Decoding EEG signals with various neural networks*

Andrea Do[1]

Australian National University, Canberra ACT 0200, Australia
ngoclinh.do@anu.edu.au

Abstract. We investigate the feasibility of building various neural network models for discerning whether an experimental subject is alcoholic or not based on EEG data collected from the subject. The model is trained on a data set of experimental results collected from 121 subjects. The hyperparameters for a two-layer feed-forward neural network with 192 input neurons and 2 output neurons, were optimized by validation accuracy to prevent over-fitting using a variety of training and validation methods. The optimized single neural network consisted of 40 hidden neurons trained with a learning rate of 10^{-5} trained over 500 epochs and no batching. The resulting accuracy is 73.47% on the training data and 73.72% on the testing data which has not been used for training and validation. This suggests that the neural network is robust enough to be a promising line of research for EEG signal analysis. To fully understand the potential of EEG signals, a stepwise neural network model is designed with an autoencoder which is trained with a learning rate of 10^{-4} over 100 epochs and a classifier which is a feedforward neural network trained over 100 epochs at 0.005 learning rate. The stepwise model achieves cross-subject accuracy at 75.97%. In addition, a parallel model of autoencoder and classifer, which is trained with a learning rate of 0.005 and a sum loss of Mean square and Cross entropy loss over 100 epochs, tops the within-subject accuracy with 79.95%. This suggests that the neural networks are robust enough to a promising line of research for EEG signal analysis.

Keywords: Neural Network · EEG · Input analysis · Autoencoder · Binary classification · CNN · Deep learning.

1 Introduction

Electroencephalography (EEG) is the collection of electrical signals from the brain from electrical sensors placed on the scalp of a subject. This typically produces complex time-series with complex patterns that are difficult to interpret using conventional methods. Being able to discern useful information from raw EEG signals about the mental state of the subject can have a massive impact on healthcare delivery in the medicine and an even greater impact on technology. With the potential to diagnose diseases and conditions from easily-obtainable EEG signals, doctors can be equipped with new tools to inform better treatments. Outside of healthcare, being able to decode user intention from EEG signals may also lay the foundations for commercially viable brain-computer interfaces that can be a platform for future technologies. However, extracting useful information from EEG signals is not trivial, since the signals are often interfered by noise. The format of EEG signals is heavily dependent on the measurement device and varies between individuals.

Neural networks are a class of machine-learning models that have been successful in solving a wide variety of classification and regression problems from large data sets. They rely on an empirically efficient set of techniques that can take advantage of large amounts of data to adjusting the parameters of the model to fit the given training data, while standard statistical techniques such as cross-validation among others can be used to prevent over-fitting.

On the other hand, deep learning has been widely used for feature learning tasks and attracted more attention from the fields of bioinformatics and machine learning. Deep learning extracts representation of data by processing data at multiple levels of abstraction [5]. Convolutional neural network (CNN) is one of the most powerful deep neural network with excellent performance in image classification [1]. The advantage of CNN is the convolution layer, which is often considered as a feature extractor. In contrast, Deconvolution neural network (DNN) performs reversing CNN to map the extracted features back to the input space.

An autoencoder can be built by combining CNN as encoder and DNN as decoder. This network topology behaves similarly to Principal Component Analysis (PCA) but without the linear constraint. Since an autoencoder is good at image compression, our inspiration is to extract features from EEG images using an autoencoder, then perform binary classification on these key features.

There is no reason *a priori* to believe that neural networks will fail at extracting useful information from EEG data given a sufficiently large enough data set of high enough quality.

* Supported by College of Engineering and Computer Science, ANU.

This paper is an attempt to investigate the feasibility of using a single neural network and a deep neural network to decode EEG data by focusing on the specific task of training and optimizing various neural network models that determine whether or not a subject is alcoholic or not based on their EEG signals when exposed to various stimuli.

2 Methodology

2.1 Dataset

The raw data is given in tabular format collected from an experiment involving 122 subjects. Upon being exposed to some external stimuli, EEG signals were collected from each subject in the experiment, which is repeated in a variable number of trials. Each row in the raw data represents such a trial. Whether the subject in each trial is alcoholic is also recorded. The goal of the model in this paper is to predict this using the other features in the table.

