Journal of Electronics and Informatics (ISSN: 2582-3825)
www.irojournals.com/iroei/

Alzheimer's Classification from EGG Signals Employing Machine Learning Algorithms

Nagarathna C R[1], Kusuma M[2], Anusha Preetham[3], Muquitha Almas[4]

[1,3]Associate Professor, BNM Institute of Technology, Bangalore

[2]Professor, DSATM, Bangalore

[4]Assistant Professor, DSCE, Bangalore

E-mail: [1]nagarathna.binu@gmail.com, [2]kusumalak@gmail.com, [3]raoanusha1606@gmail.com,

[4]almas-cs@dayanandasagar.edu

Abstract

The study has shown how classifiers behave when identifying and categorizing Alzheimer's disease stages. The main characteristics of various frequency bands were fed into the classifier as input. The accuracy of recognition is evaluated using machine learning classifiers. The effort aims to create a novel model that combines pre-processing, feature extraction, and classification to identify different stages of disease. The study starts with band filtering, moves on to feature extraction, which derives several bands from the EEG signals, and then employs KNN, SVM, and MLP algorithms to measure classification performance. AD detection and classification using machine learning classifiers such as KNN, SVM, and MLP is the main focus of this research. Five wavelet band characteristics are used by the built-in classifiers to categorize different disease phases. These characteristics are computed using DWT, PCA, and ICA, which aid in obtaining wavelet-related knowledge for learning. The proposed machine learning model achieves a classification accuracy of 95% overall.

Keywords: EEG, PCA, ICA, DWT, MLP

DOI: https://doi.org/10.36548/jei.2023.4.003
Received: 16.10.2023, received in revised form: 08.12.2023, accepted: 20.12.2023, published: 26.12.2023

1. Introduction

Currently, EEG analysis can be used to recognize the signs of diseases affecting how the brain functions. although it is not yet clear how these characteristics relate to AD. EEG provides non-invasive, cost-effective data with great temporal precision on electrical activity in the brain during neurotransmission as compared to other imaging modalities. Using a discrete wavelet transformer [1-5] to decompose the EEG signal into its frequency sub-bands and extract a set of statistical features to represent the distribution of wavelet coefficients, the proposed framework's processing and analysis of EEG were both carried out. Data dimension reduction techniques include principal component analysis [6-9] and independent component analysis (ICA) [10-12]. These attributes were then sent into Multi-Layer Perceptron (MLP) and a Support Vector Machine (SVM) that could only produce one of two outcomes: AD or NC. To determine the classification process, the process of various approaches is shown and contrasted. These findings provide an example of how to use data from particular mild epileptic patients to train and test an Alzheimer's detection prediction algorithm. These kinds of technologies will probably be required to customize intelligent epilepsy treatment devices to each individual's neurophysiology before they are placed into clinical usage, given the variety of epilepsy.

2. Proposed Model

The suggested EEG-based AD identification model is depicted in Fig.1. The EEG brain signals undergo preliminary processing to eliminate noise, separate the signals into several frequency bands, and shrink the size of the signals. The feature extraction method is used after the pre-processing step to extract latent components from the input EEG signal and identify AD. The literature shows that PCA, ICA DWT shows the best performance in extracting the feature from EEG. In our work, also experimented with these techniques [13]. These features are then fed into the classifiers SVM, KNN, and MLP for classification. With respect to KNN, SVM, and MLP displays superior results. Finally, accuracy, sensitivity, and specificity are evaluated in order to gauge the model's performance.

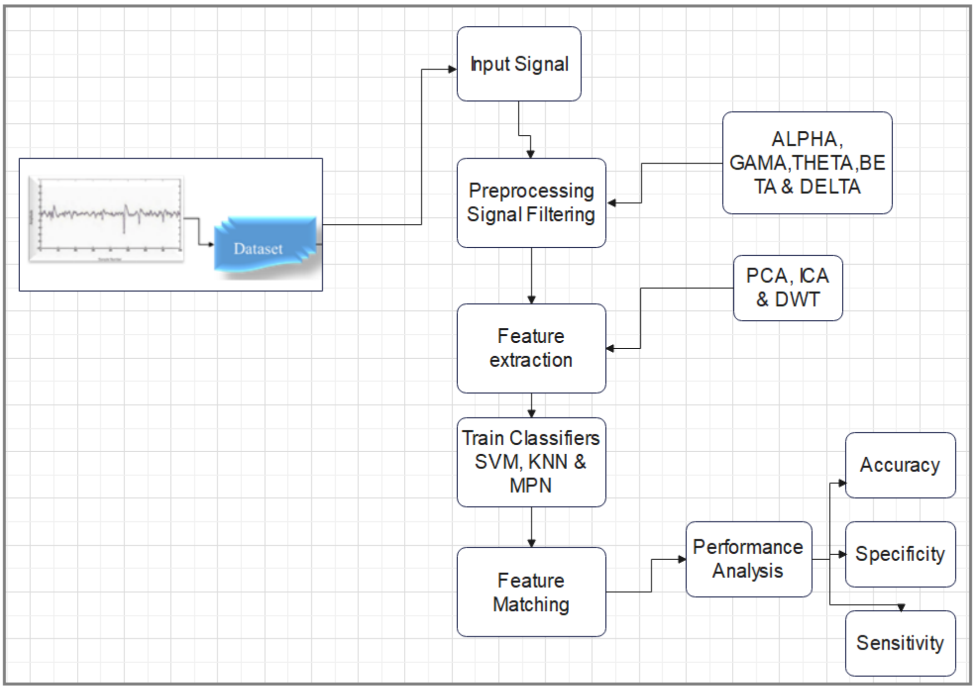


Figure 1. Proposed Model for AD Prediction

The algorithm 1.1 shows the methodology of the proposed work Prior to anything else, pre-process the input EEG signals and use the ICA approach to extract the latent components from the pre-processed signals. The model is trained using more extracted features. Using the test dataset, evaluate the model at the end.

