

Optimized k-nearest neighbours classifier based prediction of epileptic seizures

Himayavardhini Jagath Prasad¹, Roji Marjorie S.²

¹Department of Electronics and Communication Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (SIMATS), Chennai, India

²Department of Electronics and Communication Engineering, SRM Institute of Science and Technology (SRMIST), Kattankulathur, Chennai, India

Article Info

Article history:

Received Apr 28, 2023

Revised Feb 17, 2024

Accepted Feb 23, 2024

Keywords:

Decision tree classifier

Genetic algorithm

K-nearestneighbours

Random forest classifier

Support vector machine

ABSTRACT

Epileptic seizure is an unstable condition of the brain that cause severe mental disorder and can be fatal if not properly diagnosed at an early stage. Electroencephalogram (EEG) plays a major role in early diagnosis of epileptic seizures. The volume of medical databases is enormous. Classification may become less accurate if the dataset contains redundant and irrelevant attributes. To reduce the mortality rate due to epilepsy, a decision support system that can assist medical professionals in taking immediate precautionary measures prior to reaching the critical condition is required. In this work, k-nearest neighbours (KNN) classifier algorithm is optimised using genetic algorithm for effective classification and faster prediction to meet this requirement. Genetic algorithms search for optimal solutions in complex and large environments. Results are compared with other machine learning models such as support vector machine (SVM), KNN, decision tree classifier, and random forest. With optimization using genetic algorithm KNN was able to achieve an enhancement in accuracy at lower training and testing times. It was observed that the accuracy offered by optimized KNN was 92%. Random forest classifiers showed minimum complexity and KNN algorithm provided faster performance with better accuracy.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Roji Marjorie S.

Department of Electronics and Communication Engineering

SRM Institute of Science and Technology (SRMIST)

Chennai, Tamil Nadu, India

Email: rojiopp@gmail.com

1. INTRODUCTION

The body's response to any kind of demand or threat is stress. Prolonged stress may trigger problems like epilepsy, seizures, depression, obesity, heart diseases, diabetes, headaches, high blood pressure and cholesterol, anxiety, asthma [1]. If left untreated, it may also become fatal which can cause premature deaths.

The major cause for seizure is not yet fully explored [2]. They may also be caused genetically [3]. There are many other potential causes for seizures like cancer, sudden withdrawal of medicines, organ failures, hypertensive encephalopathy (it is a change in the mental state of a person due to sudden increase in blood pressure) [4]. Seizures are a sudden storm of electrical impulses in the brain which may be due to change in the brain's electrical activity. Often, it leads to convulsion which is a state when a person has uncontrollable, rhythmic, repeated contraction and relaxation of muscles. Though there are many causes for seizures, it is not an easier task to analyse and determine the cause of occurrence, the time of occurrence or

how often a person gets a seizure. Therefore, to prevent premature/sudden deaths due to seizures, a faster prediction must be done before the patient reaches a critical state.

The brain cells that produce electrical impulses are called neurons. These neurons undergo chemical changes because of the electrical activity that takes place among them. By this electrical activity, one neuron excites the other and thus messages are sent from one brain cell to another. This causes a balance among the cells that excite and stop messages. Seizure occurs when there is an imbalance between the stopping and exciting activity. Long term seizures occur for a longer duration (few milliseconds to more than 5 minutes) which will lead to fatal side effects. So, it is important to monitor long term seizures which last more than five minutes. Several effective models have been developed to detect and monitor seizures.

Classification has been used for the prediction of seizures. Different classification algorithms were developed by researchers to predict seizure of various time durations. The dataset containing seizures was classified using random forest classifier, k-nearest neighbours (KNN), naive bayes, logistic regression, decision tree, random tree, J48, and stochastic gradient descent. The results revealed that random forest classifier had better performance than other classifiers in terms of accuracy [5]. The prediction of onset of seizures is still a challenging task. Classification is a method that predicts a label of a class for a given dataset by which it generates a model. This model is tuned and reconstructed to achieve a higher prediction accuracy than before. Multiple classifiers were implemented for a comparative analysis of performances of random forest classifier, KNN classifier and naive bayes classifier. Enhancement in accuracy was achieved with KNN but it was slow with training time of 4.789 sec [6]. A classification algorithm requires a training dataset that consists of samples of input and target labels. The classification algorithm learns these samples to build a prediction model [7], [8]. A random forest classifier was implemented to learn the most prominent time domain, frequency domain entropy features. It was observed that it outperformed the conventional methods with better performance. To predict the interictal and ictal intervals, discrete wavelet transform and information entropy theory were used to extract and select the features which are then fed into support vector machines (LS-SVM), KNN, logistic regression, linear discriminant analysis, naive bayes classifier, and random forest. LS-SVM showed a superior performance.

Initially, classification algorithms were used for other practical applications. Poverty data of a village was extracted from a system called community based monitoring system (CBMS) and that data was analysed using data mining [9], [10]. Naïve Bayes, ID3, decision tree, logistic regression, and KNN are different methods of data mining classification that were implemented and compared. Neural network, KNN, and Naïve Bayes were chosen and learning curves were analysed by testing the models with three behaviours such as: overfitting, perfect case and underfitting to determine which learning rate occurs during the training process [11]. A classification was done involving a generalised structure that is to be applied to a new set of data for which the class is to be predicted [12]. For such prediction, classifiers like decision tree, neural network, nearest neighbour were combined with Boolean, and fuzzy logic techniques to achieve better results. Classification helped in monitoring of e-examinations considering performance and accuracy which helped in the detection of malpractices in electronic-examinations. The comparative analysis was done based on face matching and verification using our classification algorithms [13]. This shows that the classification algorithms have found their applications in different fields.

The primary goal of this research is to predict the seizure at a faster rate compared to other techniques. Faster predictions can be done if classification is done faster. The literature related to this work has been compared and their shortcomings have been mentioned in section 2. A detailed description of the dataset, methods and tools used in this work have been discussed in section 3. The comparison and interpretation of the results related to the implementations done have been addressed in section 4 and concluded in section 5.

