volunteers using the cross-validation method. The results allow to conclude that it is possible to implement this algorithm in an on-line BCI device. The results also shown the huge dependency of the percentage of the correct classification from the user and the setup parameters of the classifier.

### 5.2.1 Introduction.

In Figure 8 is shown the block diagram of the algorithm for the proposed classifier.

In it can be appreciated how the classification of the considered segment of the EEG signal is obtained after the evaluation of the probability generation of the pre-assignment sequence provided by three Hidden Markov Models.

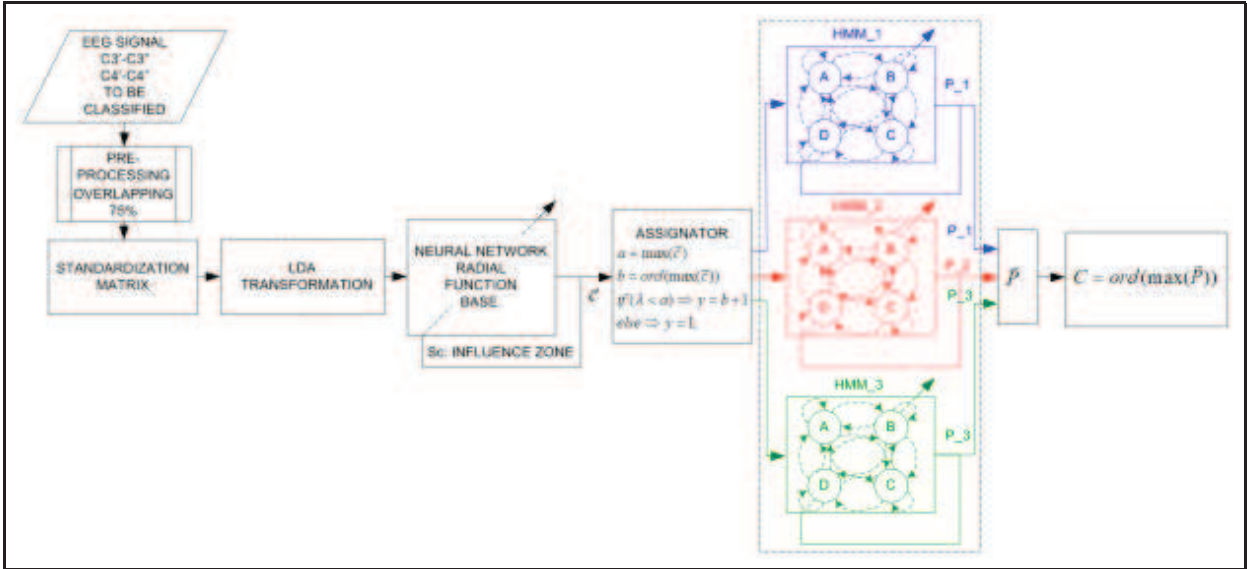


Fig. 8. Block diagram of the classifier.

There are as many Hidden Markov Models as cognitive activities to be considered for the classification, each model is trained with pre-assignment sequences of data of the cognitive activity associated to it.

The pre-assignment sequence of data are provided by a neural network, which inputs are the vectors of features obtained after the preprocessing of the segment of EEG signal, as it is described in the following subsections.

5.2.2 Training of the neural network

The considered neural network is the type of Radial Function Basis. This type of neural network is characterised by the learning of the position of the samples in the training set and by the interpolation capability between them (Bishop, 1995).

In Figure 9 is represented the architecture of this type of neural network.

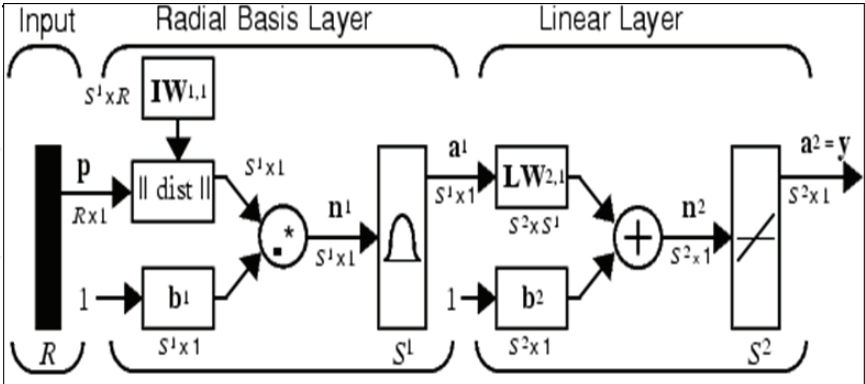


Fig. 9. Architecture of the RBF neural network.

From previous studies it has been concluded that this type of neural network behaves better than other types of neural networks, as for example Multi-Layer Perceptrons or Probabilistic Neural Networks (Martinez, J.L.; Barrientos, A., 2008).

The activation function is:

$$radbas(x) = e^{-x^2}; \quad x = (\vec{w} - \vec{p}) * S_c$$

(39)

In where  $\vec{w}$  and  $S_c$  are respectively the weights and influence zone constant of each neuron, and  $\vec{p}$  is the position of the considered sample.

During the learning phase the neurons of the hidden layer learn the position of the samples of the learning set,  $\vec{w}$ ; during the test phase when a new sample  $\vec{p}$  is presented, it is computed the distance between the sample and the learned positions, the nearest neurons to the sample will proportionate higher activation values than the rest of the neurons.

For the learning process are considered vectors of features from the EEG signal, acquired when the user was performing one of the different cognitive activities considered for the classification. The learning set is composed by the 75% of all the sample set, and the other 25% is considered for validation. After the determination of the learning and validation sets, the input vectors to the neural network are normalised, and with LDA technique is reduced their dimensionality projecting the original input vectors in the direction of the highest discrimination capability (Martinez, J.L.; Barrientos, A., 2007).

In order to minimise the over-learning effect, the RBF learning process allows a dynamic growth of the number of neurons in the hidden layer. In the output layer are considered as many linear neurons as cognitive activities between discriminate. Finally in the assignation block on Figures 8, it is weighted the output vector of the neural network and it assigns the input vector to the activity with highest output value provided it is higher than a threshold  $\lambda$ , on the contrary if the value is lower than  $\lambda$ , the input vector is labelled as unclassified.

On operation, once the neuronal network has been trained, when a new vector is presented, the cognitive activity with samples nearer to it will provide a higher activation level, and the corresponding output will have a higher value than the others cognitive activities.

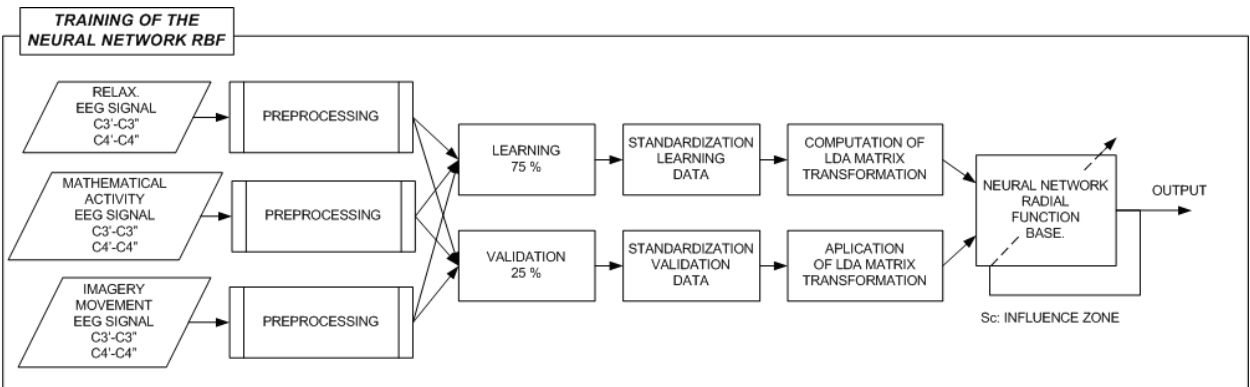


Fig. 10. Training of the RBF neural network.

5.2.3 Description of Hidden Markov Models

A Hidden Markov Model is a double stochastic statistical model, it consists of a Markov process with unknown and non-observable parameters, and a observed model which parameters depend stochastically from the hidden states. A stochastic process is called a Markovian process if the future does not depend from the past, only from the known present; considering the stochastic variable  $q(t - 1)$  the transition probability in the instant  $t$  is defined as  $P(q_t = \sigma_t | q_{t-1} = \sigma_{t-1})$ . A Markov chain is formally defined with the pair  $(Q, A)$ , where  $Q = \{1, 2, \dots, N\}$  are the possible estates of the chain and  $A = [a_{ij}]$  is the transition matrix of the model, with the constrains:

$$0 \leq a_{ij} \leq 1; \quad 1 \leq i, j \leq N \tag{40}$$

$$\sum_{j=1}^N a_{ij} = 1; \quad 1 \leq i \leq N \tag{41}$$

The transition and emission probabilities depends from the actual estate and no from the former estates.

$$\begin{aligned} P(q_t = j | q_{t-1} = i, q_{t-2} = k, \dots) &= \\ &= P(q_t = j | q_{t-1} = i) = a_{ij}(t) \end{aligned} \tag{42}$$

Formally a discrete HMM of first grade is defined by the 5-tuple:  $\lambda = \{Z, Q, A, B, \pi\}$ , in where:

- $Z = \{V_1, V_2, \dots, V_M\}$  is the alphabet or discrete set of  $M$  symbols.
- $Q = \{1, 2, \dots, N\}$  is the set of  $N$  finite estates.
- $A = [a_{ij}]$  is the transition matrix of the model.
- $B = (b_j(Q_t))_{N \times M}$  is the matrix of emission symbols, also known as observation matrix.
- $\pi = (\pi_1, \pi_2, \dots, \pi_N)$  is the prior probability vector of the initial estate.

The parameters of a HMM are  $\lambda = \{A, B, \pi\}$ . There are three types of canonic problems associated to HMM (Rabiner, 1989)(Rabiner & Juang, 1986):

1. Given the parameters of the model, obtain the probability of a particular output sequence. This problem is solved through a forward-backwards algorithm.
2. Given the parameters of the model, find the most probable sequence of hidden estates, that could generate the given output sequence. This problem is solved through the use of Viterbi algorithm.
3. Given an output sequence, find the parameters of the HMM. This problem is solved through the use of Baum-Welch algorithm.

The HMM have been applied specially in speech recognition an generally in the recognition of temporal sequences, hand written, gestures, and bioinformatics (Rabiner & Juang, 1986).

5.2.4 Training of the Hidden Markov Models

The HMM's are trained with sequences of pre-assignments coming from the EEG samples, as it is shown in the Figure 11.

For each cognitive activity a particular HMM, with the following characteristics, is trained:

- Number of hidden estates: 4.
- Number of different observable objects: 4

In the training phase, chains of nine pre-assignments were used. In a previous experiment with synthetic samples, it was concluded that for the proposed architecture of Hidden Markov Models the highest percentage of correct classifications were obtained with chains of nine elements.

After the training or solution of the third canonic problem, the probability matrices of state transitions and observation matrices are determined. The Viterbi algorithm is used in order to determine the probability that a model generates the proposed sequence.

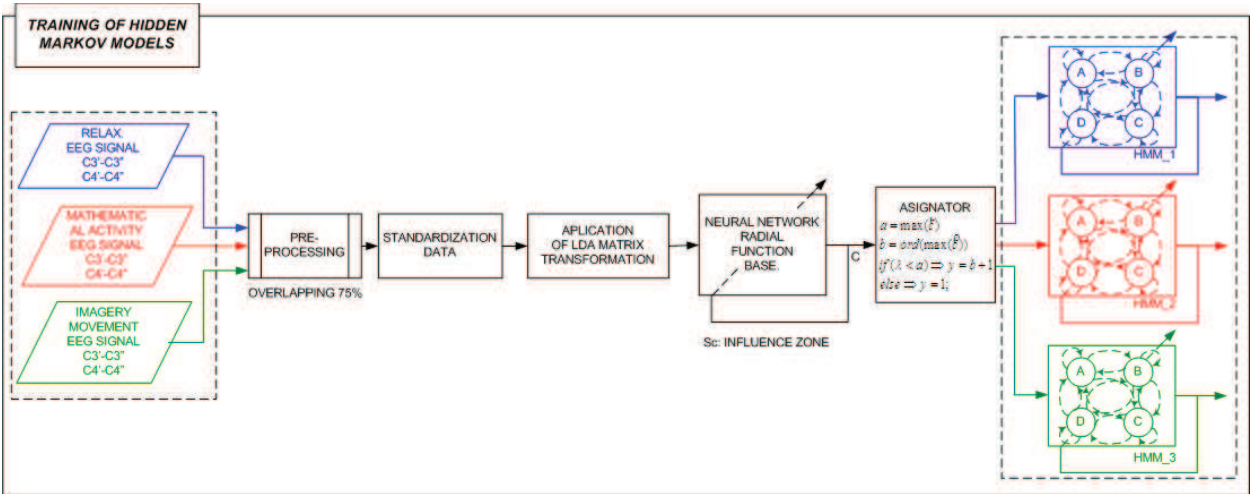


Fig. 11. Training of the HMM.

