Classifying Mental Tasks Using a Brain-Computer Interface

Charlie Nash S1008057

1 Introduction

Electroencephalograms (EEGs) can record electrical activity in the brain. In conjunction with a brain-computer interface (BCI) they can be used to augment human sensory functions or control robotic devices. For example a user might control whether a wheelchair moves forward, left or right [1, 2]. In order to perform these functions the BCI must be able to classify EEG patterns as corresponding to a certain task and relay that information to control the device of interest. Typically the process involves feature processing and classification using machine learning methods [3].

This work follows the specification of the BCI competition III Dataset V [4] in which the goal is to classify three mental tasks online. Previous approaches have been to use a mixture of Gaussian classifier [5], transition detection techniques [6] and alternative feature extraction methods [7]. We use power spectral density (PSD) for feature processing, random forests and PCA for dimensionality reduction, and compare random forest and hidden Markov model classifiers.

2 Data description

Annotated EEG data was obtained from the BCI Competition III website. Dataset V was provided by the IDIAP Research Institute [5]. The data contains EEG signals for 3 subjects across 4 sessions each, with sessions lasting 4 minutes. During a session subjects performed a task for 15-18 seconds before switching randomly at the operator's request. In the competition specification the first three sessions were to be used for training and the final session for testing. However during this project the class labels for the fourth session were not available so the first two sessions were used for training and the third for testing as an alternative. In practice the two training sessions were joined to create one training set. The three mental tasks involved repetitive left/right hand imagery and thinking of words beginning with a certain random letter. The EEG signals consist of 32 EEG potentials at each time point with a sampling rate of 512 Hz.

In addition to the raw EEG data, the competition organisers provided processed features which were computed by spatially filtering EEG potentials using a surface Laplacian before estimating the PSD in the band 8-30 Hz over the last second of data with a frequency resolution of 2 Hz for the 8 centro-parietal channels. Each channel has 12 frequency components so that the total number of processed features is $8 \times 12 = 96$. Figure 1 shows the values of the processed features for subject 1 at the first time point.

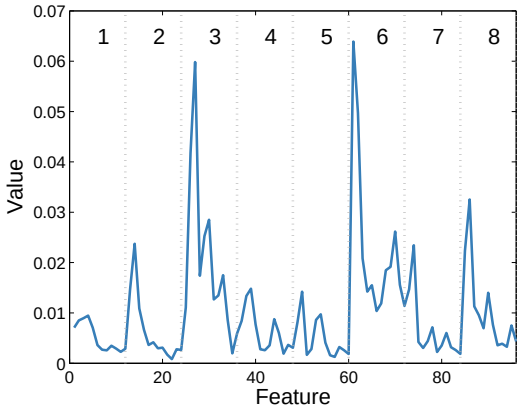

Figure 1: Input features for subject 1 at the first time point. The numbers 1 - 8 correspond to the 8 centro-parietal bands.

In this project we use the processed PSD features, although it would be of interest in an extended project to test alternative feature processing methods. To summarise, the task is a 3-class classification problem with 96 features.

2.1 Performance Evaluation

Models are trained and evaluated separately for each subject. This is reasonable as in practice the user of the BCI would first train the system on their own EEG signals before use. Performance of the classifier is evaluated by finding the accuracy of the classifier on the test set for each subject. The mean subject accuracy is taken as the overall performance measure.

Under the competition specification class labels must be estimated subject to some restrictions: A response time of 0.5s is desirable, so the classifier must

return one label for each 0.5s block. This corresponds to 8 time points. In order to guarantee a fast response time larger blocks must not be used for smoothing or as part of a HMM classification.

These specifications are followed in this work although we also consider longer blocks and investigate how this affects classifier importance.

3 Methods

Two classifiers were considered in this project: random forests [8] and hidden Markov models (HMMs) [9]. This section contains details of parameter selection for these methods and validation techniques. For the random forests we considered reducing the dimensionality of the features as well as how the classifications for each time point could be filtered.

