Evaluation of LDA Ensembles Classifiers for Brain Computer Interface

Cristian Arjona[1], José Pentácolo[1], Iván Gareis[1], Yanina Atum[1], Gerardo Gentiletti[1], Rubén Acevedo[1], Leonardo Rufiner[2,3]

[1] Laboratorio de Ingeniería en Rehabilitación e Investigaciones Neuromusculares y Sensoriales; Facultad de Ingeniería, Universidad Nacional de Entre Ríos, Argentina
[2] Centro de I+D en Señales, Sistemas e Inteligencia Computacional; Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional de Litoral, Argentina.
[3] Laboratorio de Cibernética; Facultad de Ingeniería, Universidad Nacional de Entre Ríos, Argentina.

E-mail; racevedo@bioingenieria.edu.ar

Abstract. — The Brain Computer Interface (BCI) translates brain activity into computer commands. To increase the performance of the BCI, to decode the user intentions it is necessary to get better the feature extraction and classification techniques. In this article the performance of a three linear discriminant analysis (LDA) classifiers ensemble is studied. The system based on ensemble can theoretically achieved better classification results than the individual counterpart, regarding individual classifier generation algorithm and the procedures for combine their outputs. Classic algorithms based on ensembles such as bagging and boosting are discussed here. For the application on BCI, it was concluded that the generated results using ER and AUC as performance index do not give enough information to establish which configuration is better.

1. Introduction

The brain computer interfaces are devices that provide a direct connection between brain and computer [1]. These interfaces could be the only way of communication for people that suffer motor disabilities as the ones described in [2].

The performance of the BCI is highly dependent, on the processing techniques use to extract features from electroencephalographic (EEG) signals that encoded user intentions [3] and on the classification techniques. Between the different evaluation alternatives of classifiers, the most natural one consists on feed a classifier with the features of the signals that are compared and use its performance as a measure of the discrimination power of the classifiers. In this case, you must be very careful for conceptually separate variations in the system performance (feature extractor and classifier) due to modifications in the classifier behaviour from variations due to the feature extraction techniques used.

An alternative technique to improve the performance of the classification process is the use of an ensemble of classifiers in parallel combining their outputs, where each classifier is trained in a different way. Under certain hypothesis systems based on ensemble can achieved better classification results than the individual counterpart, regarding single classifier generation algorithm and the

procedures for combine their outputs. With this technique the final classification error is reduced as a result of the contribution of each of the classifiers in the decision making [4].

This study intends to preliminary evaluate and compare the performance of three different configurations of an ensemble composed by three classifiers and a configuration of only one classifier, to distinguish EEG signals with event related evoked potentials (ERP) from signals without ERPs.

In this work we used Fisher linear discriminant analysis (LDA) because it is one of the most used classifiers in BCI applications. [5, 6].

2. Material and methods

2.1. Donchin's P300 speller

When infrequent visual, auditory or somatosensory stimuli are mixed with frequent stimuli, the first ones evoke a potential in the EEG in the parietal cortex of the subject with a peak located around the 300 ms. This phenomenon can be used for BCI implementations, that are known as P300-based paradigm, by the identification of the P300 as the principal ERP. These potentials are used to, for example, select icons from a matrix on the computer screen as is described then [7].

In the classic P300 paradigm, the user is in front of 6 x 6 matrix that has letters and numbers. During the experiment, one row or column is random intensified with a predefined frequency. In a complete block of 12 intensifications, each row and column lights one time. To make a choice of one character, the user watches this character and when it is intensified an ERP is evoked. As a result of the use of the stimulation matrix, two epochs with P300 (corresponding to the infrequent or objective stimuli) and ten epochs without it (frequent or no objective stimuli) for each block are generated. Generally, the intensifications block must be repeat to be sure that the detect character is the one choose by the user.

To detect which intensification evoked an ERP, the system must be capable of solving a binary classification problem (two possible classes: signals with ERP and signals without ERP).

2.2. Database

The Laboratorio de Investigación en Neuroimagenología (LINI) of the Universidad Autónoma Metropolitana (UAM) made available a database that has the registers of thirty healthy subjects, using the BCI2000 P300 speller [8,9]. The registers were sampling at 256 Hz using a number of channels (Nc) equal to ten. The registered channels were Fz, C3, Cz, C4, P3, Pz, P4, PO7, PO8, Oz referred to lobe of the right ear and using as earth the right mastoid.

