34th Annual International Conference of the IEEE EMBS
San Diego, California USA, 28 August - 1 September, 2012

Recursive Channel Selection Techniques for Brain Computer Interfaces

Gareth Oliver[1], Peter Sunehag[1] and Tom Gedeon[1]

Abstract— **Automated channel selection allows the dimension of EEG data to be reduced without expert knowledge. We introduce Recursive Channel Insertion, an extension to Recursive Channel Elimination, which dramatically reduces calculation time with no loss of accuracy. Furthermore we propose Repeated Recursive Channel Insertion, which shows an improvement in accuracy over the previous methods when tested on a standard dataset.**

I. Introduction

Brain Computer Interfaces(BCIs) classify brain wave data so it can be used as a control mechanism. It has a wide variety of applications as a hands-free mode of control, notably for those suffering from amyotrophic lateral sclerosis[3].

Channel selection is an important part of preprocessing the brain-wave data from EEGs for BCI tasks, and can be crucial in reducing the noise and dimensionality of the data. Most commonly, expert knowledge is used to reduce the number of channels to those that are most likely to contain information that separates the classes. A variety of different automated feature selection techniques have also been applied to EEG data. These include Recursive Feature Elimination[1], which was later extended as Recursive Channel Elimination[6], zero-norm optimisation[1] and Fischer criterion[1]. Of these, RCE has been one of the most successful [6]. Methods such as Common Spatial Subspace Decomposition and Common Spatial Patterns also reduce the number of channels, however the additional noise in channels degrade their effectiveness[8], [7], [11]. Genetic algorithms have also been used successfully for channel selection[9], as well as more general feature selection[10]. However these were not looked into for this paper.

This paper proposes two extensions to RCE, Recursive Channel Insertion (RCI) and Repeated Recursive Channel Insertion (RRCI) which seek to improve the channels selected, as well as significantly decrease the selection time when selecting from a large set of channels. First the three different channel selections methods will be introduced. This will be followed by briefly giving the structure of the classifier that will be used for each channel selection method. Finally the experimental results of each channel selection method will be given and the implications discussed.

II. Channel Selection Methods

This section will introduce RCE, as well as RCI and RRCI.

[1]G. Oliver, P. Sunehag and T. Gedeon are with the Research School of Computer Science at the Australian National University. gareth.oliver at anu.edu.au

A. Recursive Channel Elimination

RFE was designed to eliminate features that had the minimum impact on the margin of a SVM. RCE extended this to removing channels by grouping the features that came from a single channel together to find the channel which had the minimum impact. It was also generalised for any classifier by using accuracy to determine the channel with the minimum impact[2]. It has been shown to be highly successful, and can be generalised to unknown subjects, although it loses some accuracy[6]. RCE seeks to find the group of channels that give the best performance for a particular subject. As testing all possible sub-groups would be too computationally expensive it uses a greedy algorithm to recursively remove the worst channel. To begin with all channels are in the group. The channels are then ranked on the accuracy of the group without that channel. The highest ranked channel is removed, and the process is repeated with the new sub-group. This continues until a minimum number of channels is reached [2]. The total number of times the accuracy is calculated can be given as

$$T = \frac{(n+1)n - (M+1)M}{2}$$

where M is the number of elements in the selected channels and n is the total number of channels. This becomes highly expensive when M is significantly smaller than n. One solution is to remove multiple channels in each step, however this can result in less accurate selection of channels.

B. Recursive Channel Insertion

We proprose a method, Recursive Channel Insertion(RCI) that works in the reverse way to RCE. Rather than removing channels until the best subgroup is found, it builds the group of channels up. The algorithm is given in algorithm 1. RCI attempts to add each channel to the group in turn, and then ranks the channel in terms of the accuracy of the group with that channel. The channel that gives the highest accuracy is then permanently added, and the process is repeated until the accuracy ceases improving, or a preset maximum size is reached.

The total time the accuracy is calculated can be given as

$$T = \frac{(n+1)n - (n-M+1)(n-M)}{2}$$

where M is the number of elements selected and n is the total number of channels. When compared to RCE this is less when $M < \frac{n}{2}$. This means that, in cases where a small number of the total channels are optimal, RCI will be significantly faster to perform than RCE.

