# Alzheimer ’s disease Detection Using Machine Learning Algorithms

1 C.Pralavi B-tech 3rd year IARE,Hyderabad,Telangana,India,chilukuripralavi1901@gmail.com

2Dr.S.China Venkateswarlu Associate Professor IARE,Hyderabad,Telangana,India,c.venkateswarlu@iare.ac.in

***Abstract:***

Alzheimer’s disease (AD) is a neurodegenerative disease affects primarily the elderly population. It is a progressive disease and the fact that there is no treatment to stop or reverse the progression of the disease. It is the one amongst neurodegenerative disorders. Though the symptoms are benign initially, they become more severe over time. Alzheimer's disease is a prevalent sort of dementia. According to the reports from 2005 through 2030, there is a steady growth in the percentage estimate of the number of people affected by AD. Presently 40 million people suffer from AD worldwide. It is distinctly possible to reach 135 million by 2050. However, an interesting feature of AD is, though incurable, early detection and appropriate treatment of the disease can control the degeneration of neurons. In the current context, Computer-Aided Diagnostics uses advanced computer programs and algorithms in the field of image processing and pattern recognition for identification of Features of Interest or Region of Interest (FOI / ROI) in the MR image under observation. This disease is challenging one because there is no treatment for the disease. Diagnosis of the disease is done but that too at the later stage only. Thus if the disease is predicted earlier, the progression or the symptoms of the disease can be slowed down. This paper uses machine learning algorithms to predict Alzheimer disease using psychological parameters.

***Keywords:***

Alzheimer’s disease; Machine learning algorithms; Computer-Aided Diagnostics; MMSE (Mini Mental State Examination).

1. **Introduction:**

Alzheimer disease is caused by both genetic and environmental factors, those affects the brain of a person over time. The genetic changes guarantee a person will develop this disease. This disease breaks the brain tissue over time. It occurs to people over age 65. However people live with this disease for about 9 years and about 1 among 8 people of age 65 and over have this disease. MMSE (Mini Mental State Examination) score is the main parameter used for prediction of the disease. This score reduces periodically if the person is affected. Those people having MCI have a serious risk of growing dementia. When the fundamental MCI results in a loss of memory, the situation expects to develop dementia due to this kind of disease. There is no treatment to cure Alzheimer's disease. In advanced stages of the disease, complications like dehydration, malnutrition or infection occurs which leads to death. The diagnosis at MCI stage will help the person to focus on a healthy approach of life, and good planning to take care of memory loss. Alzheimer’s disease (AD) is a neurodegenerative disease affects primarily the elderly population. It is a progressive disease and the fact that there is no treatment to stop or reverse the progression of the disease. According to the reports from 2005 through 2030, there is a steady growth in the percentage estimate of the number of people affected by AD. Presently 40 million people suffer from AD worldwide. It is distinctly possible to reach 135 million by 2050. However, an interesting feature of AD is, though incurable, early detection and appropriate treatment of the disease can control the degeneration of neurons. In the current context, Computer-Aided Diagnostics uses advanced computer programs and algorithms in the field of image processing and pattern recognition for identification of Features of Interest or Region of Interest (FOI / ROI) in the MR image under observation. The developed programs are expected to highlight the necessary features while keeping a control on the false negative rate systems when carefully developed are much better inaccuracies and can greatly assist the neurologist to understand the physiological changes in the brain. It is, for this reason, a significant amount of research is underway across the globe towards the classification and detection of different stages of neurodegenerative diseases including Alzheimer’s disease. Some of the research articles s found in the literature, Shide Song et al. proposed a method of classification and detection of AD based on the cortical thickness in MR images using the Gaussian Mixer Model (GMM). GMM algorithm is used for dimensionality reduction and required feature extraction. Then GMM model using a Bayesian framework is used for the classification and detection of AD. The developed method is compared with other traditional classifiers like Support Vector Machine (SVM), Linear Discriminant Analysis (LDA) and the authors claim that the proposed model has more classification accuracy. It was also stated that the maximum number of components of the distribution of each class is 2. Ruben Armananzas et al. Proposed a machine learning technique for the classification and diagnosis of the AD using Functional MR images. Functional MRI images were initially pre-processed to produce individual statistical maps of voxels using a statistical parametric mapping toolbox. Then the active voxels were selected using active filters. Then relevant voxels were selected using four feature ranking schemes. Six pattern recognition techniques were used to guide this wrapper approach scheme. The classification is further assessed by nested internal and external cross-validation loops taking several voxel sets ordered by importance. Xiao Zheng et al. proposed an algorithm for the diagnosis of the AD in earlier stages using Boltzmann machines (RBM). The proposed algorithm uses Boltzmann machine for classification of multimodal data using learning using privileged information (LUPI). Although SVM classifiers can efficiently classify single modal data, multimodal data cannot be efficiently handled by SVM. A combination of RBM and SVM classifiers using Learning using Privileged Information (LUPI) was also proposed in this paper for the efficient classification of brain diseases. Salim Lahmiri and MounirBoukadoum proposed a novel classification method for the diagnosis of the AD by modeling MR images as fractal objects. In the proposed method feature extraction is performed using feature multiscale analysis using Hursts exponents. Then SVM classifiers are used for the classification based on the features extracted. Initially, SVM classifiers were used to classify AD from healthy people, then SVM classifiers were used to classify AD from Mild Cognitive Impairment (MCI). In the third experiment, multilevel SVM classifiers were used to classify all the three classes. Results show that the classification accuracy of all three methods is high. In their experiment, four classification techniques are carried out for performance analysis the results exhibit that symlet wavelet-based fractal feature with SVM classification provides maximum accuracy of 89.7%.