Each subject was in four sessions of fifteen repetitions of the 12 stimuli of the matrix, with this configuration different sizes of words were spelled. From this procedure a labelled data base of Nt=630 registers with P300 and Nnt = 3150 registers without P300 is obtain for each subject.

One of the premises of this data base is preserving the registers as real as possible. As a result, many of them have significant number of artifacts. A selection of ten subject registers that present the lowest number of artifacts has been made. For this work, a random selection of two subject's registers from the previous ten was made.

2.3. Preprocess

From the original registers, a new data set for each subject was generated by decimation. The decimation frequency was 16 Hz (Fsi).

Before the down sampling, the registers were filter with a low pass eight order Chevysheb filter, with a cutoff frequency of 7 Hz. The filter was passed forward and backward in order to avoid phase distortion. Then the signal from each electrode was normalized independently to obtain zero mean and unit standard deviation.

Individual epochs of one second were extracted from the registers. The first sample corresponds to the instant when the chosen row or column was lighted (stimuli). Due to the duration of the epochs and the values of Fsi, the number of samples per epoch Nei is equal Fsi.

sinc(*i*) Research Center for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
C. Arjona, J. Pentácolo, I. E. Gareis, Y. Atum, G. Gentiletti, R. C. Acevedo & H. L. Rufiner; "Evaluation of LDA Ensembles Classifiers for Brain Computer Interface"
Memorias del XVIII Congreso Argentino de Bioingeniería (SABI 2011), set, 2011.

Last, the feature vectors (patterns) were built concatenating the epochs of 4 channels (PO7, PO8, Oz y Cz). These channels were chosen based on a preliminary behaviour analysis done in [10].

Patterns of 64 features were generated. This is the minimum number of features to use in BCI application with LDA according to [10].

2.4. Classifier

The purpose of the LDA is calculate a discriminant vector $w \in \Re^D$ (where D is the number of features), such that, given a set of training patterns, with their $x_j \in \Re^D$ corresponding class label, $j \in \{1..N\}$ separate the classes using a given discriminant function. This is achieved by maximizing the function given by:

$$J(w) = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{\tilde{s}_1^{\,2} + \tilde{s}_2^{\,2}} \tag{1}$$

where:

$$\tilde{m}_k(w) = \frac{1}{N_k} \sum_{i \in Y_k} w^{\mathrm{T}} x_i, \quad \tilde{s}_k^{\,2} = \sum_{i \in Y_k} (w^{\mathrm{T}} x_i - \tilde{m}_k)^2, \tag{2}$$

E and *k* represent the set of index *i* corresponding to the class *k* and N*k* is the number of training patterns of the *k* class [11].

2.5. Ensemble classifiers

The use of ensemble classifiers is a strategy where more than one classifier are combine taking into account the diversity of their answers to get a better classification performance.

There are many reasons to use ensemble classifiers among which can be mention: statistical reasons, big volumes of data, few data, divide and conquer and data fusion [4].

All of the ensemble classifiers have two key components. The first one is that a strategy to build an ensemble as diverse as possible is required, that is why fusion and selection strategies are used to combine the classifiers [4].

In second place is necessary a strategy to combine each of the classifiers outputs that integrate the ensemble in a way that the right decisions will be amplify and attenuated the wrong ones. Some available options to do this are: simple voting, weighted voting, BKS, addition rule, product rule, mean rule, minimum rule, maximum rule and weighted addition [4].

2.6. Building strategies

Bagging stands for bootstrap aggregating, that is one of the first algorithms based on ensemble. The Bagging method obtains the diversity through bootstrap copies of the training data. For each training subset is used a different classifier. These classifiers are all of the same type. The decisions of individual classifiers are combined through a majority voting. For a particular case, the class chooses by the majority of the classifiers is the decision of the ensemble.

Boosting creates a classifier ensemble by resampling the data, this strategy tends to get a more informative training set to present to the next classifier.

In essence, creates three weak classifiers: the first classifier (C1) is train with a random subset of the available data. The training subset to feed the second classifier (C2) is chosen as the most informative subset, given C1. C2 is train with a database where only half of the patterns were correctly classified by C1. The third classifier (C3) is trained with the patterns that C1 and C2 have classified from different class. The three classifiers are combined through majority voting [4].