2. RELATED WORK

This section reviews the different classification models in machine learning and the way in which they have been used for classification and prediction. Classification has also been applied to classify images used for real-time applications. A student monitoring system was developed [14]. The image of the class is captured and saved for further classification by SQLite3. The entire process was carried out in the OpenCV library. Highly robust security systems were developed using classification [15]. Design was created with the help of modelling tools like unified modelling language (UML), flowcharts, and entity relationship (E-R) diagrams. The design was then incorporated with the help of platforms like hyper text markup language (HTML), hypertext preprocessor (PHP), my structured query language (MySQL), and JavaScript. Taxonomy of features were considered to achieve better classification accuracy. Seizure detection was carried out by revealing relevant patterns. Windows, Apache, MySQL, and PHP (WAMP) was used to test the system and performed effectively in minimizing the authenticity and confidentiality issue of exam questions [16]. SVM was implemented to identify the difference between epochs containing seizure and epochs not containing

seizures. Pre-processing and segmentation are done on the signals from individual EEG channels. The signals are then divided into separate epochs. Each epoch extracts a set of features. SVM classifier considers feature vectors as input and computes the likelihood of a seizure for each EEG epoch and converts it to a binary decision [17].

In a new approach, seizure detection was done considering the heart rate of the patient. The personalised method achieved a sensitivity of 71% with false detections of 1.9 per hour. This represents a 37% decrease in false detection rate on average [18]. A weighted KNN classifier employing Bray Curtis distance (WBCKNN) in which the Fourier transform is first applied and tested using k-fold cross-validation on a public dataset. Results show that the classification problems show improvement in terms of accuracy [19]. Brain waves were recorded and features were extracted using a spectro-temporal transformation of bio signals. Random forest classifier was applied to those bio signals to achieve better performance but the process was slow [20].

Machine learning and cloud computing were collaborated with existing communication technologies to develop a system with layered architecture that was used in early detection of epileptic seizures [21]. Histograms were applied to the recorded brain images and the derived feature vectors were applied to random forest algorithm which gave better results and helped in on-time detection of the disease. Prediction using brain images as input does not help in detection at an early stage [22]. Different types of seizures were classified in a work where it was found that the input dataset had imbalance and it also had an un-uniform distribution. The random forest classifier outperformed KNN, naive bayes, logistic regression, decision tree, random tree, J48, and stochastic gradient descent with enhancement in accuracy [23].

Lung cancer related CT images were given as input to KNN, SVM, stochastic gradient descent classifier, random forest classifier, decision tree classifier, multinomial Naïve Bayes classifier, and multi-layer perceptron. Multilayer perceptron (MLP) gave better results with good robustness, but early detection was not possible because of image input [24]. A combination of machine learning classifiers and convolution neural networks (CNN) with Butterworth filter was used to pre-process the EEG signal and CNN to extract features. To reduce the complexity and improve classification accuracy, only the relevant features were selected [25]. A personalization method was developed, in which there is improvement in the quality of seizure detection by personalising the preictal data using a work done [26]. Analysing intracranial EEG time series. These estimates are tested using KNN to classify the intervals of seizures correctly. With a large dataset, KNN might require huge memory to store all the data and become computationally slow and expensive [27].

Many contributors have contributed many findings to this area with enhancement in parameters like accuracy, specificity, precision, and recall. Hilbert probability similarity and PSO were used as an effective tool for classifying EEG signals after which precision, recall, and F1 score were calculated with better accuracy [28]. Machine learning techniques like KNN, SVM, ELM, and deep learning technique called long short-term memory (LSTM) were used to differentiate poor signals of dyslexic children. SVM achieved better performance [29]. A real-time EEG compression technique was proposed with the help of set partitioning in hierarchical trees (SPIHT). Compression and detection rates were better using the proposed algorithm [30]. An automated channel selection method was proposed to achieve a better false positive rate [31].

All the aforementioned works elaborate the role of classifiers in diagnosing epilepsy. It is seen that although the classifiers are widely used in the detection of seizures, their performances depend on the dataset. For large datasets, the time and computational complexity increases. This work aims at achieving highly efficient models in predicting seizures. Machine learning algorithms were implemented and the one that gives better results is taken into consideration. Random forest classifiers can handle large datasets and produce better results with minimal complexity.

3. MATERIALS AND METHOD

The existing models for classification perform based on the type of data. They require huge data which in turn demands huge memory to store all the trained data. The time required to classify all the data points was also increased. This system is aimed at diagnosing seizures using an Automatic decision-making algorithm. Accurate and timely processing is carried out with remote monitoring.

Figure 1 shows the block diagram that represents the workflow. EEG dataset was given as input to the classifier. It trains and tests the dataset to produce prediction results that tell us how well our classifier worked. The machine learning algorithms used in the analysis were SVM, KNN, decision tree, and random forest.

3.1. Dataset

Each of the five folders that make up the original dataset [32] contains one hundred files, each of which represents a single subject or person. Every file contains a 23.6 second recording of brain activity. Data

points totaling 4097 are sampled from the corresponding time-series. The value of the EEG recording at a particular moment in time is represented by each data point. There are 500 people in total, and each of them has 4097 data points for 23.5 seconds. A total of 11500 data points is present in each piece of information, with each data point lasting one second. Table 1 shows the different classes in the dataset and the conditions related to each class of the dataset. Table 2 depicts the description of the signals recorded under each class.

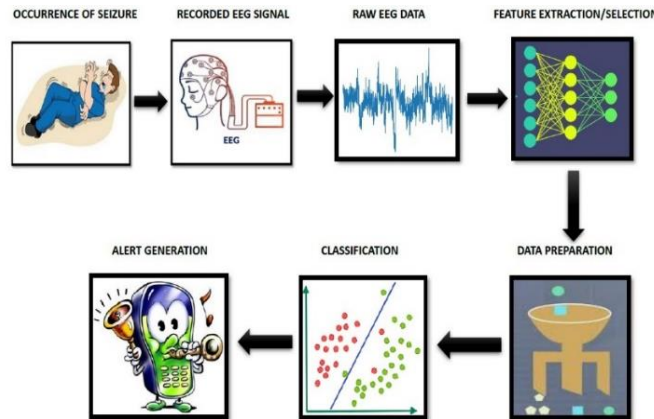


Figure 1. Basic model of epilepsy seizure detection

Table 1. EEG dataset

Class	Samples	Duration of each epoch (s)	Segment length	State of patient	Condition
5	100	23.6	4097	Eye open (normal)	Non-epileptic
4	100	23.6	4097	Eye close (normal)	Non-epileptic
3	100	23.6	4097	Seizure free	Non-epileptic
2	100	23.6	4097	Epileptic zone	Non-epileptic
1	100	23.6	4097	Seizure activity	Epileptic

