ORIGINAL RESEARCH

Anomalous pattern based clustering of mental tasks with subject independent learning – some preliminary results

Renato Cordeiro de Amorim[1], **Boris Mirkin**[1], **John Q. Gan**[2]

1. School of Computer Science and Information Systems, Birkbeck, University of London, UK. 2. School of Computer Science and Electronic Engineering, University of Essex, UK.

Correspondence: Renato Cordeiro de Amorim. Address: School of Computer Science and Information Systems, Birkbeck, University of London, Malet Street, London WC1E 7HX, UK. Tel: 440-207-631-6746. E-mail: renato@dcs.bbk.ac.uk

Received: March 26, 2012 **Accepted:** May 20, 2012 **Published:** September 1, 2012
DOI: 10.5430/air.v1n1p55 **URL:** http://dx.doi.org/10.5430/air.v1n1p55

Abstract

In this paper we describe a new method for EEG signal classification in which the classification of one subject's EEG signals is based on features learnt from another subject. This method applies to the power spectrum density data and assigns class-dependent information weights to individual features. The informative features appear to be rather similar among different subjects, thus supporting the view that there are subject independent general brain patterns for the same mental task.

Classification is done via clustering using the intelligent k-means algorithm with the most informative features from a different subject. We experimentally compare our method with others.

Key words

EEG, Clustering, Feature extraction, Intelligent K-Means

1 Introduction

Wide research has shown that electroencephalography (EEG) signals contain useful information about the state or intention of the mind [1-4]. They are considered to be one of the best non-invasive approaches to acquiring information for classifying mental tasks [5].

EEG signals may provide an individual with an alternative channel for communication with the external environment [2, 5, 6]. It could be the only possibility to communicate with other people, if the individual is completely motor-paralyzed but has intact sensory and cognitive brain functions (locked-in-syndrome). In this case, the communication could be conducted via a brain computer interface (BCI). Other interesting applications of EEG signals include the diagnosis of neurological disorders and other abnormalities of the human body [1, 2], and even monitoring the depth of anesthesia [7].

The understanding of EEG signals may provide a number of benefits, but their processing is far from being trivial. Indeed each electrode records the activity of thousands of neurons simultaneously [5], which makes EEG recording very noisy, and thus EEG patterns are difficult to discern.

Although most of the noise is supposed to come from either within the brain or over the scalp [1], the truth is that there can be many other sources of noise, sometimes rather significant, such as eye movement, muscle activity, cardiac activity, respiration and skin potential. Also, even if the pure biological EEG source were to be noise free, amplification and digitalization would add noise (systematic bias) [3].

More irregularities in the EEG patterns may also be generated by the use of devices such as mobile phones, as research suggests [8, 9] exposure to pulse modulated electromagnetic fields generated by them may affect the cerebral blood flow in certain areas of the brain.

Signal averaging is a well-known technique used to reduce noise by smoothing the data. This also allows estimation of the amplitude of signals that may be buried in noise, which involves the following, not necessarily realistic, assumptions [3]:

- The signal and the noise are uncorrelated;
- The timing of the signal is known;
- A consistent signal component exists to be extracted using repeated measurements;
- The noise is truly random with zero mean.

The averaging technique has proven sufficiently robust to survive minor violations of the above assumptions and it is currently used by researchers [2, 4].

Another problem faced by EEG signal processors is that classifying these signals is an intrinsically high dimensional task [10]; a recording of one hour using 128 electrodes at 500 samples per second would generate around 0.45 GB [1].

In this paper we propose a method to cluster mental tasks based on anomalous patterns. Our proposed method has a pre-processing step that averages the power spectrum density data belonging to the same class. This allows us to find information weights of the most informative pixels and use solely these for clustering, reducing the noise and dimensionality of the data.

Our experiments show our method to have two main advantages: (i) it enjoys a good level of accuracy compared to other algorithms; (ii) the weights can be reasonably similar between different subjects, so we can apply our method in a subject independent learning framework. The algorithm learns the information weights from one subject but cluster classes from another.