* 1. **Literature Survey:**

1. **2021 (Perez-Valero, E.; Lopez-Gordo,) A Review of Automated Techniques for Assisting the Early Detection of Alzheimer’s Disease with a Focus on EEG:**

The state-of-the-art approaches that apply signal processing (SP) and machine learning (ML) to automate the detection of Alzheimer’s disease (AD) and its prodromal stages. the economic and social implications of the disease, traditional diagnosis techniques, and the fundaments of automated AD detection. Then, we present electroencephalography (EEG) as an appropriate alternative for the early detection of AD, owing to its reduced cost, portability, and non-invasiveness.[7]

1. **2020 (Zhao, Y.; Zhao, Y.; Durongbhan, P.;) Imaging of Nonlinear and Dynamic Functional Brain Connectivity Based on EEG Recordings With the Application on the Diagnosis of Alzheimer’s Disease:**

This accurately diagnoses the condition based on information derived from resting state electroencephalogram (EEG) recordings, aiming to detect brain network disruption. This article proposes a novel brain functional connectivity imaging method, particularly targeting the contribution of nonlinear dynamics of functional connectivity, on distinguishing participants with AD from healthy controls (HC).[15]

1. **2019 (Musaeus, C.S.; Salem, L.C.; Sabers, A.; Kjaer, T.W.;) Associations between electroencephalography power and Alzheimer's disease in persons with Down syndrome:**

This included 21 persons with DS-AD and 16 with DS without cognitive deterioration assessed by the informant-based Dementia Screening Questionnaire in Intellectual Disability. EEG was recorded for all participants. Absolute power for each electrode and global power were calculated for all frequency bands for both eyes open and eyes closed.[17]

1. **2018 (Bairagi, V.) EEG signal analysis for early diagnosis of Alzheimer disease using spectral and wavelet-based features:**

Electroencephalogram (EEG) signal analysis can be well suited for automated diagnosis of Alzheimer’s disease. This paper focuses on spectral and wavelet features for diagnosis of Alzheimer using EEG signals with effective increase in diagnostic accuracy for improvement in diagnosis in early stage. The use of spectral and wavelet-based features is proposed in this paper, with effective increase in classification accuracy by use of supervised classifiers giving 94% diagnostic accuracy for early Alzheimer’s diagnosis.[45]

1. **2021 (Susana, C.F.; Mónica, L.; Fernando, D.) Event-related brain potential indexes provide evidence for some decline in healthy people with subjective memory complaints during target evaluation and response inhibition processing:**

 In this study, healthy adult participants with high and low SMCs (HSMCs and LSMCs) performed a Go/NoGo task during electroencephalogram (EEG) recording. Relative to LSMC participants, HSMC participants performed the task slower (longer reaction times) and showed changes in the event-related potential (ERP) components associated with response preparation (lower [readiness potential](https://www.sciencedirect.com/topics/neuroscience/contingent-negative-variation) -RP- amplitude in the Go condition), and also related to response inhibition processes (lower N2-P3 amplitude in the NoGo condition).[14]

1. **2020 (Fu, R.; Tian, Y.; Shi, P.; Bao, T.) Automatic Detection of Epileptic Seizures in EEG Using Sparse CSP and Fisher Linear Discrimination Analysis Algorithm:**

This combined sparse idea and greedy search algorithm to improve the feature extraction of common space pattern. The feature extraction can effectively overcome the repeating selection problem of feature pattern in the eigenvector space by the traditional method. Then we used the Fisher linear discriminant analysis to realize the classification. The results show that the proposed method can get high classification accuracy using fewer data. For 10 subjects, the averaged accuracy of epilepsy detection is more than 92%.[35]

1. **Methodology:**

**2.1. Existing System:**

Method of deep learning along with the brain network and clinically significant information like age, ApoE gene and gender of the subjects for earlier examination of Alzheimer’s. Brain network was arranged, calculating functional connections in the brain region by employing the resting-state functional magnetic resonance imaging (R-fMRI) data. To produce a detailed discovery of the early AD, a deep network like autoencoder is used where functional connections of the networks are constructed and are susceptible to AD and MCI. The dataset is taken from the ADNI database. The classification model consists of the early diagnosis, initially preprocessing of raw R-fMRI is done. Then, the time series data (90 ×130matrix) is obtained and that indicates blood oxygen levels in each and every region of the brain and changes over a long period. Then, a brain network is built and transformed to a 90 ×90 time series data correlation matrix. The targeted autoencoder model is used which is a three-layered model which gives intellectual growth of the nervous system then extracts brain networks attributes completely When a finite amount of data cases is taken, k-fold cross verification was implemented mainly to avoid the over fitting complication.

**DISADVANTAGES OF EXISTING SYSTEM:**

⮚The ability to collect, store, manage and process data has been difficult in existing methods.

⮚The stage of artificial intelligence is also defined as a discipline about knowledge, namely the technology about how to acquire and express the knowledge and convert it into practical applications

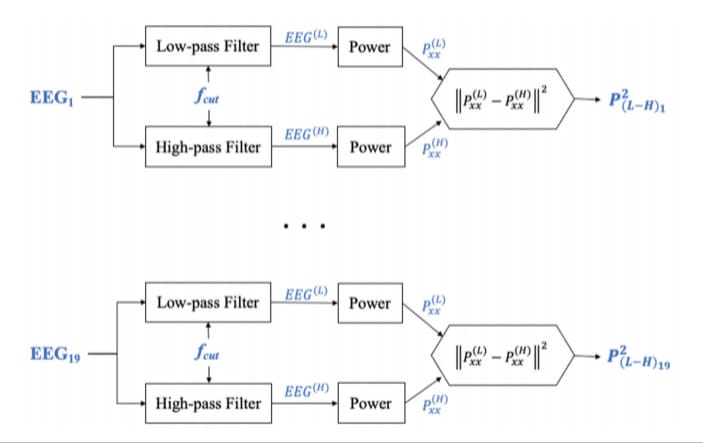


Figure-1

**2.2.**  **Proposed System:**

Proposed a method called multistage classifier by using machine learning algorithms like Support Vector Machine, Naive Bayes and K-nearest neighbor to classify between different subjects. PSO (particle swarm optimization) which is a technique that best selects the features was enforced to obtain best features. Naturally image retrieving process requires two stages: the first stage involves generating features so that it reproduces the query image and then later step correlate those features with those already gathered in the database. The PSO algorithm is used to select the finest biomarkers that show AD or MCI. The data is Alzheimer's disease Neuroimaging Initiative (ADNI) database. The MRI scans are preprocessed first after taking from the database. The feature selection includes volumetric and thickness measurements. Then the optimum feature lists were obtained from PSO algorithm. The Gaussian Naïve Bayes, K- Nearest Neighbor, Support vector machine was used to distinguish between the subjects. Here a 2-stage classifier was used where in the initial stage GNB classifier was used to classify the objects between AD, MCI and NC and in later stages SVM and KNN were used to analyze the object based on the performance of the initial one. Control Based Image Retrieval was used for retrieving images from the database.