Table 2. Dataset description

Class	Description of each class
5	EEG signals of the patient recorded when eyes open
4	EEG signals of the patient recorded when eyes closed
3	EEG activity recorded from seizure free zone of the brain
2	EEG activity recorded from epileptic zone of the brain
1	Seizure activity recorded

3.2. Support vector machine-based seizure prediction

The kernel, gamma, and penalty parameters were used to train the SVM. The prediction and classification process took longer to finish when SVM was implemented. Figure 2 displays the SVM's flowchart. Pre-processing functions are used to normalise the data set. Subsequently, the SVM is trained with the assigned gamma values and penalty parameter C. Every algorithmic step performs cross validation using the corresponding gamma and C. The process of testing and training is then completed. An option is selected and saved considering the findings. If more execution is needed, the C and gamma values are updated in the following step.

For the updated C and gamma values, the training and testing procedures are still ongoing. Until the ideal answer is found, these procedures are repeated. The average success rate is used to determine the values of C and gamma, and the parameters are then used to repeat the second step. SVM cannot be applied to datasets with a higher number of features due to their limitations in certain scenarios. It can manage large amounts of data, but not an excessive number of features with significant overlap. This is since it can only make the classes into a hyperplane. By drawing a border between the classes, this hyperplane prevents overlapping data from being classified. Thus, it was applied in minor ways that supported clinical assistance.

3.3. K-nearest neighbours based seizure prediction

For classification, every data point in the dataset is considered. The number of closest neighbours, denoted by the parameter "K" is the basis for the classification process. A newly entered data point is classified according to the distance measured between it and its closest neighbours, which are simply sets that have already been classified. Figure 3 displays the KNN classifier flowchart. Every input from the data set is

first considered. Whichever classification needs to be done determines which subset of the data set should be used. The number of neighbours that should be assigned to a new data point for classification is provided by the parameter "K". The new data point's distance from its neighbours is then determined.

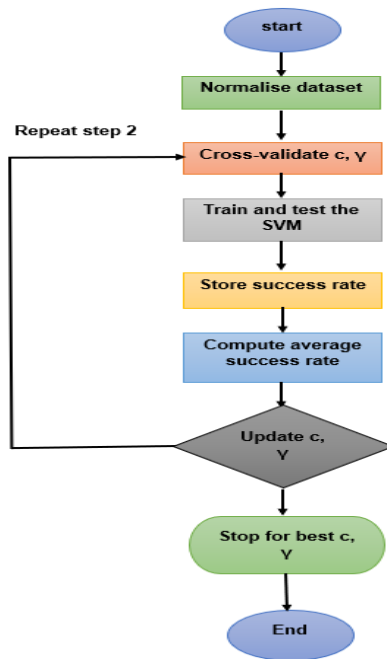


Figure 2. Flowchart of SVM

The new data point is sorted into the class that the neighbour with the shortest distance belongs to after considering the distance that is the smallest. Until every data point in the datasets is classified, this process is repeated. The process can be restarted from the first step and the subset updated for a fresh set of classified points.

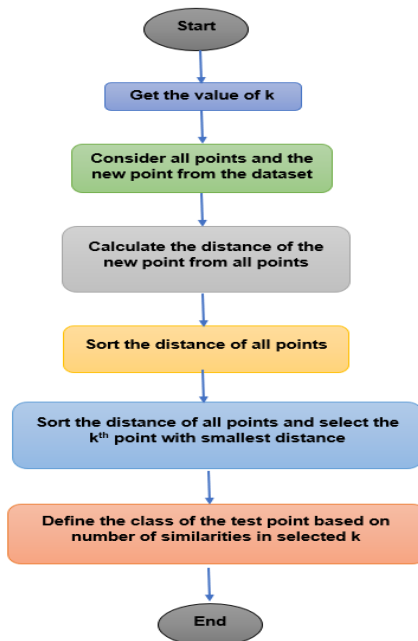


Figure 3. Flowchart of KNN classifier

3.4. Decision tree based seizure prediction

The biomedical field faces two main difficulties. Data acquisition is the first step, and data saving is the second. Any recorded medical information must be stored and kept private. At all costs, the patient's identity should remain confidential. Since the data must be used to record the medical history of the patients, it should not be tampered with or erased. As a result, handling data initially becomes more difficult, and processing data becomes much more time-consuming. Numerous methods were found to operate input data in the biomedical domain effectively. Decision trees are a useful tool for dataset analysis. A spot system that can store recorded medical data and construct a patient's medical history from that saved data has always been in demand. A minor misclassification can result in serious medical errors, so the system's data sorting ability must be strong. The medical community needs a long-term solution to cut down on the long-standing medical errors. The model must have a clear idea of how to decide on the defined problem while taking the input's features and parameters into account. The use of decision trees facilitates decision making by providing a visual aid.

The classifier that uses decision trees is tree-based. A root node, internal branch nodes, and leaf nodes make up the tree, which is a hierarchical algorithm. The nodes that appear first in this instance are the root and leaf nodes, respectively. To decide, the internal nodes-referred to as the decision nodes-are divided further. First, the root node is where the process begins. There are no incoming branches at the root node. The internal nodes, also known as the decision nodes, receive the outgoing branches from the root nodes. To reach a decision, the decision node divides the data into subsets and applies the necessary operations to each subset. There is no more splitting in the decision node where it becomes the terminal node or leaf node for that specific path if the desired result has been reached. The next subset produced by the subsequent decision node is used in the process and so forth. Further splitting occurs in the decision node if the intended result is not reached, and the process is carried out again until a solution is found. The decision tree's depth is determined by the split levels on the decision nodes.

The flowchart of the decision tree classifier is shown in Figure 4. The data set's numerical values are all arranged in ascending order in the first step. The algorithm then determines a threshold value. This threshold value aids in the division of nodes into several paths for the classification of the data. The threshold value is compared with each value of X from the input. The input's initial value is found in the root node. This node divides into two branches: one for X values below the threshold and another for X values above the threshold, which determines how the two internal nodes receive the sorted X values. The two internal nodes that have already been formed have further split in the following step, which also fixes another threshold. A similar split occurs and the newly created internal nodes in this step are assigned values that are either less than or greater than the new threshold. Each time X in the data set has a value, this process is repeated. By computing metrics like information gain entropy and Gini impurity, the optimal split is found.