2 Description of the data and the power spectrum density

In this paper we analyze the EEG data of two healthy subjects. Their datasets were constructed by capturing 250 samples of data per second, under five bipolar electrodes. These electrodes followed the extended 10-20 system, a common standard for their position, using, cz to pz, fc1 to pc1, fc2 to pc2, fc3 to pc3, and fc4 to pc4.

The use of this rather small number of electrodes has proven successful in our previous work [11], in spite of suggestions that a higher quantity of electrodes could facilitate classification [6, 12].

We have set the length of a trial to be 8 seconds, and in each of these trials we instruct the subject to imagine a body movement (task). The subjects do not perform any actual body movement, but simply imagine performing one of the tasks below:

- (T1) Moving left hand,
- (T2) Moving right hand,
- (T3) Moving feet.

This set of tasks will be denoted as Ω = {T1, T2, T3}, and the two subjects from which the data have been collected, A and B. We have recorded 240 trials from Subject A and 120 trials from Subject B respectively.

We have generated the power spectrum density (psd) of the original raw data, producing 71 time samples, each with 80 features. The features consist of the spectrum over 8~45 Hz. Using 5 electrodes and 16 frequency bands, we get 80 features.

The interesting point of the above method is that now, each trial can be seen as an image of 71x80 pixels, and the datasets are then a set of such images. We refer to these images as trials in the remainder.

In order to average the trials per task, take a set S of N trials Si (*i*=1, 2, …, N), and tasks ω=T1, T2, T3. We partition S, so that

$$S_{\omega} = {}^{1}/_{N_{\omega}} \sum_{i \epsilon S(\omega)} S_i \tag{1}$$

where N_{ω} is the number of trials in S(ω).

3 The method

Feature extraction is especially important in EEG classification because of the rather high dimensionality of the datasets (71×80) and the fact that some features may be misleading for classification purposes.

We follow a common structure to most BCI techniques [10], and perform the classification only after the feature extraction. Here the classification is made via clustering.

3.1 Feature extraction

Since EEG data are likely to have a certain degree of noise, assuming that this noise is truly random, it can be minimized by averaging the trials belonging to the same tasks by using formula (1), which leads to three averaged trials S_{ω}, one for each of the tasks ω∈ Ω.

In the data used for experiments, each trial $s \in S$ has 5680 pixels (71×80), so that it becomes important to identify a small group of pixels to be used as features in the follow-up clustering.

In order to create such a group one needs a measure of importance of a pixel (m, n) (m=1,…,71; n=1,…,80) for classification purposes. We measure the distance of a pixel in trial *s*∈ S to task ω∈ Ω by comparing its brightness with the brightness of the corresponding pixel in the averaged trial S_{ω}:

$$d(s(m,n),\omega) = |\mathrm{s(m,n)} - \mathrm{S}_{\omega}(\mathrm{m,n})|^2 \tag{2}$$

We refer to the pixel as being good in trial s if the difference of brightness (2) is smaller to the mean of the subset S(ω) that has the correct label. The goodness value for pixel (m, n) is defined then as the proportion of trials in which it is good. After computing the goodness values for all pixels, one can use them as the importance weights.

Moreover, our experiments have shown that clustering results can be improved if pixels with low goodness values are discarded. Hence we introduce a threshold θ such that all pixels whose goodness value is less or equal than θ are removed from the process of classification. The features remaining after removal of all those pixels whose goodness is less or equal than θ =0.43 can be seen in Figures 1 (a) and (b) for subjects A and B respectively. These figures show the pixels found as being the most important ones for classification in the two subjects. It can be seen that the pixels before second 4 in each trial tend to be unimportant to the classification (check the y-axis, about the 35th time point), psds over 8~17Hz (the first five frequency bands in each channel) are most useful, and all the 5 channels provide important pixels and thus make contributions to the classification (check the x-axis).

