Fuzzy rule generation using data mining techniques to classify Two-Class BCI experiment

Omid Dehzangi[1], Reza Boostani[2], Mansoor Zolghadri Jahromi [2]

[1]School of Computer Engineering, Nanyang Technological University, Singapore
omid0002@ntu.edu.sg
[2]Department of Computer Science and Engineering, Shiraz University, Shiraz, Iran
{boostani, zjahromi}@shirazu.ac.ir

Abstract. In recent years, the Graz Brain-Computer Interface (BCI) has been developed and different aspects of this new field of research such as feature extraction and classification, mode of operation, mental strategy, and type of feedback have been investigated. In this paper, a Fuzzy Rule-Based Classification System (FRBCS) is presented in which a novel approach for fuzzy rule generation is proposed. The proposed algorithm makes the use of data mining principles, which are used by frequent pattern mining algorithms. Employing these principles enables us to well generate rules for subsequent classification purposes. Finally, a rule-weighting mechanism is investigated to tune the rule-base to have better classification ability. To evaluate the performance of the proposed scheme features containing standard bandpower and adaptive autoregressive coefficients are determined on four subjects in order to increase the performance of a cue-based BCI system for imagery classification tasks (left and right hand movements). As comparative classifiers, a number of successful methods of classification including Adaboost, Support Vector Machine (SVM), and Linear Discriminant Analysis (LDA) have been assessed. The results show that the proposed method of classification is effective in prediction ability of choosing between the left and right imagery tasks.

Keywords: Brain-Computer Interface, EEG Signals, Fuzzy Systems, Data mining, Rule-Weighting.

1 Introduction

In order to assist disabled people who suffer from motor impairment like amyotrophic lateral sclerosis (ALS) patients, an alternative communication channel can be provided by a brain computer interface (BCI) system [1]. BCI systems can enable the patients to move the cursor on a screen [1, 2], or grasp a glass by sending the signal commands to their orthesis, prosthesis, or functional electrical stimulation unit [3]. To increase the performance of BCI systems, much research has been done by BCI research groups. In this way, the Graz-BCI research group has employed

M. Chetty, S. Ahmad, A. Ngom, S. W. Teng (Eds.): PRIB2008,
Supplementary Conf. Proc., pp.216-226, 2008.
(c) PRIB 2008

discriminative features based on second order statistics such as bandpower [3], adaptive autoregressive coefficients (AAR) [4], wavelet coefficients [5], and also combination of features by distinction sensitive learning vector quantization (DSLVQ) [6] with well-known classifiers containing Fisher's linear discriminant analysis (FLDA) [7], linear vector quantization (LVQ) [8], minimum distance classifiers [4], etc. to improve the classification rate between the various movement imagery tasks. The aim of this paper is to develop a fuzzy rule-based classifier that improves the performance of the cue-based BCI, based on electroencephalogram (EEG) data from two motor imagery tasks in the synchronous mode. It should be considered that most BCIs operate in a cue-based (synchronous or system-driven) mode where the subject can change his or her mental task only in certain intervals that are determined by the system.

As a powerful supervised learning algorithm, fuzzy rule-based classification systems have been used successfully on pattern classification problems [9]. Basic idea for designing a FRBCS is to automatically generate fuzzy rules from numeric data (i.e., a number of pre-labeled training examples). Hence, rule-base construction for a classification problem always has been a challenging part of it. In this paper, a novel approach for generating a set of candidate rules of each class is presented using data mining principles in which the number of generated rules is reduced dramatically. Finally, a rule-weighting scheme is presented to adjust the weights of the rules in the final rule-base based on accuracy on the training data.

The rest of this paper is organized as follows. In section 2, subjects and the method of data acquisition are described. In section 3, extracted features are introduced. In section 4, the proposed method of fuzzy classification is presented. In section 5, the experimental results are discussed. Section 6 concludes the paper.