**Advantages Of Proposed System:**

SVM is a directed study model that classifies by separating the objects using a hyperplane. It can be used for both classification and regression. The hyperplanes are drawn with the help of the margins. The main goal is to maximize the distance between the hyperplane and the margin. The margins are drawn with the help of support vectors that are belonging to the objects. The main advantage of SVM is that it can distinguish linear and non-linear objects

**2.3 System Architecture:**

Psychological parameters



Decision tree Classifier

SVM

Classifier

Performance Comparison (Accuracy)

Figure-2

**2.4 Algorithms:**

**2.4.1 Machine Learning:**

Machine learning refers to the computer’s acquisition of a kind of ability to make predictive judgments and make the best decisions by analyzing and learning a large number of existing data. The representation algorithms include deep learning, artificial neural networks, decision trees, enhancement algorithms and so on. The key way for computers to acquire artificial intelligence is machine learning. Nowadays, machine learning plays an important role in various fields of artificial intelligence. Whether in aspects of internet search, biometric identification, auto driving, Mars robot, or in American presidential election, military decision assistants and so on, basically, as long as there is a need for data analysis, machine learning can be used to play a role**.**

**2.4.2 Support Vector Machine:**

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.

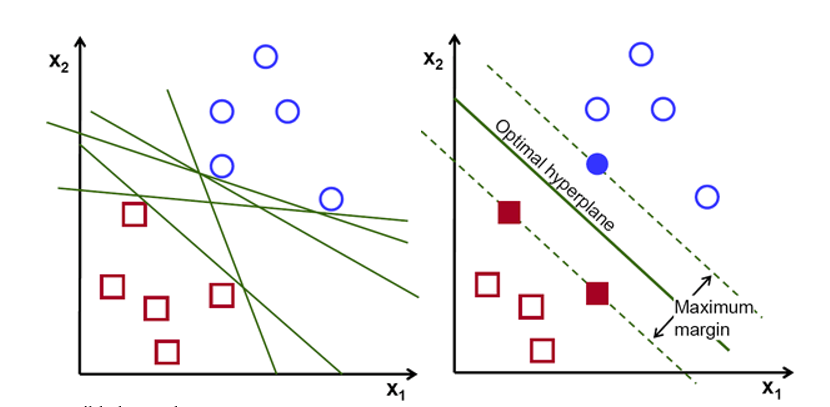
****

Figure-3

Possible hyperplanes

To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Hyperplanes and Support Vectors

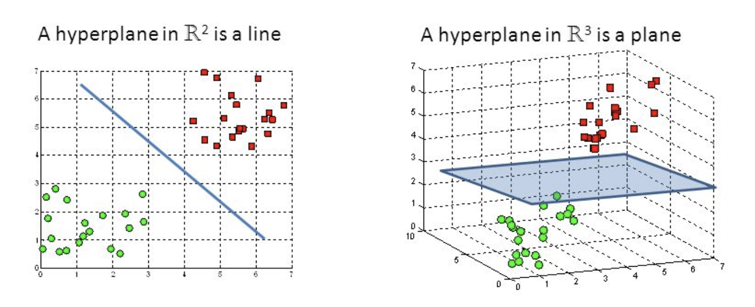
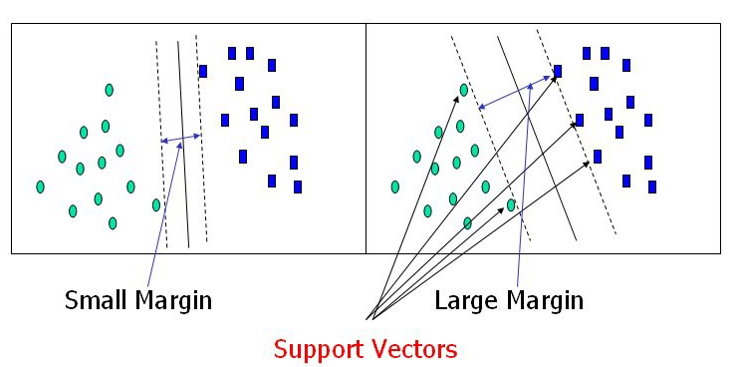
****

Figure-4

Hyperplanes in 2D and 3D feature space

Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. It becomes difficult to imagine when the number of features exceeds 3.

****Figure-5

Support Vectors

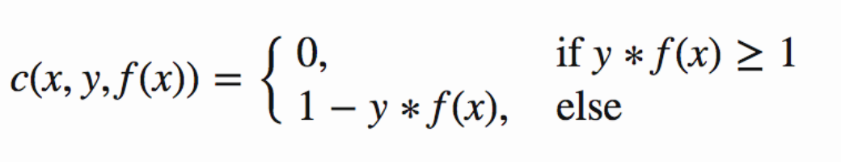
Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.

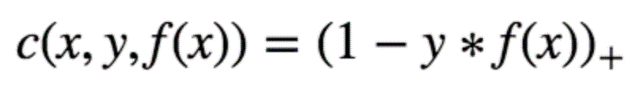
Large Margin Intuition

In logistic regression, we take the output of the linear function and squash the value within the range of [0,1] using the sigmoid function. If the squashed value is greater than a threshold value(0.5) we assign it a label 1, else we assign it a label 0. In SVM, we take the output of the linear function and if that output is greater than 1, we identify it with one class and if the output is -1, we identify is with another class. Since the threshold values are changed to 1 and -1 in SVM, we obtain this reinforcement range of values([-1,1]) which acts as margin.

Cost Function and Gradient Updates

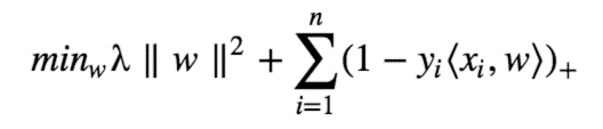
In the SVM algorithm, we are looking to maximize the margin between the data points and the hyperplane. The loss function that helps maximize the margin is hinge loss.