Algorithm 1.1: AD disease classification using EEG signal.

Input: EEG Signals

Output: Classify the input signals

Begin

Step1: Pre-process the EEG signals.

- Filter the input signals.
- Obtain subbands –Alpha, Beta, Gamma, Theta and Delta, of EEG signal.

Step 2: Label the dataset.

Step 3: Divide the dataset into train and test sets.

Step 4: Extract the features by applying PCA, ICA and DWT.

Step 5: Train the classifier models (MLP and SVM) by using extracted features.

Step 6: Accept test signals and record the result.

End

2.1 Dataset

The EEG signals gathered from Baskent University Hospital's Neurology Department were taken into consideration for this proposed effort [4]. "19 electrodes (Fp2, Fp1, F8, F7, F4, F3, A2, A1, T4, T3, C4, C4, T6, T5, P4, P3, O2, O1) were used to record the EEG signals". These electrodes were positioned in accordance with the global 10-20 system. Three channels were referenced using monopolar montage, and the remaining 15 channels were all referenced using longitudinal bipolar montage. The channel electrode montage is shown in Fig 2[5].

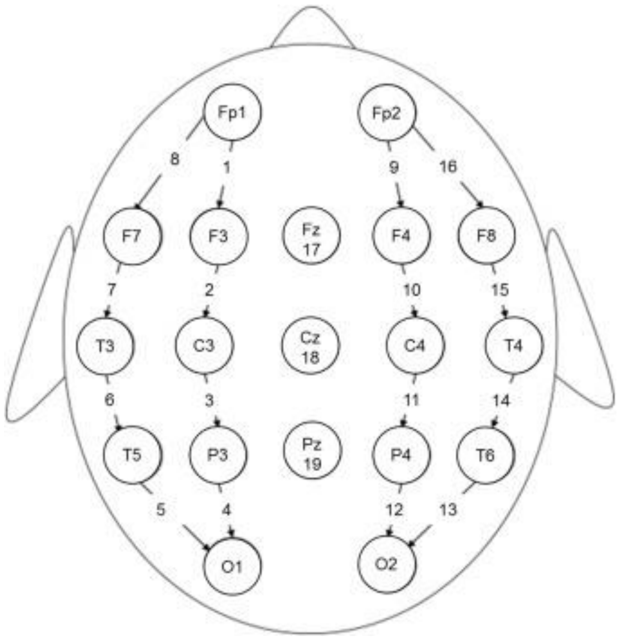


Figure 2. Channel Electrode Montage [5]

2.2 Dataset Preprocessing

EEG signals are routinely corrupted with different artifacts while being recorded. Motion, muscular, ocular, and cardiac artefacts are the most typical kinds of artefacts. Thus, the first step in the proposed approach is to apply a bandpass filter to eliminate the noise. Frequencies above 60 Hz are frequently labelled as noise and eliminated. In order to reduce the noise, a bandpass filter and an IIR filter is utilized to perform the preprocessing. The filter' networks are used to handle signals in a frequency-dependent way. Signal distortion, also known as one common term used to characterize a persistent unwanted change in a signal is "signal fading," which can also refer to reverberations, echo, multipath reflections, and missing samples. A bandpass filter selectively rejects signals at other frequencies, allowing signals between two preset frequencies to pass through the poles of the analog lowpass filter. The poles of a Butterworth filter of order N with Ωc = 1 rad/s are as follows:

$$p'ak = -sin(\theta) + jcos(\theta) \quad 1.1$$

$$\theta = \frac{(2k-1)}{\pi 2N}, k = 1\colon N \quad 1.2$$

Here, in order to differentiate the prototype poles of lowpass from the bandpass poles that have yet to be calculated, we place a prime superscript on p. Given the upper and lower -3 dB frequencies of the digital bandpass filter, find the equivalent frequencies of the analog bandpass filter. The - bwHz is the 3 dB bandwidth in Hz, and the center frequency is the fcenter. Then, at -3 dB, the discrete frequencies are:

$$f1 = fcenter - bwHz/2 \quad 1.3$$

$$f2 = fcenter + bwHz/2 \quad 1.4$$

As previously, analog frequencies are prewrapped to account for the bilinear transform's nonlinearity.

$$F1 = \frac{fs}{\pi} tan(\pi f1fs) \quad 1.5$$

$$F2 = \frac{fs}{\pi} tan(\pi f2fs) \quad 1.6$$

Two further quantities need to be defined:

pre-warped -3 dB bandwidth $BWHz = F2 - F1$ Hz

F1 and F2 geometric mean $F0 = \sqrt{F1F1}$.

