Deep Learning Applications on EEG Signal Classification

Haoran Niu [1] **Christopher Perng** [2] **Jiahan Zhang** [1] **Akhil Surapaneni** [3]

Abstract

This project is designed to explore the utilization of machine learning algorithms for analyzing datasets of Electroencephalography (EEG) based Brain-Computer Interface (BCI). Steps to complete the task includes processing the EEG signal and building binary classification models. After the raw database is cleaned and analyzed, we will conduct PCA (principal components analysis) (2), and create, train and evaluate our own models. We found a lot of success with linear and non-linear models without needing to use a deep learning model for this task.

1. Introduction

Brain-computer interfaces (BCI) are systems that can decode a user's brain state and perform an action. They have shown promise for patients with neurological disorders to control prosthetic and assistive devices (7). While brain-computer interfaces based on invasively implanted recording arrays offer the highest decoding performance, even non-invasive methods of recording cortical activity have been used to build BCI's that can help patients (13). Electroencephalography (EEG) is a powerful tool for non-invasively monitoring brain activity, offering insights into neurological function, cognitive processes, and clinical diagnostics. In particular, sensorimotor oscillations hold promise to control devices for movement assistance and promote neuroplasticity for recovery. This project delves into the classification of EEG signals of sensorimotor activity with machine learning. We aim to combine data collection, data preprocessing, and advanced machine learning models to reveal the intricate patterns embedded within the EEG data. Accurately classifying EEG patterns corresponding to a state of motor cortex activation from a state of resting could enable a starting and stopping control signal for prosthetic devices. The purpose of this work is to compare linear and non-linear methods to classify motor imagery EEG signals and consider the benefit of using deep learning approaches.

[1]Electrical and Computer Engineering, University of Texas at Austin [2]Operations Research and Industrial Engineering, University of Texas at Austin [3]Biomedical Engineering, University of Texas at Austin. Correspondence to: Anonymous Author <ece>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

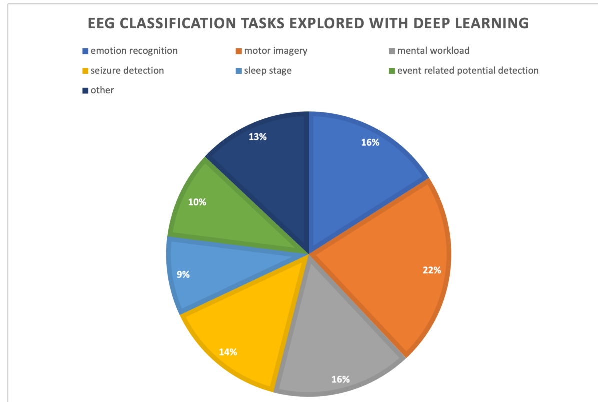

Figure 1. EEG classification tasks

.

2. Related Work

EEG analysis is important to neuroscience. Therefore, there are an increasing number of deep learning techniques developed for EEG signal classification (5). Based on the survey (5) in year 2019, the EEG classification tasks which have alread been explored with deep learning mainly fell into six groups. We used a pie chart Figure 1 to display the survey results. Our project is in the motor imagery category.

EEG signals are highly correlated with temporal information, and that's should be why time-dependent neural networks are popular in EEG classification tasks such as (6) and (14).

Since EEG signals can be converted to spectrograms, convolutional neural networks (CNN) is widely used for developing EEG classifiers (5). Paper (9) and paper (12) are two examples for CNN classifiers for EEG.

According to paper (5), classification models with CNN, recurrent neural networks (RNN) and deep belief networks (DBN) usually outperformed than others. However, in our project, we showed that it is still possible to create high-performance models without using neural networks of these above three.

