New KF-PP-SVM classification method for EEG in brain-computer interfaces

Yang Banghua[*], Han Zhijun, Zan Peng and Wang Qian

Department of Automation, School of Mechatronic Engineering and Automation, Shanghai Key Laboratory of Power Station Automation Technology, Shanghai University, Shanghai 200072, China

Abstract. Classification methods are a crucial direction in the current study of brain-computer interfaces (BCIs). To improve the classification accuracy for electroencephalogram (EEG) signals, a novel KF-PP-SVM (kernel fisher, posterior probability, and support vector machine) classification method is developed. Its detailed process entails the use of common spatial patterns to obtain features, based on which the within-class scatter is calculated. Then the scatter is added into the kernel function of a radial basis function to construct a new kernel function. This new kernel is integrated into the SVM to obtain a new classification model. Finally, the output of SVM is calculated based on posterior probability and the final recognition result is obtained. To evaluate the effectiveness of the proposed KF-PP-SVM method, EEG data collected from laboratory are processed with four different classification schemes (KF-PP-SVM, KF-SVM, PP-SVM, and SVM). The results showed that the overall average improvements arising from the use of the KF-PP-SVM scheme as opposed to KF-SVM, PP-SVM and SVM schemes are 2.49%, 5.83 % and 6.49 % respectively.

Keywords: Kernel fisher, posterior probability, support vector machine, brain computer interface, common spatial pattern

1. Introduction

Brain–computer interfaces (BCIs), developed in the early 1960s, are used to create a direct channel of communication between one subject's brain and one computer, without a traditional muscle-dependent pathway [1]. An electroencephalogram (EEG), an electrical signal collected from the scalp, is now widely used in the field of BCIs mainly due to its excellent temporal resolution, ease of use, portability and low set-up cost. The EEG signals are analysed and processed through several mathematical techniques to extract useful information represented in the form of feature vectors. The feature vectors are then recognised and translated into meaningful control commands. Research on EEG feature extraction and pattern classification, has become a pressing topic [2–4].

Motor imagery, a kind of spontaneous EEG, is now widely applied in BCIs, because it is more natural than using an evoked EEG. Currently there is plenty of literature about motor imagery EEG recognition in BCIs, from its feature extraction to its classification. Some typical feature extraction methods include common spatial pattern (CSP) [5,6], autoregressive parameters and wavelet packet decomposition. Among these methods, CSP has been proven to be an effective feature extraction method by

---

[*]Corresponding author: Yang Banghua, Department of Automation, Shanghai University, Yan Chang Road 149, Shanghai 200072, China. Tel.: +86 21 56331278; Fax: +86 21 56334241; E-mail: yangbanghua@shu.edu.cn.

many researchers and is widely used in BCIs [7]. The classifier is concerned with the final output and determines the final recognition effect. This paper mainly focuses on the classification of motor imagery EEG in BCIs. The most commonly used algorithms for classification are linear discriminant analysis [8], artificial neural networks (ANN) [9], support vector machines (SVM). SVM is a proven powerful classification method and performs better than other classifiers for EEG classification [10].

Although SVM has been widely used in BCI systems, there are some drawbacks, for example, some deviations will occur in the presence of intense noise or when the distribution of data is uneven. The main reason for such deviations is that traditional SVM does not take into consideration the global properties of the class distribution. As an effective global algorithm, the kernel fisher (KF) captures the essential characteristics of the data distributions (mean and covariance) from the kernel training data, and then estimates the decision boundary using these global data characteristics. Thus KF can overcome these defects in SVM [11]. At the same time, the posterior probability (PP) is important in the design of a classifier, as it can ensure a more stable classifier [12,13]. The present work proposes the KF-PP-SVM (kernel fisher, posterior probability, support vector machine) algorithm, which is based on the idea of combining SVM with a fisher from Riadh [11], and combines PP at the same time.