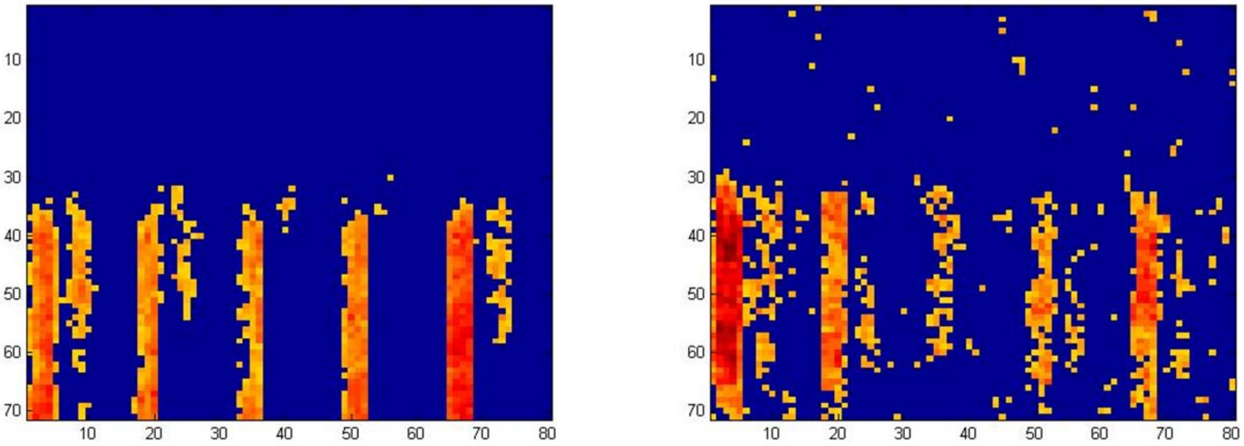

Figure 1. The features extracted for subjects *A* (a, left) and *B* (b, right). The brightness of a given pixel represents the proportion of trials for which it was close to the correct average trial S_ω

The Jaccard coefficient comparing both sets of features is equal to 0.452. The feature extraction algorithm is as follows:

0. Initialize all goodness values g(m, n)=0.

1. Calculate the mean of all trials per task S_ω using formula (1).

2. For each pixel (m, n) in each trial s.

    2.1 Calculate its distance to the same pixel in S_ω using formula (2).

    2.2 If the distance is of pixel $(m,n) \in s$ is closer to its peer pixel in the correct average trial S(ω), increase its goodness value g(m, n) by 1.

3. Update the goodness values by dividing them by the total number of trials and remove all pixels whose goodness values are smaller than θ (i.e., set g(m,n) as 0).

3.2 Classification via clustering

Clustering is a widely used method [13-18] and K-means is probably its most well known algorithm. With it one can partition data into K non-overlapping clusters $S=\{S_1, S_2,\ldots, S_K\}$ without the need of previous learning. Each of these clusters is represented by a centroid c_k and defined by the set of entities that are closer to it rather than to other centroids. The objective is to minimize the sum of the within-cluster distances to the centroids as in formula 3.

$$W(S,C) = \sum_{k=1}^{K} \sum_{i \in S_k} d(i, c_k) \tag{3}$$

where $d(i, c_k)$ represents the distance between an instance and the centroid c_k. The K-means algorithm is highly sensitive to the initial centroids it uses, and these are normally chosen at random. There are a considerable amount of algorithms that attempt to deal with this issue, and Intelligent K-means (iK-Means) [17] is a viable option. The original version of iK-Means outperforms many other algorithms used to identify good initial centroids [18].

IK-Means works as a pre-step to K-means. After data normalization it uses the farthest away entity from the centre of gravity as an initial centroid and cluster entities against it and the centre of gravity itself. These centroids are updated in relation to their clusters and their final values are used as initial centroids in K-means. The algorithm can be described then as follows:

1) Identify the farthest away entity from the centre of gravity as an initial centroid

2) Cluster entities using the minimum distance rule between the centroid or the centre of gravity

3) Update the centroid to the average of its cluster

4) Repeats 2 and 3 until stabilizes

5) Repeats 1 to 4 until there are no more entities to cluster

6) Removes any centroid whose cluster size is smaller than a pre-specified parameter, which we set to 1.

