Using the BCI Competition Dataset for Motor Classification

Jonathan Nguyen, Richard Gross, Sam Schwartz, Alex Nguyen, William Li, Audrey de Leon

Introduction & Motivation

Brain-computer interfaces (BCI's) research is starting to become more and more popular in recent years. The aim of these systems is to try to establish a communication between the brain and an output device like a computer. The one problem is, how is the computer able to extract features from the signals given like EEG or MEG data? Ultimately, this problem boils down to a classification issue, which is used as a model to predict certain types of inputted data. To replicate this problem, we decided to use the BCI competition IV dataset III, which consists of MEG signals recorded during right hand movements in four directions. Our proposed solution to this problem is to use Principal Component Analysis (PCA) to reduce the dimensions of the data, use the projected data in Independent Component Analysis (ICA) to remove artifacts, and subsequently use the output data in a Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), and K-Nearest Neighbor (KNN) classifier respectively.

Related Work

To prepare for our project, we found several different papers relating to the BCI competition IV dataset III. The first was the BCI competition winners for the third dataset which conducted data preprocessing, feature extraction, feature selection, and classification. For feature extraction, Sardouie & Shamsollahi created various time, frequency, and time-frequency domain features. These features were selected using two different methods, scattering matrices measure and genetic algorithm. Following feature selection, Sardouie & Shamsollahi used different classifiers, but ended up deciding on the linear SVM and LDA classifiers as they had better accuracy than others. The resulting classification accuracy was 59.5% and 34.3% accurate for subject 1 and 2 respectively, which achieved the best result in the BCI competition for this particular dataset.

Feature set	SVM cross-validation	LDA cross-validation	SVM test	LDA test
Set 1: time mean	32.55 ± 5.99	30.72 ± 7.12	36.98	28.76
Set 2: variance	22.85 ± 5.94	23.25 ± 6.55	35.61	30.13
Set 3: AR coefficients	20.75 ± 5.10	23.97 ± 6.22	21.91	17.80
Set 4: form factor	25.72 ± 6.59	25.70 ± 6.29	38.35	28.76
Set 5: median frequency	27.10 ± 6.50	26.30 ± 6.04	26.02	23.38
Set 6: mean frequency	28.67 ± 6.79	27.50 ± 7.20	38.35	36.79
Set 7: mode frequency	25.90 ± 5.57	26.27 ± 5.95	30.13	27.39
Set 8: power spectral ratio	22.62 ± 6.28	23.55 ± 6.23	24.65	28.76
Set 9: Haar coefficients	33.90 ± 6.08	28.17 ± 6.92	35.61	27.39
Set 10: db2 coefficients	32.62 ± 6.80	26.32 ± 6.50	30.13	26.02
Set 11: db4 coefficients	34.07 ± 6.90	28.17 ± 6.68	31.50	24.65

Table 1. Sardouie & Shamsollahi Classification accuracy %

Another paper we reviewed, published in EECOS 2016, utilized a different set of feature extraction techniques and classification algorithms. In the second paper, Shahrukh, Usmani, and Rafiuddin calculated the cross-correlation between pre and post movement-onset data for each class to extract key features and computed principal component analysis on the mean and median of results from cross-correlation to select representative features for each class. In terms of the classification algorithm, the authors used K-Nearest Neighbors and arrived at 25.7% and 31.5% accuracy for subject 1 and 2, respectively (average accuracy at 29%), which is also relatively high among participant teams from the BCI competition.

Methods

Data Preparation and Cleaning

To prepare our data for use, we loaded each MATLAB file and extracted both the train and test sets. After doing so, each training set had each of its MEG channel sensor columns extracted, then put into a list for further processing. The data are grouped by MEG channel with 160 total trial observations (40 per target) and 400 samples per observation. This processing has also been done with the test set, which comprised 74 trial data from subject 1 and 73 trial data from subject 2.

Each trial contains 10 channels. In order to clean the data, we looped through each trial and grouped up each channel. With each channel in its own group, we will be able to perform dimension reduction techniques to reduce the 400 samples for each trial per channel efficiently.

PCA

After finishing data preparation, the first step we took was to standardize the data to unit variance by finding the mean and standard deviation of each parameter, calculating the deviation of each datapoint to the mean and dividing by the standard deviation. This is done by using sklearn's StandardScaler function. After standardizing the data, we were able to reduce the dimensions of the multichannel MEG data down to 65 by running PCA on each channel. Since each channel of MEG data has different variability, we set up our PCA to seek the proper number of components that can guarantee a baseline variance level of 95%. The result shows that in each channel, around 95% of the variance in the original data could be captured if we use the first largest 65 components. This embedded data is then appended to an array to concatenate it into the final dataset for the classifiers. The labels were then added to its own array. The same procedure of feature extraction was also done to the test set.