K-means is an alternative strategy to select the training group, where the training pattern set is divide into k groups, each group is form with the patterns that are nearest to one of the *k* centroids,

sinc(*i*) Research Center for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
C. Arjona, J. Pentácolo, I. E. Gareis, Y. Atum, G. Gentiletti, R. C. Acevedo & H. L. Rufiner; "Evaluation of LDA Ensembles Classifiers for Brain Computer Interface"
Memorias del XVIII Congreso Argentino de Bioingeniería (SABI 2011), set, 2011.

these centroids represent different statistical distributions of the features, this way, there are specific classifiers for different local areas in the feature space [4,11].

2.7. Strategies to combine outputs.

Simple voting: a count of the number of votes that each classifier gives to each class is made and the class select by the ensemble is the one that gets more votes.

Weighted voting: each classifier received a weight in the final voting based on its performance during the training process.

Minimum/maximum/median rule: as the name indicates, they are simple functions to find the minimum, maximum and median of outputs of individual classifiers.

Product rule: multiply the output values of each classifier. This rule is very sensitive to the most pessimistic classifiers.

Mean rule: for each class the average of all the classifiers' output of that class is the parameter that is consider to make the final decision. The mean rule is equivalent to the sum rule (within a normalization factor of *1/T*, where *T* is the number of available patterns).

Behaviour Knowledge Space (BKS): uses a look up table, constructed based on the classification of the training data that keeps track of how often each labelling combination is produced by the classifiers. Then, the true class, for which a particular labelling combination is observed most often during training, is chosen every time that combination of class labels occurs during testing. [4].

2.8. Performance evaluation

The error rate (ER), in the most use metric as a tool to evaluate the performance of a classifier [12].

In the ROC curves the true positive rate versus the false positive rate is represent, and a bigger area under the curve (AUC) generally indicate a better performance of the classifier [13, 14].

Based on the different characteristics of ER and AUC, both performance estimators are use in this work for the ensemble classifiers.

Where the Error Rate is:

$$ER = \frac{Nerror}{Nts}$$

Nerror is number of misclassifications tested registers and *Nts* is the total number of tested registers.

and AUC was calculate:

$$AUC = \sum_{th}^{\max th} S(th) * [Sp(th+1) * Sp(th)]$$

S is sensibility in function of threshold (*th*) and *Sp* specificity in function of threshold too.

3. Experiments

As it was mention before, patterns with a fix number of features were generated, resulting in a 64-dimension space.

The experiment series has 20 iterations, and the results were averaged to valid and evaluated the performance.

An individual classifier (control) and the three groups of classifiers were trained, on iterations. Bagging, Boosting and a variation of Bagging with k-means clustering with three centroids methods were used. The parameters that were modified in the experiments were the threshold; the patterns training sets; the patterns test sets and the method of output combination.

Threshold: the classification threshold is simultaneously modified for each of the LDA classifier of the ensemble to generate a ROC curve for then calculate the AUC.

Pattern training set: the 80% of the available patterns were randomly chosen, this process was repeated for each iteration.

Pattern test set: the 20% of the patterns that not form the training set were used to test the classifiers.

4. Results

In table 1 and 2 are shown the results obtain using the registers of subject 1 and 2 respectively. There you can find the values of AUC and ER of the experiments.

Table 1. Experiments results with subject's 1 registers.

Subject 1	Simple voting	Weighted voting	BKS	Product rule	Addition rule	Weighted addition rule	Maximum rule	Minimum rule	Mean rule
Type	AUC	AUC		AUC	AUC	AUC	AUC	AUC	AUC
Boosting	0,81	0,81	-	0,81	0,81	0,81	0,80	0,80	0,81
Bagging	0,82	0,82	-	0,82	0,82	0,82	0,82	0,82	0,82
Bagging kmeans	0,79	0,79	-	0,80	0,80	0,80	0,80	0,80	0,79
Single LDA	0,82								
Type	ER	ER	ER	ER	ER	ER	ER	ER	ER
Boosting	26,75%	26,75%	26,85%	28,10%	27,70%	26,92%	29,96%	29,96%	26,75%
Bagging	25,32%	25,32%	25,54%	25,30%	25,28%	25,42%	25,04%	25,04%	25,32%
Bagging kmeans	29,09%	29,09%	28,59%	28,57%	28,47%	28,49%	29,27%	29,27%	29,09%
Single LDA	24,82%								

Table 2. Experiments results with subject's 2 registers.