3.1 Random Forests

Random forests can be used for classification and regression. Classification is performed by constructing a number of decision trees and outputting the most frequent classification. We used random forests with 100 trees and the number of variables to select at random for each decision split was set to the square root of the number of features.

3.2 Dimensionality Reduction

Reducing the dimensionality of the input features by feature selection or extraction can improve classification accuracy by removing irrelevant features or constructing variables which separate the classes well. Validation sets for each subject were constructed in order to assess the performance of the dimensionality reduction techniques. The validation sets were constructed by taking the last quarter of the training set of each subject. As the labels are evenly distributed over the training data, the validation set will contain a good representation of each class.

Principal component analysis (PCA) is a common choice for feature extraction. PCA works by linearly transforming features to an alternative space in which the variability is concentrated in the first few dimensions. Dimensionality can be reduced by selecting a number of projected features dependent on how much of the overall variance is desired. PCA was applied to the normalised features, and transformed features were kept such that certain levels of the variability was captured. Classification was then performed using random forests and the transformed features. Table 1 shows the classification accuracy on the validation set and figure 2 shows the distribution of the data on the first two principal components.

	Subject 1		Subject 2		Subject 3	
Var	Acc	n	Acc	n	Acc	n
70	0.386	33	0.317	34	**0.278**	35
80	0.386	44	0.326	45	0.273	45
90	0.386	59	0.326	60	0.273	61
100	0.386	96	**0.328**	96	0.273	96

Table 1: Validation accuracy using random forests for principal component features. Var refers to the variance explained by the features.

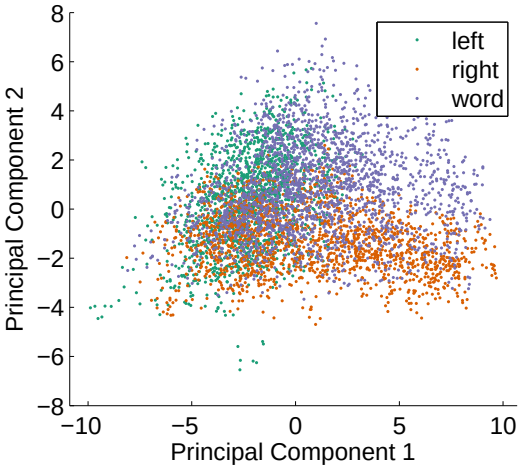

Figure 2: Subject 1 data plotted on the space of the first two principal components.

It is notable that the classification accuracy is very low for the transformed features at all levels of explained variance. Figure 2 shows that the classes are not well separated on the first two principal components. As such PCA was rejected as a feature extraction method.

As an alternative to PCA, feature selection was performed using random forests, where variable importance can be used to rank the variables by permuting variable values and and recording the out-of-bag error [8]. Figure 3 shows classification accuracy on the validation set against the number of features retained. It is notable that the number of features makes very little difference to classification accuracy except for a small increase for subject 3, and a small decrease for subject 2. A heuristic measure of choosing the number of features to retain is to select the minimum number such that the accuracy is within one standard deviation of the maximum accuracy. 3 features were kept for subject 1, 4 features

were kept for subject 2 and 35 features were kept for subject 3.

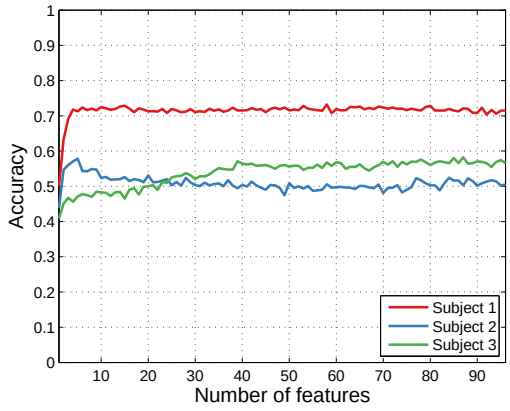

Figure 3: Classification accuracy vs number of features for each subject using the random forest classifier.