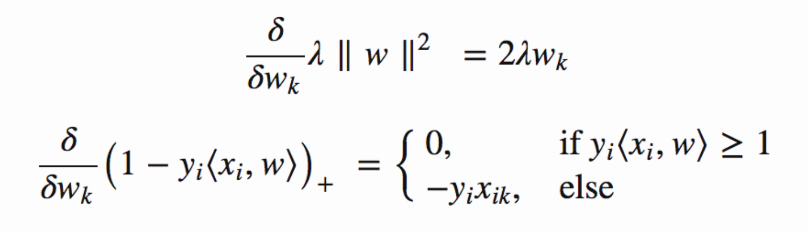
Hinge loss function (function on left can be represented as a function on the right)

The cost is 0 if the predicted value and the actual value are of the same sign. If they are not, we then calculate the loss value. We also add a regularization parameter the cost function. The objective of the regularization parameter is to balance the margin maximization and loss. After adding the regularization parameter, the cost functions looks as below.



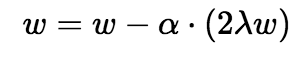
Loss function for SVM

Now that we have the loss function, we take partial derivatives with respect to the weights to find the gradients. Using the gradients, we can update our weights.



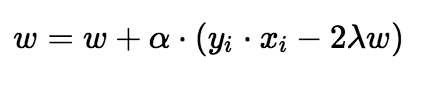
Gradients

When there is no misclassification, i.e our model correctly predicts the class of our data point, we only have to update the gradient from the regularization parameter.



Gradient Update — No misclassification

When there is a misclassification, i.e our model make a mistake on the prediction of the class of our data point, we include the loss along with the regularization parameter to perform gradient update.



Gradient Update — Misclassification

**2.4.3.Random Forest Algorithm:**

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

The below diagram explains the working of the Random Forest algorithm:



Figure-6

Note: To better understand the Random Forest Algorithm, you should have knowledge of the Decision Tree Algorithm.

Assumptions for Random Forest:

Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random forest classifier:

* There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.
* The predictions from each tree must have very low correlations.

Why use Random Forest?

Below are some points that explain why we should use the Random Forest algorithm:

<="" li="">

* It takes less training time as compared to other algorithms.
* It predicts output with high accuracy, even for the large dataset it runs efficiently.
* It can also maintain accuracy when a large proportion of data is missing.

How does Random Forest algorithm work?

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

**Step-1:** Select random K data points from the training set.

**Step-2:** Build the decision trees associated with the selected data points (Subsets).

**Step-3:** Choose the number N for decision trees that you want to build.

**Step-4:** Repeat Step 1 & 2.

**Step-5:** For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

The working of the algorithm can be better understood by the below example:

**Example:** Suppose there is a dataset that contains multiple fruit images. So, this dataset is given to the Random forest classifier. The dataset is divided into subsets and given to each decision tree. During the training phase, each decision tree produces a prediction result, and when a new data point occurs, then based on the majority of results, the Random Forest classifier predicts the final decision. Consider the below image:



Figure-7

Applications of Random Forest:

There are mainly four sectors where Random forest mostly used:

1. **Banking:** Banking sector mostly uses this algorithm for the identification of loan risk.
2. **Medicine:** With the help of this algorithm, disease trends and risks of the disease can be identified.
3. **Land Use:** We can identify the areas of similar land use by this algorithm.
4. **Marketing:** Marketing trends can be identified using this algorithm.

Advantages of Random Forest:

* Random Forest is capable of performing both Classification and Regression tasks.
* It is capable of handling large datasets with high dimensionality.
* It enhances the accuracy of the model and prevents the overfitting issue.

Disadvantages of Random Forest:

* Although random forest can be used for both classification and regression tasks, it is not more suitable for Regression tasks.

**2.4.4 Naive Bayes:**

Naive Bayes classifiers are a collection of classification algorithms based on Bayes’ Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

To start with, let us consider a dataset.

Consider a fictional dataset that describes the weather conditions for playing a game of golf. Given the weather conditions, each tuple classifies the conditions as fit(“Yes”) or unfit(“No”) for plaing golf.

Here is a tabular representation of our dataset.

|  | Outlook | Temperature | Humidity | Windy | Play Golf |
| --- | --- | --- | --- | --- | --- |
| 0 | Rainy | Hot | High | False | No |
| 1 | Rainy | Hot | High | True | No |
| 2 | Overcast | Hot | High | False | Yes |
| 3 | Sunny | Mild | High | False | Yes |
| 4 | Sunny | Cool | Normal | False | Yes |
| 5 | Sunny | Cool | Normal | True | No |
| 6 | Overcast | Cool | Normal | True | Yes |
| 7 | Rainy | Mild | High | False | No |
| 8 | Rainy | Cool | Normal | False | Yes |
| 9 | Sunny | Mild | Normal | False | Yes |
| 10 | Rainy | Mild | Normal | True | Yes |
| 11 | Overcast | Mild | High | True | Yes |
| 12 | Overcast | Hot | Normal | False | Yes |
| 13 | Sunny | Mild | High | True | No |

The dataset is divided into two parts, namely, feature matrix and the response vector.

* Feature matrix contains all the vectors(rows) of dataset in which each vector consists of the value of dependent features. In above dataset, features are ‘Outlook’, ‘Temperature’, ‘Humidity’ and ‘Windy’.
* Response vector contains the value of class variable(prediction or output) for each row of feature matrix. In above dataset, the class variable name is ‘Play golf’.

**Assumption:**

The fundamental Naive Bayes assumption is that each feature makes an:

* independent
* equal

contribution to the outcome.

With relation to our dataset, this concept can be understood as:

* We assume that no pair of features are dependent. For example, the temperature being ‘Hot’ has nothing to do with the humidity or the outlook being ‘Rainy’ has no effect on the winds. Hence, the features are assumed to be independent.
* Secondly, each feature is given the same weight(or importance). For example, knowing only temperature and humidity alone can’t predict the outcome accuratey. None of the attributes is irrelevant and assumed to be contributing equally to the outcome.
  1. **SOFTWARE USED:**

**2.5.1 PYTHON**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An interpreted language, Python has a design philosophy that emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer lines of code than might be used in languages such as C++or Java. It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open source software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit Python Software Foundation. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

**2.5.2 Jupyter Notebook:**

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and explanatory text. It's a popular tool for data exploration, prototyping, and collaboration in various fields such as data science, machine learning, and scientific computing.