7) Runs K-means using the found number of clusters and respective initial centroids.

Intelligent K-means can also be utilized to find the number of centroids. In our case this number is known, and for this reason we set the parameter in step six to 1. We prefer to simply select as initial centroids the 3 which produced the largest clusters and then run K-means. To normalize the psd data we subtract it by its average and then divide by half its range.

In our method we apply the iK-means to the data using only the features whose weight is higher than a pre-specified threshold θ. The labels generated by iK-Means are then mapped to the known tasks by using a confusion matrix

4 Experimental results

In our algorithm the classification of one subject's EEG signals is based on features learnt from another subject. We find the results we present here rather interesting and they suggest that subject independent features could be used to successfully classify mental tasks.

Another possibility would be to use our method as a starting point for subject dependent learning. Algorithms following the latter approach may take a considerable amount of time to learn what features are meaningful. We believe one could use our method to select what features those algorithms should evaluate first.

As we know the labels of all trials per subject we measure the accuracy of our algorithm, by simply dividing the correctly labeled tasks by its total number and then multiply by 100. These results can be seen in table 1.

Table 1. The accuracy of our proposed algorithm using an optimal threshold

Learnt from	Applied to	Threshold *θ*	Accuracy
A	B	0.44	75.8%
B	A	0.43	59.2%

We have tested our algorithm using different thresholds as one can see in figure 2.

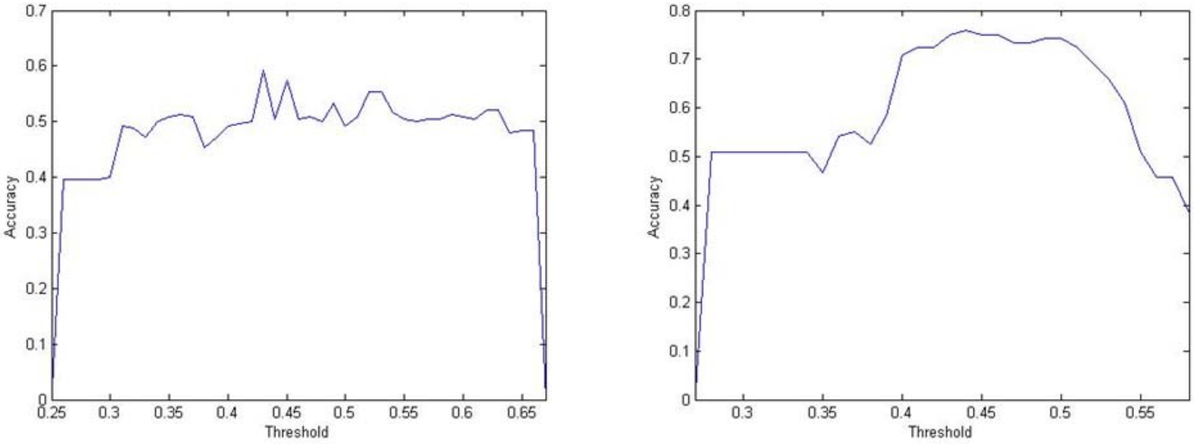

Figure 2. The accuracy of our algorithm with different thresholds for subject *A* and *B* respectively.

From these we gather that it may be possible to have a generally good threshold that would work well in most scenarios. We show these results in table 2.

Table 2. The accuracy of our proposed algorithm using a generally good threshold

Learnt from	Applied to	Threshold θ	Accuracy
A	B	0.43	75.0%
B	A	0.43	59.2%

5 Comparative results

We have experimented using other algorithms with the normalized psd datasets for subjects *A* and *B* under the same framework of subject independent learning. A summary of our experiments is presented in Table 3.