Analog bandpass poles are created by converting the analog lowpass poles. We obtain two bandpass poles for each lowpass pole pa'.

$$P_a = 2\pi[\frac{BW_{Hz}}{2F0} p'_a \pm j\sqrt{1 - (\frac{BW_{Hz}p'_a}{2F0}^2)}] \quad 1.7$$

Utilize the bilinear transform to move the poles connecting the s- and z-planes. The only difference is that there are 2N poles rather than N poles, much like with the IIR lowpass:

$$P_k = \frac{1+p_{ak}/2f_s}{1-p_{ak}/2f_s} \quad k = 1 \; to \; 2N \tag{1.8}$$

N zeros at z=+1 and at z=-1 must be added. The H(z) can be written as follows:

$$H(z) = K\frac{(z+1)^N(z-1)^N}{(z-p_1)(z-p_2)(z-p_3)\ldots} \tag{1.9}$$

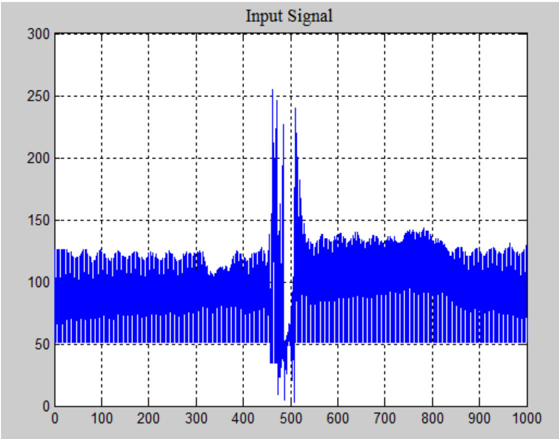
The N zeros at +1 and -1 are represented as a vector in bp_synth:

q= [-ones(1,N) ones(1,N)] 1.10

Poles and zeros can be converted to generate polynomials with coefficients a and bn. If we multiply the numerator and denominator of equation 1 by z2N and then expand the numerator and denominator, we get polynomials in z-n.

$$H(z) = K(z + 1\frac{b_0+b_1z^{-1}+\cdots b_{2N}z^{-2N}}{1+a_1z^{-1}\ldots a_{2N}z^{-2N}}) \tag{1.11}$$

The input noise signal and output of bandpass filter are shown in Fig 3 and 4 respectively.



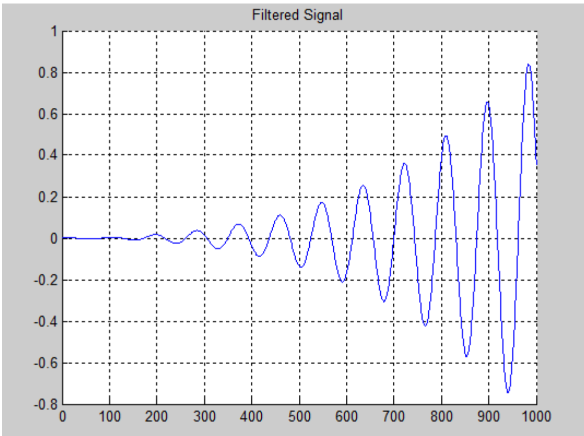


Figure 3. Input Noise Signal **Figure 4.** Filtered Signal

2.3 Feature Extraction

With the help of EEG investigations, it has been determined that there are five main frequency bands for EEG signals and a relationship exists between a certain brain region's neuronal activity and behavior. Delta (0.1 Hz or 0.5 Hz), Theta (4 Hz), Alpha (8 Hz), Beta (14 Hz), and Gamma (30 Hz–63 Hz are the most commonly used frequency bands [10-15]. To extract characteristics from a signal on various scale DWT is utilized after sequential high pass and low-pass filtering. The EEG signals are divided into five frequency sub-bands in this case using the DWT. The approximation and detail coefficients continue in the wavelet coefficients in a sequential manner. A transform that generates the fewest coefficients necessary to accurately recover the original signal is favored in applications that call for bilateral transformations. In order to achieve this symmetry, the DWT minimizes the range of translation and scale change, which are frequently in powers of 2. Most signal and image processing applications can be best described in terms of filter banks when utilizing DWT-based analysis. Sub-band coding is the technique of breaking a signal up into its component spectral areas using a series of filters. The five decomposed bands of the Normal and AD signals are shown in Figs. 5 through 14.