5.3 Classifier based on Support Vector Machines

Under the denomination of Support Vector Machines are known a set of supervised learning methods that belong to the generalised linear classifiers with applicability into classification and regression problems.

It structure is based on a net of static kernels operating over feature vectors which have been transformed to a space with higher dimension than the original feature space, see Figure 12. The main property of the SVMs is its good generalisation capability founded on the determination of a hiperplane with maximum separation distance between the transformed vector of each class. This separation distance is the one between to hiperplanes parallel to the optimum separation hiperplane containing at least one transformed vector called support vector. It is assumed that as bigger is this distance, bigger is the generalisation capability.

The operations performed by a SVM classifier are:

- Transformation of the sample data or input feature vectors to a higher dimension space through the application of the kernel function  $\phi$ . The objective is to formulate the classification problem using the kernel function.
- Obtaining of the optimum hiperplane which maximises the distance between the considered classes. If the input vectors are lineally separable, the optimum hiperplane besides the maximisation of the separability, minimises the penalty function that considers the incorrect classifications.

6. Description of experimental procedures and results

6.1 Esperimental procedures

6.1.1 LDA procedure

The Figure 13 represents the activity diagram associated to the experimental procedure used with the Linear Discriminant Analysis technique.

The experimental procedure is performed with the feature vectors obtained after processing the samples of electroencephalogram activity with each type of preprocessing window.

In order to assess the discriminant power of each type of preprocessing window a bilateral contrast test is performed with the transformed populations of feature vectors.

The results are graphically represented in subsection 6.2.2.



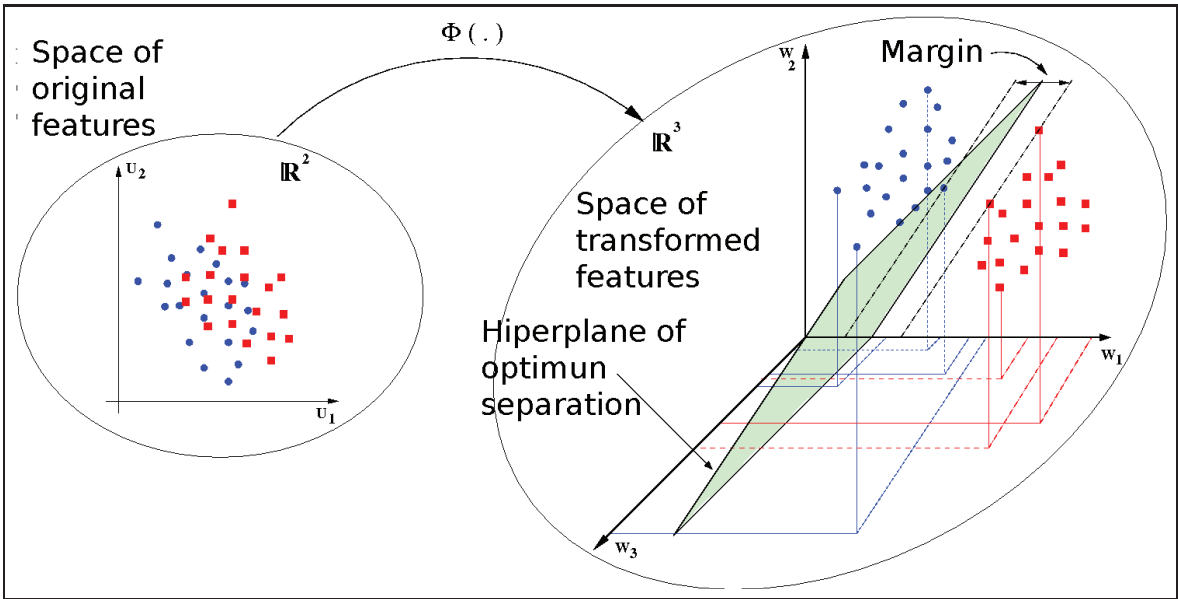


Fig. 12. Operacional description of Support Vector Machines.

6.1.2 Procedure for classifiers based on Artificial Neural Networks

The Figure 14 represents the activity diagram associated to the experimental procedure used with the Artificial Neural Networks classifiers.

- The first stage loads the registers sampled when the user performed the different mental tasks and associates them to each proposed cognitive activity. After this the data sets are normalised and reduced their dimension through LDA.
- In the second stage the learning data sets are defined; 50% of the samples are used for the learning data set, 25% for the validation data set, and the other 25% for the testing data set.
- In the third stage the classifiers are created, trained, validated and tested using respectively the previous data sets.
- In the fourth and last stage the confusion matrices are obtained and saved.

The results are graphically represented in subsection 6.2.3.

6.1.3 Procedure for RBF-HMM bi-stage classifier

The Figure 15 represents the activity diagram for the experimental procedure used with the RBF-HMM classifier. It is composed by four different blocks:

- The first block generates the different data sets for learning and testing, considering the three different cognitive tasks. The cross validation procedure is used for the results, ten different repetitions of cross validation are considered, in each repetition a different data set session is reserved for the validation, employing the rest data sets for learning and testing.
- In the second block the pre-classifier based on RBF is trained.
- In the third block three different Hidden Markov Models are trained, one for each cognitive activity, considering pre-assignation sequences of nine elements.
- Finally in the fourth block the validation procedure is performed and the results saved.

The results are graphically represented in subsection 6.2.4.



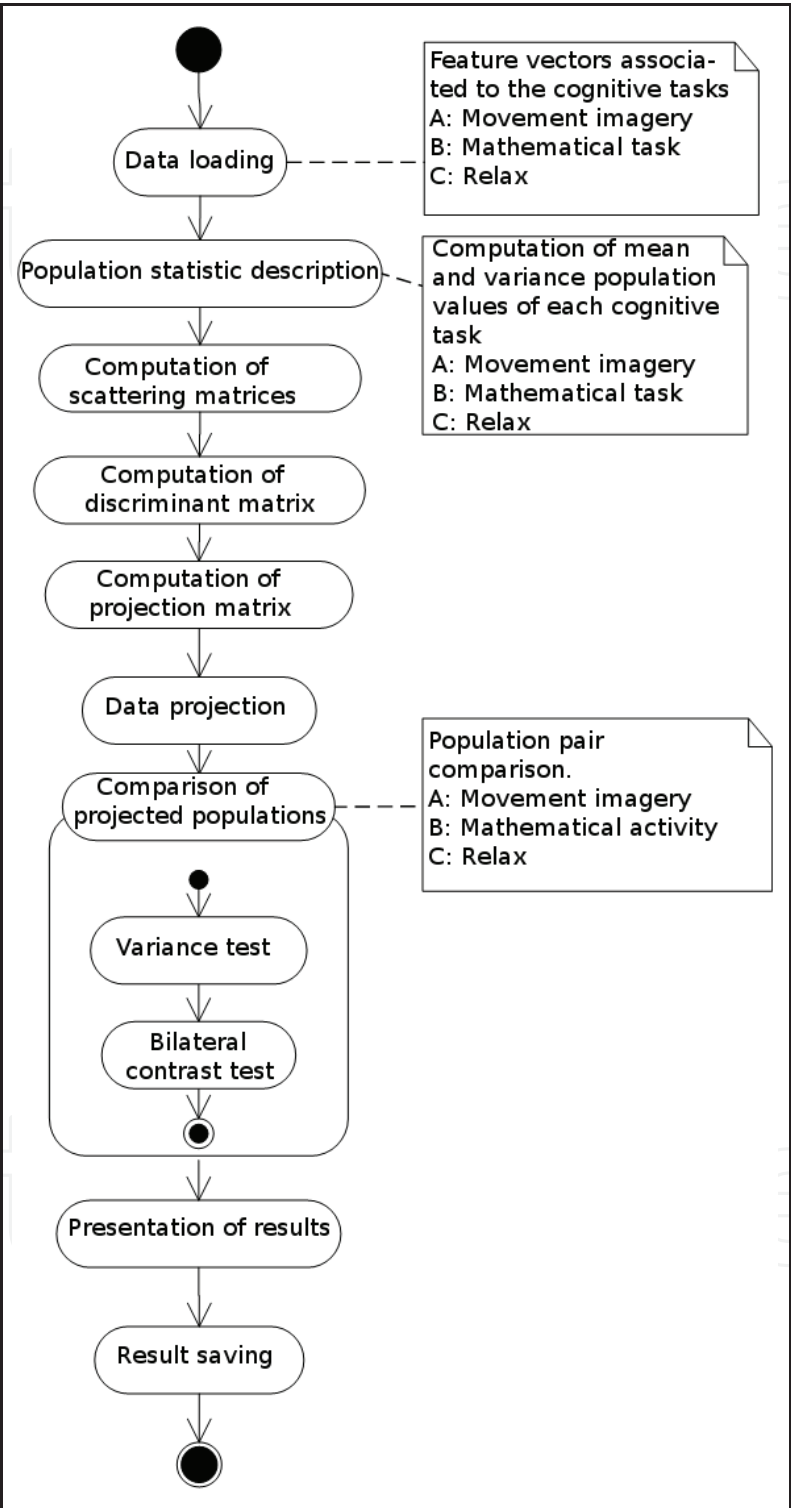


Fig. 13. Experimental procedure for LDA.

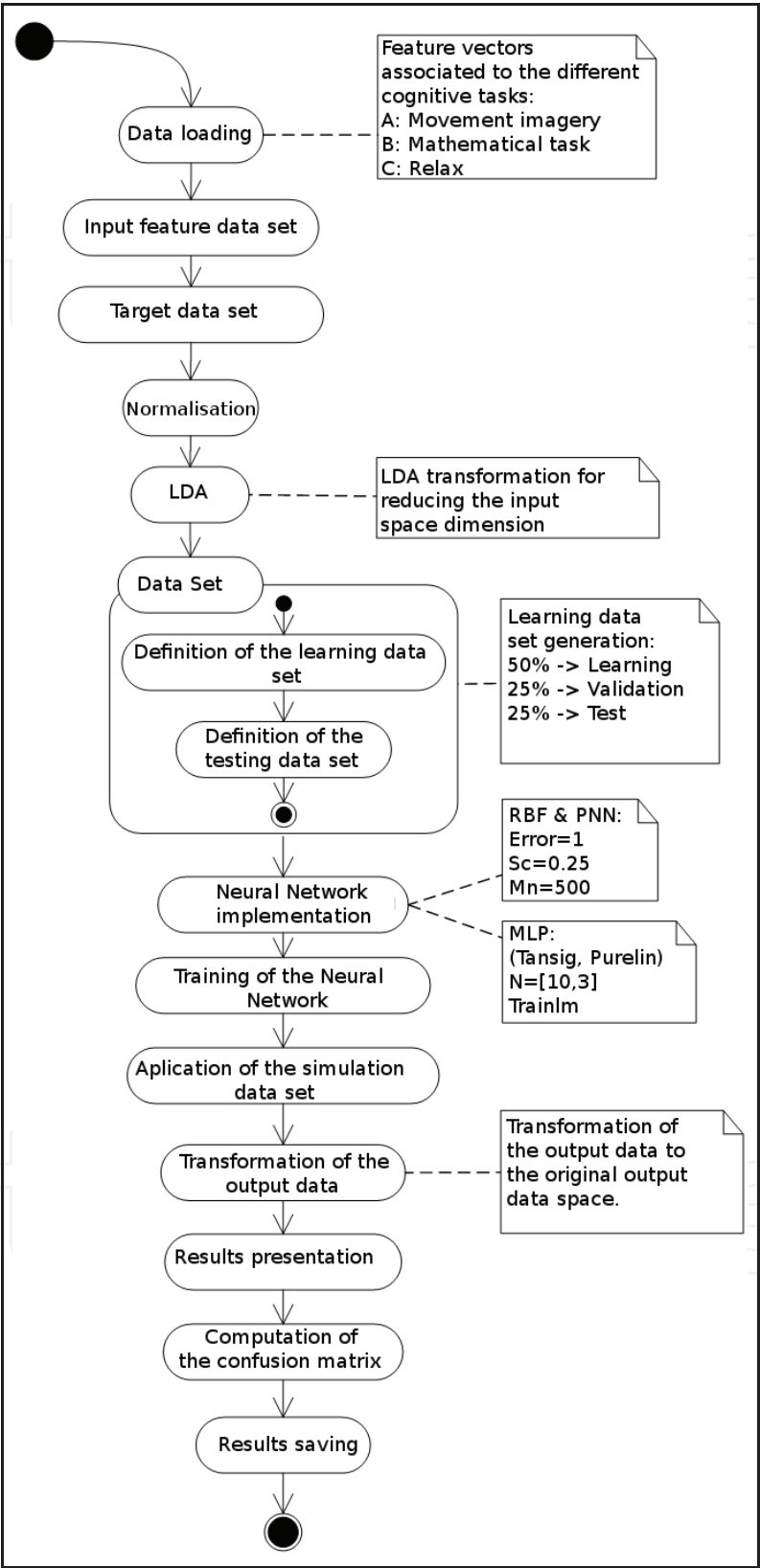


Fig. 14. Experimental procedure for ANN classifiers.