2 Subjects and Data Acquisition

Four subjects (L1, O3, O8, and G8), familiar with the Graz-BCI, participated in this study. Subjects are ranged from 25 to 35 years old. Each subject sat in an armchair about 1.5 meters in front of the computer screen. Three bipolar EEG-channels were recorded from 6 Ag/AgCl electrodes placed 2.5 cm anterior and 2.5 cm posterior to the standardized positions C3, Cz and C4 (international 10-20 system). The EEG was filtered between 0.5 and 50 Hz and recorded with a sample frequency of 128 Hz.

The training in Graz-BCI training paradigm consisted of a repetitive process of triggered movement imagery trials. Each trial lasted 8 seconds and started with the presentation of a blank screen. A short acoustical warning tone was presented at second 2 and a fixation cross appeared in the middle of the screen. At the same time, the trigger was set from 0 to 1 for 500 milliseconds. From second 3 to second 7, the subjects performed left or right hand motor imagery according to an arrow (cue) on the screen. An arrow pointing either to the left or to the right indicated the imagination of a left hand or right hand movement. The order of appearance of the arrows was randomized and at second 7 the screen content was erased. The trial finished with the presentation of a randomly selected inter-trial period (up to 2 seconds) beginning at second 8. Fig. 1 shows the timing scheme. Three sessions were recorded for each subject on 3 different days. Each session consisted of 3 runs with 40 trials each.

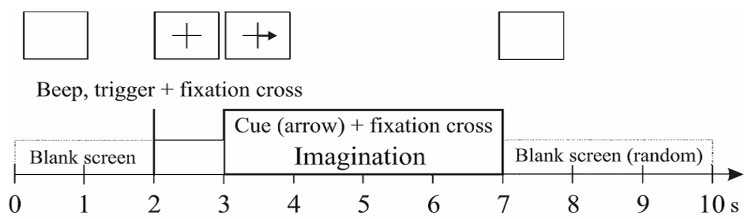

Fig. 1. Training paradigm and time scheme for each recording

3 Feature Extraction

The goal of feature extraction is to find an informative representation of the data that simplifies the detection of brain patterns. The signal features should encode the commands sent by the user. To extract discriminative features, band power and adaptive autoregressive coefficients are selected in this paper. These are described briefly below.

3.1 Band Power

The EEG contains different specific frequency bands, that is standard alpha (8-12Hz) and beta (16-24Hz) bands, which are particularly important in classifying different brain states, especially for discriminating imagery tasks. For this study band power features were calculated by applying a Butterworth filter (order 5), squaring of the samples and then averaging of subsequent samples (1 s average with 250 ms overlap) [4].

3.2 Adaptive Autoregressive Coefficients

AAR coefficients are extracted from a signal and reflect the whole variation of the signal. In contrast to the AR model, AAR coefficients are adapted sample by sample as described in [4]. Update coefficient and model order are two important factors that should be selected regarding the minimum error of the model.

4 Fuzzy rule-based classification system

In this section, the architecture of the FRBCS containing a novel approach for fuzzy rule generation and the proposed method of rule-weighting is described in detail.

4.1 Fuzzy Inference System

Let us assume that we have m training patterns $X_p = (x_{p1},\ldots,x_{pn})$, $p = 1,2,\ldots,m$ from M different classes where X_p is an n-dimensional vector of attributes in which x_{pi} is the i-th attribute value of the p-th training pattern ($i = 1,2,\ldots,n$). For our M-class, n-dimensional classification problem, we use fuzzy if-then rules of the form below:

Rule R_q: If x_1 is A_{q1} and … and x_n is A_{qn} then class C_q with CF_q **(1)**

where, R_q is the label of the q-th fuzzy if-then rule, $X = (x_1,\ldots,x_n)$ is n-dimensional vector of a pattern, A_{qi} presents an antecedent fuzzy set, C_q is a class label, and CF_q is the weight assigned to the q-th rule. To calculate the compatibility grade of each training pattern X_p with the antecedent part of the rule $\mathbf{A}_q = (A_{q1},\ldots,A_{qn})$, we use the product operator as follows:

$$\mu_{\mathbf{A_q}}\left(X_p\right) = \mu_{A_{q1}}\left(x_{p1}\right)\cdot\mu_{A_{q2}}\left(x_{p2}\right)\cdot\cdots\cdot\mu_{A_{qn}}\left(x_{pn}\right),\ p = 1,2,\ldots,m \tag{2}$$