Inputs
C - Set of Channels
X - Training Data
M - Maximum Channels Inserted

Algorithm
$acc \leftarrow 0$
$c \leftarrow \emptyset$
$S \leftarrow \emptyset$
repeat
  $S \leftarrow S \cup c$
  $C \leftarrow C \setminus c$
  $pacc \leftarrow acc$
  $acc, c \leftarrow \max\ accuracy(S \cup n, X) | n \in C$
until ($pacc \geq acc$ **or** $sizeof(S) > M$)
return S,C

Fig. 1. RCI algorithm

C. Repeated Recursive Channel Insertion

Initial experiments using RCI showed that, for individual subjects, multiple sub-groups of channels performed well. Therefore we developed Repeated Recursive Channel Insertion (RRCI) to exploit this. As shown in algorithm 2, RRCI makes use of the RCI channel selection algorithm to select a group of channels. It then repeats RCI to find other groups of channels until either the desired number of sub-groups are found or the best accuracy drops below a certain threshold. These multiple sub-groups can be used to train an array of classifiers, which can be combined by some form of combination classifier. In this case the weighted sum of the normalised output weights was used, although it is possible other more sophisticated methods could be more effective.

Inputs
C - Set of Channels
X - Training Data
m - Maximum Channels Inserted
n - Maximum Groups of Channels

Algorithm
$G \leftarrow \emptyset$
repeat
  $g, C \leftarrow$ RCI(C, X, m)
  G.append(g)
until ($sizeof(S) > m$)
return G

Fig. 2. RRCI algorithm

The total time is the same as for RCI with $M = \sum m_n$ where m_n is the size of the n^{th} subgroup. As before when $M < \frac{n}{2}$ RRCI will be faster than RCE. Due to the use of an array of classifiers, the classification process will also be slower than that of either RCI or RCE.

III. Structure of Classifier

The structure of a BCI classifier can be broken into three parts, preprocessing, feature extraction and classification. Each of the methods used for the channel selection experiment will be discussed briefly below.

A. Preprocessing

Bandpass frequency filtering was used in the preprocessing stage. Responses to motor imagery classification tasks are known to occur in the α (8-12hz), β (14-28hz) and θ(1-7hz) bands. As such each trial was based through an array of three bandpass butterworth filters in each of these frequency ranges. Additionally the data was normalised using statistical normalisation within each channel.

B. Feature Extraction

Common Spatial Subspace Decomposition (CSSD)[8], along with Common Spatial Patterns (CSP)[7] are among the most popular and widespread feature extraction techniques used in motor imagery BCI. CSSD seeks to reduce the dimensionality of the data by selecting the channels that maximise the variance between the classes. This is done through the simultaneous diagonalisation of the co-variance matrices, and can be found in [8].

CSSD generates a spatial filter matrix SF_y for each class y. To extract the feature vector the trial being classified is divided into three parts corresponding with the onset, middle and end of the response. The log of the variance of each is then selected as a feature. The calculation of the feature vector is given below for some input X and class y.

$$f(X, y) = (var(SF_y.X))$$
$$features = [f(X_{onset}), f(X_{middle}), f(X_{end})]$$

C. Classification

A Support Vector Machine(SVM) was used to perform the classification. SVMs are a popular classifier that has been used to great success in the classification of motor tasks[11]. SVM seek to maximise a decision boundary between the classes. For simplicity a linear kernel was used. For completeness the formalisation of the SVM used follow.The decision boundary problem can be formalised as finding a discriminant function F(x,y) so that the prediction $\hat{y}(x)$ satisfies

$$\hat{y}(x) = \arg\max_{y \in C}\ F(x, y)$$

where

$$F(x, y) = \langle \phi(x, y), w \rangle$$

ϕ is the feature map and C is the set of classes. Given a set of *n* training pairs (x^i, y^i) a SVM chooses a w so as to minimise