1.1. PP-SVM

Developed from the theory of structural risk minimisation, SVM is one of the best classifiers since it finds the hyperplane maximising the separating margin between classes [14–16]. The least square SVM (LS-SVM) is an expansion of the SVM and is superior in its simplicity and reduced demand for computing resources. For simplification, the term SVM refers to LS-SVM in the present work. The essence of SVM is the following optimization

$$\min \ (\tfrac{1}{2})w^T I w + \gamma(\tfrac{1}{2})\sum_{i=1}^{n+m} e_i^2 \tag{1}$$

$$s.t. \quad y_i[(w \cdot \phi(x) + b] = 1 - e_i \ \ e_i \geq 0, i = 1, 2, ..., n + m$$

where I is the identity matrix, the regularisation term $\gamma \geq 0$, $n + m$ is the number of training samples, $\{e_i\}_{i=1}^{n+m}$ is the error vector, $y_i \in \{-1, 1\}$ is the associated label of the training sample, $\phi(x)$ is a mapping function from data space R^d to feature space R^{d_f}, the vector $w \in R^{d_f}$ and scalar $b \in R$ are the weighting vector and bias of SVM respectively. According to Eq. (1), this optimization problem can be solved by the following Lagrange multipliers

$$\begin{bmatrix} 0 & -Y^T \\ Y & zz^T + \gamma^{-1}I \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{2}$$

$$z = [\phi(x_1)^T y_1, ..., \phi(x_{m+n})^T y_{m+n}]$$
$$Y = [y_1, ..., y_{m+n}]$$
$$I = [1, ..., 1], \alpha = [\alpha_1, ..., \alpha_{m+n}]$$

where $(zz^T)_{ij} = y_i y_j k \ (x_i, x_j)$, α is the dual variable vector, and this function $k(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ is called a kernel function. From Eq. (2), α and b can be obtained (α is non-zero). It should be noted that samples that make α non-zero are called support vectors, which are used to establish final decision function. Then the decision function can be shown as follows:

$$f(x) = \sum_{i=1}^{n+m} \alpha_i y_i k(x_i, x) + b \tag{3}$$

According to Eq. (3), the classification surface of classical SVM only depends on those two classes' boundary samples and misclassified samples. The general output model of the SVM classifier is expressed as follows:

$$z(x) = \begin{cases} 1 & f(x) > 0 \\ 2 & f(x) < 0 \end{cases} \tag{4}$$

where $z(x)$ gives a value of 1 or 2 after classification to illustrate the final decision, without giving the probability value of a sample belonging to either class. The idea of PP-SVM was that the output $f(x)$ of the SVM could be described by a best-fit sigmoid function, rather than directly judged by whether $f(x)$ is greater than zero or not. Clearly, if the classification decisions were based on complete information about the PP distribution function, the expected risk could be minimised and meanwhile the stability would remain better than that obtained with SVM. Platt reassumed the PP model as a sigmoid function with parameters

$$p(y = 1 \mid f(x)) = \frac{1}{1 + \exp(Af(x) + B)} \tag{5}$$

where $p(y = 1 \mid f(x))$ represents the probability of the given sample x belonging to Class 1. The parameters A and B can be obtained by maximum likelihood estimation [17]. Then obtained parameters A and B are substituted into Eq. (5) to get the PP value of a sample belonging to one class. The final decision function is

$$i = \arg \max_i p(i \mid x) \quad i \in \{1, 2\} \tag{6}$$

where i represents the class label, x represents a sample. PP-SVM has thus been formed.

1.2. KF

KF discriminant analysis (KFDA) uses a kernel trick to map input data into an implicit feature space. In such a way, the non-linear relationship inherent in the input data can be converted into a linear form [18]. Fisher's linear discriminant is found by maximizing $J(w)$:

$$J(w) = \frac{w^T M w}{w^T N w} \qquad \text{where} \qquad N = \sum_{j=1,2} k_j (I - 1_{l_j}) k_j^T \ .$$

where N refers to the within-class scatter matrix, k_j is the matrix of $(n+m)*n$ or $(n+m)*m$, $k_j = k(x_{n+m}, x^j)$ is the kernel function matrix of class j, m is the number of samples of the first class, n is the number of samples of the second class, 1_{l_j} is the matrix in which all elements are n-1 or m-1, w is the transform vector, and M is the distance between classes. The KFDA has achieved outstanding effects on practical data. Besides, this method is insensitive to noise because it makes full use of all sample information.