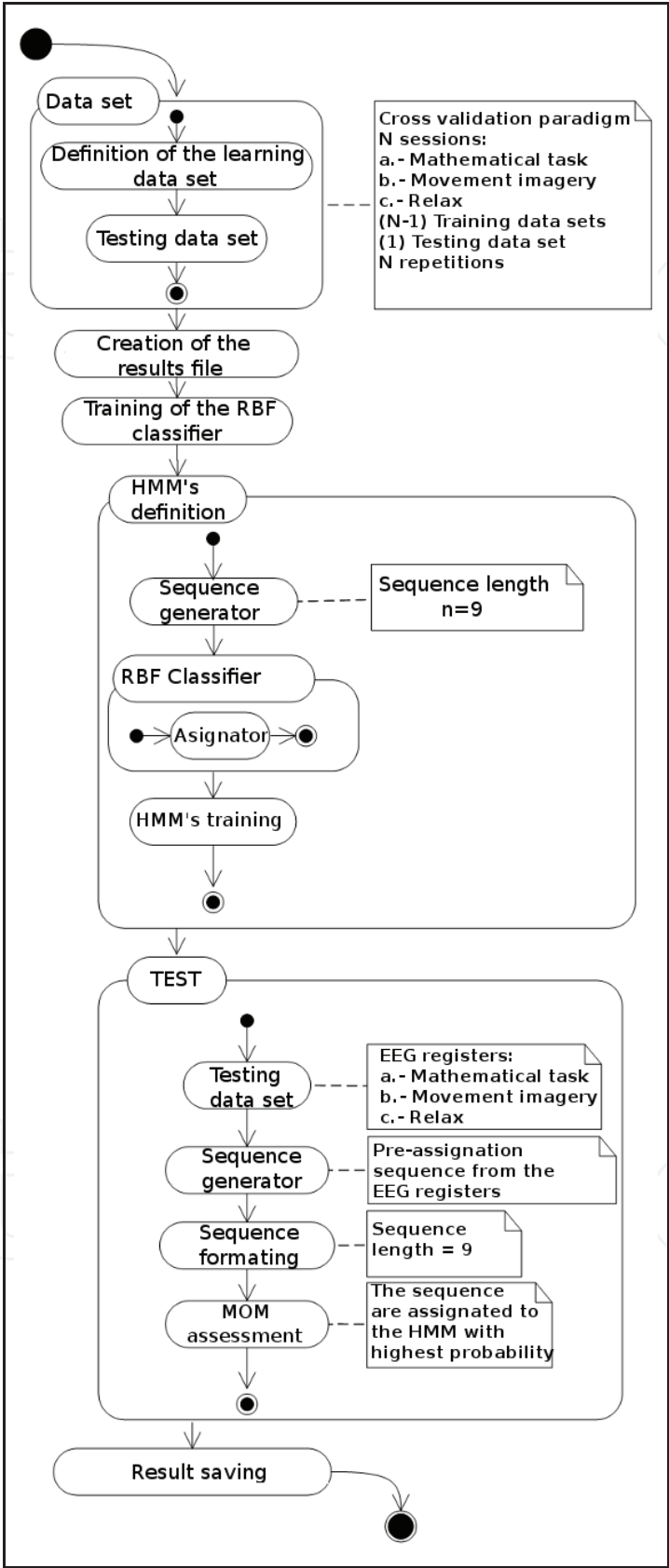


Fig. 15. Experimental procedure for RBF-HMM classifier.

6.1.4 SVM procedure

The Figure 16 represents the experimental procedure used with the SVM classifiers. In the first stage the data sets of each cognitive activity are loaded. In the second stage the SVM classifiers are created with the different kernel parameters, the training and testing data sets are defined, and the classifiers are trained considering three subclassifiers under the one against one classification paradigm. Finally in the last stage a classification test is performed and the results saved. The results are graphically represented in subsection 6.2.5.

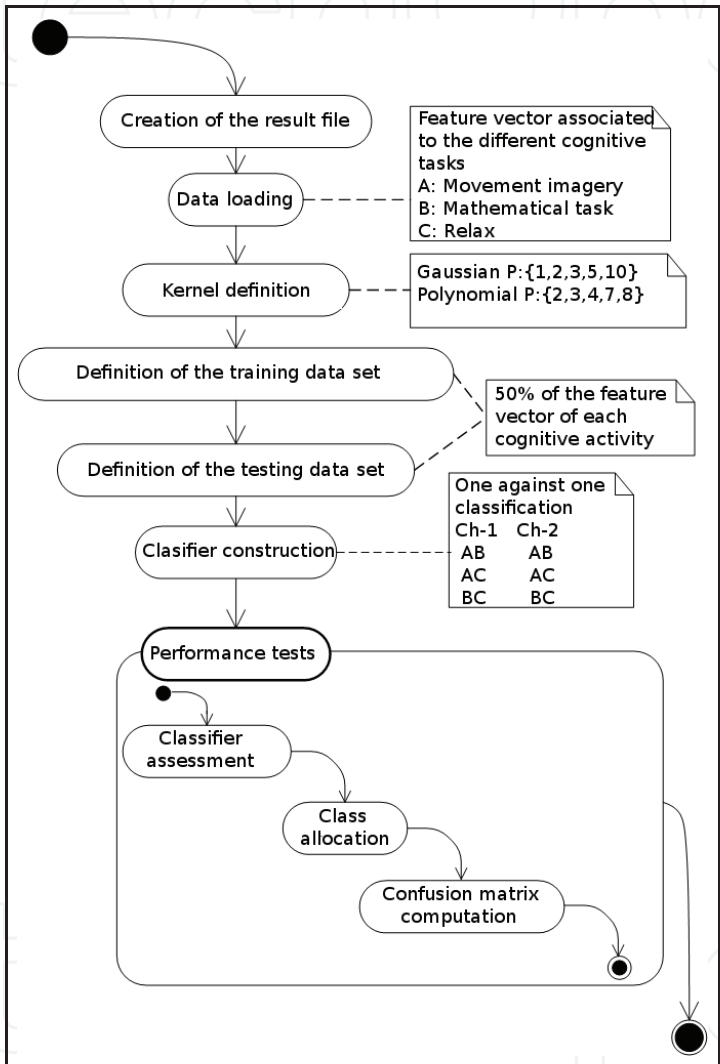


Fig. 16. Experimental procedure for SVM classifiers.

6.2 Results

6.2.1 Results of the statistical analysis procedure

The following figures summarise the results of the former tests. The contrasts between mental activities are shown on the horizontal axis. The Figure 17 shows the results of the contrast tests between the cognitive tasks for channel one:  $C3' - C3''$ , meanwhile the Figure 18 shows the results for channel two:  $C4' - C4''$ .

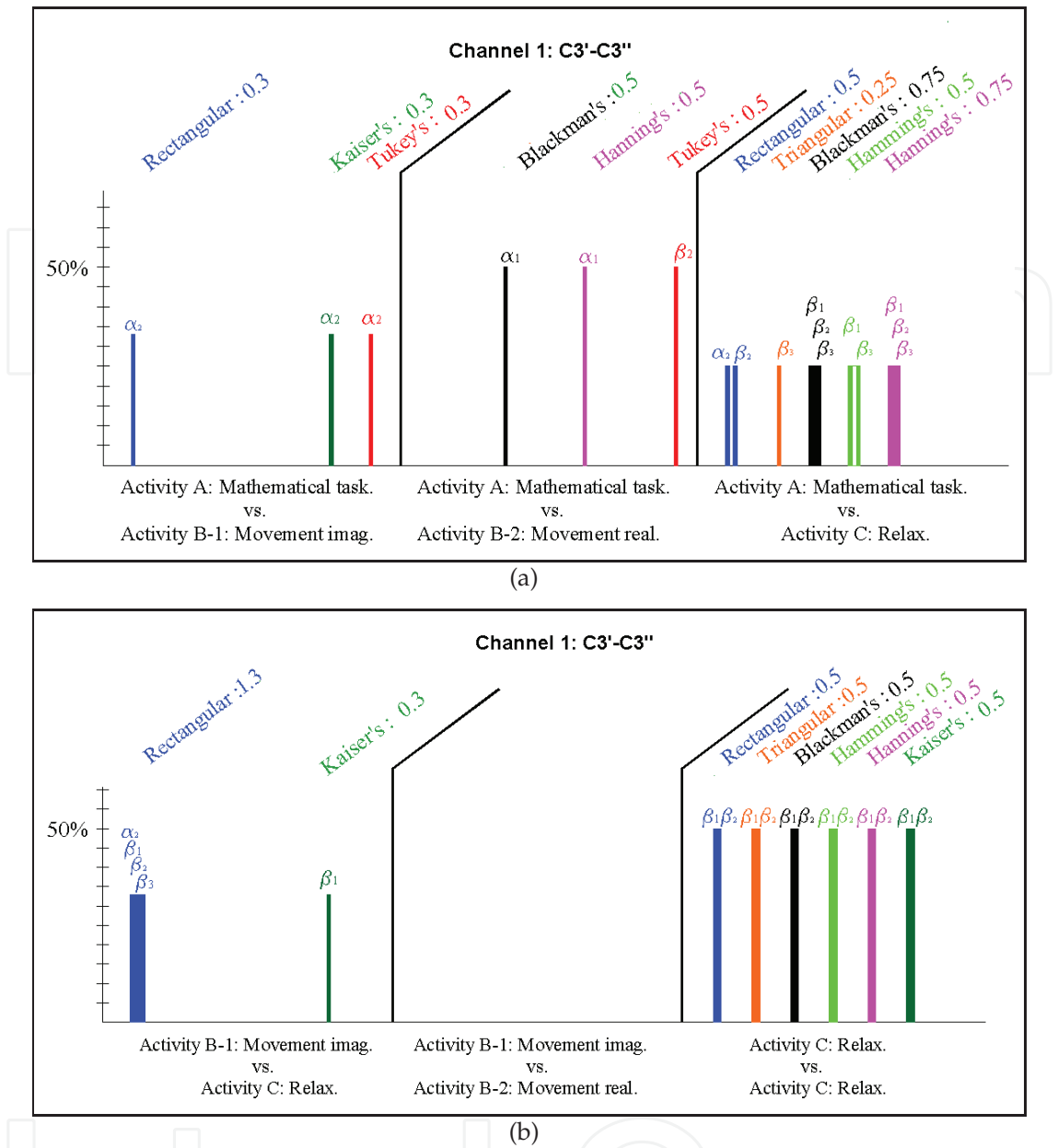


Fig. 17. Results channel 1.

All the seven types of windows have been applied to each comparison.<sup>2</sup> In the top of each figure appears both the type of window and a number. This number indicates the average of significant features obtained with this window, it is the total of the features that shown statistical evidence of difference,  $p < 0.05$ , divided by the number of times the experiment has been replicated.

Finally in the bars are the significant features for each kind of window<sup>3</sup>, in the vertical axis is the percentage of times that this feature has been significant.

Making a comparison between **mathematical activity** and **movement imagination** the result is, that among all windows, the Tukey's and Kaiser's windows are the ones with more

<sup>2</sup> See sections 2.5.3 and 3.

<sup>3</sup> See section 2.5.5.