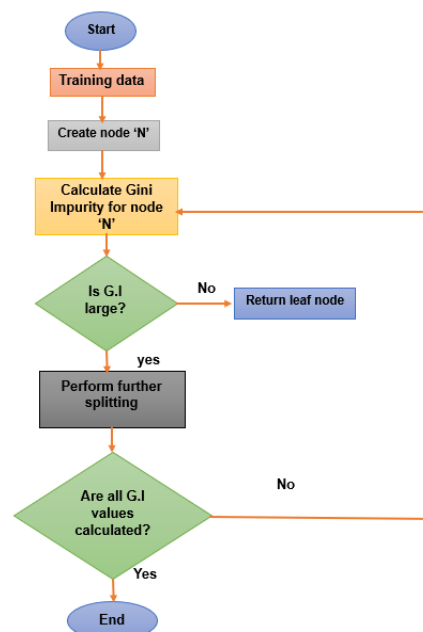


Figure 4. Flowchart of decision tree

Gini impurity is taken into account in this work and the optimal split is thought to be the one that yields the lowest value of Gini impurity. Every decision node created for every threshold value goes through this process again. The Gini impurity is a metric that quantifies how a dataset's features should split nodes to form a tree. It shows the degree of misclassification that results from randomly assigning a new data point to a class label. Typically, a gini impurity falls between zero and 0.5. Gini impurity is defined as:

$$Gini(D) = 1 - \sum_{i=1}^k P_i^2 \quad (1)$$

The dataset "D" in (1) contains "k" samples total. The probability that input samples at a specific root node belong to a given class 'i' is denoted by 'P_i'. In the case of a uniform class distribution, a node exhibits higher impurity. When every input sample is in the same class and the misclassification rate is at its lowest, minimum impurity is reached.

3.5. Random forest-based seizure prediction

Several trees make up the useful classification method known as random forest, which is an ensemble technique. Its foundation is the idea of ensemble learning, which combines several classifier models to finish the classification procedure. The input data set is divided into various subsets. There is a distinct tree with root node, branches, and decision nodes for every subset of data. The goal of solving each tree is to get a result that nearly matches the best answer. All the trees' results are compared. Every result that reaches a tree generates a decision. A majority voting system is used to cast votes for each decision. The best prediction outcome is thought to be the choice that results from a majority vote. The algorithm will be more robust the more trees there are. Moreover, the algorithm's accuracy increases as the number of trees increases. By averaging the predictive accuracies derived from each tree, this improvement is accomplished. This has allowed the random forest model to be designed with a higher capacity for problem-solving.

The flowchart of random forest classifier is shown in Figure 5. The number of trees that must be built in the random forest is determined in the first step. The size of the data set influences the choice of this parameter. The value of the number of estimators increases with the size of the data set. The trees are built using the decision tree algorithm. Depending on the result, each tree receives one vote. The prediction accuracy of each tree is averaged to determine which votes receive the most weight. This provides the ultimate prediction result, from which a choice is made and the accuracy is ultimately determined.

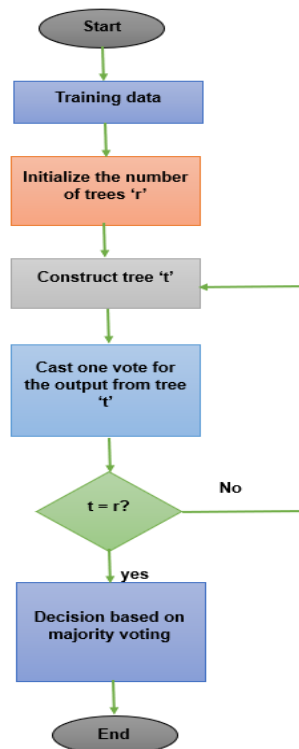


Figure 5. Flowchart of random forest classifier

3.6. Optimized k-nearest neighbours based seizure prediction

Implementing KNN for classification of seizures gave better training and testing times with moderate accuracy. The primary goal of this work is to predict the occurrence of seizures well before its onset so that immediate medical attention can be provided before reaching the critical state. This type of monitoring can be of great help to elderly patients and patients in remote locations. With faster training and testing times, KNN is incorporated with genetic algorithm to achieve better accuracy. Genetic algorithm can perform search in vast environments.

Figure 6 shows the flow in which the optimized KNN works. Initially, the parameters required for the process like fitness function and number of neighbors are defined. An initial population is created to implement the algorithm. The algorithm is then implemented with the defined parameters to get the best individual. With the obtained hyperparameters, the KNN algorithm is implemented. The model is evaluated to get the training time, testing time and accuracy. It was found that an enhancement in accuracy was achieved at lower training and testing times. If enhancement is not considerable, the algorithm can be evaluated with the individuals created in the next generation.

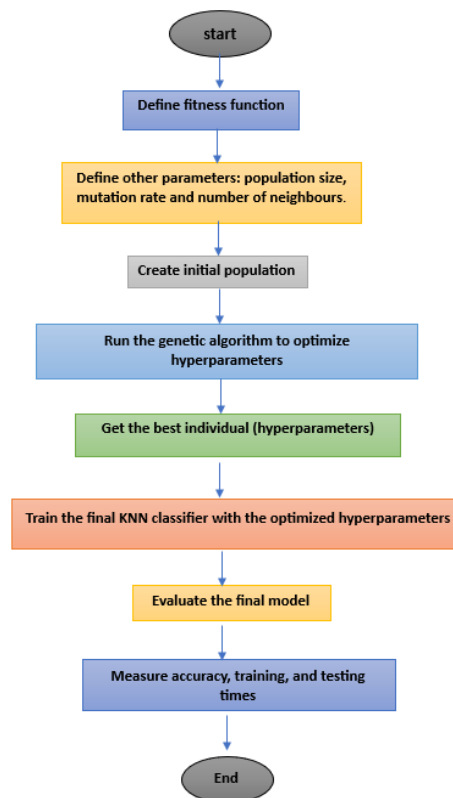


Figure 6. Flowchart of optimized KNN

3.7. Tuning of hyperparameters for seizure prediction

All machine learning models are tuned using data-driven parameters. Model parameters can be fit by training a model with existing data. Hyperparameters are a type of parameter that cannot be learned through routine training. They are usually fixed before the programme begins. These parameters describe important model characteristics such as complexity and learning rate. Hyper parameters are parameters that are specified in the classifier at the start of the process. The parameters given in Table 3 are tuned with different values to improve the classifier's current performance.