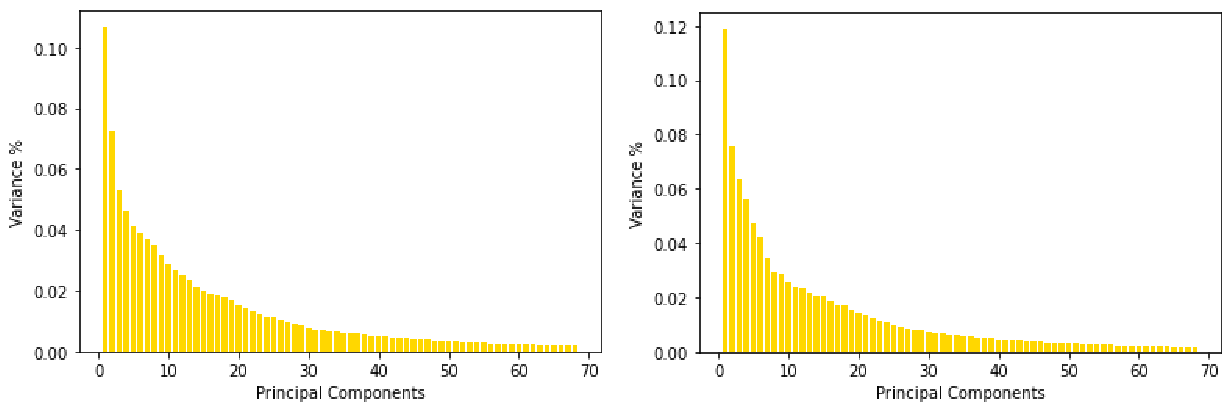

Figure 1. Percentage of variance explained by each selected components on two channels of MEG data

ICA

In addition to the first round feature extractions using PCA, which aimed primarily to reduce the dimensionality of our MEG dataset, we attempted to separate artifacts from our data through ICA. As the PCA reduced the number of dimensions and accordingly adjusted the variance explained by different dimensions of the original MEG data, the application of ICA here became more suitable as it will less than likely generate overfitting results.

Classifiers

Following the PCA and ICA data preprocessing, we used our training set to train three different classifiers, Support Vector Machines, Linear Discriminant Analysis based classifier, and K-Nearest Neighbor, as each algorithm is used often for MEG/EEG classification in research. In terms of the algorithms we selected, SVM and KNN are two supervised learning algorithms that are suitable for the class labeling task here, and the LDA classifier we used is a Bayes-based classifier. Data with features constructed from PCA was input into each classifier. For certain classifiers (e.g. KNN), we also tested out a number of parameter settings in fine tuning with the hope that level of performance can be improved. At the end, accuracy was compared to evaluate the performance of each classifier.

Results and Discussion

After training, we ran our trained classifiers on our test set. We chose to evaluate based upon the accuracy from two subjects, S1 and S2. From the classifiers trained with PCA pre-processed data, we obtained an average overall (for both subjects 1 and 2), an accuracy of 25.17% from SVM, 29.93% from LDA, and 29.25% from KNN with 5 neighbors (we obtained the n_neighbors value from hyperparameter tuning and 5 neighbors generated the best accuracy performance). Subsequently, we attempted to re-train the classifiers by feeding the data processed by both PCA and ICA sequentially. As a result we were able to further improve the accuracy from the KNN classifier. (Table 2.) Overall, the LDA classifier and KNN classifier had leading performances in predicting labels in the test set. The accuracy could be improved to 32.65% in KNN if we add ICA as part of data-preprocessing. Compared to the other competition participants, we achieved a 4.83% (PCA) and a 7.55% (PCA and ICA) greater accuracy on average than the second place entry of the competition.

	Only PCA accuracy	PCA and ICA accuracy
SVM	25.17%	22.45%
LDA	29.93%	25.85%
KNN	29.25%	32.65%

Table 2. Result comparison with and without ICA on each classifier algorithm.

Limitations and Future Work

Currently our project was limited to testing with a single dataset provided by the BCI competition which only included data from two participants. Using a dataset with a larger number of participants from a more diverse background would have been ideal to improve the generalizability of our results. In terms of our algorithm, a potential limitation to it is that we used PCA to reduce the dimensions of our data, yet the reduction hurt our algorithm instead of improved its accuracy. At first, we tried to run PCA to try to capture at least 80% of the variance, but that substantially decreased the classifier algorithm's performance. We decided, in turn, to capture 95% of the variance, as this increased the classifier's performance. This shows that our dimensionality reduction algorithm may have not performed as well as we wanted it to. One possible reason for this is because there are different PCA iterations happening per channel due to their varying behavior, so each channel has its distinct variance coverage when we manually set up the number of components.

For future projects, we could test more classifier algorithms on the data such as a convolutional neural network (CNN) and Weighted Robust Distance (WeiRD) algorithm. The fine tuning of feature extraction methods and classifiers can be achieved by applying more thorough hyperparameter tunings through GridSearchCV. Further improvements could also be done by testing the strength of the classifier models using a K-fold cross validation test.

References:

Hajipour Sardouie S and Shamsollahi MB (2012) Selection of efficient features for discrimination of hand movements from MEG using a BCI competition IV data set. *Front. Neurosci.* **6**:42. doi: 10.3389/fnins.2012.00042

Shahrukh, M., A.A. Usmani, and N. Rafiuddin. "Decoding Wrist Movement Directions Using Directionally Modulated MEG Activity." 3rd International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences (EEECOS 2016) n. pag. Web.