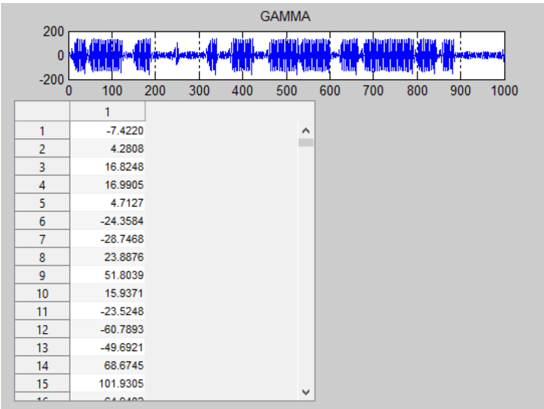


Figure 5. NC GAMA

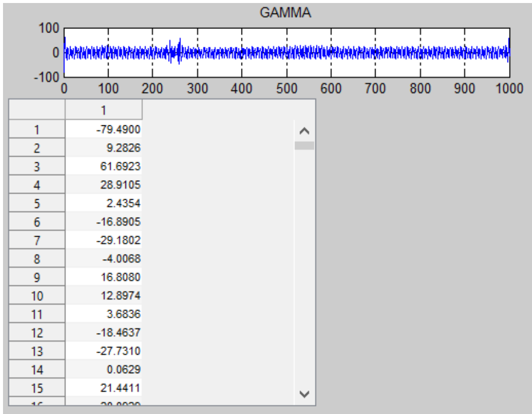


Figure 6. AD GAMA

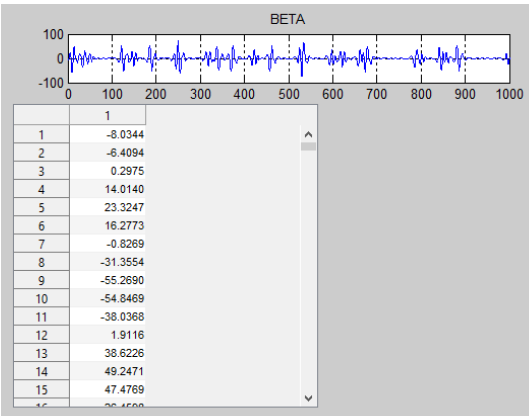


Figure 7. NC BETA

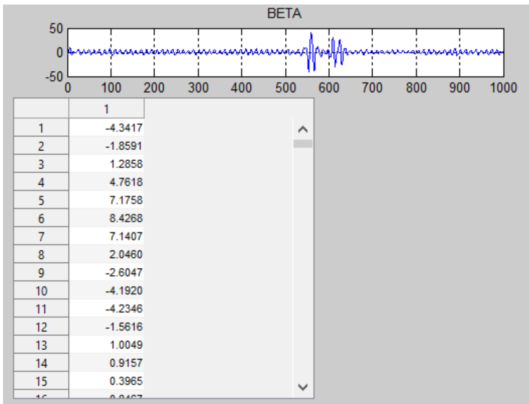


Figure 8. AD BETA

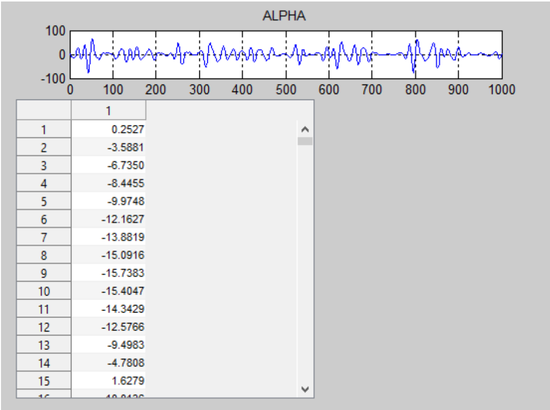


Figure 9. NC ALPHA

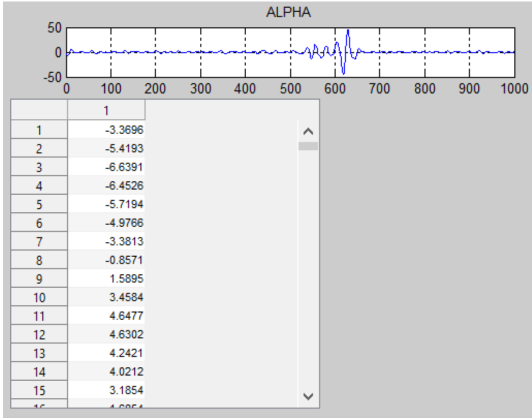


Figure 10. AD ALPHA

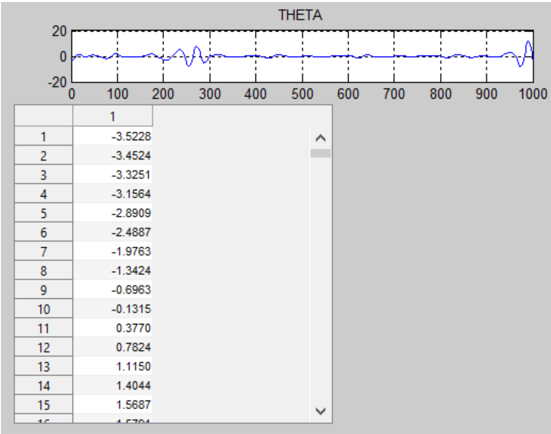


Figure 11. NC THETA

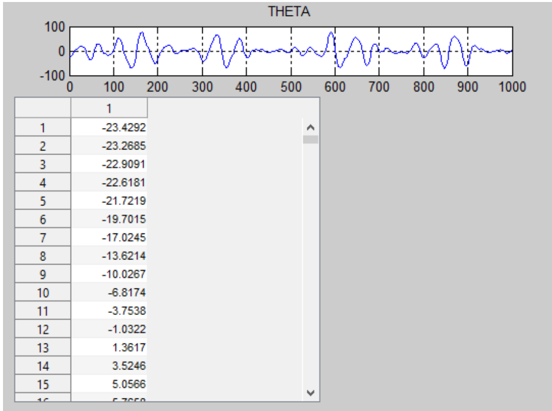


Figure 12. AD THETA

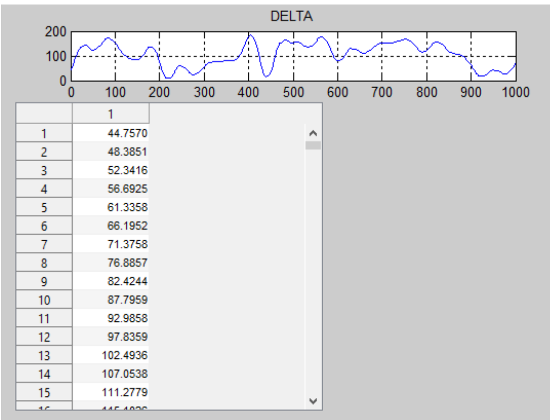


Figure 13. NC DELTA

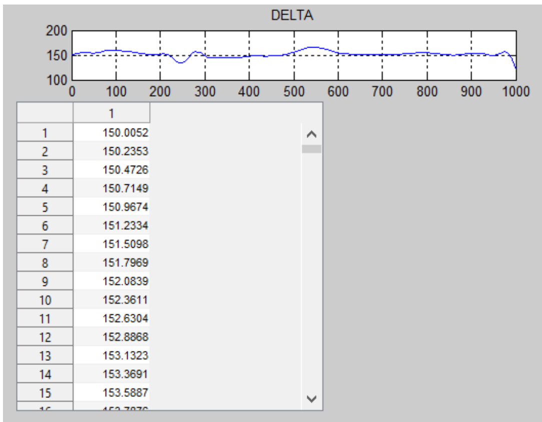


Figure 14. AD DELTA

The raw EEG shows that high frequency Gamma activity is essentially noise-like with limited amplitude, while low frequency Delta activity is predominant. The intensity data of the various bands demonstrate that, for AD patients, the intensity of the delta and theta bands increases, the intensity of the alpha and beta bands diminishes, and the intensity of the gamma band declines as compared to NC.

PCA is a reliable method for dimensionality reduction and feature extraction. The objective is to use PCA to represent the d-dimensional data in a lower-dimensional space. To accurately reflect the variance in terms of sum-squared error, the data must be represented in a space. It greatly helps if we know in advance the number of independent components, like with conventional clustering approaches. Theoretically, the main components approach is quite straightforward. While retaining as much data as feasible, it decreases the number of variables in a data set. It primarily carries out the five operations outlined and detailed below.

- Standardize the range of continuous starting variables.With this step, it is possible to make sure that every continuous initial variable contributes equally to the analysis.
    - [M,N] = size(data);
    - mn = mean(data,2);