Jupyter Notebook provides an interactive computing environment where you can write and execute code in cells. Each cell can contain code written in different programming languages such as Python, R, Julia, and others, depending on the available kernel. You can run individual cells or the entire notebook sequentially, which allows for a flexible and iterative workflow.

Some key features of Jupyter Notebook include:

* **Code execution:** You can write and execute code directly within notebook cells. This allows you to see the output and interact with the results immediately.
* **Rich media:** Jupyter Notebook supports the inclusion of images, videos, and interactive visualizations to enhance your documents.
* **Markdown support:** You can use Markdown syntax to create formatted text, headings, lists, tables, and more. Markdown cells allow you to provide explanations, documentation, or any other text-based content.
* **Notebook sharing:** Jupyter Notebook files have a .ipynb extension and can be easily shared with others. They can be opened and executed in Jupyter Notebook or JupyterLab environments.
* **Kernel support:** Jupyter Notebook supports multiple programming languages through kernels. This means you can have notebooks with cells written in different languages within the same document.
* **Extension ecosystem:** Jupyter Notebook has a rich ecosystem of extensions and plugins that enhance its functionality and enable features like code linting, code snippets, and more.

**2.6 Source code**

**Dataset**

I have collected few images from internet and saved all those images into a folder. The image file name must be checked properly because OpenCV can not process the image if the file name consists of Unicode characters. I have used 30 images for detecting the number plate.

***3. Results And Discussions:***

import warnings

warnings.filterwarnings('ignore')

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline

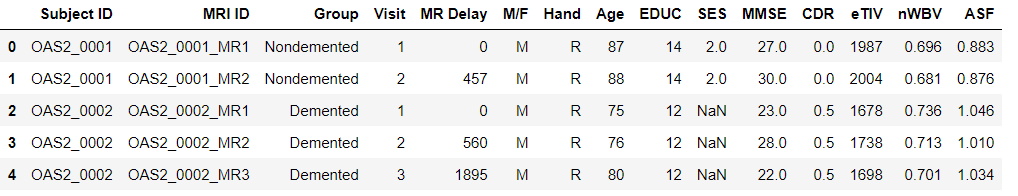
from sklearn.model\_selection import train\_test\_split

from sklearn import model\_selection

data=pd.read\_csv('oasis\_longitudinal.csv')

data.head()

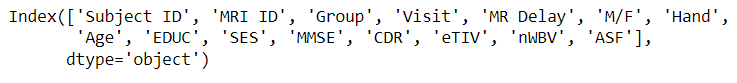
**Tabular Result:**



data.shape

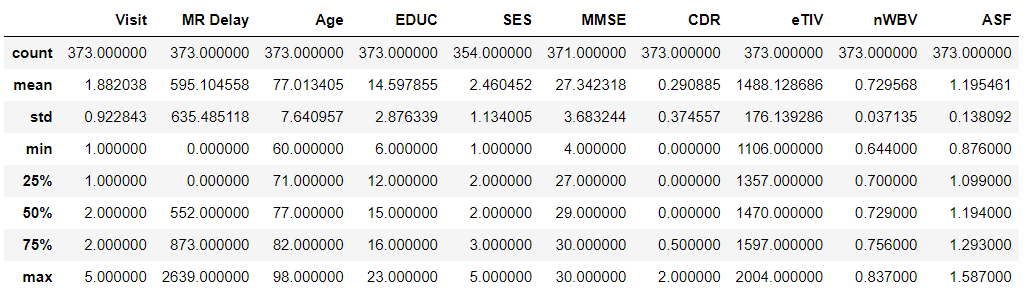


data.columns



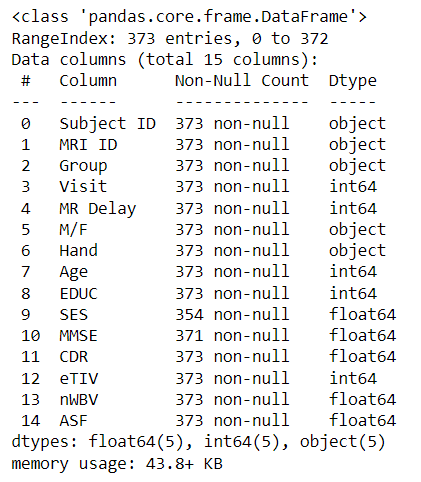
data.describe()

**Tabular Result:**



data.info()

**Output:**

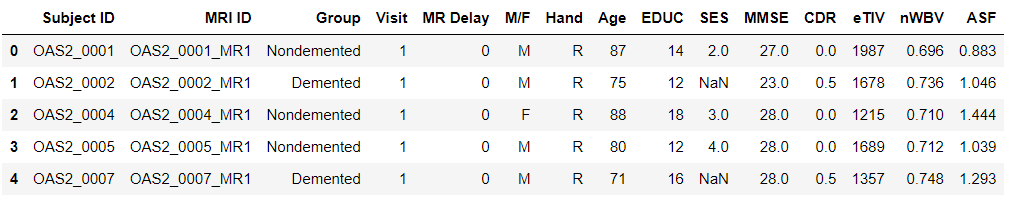


df= data.loc[data['Visit']==1]

df = df.reset\_index(drop=True)

df.head()

**Tabular Result:**



from sklearn.preprocessing import LabelEncoder

lab=LabelEncoder()

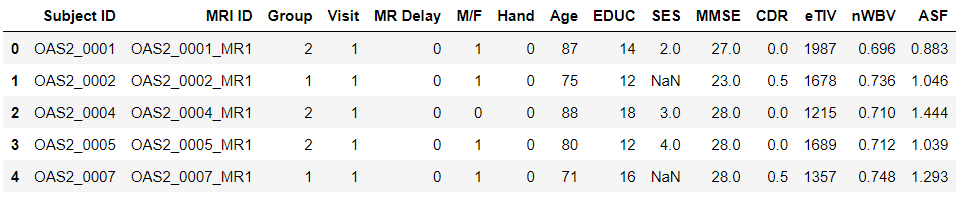
df['M/F']=lab.fit\_transform(df['M/F'])

df['Group']=lab.fit\_transform(df['Group'])

df['Hand']=lab.fit\_transform(df['Hand'])

df.head()