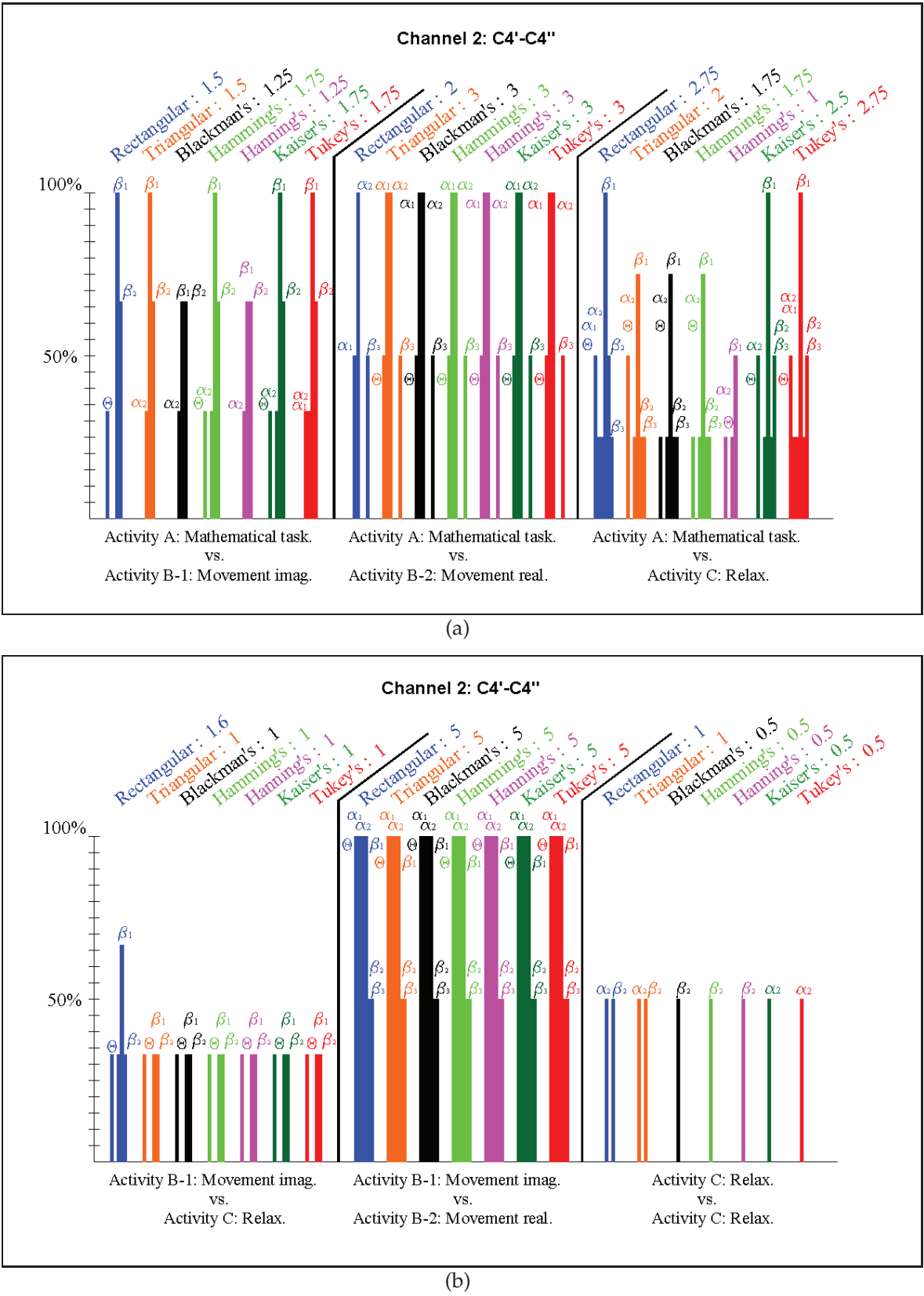


Fig. 18. Results channel 1.



significant features. Among features, the most significant are  $\beta_1$  and  $\beta_2$  for all kind of windows.

In the comparison between **mathematical activity** and **movement realization** the result is that the most significant windows are Hanning's, Tukey's and Blackman's, and the most significant features are  $\alpha_1$  and  $\alpha_2$ .

Taking **mathematical activity** versus **relax**, the result is that the rectangular and Tukey's windows are the most significant. The component of the feature vector with more discriminating power is  $\beta_1$  followed by  $\beta_2$  and  $\beta_3$ .

Comparing **movement imagination** and **relax**, the obtained result is that the rectangular window is the most significant. The features with more significant power are  $\theta$ ,  $\beta_1$  y  $\beta_2$ .

**Movement imagination** versus **movement realization** are the two mental activities with more discriminating power. The most important features are  $\theta$ ,  $\alpha_1$ ,  $\alpha_2$ , and  $\beta_1$ . All types of windows obtain a very good result.

Making the comparison between **relax** activities appears significant difference between populations in the features  $\alpha_2$  and  $\beta_2$  of the channel 2, and  $\beta_1$   $\beta_2$  of channel 1. It is a case of false positive identification due to the noise in the signal, for this reason the window with better behaviour is the Tukey's one, it only detect a false positive in  $\alpha_2$ .

### 6.2.2 Results of reduction of the feature space dimensionality

The LDA technique produces only two eigen-values bigger than  $1 * 10^{-4}$  for all the experiments, this originates that only two eigen-vectors are considered in the transformation matrix,  $W$ , hence the population of the six dimension feature vectors are projected in a 2D space:  $X_1, X_2$ . Matrices in eqs. 43 and 44 show typical experimental values for  $\lambda$  and  $W$ .

$$\lambda = \begin{pmatrix} 0.109 & 0 \\ 0 & 0.020 \end{pmatrix} \quad (43)$$

$$W = \begin{pmatrix} \theta & \alpha_1 & \alpha_2 & \beta_1 & \beta_2 & \beta_3 \\ -0.06 & 0.22 & 0.05 & -0.05 & 0.06 & -0.9 \\ -0.37 & 0.01 & -0.002 & -0.56 & 0.73 & 0.16 \end{pmatrix}^T \quad (44)$$

In figures 19 and 20 are presented the results of the bilateral contrast test for the transformed coordinate  $X_1$  and  $X_2$  respectively, considering the Off-line and On-line experiments. The figures show for each channel (C3'-C3'' and C4'-C4''), and for each type of preprocessing window, the results  $p$  of the associated probability of the bilateral contrast tests between the mental tasks. In order to represent the dispersion of the results the mode value and bars from 15th to 85th percentile have been used.

The comparisons between the discrimination capabilities of On-line and Off-line experiments are shown in the figures 19 and 20. From the bilateral contrast test carried out with a significant level of  $\alpha = 2.5\%$ ,  $\alpha = 1 - p$ , represented in Figure 19 for  $X_1$ , it is obtained that in almost all cases the null hypothesis  $H_0$ , which maintains the equality in the populations of the features associated to the mental tasks, shall be rejected for both types of experiments; it is observed that  $p$  values obtained in the comparison of mathematical task versus motor imagery,  $p$  values are lower for the On-line case in both channels and with all types of preprocessing windows than the ones  $p$  values obtained for the Off-line case; the dispersion of the results is similar in both experiments. It is also shown that for  $X_1$ , channel C4'-C4'' performs better than C3'-C3''. The best results are obtained for  $X_1$  with Tukey's and Kaiser's windows. The same analysis for  $X_2$ , Figure 20, shows that the difference rarely appears for Off-line experiments, and in any case for in the On-line cases,  $p < 0.975$ . The same analysis for  $X_2$ , figures 10 to 15, shows that

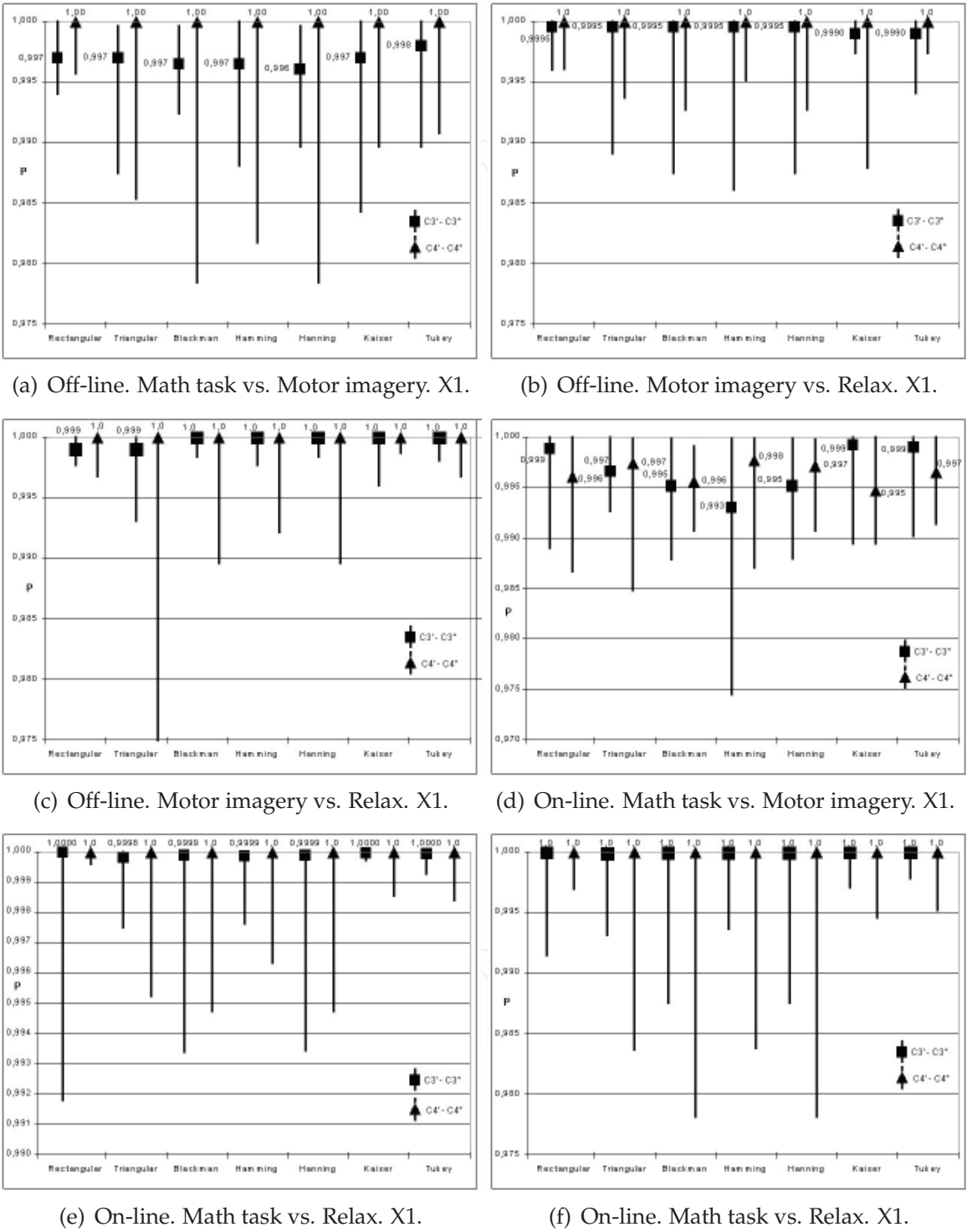


Fig. 19. Off-line. Math task vs. Motor imagery. Coordinate X1.

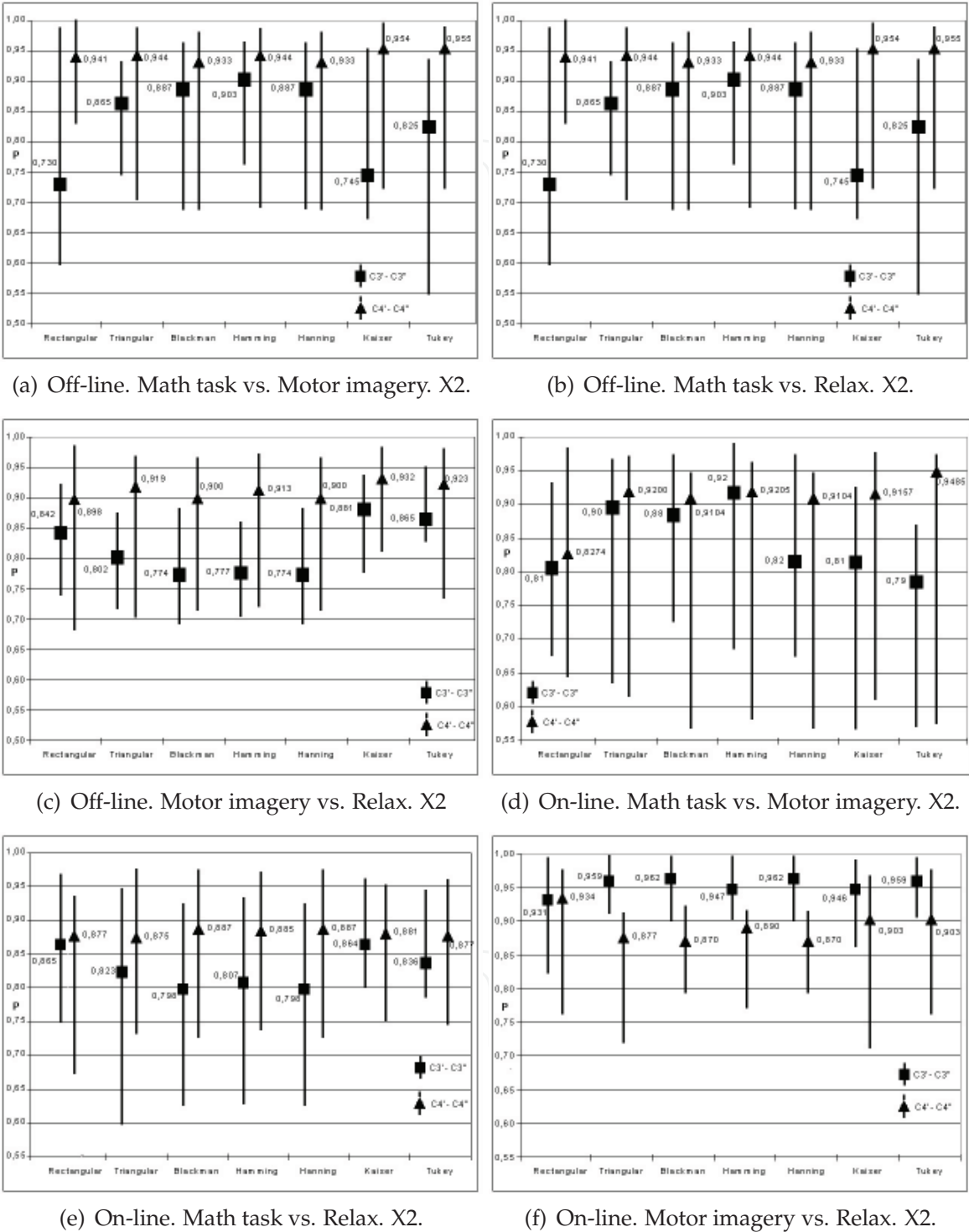


Fig. 20. Off-line. Math task vs. Motor imagery. Coordinate X2.

the difference rarely appears for Off-line experiments, and in any case for in the On-line cases,  $p < 0.975$ .