Table 3. General hyperparameters of classification models

Algorithm	Hyperparameters
SVM	Generalization parameter (C), gamma
KNN	Number of neighbor (K)
Decision tree	Maximum depth
Random forest	Number of estimators

4. RESULTS AND DISCUSSION

Python version 3.9.7 has been used for training and testing data with various machine learning algorithms. The algorithms were trained with modules imported from scikit tutorial. The dataset was read using the pandas module. Training and testing set of the dataset are then split using model selection in scikit learn module. For each classifier, parameters were tuned and trained with the split training data. In the given work, 80% of dataset was considered for training and 20% of dataset was considered for testing. Dataset for analysis was obtained from Kaggle repository [32].

In all the experiments conducted, the accuracy is considered as the dependent variable and the penalty parameter (c), maximum depth (d), number of neighbours (k) and number of estimators (n) are considered as independent variables for SVM, decision tree, KNN, and random forest classifiers respectively.

The most important task in building any machine learning model is to evaluate its performance. This is done in order to check the quality of the analysed models. Some metrics have been developed based on the characteristics of true positives (TP), true negatives (TN), false positive (FP), false negative (FN). The following metrics are evaluated to check and compare the performances of the algorithms.

4.1. Accuracy

The ratio of the number of correct predictions and the total number of predictions is called accuracy. The quality of the prediction is checked with the help of accuracy score by telling how often a classifier predicts correctly. In (2) shows accuracy:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

In SVM, the accuracy is the dependent variable and the parameter 'c' is the independent variable. This parameter is called the penalty parameter of the error term. It is considered as the degree of correct classification that the classifier is supposed to achieve. From Figure 7 it is seen that as the value of the penalty parameter (c) increases the training accuracy increases, reaches a peak at c=2 and then keeps on increasing with the value of c. on the other hand, the testing accuracy increases, reaches a peak at c=2 and has a constant accuracy till c=3. The testing accuracy again reaches a peak at c=4 and decreases for c>4. For c>5, there is a constant increase in the testing accuracy. Though there is an increase in the test accuracy at some instances, it does not produce a better accuracy with the increasing penalty parameter. It is inferred that the trained SVM classifier is an overfitting model.

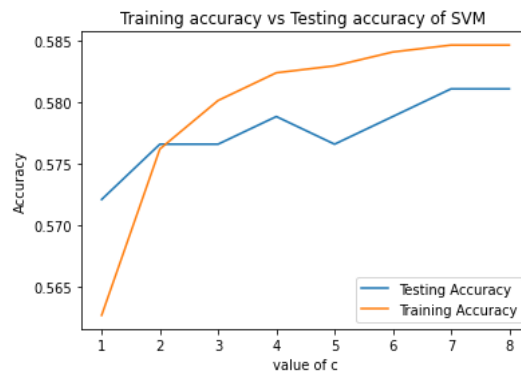


Figure 7. Accuracy plot for SVM

In KNN, the accuracy is the dependent variable, and the value of 'k' is the independent variable 'k' is nothing but the number of neighbours that the classifier is supposed to check in order to classify a new data point. From Figure 8 it is seen that training accuracy is drastically decreased with the increasing number of neighbours (k). The testing accuracy increases, reaches a peak at k=3, decreases till k=5, reaches a peak at k=6 again. This continues till the end of the plot. So, the trained KNN classifier is found to be an underfitting model. In decision tree, the accuracy is the dependent variable, and the maximum depth of the tree 'd' is the independent variable. It is considered as the longest part of the decision tree. Figure 9 indicates an increase in training score with increase in the maximum depth of the decision tree till the end of the plot. The testing scores increase as there is increase in the maximum depth of the decision tree. The values keep increasing, reaching a peak at d=3 and then again decreases at d=4. The test accuracy continues to increase till d=5 after

which it has a constant value. For $d > 6$, the testing accuracy gradually decreases. The decision tree classifier produces better accuracy for $d=3$. So, the trained decision tree classifier is found to be a perfect model for classification. For larger values of d , a decrease in accuracy is observed because of the increased complexity. The execution time was also increased automatically for such a complex decision tree.

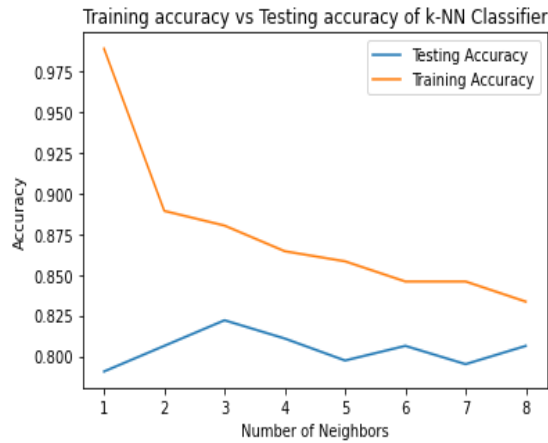


Figure 8. Accuracy plot KNN classifier

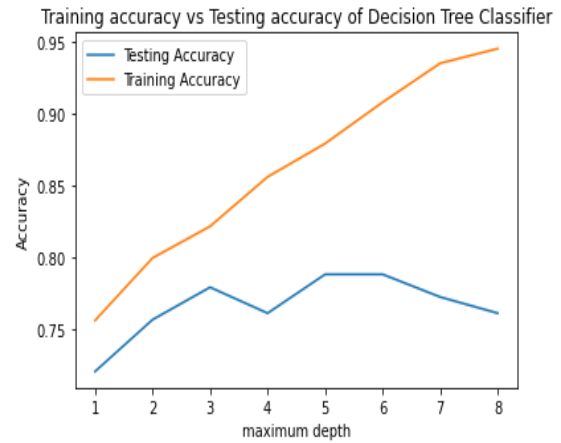


Figure 9. Accuracy plot for decision tree classifier

In random forest, the accuracy is the dependent variable, and the number of estimators ‘n’ is the independent variable which decides the number of trees to be designed in the forest. Figure 10 indicates an increase in training accuracy as there is increase in the number of estimators (n) of the random forest classifier. It keeps increasing after $n=2$ till the end of the plot. The testing accuracy reaches a peak at $n=3$, drops at $n=4$, and again increases for $n > 4$. It keeps increasing till $n=7$ and after $n > 7$ the testing accuracy decreases. So, better accuracy is obtained at $n=7$. The implemented random forest classifier was able to achieve a higher accuracy with a greater number of estimators and the complexity was handled better than the decision tree. So, it was taken into further investigation to reduce complexity.