    - data = data - repmat(mn,1,N);
- Make a covariance matrix calculation to find correlations.

  covariance = 1 / (N-1) * data * data';

- To find the major components, eigenvalues and eigenvectors for the covariance matrix is determined.

  [PC, V] = eig(covariance);

- To determine which primary components should be retained, create a feature vector.
    - [junk, rindices] = sort(-1*V);
    - V = V(rindices);
    - PC = PC(:,rindices);
- Data should be recast along the primary component axes.

  signals = PC' *data;

The PCA outputs are shown in Fig 15.

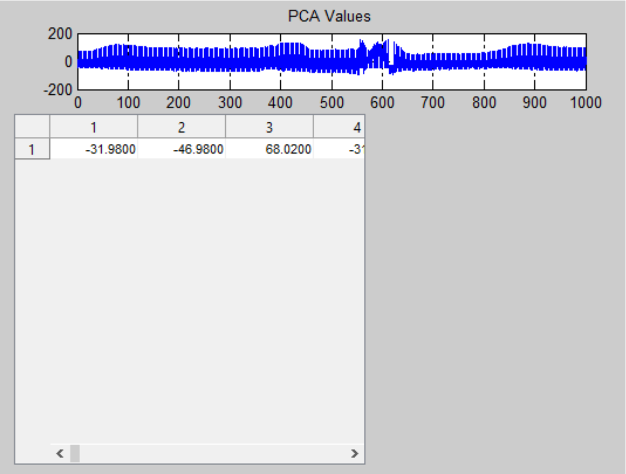


Figure 15. Features Extracted by PCA

Independent component analysis (ICA) is a feature extraction technique that converts a multivariate random signal into a signal with independently variable components. It is possible to extract independent components from the mixed signals using this technique. As a result, independence means that one cannot infer information from that carried by another component. This means that, in terms of statistics, the total probability of the independent quantities is the product of the probabilities of each independent item. The Fig 16 shows the output of ICA.

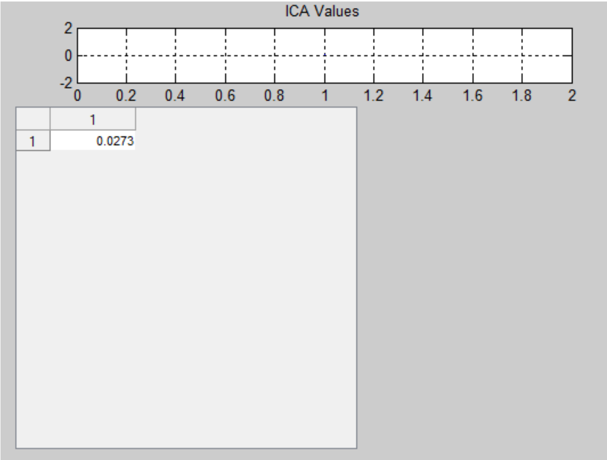


Figure 16. Features Extracted from Input Signal by using ICA

Perform DWT to locate the approximation and detail coefficient vectors after locating PCA and ICA. Applying the DWT function to the original signal with the chosen wavelet accomplishes this. It appears in the equation.

[CA1, CD1] = dwt(Orig_Sig,'db8');

The equation gives the vector x's DWT at the single-level level using the wavelet supplied by wname-db1. The DWT's detail coefficients vector CD1 and approximation coefficients vector CA1 are returned by the dwt command. The detail coefficient and approximation coefficients of the original signal and DWT signal are displayed in the Fig 17.