Subject 2	Simple voting	Weighted voting	BKS	Product rule	Addition rule	Weighted addition rule	Maximum rule	Minimum rule	Mean rule
Type	AUC	AUC		AUC	AUC	AUC	AUC	AUC	AUC
Boosting	0,88	0,88	-	0,88	0,88	0,88	0,87	0,87	0,88
Bagging	0,89	0,89	-	0,89	0,89	0,89	0,89	0,89	0,89
Bagging kmeans	0,87	0,79	-	0,79	0,88	0,79	0,87	0,80	0,87
Single ADL	0,89								
Type	ER	ER	ER	ER	ER	ER	ER	ER	ER
Boosting	20,58%	20,58%	20,60%	21,11%	20,93%	20,48%	21,94%	21,94%	20,58%
Bagging	18,99%	18,99%	18,99%	19,13%	19,13%	19,13%	19,19%	19,19%	18,99%
Bagging kmeans	22,58%	23,81%	23,75%	23,35%	22,44%	23,83%	23,83%	23,83%	22,58%
Single LDA	18,75%								

The figures 1 and 2 belong to subject's 1 registers and figures 3 and 4 belong to subject's 2 registers. In these figures are resume graphically the data from tables 1 and 2. Based on all the obtain information, it can be seed that the performance of the classifiers trained by Bagging method are very near to the performance of the individual classifiers. The results using other training method as Boosting and Bagging k-means show a less performance than the methods mention before.

sinc(*i*) Research Center for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
C. Arjona, J. Pentácolo, I. E. Gareis, Y. Atum, G. Gentiletti, R. C. Acevedo & H. L. Rufiner; "Evaluation of LDA Ensembles Classifiers for Brain Computer Interface"
Memorias del XVIII Congreso Argentino de Bioingeniería (SABI 2011), set, 2011.

sinc(*i*) Research Center for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
C. Arjona, J. Pentácolo, I. E. Gareis, Y. Atum, G. Gentiletti, R. C. Acevedo & H. L. Rufiner; "Evaluation of LDA Ensembles Classifiers for Brain Computer Interface"
Memorias del XVIII Congreso Argentino de Bioingeniería (SABI 2011), set, 2011.

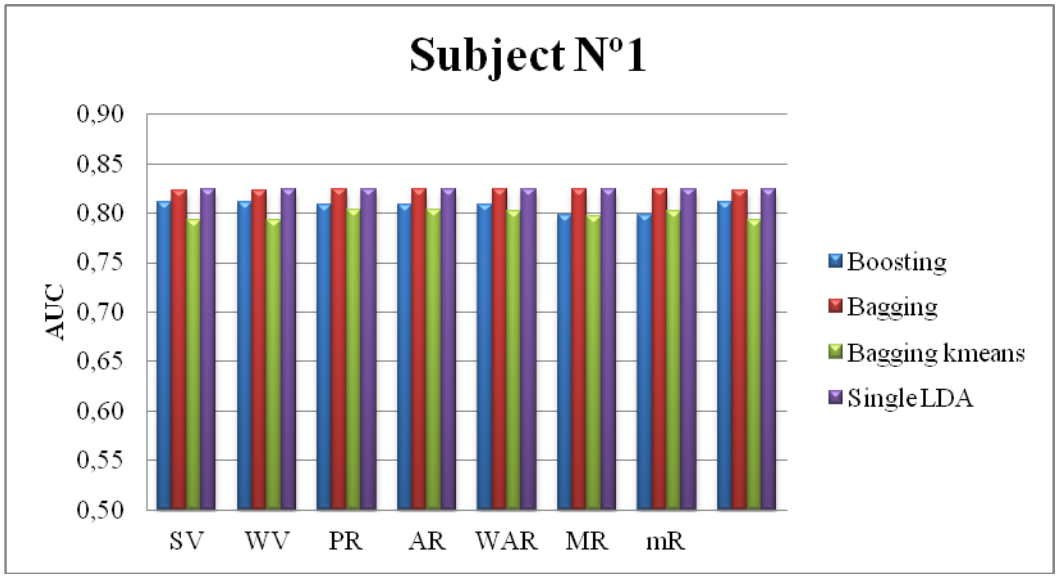

Figure 1. AUC vs. Output combination methods for subject Nº 1, where in the horizontal axis are represented the different output combination methods: simple voting (SV), weighted voting (SW), product rule (PR), addition rule (AR), weighted addition rule (WAR), maximum rule (MR), minimum rule (mR), mean rule (meR).