2. KF-PP-SVM classification algorithm

2.1. Basic scheme of KF-PP-SVM

Considering that SVM ignores within-class scatter, and at the same time to improve the stability of a classifier, the present work proposed a novel classification method named KF-PP-SVM which combined a kernel fisher, SVM, and posterior probability. The basic scheme is shown in Figure 1. A feature vector can be obtained by using CSP feature extraction method. The within-class scatter is calculated according to the description in Section 1.2. Then within-class scatter is added into SVM to reconstruct a new kernel function. The output of SVM whose kernel has been changed is further decided by PP. The detailed principle and realization procedure can be seen in Sections 2.2 and 2.3.
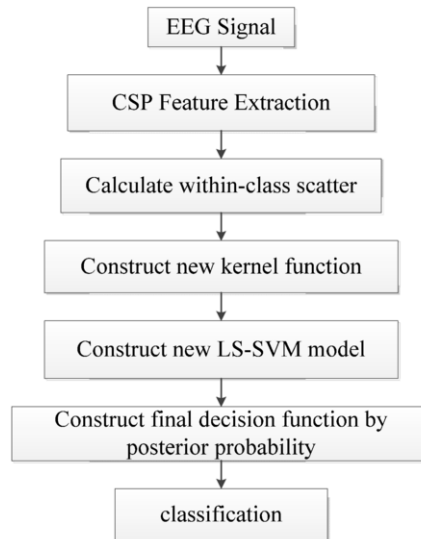

Fig. 1. Specific processes of KF-PP-SVM.

2.2. Principle of KF-PP-SVM

With the KF, the SVM optimisation problem can be described as follows:

$$\min\ (\tfrac{1}{2})w^T(CN+I)w+\gamma(\tfrac{1}{2})\sum_{i=1}^{n+m}e_i^2 \tag{7}$$

$$s.t.\quad y_i[(w\cdot\phi(x))+b]=1-e_i\ \ e_i\geq 0, i=1,2,3...,n+m$$

In a manner similar to the PP-SVM, according to Eq. (7), this optimization problem can be solved by using Eq. (2), but with $(zz^T)_{ij}=y_iy_jk^*(x_i,x_j)$, this kernel function has already been changed after adding within-class scatter to the kernel function. Now, $k^*(x_i,x_j)=\phi(x_i)*\Sigma^{-1}*\phi(x_j)^T$, where $\Sigma=CN+I$, C is parameter of within-class scatter N and it is a scalar. To obtain $k^*(x_i,x_j)$, according to the Mercer condition, the following deduction can be obtained: suppose input space $x=x_1,x_2,...,x_n$ and $k(x,y)$ is a symmetric function, for all samples, this matrix can be given by: $k=(k(x_i,x_j))\,(i,j=0,1,....,n)$, obviously this was a symmetric matrix. So there must be an orthogonal matrix to make $P^TkP=\Lambda$, where Λ is a diagonal matrix made up of eigenvalue λ_i, and the eigenvector of λ_i is $v_t=(v_{t1},v_{t2},...,v_{tn})^T$, where n is the sample size. The input space is mapped as follows:

$$\phi:x_i\rightarrow(\sqrt{\lambda_1}v_{1i};\sqrt{\lambda_2}v_{2i};....,\sqrt{\lambda_n}v_{ni})\in R^n\,(i=1,2,...,n) \tag{8}$$

where $\varphi_i(x_j)=\sqrt{\lambda_i v_{ij}}$. Then

$$\phi(x_i)=(\phi_1(x_i),\phi_2(x_i),....,\phi_n(x_i)) \tag{9}$$

$$<\phi(x_i),\phi(x_j)>=\sum_{t=1}^{n}\lambda_t v_{ti}v_{tj}=k(x_i,x_j) \tag{10}$$