In total, there are 192 columns for the pre-processed EEG data, which are of numeric type, plus 1 column for the subject ID, 1 column for the trial number for that subject, 1 column for whether or not the subject is alcoholic and 1 column for the stimulus the subject is given in the form of a number from 1 to 5.

The $32 \times 32 \times 3$ pre-processed EEG images are produced from the raw EEG signals by combining temporal and spatial information [2]. To map the three dimensional 64 channel position into a 2-D space, all electrodes positions are projected using Azimuthal Equidistant Projection (AEP). On an RGB EEG image, the red colour is the theta frequency at 4-7 Hz, the green colour is the alpha frequency at 8-13 Hz and the blue colour is the beta frequency at 13-30 Hz [8].

Two-thirds of the subjects were alcoholic, which means the data is skewed, so care must be taken in the model to account for this potential bias.

2.2 Batch training

In our data set, the distribution of alcoholic and non-alcoholic subjects is heavily skewed, with twice as many alcoholic participants as non alcoholic participants. The imbalanced data set might encourage the model to favour the dominant population, in this case, alcoholic label. Training data in mini batches to re-balance the label distribution would allow the model to train on as many non-alcoholic participants as alcoholic participants. As a consequence, the model should learn to perform a fair binary classification instead of blindly classifying all input features as one class to greedily reduce loss.

2.3 Within-subject and Cross-subject validations

EEG data is highly variable among different individuals [8]. There are two common ways to split data into training, validation and training data sets: cross-subject and within-subject, both of which we investigate.

To test a model with cross-subject validation, the model is trained and then tested on two separate groups of individuals. Whereas, within-subject validation uses a proportion of data from all participants for training and the remaining data for validating.

The cross-subject test result is often much lower than the within-subject test result. Inspired by Yao [8], the data set was randomly split in a 7:1:2 ratio for training, validation and testing respectively for both within-subject and cross-subject testing.

2.4 Single neutral network

All single neural networks in this paper are trained using error back-propagation [7], in which all neurons are fully connected to the next layer with no loops or multi-layer connections. The sigmoid function is used as the activation function for the last hidden layer to perform binary classification,

$$\mathcal{S}(x) = \frac{1}{1 + e^{-z}}. \tag{1}$$

To measure the classification performance of the network, the training and testing loss are quantified by the cross-entropy loss function

$$\mathcal{L}(X, Y) = -\frac{1}{n} \sum_{i=1}^{n} y_i \ln a(x_i) + (1 - y_i) \ln(1 - a(x_i)). \tag{2}$$

The goal of this study is to classify participants into two groups: non-alcoholic and alcoholic, purely based on EEG signal data. Since it is a classification problem with two possible outputs, the number of output neurons in the neural network is two, corresponding to a one-hot encoding of the labels alcoholic and non-alcoholic, while the number of input neurons is 192, which is the number of signal channels for each trial. We first build a simple neural network with one layer of hidden neurons. The network is trained, validated and tested with the Adam optimiser and the batch size is set to 64. After testing a range from 10-190 hidden neurons, we choose to train the network with 40 hidden neurons. The network is tuned by testing six different learning rates ranging from 10^{-5} to 1, coupling accuracy and loss values in cross-subject and within-subject validations, we decide to train the network at 10^{-4} learning rate with 500 epochs.

Weight matrix analysis (W) The model has almost 200 inputs neurons, which is a relatively large number. Performing back-propagation on a large neural network is computationally expensive and slow to train. The contribution of an input feature to an output neuron can be calculated as a product of the contribution of the input to all neurons in the hidden layer and the contribution of all hidden neurons to an output neuron [4].

$$Q_{ik} = \sum_{j=1}^{nh} \left(\frac{|w_{ij}|}{\sum_{p=1}^{ni} |w_{pj}|} \times \frac{|w_{jk}|}{\sum_{r=1}^{nh} |w_{rk}|} \right) \tag{3}$$

where w_{hj} is weight between the input layer and the hidden layer, whereas w_{hk} is weight between the hidden layer and the output layer. This approach only considers the magnitude of the weight while disregarding the sign, which has been shown to perform better than other earlier methods in the literature [3]. The contributions of all input features are ranked based on the average contribution of an individual input neuron to the two output neurons.