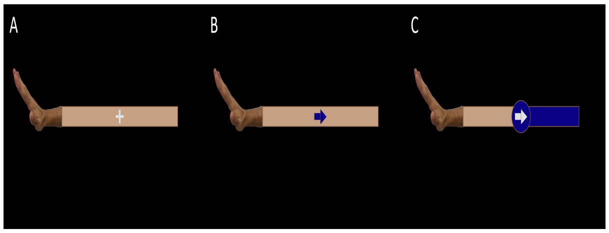

Figure 2. Rest Trial Task. A) Fixation Cross B) Cue Presentation C) Task Completion

3. Data Collection and Processing

EEG was recorded in 16 subjects performing motor imagery and resting state tasks. Each subject performed 3-5 runs with 10 trials of each class: motor imagery and resting state. For the motor imagery task, subjects were instructed to imagine the kinesthetic motion of extending their dominant hand in one continuous motion. For rest, subjects were only instructed to look at the computer screen. The order of resting state and motor imagery trials within each run was randomized. Each trial began with a fixation cross followed by both a visual and auditory cue for the corresponding class. Then, a progress bar filled the screen for five seconds, during which the subjects were asked to execute the corresponding task. Once the progress bar filled the screen, a cue indicating the completion of the trial was displayed (Figure 2

The subject's EEG signals were recorded using a 64-channel cap in the standard 10-20 configuration at a sampling rate of 64Hz, with ground electrode at the Oz position and the reference electrode at the CPz position. The last three seconds were used to avoid the presence of visual-evoked potentials due to cue presentation. In each trial, 33 overlapping epochs of one second in length were extracted. EEG was pre-processed by spatial filtering using an approximation of the surface Laplacian by the finite-difference method, creating a reference-free estimate of the current source density at each channel location (10). Following existing approaches, power spectral density bands of 2Hz between 4 and 40Hz were extracted for each epoch using Welch's method with a window of 0.5 seconds and a 50% overlap, creating a feature space of *63* channels by *14* PSD bands for each trial in each run (4). This resulted in a feature space of 882 indices and 13,680 samples per class (Figure 3).

4. Methodology and Experimental Setup

We trained and evaluated several linear and non-linear models – namely MLP, Logistic Regression, Support Vector Machine, Random Forest classifier and also employed the Stacking Ensemble method. These models all used 5-fold cross validation for performing grid search on hyperparameters. We used an 80/20 train/test split and performed cross validation with the training set. We also trained a LSTM model, but the temporal model did not provide good results. The bad performance may be caused by the short duration of the EEG signals that are collected.

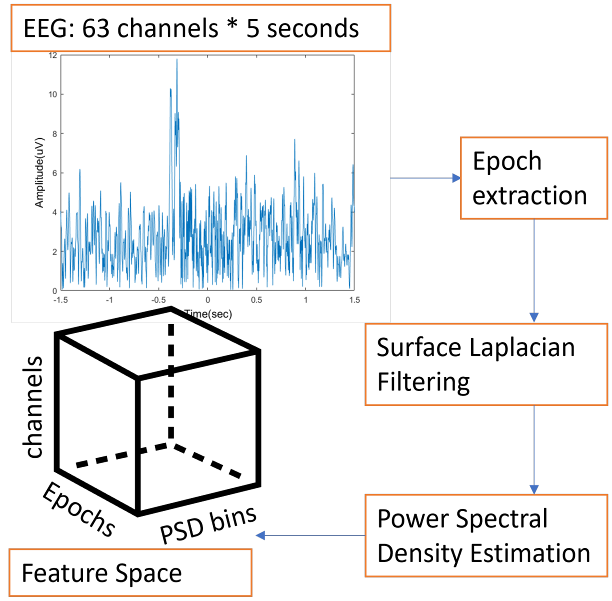

Figure 3. EEG Pre-Processing Pipeline to Obtain Feature Space

# of components	200	300	400	500	600
var explained	0.92	0.95	0.96	0.98	0.99

Table 1. number of components vs. cumulative variance explained.