where, $\mu_{Aqi}(x_{pi})$ is the compatibility grade of x_{pi} with fuzzy membership function A_{qi}. To determine the consequent class of the q-th rule C_q, we measure the confidence degree of the association rule "$\mathbf{A_q} \Rightarrow$ Class h" from the field of data mining for each class, where $\mathbf{A_q}$ is a multi dimensional fuzzy set representing the antecedent conditions and h is a class label. Confidence of a fuzzy association rule R_q is defined as follows [10]:

$$c\left(\mathbf{A_q} \Rightarrow class\,h\right) = \sum_{X_p\in class\,h}\mu_{\mathbf{A_q}}\left(X_p\right)\Big/\sum_{i=1}^{m}\mu_{\mathbf{A_q}}\left(X_i\right),\ h = 1,2,\ldots,M \tag{3}$$

where, $\mu_{A_q}(X_p)$ is the compatibility grade of pattern X_p with the antecedent part of the rule R_q calculated in (2), m is the number of training patterns and C_q is a class label. The class with maximum confidence degree is identified to determine the consequent class C_q:

$$q = \arg\max\left\{c\left(\mathbf{A_q} \Rightarrow class\,h\right) \mid h = 1,2,\ldots,M\right\} \tag{4}$$

An input pattern is classified regarding to the consequent class of the winner rule. By using rules of the form (1), a weight assigned to each rule is used to find the winner rule. Rule weighting has a profound effect on the classification ability of FRBCSs [11]. In recent years [12, 13], several methods of rule weighting have been introduced. In this paper, we use a learning mechanism to find the weight of each rule. The winner rule R_w is chosen for the input pattern X_t in the following manner:

$$w = \arg\max_{q}\left\{\mu_{\mathbf{A_q}}\left(X_t\right)\cdot CF_q \mid q = 1,2,\ldots,N\right\} \tag{5}$$

Where, N is the number of rules in the rule-base and CF_q is the weight assigned to the q-th rule. Note that the classification of a pattern not covered by any rule in the rule-base is rejected. The classification of a pattern X_t is also rejected if two rules with different consequent classes have the same value of $\mu_{\mathbf{A_q}}\left(X_t\right)\cdot CF_q$ in (5).

4.2 Rule-base Construction

A simple approach for generating fuzzy rules is to partition the domain interval of each input attribute using a pre-specified number of fuzzy sets (i.e., grid partitioning). Given a partitioning of pattern space, one approach for fuzzy rule generation is to consider every possible combination of antecedents to generate the fuzzy rules. The problem with grid partitioning is that an appropriate partitioning of each attribute is not usually known. One solution is to simultaneously consider different partitions, as shown in Fig. 2. That is, for each attribute, one of the 14 fuzzy sets shown in Fig.2 can be used when generating a fuzzy rule.

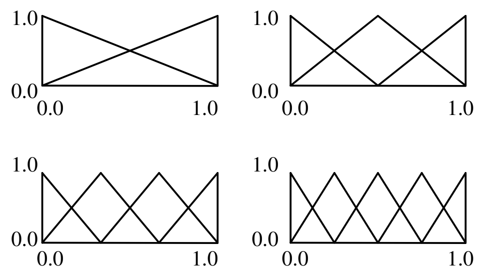

Fig. 2. Different partitioning of each feature axis.

The problem is that for an n-dimensional problem, 14^n antecedent combinations have to be considered. It is impractical to consider such a huge number of antecedent combinations when dealing with high dimensional problems. One solution for the above problem is presented in [10] by adding the fuzzy set "*don't care*" to each attribute. The membership function of this fuzzy set is defined as $\mu_{don't\ care}(x) = 1$ for all values of x. The trick is to not consider all antecedent combinations (which are now 15^n) and only short fuzzy rules having a limited number of antecedent conditions are generated as candidate rules. For example, fuzzy rules having only two or less antecedent fuzzy sets (excluding don't care) are investigated.