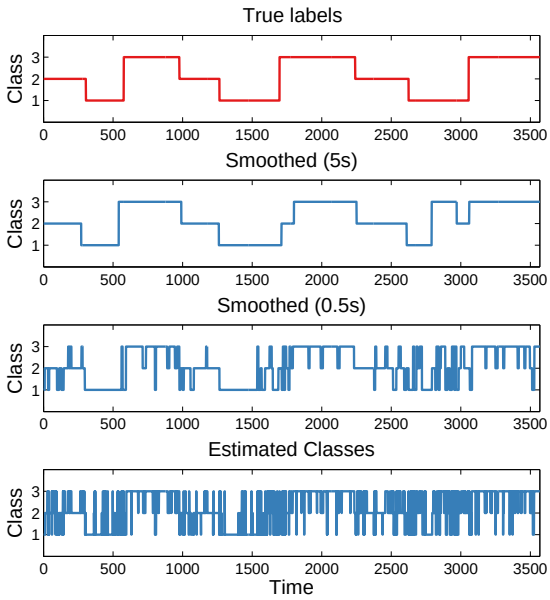

Figure 4: Classification using random forests with varying degrees of filtering. Plots produced using subject 1 and the full feature set as input.

3.3 Filtering

Random forests were trained for each subject using both the full feature set and the reduced features as selected by random forest variable importance and the heuristic cut-off. Each point in the test set was assigned a class using the classifier, however we can take advantage of the fact that subjects will remain performing one task for a sizeable time before transitioning to another. Predictions can be filtered by majority voting: classifying an entire time segment as the most commonly occurring classification in that segment. Under the BCI competition specification time windows for classifications must be 0.5 seconds long - 8 data points, otherwise the response time of the system would be too slow. In addition, previous data points could not be used. Figure 4 shows the results of classifications for subject 1 using time windows that fit the competition rules, as well as longer windows.

Classifications were made using raw classifications and filtering over 0.5 second and 5 second windows. In an extended project it would be of interest to investigate the suitability of different filtering methods such as growing windows as in [6].

3.4 Hidden Markov Models

HMMs with two hidden states were used as an alternative classifier. The joint distibution of a HMM is

$$P(s_{1:T}, \mathbf{x}_{1:T}) = P(s_1)P(\mathbf{x}_1|s_1)\prod_{t=2}^{T} P(s_t|s_{t-1})P(\mathbf{x}_t|s_t)$$

where s_t and $\mathbf{x}_t$ are the hidden state and the visible variables at time t. In this case we used a multivariate Gaussian as the emission distribution so that the conditional probability of the visible variables given the hidden state is

$$p(\mathbf{x}_t|s_t) = \mathcal{N}(\boldsymbol{x}_t|\boldsymbol{\mu}_{s_t}, \Sigma_{s_t}).$$

Separate models were trained for each subject and for each class c using Kevin Murphy's PMTK Matlab toolkit [10]. The implementation uses expectation maximisation to find MAP estimates of the model parameters. The prior on the emission distributions is Gaussian-inverse-Wishart with parameters $\boldsymbol{\mu}_0 = \mathbf{0}$, $\lambda = 0.01$, $\boldsymbol{\Psi} = 0.1\mathbb{I}_{96}$ and $\nu = 97$. The prior on the starting distribution $P(s_1)$ is Dirichlet with parameters $\boldsymbol{\alpha} = (1, 1)$ and the prior over transitions is the matrix of pseudo-counts T where $T_{ij} = 1$. A sequence of testing input $(\mathbf{x}_t, .., \mathbf{x}_{t+k})$ over time window w can be classified as $\hat{c}_w = \arg\max_c \; p(\mathbf{x}_t, .., \mathbf{x}_{t+k} \mid c)$ where the probability $p(\mathbf{x}_t, .., \mathbf{x}_{t+k} \mid c)$ can be computed using the forward algorithm. The assumption here is that the probabilities of belonging to each class are equal; this is valid as the subjects perform tasks in a random sequence in each session. Time windows of 0.5 seconds and 5 seconds were chosen in order to compare the HMMs to filtered random forest classifiers.