Figure 11 shows the importance values plotted for each variable of the dataset. A further reduction in complexity was then achieved by incorporating feature importance. It involved selection of features from the dataset based on their calculated importance values. Features with negligible importance were removed and features with higher importance values were selected and the classifier was trained with only those features. By this, the execution time was also reduced, and a better performance was achieved compared to the other classifiers.

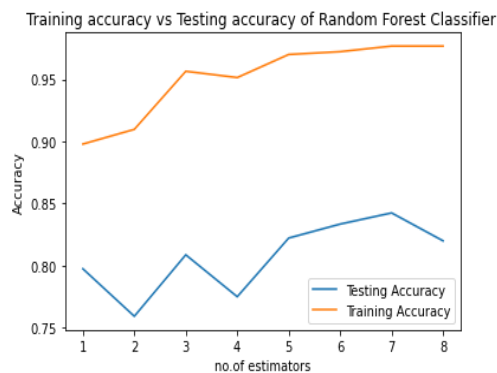


Figure 10. Validation graph with random forest classifier

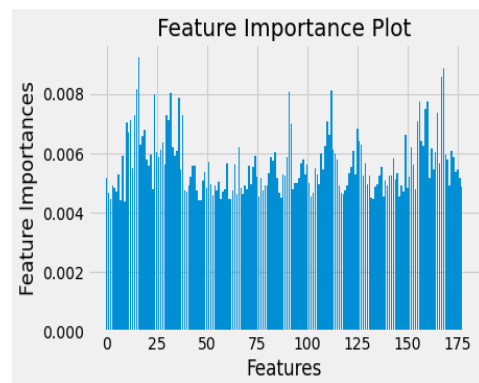


Figure 11. Importance values of variables from the dataset

From Table 4 it is clearly seen that the random forest classifier outperforms all the other models. The training and testing accuracies of decision tree and random forest algorithm were better. SVM gave higher accuracy than decision tree but it was not flexible enough to handle different sets of data. Random forest classifier was able to perform well with high accuracy than other models but resulted in higher

execution time as well. Though random forest classifier handled larger datasets with minimum complexity, it can be further taken into consideration to achieve higher accuracy by reducing the complexity further.

Table 4. Comparison of accuracies of all machine learning models

Algorithm	Training accuracy (%)	Testing accuracy (%)
SVM	97	80.40
KNN	89.94	76.39
Decision tree	97.23	77.97
Random forest	93.10	84.90

4.2. Training and testing times

Training and testing times are metrics which indicate how fast a model is capable of training and testing the datapoints. Table 5 gives a clear picture of the training and testing times of different models used in this work.

Table 5. Comparison of training and testing times of all machine learning models

Algorithm	Training time (s)	Testing time (s)
SVM	3782.5937	27.0282
KNN	0.0140	0.0155
Decision tree	3.2240	1.1933
Random forest	10.7116	4.0265

From Table 5 it is seen that the training and testing times are lowest for KNN and highest for SVM. Decision tree and random forest algorithms provide a moderate training and testing times because of their iterative operation. Table 6 shows the accuracy, training time and testing time of optimized KNN algorithm and it is seen that there is improvement in accuracy compared to the result in Table 4 and the training and testing times are almost the same as seen in Table 5.

Table 6. Accuracy, training and testing time of optimized KNN

Accuracy (%)	Training time (s)	Testing time (s)
92	0.0001	0.0156

Figure 12 shows the training time plot for different machine learning models used in this work. It is seen that the SVM is the slowest model with a training time of 3782.5937 seconds. Decision tree and random forest algorithms provide moderate training time due to their iterative nature. Both KNN and optimized KNN provide the lowest training times of 0.014 seconds and 0.0001 seconds respectively making them the fastest model in training the dataset.

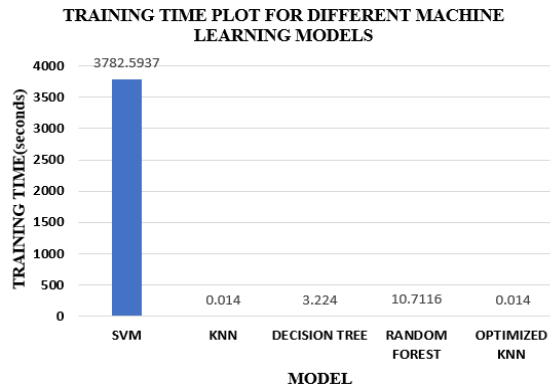


Figure 12. Training time plot for different machine learning algorithms

Figure 13 shows the testing time plot for different machine learning models used in this work. It is seen that the SVM is the slowest model with a training time of 27.0282 seconds. Decision tree and random forest algorithms provide moderate testing times due to their iterative nature. Both KNN and optimized KNN provide the lowest testing times of 0.0155 seconds and 0.0156 seconds respectively making them the fastest model in testing the dataset.

Figure 14 shows the accuracy plot for different machine learning models used in this work. It is seen that the KNN provides the minimum accuracy of 76.39% and optimized KNN provides maximum accuracy of 92%. SVM, decision tree, and random forest algorithms provide moderate accuracies. Optimized KNN provides the better accuracy compared to other algorithms. Table 7 gives a comparison of accuracies of previous works with our work. It is clearly seen that the method used in this work provides better accuracy than the previous other works.