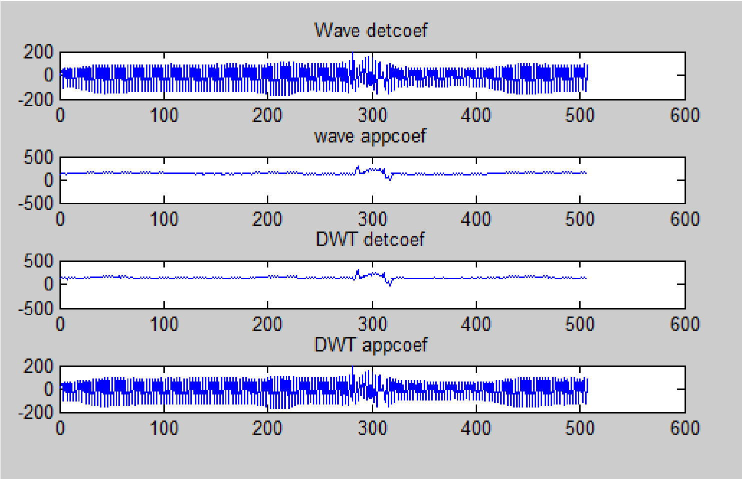


Figure 17. Detail and Approximation Coefficients of Original and DWT Waves Respectively

Finally, test features are created using a combination of DWT detail coefficients, independent coefficients, and PCA features. The combined test features are shown in Fig 18.

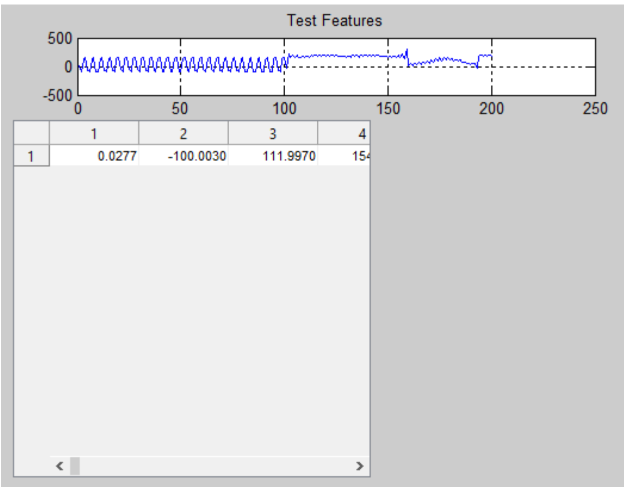


Figure 18. Combined Test Features

2.4 Classification Model

SVM: is a discriminative classifier with a separate hyperplane. This hyperplane, divides the plane into two portions in two-dimensional space, places one class on either side of the line. Numerous benefits come with the support vector classifier. Standard optimization software can be used to locate a distinct global optimum for its parameters. There is a little more computing work required to use nonlinear bounds. And compared to other ways, its performance is really strong. The fact that the difficulty of the problem is inversely correlated with the number of samples, rather than the size of the samples, is a disadvantage.

MLP: The model is also used to categorize the input signals as AD or normal. The MLP has an input layer with 50 neurons, two hidden layers with 256 neurons, and because of binary classification, an output layer with 2 neurons is used. The Leakyrelu activation function activates the hidden layer neurons, and softmax activation function activates the output neurons. Adam Optimizer uses a crossentropy loss function to optimize the MLP model. Fig 19 displays the layer summary.

```
Network Type: MLP
Loss: crossentropy
Momentum: adam
Regularization: L2
Learning Rate: 0.000500
Dropout Rate (0=none): p=1.000000
Trainable: 1
-------Network Architecture--------
Input Layer:     50 Neurons
Layer 2:     256 Neurons     leakyrelu Activation
Layer 3:     256 Neurons     leakyrelu Activation
Layer 4:     1 Neurons   softmax Activation
```

Figure 19. MLP Layer Summary

KNN: One of the simplest and most popular classification algorithms is the KNN, which classifies new data points according to how similar they are to their nearby neighbors. The outcome is competitive. The algorithms determine the distances between a specific data point in the collection and all other K numbers of data points nearby before voting for the category with the highest frequency for that particular data point. The standard unit of measurement for distance is the Euclidean distance. Thus, the final model is essentially the labeled data arranged spatially. Numerous applications, including genetics and forecasting, make use of this approach. The algorithm performs best when there are more features, and in this situation, it outperforms SVM. In the proposed work, KNN also shows better result by giving an accuracy of 100%.

3. Results

To complete the proposed task, Matlab R2015b was used. The interactive environment and high-level technical computing language MATLAB are used to build algorithms, visualize data, analyze data, and perform numerical calculations. Technical computing issues can be resolved more quickly with MATLAB than with more conventional programming languages like C, C++, and Fortran. In this experiment, the EEG signals from two separate classes—Normal and AD—were used. With a ratio of 1:5, the dataset is divided into the training dataset and the test dataset. The training dataset is used to train the SVM, MLP and KNN models. The MLP which go through 40 iterations with a learning rate of 0.00050. After the networks have been trained with test features using the Adam optimizer, the network weight is modified using the category cross entropy function. The parameters considered in this experiment are given table 1.

Table 1. Parameters

Parameters	MLP Model
Optimizer	Adam
Activation Function	leakyReLU and Softmax
Loss function	Categorical cross entropy
Batch size	128
Dataset	EEG
Epoch	40
Learning Rate	0.00050
Normalization	Batch Normalization
Pooling	Maxpooling

For classification, the SVM and KNN are also taken into account. The cross-verification of the training data in this model is tenfolds. Ten subsets are chosen at random from the training set, which includes the label set and data collection. The remaining 1 in 10 holds are randomly chosen as a test to evaluate the use of the sample set, while the remaining 9 samples are used as input in the training sample set process. The classification result obtained for different classifiers (KNN, SVM and MLP) are shown in Fig 20,21,22 respectively.