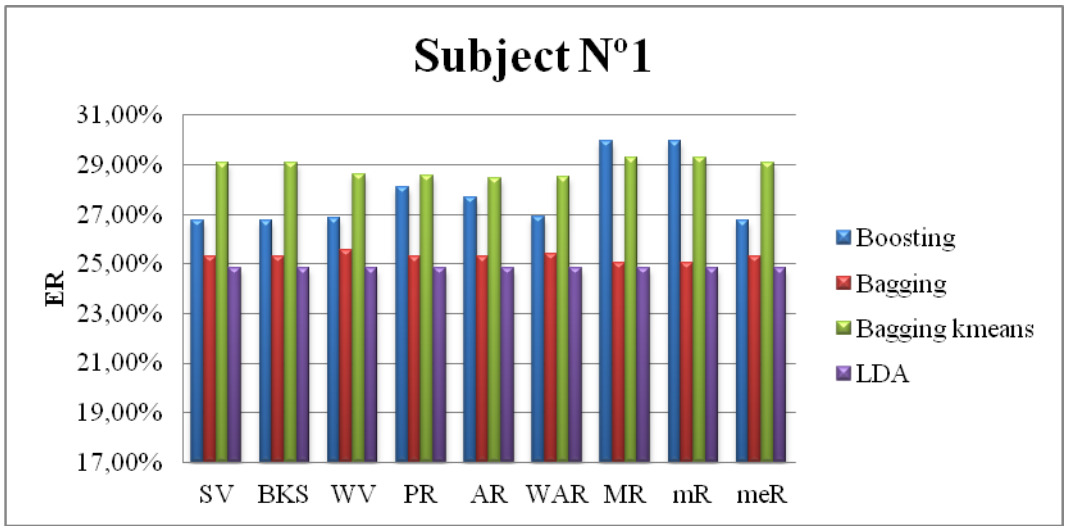

Figure 2. ER vs. Output combination methods for subject Nº 1, where in the horizontal axis are represented the different output combination methods: simple voting (SV), weighted voting (WP), product rule (PR), addition rule (AR), weighted addition rule (WAR), maximum rule (MR), minimum rule (mR), mean rule (meR).

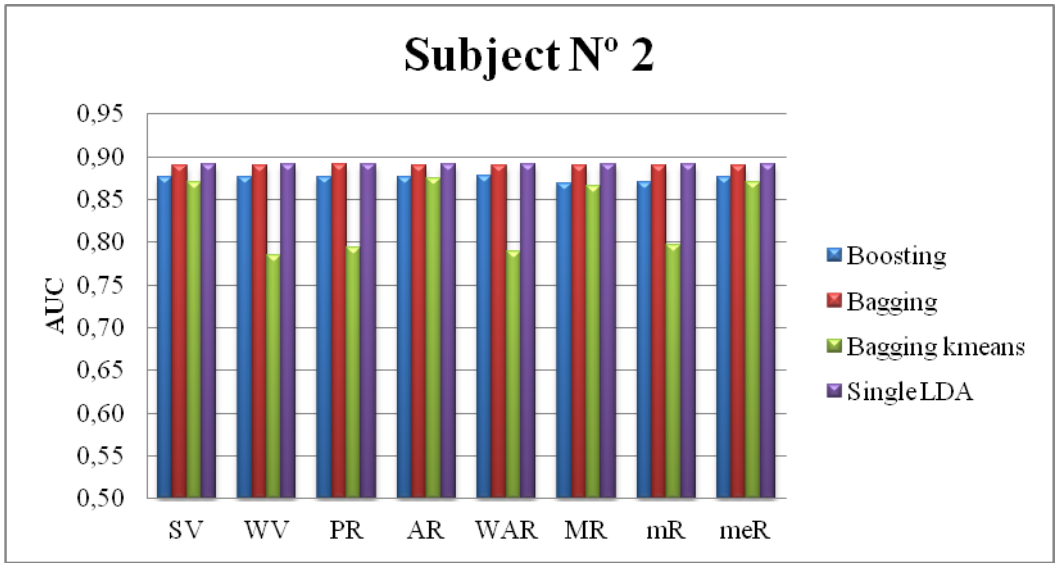

Figure 3. AUC vs. Output combination methods for subject Nº 2, where in the horizontal axis are represented the different output combination methods: simple voting (SV), weighted voting (WV), product rule (PR), addition rule (AR), weighted addition rule (WAR), maximum rule (MR), minimum rule (mR), mean rule (meR).