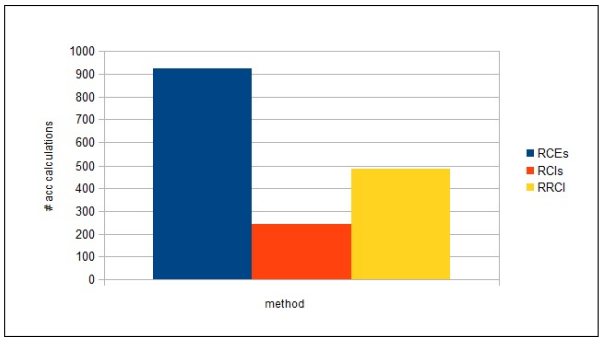

Fig. 3. Number of accuracy calculations performed

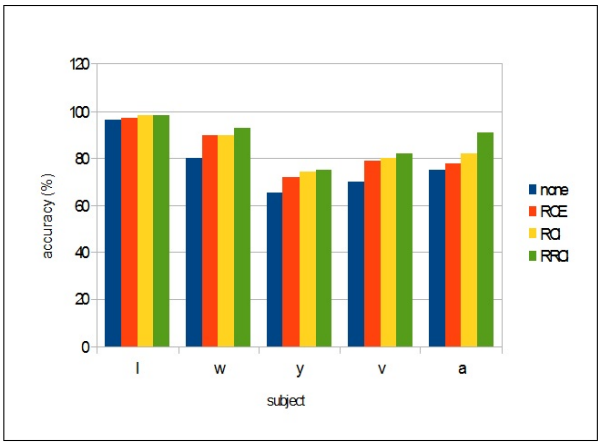

Fig. 4. Accuracy of channel selection methods

$$\lambda\frac{\|w\|^2}{2}+\frac{1}{n}\sum_{i=1}^{n}\max_{y'\in C}\left\langle\phi(x^i,y')-\phi(x^i,y^i),w\right\rangle+\delta_{y^i,y'} \quad (1)$$

Where λ is a regularisation constant and greater than 0. Stochastic gradient descent can be performed to find a solution to this convex optimisation problem. From equation 1 it follows that the weight update law is

$$w_{t+1}=w_t-\rho_t(\lambda w_t+\phi(x^i,y^*(x^i,y^i))-\phi(x^i,y^i))$$

where ρ is the learning rate. By using $\rho_t=\frac{\tau}{1+t}$ convergence can be gaurrenteed[13].

IV. Experiment

To compare the accuracy of the proposed channel selection methods they were each used on a standard dataset. This was the BCI competition III dataset IVa[12], a synchronous two class motor imagery classification problem.

A. Results

B. Dataset

Dataset IVa consists of 5 subjects, a, l, v, w and y. Each contains 280 trials, between testing and training data. They were recorded with an 118 channel EEG at 100hz and are each 3.5 seconds in length. Each trial was a motor imagery task moving either the right hand, or the left foot. Subjects a, v and y contain induced uncorrelated eye movements to increase noise in the data.

The testing and training datasets for each subject were combined and randomised, then split into two halves. The first half was used to perform the channel selection method, with the accuracy being calculated by 5-fold cross-validation. The selected channels were then used to calculate the overall accuracy of the method using 5-fold cross-validation, with the data used to perform the channel selection being added to the training data of each fold. This was then repeated using the second half of the data as the training for the channel selection methods. Each channel contained 350 entries. For CSSD the onset is defined as the first 50, and the end as the last 50, while the middle is the remaining 250. Finally the value used for m in CSSD was 3. To help the channel selection algorithms, only the channels over the motor cortex are included in the initial set of channels, giving 43 initial channels.

RCE and RCI were initially stopped when their respective accuracies decreased from one interation to the next. RRCI was run completely through, and the set of subgroups with the greatest accuracy was used. Additionally, RCE and RCI were also run completely through, to ensure that neither were stopped before reaching the maximum accuracy, and thus unfairly bias the RRCI algorithm.

The total number of channels returned by each channel selection method is given in Table: I. RCE_s was the number of channels returned when the stopping condition was satisfied while RCE_b was the best when run completely through. The same was true for the two RCI versions. As the b and s versions were always the same, it can be concluded that the stopping condition was sufficient in these cases. For RRCI the total number of channels in all subgroups is given. The final column gives the total number of times the accuracy is calculated. By this measure RCI performs less than a third as many as RCE, half as many as RRCI. Naturally without a stopping condition both RCE and RCI perform the maximum number of accuracy calculations (946).