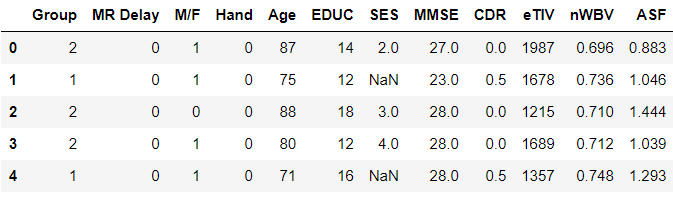
**Tabular Result:**



df1=df.drop(df[['Subject ID','MRI ID','Visit']],axis=1)

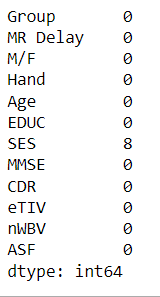
df1.head()

**Tabular Result:**



df1.isnull().sum()

**Output:**

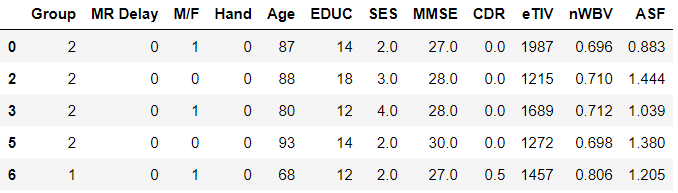


df2= df1.dropna()

df2.isnull().sum()

df2.head()

**Tabular Result:**



sns.countplot(x='Group',data=df2)

**Graphical Result:**

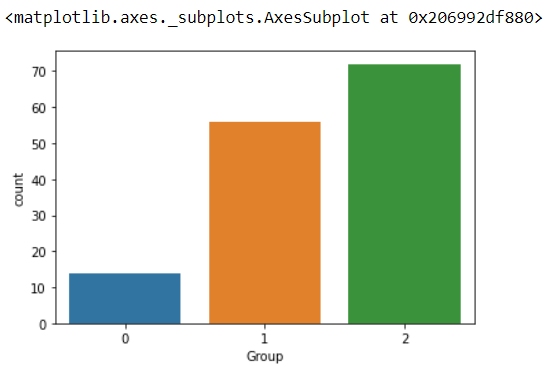


Figure-8

def bar\_chart(feature):

Demented =df[df['Group']==1][feature].value\_counts()

Nondemented = df[df['Group']==0][feature].value\_counts()

df\_bar = pd.DataFrame([Demented,Nondemented])

df\_bar.index = ['Demented','Nondemented']

df\_bar.plot(kind='bar',stacked=True,figsize=(8,5))

bar\_chart('M/F')

plt.xlabel('Group')

plt.ylabel('Number of patients')

plt.legend()

plt.title('Gender and Demented rate')

**Graphical Result:**

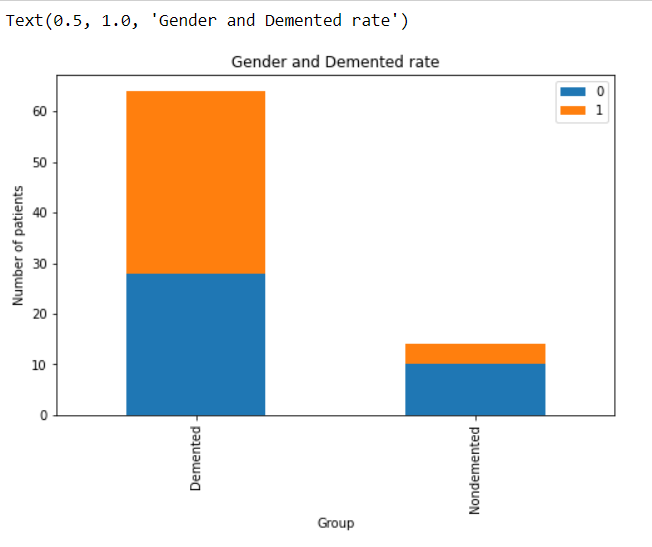


Figure-9

facet=sns.FacetGrid(df,hue='Group', aspect=3)

facet.map(sns.kdeplot,'MMSE',shade=True)

facet.set(xlim=(0,df['MMSE'].max()))

facet.add\_legend()

plt.xlim(15,30)

**Graphical Result:**

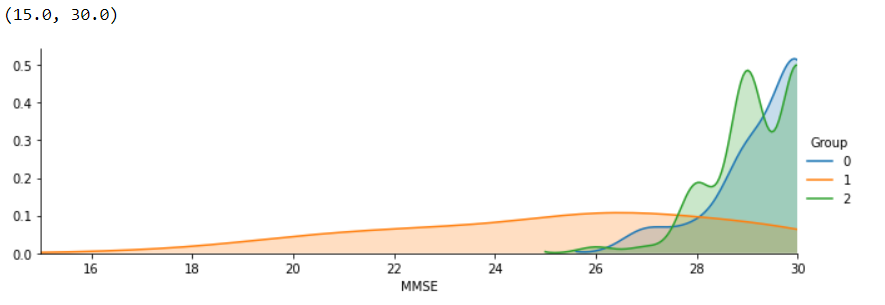


Figure-10

facet=sns.FacetGrid(df,hue='Group', aspect=3)

facet.map(sns.kdeplot,'ASF',shade=True)

facet.set(xlim=(0,df['ASF'].max()))

facet.add\_legend()

plt.xlim(0.5,2)

**Graphical Result:**

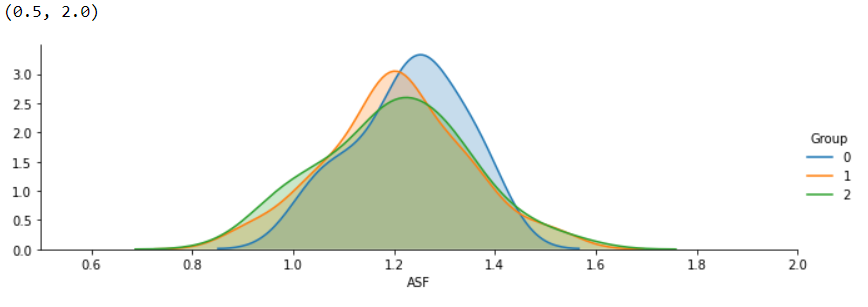


Figure-11

facet=sns.FacetGrid(df,hue='Group', aspect=3)

facet.map(sns.kdeplot,'nWBV',shade=True)

facet.set(xlim=(0,df['nWBV'].max()))

facet.add\_legend()

plt.xlim(0.6,0.9)