$$k^*(x_i,x_j)=\phi(x_i)*\Sigma^{-1}*\phi(x_j)^T \tag{11}$$

So k^* was obtained and new values of α and b were obtained from Eq. (2). According to this new function and its values of α and b, f(x) can be calculated as follows:

$$f(x)=\sum_{i=1}^{n+m}\alpha_i y_i k^*(x_i,x)+b \tag{12}$$

At this time, the classification result may be obtained when a new sample x was continuously entered. The final decision was obtained by the posterior probability (Eq. (5)) p of the new sample belonging to Class 1. At the same time, the posterior probability 1 - p of the new sample belonging to Class 2 can be obtained. The final class label was determined by Eq. (6). Compared with Eq. (3) for the SVM, the decision function in Eq. (12) for the KF-PP-SVM was changed.

2.3. KF-PP-SVM realization procedure

To better explain the construct of the KF-PP-SVM, the procedures are summarised:

Step 1 : Let $Z = (x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ be a set of training samples, where x_i represents the feature vectors, and $y_i \in \{-1, 1\}$ represents classification labels.

Step 2 : Calculate the within-class scatter N by $N = \sum_{j=1,2} k_j (I - 1_{l_j}) k_j^T$, where I is a unit vector, and k_j is a kernel function matrix of the class j.

Step 3 : The within-class scatter was added into the kernel function $k(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ by $k^*(x_i, x_j) = \phi(x_i) * \Sigma^{-1} * \phi(x_j)^T$, where $\Sigma = cN + I$.

Step 4 : The old kernel function of SVM was replaced by the new kernel function and a new SVM model was thus constructed.

Step 5 : The final classification labels were calculated by $y_i = \arg \max_i p(y_i | x) \ y_i \in \{-1, 1\}$.

3. Experimental results

To evaluate the effectiveness of the proposed KF-PP-SVM method, the data recorded from authors' laboratory experiments were processed. At the same time, different classification methods including the SVM and the PP-SVM were used to verify the KF-PP-SVM.
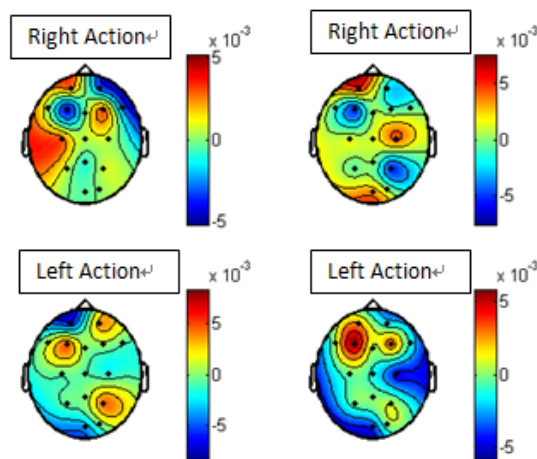

Fig. 2. Spatial patterns of motor imagery EEG.

3.1. Preparation

Dataset was from the authors' laboratory experiments. The authors recorded EEG signals using a 16-channel electrode cap. The EEG amplifier was a high-precision biological amplifier developed by Tsinghua University. The EEG signals were transformed by a 24-bit A/D converter and then collected through EEG signal acquisition software. The sampling frequency was 100 Hz. In this experiment, each of seven healthy subjects was asked to complete 60 trials in one session. Each trial included a 4 s left or right hand imagination task. There were two sessions for each subject. And two data sets (120 trials) for each subject were obtained.

The CSP method is used to extract features based on a decomposition of the raw EEG signals into spatial patterns, which are extracted from two classes of a single trial EEG. These patterns maximise the difference between classes. Figure 2 shows spatial patterns of motor imagery EEG extracted by the CSP method under four dimensional features. It can be seen from Figure 2 that weights in motor cortex areas were larger than those elsewhere. At the same time, the distribution of weights of right hand movement and left hand movement revealed differences especially in the motor cortex region. It should be noted that the typical number of projections (2, 4, 6, 8, ..., 20) of CSP were investigated to build the feature vector, demonstrating that the optimal number is 4. Therefore, a 4-dimensional feature vector is taken in this paper. Considering that KF-PP-SVM was a relatively new algorithm, for the sake of comparison, three classifiers including SVM, PP-SVM and KF-SVM were used in the experiments.