Table 3. Summary of experiments

Algorithm	Learnt From	Applied to	(%)	Parameter
Pixel Selective iK-Means	A	B	75.0	*θ* =0.43
Decision Tree	A	B	50.0	-
Knn classify	A	B	63.3	K=17
Naïve Bayes	A	B	62.5	-
Random Forest	A	B	60.8	$\sqrt{5680}$
Pixel Selective iK-Means	B	A	59.2	θ =0.43
Decision Tree	B	A	58.3	-
Knn classify	B	A	55.8	K=13
Naïve Bayes	B	A	59.2	-
Random Forest	B	A	59.2	$\sqrt{5680}$

5.1 Decision tree

We compare our method to Decision Trees. Here we use the standard decision tree [19] present in MATLAB's Statistics Toolbox. The accuracy results are as below:

- Learnt from *A* applied to *B*: 50.0%
- Learnt from *B* applied to *A*: 58.3%

5.2 Nearest neighbor

In our second comparison we experiment with the K-Nearest neighbor algorithm [20]. We have experimented with several values for K going from 1 to 40. The accuracy results are presented below.

- Learnt from *A* applied to *B*: 63.33% (K=17)
- Learnt from *B* applied to *A*: 55.83% (K=13)

We find it potentially problematic that the optimal K for both subjects is so different. In our view the experiments suggest that K is a subject dependent parameter that has to be found.

5.3 Naïve Bayes

Our third comparison is to the popular Naïve Bayes classifier. This classifier treats features independently.

- Learnt from A apply to B: 62.50%
- Learnt from B apply to A: 59.17%

5.4 Random Forest

Lastly, we compare with Random Forest [21]. In the setting we have used a total of 50 trees and selected $\sqrt{5680}$ features at random, as suggested, for each decision split.

Learnt from A apply to B: 60.83%

Learnt from B apply to A: 59.17%

6 Conclusion and discussion

In this paper we proposed a clustering method involving a measure of the goodness of pixels as both the pixel weighting coefficient and the pixel rejection base.

Even taking into account that the results of EEG classification heavily depend on subjects, as was pointed out by other researchers [5], it can be seen in Figures 1 (a) and (b) that, although subjects *A* and *B* and provide for very different accuracy rates, their good pixels are somewhat similar. This suggests that there may be subject independent general brain patterns for the same tasks.

Our experiments suggest that these subject independent features can be used in clustering and by consequence classification. We also suggest that these features could be used as a starting point for a more subject dependent feature selection method. We hope that this would speed up the process of learning in EEG classification algorithms.

One can notice, too, that a good classification is not a matter of having less sparse goodness value tables: subject *A* even having a much denser feature clusters than subject *B* has led to poorer result.

At this stage we find that we could attempt to get more meaning from the feature weights in instead of using a crisp threshold for classification, and that this meaning could be task dependent. These points will be address in our future research.

References

[1] Sanei, S., Chambers, J. A. EEG Signal Processing. WileyBlackwell. 2007.

[2] Ungureanu, M., Bigan, C., Strungaru, R., Lazarescu, V. Independent component analysis applied in biomedical signal processing, Measurement Science Review. 2004; 4(2): 1-8.

[3] Drongelen, W. V. Signal Processing for Neuroscientists: An Introduction to the Analysis of Physiological Signals, Academic Press/Elsevier. 2007.

[4] Geng, T., Gan, J. Q., Dyson, M., Tui, C. S. L. and Sepulveda, F. A novel design of 4-class BCI using two binary classifiers and parallel mental tasks, Journal of Computational Intelligence and Neuroscience. 2008. http://dx.doi.org/10.1155/2008/437306

[5] Lee, F., Scherer, R., Leeb, R., Neuper, C., Bischof, H., Pfurtscheller, G. A Comparative analysis of multi-class EEG classification for brain computer interface, Proceedings of the 10th Computer Vision Winter Workshop. 2005; 195-204.