It seems that ignoring majority of possible antecedent combinations for rule generation would degrade the accuracy of the FRBCSs. On the other hand, increasing the limitation of two or less antecedent fuzzy sets may be impossible in some cases. This limitation prevents generating a number of useful rules, which would have positive effects on the classification accuracy, to be present in the rule-base.

The purpose of the solution presented in this paper is to avoid the exponential growth of the rule-base and yet generate rules having more antecedents. In this approach, we do not generate rules that are hardly probable to be interesting. The method is based over two data mining principles, used for mining frequent item sets:

1) Increasing the length of an item set, the support value will not improve.
2) A set of n items is probable to be frequent (have a good support), if and only if all of its subsets of size n-1 are frequent (the Apriori principle) [14].

The common usage of the above principles is in data warehouses, to find item sets with good supports (i.e., set of items that have frequently occurred together). In this work, we observe them from a different viewpoint and use them to find fuzzy rules having good supports.

4.3 Generating Fuzzy Rules

Major purpose in this paper is to propose a solution that enables us to generate fuzzy rules with any number of antecedents, i.e., There would be no restriction on the number of antecedents especially for high dimensional problems. For this purpose, we consider the well-known evaluation measure, *Support* as the primary factor for rule filtering. In the following, a simple definition for the fuzzy aspect of the *Support* measure is presented.

$$\text{s}(A_q \Rightarrow \text{Class h}) = \frac{1}{m} \sum_{X_p \in Class\ h} \mu_{\mathbf{A_q}}(X_p) \tag{6}$$

Where, $\mu_{\mathbf{A_q}}(X_p)$ is the compatibility degree of X_p with the antecedent part of the rule R_q, m is the number of training patterns and h is a class label. After determining a minimum support threshold (denoted by *MinSupp*), a set of 1-dimensional rules (containing one antecedent), is generated. This set is then filtered by selecting only rules having a support value above the *MinSupp*. Combining the rules within this set in the next step, results in the set of 2-dimensional candidate rules. The reason of this issue (that we just combine rules having good supports) refers to the first principle mentioned in Section 2. In other words, the 1-dimensional rules which are pruned through the first step because of their bad supports, can not lead to 2-dimensional rules with good supports and thus there is no need to consider them. Another key point in combination of a pair of 1-dimensional rules is the conditions under which the rules can be combined:

1) The rules must not contain similar antecedents on their left-hand sides. 2) The consequent classes of the two rules must be identical. Similarly, the resulting rule-base is filtered with respect to the *MinSupp* value. Note that the rules being selected according to their higher support values will be fine-tuned according to their effectiveness in the next step. The rule-weighting mechanism will be discussed later.

In order to generate rules containing more than two antecedents, a similar procedure is followed. However, in this case, both of the principles (used for mining frequent item sets) must be regarded. Generating 3-dimensional rules is accomplished using the 1 and 2-dimensional candidate rules. Any possible combination of the rules from these two sets, having the same consequent and not containing common antecedents would be a 3-dimensional candidate rule. The second principle is used to avoid the time-consuming evaluation of some useless rules (which can not have high support values). A rule resulting from a combination will be evaluated only if all of its 2-dimensional sub-rules are present in the candidate set of the previous stage (i.e., all the sub-rules have good supports). Otherwise, we do not measure the support of the rule, since it can not be even a candidate rule. Similarly, generating an n-dimensional rule is performed by the combination of n-1 and 1-dimensional candidate rules.

4.4 Rule Weighting Mechanism

For a specific problem, assume that a rule-base is constructed using the mechanism described in previous Sections. Our aim in this section is to propose a rule-weight specification method based on ROC analysis. Consider the following rule as a typical rule of the system.