**Graphical Result:**

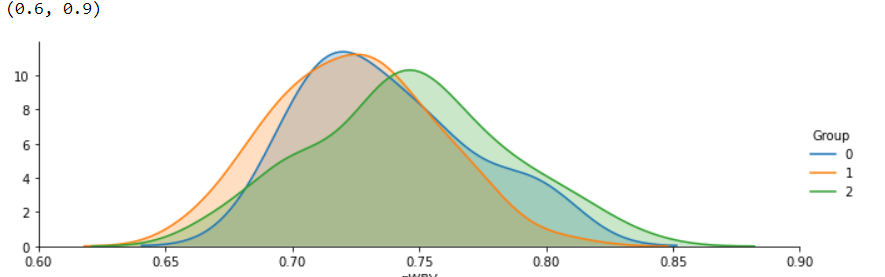


Figure-12

facet=sns.FacetGrid(df,hue='Group', aspect=3)

facet.map(sns.kdeplot,'eTIV',shade=True)

facet.set(xlim=(0,df['eTIV'].max()))

facet.add\_legend()

plt.xlim(900,2100)

**Graphical Result:**

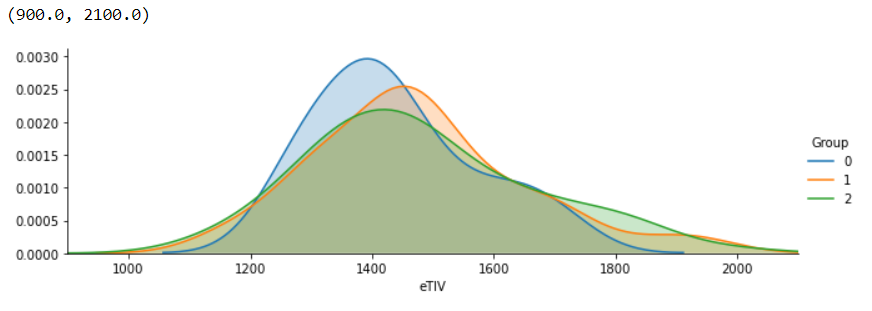


Figure-13

facet=sns.FacetGrid(df,hue="Group", aspect=3)

facet.map(sns.kdeplot,'EDUC',shade=True)

facet.set(xlim=(df['EDUC'].min(),df['EDUC'].max()))

facet.add\_legend()

plt.ylim(0,0.16)

**Graphical Result:**

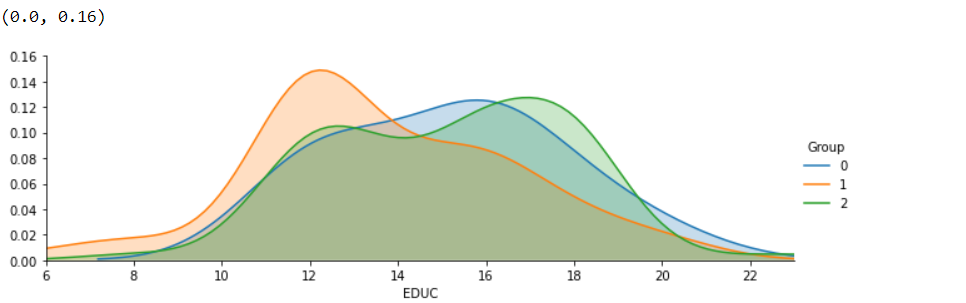


Figure-14

x=df2.iloc[:,df2.columns!='Group']

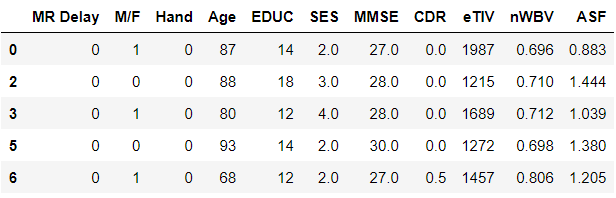
y=df2.iloc[:,df2.columns=='Group']

x.shape



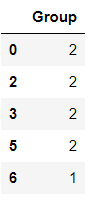
x.head()

**Tabular Result:**



y.head()

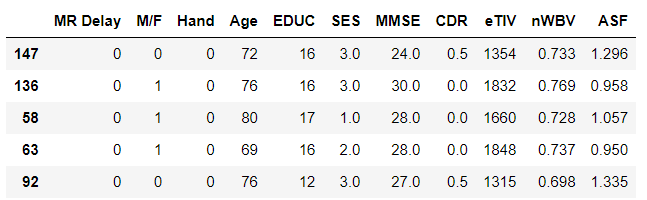
**Tabular Result:**



xtrain,xtest,ytrain,ytest=train\_test\_split(x,y,test\_size=0.3)

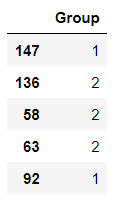
xtrain.head()

**Tabular Result:**



ytrain.head()

**Tabular Result:**



### **svm classifier**

from sklearn.svm import SVC

model1=SVC()

model1.fit(xtrain,ytrain)



predic1=model1.predict(xtest)

from sklearn.metrics import accuracy\_score

acc1=accuracy\_score(predic1,ytest)

acc1

**Output:**



from sklearn.metrics import classification\_report

def apply\_classifier(model1,xtrain,xtest,ytrain,ytest):

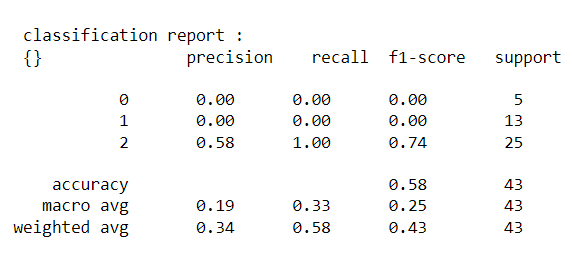
model1.fit(xtrain,ytrain)

predictions=model1.predict(xtest)

print("\n classification report : \n {}",format(classification\_report(ytest,predictions)))

apply\_classifier(model1,xtrain,xtest,ytrain,ytest)