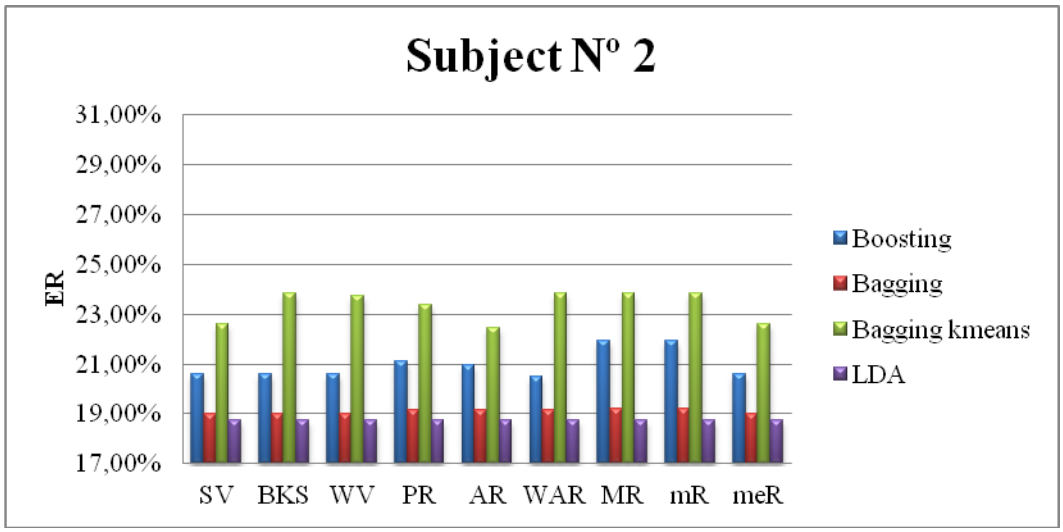

Figure 4. ER vs. output combination methods for subject Nº 2, where in the horizontal axis are represented the different output combination methods: simple voting (SV), weighted voting (WV), product rule (PR), addition rule (AR), weighted addition rule (WAR), maximum rule (MR), minimum rule (mR), mean rule (meR).

5. Discusión

We have also evaluated the statistical significance of these results by computing the probability that a given ensemble classifier is better than single LDA classifier. In order to perform this test we assumed the statistical independence of the classification errors for each register and we approached the errors' Binomial distribution by means of a Gaussian distribution. This is possible because we have a sufficiently high number of registers for each subject (3780). In this way, for Subject 1 and Subject 2 we have that the confidence of the relationship obtained between error rates of simple LDA (ref) and

each ensemble, ie Pr (Err ensemble > Err ref <), is detailed in Table 3 for each ensemble and each combination of outputs.

Table 3. Percentage of confidence of the results of each ensemble with respect to single LDA.

Confidence Subject 1	Voting	BKS	W. V.	P.R.	S.R.	W. S. R.	Max. R.	Min. R.	Med. R.
Boosting	97,23%	97,23%	97,23%	99,96%	99,85%	97,23%	100,00%	100,00%	97,23%
Bagging	69,92%	69,92%	80,78%	69,92%	69,92%	69,92%	56,91%	56,91%	69,92%
Bagging Kmeans	100,00%	100,00%	99,99%	99,99%	99,99%	99,99%	100,00%	100,00%	100,00%
Confidence Subject 2									
Boosting	97,23%	97,23%	97,23%	99,55%	98,82%	97,23%	99,96%	99,96%	97,23%
Bagging	56,91%	56,91%	56,91%	69,92%	69,92%	69,92%	69,92%	69,92%	56,91%
Bagging Kmeans	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%

From the result analysis, it can be observed that the use of ensemble classifiers instead of individual ones, seems to not generate relevant improvements in the classification performance of ERP using ER and AUC as index of performance. It is estimated that this situation is due to the lack of diversity of the classifiers that form the ensemble, and to the statistical distribution of the features of the used patterns. It should be emphasized that the results of this work were generated with patterns of only one epoch.