Rule R_k: If x_1 is A_{k1} and … and x_n is A_{kn} then class T with CF_k **(7)**

In order to use the rule-weighting mechanism, a 2-class situation is formed by denoting the consequent class of the rule as positive and all other classes as negative. That is, in rule weighting mechanism given in this section, class p denotes the consequent class of the rule under investigation and class n is formed by merging all other classes. The contrast of each training pattern X_t covered by the rule R_k is defined as:

$$Contrast\left(X_t\right)=\frac{\mu_n\left(X_t\right)}{\mu_n\left(X_t\right)+\mu_p\left(X_t\right)} \quad \textbf{(8)}$$

where $\mu_p\left(X_t\right)$ denotes the compatibility grade of pattern X_t with the rule R_k and $\mu_n\left(X_t\right)$ is defined as:

$$\mu_n\left(X_t\right)=\max\left\{\mu_j\left(X_t\right) \mid R_j \in RuleSet, C_j \neq \text{Class } p\right\} \quad \textbf{(9)}$$

Where, C_j is the class label of the rule j. The contrast of pattern X_t calculates the degree to which pattern X_t is compatible to the rules of merging all other classes rather than the rule R_k (i.e. Contrast is considered as a dissimilarity measure between the pattern X_t and the rule R_k). Using Eq. (8), the contrast of each training pattern will be in the range of 0 to 1. In the extreme case, if pattern X_t is not in the covering area of the rule R_k and it is fired by at least one of the rules from the negative class, the calculated contrast will be 1. On the other hand, if pattern X_t is fired by the rule R_k and it is not in covering area of any of the rules from the negative class, the contrast is 0. In order to specify the weight of fuzzy rule R_k, in the first step, the contrast measures corresponding to the patterns in the covering area of the rule R_k are calculated. In the next step, the best threshold on the contrast resulting in maximum accuracy is calculated. That is, the proposed rule-weighting scheme finds the best threshold by maximizing *Accuracy*= *TP*[1]-*FP*[2]. An algorithm for doing this is given in Fig. 3.

This algorithm receives a set of patterns X_t and their contrast measures $Cont(X_t)$ as input and returns the best threshold as output. The value of the best threshold is simply used as the weight of the rule (i.e., CF_k = best-threshold). Notice that with contrast measures being in the interval [0, 1], the weight assigned to each rule will remain in this interval.

---

[1] True Positive

[2] False Positive

(1) Given: patterns X_t (t=1,…, N_{remain}), contrasts $Cont(X_t)$
(2) best-th = 0
(3) current = classification accuracy corresponding to the threshold of th = 0 (i.e., classifying everything as negative)
(4) optimum = current
(5) rank the patterns in ascending order of their contrasts
{assume that X_k and X_{k+1} are two successive patterns in the list}
(6) for k=1 to N_{remain}
  (7) Let threshold th =($Cont(X_k)$+ $Cont(X_{k+1})$)/2
  (8) current = accuracy corresponding to the specified threshold (i.e., all patterns X_t having $Cont(X_t) < th$ are classified as positive)
  (9) if current > optimum then
    (10) optimum = current
    (11) best-th = th
  (12) end if
(13) end for
{assume that ε is a small positive number}
(14) current = accuracy corresponding to $th = Cont(X_{N_{remain}}) + \varepsilon$ (i.e., classifying everything as positive)
(15) if current > optimum then
  (16) optimum = current
  (17) best-th = th
(18) end if
(19) return best-th

Fig. 3. Algorithm for finding the best threshold.

5 Computational experiments

At first, EEG signals from four trained subjects (L1, O3, O8, and G8) were recorded. Out of 360 trials recorded for each subject, 240 trials were used in the train and the rest in the test phase. AAR and band power features were extracted from the signals and Adaboost, SVM, LDA, and our proposed FRBCS were used as the classifiers (i.e. one of the features is applied to one of the classifiers), which referred to as single feature classifier. In the train phase, significant features were selected using average accuracy on the validation set by ten times ten folds cross validation. In this way, a classifier is trained with the features of all trials in each 250 ms through the paradigm. Our paradigm is 8sec.; therefore, we have different 32 feature sets for all trials. Cross validation is performed on each feature set and the best feature set according to the minimum error rate is selected. Classifiers, trained with the best feature set for each subject, were then used to classify test feature vectors.

To evaluate the proposed method, we first normalized each attribute to the interval [0,1]. Using the method of section 4, we generated fuzzy rules with different antecedent conditions and constructed an initial rule-base. The minimum support threshold was determined by cross-validation over the training data. In our experiments, minimum support of 0.3 led to the best performance on the four subjects. Finally, the rule-base is fine-tuned using the rule-weighting mechanism which attempts to maximize the accuracy on the training data.