4.1. Principal Component Analysis (PCA)

Given the large dimension of the input features, we used PCA to analyze and identify the most meaningful features (8). PCA transformation reduces the dimension of the dataset and lowers the chances of model overfitting (1). Based on our experiment results, it does help improve model performance. The scree plot is shown as Figure 4.

The plot above tells us that PCA transformation with number of components greater than 200 is more valuable. For instance, the cumulative variance explained by 300 components is around 0.95 (shown in Table 1).

Next, we need to test if PCA transformation could improve the classification performance. To achieve this goal, we test a Multilayer Perceptron model (MLP) with the original dataset. Next, we construct different reduced datasets with number of dimensions displayed above (Table 1). We use the reduced datasets to train new MLP models. Finally, we make performance comparison among these classifiers.

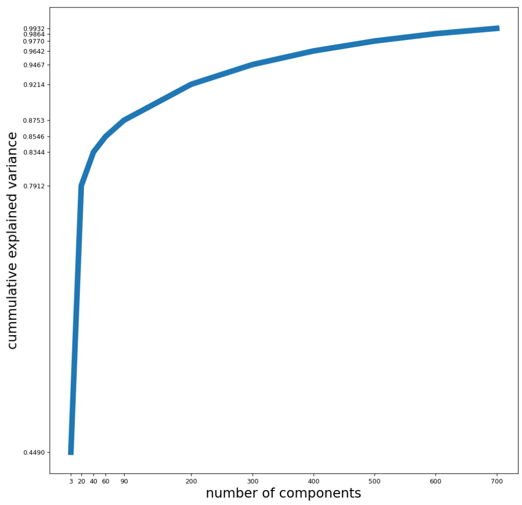

Figure 4. PCA Results

parameter	hidden layer size	learning rate
choices	(8), (4)	0.001, 0.01, 0.05
result	(8)	0.001

Table 2. Grid search result with original dataset

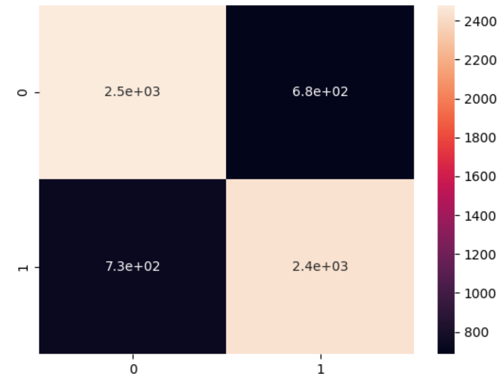

Figure 5. Confusion matrix of MLP without reducing the dimension of dataset

eval / num	training time	accuracy	AUC-ROC
200	25.04	0.86	0.86
300	30.33	0.88	0.88
400	30.63	0.90	0.90
500	29.30	0.92	0.92
600	39.36	0.93	0.93

Table 4. MLP performance with PCA transformation.

4.2. MLP

4.2.1. MLP with Original Dataset

MLP is widely used in pattern recognition and classification (11). After performing grid search on the grid shown row 2 of Table 2, the best model to use for the original non-reduction dataset has eight hidden layers with learning rate 0.001. Besides, we choose ReLU activation function and Adam optimizer.

The performance could be summarized by Table 3. To better explain the performance on classifying two different classes, the confusion matrix is shown as Figure 5. Since we only have two classes, we would not say we satisfy with the classification result, not to mention the computation efficiency.

measure	training time	accuracy	AUC-ROC
results	195.12	0.78	0.78

Table 3. MLP performance without PCA transformation.

4.2.2. MLP with PCA Transformed Datasets

The number of components we test for PCA transformation are 200, 300, 400, 500, and 600. Table 4 can be used as a reference for model comparison.

From the table 3 and 4, we can summarize that PCA does improve the model time and reduce the training time greatly. However, when we are trying to analyze the original input features, we should avoid using PCA since it maps the original features to their linear combinations.