3.2. Results, analysis and discussion

The time duration from 2.1 s to 3.1 s for each trial was selected as the processing period from the total 4 s imagination-based task trial. In this way, the length of every trial was 1 s and 120×4 (120 trials, four dimensional features) were input to the classifier. During the experiment, different training data and testing data were tested by the SVM, PP-SVM, KF-SVM and KF-PP-SVM. It should be noted that KF-SVM is a simplification of KF-PP-SVM. The difference between KF-SVM and KF-PP-SVM is that the former does not need to calculate PP. Table 1 lists the average accuracies of seven subjects for four different classification methods. It should be noted that the well-known Radial Basis Function (RBF) kernel is used in the four methods and all results in Table 1 are also obtained with this kernel function. The punish parameter C in SVM is selected based on experience during training procedure. These 120 trials for each subject are divided into four equal groups (B1, B2, B3, and B4) according to time sequence of obtaining trials. That is also to say that the first 30 trials of 120 trials is grouped as B1, the second 30 trials as B2, the third 30 trials as B3 and the last 30 trials as B4. All trials in the same group are continuous in time. The groups before the "-" symbol represented training samples: those after represented test samples. Test data were classified with SVM, KF-SVM, PP-SVM and KF-PP-SVM respectively. The overall average improvement using the KF-PP-SVM as compared to KF-SVM, PP-SVM and SVM was 2.49 % (74.17 % - 71.68%), 5.83 % (74.17 % - 68.34 %) and 6.49 % (74.17 % - 67.68 %) respectively. This was because the KF-PP-SVM contained within-class scatter that may overcome some noise leading to shifting within-class. The KF-PP-SVM had better data processing ability in the face of intense noise, which was appropriate for practical BCIs. To further verify that KF-PP-SVM had a better classification performance than KF-SVM, PP-SVM and the SVM, five-fold cross-validation was performed for these four classification methods. The last line in Table 1 shows that the KF-PP-SVM has a higher accuracy than KF-SVM, PP-SVM and SVM, which is consistent with average results. In addition, power spectral density (PSD) was taken to verify the proposed

Table 1

Classification accuracies with four methods

Group Combination	Classification accuracy (%)			
	SVM	PP-SVM	KF-SVM	KF-PP-SVM
B1-B3	60.0	60.0	63.7	70.0
B1-B2B3	58.3	58.3	67.8	71.67
B3-B4	60	60	64.33	66.67
B3-B1	70	70.6	73.8	80
B3-B2	60	66.67	65.2	66.67
B3-B1B2	71.67	68.33	68.33	73.33
B3B4-B3	83.33	86.67	85.7	86.67
B3B4-B1B2	73.33	76.67	76.77	76.67
B3B4-B2B3	76.67	81.67	79.5	80
Mean ± Std	67.68 ± 7.97	68.34 ± 9.1	71.68± 7.70	74.17 ± 5.84
Five-cross Validation	70.86	71.52	74.11	75.73

method during our research. Compared with CSP, a similar result can be obtained by using PSD. Theoretically, the KF-PP-SVM should do better than PP-SVM for any group. However, different kernel functions and the parameter C of within-class scatter also affect the classification result. In this paper, the classification results were obtained according to the well-known radial basis kernel function and empirical parameter C. Further optimization of kernel function and parameter C will be performed in the future to improve the classification performance for any data group.

4. Conclusion

A new classification method combining the kernel fisher, posterior probability, and the SVM was presented. It took advantage of the properties of the kernel fisher and overcame some defects inherent to the SVM. At the same time, the method used posterior probability as a final decision function, which increased the stability of the classifier. The proposed method was verified by classification accuracy. The results showed that the proposed method could obtain high recognition accuracy. It may be feasible for practical application in BCIs. Future research should be aimed at attempts to generate more experimental data to verify the algorithm more effectively as well as at the provision of an improved theoretical understanding thereof.