In Fig. 4, a comparison has been made between the base-line rule generation technique presented in [10] and our proposed rule generation scheme using data mining principles. Using the base-line rule generation, we are able to generate fuzzy rules having only three or less antecedent fuzzy sets.

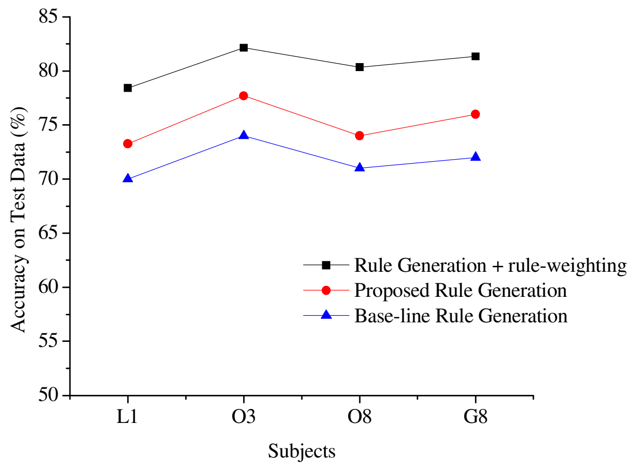

Fig. 4. Accuracy on test data of different subjects using base-line rule generation technique, the proposed rule generation scheme, and by further adjusting the rule-weights.

As it can be seen, the results show the supremacy of the proposed rule generation scheme over the base-line. Fig. 4 also illustrates the effectiveness of the rule-weighting mechanism by which further improvement in classification accuracy is achieved.

Results achieved by applying all the classifiers and features on test data of the 4 subjects are shown in Table 1. The results of other classifiers in Table 1 are taken from a recent work by Boostani et. al. [15] in which different combinations of classifiers and features were compared. Our experimental set up and data acquisition are matched with their experiments.

Table 1. Minimum test error rates using different classifiers on the four subjects.

Subject	Feature	Adaboost	SVM	LDA	proposed FRBCS
L1	BP	33.33	27.38	28.57	27.46
	AAR	22.62	26.19	26.19	24.33
O3	BP	24.66	19.8	9.59	18.86
	AAR	30.14	30.14	28.77	26.35
O8	BP	20.03	21.90	17.14	18.57
	AAR	25.71	27.62	23.81	22.85
G8	BP	20.71	16.43	16.43	18.67
	AAR	27.14	30.00	22.14	22.65

In Fig. 5, the average error rates of each method in combination with Band power features and adaptive autoregressive coefficients on four different subjects are shown.

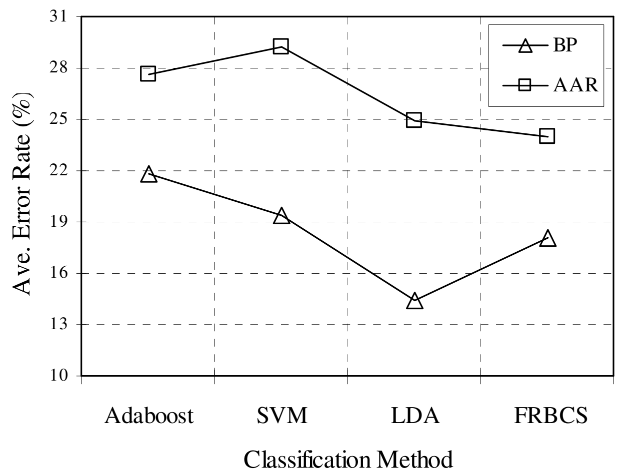

Fig. 5. Average error rates on four subjects obtained by applying compared methods in combination with Band power features and AAR Coefficients.

It can be observed that in case of using Band power features, the proposed FRBCS performs better than Adaboost and SVM, but LDA method outperforms the others in this case. In case of using AAR features, it can be seen that the proposed FRBCS has the highest average classification accuracy among the other classifiers. It is inferred that our fuzzy approach has a good compatibility with the AAR features to distinguish between the left and right imagery tasks compared to LDA, SVM, and Adaboost.