Although these results do not lead to a better solution to the classification of ERP problem, they orient the search to the study of other types of classifiers with diversity in the classification method, to be use in ensembles, as are support vector machines (SVM), multilayer perceptrons (MLP) or decision trees (DT). This work could also guide the search to other preprocessing techniques that make a different mapping of the feature space where the separation between classes is maximized. As an extension to this work it is presented as an option to explore the separation of the feature space into subspaces, using the strategy “divide and conquer”.

6. Conclusions

In this work probes with ensemble classifiers of LDAs were performed taking basic algorithms as are Bagging and Boosting, also probes using different output combination methods and patterns preselection techniques, could be carried out.

When all the variations, proposed in this work were performed and the AUC and ER were calculated. What could be seen after that was a little different in the performance indexes between ensemble and single classifiers. Although it was done a variation of the training parameters in many ways, the results do not change.

Acknowledgments

This work was supported by the Universidad Nacional de Entre Ríos (grant PID 6101), the Universidad Nacional de Litoral (grant CAI+D 012-72) and the Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) from Argentina.

References

[1] Wolpaw J R, Birbaumer N, McFarland D J, Pfurtscheller G, and Vaughan T M 2002 Brain Computer Interfaces for Communication and Control. *Clin. Neurophysiol*. **113**, no. 6 767-91.

[2] Kübler A, Kotchoubey B, Kaiser J, Wolpaw J R and Birbaumer N 2001 Brain–computer communication: unlocking the locked in. *Psychol. Bull*. **127** 358–75.

[3] Bashashati A, Fatourechi M, Ward R K and Birch G E 2007 A survey of signal processing

algorithms in brain-computer interfaces based on electrical brain signals. *Journal of Neural Eng.* **4** 32-57.
[4] Polikar R 2006 Ensembe Based Systems in Decision Making. *IEEE Circuits and System Magazine.* **6** 21-40.
[5] Lotte F, Congedo M, Lecuyer A, Lamarche F and Arnaldi B 2007 A review of classification algorithms for EEG-based brain-computer interfaces. *Journal of Neural Engineering*. **4** 1-13.
[6] Krusienski D J, Sellers E W, Cabestaing F, Bayoudh S, McFarland D J, Vaughan T M and Wolpaw J R 2006 A Comparison of Classification Techniques for the P300 Speller. *Journal of Neural Eng.* **3** 299-305.
[7] Farwell L A and Donchin E 1988 Talking off the top of your head: toward a mental prothesis utilizing event-related brain potentials. *Electroenceph. clin. Neurophysiol*. **70** 510.
[8] http://www.bci2000.org/BCI2000/Home.html.
[9] Ledesma Ramírez C, Bojorges Valdez E, Yañez Suárez O, Saavedra C, Bougrain L and Gentiletti G 2010 *An open-access P300 speller database*. Fourth international BCI meeting, Paper L-12, Monterrey, California.
[10] Gareis I E, Acevedo R C, Atum Y V, Medina Bañuelos V, Rufiner H L and Gentiletti G G 2011 Efecto de la cantidad y dimensión de los patrones en una interfaz cerebro computadora basada en LDA, Proceedings of V Latin American Congress on Biomedical Engineering (CLAIB2011), La Habana, May 16 – 21, 2011.
[11] Duda R O, Hart P E and Stork D G 2000 *Pattern Classification (2nd Edition*) Wiley-Interscience (Estados Unidos) pp 117-24 526-28
[12] Weiss G M 2004 Mining with rarity: a unifying framework. *SIGKDD Explor.* **6** 7.
[13] Zweig M H and Campbell G 1993 Receiver-Operating Characteristic (ROC) Plots: A Fundamental Evaluation Tool in Clinical Medicine. *Clinical Chemistry.* **39** 561.
[14] Bradley A P 1997 The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition* **30** 1145.

sinc(*i*) Research Center for Signals, Systems and Computational Intelligence (fich.unl.edu.ar/sinc)
C. Arjona, J. Pentácolo, I. E. Gareis, Y. Atum, G. Gentiletti, R. C. Acevedo & H. L. Rufiner; "Evaluation of LDA Ensembles Classifiers for Brain Computer Interface"
Memorias del XVIII Congreso Argentino de Bioingeniería (SABI 2011), set, 2011.