Groups	l	w	y	v	a	ave acc calc
RCE_s	6	6	6	6	6	925
RCE_b	6	6	6	6	6	946
RCI_s	6	6	6	6	6	243
RCI_b	6	6	6	6	6	946
RRCI	6	14	12	21	15	487

TABLE I
NUMBER OF CHANNELS SELECTED

	l	w	y	v	a
none	96	80	65	70	75
RCE	97	90	72	79	78
RCI	98	90	74	80	82
RRCI	98	93	75	82	91

TABLE II
ACCURACY OF CHANNEL SELECTION METHODS

Table:II compares the accuracies of each channel selection method on each subject. From this it can be seen that RCE is marginally worse than RCI, while RRCI is better or equal to each individual method.

V. Conclusion

This paper introduced two extension to the RCE channel selection method for motor imagery brain computer interfaces, RCI and RRCI. It found that RCI was significantly faster than RCE when a small group of channels are desirable, as was the case on the datasets tested. Additionally it found that RCI slightly better than RCE, and that RRCI was able to improve the overall accuracy of the classification.

References

[1] T N. Lal, M. Schroder, T. Hinterberger, N. Birbaumer, and B. Scholkopf, Support Vector Channel Selection in BCI, Technical Report, Max-Planck Institute for Biological Cybernetics, Tubingen, Germany, December 2003.

[2] A. Rakotomamonjy and V. Guigue, BCI Competition III: Dataset II - Ensemble of SVMs for BCI P300 Speller, *IEEE Transactions on Biomedical Engineering* 3:1147-1154, 2008.

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

[4] G. Pfurtscheller, and F. H. L. Da Silva, U Event-related eeg/meg synchronization and desynchronization: basic principles, *Clinical Neuroscience* , 110:1842-57, 1999

[5] M. Grosse-Wentrup, M. Gramann, and M. Buss, Adaptive spatial filters with predefined region of interest for EEG based brain computer interfaces, *Advances in Neural Information Processing Systems* ,19:537-44, 2007

[6] M. Schroder, T. Lal, N. Hintenberger, M. Bogdan, J. Hill, N. Birbaumer, W. Rosenstiel and B. Scholkopf, Robust eeg channel selection across subjects for brain-computer interfaces, *EURASIP Journal on Applied Signal Processing*, 19:3103-12, 2005

[7] J. Muller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, Designing optimal spatial filters for single-trial eeg classification in movement tasks. *Clinical Neurophysiology*, 101:787-798, 1998.

[8] Y. Wang, P. Berg, and M. Scherg, Common spatial subspace decomposition applied to analysis of brain responses under multiple task conditions: a simulated study. *Clinical Neurophysiology*, 110:604-614, 1999

[9] M. Schroder, M. Bogdan, T. Hinterberger, and N. Birbaumer, Automated EEG feature selection for brain computer interfaces, *Neural Engineering*, 2003. Conference Proceedings. First International IEEE EMBS Conference on , vol., no., pp. 626- 629, 2003

[10] R. Corralejo, R. Hornero and D. Alvarez, Feature selection using a genetic algorithm in a motor imagery-based Brain Computer Interface, *Engineering in Medicine and Biology Society*,EMBC, 2011 Annual International Conference of the IEEE , vol., no., pp.7703-7706, 2011

[11] F. Lotte, M. Congedo, A. Lecuyer, F. Lamarche, and B. Arnaldi, A review of classification algorithms for EEG-based brain-computer interfaces, *J. Neural Eng.*, 4:R1-R13, 2007.

[12] G. Dornhege, B. Blankertz, G. Curio, and K-R. Muller, Boosting bit rates in non-invasive EEG single-trial classifications by feature combination and multi-class paradigms. *Trans. Biomed. Eng.,* , 51:993-1002, 2004.

[13] H.E. Robbins, and S. Monro, A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400-407, 1951.