6 Conclusion

In this paper, we proposed a novel fuzzy classification approach to classify the left and right imagery tasks in the cue-based BCI. Our goal was to improve classification accuracy of two-class BCI experiment. A novel method of rule-base construction using data mining principles and a rule weighting mechanism was presented. Using the proposed method for rule generation, it will be possible to efficiently generate rules having different lengths for subsequent classification purposes. Then, the rule-weighting mechanism was used to fine-tune the generated rules. In order to evaluate our method, it has been applied on the band power and AAR features of two imagery tasks of four subjects (L1, O3, O8, and G8) participated in this study. The results showed that the proposed method is effective to achieve good accuracies to choose between the left and right tasks compared to LDA, SVM, and Adaboost.

Acknowledgments. We would like to express our sincere thanks and appreciation to Dr. Jinyan Li (NTU) for his invaluable guidance, support and suggestions. His knowledge and suggestions helped me in this research work.

References

1. G. Pfurtscheller, C. Neuper, Motor imagery and direct Brain-Computer communication, Proc. of the IEEE, 89, 1123-1134 (2001)
2. J.R.Wolpaw, D.J. McFarland, T.M. Vaughan, Brain-Computer interface research at the Wadsworth Center. IEEE Trans Rehab Eng, 8, 222-226 (2000)
3. J. Kalcher, D. Flotzinger, G. Pfurtscheller, A new approach to a Brain-Computer-interface (BCI) based on learning vector quantization (LVQ3), Proc Annu Int Conf IEEE, 4, 1658-1659 (1992)
4. G. Pfurtscheller, C. Neuper, A. Schlogl, K. Lugger, Separability of EEG signals recorded during right and left motor imagery using adaptive autoregressive parameters. IEEE Trans on Rehab Eng, 6, 316--328 (1998)
5. G. Pfurtscheller, S. Lopes, Event related desynchronization. Hand book of electroenceph. and clinical Neurophisiology. 6, revised edition. Elsevier, Amsterdam, (1999)
6. D. Flotzinger, M. Pregenzer, G. Pfurtscheller, Feature selection with distinction sensitive learning vector quantization and genetic algorithms. IEEE Int Conf Comput Intell, 6, 3448-3451 (1994)
7. E. Haselsteiner, G. Pfurtscheller, Using time-dependent neural networks for EEG classification. IEEE Trans on Rehab Eng, 8, 457-463 (2000)
8. B. Obermaier, C. Neuper, C. Guger, G. Pfurtscheller, Information transfer rate in a five-classes Brain-Computer interface. IEEE Rehab Eng, 9, 283-288 (2001)
9. J. A. Roubos, M. Setnes, J. Abonyi, Learning fuzzy classification rules from labeled data, Information Sciences, 150, 77-93 (2003)
10. H. Ishibuchi, T. Yamamoto, Comparison of heuristic criteria for fuzzy rule selection in classification problems, Fuzzy Optimization and Decision Making, 3, 119-139 (2004)
11. H. Ishibuchi, T. Yamamoto, Rule Weight Specification in Fuzzy Rule-Based Classification systems, IEEE Transaction on Fuzzy Systems, 13, 428-435 (2005)
12. M. J. Zolghadri, E. G. Mansoori, Weighting fuzzy classification rules using receiver operating characteristics (ROC) analysis. Inf. Sci. 177, 2296-2307 (2007)
13. M. J. Zolghadri, M. Taheri, A proposed method for learning rule weights in fuzzy rule-based classification systems. Fuzzy Sets and Systems 159, 449-459 (2008)
14. R. Agrawal, R. Srikant, Fast Algorithms for Mining Association Rules in Large Databases, Proc. of the 20th nternational Conference on Very Large Data Bases, 487-499 (1994)
15. R. Boostani, B. Graimann, M.H. Moradi, G. Pfurtscheller, A Comparison Approach toward Finding the Best Feature and Classifier in Cue-Based BCI. Journ Medic. Bio. Eng. Comp. 6, 403-412 (2007).