On average, for both types of experiments, all preprocessing windows show statistical difference between mental tasks; the best results, with higher quantities for the mode values and lower dispersion, are obtained for  $X_1$  with Tukey's and Kaiser's preprocessing windows. From the numerical results is observed that as higher the eigen-value magnitude, case of  $X_1$ , the higher the value of one component of the eigen-vector, normally in  $\beta$  frequency band, by the contrary, as lower the eigen-value more the contribution of the rest of eigen-vector components.

The highest contrast power is obtained in the comparison of *Motor imagery* vs. *Relax*, it is followed by *Mathematical task* vs. *Relax*, and the lowest is for *Mathematical task* vs. *Motor imagery*.

### 6.2.3 Results of Neural Networks Classifiers for BCI devices

The figures 21 to 23 summarize on vertical axis the percentage of correct classifications that were obtained from the confusion matrices and applied to each one of the three classifiers. It should be noted that the scale was broken in order to convey the scattered results. On the horizontal axis are displayed the different types of preprocessing windows that were considered.

For each preprocessing window, there appears a bar with the results of each classifier: maximum, minimum and median percentage values.

For each volunteer, the first two figures show the correct classifications obtained from each dedicated neural network:  $C3 \Rightarrow$  Channel 1, and  $C4 \Rightarrow$  Channel 2. The third figure shows the correct classifications obtained when the classifier was based on only one neural net. It gathers the feature vectors of both electroencephalographic channels at the same time.

The following considerations can be extracted from an analysis of the results:

- The data obtained from the experiments suggests that classifiers that were based on Probabilistic Neural Networks or Radial Basis Functions perform better than classifiers that were based on MLP (i.e., approximately 84% versus 33% of correct classifications). This indicates that the interpolation capability of the RBF outweighs the extrapolation capability of the MLP. A similar percentage of approximately 88% of correct classifications was obtained by N.Nicolau using classifiers based on Support Vector Machines with Gaussian kernels (Georgiou; & M.Polycarpou, 2008). A.Ferreira reported similar results in the best case of experiments for a robotic wheelchair using SVM as classifier (Andre Ferreira, Teodiano Freire, Mario Sarcinelli, Jose Luis Martin Sanchez, et al., 2009).
- Result stability. For all tests, the procedure was replicated three times, wherein both PNN and RBF classifiers produced the same confusion matrices. Thus, instead of the MLP classifiers, different matrices were produced for each replica. This indicates that with the proposed training data sets, the MLP classifiers are unable to differentiate between the different tasks. ( In the learning phase the RBF nets allocate the center of each radial basis function, meanwhile the MLP nets get the weights associated to the inputs of each neuron, if there is not a clear difference in the training sets of the input data populations, the MLP classifiers are unable to stablish the difference between them (Bishop, 1995), (Ripley, 1996)). It also explains the rate of correct classifications obtained with PNN, RBF and MLP neural networks.
- A comparison of PNN and RBF classifiers shows that, in some cases, the PNN produced better classification rates but also a higher variability. It should be noticed that although

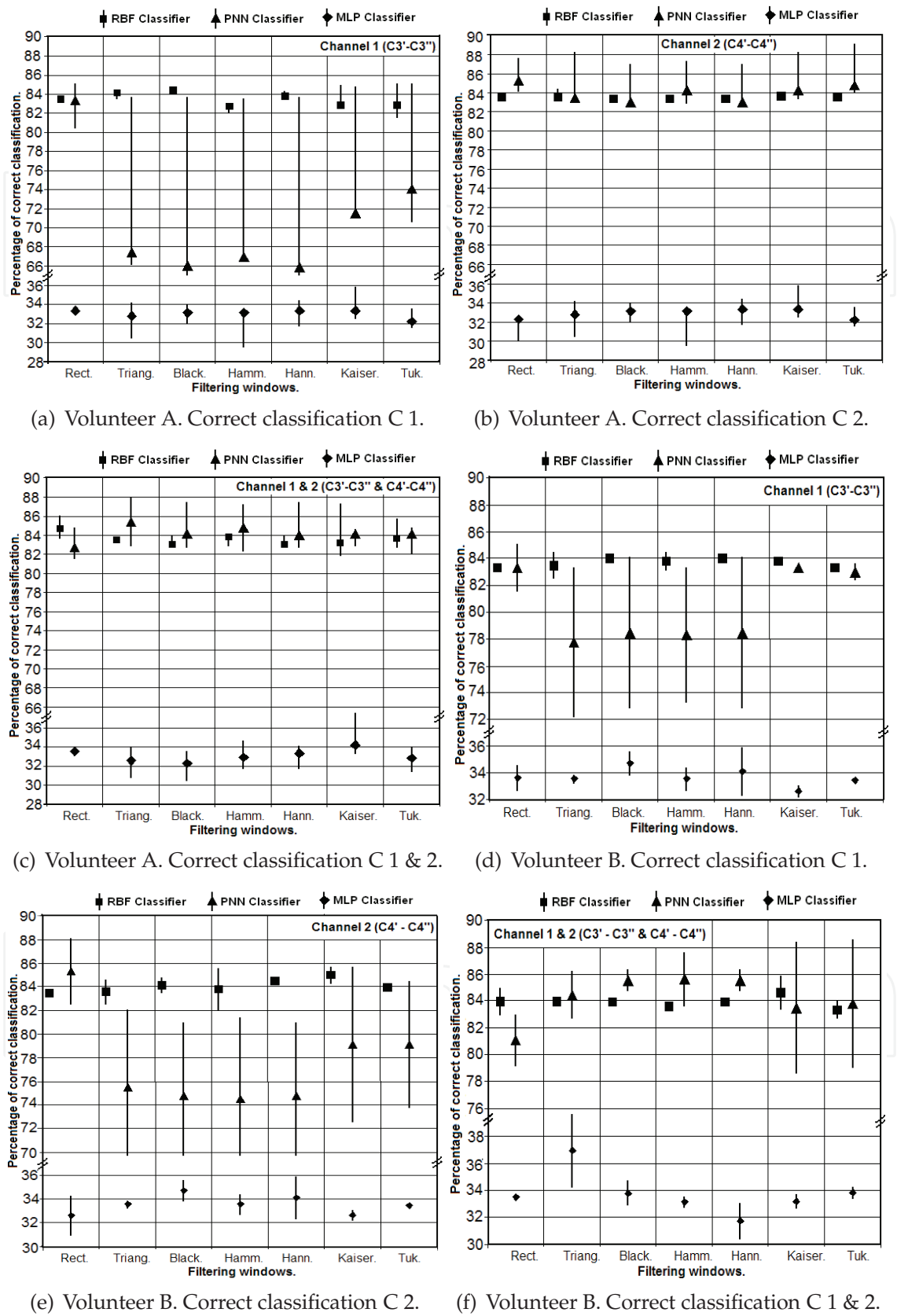


Fig. 21

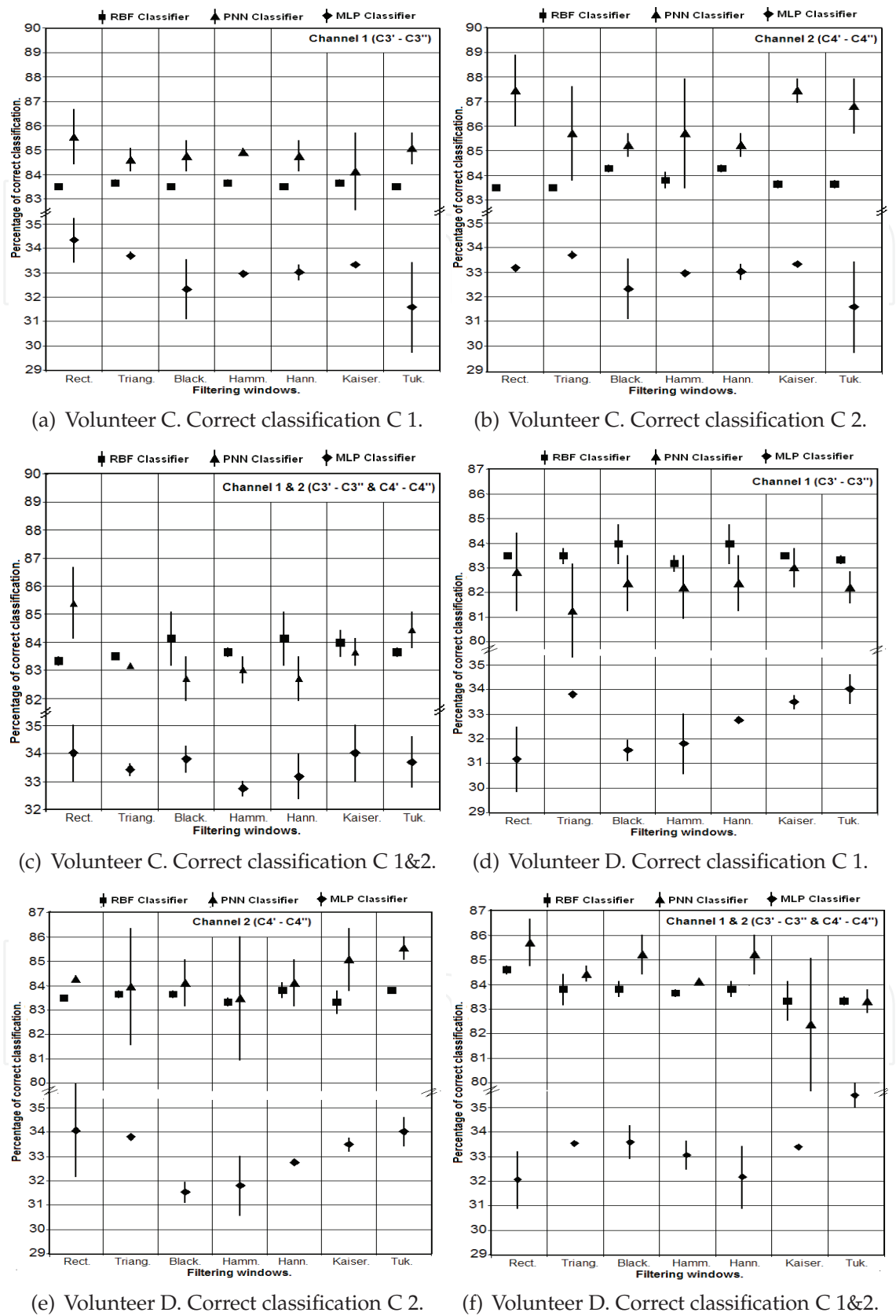
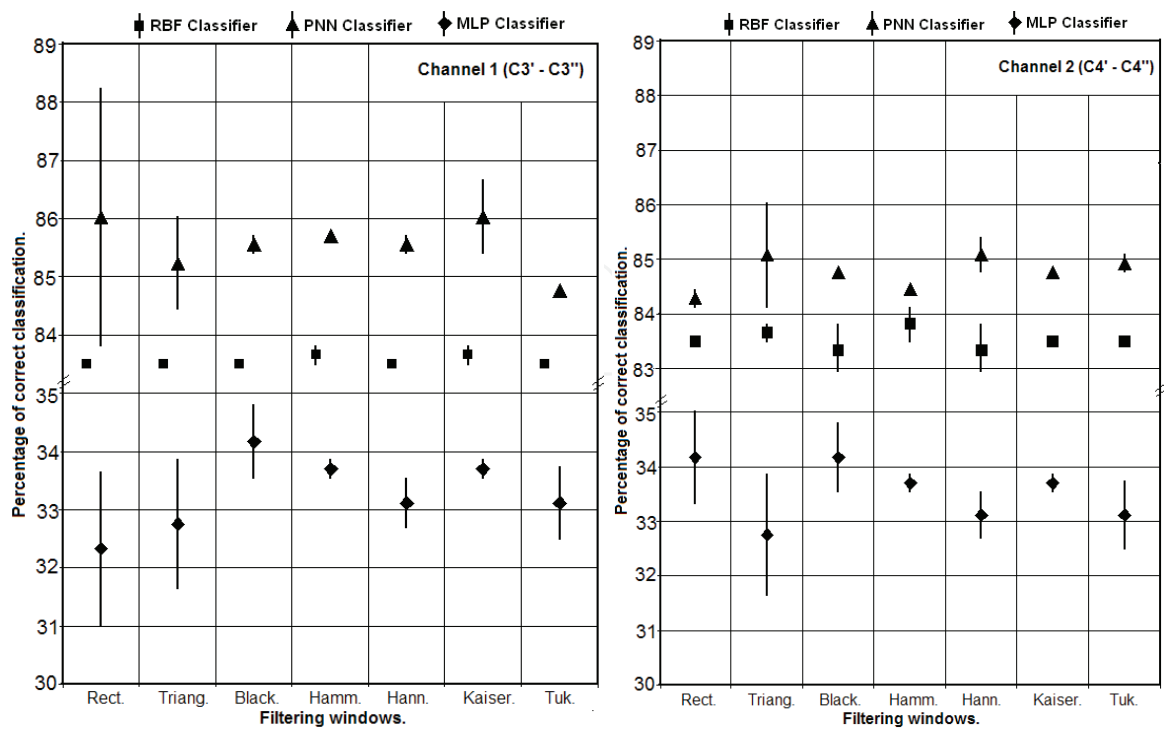
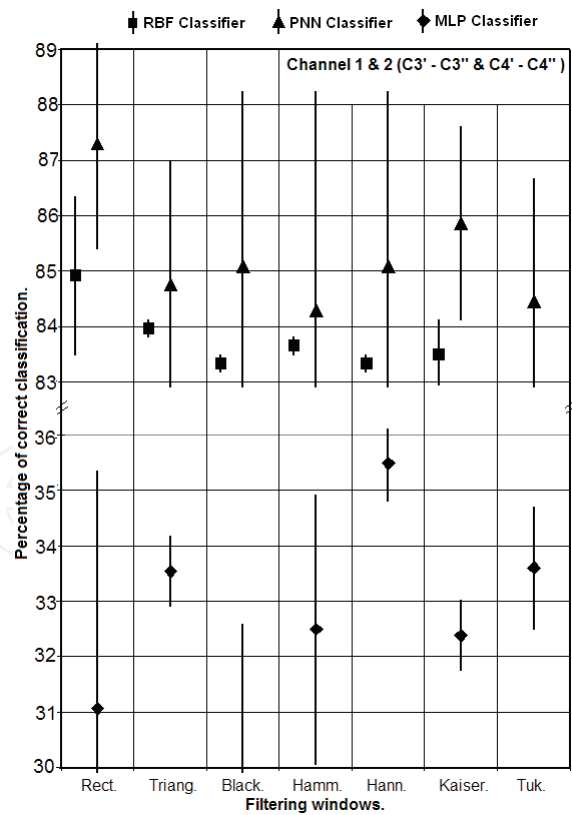