The choice of two hidden states was made for the sake of simplicity, however it may be the case

that different numbers of hidden states would be appropriate for different subjects, or even different classes within a single subject. In a longer project it would be sensible to use a validation process to assess the suitability of alternative numbers of states.

An alternative model was considered in which the hidden states of the HMM correspond exactly to the tasks. A potential issue with this approach is that once trained, the hidden states would transition to the same state with high probability. Therefore when classifying a time segment $(\mathbf{x}_t, .., \mathbf{x}_{t+k})$, the model would choose an initial hidden state, and then stay in that state for the duration of the segment. As such the segment is likely to classify the entire segment as the initial hidden state.

4 Results and Discussion

Table 2 shows the accuracy of the classifiers on the test set for each subject. Overall accuracy is the mean accuracy across the subjects.

		Subject			
	Classifier	1	2	3	Overall
	Baseline	0.33	0.33	0.33	0.33
	Galan* [4]	0.80	0.70	0.56	0.69
F	RF	0.75	0.61	0.40	0.59
	RF 0.5s	**0.76**	**0.65**	**0.42**	**0.61**
	RF 5s	**0.86**	0.68	0.50	0.68
	HMM 0.5s	0.50	0.39	0.37	0.42
	HMM 5s	0.52	0.37	0.34	0.41
R	RF	0.72	0.60	0.42	0.58
	RF 0.5s	**0.76**	**0.65**	0.41	**0.61**
	RF 5s	**0.86**	**0.79**	**0.53**	**0.73**

Table 2: Classification accuracy on the test sets for each classifier. Classifiers are divided into those using all the features (**F**) and reduced feature sets (**R**). Overall accuracy is the mean subject accuracy for a classifier. Top performing classifier in **red**, top performing classifier within competition specification in **blue**. Baseline and BCI competition winners included for comparison. *Note that competition entrants had used the fourth session for testing rather than the third.

Classification accuracy is variable across the subjects. Subject 1 produced good results for most of the classifiers whereas subject 3's activities were more difficult to classify. This may be due to inherent variability in electrical activity in the brain across subjects, or that some subjects better understood the directions in the experiment.

The highest overall accuracy was achieved by the random forest classifier with reduced input features and with filtering on 5 second blocks. It is of note that the RF classifier with full input features had very similar performance on subjects 1 and 3, but the accuracy for subject 2 was 11% higher for the reduced feature set.

For classifiers within the BCI competition specification, random forests filtered over 0.5 seconds with both the full and reduced feature set had the best performance. On average this classification accuracy is 8% lower than the best work from the competition and would place the author 11th in the competition of 20 legitimate entrants. The winning entry incorporated change-point detection in their classifier [4] and it is likely that this would similarly improve the accuracy of the RF classifier. An alternative approach is a conditional random field, with which neighbouring samples can be taken into account when making a classification [11].

The performance of the HMMs using both 0.5s and 5s blocks was roughly similar, and similarly poor, beating the baseline by just 8-9%. Closer examination of the fitted HMMs reveals some issues that might explain the low classification accuracy. The following parameters were estimated for a HMM trained on the left hand movement task for subject 1. The estimated initial state probabilities are $\hat{\boldsymbol{\pi}} = (0.57, 0.43)$, and the transition matrix is

$$\hat{A} = \begin{pmatrix} 0.978 & 0.0282 \\ 0.0290 & 0.9710 \end{pmatrix}.$$

The parameters show that once the initial hidden state has been set, there is very high probability of remaining in that state. Figure 5 shows the mean of the emission Gaussians for each hidden state.