Functional measures (F) were introduced in [4] as a better method to quantify the contribution of inputs to outputs over the weight matrix analysis approach. Each input neuron is represented by a weight vector consisting of weights between itself and all connecting hidden neurons. The distinctiveness of any two input features is measured using the angle formed between two weight vectors.

The angle between two weight vectors of the input neurons was calculated using the modified weight vector

$$\text{norm}(\text{weight}(h)) - 0.5 \tag{4}$$

where h is weight of the input neuron and norm is interpreted as being the L^1 norm. However this gives unreasonable results, since the angle of a vector with itself becomes non-zero, apparently forming an 80 degree angle with itself. As such, this approach is abandoned. With an inspiration from the theory, a simpler calculation is implemented to compute a dot product between two weight vectors of the input neurons then convert it to degree.

Pruning Redundant Neurons To make network training more efficient, one may attempt to reduce the number of input neurons by eliminating redundant input neurons. Weight matrix and functional measures analysis are applied on all 192 input neurons to rank the average distribution of an individual input neuron to an output neuron. The top 10 most distinct input neurons are listed in Table 1. Input neuron Beta 33 constantly contributes the highest weight in both weight and functional measures analysis. Even though it is a two-fold difference in the contribution of weights between that of the most distinct neuron to the least distinct neuron, they are still within the same order of magnitude and therefore not unreasonable. The boldfaced signals in Table 1 are the ones that consistently rank in the top 10 most distinct input neurons across all validation methods.

Table 1. Top 10 most distinct input neurons

W, Cross validation	**Beta 33**	Beta 28	**Beta 29**	Beta 30	Alpha 34	**Beta 9**	Beta 18	Alpha 30	Alpha 49	Theta 2
W, Within validation	**Beta 33**	**Beta 12**	**Beta 9**	Alpha 49	Theta 29	Beta 60	**Alpha 61**	Alpha 34	Beta 30	Theta 11
F, Cross validation	**Beta 33**	**Beta 9**	Alpha 29	Beta 42	Theta 44	Alpha 31	Beta 30	**Beta 29**	Theta 38	Alpha 46
F, Within validation	**Beta 33**	**Alpha 61**	**Beta 9**	Alpha 43	Theta 44	**Beta 12**	Beta 30	Beta 60	Theta 62	Alpha 46

Since the least distinct neurons contribute less, it is possible that removing the less distinct neurons will increase the efficiency while keeping performance high. We attempt to prune the network by distributing weights of the pruned neurons to the remaining input neurons. To compare the performance of weight matrix and functional measures analysis, the top 10 and the bottom 10 distinct input neurons are pruned, respectively. A randomly chosen group of 10 input neurons is pruned to act as a control.

A majority of pruned networks outperform the three layer neural network in both within and cross-subject validation, as shown in Tables 2 and 3. As expected, excluding the redundant input neurons improves the classification performance of the network. Surprisingly, removing the most distinct input neurons also increases the performance of the binary classification. In addition, the highest testing accuracy is achieved when dropping 10 random input neurons. It suggests that the remaining neurons learn more efficiently after pruning the network. As a consequence, neural network pruning improves the generalisation of the model.

Table 2. Compare pruned models where 10 input neurons have been removed using within-subject validation.

	Testing accuracy (%)	Recall	Precision	F-measure
Single neural network	72.79	70.50	43.31	53.65
Pruned bottom 10, W	**74.78**	**73.33**	48.18	58.15
Pruned top 10, W	74.34	72.92	46.83	57.04
Pruned bottom 10, F	74.51	72.78	47.81	57.71
Pruned top 10, F	74.47	73.07	47.20	57.35
Pruned random 10	**74.82**	73.30	**48.42**	**58.32**

Table 3. Compare pruned models where 10 input neurons have been removed using cross-subject validation.