[6] Peterson, D., Knight, J., Kirby, M. Anderson, C., Thaut, M. Feature selection and blind source separation in EEG-based brain-computer interface, EURASIP Journal on Applied Signal Processing. 2005; 19: 3128-3149. http://dx.doi.org/10.1155/ASP.2005.3128

[7] Ortolani, O., Conti, A., Di Filippo, A., Adembri, C., Moraldi, E., Evangelisti, A., Maggini, M., Roberts, S. J. EEG signal processing in anaesthesia: Use of neural network technique for monitoring depth of anaesthesia, British Journal of Anaesthesia. 2002; 88(5): 644-648. PMid:12067000 http://dx.doi.org/10.1093/bja/88.5.644

[8] Von Klitzing, L. Low-frequency pulsed electromagnetic fields influence EEG of man, Physica Med. 1995; 11; 77-90.

[9] Huber, R., Treyer, V., Borbe, A. A., Schuderer, J., Gottselig, J. M., Landol, H. P. Electromagnetic fields, such as those from mobile phones alter regional cerebral blood flow and sleep and awaking EEG, Journal of Sleep Research. 2002; 11(4): 289-295. PMid:12464096 http://dx.doi.org/10.1046/j.1365-2869.2002.00314.x

[10] Tomioka, R., Aihara, K., Muller, K. R. Logistic regression for single trial EEG classification, Advances in Neural Inf. Proc. Systems. 2007; 19; 1377-1384.

[11] Tsui, C., Gan, J. Q., Roberts, S. A self-paced brain-computer interface for controlling a robot simulator: an online event labelling paradigm and an extended Kalman filter based algorithm for online training, Medical and Biological Engineering and Computing. 2009; 47; 257-265. http://dx.doi.org/10.1007/s11517-009-0459-7

[12] Wolpaw, J. R., McFarland, D. J., Neat, G. W., Forneris, C. A. An EEG-based brain-computer interface for cursor control, Electroencephalography and Clinical Neurophysiology. 1991; 78(3): 252-259. http://dx.doi.org/10.1016/0013-4694(91)90040-B

[13] Amorim, R. C. Constrained intelligent k-means: Improving results with limited previous knowledge. Proceedings of the Second International Conference on Advanced Engineering Computing and Applications in Sciences. 2008; 176-180. http://dx.doi.org/10.1109/ADVCOMP.2008.30

[14] Amorim, R. C. An adaptive spell checker based on PS3M: Improving the clusters of replacement words. Computer Recognition Systems 3. 2009; 57; 519-526. http://dx.doi.org/10.1007/978-3-540-93905-4_61

[15] Amorim, R. C. and Mirkin, B. Minkowski metric, feature weighting and anomalous cluster initializing in K-means clustering, Pattern Recognition. 2012; 45(3): 1061-1075. http://dx.doi.org/10.1016/j.patcog.2011.08.012

[16] Amorim, R.C. and Komisarczuk, P. On Partitional Clustering of Malware. The First International Workshop on Cyber Patterns: Unifying Design Patterns with Security, Attack and Forensic Patterns. 2012; 47-51.

[17] Mirkin, B. Clustering for Data Mining: A Data Discovery Approach, Chapman and Hall/CRC. 2005. http://dx.doi.org/10.1201/9781420034912

[18] Chiang, M. M. and Mirkin, B. Intelligent choice of the number of clusters in k-means clustering: An experimental study with different cluster spreads. Journal of Classification. 2010; 27(1): 1-38. http://dx.doi.org/10.1007/s00357-010-9049-5

[19] Breiman L., Friedman J., Olshen R., and Stone C. Classification and Regression Trees. CRC Press. 1984.

[20] Mitchell, T. Machine Learning, McGraw-Hill. 1997.

[21] Breiman L. Random Forests. Machine Learning. 2001; 45: 1; 5-32. http://dx.doi.org/10.1023/A:1010933404324