Acknowledgment

References

[1] J.R. Wolpaw, N. Birbaumer and W.J. Heetderks, Brain-computer interface technology: A review of the first international meeting, IEEE Transactions on Rehabilitation Engineering **8** (2000), 164–173.

[2] D.D. Baron, J.M. Ramirez and M. Baker, A BCI motor imagery experiment based on parametric feature extraction and fisher criterion, 22nd International Conference on Electrical Communications and Computers, 2012, 257–261.

[3] W.Y. Hsu, EEG-based motor imagery classification using enhanced active segment selection and adaptive classifier, Computers in Biology and Medicine **6** (2011), 633–639.

[4] S. R. Liyanage, C.T. Guan and H.H. Zhang, Dynamically weighted ensemble classification for non-stationary EEG processing, Journal of Neural Engineering **10** (2013), 1–10.

[5] H. Ramoser, G. Muller and G.J. Pfurtscheller, Optimal spatial filtering of signal trial EEG during imagined hand movement, IEEE Transactions on Rehabilitation Engineering **8** (2000), 441–446.

[6] Q.B. Zhao, L.Q. Zhang, C. Andrzej and J. Li, Incremental common spatial pattern algorithm for BCI, Proceedings of the International Joint Conference on Neural Networks, 2008, 2656–2659.

[7] K.P. Thomas, C.T. Guan, L. Chiew Tong and A.P. Vinod, A study on the impact of spectral variability in brain-computer interface, Proceedings of IEEE International Symposium on Circuits and Systems, 2010, 1189–1192.

[8] C. Vidaurre, M. Kawanabe1, P.von Bunau and B. Blankertz, Toward an unsupervised adaptation of LDA for brain-computer interfaces, IEEE Transactions on Biomedical Engineering **58** (2011), 587–597.

[9] M.K. Hazrati and A. Erfanian, An online EEG-based brain-computer interface for controlling hand grasp using an adaptive probabilistic neural network, Medical Engineering and Physics **32** (2010), 730–739.

[10] M. Kaper, P. Meinicke and U. Grossekathoefer, BCI competition 2003-data set II: Support vector machines for the P300 speller paradigm, IEEE Transactions on Biomedical Engineering **51** (2004), 1073–1076.

[11] R. Ksantini, B. Boufama and I.S. Ahmad, A new combined KSVM and KFD model for classification and recognition, 2010 Fifth International Conference on Digital Information Management, 2010, 188–193.

[12] Q. Tao, G.W. Wu and F.Y. Wang, Posterior probability support vector machines for unbalanced data, IEEE Transactions on Neural Networks **6** (2005), 1561–1573.

[13] E. Koyas, M. Sarac and A. Erdogan, Control of a BCI-based upper limb rehabilitation system utilizing posterior probabilities, 21st Signal Processing and Communications Applications Conference, 2013, 1–4.

[14] J. Wang, L. Gao and H.S. Zhang, Adaboost with SVM-Based classifier for the classification of brain motor imagery tasks, Universal Access in HCI, Part II, HCII **6766** (2011), 629–634.

[15] Y.L. Zhao, L.L. Wang and X.S. Zheng, Image texture classification based on LS-SVM and its use in digital watermarking, International Journal of Digital Content Technology and Its Applications **6** (2012), 513–519.

[16] Z.H. Gu, Z.L. Yu and Y.Q. Li, Sequentially updated least squares support vector machine with applications in online brain-computer interfaces, 6th IEEE Conference on Industrial Electronics and Applications, 2011, 459–462.

[17] H.T. Lin and C.J. Lin, A note on platt's probabilistic outputs for support vector machines, Machine Learning **68** (2007) 267–276.

[18] D.Q. Gao and J. Li, Kernel fisher discriminants and kernel nearest neighbor classifiers: A comparative study for large-scale learning problems, 2006 International Joint Conference on Neural Networks, 2006, 1333–1338.