	Testing accuracy (%)	Recall	Precision	F-measure
Single neural network	73.72	66.19	42.22	51.56
Pruned bottom 10, W	73.33	63.93	44.71	52.62
Pruned top 10, W	74.03	65.60	**45.36**	**53.63**
Pruned bottom 10, F	**74.24**	**66.87**	44.05	53.11
Pruned top 10, F	73.77	65.20	44.58	52.95
Pruned random 10	73.46	64.18	44.97	52.88

2.5 Autoencoder

To fully understand the potential of the EEG signals, we take extra steps to design an autoencoder, which is inspired by the image-wise autoencoder designed by Yao, Plested and Gedeon [8]. The autoencoder consists of two parts: encoder and decoder. To speed up the training process, the Rectified Linear Unit (ReLU) is used for activation layers. A 25% dropout is applied after every activation layers to improve model generalisation. The autoencoder is optimised with Adam optimiser and Mean Square Loss function. The batch size is set to 64. Weights for convolution kernels are initialised using Xavier normal initialisation. In addition to a normal CNN autoencoder, we also train a shared weight CNN autoencoder in which the weight of the deconvolution layers is the same as the weight of the convolution layer, Table 4.

Encoder	Decoder
Input 32 × 32 ×3 EEG image	Input 16 × 8 ×8 compressed matrix
3 × 3 conv, ReLU, 2 × 2 max-pooling, 0.25 dropout	3 × 3 deconv, ReLU, 2 × 2 max-un-pooling, 0.25 dropout
3 × 3 conv, ReLU, 2 × 2 max-pooling, 0.25 dropout	3 × 3 deconv, ReLU, 2 × 2 max-un-pooling, 0.25 dropout
3 × 3 conv, ReLU	3 × deconv

Table 4. Autoencoder design structure

2.6 Classifier

To perform binary classification, the feature extracted from image-wise autoencoder is flattened into a long vector which is then fed to a feedforward network with 16 hidden neurons and 2 output neurons. Sigmoid activation function is used to define whether the EEG signals belong to a non-alcoholic or alcoholic participant. The network is trained in batches with the batch size is set to 64. Cross Entropy Loss and Adam optimiser are used to optimise the model at 0.005 learning rate.

2.7 Stepwise and parallel design

The key feature of our network topology is that an autoencoder is trained on EEG images then the extracted features is passed to a classifier to perform binary classification. There exists two network structures that both meet the requirements.

In the stepwise design, the network is trained in two separate steps. The autoencoder is trained first, at 10^{-4} learning rate with 100 epochs, but only the weight of the encoder is saved. In addition to the fully connected layers, the classifier is built with an encoder layer whose weight is initialised to be the weight of the encoder layer in the pretrained autoencoder. The learning rate of the encoder layer is set to 10^{-7} with 100 epochs.

On the other hand, the autoencoder and classifier are trained together at 0.005 learning rate with 100 epochs, in the parallel design. A combined loss of Mean Square Loss and Cross Entropy Loss is used to optimise the network. The extracted features are directly inputted to the fully connected layers to perform binary classification.

The deep neural networks are trained on GPU provided by Google Colab.

Table 5. Performance of various deep neural networks in predicting non-alcoholic participants using cross-subject validation.

	Testing accuracy (%)	Recall	Precision	F-measure
Single neural network	73.72	66.19	42.22	51.56
Parallel shared weight autoencoder and classifier	68.83	53.00	52.03	52.51
Parallel normal autoencoder and classifier	64.55	47.11	57.52	51.8
Stepwise shared weight autoencoder and classifier	72.08	58.04	56.6	57.31
Stepwise normal autoencoder and classifier	**75.97**	**73.44**	53.16	**61.67**
CNN without pretrained autoencoder	66.88	50.00	**69.67**	58.22

Table 6. Performance of various deep neural networks in predicting non-alcoholic participants using within-subject validation.