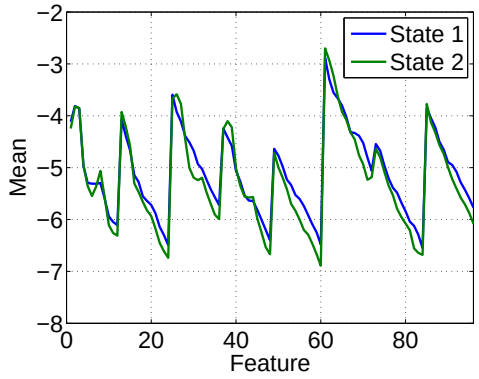

Figure 5: Classification using random forests with varying degrees of filtering. Plots produced using subject 1 and the full feature set as input.

An issue is immediately apparent: the Gaussians are almost identical. Therefore no matter what the hidden state is the probability of the observed values will be very close. As such the HMM is close to a Gaussian model, and the classifier will simply choose the most probable of three Gaussians. A next step would be to use alternative priors to encourage the transition matrix to be less symmetric, and the emission distributions to differ from one another.

5 Conclusion

This work presents and compares methods for the classification of mental tasks using EEG data. Power spectral density was used to process the raw features. PCA and random forest variable importance were used for feature selection. Random forests and hidden Markov models were used for classification.

The task was guided by The BCI competition III Dataset V [4] and evaluation of the methods was made in accordance with the competition specification. The competition required that classifications be returned in 0.5 second blocks, and that no other samples outside of that block were used in the classification. Majority vote filtering was used to provide the required output for the RF classifier and the probability of sequences lasting 0.5 seconds was evaluated in order to meet the specification for the HMMs.

Applying PCA to reduce dimensionality before classifying with random forests reduced classification accuracy. Random forest variable importance was used as an alternative method.

The results of the competition specification classifiers are reasonable among the other entrants although not at the top of the table. Accuracy can be improved by using a larger 5 second window for filtering although this is not within the rules of the competition. The highest overall accuracy was achieved by the random forest classifier with reduced input features and with filtering on 5 second blocks.

Future work could investigate the suitability of change-point detection as used by Galan [4], moving windows as in [6] or alternative models such as a conditional random field. It would also be of interest to investigate the scientific explanation for which variables were considered most important by the random forest. Perhaps certain frequencies, or certain locations on the scalp have the greatest ability to separate the mental tasks.

References

[1] J. R. Millan, F. Renkens, J. Mouriño, and W. Gerstner, "Noninvasive brain-actuated control of a mobile robot by human EEG," *Biomedical Engineering, IEEE Transactions on*, vol. 51, no. 6, pp. 1026–1033, 2004.

[2] Ingenaire, "EEG-controlled wheelchair - McMaster University," 2012. [Video file] Retrieved from `https://www.youtube.com/watch?v=UUcubnQML9s`.

[3] G. Dornhege, M. Krauledat, K.-R. Muller, and B. Blankertz, "13 general signal processing and machine learning tools for bci analysis," *Toward brain-computer interfacing*, p. 207, 2007.

[4] "BCI Competition III Final Results." Available `http://www.bbci.de/competition/iii/results/index.html`, 2005.

[5] J. del Millán, "On the need for on-line learning in brain-computer interfaces," in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, vol. 4, pp. 2877–2882, IEEE, 2004.

[6] R. Aler, I. M. Galván, and J. M. Valls, "Transition detection for brain computer interface classification," in *Biomedical Engineering Systems and Technologies*, pp. 200–210, Springer, 2010.

[7] A. B. Benevides, T. F. Bastos Filho, and M. Sarcinelli Filho, "Pseudo-online classification of mental tasks using kullback-leibler symmetric divergence," *Journal of Medical and Biological Engineering*, vol. 32, no. 6, pp. 411–416, 2012.

[8] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in *PROCEEDINGS OF THE IEEE*, pp. 257–286, 1989.

[10] K. Murphy and M. Dunham, "PMTK3." `https://github.com/probml/pmtk3`, 2014.

[11] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[12] B. Blankertz, K. Muller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlogl, G. Pfurtscheller, J. R. Millan, M. Schroder, and N. Birbaumer, "The BCI competition III: Validating alternative approaches to actual BCI problems," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 14, no. 2, pp. 153–159, 2006.