**Output:**



### **SVM with Kernal Tricks**

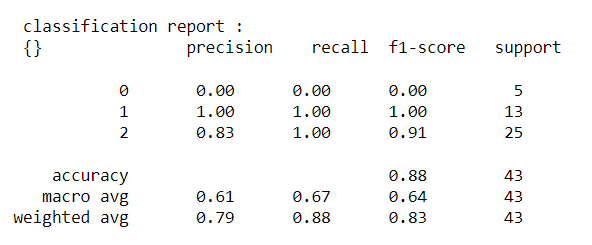
model\_linear\_kernal = SVC(kernel='linear')

model\_linear\_kernal.fit(xtrain, ytrain)



apply\_classifier(model\_linear\_kernal,xtrain,xtest,ytrain,ytest)

**Output:**



results = model\_selection.cross\_val\_score(model\_linear\_kernal, x, y)

print("MeanSqareError(MSE): %.3f (%.3f)" % (results.mean(), results.std()))

**Output:**



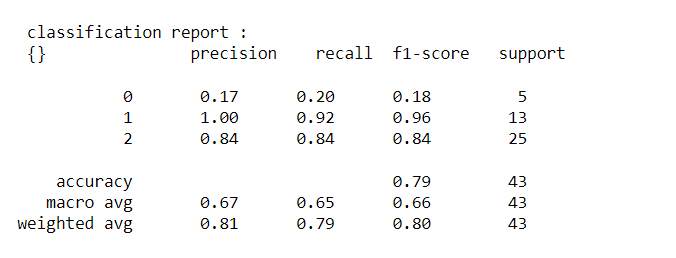
### **Decision Tree Classifier**

from sklearn.tree import DecisionTreeClassifier

decision\_tree=DecisionTreeClassifier(random\_state= 42)

apply\_classifier(decision\_tree,xtrain,xtest,ytrain,ytest)

**Output:**



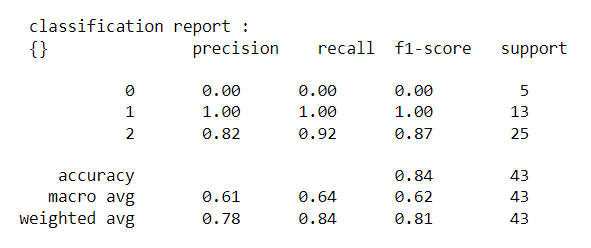
**Random Forest**

from sklearn.ensemble import RandomForestClassifier

random\_forest=RandomForestClassifier(random\_state=42)

apply\_classifier(random\_forest,xtrain,xtest,ytrain,ytest)

**Output:**



### **Gaussian Naive\_bayes**

from sklearn.naive\_bayes import GaussianNB

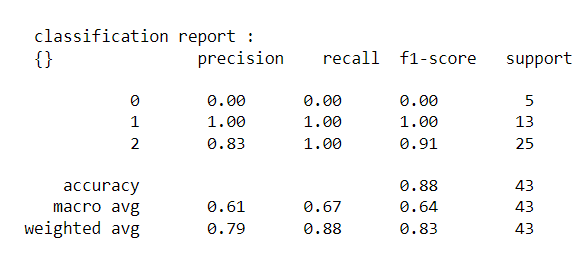
model = GaussianNB()

model.fit(xtrain,ytrain)

model.score(xtest,ytest)

apply\_classifier(model,xtrain,xtest,ytrain,ytest)

**Output:**

**

**4.Conclusion:**

Machine learning approach to predict the Alzheimer disease using machine learning algorithms is successfully implemented and gives greater prediction accuracy results. The model predicts the disease in the patient and also distinguishes between the cognitive impairment. The future work can be done by combining both brain MRI scans and the psychological parameters to predict the disease with higher accuracy using machine learning algorithms. When they are combined, the disease could be predicted with a higher accuracy in the earlier stage itself.

**5.Acknowledgement:**

I would like to express my sincere gratitude and heartfelt thanks to my mentor, Dr. S China Venkateswarlu, for his invaluable guidance and support throughout the duration of this project. His expertise, encouragement, and constant motivation were instrumental in shaping this project and ensuring its successful completion. Dr. S China Venkateswarlu’ s profound knowledge in various field and his unwavering commitment to excellence have been truly inspiring. His insightful feedback, constructive criticism, and continuous encouragement have greatly contributed to the development of my skills and understanding of the subject matter. I am truly grateful for his mentorship, patience, and unwavering support, which have played a pivotal role in the successful execution of this project.

**6.References:**

1. Perez-Valero, E.; Lopez-Gordo, M.A.; Morillas, C.; Pelayo, F.; Vaquero-Blasco, M.A. A Review of Automated Techniques for Assisting the Early Detection of Alzheimer’s Disease with a Focus on EEG. J. Alzheimers Dis. 2021, 80, 1363–1376.
2. Zhao, Y.; Zhao, Y.; Durongbhan, P.; Chen, L.; Liu, J.; Billings, S.A.; Zis, P.; Unwin, Z.C.; De Marco, M.; Venneri, A.; et al. Imaging of Nonlinear and Dynamic Functional Brain Connectivity Based on EEG Recordings With the Application on the Diagnosis of Alzheimer’s Disease. IEEE Trans. Med. Imaging 2020, 39, 1571–1581.
3. Musaeus, C.S.; Salem, L.C.; Sabers, A.; Kjaer, T.W.; Waldemar, G. Associations between electroencephalography power and Alzheimer’s disease in persons with Down syndrome. J. Intellect. Disabil. Res. 2019, 63, 1151–1157.
4. Bairagi, V. EEG signal analysis for early diagnosis of Alzheimer disease using spectral and wavelet based features. Int. J. Inf.Technol. 2018, 10, 403–412.
5. Susana, C.F.; Mónica, L.; Fernando, D. Event-related brain potential indexes provide evidence for some decline in healthy people with subjective memory complaints during target evaluation and response inhibition processing. Neurobiol. Learn. Mem. 2021,182, 107450.
6. Fu, R.; Tian, Y.; Shi, P.; Bao, T. Automatic Detection of Epileptic Seizures in EEG Using Sparse CSP and Fisher Linear Discrimination Analysis Algorithm. J. Med. Syst. 2020, 44, 43.