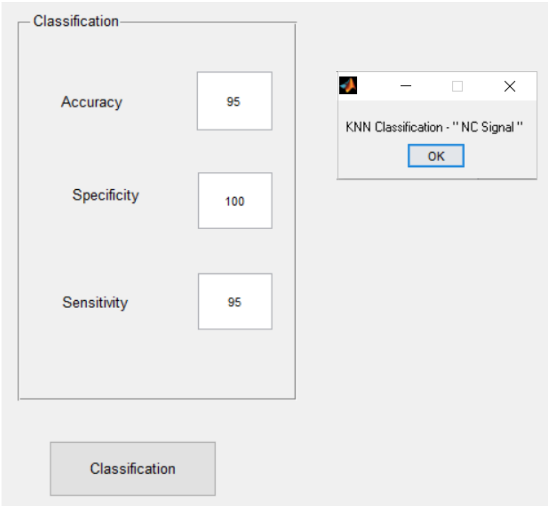


Figure 20. KNN Results

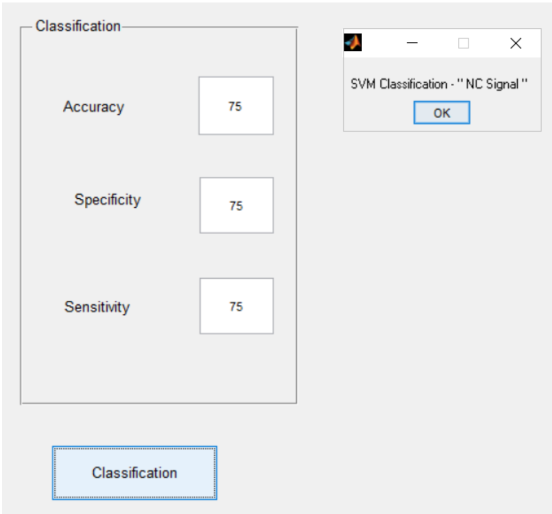


Figure 21. SVM Results

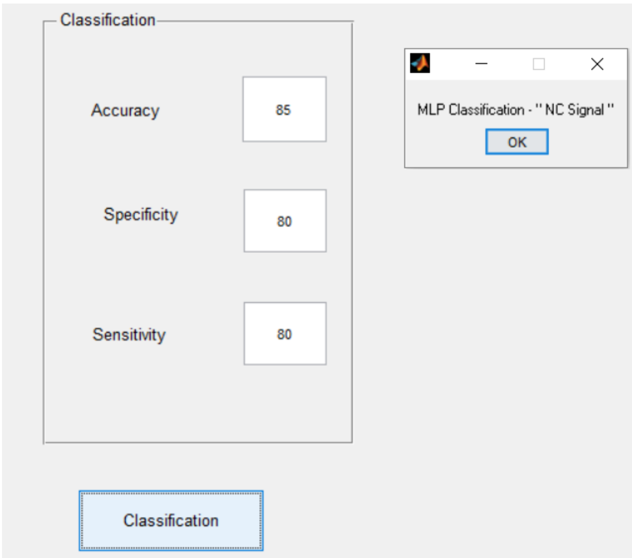


Figure 22. MLP Result

Table. 2 lists each classifier's recorded projected outputs for test signals. From the table, it is noticed that the KNN model accurately predicts all test features correctly, SVM wrongly predicts many test cases, and the MLP does average prediction.

Table 2. Predicted Outputs

Test EEG	**SVM**	**KNN**	**MLP**
1	NC	NC	NC
2	AD	NC	AD
3	AD	NC	NC
4	AD	NC	NC
5	NC	NC	NC
6	NC	NC	NC
7	AD	NC	NC
8	NC	NC	NC
9	AD	NC	NC
10	AD	NC	NC
26	NC	AD	AD
27	NC	AD	AD
28	AD	AD	AD
29	NC	AD	NC
30	AD	AD	AD
31	AD	AD	AD
32	AD	AD	AD
33	NC	AD	AD
34	AD	AD	NC
35	NC	AD	NC

ISSN: 2582-3825

The Confusion matrix is used to evaluate the effectiveness of the system. The SVM, KNN, and MLP confusion matrices are displayed in Tables 8.3, 8.4, and 8.5, respectively. The KNN, which had a 95% accuracy rate, could effectively anticipate every test involving an EEG signal. While the SVM performs poorly with an accuracy of 75%, the MLP displays an average accuracy of 85%. The SVM confusion matrix indicates the frequent in-correct prediction of the test features.

Table 3. KNN Confusion Matrix

Classes	AD	NC	Accuracy	Sensitivity	Specificity
AD	10	0	100%	100%	100%
NC	0	10	100%	100%	100%
Average			100%	100%	100%

Table 4. SVM Confusion Matrix

Classes	AD	NC	Accuracy	Sensitivity	Specificity
AD	5	5	45%	50%	40%
NC	6	4	45%	40%	50%
Average			45%	45%	45%

Table 5. MLP Confusion Matrix

Classes	AD	NC	Accuracy	Sensitivity	Specificity
AD	8	2	85%	80%	90%
NC	1	9	85%	90%	80%
Average			85%	80%	80%