4.3. Logistic Regression

The logistic regression model obtained mediocre results with an accuracy of .76 and AUC-ROC score of .93. The hyperparameter grid search was only performed on the regularization strength (which achieved a value of .5). However, one of the benefits of this model is the ability to examine relevant features by looking at absolute weights.

Figure 6 shows several important features determined by the logistic regression model. Namely, one can reference Channel 14/C1 (electrode) and channel 46/C3, both of which overlay the sensorimotor cortex in the contralateral hemisphere

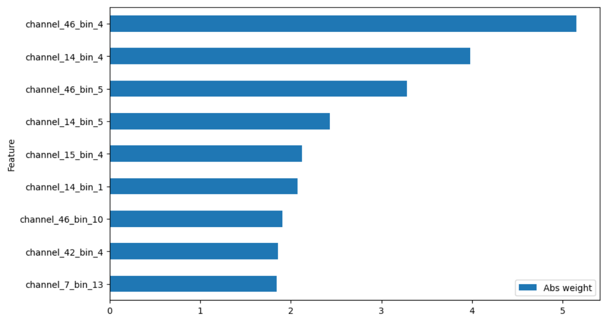

Figure 6. Top nine parameter weights of Logistic Regression model

of these subjects. Bin 4 and Bin 5 correspond to frequency bins of 10-12 Hz and 12-14 Hz, which is the frequency of the sensorimotor oscillations. A decrease in amplitude of these values is typically observed prior to and during movements (3). The key insight is that the physiologically-relevant features emphasized are related to our target variable. In addition, the training time is also relatively shorter at just 4.4 seconds.

4.4. Support Vector Machine

We used the hyperparameter C=1.0, kernel='rbf', and degree=3, for the SVM model. The AUC-ROC score achieved from this method is 0.86. Despite the model's potential, the achieved accuracy score was moderate at 0.78, and the training time is approximately 6 minutes. It's essential to note that SVMs, especially with large datasets, can be computationally intensive and demand substantial resources during training.

These findings highlight the trade-off between computational cost and model performance, emphasizing the need for careful consideration when choosing algorithms for large datasets.

4.5. Random Forest

The random forest model used hyperparameters of 64 estimators, 16 features, and a max depth of 32. Random forests use multiple decision trees (bagging) to increase variance within each decision tree and aggregating results. One of the benefits of the random forest model is we can increase the number of trees in the model without causing overfitting. This can be seen from the validation curves in Figure 8.

Similarly, we see the same kinds of features emphasized as in the logistic regression model once features are sorted by Gini importance in Figure 9. The training time of this model is longer than the logistic and dimensionality reduced MLP models at 42 seconds. This is expected due to needing to train many decision trees.

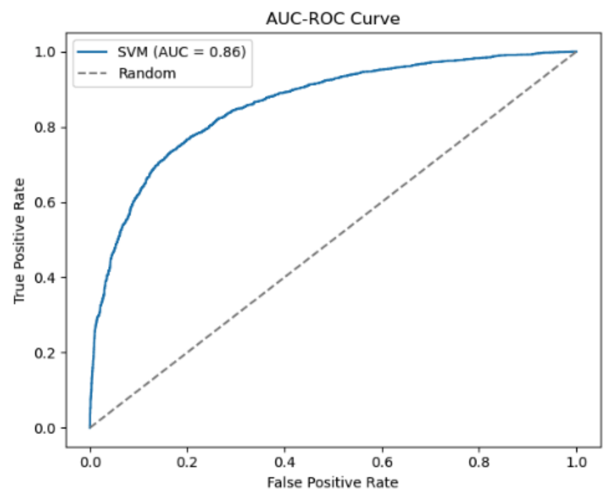

Figure 7. AUC-ROC curve for Support Vector Machine

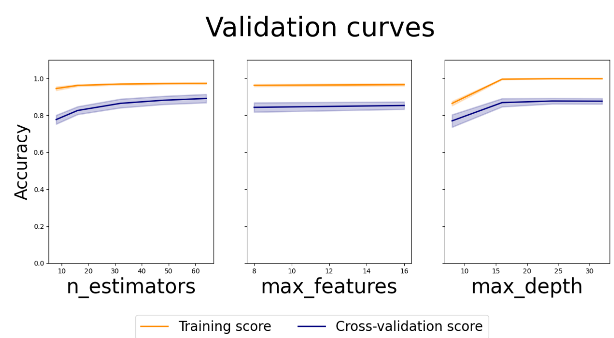

Figure 8. Validation curves across different hyperparameters for the Random Forest model