	Testing accuracy (%)	Recall	Precision	F-measure
Single neural network	72.79	70.50	43.31	53.65
Parallel shared weight autoencoder and classifier	75.49	66.92	64.48	65.58
Parallel normal autoencoder and classifier	**79.96**	72.58	72.14	**72.36**
Stepwise shared weight autoencoder and classifier	74.73	68.89	53.65	60.70
Stepwise normal autoencoder and classifier	74.87	**73.18**	48.78	58.54
CNN without pretrained autoencoder	74.42	61.17	**81.27**	69.80

Table 7. Performance of various deep neural networks in predicting alcoholic participants using cross-subject validation.

	Testing accuracy (%)	Recall	Precision	F-measure
Single neural network	73.72	**91.74**	73.51	81.62
Parallel shared weight autoencoder and classifier	68.83	78.44	74.08	76.20
Parallel normal autoencoder and classifier	64.55	68.57	73.83	71.10
Stepwise shared weight autoencoder and classifier	72.08	80.94	76.52	78.67
Stepwise normal autoencoder and classifier	**75.97**	**89.02**	76.86	**82.50**
CNN without pretrained autoencoder	66.88	65.28	**79.00**	71.49

Table 8. Performance of various deep neural networks in predicting alcoholic participants using within-subject validation.

	Testing accuracy (%)	Recall	Precision	F-measure
Single neural network	72.79	89.66	73.43	80.74
Parallel shared weight autoencoder and classifier	75.49	81.79	80.01	80.03
Parallel normal autoencoder and classifier	**79.96**	84.43	84.12	**84.28**
Stepwise shared weight autoencoder and classifier	74.73	86.79	76.60	81.38
Stepwise normal autoencoder and classifier	74.87	**89.80**	75.39	81.97
CNN without pretrained autoencoder	74.42	70.50	**86.80**	77.80

Deep neural networks seem to perform better than the single neural network in binary classification task, Table 5, 6, 7, 8. It is important to note that the three-layer neural network has higher cross-subject accuracy score than most of the CNNs, excluding the stepwise normal autoencoder and classifier. Due to the imbalanced data set, we do not only compare the testing accuracy but also recall, precision and F-measure in both alcoholic and non-alcoholic identification. In general, all neural networks perform better in predicting alcoholic than non-alcoholic participants. This could be explained by the dominant of EEG signals collected from alcoholic subjects in the data set.

With respect to identifying alcoholic participants, recall is the proportion of real alcoholic participants that are correctly predicted alcoholic, whereas, precision is the proportion of the alcoholic predicted individuals that are correctly alcoholic [6]. F-measure is a balanced mean of recall and precision. The single neural network has the lowest cross-subject and within-subject precision, meaning that the proportion of the alcoholic predicted individuals that are correctly alcoholic is insignificant. The network has a tendency to predict more alcoholic individuals with an intended of increasing the chance that some of these participants are actually alcoholic.

Despite the absence of the pre-trained weight from the autoencoder, the CNN consistently achieves the highest within-subject and cross-subject precision scores. Conversely, the CNN has the lowest within-subject and cross-subject recall among all other neural networks. The CNN is not good at detecting alcoholic participants, but once it does classify an individual as alcoholic, the model has a great chance of being correct. Furthermore, it illustrates the vital role of the autoencoder in extracting the distinctiveness in the EEG signals measured in alcoholic and non-alcoholic individuals.

In cross-subject validation, the stepwise normal autoencoder and classifier achieves state-of-art accuracy at 75.97%, Table 5. The current highest published cross-subject accuracy is 75.60% by Yao et.al, [8]. Consistently in within and cross-subject validation, the stepwise network correctly classifies more than 73% of non-alcoholic participants and 89% alcoholic participants. It suggests that the autoencoder is capable of extracting and detecting the distinct features in the EEG images of alcoholic and non-alcoholic participants.