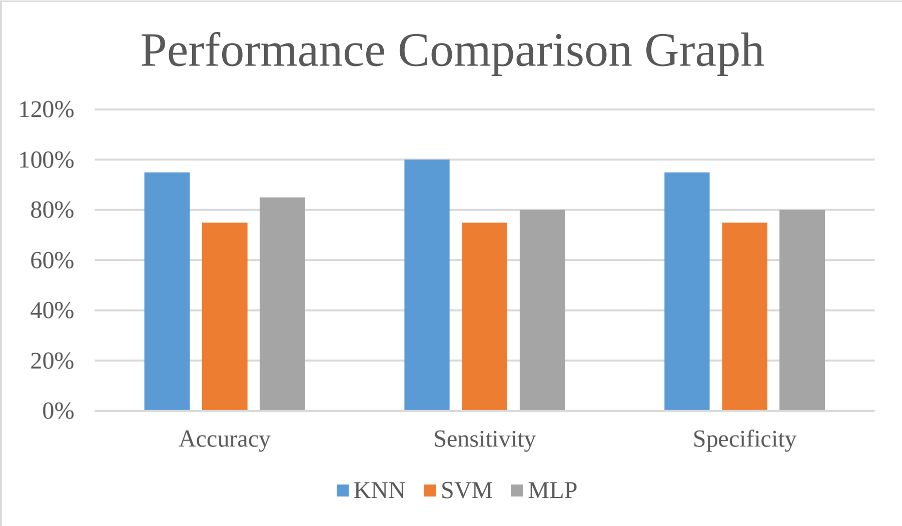


Figure 23. Performance Comparison Graph

Fig. 23 depicts the performance of MLP, SVM, and KNN. The comparison demonstrates that KNN performs at the highest level by providing 95% accuracy, 100% sensitivity, and 95% specificity. According to the comparison graph, the SVM performs the least, scoring only 75% of accuracy, sensitivity, and specificity.

4. Conclusion

Since AD requires different therapies, KNN, which classifies people as having or not having an AD, offers medical experts, who are treating a suspected epilepsy with a useful diagnostic decision-support tool. The PCA and ICA, were utilized with SVM, KNN' and MLP for extracting the features and their accuracy in predicting the observed AD/NC patterns. The extracted features were further cross-compared with the DWT sub-band statistical features of EEG data. Three scalar performance measures—specificity, and sensitivity, and accuracy—that were developed from the confusion matrices served as the basis for the comparisons. This finding suggests that nonlinear feature extraction using KNN can be a viable future replacement for intelligent diagnosis systems. This investigation was led on Intel Pentium Matlab R2015b, with 8GB of RAM, and 64bit Operating framework.

References

[1] Al-Qerem, Ahmad, Faten Kharbat, Shadi Nashwan, Staish Ashraf, and Khairi Blaou. "General model for best feature extraction of EEG using discrete wavelet transform wavelet family and differential evolution." International Journal of Distributed Sensor Networks 16, no. 3 (2020): 1550147720911009..

[2] . Buzzell, George A., Yanbin Niu, Selin Aviyente, and Edward Bernat. "A practical introduction to EEG time-frequency principal components analysis (TF-PCA)." *Developmental Cognitive Neuroscience* 55 (2022): 101114.

[3] Placidi, Giuseppe, Luigi Cinque, and Matteo Polsinelli. "A fast and scalable framework for automated artifact recognition from EEG signals represented in scalp topographies of independent components." Computers in Biology and Medicine 132 (2021): 104347.

[4] https://ars.els-cdn.com/content/image/1-s2.0-S1746809420303530-gr1_lrg.jpg [Online Resource Downloaded on 21st January 2023]

[5] Oltu, Burcu, Mehmet Feyzi Akşahin, and Seda Kibaroğlu. "A novel electroencephalography based approach for Alzheimer’s disease and mild cognitive impairment detection." Biomedical Signal Processing and Control 63 (2021): 102223.

[6] Woon, W. L., A. Cichocki, F. Vialatte, and T. Musha. "Techniques for early detection of Alzheimer's disease using spontaneous EEG recordings." Physiological Measurement 28, no. 4 (2007): 335.

[7] Melissant, Co, Alexander Ypma, Edward EE Frietman, and Cornelis J. Stam. "A method for detection of Alzheimer's disease using ICA-enhanced EEG measurements." Artificial Intelligence in Medicine 33, no. 3 (2005): 209-222.

[8] . Perez-Valero, Eduardo, Miguel A. Lopez-Gordo, Christian Morillas, Francisco Pelayo, and Miguel A. Vaquero-Blasco. "A review of automated techniques for assisting the early detection of Alzheimer’s disease with a focus on EEG." Journal of Alzheimer's disease 80, no. 4 (2021): 1363-1376.

[9] .Jeong, Jaeseung. "EEG dynamics in patients with Alzheimer's disease." Clinical neurophysiology 115, no. 7 (2004): 1490-1505.

[10] Kim, D., & Kim, K. (2018, July). Detection of early stage Alzheimer’s disease using EEG relative power with deep neural network. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 352-355). IEEE.

[11] Dogan, S., Baygin, M., Tasci, B., Loh, H. W., Barua, P. D., Tuncer, T., ... & Acharya, U. R. (2023). Primate brain pattern-based automated Alzheimer's disease detection model using EEG signals. *Cognitive Neurodynamics*, *17*(3), 647-659.

[12] Rodrigues, Pedro M., Bruno C. Bispo, Carolina Garrett, Dílio Alves, João P. Teixeira, and Diamantino Freitas. "Lacsogram: A new EEG tool to diagnose Alzheimer's disease." IEEE Journal of Biomedical and Health Informatics 25, no. 9 (2021): 3384-3395.

[13] Bennys, Karim, Gérard Rondouin, Christine Vergnes, and Jacques Touchon. "Diagnostic value of quantitative EEG in Alzheimer’s disease." Neurophysiologie Clinique/Clinical Neurophysiology 31, no. 3 (2001): 153-160.