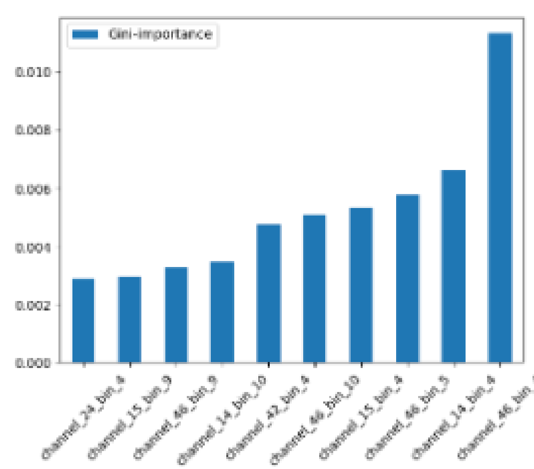

Figure 9. Top ten parameter weights of Random Forest model

4.6. Stacking

The Stacking ensemble method combines the strengths of multiple individual models to create a more robust and accurate predictive model. The best result we obtain from the Stacking method is using Decision Tree, Random Forest and Logistic Regression as base models, and Logistic Regression as the final estimator, with an accuracy score of 0.96. Although the accuracy score shows that the ensemble model is well performed, the training time for this model is approximately 10 minutes.

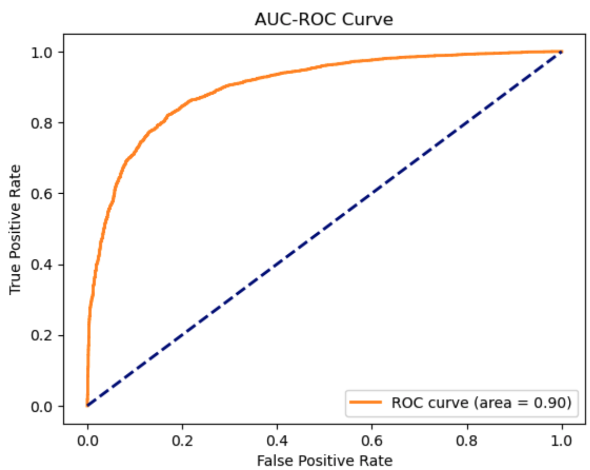

Figure 10. AUC-ROC curve for Stacking

5. Results and Comparison

The EEG dataset is provided by one of the authors Akhil Surapaneni. There is no previous work to compare with. However, we would like to compare the models we have already built and provide our suggestions on possible applications.

The following discussion is based on our current project. If researchers need to balance between computation efficiency and accuracy, models with PCA-MLP combination could be the final choice.

In the meanwhile, if researchers do not want to alter the input features and would like to analyze the original features, models with random forest and/or logistic regression are good choices.

Last but not least, there are circumstances where people only care about the classification performance. In this case, the stacking model in Section 4.6 should be under consideration.

6. Conclusion

We experimented with various models for the purposes of binary classification of EEG-based BCI for patients. In our experiments we found promising results without needing to use deep learning models by using dimension reduction with PCA and bagging methods. The MLP model with the transformed dataset was able to achieve an AUC-ROC score of .88 and the random forest model achieved an AUC-ROC score of .98. In addition, training time is crucial for neural engineering so we included those numbers in our report as well. The LSTM model we experimented with did not provide good results, likely because in EEG signals there is not enough temporal information over long periods of time that make this model meaningful. The Stacking ensemble method has a great performance of 0.96 accuracy score, but the trade-off for this model is that it is computational expensive and the training time is too significant.