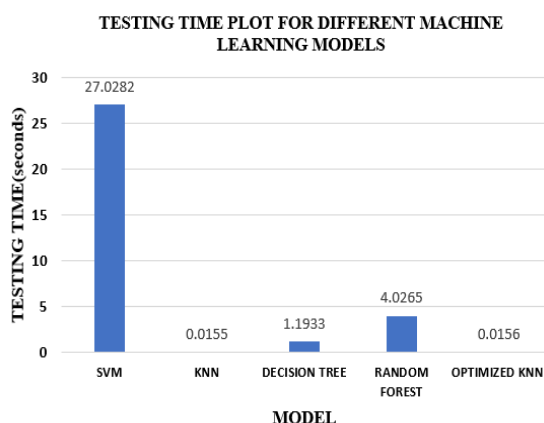


Figure 13. Testing time plot for different machine learning algorithms

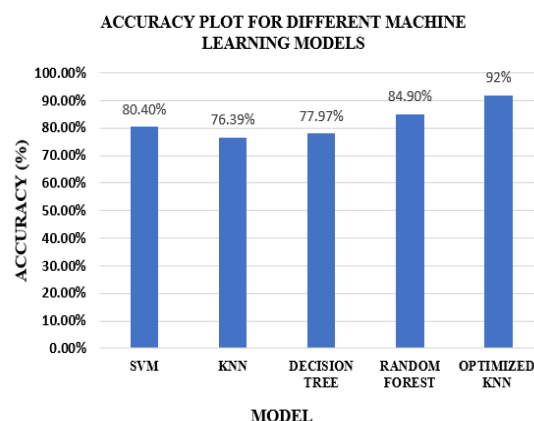


Figure 14. Accuracy plot for different machine learning algorithms

Table 7. Comparison of previous works based on accuracy

Author	Method	Accuracy (%)
Al-Hamzawi <i>et al.</i> [28]	SVM	48.33
Al-Hamzawi <i>et al.</i> [28]	KNN	81.67
Zainuddin <i>et al.</i> [29]	KNN	78.33
Daou and Labeau [30]	Threshold	90
Our work	Optimized KNN	92

From the critical discussions and interpretations made using the above results, it can be concluded that the optimized KNN algorithm performs better in seizure prediction compared to other machine learning algorithms. Accuracy, training and testing times are comparatively better than the other models. Decision tree and random forest models perform moderately which can be improved by enhancing them in terms of accuracy and time.

5. CONCLUSION





In the improved model, optimized KNN was found to be faster with lower training and testing times with enhanced accuracy of 92% compared to other models. On the other hand, decision tree, and random forest classifiers with set parameters were found to perform well for the large dataset but the decision tree showed an increase in complexity. So, these two models will be used in further investigation to achieve even better results with enhancement in accuracy at lower training and testing times. Future investigation is aimed at pruning the decision trees and random forest to achieve minimal time and computational complexity and KNN is considered for even more improvement in accuracy at faster rates.

REFERENCES





[1] C. Espinosa-Garcia, H. Zeleke, and A. Rojas, "Impact of Stress on Epilepsy: Focus on Neuroinflammation—A Mini Review," *International Journal of Molecular Sciences*, vol. 22, no. 8, p. 4061, Apr. 2021, doi: 10.3390/ijms22084061.
 [2] C. E. Stafstrom and L. Carmant, "Seizures and Epilepsy: An Overview for Neuroscientists," *Cold Spring Harbor Perspectives in*