Fig. 22





(a) Volunteer E. Correct classification C 1. (b) Volunteer E. Correct classification C 2.



(c) Volunteer E. Correct classification C 1 & 2.

Fig. 23

RBF’s and PNN’s are both based on radial basis functions, in the case of PNN’s exist as much functions as samples in the training set, whereas in the case of RBF’s this quantity is determined through an iterative process and it is usually lower (Horward Demuth, 2006). This may cause the overlearning of the training data set and a lower generalization capability for PNN’s which explains its higher variability (Bishop, 1995).

- Classifiers based on only one neural network, and that simultaneously consider features obtained from both electroencephalographic channels, do not always perform better than classifiers based on two neural networks, one for each channel.
- A consideration of different types of preprocessing windows demonstrates that the results with lowest variability values are achieved with Kaiser’s, Tukey’s, and rectangular windows. This corroborates the conclusions of previous studies concerning the influence of preprocessing windows in mental tasks discrimination capability (Martinez, J.L.; Barrientos, A., 2006),(Martinez, J.L.; Barrientos, A., 2007). On average, however, similar classification rates are obtained for all. This indicates that although the type of preprocessing window is important, the behaviour of the classifier is even more relevant.

6.2.4 Results of adaptative bi-stage classifier based on RBF-HMM

In order to test the behavior of the proposed algorithm, determine the influence of the threshold assignation parameter ( $\lambda$ ), and the influence zone of the neuron ( $S_c$ ), the EEG samples of the session tests from the volunteers were used as follows.

6.2.4.1 Evaluation of the learning capability.

With a subset of 75% of the all EEG samples, the algorithm was trained with different  $\lambda$  and  $S_c$  values<sup>4</sup>:

$\lambda =$	0.55	0.65	0.8
$S_c =$		0.5	0.95

After the learning, the same samples were processed with the trained algorithm, and a comparison between the results obtained with the algorithm and the ones employed for the learning was done, in all cases a 100% of correct classification has been obtained.

6.2.4.2 Evaluation of the generalization capability.

After the good results obtained from the learning phase, a cross-validation methodology is used to estimate the generalization capability. From the whole ten sessions, nine are used for learning and one is used for validation, the process is repeated ten times changing each time the session used for validation.

In the following tables are shown the results obtained for each volunteer, considering the  $\lambda$  and  $S_c$  parameters.

For each combination, the process is replicated three times. In the upper row it is shown the number of correct classifications. In the lower row it is shown the percentage of improvement against a naive classifier.

From the results of the proposed classification algorithm, it is observed that:

- The learning capability is better that the one achieved only with the RBF neural network (Martinez, J.L.; Barrientos, A., 2008).

<sup>4</sup> These values have been fixed after a seek in wide with the samples of the first volunteer.

$S_c = 0.5$ $\lambda = 0.65$			$S_c = 0.95$ $\lambda = 0.55$			$S_c = 0.5$ $\lambda = 0.55$			$S_c = 0.95$ $\lambda = 0.80$		
94	103	103	94	81	87	93	92	87	86	97	81
+4	+14	+14	+4	-10	-3	+3	+2	-3	-4	+8	-10

Table 3. Volunteer: Al01.

$S_c = 0.5$ $\lambda = 0.65$			$S_c = 0.95$ $\lambda = 0.55$			$S_c = 0.5$ $\lambda = 0.55$			$S_c = 0.95$ $\lambda = 0.80$		
103	97	92	118	109	118	97	87	86	117	106	110
+14	+8	+2	+31	+21	+31	+8	-3	-4	+30	+18	+22

Table 4. Volunteer: Ro01.

$S_c = 0.5$ $\lambda = 0.65$			$S_c = 0.95$ $\lambda = 0.55$			$S_c = 0.5$ $\lambda = 0.55$			$S_c = 0.95$ $\lambda = 0.80$		
106	97	110	87	90	107	99	106	107	98	108	99
+18	+8	+22	-3	0	+19	+10	+18	+19	+9	+20	+10

Table 5. Volunteer: Ja01.

$S_c = 0.5$ $\lambda = 0.65$			$S_c = 0.95$ $\lambda = 0.55$			$S_c = 0.5$ $\lambda = 0.55$			$S_c = 0.95$ $\lambda = 0.80$		
109	102	104	83	92	92	106	91	110	86	87	92
+21	+13	+15	-8	+2	+2	+18	+1	+22	-4	-3	+2

Table 6. Volunteer: Da01.

$S_c = 0.5$ $\lambda = 0.65$			$S_c = 0.95$ $\lambda = 0.55$			$S_c = 0.5$ $\lambda = 0.55$			$S_c = 0.95$ $\lambda = 0.80$		
106	97	110	87	90	107	99	106	107	91	76	99
+18	+8	+22	-3	0	+19	+10	+18	+19	+1	-15	+10

Table 7. Volunteer: Ra01.

$S_c = 0.5$ $\lambda = 0.65$			$S_c = 0.95$ $\lambda = 0.55$			$S_c = 0.5$ $\lambda = 0.55$			$S_c = 0.95$ $\lambda = 0.80$		
102	102	98	102	107	114	103	105	96	116	99	98
+13	+13	+9	+13	+19	+26	+14	+16	+6	+29	+10	+9

Table 8. Volunteer: Ra02.

- From the analysis of the results of the replicas it has been detected that the variability in the percentage of correct classifications is caused by the HMM's, both in the learning and validation phases. The sequences of pre-assignments provided by the neural network were stable, but the generation probabilities of the HMM's changed in each replica. In the learning phase the HMM's probabilities allowed a perfect classification, but they were not maintained in the cross validation phase; for this stage a lower percentage of correct classification was obtained, as it is summarized in the tables 3 to 8. But until in this case, almost in all replicas, the cross-validation test results were better than the ones hoped from a naive classifier.
- The values of correct classifications depend highly from the user. There has not been identified a pair of  $\lambda$  and  $S_c$  values which proportionate the highest percentage of correct classification for all users. The discrepancy in the results between RA1 and RA2 is explained by the user's learning process, session RA1 is previous to RA2.

6.2.5 Results of SVM

This section presents graphically the results obtained from the application to eleven experimental sessions, following the off-line procedure in 6.1.4, the SVM classifier described in the subsection 5.3.

In order to determine the SVM classifier with biggest generalisation capability or the classifier with highest percentage of correct classifications and lower number of support vectors the following values for the kernel parameters in procedure 6.1.4 have been considered.

Kernel type.	Kernel parameter value.
Gaussian	1, 2, 3, 5 y 10.
Polynomial	2, 3, 4, 7 y 8.

Table 9. Kernel function family and Kernel parameters.

In figures 24 to 27 the following figures are shown for each cerebral channel and type of SVM kernel, the percentage of correct classifications and support vectors obtained for each classifier taking in consideration each type of the processing windows.

The percentages are shown in box diagrams with the notch in the mean value and its uncertainty represented by the size of the box, the segments in both extremes of each box represent the variability<sup>5</sup>. If there are outlander samples they are represented by small empty circles at the extreme of each segments.

From the inspection of the results of classifiers based on Gaussian kernels it is observed that:

- When the zone of influence of the kernel function increases the percentage of correct classifications is reduced independently of the electroencephalographic channel and processing window type. It goes from 100% of correct classifications for  $n = 1$  to 90% – 80% with Tukey’s, Kaiser’s and rectangular window types<sup>6</sup> with gaussian kernels with  $n = 10$ .
- When the parameter of the gaussian kernel increases, the percentage of support vector reduces, it goes from 100% of learning data set with  $n = 1$ , to 85% – 74% for  $n = 10$ . The lower values are achieved for rectangular, Tukey’s and Kaiser’s type of window.

Performing the same analysis with the results of classifiers based on polynomial kernels it is observed that:

- When the order of the polynomial kernel increases the number of support vector is reduced, getting 100% of correct classifications with kernels of order four or higher with Tukey’s, Kaiser’s and rectangular processing window types.
- With kernel of order four or higher the percentage of support vector is established between 45% and 50% of the learning data set depending on the processing window type, the lowest percentage are obtained with Tukey’s, Kaiser’s and rectangular’s window types, the lowest variability is obtained with the last one.
- Comparing both electroencephalographic channels it is observed that the behaviour of the right one, C2, it is better than the left one, C1.

The classifiers based on Gaussian kernels trend to over-learn the training data set, when the kernel parameter is small all training samples become support vectors, meanwhile the

<sup>5</sup> The length of each segment is 1.5 times the standard deviation.  
<sup>6</sup> For the rest of window types the percentage of correct classification is lower.