In summary, our findings emphasize the efficacy of non-deep learning models, specifically highlighting the strengths of the MLP and random forest approaches. While the Stacking ensemble method exhibits impressive accuracy, its computational demands underscore the importance of considering practical constraints in model selection for real-world applications.

7. Limitation and Future Work

There are several things we could do to optimize the models. Firstly, we could search with larger parameter grids when looking for the hyper-parameters of the classifiers. Secondly, the experiment could be re-designed to capture EEG signals within a longer duration, so that we could get more temporal information. In addition, a suitable temporal related model is needed in order to achieve better performance.

8. Software and Data

We included a link to the code we used to train and evaluate our models in the reference section (15).

References

[1] How to mitigate overfitting with dimensionality reduction. https://towardsdatascience.com/how-to-mitigate-overfitting-with-dimensionality ~:text=This%20is%20because%20PCA%20removes,and%20increase%20the%20models%20performance.,.

[2] Abdi, H. and Williams, L. J. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[3] Andreas K Engel, P. F. Beta-band oscillations—signalling the status quo? *Current Opinion in Neurobiology*, 2010.

[4] Biasiucci, A., Leeb, R., Iturrate, I., Perdikis, S., Al-

Khodairy, A., Corbet, T., Schnider, A., Schmidlin, T., Zhang, H., Bassolino, M., Viceic, D., Vuadens, P., Guggisberg, A. G., and n, J. D. R. Brain-actuated functional electrical stimulation elicits lasting arm motor recovery after stroke. *Nat Commun*, 9(1):2421, Jun 2018.

[5] Craik, A., He, Y., and Contreras-Vidal, J. L. Deep learning for electroencephalogram (eeg) classification tasks: a review. *Journal of neural engineering*, 16(3): 031001, 2019.

[6] Haselsteiner, E. and Pfurtscheller, G. Using time-dependent neural networks for eeg classification. *IEEE transactions on rehabilitation engineering*, 8(4):457–463, 2000.

[7] Hochberg, L. R., Bacher, D., Jarosiewicz, B., Masse, N. Y., Simeral, J. D., Vogel, J., Haddadin, S., Liu, J., Cash, S. S., van der Smagt, P., and Donoghue, J. P. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485(7398): 372–375, May 2012.

[8] Kurita, T. Principal component analysis (pca). *Computer Vision: A Reference Guide*, pp. 1–4, 2019.

[9] Mao, W., Fathurrahman, H., Lee, Y., and Chang, T. Eeg dataset classification using cnn method. In *Journal of physics: conference series*, volume 1456, pp. 012017. IOP Publishing, 2020.

[10] McFarland, D. J. The advantages of the surface Laplacian in brain-computer interface research. *Int J Psychophysiol*, 97(3):271–276, Sep 2015.

[11] Noriega, L. Multilayer perceptron tutorial. *School of Computing. Staffordshire University*, 4(5):444, 2005.

[12] Tang, Z., Li, C., and Sun, S. Single-trial eeg classification of motor imagery using deep convolutional neural networks. *Optik*, 130:11–18, 2017.

[13] Tonin, L., Perdikis, S., Kuzu, T. D., Pardo, J., Orset, B., Lee, K., Aach, M., Schildhauer, T. A., nez Olivera, R., and n, J. D. R. Learning to control a BMI-driven wheelchair for people with severe tetraplegia. *iScience*, 25(12):105418, Dec 2022.

[14] Wang, P., Jiang, A., Liu, X., Shang, J., and Zhang, L. Lstm-based eeg classification in motor imagery tasks. *IEEE transactions on neural systems and rehabilitation engineering*, 26(11):2086–2095, 2018.

[15] https://drive.google.com/file/d/19qkjj1F_J-EZTDRnCBh47xkTheW4Gyfr/view?usp=sharing