- Medicine*, vol. 5, no. 6, pp. a022426–a022426, Jun. 2015, doi: 10.1101/cshperspect.a022426.
- [3] G. Wang, W. Wu, Y. Xu, Z. Yang, B. Xiao, and L. Long, “Imaging Genetics in Epilepsy: Current Knowledge and New Perspectives,” *Frontiers in Molecular Neuroscience*, vol. 15, May 2022, doi: 10.3389/fnmol.2022.891621.
 - [4] V. T. Cunliffe *et al.*, “Epilepsy research methods update: Understanding the causes of epileptic seizures and identifying new treatments using non-mammalian model organisms,” *Seizure*, vol. 24, pp. 44–51, Jan. 2015, doi: 10.1016/j.seizure.2014.09.018.
 - [5] K. M. Almustafa, “Classification of epileptic seizure dataset using different machine learning algorithms,” *Informatics in Medicine Unlocked*, vol. 21, p. 100444, 2020, doi: 10.1016/j.imu.2020.100444.
 - [6] F. P. Lestari, M. Haekal, R. E. Edison, F. R. Fauzy, S. N. Khotimah, and F. Haryanto, “Epileptic Seizure Detection in EEGs by Using Random Tree Forest, Naïve Bayes and KNN Classification,” *Journal of Physics: Conference Series*, vol. 1505, no. 1, p. 12055, Mar. 2020, doi: 10.1088/1742-6596/1505/1/012055.
 - [7] M. Mursalin, Y. Zhang, Y. Chen, and N. V Chawla, “Automated epileptic seizure detection using improved correlation-based feature selection with random forest classifier,” *Neurocomputing*, vol. 241, pp. 204–214, Jun. 2017, doi: 10.1016/j.neucom.2017.02.053.
 - [8] S. Chen, X. Zhang, L. Chen, and Z. Yang, “Automatic Diagnosis of Epileptic Seizure in Electroencephalography Signals Using Nonlinear Dynamics Features,” *IEEE Access*, vol. 7, pp. 61046–61056, 2019, doi: 10.1109/access.2019.2915610.
 - [9] E. T. R. Baitharu and D. S. K. Pani, “A Comparative Study of Data Mining Classification Techniques using Lung Cancer Data,” *International Journal of Computer Trends and Technology*, vol. 22, no. 2, pp. 91–95, Apr. 2015, doi: 10.14445/22312803/ijctt-v22p118.
 - [10] J. A. Talingdan, “Performance Comparison of Different Classification Algorithms for Household Poverty Classification,” in *2019 4th International Conference on Information Systems Engineering (ICISE)*, May 2019, doi: 10.1109/icise.2019.00010.
 - [11] R. Muhamedyev, K. Yakunin, S. Iskakov, S. Sainova, A. Abdilmanova, and Y. Kuchin, “Comparative analysis of classification algorithms,” in *2015 9th International Conference on Application of Information and Communication Technologies (AICT)*, Oct. 2015, doi: 10.1109/icaict.2015.7338525.
 - [12] M. Alghobiri, “A Comparative Analysis of Classification Algorithms on Diverse Datasets,” *Engineering, Technology & Applied Science Research*, vol. 8, no. 2, pp. 2790–2795, Apr. 2018, doi: 10.48084/etasr.1952.
 - [13] B. T. Adetoba, O. Awodele, O. D. Alao, and V. O. Nwaocha, “Comparative Analysis of Classification Algorithms for Face Matching and Verification in E - Examination,” *International Journal of Innovative Science and Research Technology*, vol. 5, no. 3, pp. 467–473, 2020.
 - [14] A. Krishna and H. Tuli, “Live Class Monitoring Using Machine Learning,” in *Algorithms for Intelligent Systems*, Springer Singapore, 2020, pp. 385–389, doi: 10.1007/978-981-15-0222-4_35.
 - [15] O. Sarjiyus, “Securing Computer Based Testing (CBT) System for Tertiary Institutions in Nigeria,” *Asian Journal of Research in Computer Science*, pp. 1–16, Jul. 2019, doi: 10.9734/ajrcos/2019/v3i330094.
 - [16] M. K. Siddiqui, R. Morales-Menendez, X. Huang, and N. Hussain, “A review of epileptic seizure detection using machine learning classifiers,” *Brain Informatics*, vol. 7, no. 1, May 2020, doi: 10.1186/s40708-020-00105-1.
 - [17] A. Temko, E. Thomas, W. Marnane, G. Lightbody, and G. Boylan, “EEG-based neonatal seizure detection with Support Vector Machines,” *Clinical Neurophysiology*, vol. 122, no. 3, pp. 464–473, Mar. 2011, doi: 10.1016/j.clinph.2010.06.034.
 - [18] T. De Cooman *et al.*, “Personalizing Heart Rate-Based Seizure Detection Using Supervised SVM Transfer Learning,” *Frontiers in Neurology*, vol. 11, Feb. 2020, doi: 10.3389/fneur.2020.00145.
 - [19] Z. Wang, J. Na, and B. Zheng, “An Improved kNN Classifier for Epilepsy Diagnosis,” *IEEE Access*, vol. 8, pp. 100022–100030, 2020, doi: 10.1109/access.2020.2996946.
 - [20] M. A. Pinto-Orellana and F. R. Cerqueira, “Patient-Specific Epilepsy Seizure Detection Using Random Forest Classification over One-Dimension Transformed EEG Data,” in *Intelligent Systems Design and Applications*, Springer International Publishing, 2017, pp. 519–528, doi: 10.1007/978-3-319-53480-0_51.
 - [21] K. Singh and J. Malhotra, “IoT and cloud computing based automatic epileptic seizure detection using HOS features based random forest classification,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 11, pp. 15497–15512, Dec. 2019, doi: 10.1007/s12652-019-01613-7.
 - [22] E. Alickovic and A. Subasi, “Automatic Detection of Alzheimer Disease Based on Histogram and Random Forest,” in *CMBEI 2019*, Springer International Publishing, 2019, pp. 91–96, doi: 10.1007/978-3-030-17971-7_14.
 - [23] A. Basri and M. Arif, “Classification of Seizure Types Using Random Forest Classifier,” *Advances in Science and Technology Research Journal*, vol. 15, no. 3, pp. 167–178, Sep. 2021, doi: 10.12913/22998624/140542.
 - [24] G. A. P. Singh and P. K. Gupta, “Performance analysis of various machine learning-based approaches for detection and classification of lung cancer in humans,” *Neural Computing and Applications*, vol. 31, no. 10, pp. 6863–6877, May 2018, doi: 10.1007/s00521-018-3518-x.
 - [25] A. Subasi, J. Kevric, and M. A. Canbaz, “Epileptic seizure detection using hybrid machine learning methods,” *Neural Computing and Applications*, vol. 31, no. 1, pp. 317–325, Apr. 2017, doi: 10.1007/s00521-017-3003-y.
 - [26] J. Birjandtalab, V. N. Jarmale, M. Nourani, and J. Harvey, “Impact of Personalization on Epileptic Seizure Prediction,” in *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, May 2019, doi: 10.1109/bhi.2019.8834648.
 - [27] S. Lahmiri and A. Shmuel, “Accurate Classification of Seizure and Seizure-Free Intervals of Intracranial EEG Signals From Epileptic Patients,” *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 3, pp. 791–796, Mar. 2019, doi: 10.1109/tim.2018.2855518.
 - [28] A. Al-Hamzawi, D. Al-Shammary, and A. H. Hammadi, “Health Electroencephalogram epileptic classification based on Hilbert probability similarity,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 3, p. 3339, Jun. 2023, doi: 10.11591/ijece.v13i3.pp3339-3347.
 - [29] A. Z. A. Zainuddin, W. Mansor, K. Y. Lee, and Z. Mahmoodin, “Machine learning and deep learning performance in classifying dyslexic children’s electroencephalogram during writing,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 6, p. 6614, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6614-6624.
 - [30] H. Daou and F. Labeau, “Dynamic Dictionary for Combined EEG Compression and Seizure Detection,” *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 1, pp. 247–256, Jan. 2014, doi: 10.1109/jbhi.2013.2263198.
 - [31] A. A. Alsakaa, M. H. Hussein, Z. H. Nasralla, H. Alsaqaa, K. Nermend, and A. Borawska, “Effective electroencephalogram based epileptic seizure detection using support vector machine and statistical moment’s features,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 5, p. 5204, Oct. 2022, doi: 10.11591/ijece.v12i5.pp5204-5213.
 - [32] Y. H. Shakir, “Epileptic Seizure Recognition,” datasheet, <https://www.kaggle.com/datasets/yasserhessein/epileptic-seizure-recognition>.

BIOGRAPHIES OF AUTHORS

Himayavardhini Jagath Prasad     received her Engineer degree in Electronics and Communication Engineering and Master degree in Applied Electronics from Saveetha Engineering college in 2013 and 2015 respectively. She was an Assistant Professor in the Department of Electronics and Communication Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences from 2016 to 2018. Currently she is a research scholar in Saveetha School of Engineering, Saveetha Institute of Medical, and Technical Sciences, Chennai. Her research interests include signal processing, image processing, and machine learning. She can be contacted at email: himayavardhini91@gmail.com.



Associate Professor Roji Marjorie S.     received her Engineer degree in Electronics and Communication Engineering from Government College of Engineering, Tirunelveli and M.Tech. in Microelectronics and VLSI from IIT Madras. She received her Ph.D. in High Power MOSFETs from JNTU Hyderabad. Currently she is working as an associate professor in the SRM Institute of Science and Technology, Kattankulathur, Chennai. Her research interests include semiconductor devices, digital electronics, high power devices, and nano technology. She can be contacted at email: rojiopp@gmail.com and rojimars@srmist.edu.in.