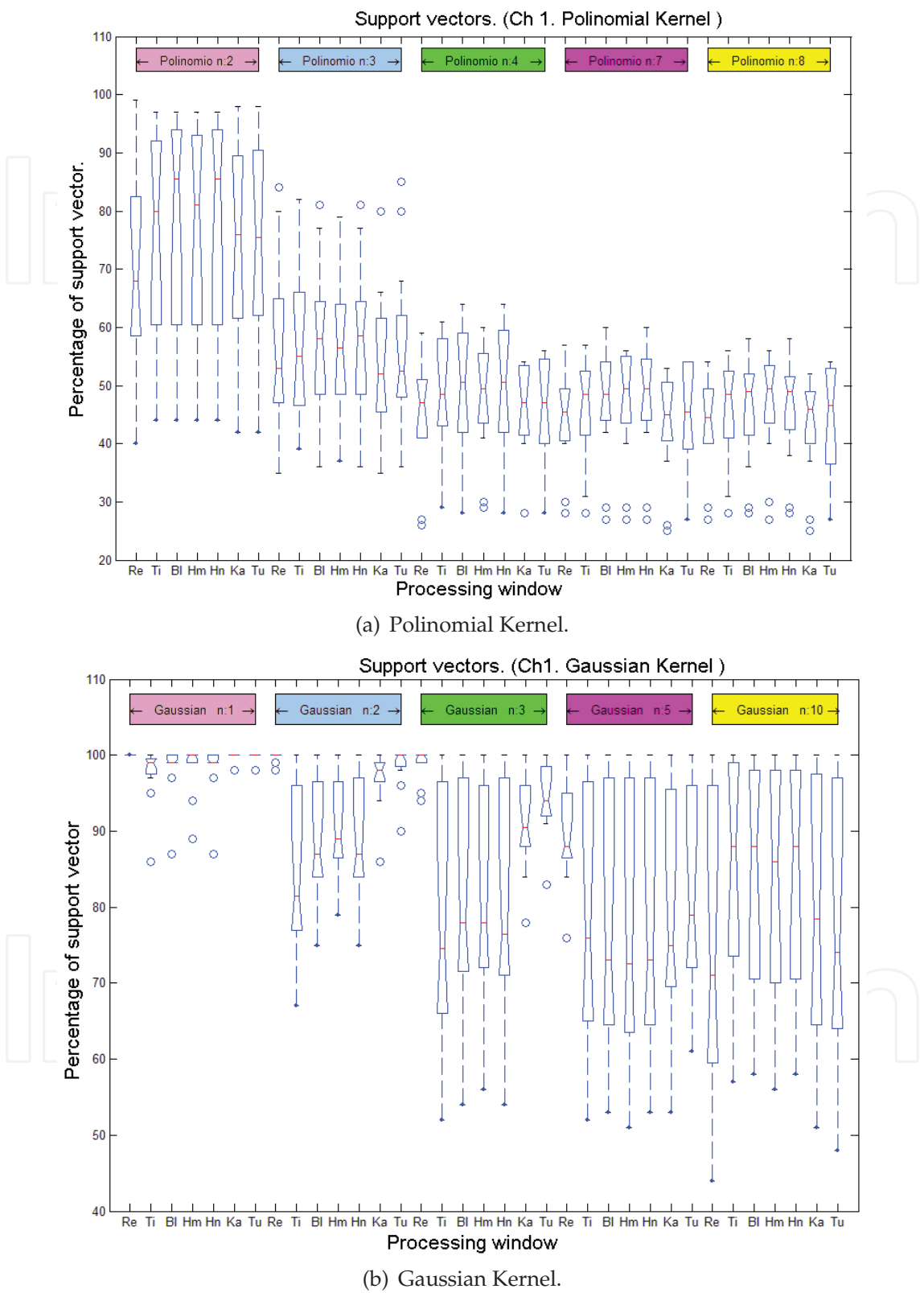


Fig. 24. Percentage of support vector. Channel 1.

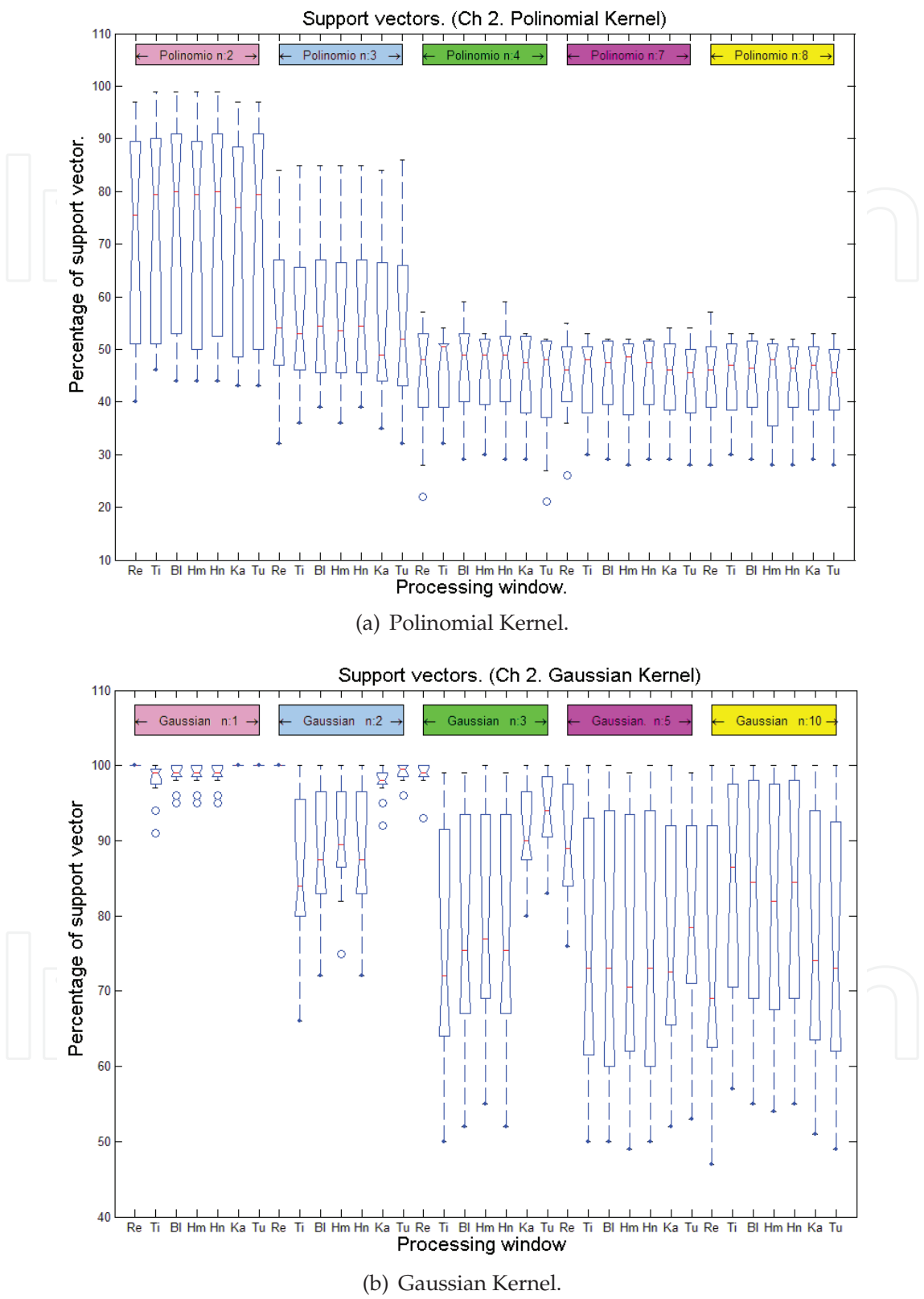
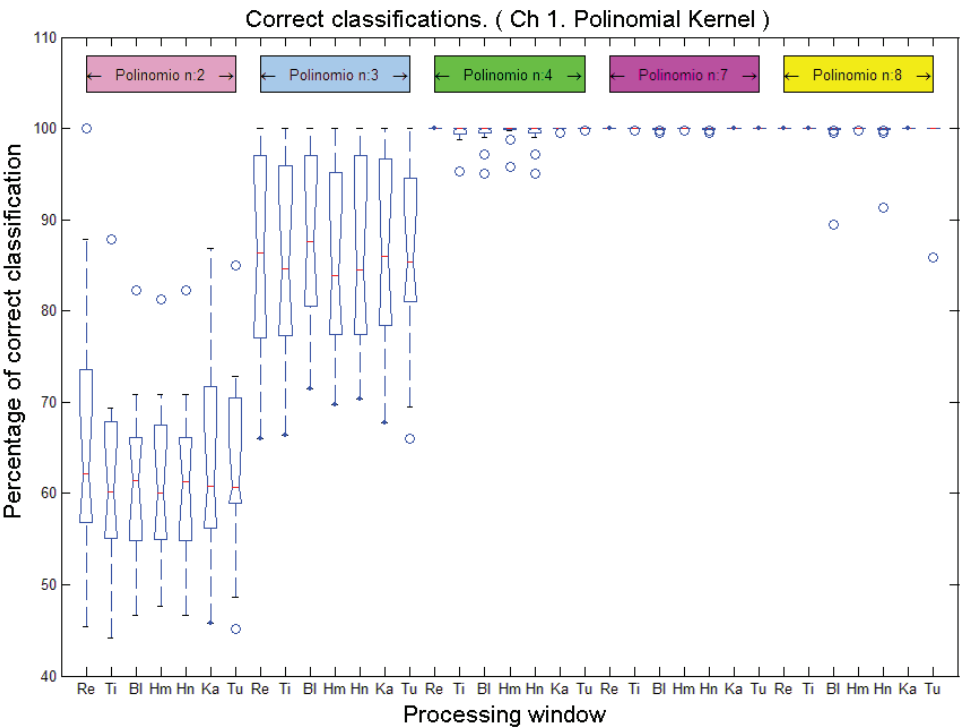
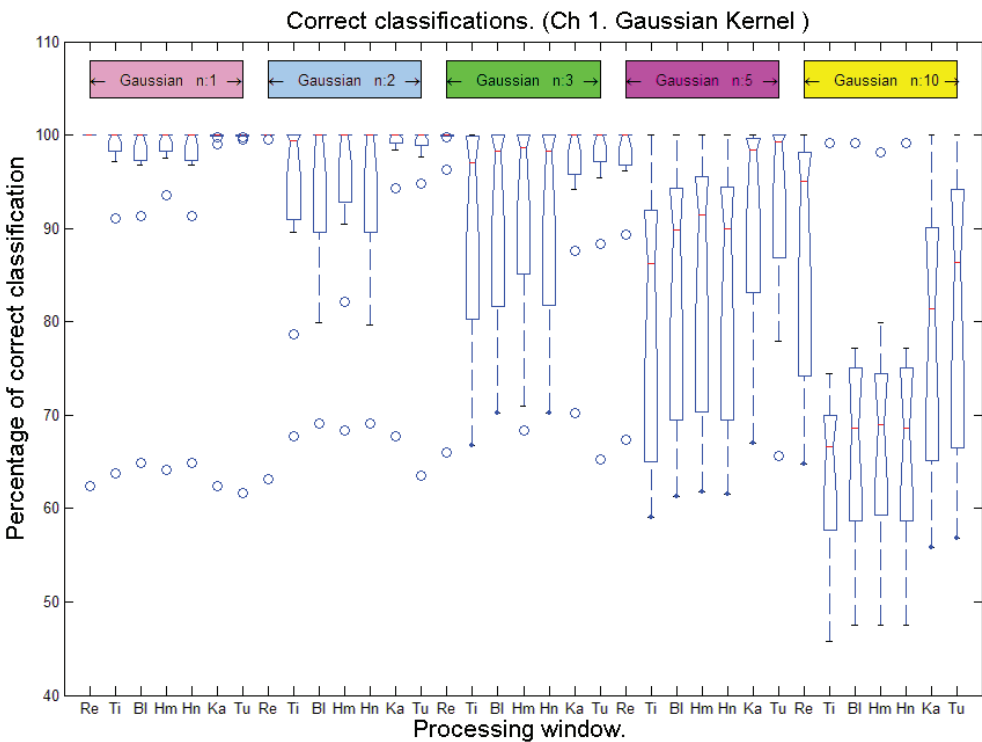


Fig. 25. Percentage of support vector. Channel 2.



(a) Polinomial Kernel.



(b) Gaussian Kernel.

Fig. 26. Percentage of correct classifications. Channel 1.