EEG signals are highly variable between individuals and measuring devices. This leads to a trade off between the within-subject and cross-subject accuracy. The parallel normal autoencoder and classifier has the highest within-subject accuracy at 79.96%. However, the parallel design of autoencoder and classifier performs poorly in cross-subject validation, Table 5. As the model is better at predicting EEG signals from a particular individual, it losses its ability to generalise the distinct features between non-alcoholic and alcoholic subjects. Conversely, the stepwise autoencoder and classifier has a greater advantage in cross-subject validation but its performance is not as good as the parallel neural network in within-subject validation. Hence, it depends on the practical application of the deep neural network, one would choose the more suitable network topology. Even though the binary classification performances of normal and shared weight autoencoders are negligibly different, fixing the weight of the decoder to be the same as the weight of the encoder results in better reconstructed images, Figure 1.

3 Limitations and Further Work

The main limitation to this study is the available computational power, which severely restricts the ability to comprehensively search the space of hyperparemeters to optimize the model. While the tuning strategy for the hyperparameters may be sufficient as a proof-of-concept demonstration for this particular problem, the performance of the model can certainly be improved with greater computational power even with the same data set. For example, with greater computing resources, batch training can be conducted on a higher number of epochs to ensure escape from any local minima in parameter space the model may be confined within.

On the other hand, with more structured time-series data where the subjects are exposed to more targeted stimuli, more elaborate and sensitive experiments could be performed to yield data that is more amenable to decoding by Long short-term memory (LSTM) and other recurrent neural networks. It would be interesting to know the matrix that separates participants into non-alcoholic and alcoholic groups. As it is a continuous scale

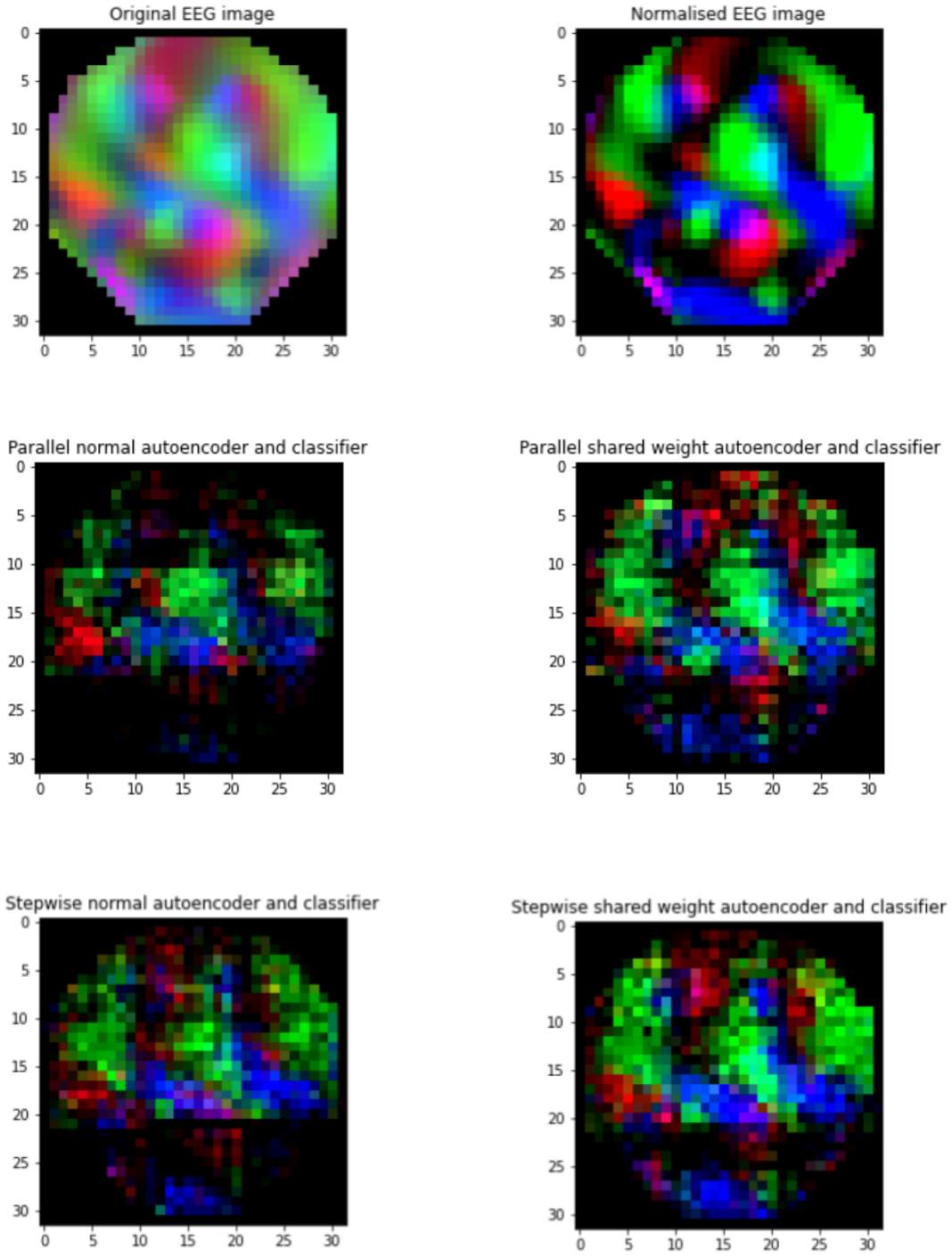

Fig. 1. Recreated EEG images from autoencoder.

from non-alcoholic to alcoholic, fuzzy logic would also be implemented. It is important to monitor the sensitivity and accuracy of the neural networks when individual(s) are moved between the binary groups.

4 Conclusion

We have investigated the feasibility of building various neural network classification models for discerning whether an experimental subject is alcoholic or not from EEG data collected from the subject, trained using a data set of experimental results collected from over a hundred subjects. Starting from a two-layer feed-forward neural network with 192 input neurons, 40 hidden neurons and 2 output neurons, its hyperparameters are optimized for validation accuracy to prevent over-fitting using a variety of training and validation methods. The optimized network is trained with a learning rate of 10^{-5} trained over 500 epochs and no batching. The resulting accuracy is 73.47% on the training data and 73.72% on the testing data which the model has not seen for training and validation.

We take extra steps in designing deep neural networks with autoencoder and classifer. The stepwise neural network architecture consists of an autoencoder which is trained at 10^{-4} with 100 epochs and optimised by Mean square loss and a classifer which is trained at 0.005 learning rate with 100 epochs and optimised by Cross entropy loss. The stepwise topology achieves cross-subject state-of-art accuracy at 75.97%. Whereas, the autoencoder and the classifier are trained together in the parallel neural network architecture. The parallel model, which is trained by a combined loss of Mean square and Cross entropy loss at 0.005 learning rate with 100 epochs and optimised by Adam optimiser, achieved the highest within-subject accuracy at 79.95%.

With future improvements previously mentioned in both increased computational power, more sophisticated architectures, training and validation techniques and better experiments, this accuracy can be future improved upon in the future.

Therefore it can be concluded that using neural networks to decode EEG signals is possible, although many further studies and improvements can be made.

References

1. Albawi, S., Mohammed, T.A., Al-Zawi, S.: Understanding of a convolutional neural network. In: 2017 International Conference on Engineering and Technology (ICET). pp. 1–6. Ieee (2017)
2. Bashivan, P., Rish, I., Yeasin, M., Codella, N.: Learning representations from eeg with deep recurrent-convolutional neural networks. arXiv preprint arXiv:1511.06448 (2015)
3. Garson, D.G.: Interpreting neural network connection weights (1991)
4. Gedeon, T.D.: Data mining of inputs: analysing magnitude and functional measures. International Journal of Neural Systems **8**(02), 209–218 (1997)
5. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
6. Powers, D.M.: Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. arXiv preprint arXiv:2010.16061 (2020)
7. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. Tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science (1985)
8. Yao, Y., Plested, J., Gedeon, T.: Deep feature learning and visualization for eeg recording using autoencoders. In: International Conference on Neural Information Processing. pp. 554–566. Springer (2018)