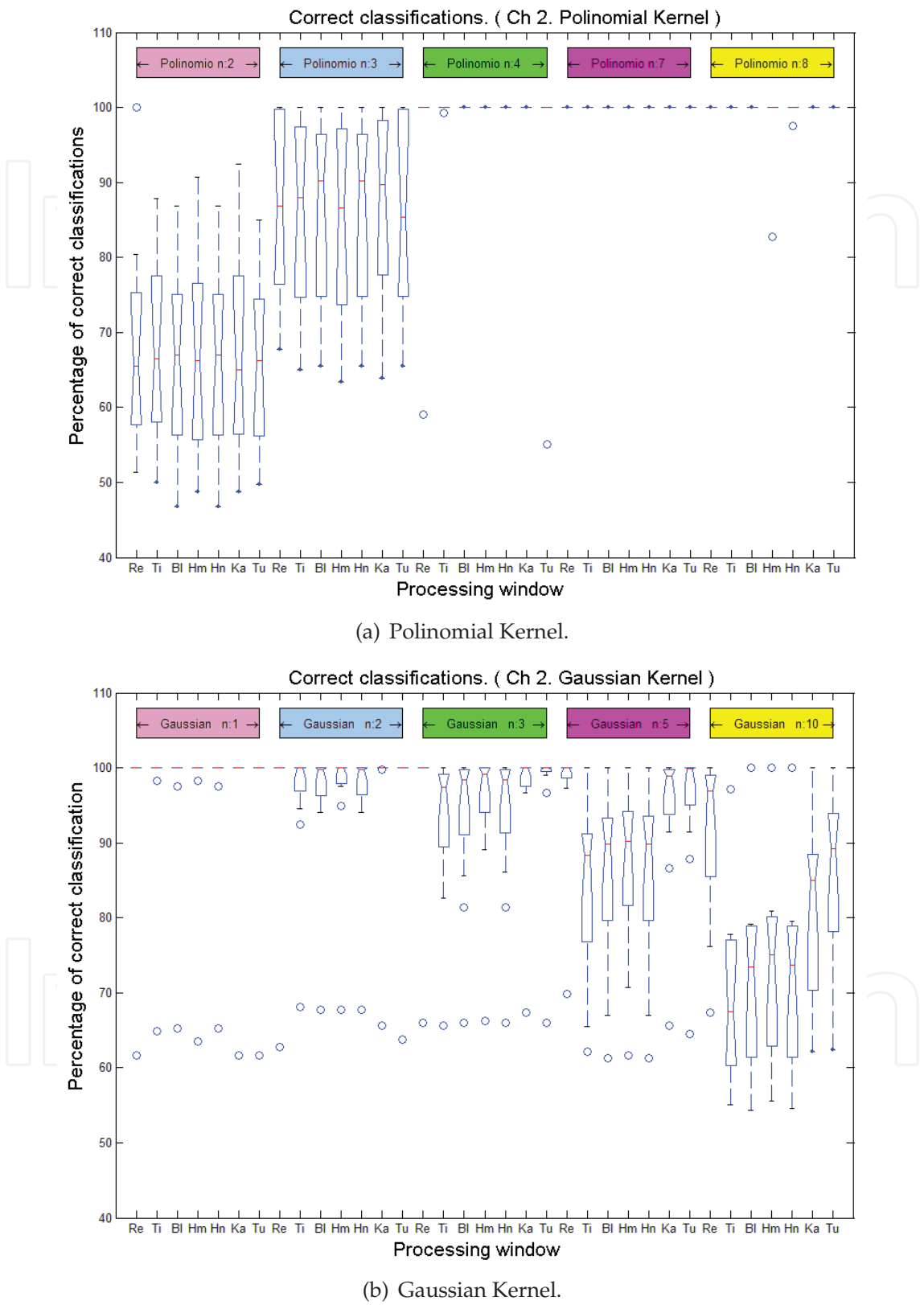


Fig. 27. Percentage of correct classifications. Channel 2.

parameter increases the number of support vectors decreases, but in the same way the percentage of correct classifications is reduced.

As it can be seen in the firsts points of the former analysis, the best results are obtained with the Tukey's, Kaiser's and rectangular processing windows.

The classifiers based on polynomial kernels do not over-learn the training data set, because as it is shown in the third and fourth items of the former analysis, when the polynomial order of the kernel increases the number of support vector reduces, but on the contrary the percentage of correct classification gets bigger, nevertheless for polynomial kernels with order higher than five there is not an improvement in their behaviour.

The classifiers based on polynomial kernels exhibit better behaviour than the classifiers based on Gaussian kernels, with polynomial kernels of order four and higher practically a percentage of 100% of correct classification is achieved with 45% – 50% of support vectors, meanwhile the classifiers with gaussian kernels need 74% – 85%, this means that the generalization capability of polynomial kernels is bigger than the Gaussian ones.

The better results of channel two is in line with the exposition in 6.2.2.

## 7. Conclusions

A classifier which discriminates between mathematical activity and movement imagination should consider the Tukey's window as filtering window, and the features  $\alpha_1$ ,  $\beta_1$  and  $\beta_2$ .

It is important to note that the Tukey's window minimises false positive, so it is more reliable than the others types of window.

In these test the channel two (C4) is more significant than channel one(C3), it is probably due to the subject dexterity, more tests should be realized with left-handed and right-handed subjects to determine the influence of it.

It has been statistically proven that through the use of LDA technique it is possible to reduce the dimensionality of the original input feature space, meanwhile the discrimination capability between the proposed mental tasks is maintained, allowing the control of external devices through the association of these tasks to device commands.

From the experiment results carried out by five volunteers under the Off-line an On-line experimental procedures, it is possible to conclude that the user's feedback influence provokes a lower discrimination capability, but enough to be used in an On-line BCI device, (Pineda, J.A. et al., 2003).

It is also shown that Tukey's and rectangular preprocessing windows improve the discrimination capability between the considered mental tasks.

The LDA technique allows to weight the power amplitude of frequency bands, and at the same time, allows to reduce the feature input space maintaining the particularities of the considered cerebral activities. On the other hand, eigen-vector analysis shows that the discrimination power is manifested over  $\beta$  band, mainly  $\beta_2$  and  $\beta_3$ , they are the vector components with highest contributions in the transformation matrices.

From the result analysis of classifiers based on neural networks are obtained the following conclusions:

1. It is feasible to discriminate between mental tasks using the user's electroencephalographic signal from only the two proposed channels:  $C3' - C3''$  and  $C4' - C4''$  (J. del R. Millan, 2003).
2. In the case of vectors of features that are based on the power of the signal frequency bands, the election of classifiers based on RBF's shows better classification rates than those that are based on MLP. A similar conclusion is obtained by Garrett in (Peterson, 2003).

3. Classifiers that are based on an architecture with multiple neural networks (one for each electroencephalographic channel, followed by a block that weighs the classification results given by the networks) are in a better position to produce higher correct classification rates than those classifiers that are based on only one neural network.
4. The use of Kaiser's or Tukey's preprocessing window allows for an increase in the mental tasks discrimination capability and improves the behaviour of the classifier.

On the other hand, from the analysis of the results of the adaptative bi-stage classifier it is possible to conclude that the information inside the pre-assignment sequences improves the classification capability, therefore the Hidden Markov Model technique is useful for the extraction and use of this information in an On-line BCI device.

The scattering of the maximum values, of the correct classifications obtained from the cross-validation tests, shows that the combination of  $\lambda$  and  $S_c$  parameters are highly dependent on the user, for this reason a BCI device based in this kind of algorithm should have a setup stage, that allows to initialise correctly these parameters.

The algorithm behaves better than a naive algorithm, but it is not as good as it should be taking into account the good results obtained during the learning phase. The size of the learning data set is critical in the results obtained during the validation phase. With a bigger learning data set the validation results will improve, because of the minimisation of the overlearning.

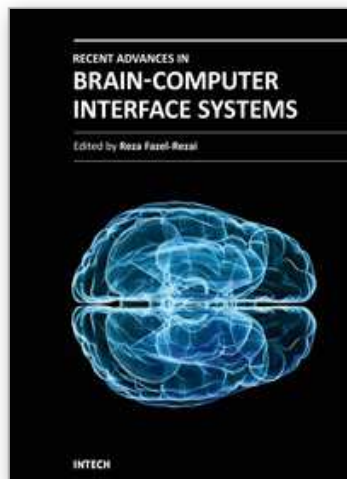
Finally from the analysis and discussion of the results of the tests carried out using classifiers based on MSV, the following conclusions are obtained:

- It is better to choose polynomial kernels instead of Gaussian ones.
- The architecture of the classifier should employ polynomial kernels of order 4 or 5 (it is preferable to choose 5 in order to maximise the generalisation capability of the classifier, but not higher because it does not mean an improvement in the classifier behaviour); and using a processing window of the Tukey's, Kaiser's and rectangular types.

## 8. References

- Andre Ferreira, Teodiano Freire, Mario Sarcinelli, Jose Luis Martin Sanchez, et al. (2009). Evaluation of psd components and aar parameters as input features for a svm classifier applie to a robotic wheelchair, *Proceedings of the BIODEVICES International Conference on Biomedical Electronics and Devices*. 1(1): 7–12.
- Birbaumer, N; et al. (2000). The thought translation device (TTD) for completely paralyzed patients, *IEEE TRANS. ON REH. ENG.* 8(2): 190–193.
- Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*, Oxford University Press, Great Britain.
- E. Donchin and K. M. Spencer and R. Wijesinghe (2000). The mental prosthesis: assessing the speed of a P300-based brain-computer interface, *IEEE Transactions on Rehabilitation Engineering* 8(2): 174–179.
- Georgiou, N. N. J. & M.Polycarpou (2008). Autoregressive features for thought to speech converter, *Proceedings of the BIODEVICES International Conference on Biomedical Electronics and Devices*. 1(1): 11–16.
- Harris, F. J. (1978). On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform.
- Horward Demuth, e. a. (2006). *Neural Network Toolbox. User's Guide*, 5 edn, The MathWorks.
- J. del R. Millan (2003). Adaptative brain interfaces., *Communications of the ACM*. (46): 74–80.

- Kostov, A. (2000). Parallel man-machine training in development of EEG-based cursor control., *IEEE TRANS. ON REH. ENG.* 8(2): 203–205.
- Martinez, J.L.; Barrientos, A. (2006). The windowing Effect in Cerebral Pattern Classification. An Application to BCI Technology., *IASTED Biomedical Engineering BioMED 2006* pp. 1186–1191.
- Martinez, J.L.; Barrientos, A. (2007). Linear Discriminant Analysis on Brain Computer Interface., *2007 IEEE. International Symposium on Intelligent Signal Processing. Conference Proceedings Book.* pp. 859–864.
- Martinez, J.L.; Barrientos, A. (2008). BRAIN COMPUTER INTERFACE. Comparison of Neural Networks Classifiers., *Proceedings of the BIODEVICES International Conference on Biomedical Electronics and Devices.* 1(1): 3–10.
- Neuper, C.; et al. (2001). Motor Imagery and Direct Brain-Computer Communication., *Proceedings of the IEEE* 89(7): 1123–1134.
- Obermaier, B., Neuper, C., Guger, C. & Pfurtscheller, G. (2001). Information transfer rate in a five-classes brain-computer interface, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 9(3): 283–288.
- Penny, W. D.; et al. (2000). EEG-based communication: A pattern recognition approach, *IEEE TRANS. ON REH. ENG.* 8(2): 214–215.
- Peterson, D. G. D. A. (2003). Comparison of linear, nonlinear, and feature selection methods for eeg signal classification., *Neural Systems and Rehabilitation Engineering.* (11): 141–144.
- Pfurtscheller et al. (2000a). Brain oscillations control hand orthosis in a tetraplegic., *Neuroscience Letters* 1(292): 211–214.
- Pfurtscheller et al. (2000b). Current trends in Graz brain-computer interface (BCI) research., *IEEE TRANS. ON REH. ENG.* 8(2): 216–219.
- Pineda, J.A. et al. (2003). Learning to Control Brain Rhythms: Making a Brain-Computer Interface Possible., *IEEE TRANS. ON REH. ENG.* 11(2): 181–184.
- Proakis, J. G. & Manolakis, D. G. (1997). *Tratamiento digital de señales : [principios, algoritmos y aplicaciones]*, Prentice-Hall, Madrid.
- Rabiner, L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.
- Rabiner, L. R. & Juang, B. H. (1986). An Introduction to Hidden Markov Models.
- Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*, Cambridge University Press, United Kingdom.
- Vidal, J. J. (1973). Toward direct brain-computer communication, *Annu. Rev. Biophys. Bioeng.* 2: 157–180.
- Wolpaw, J. R. (2007). Brain-Computer Interfaces as New Brain Output Pathways, *The Journal of Physiology*.
- Wolpaw, J. R.; et al. (2002). Brain-Computer interface for communication and control., *Clinical Neurophysiology* 113: 767–791.
- Wolpaw, J.R.; et al. (2000). Brain-Computer Interface Technology: A Review of the First International Meeting., *IEEE TRANS. ON REH. ENG.* 8(2): 164–171.



## **Recent Advances in Brain-Computer Interface Systems**

Edited by Prof. Reza Fazel

ISBN 978-953-307-175-6

Hard cover, 222 pages

**Publisher** InTech

**Published online** 04, February, 2011

**Published in print edition** February, 2011

Brain Computer Interface (BCI) technology provides a direct electronic interface and can convey messages and commands directly from the human brain to a computer. BCI technology involves monitoring conscious brain electrical activity via electroencephalogram (EEG) signals and detecting characteristics of EEG patterns via digital signal processing algorithms that the user generates to communicate. It has the potential to enable the physically disabled to perform many activities, thus improving their quality of life and productivity, allowing them more independence and reducing social costs. The challenge with BCI, however, is to extract the relevant patterns from the EEG signals produced by the brain each second. Recently, there has been a great progress in the development of novel paradigms for EEG signal recording, advanced methods for processing them, new applications for BCI systems and complete software and hardware packages used for BCI applications. In this book a few recent advances in these areas are discussed.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

José Luis Martínez, Antonio Barrientos (2011). Applied Advanced Classifiers for Brain Computer Interface, Recent Advances in Brain-Computer Interface Systems, Prof. Reza Fazel (Ed.), ISBN: 978-953-307-175-6, InTech, Available from: <http://www.intechopen.com/books/recent-advances-in-brain-computer-interface-systems/applied-advanced-classifiers-for-brain-computer-interface>

**INTECH**  
open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen