Predicting Brain Age from EEG Signals Using Machine Learning and Neural Network

Submitted by

Abul Mushfique Muslah Pratanu
18201183
Fuad Ibne Jashim Farhad
18301229
Aysha Afnan
18301039
Nusrat Jahan Mim
18301003
Farhin Rahman
18301001

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 29, 2022

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Pratanu

Abul Mushfique Muslah Pratanu
18201183

Fuad Farhad

Fuad Ibne Jashim Farhad
18301229

Aysha Afnan

Aysha Afnan
18301039

Nusrat Jahan Mim

Nusrat Jahan Mim
18301003

Farhin Rahman

Farhin Rahman
18301001

Approval

The thesis titled "Predicting Brain Age from EEG Signals Using Machine Learning and Neural Network" submitted by

1. Abul Mushfique Muslah Pratanu-18201183
2. Fuad Ibne Jashim Farhad-18301229
3. Aysha Afnan- 18301039
4. Nusrat Jahan Mim-18301003
5. Farhin Rahman - 18301001

Of Spring, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 28, 2022.

Examining Committee:

Supervisor:
(Member)

Amitabha Chakrabarty, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Muhammad Iqbal Hossain,PhD
Assistant Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Ethics Statement

Our personal findings have brought us to the conclusion that all the claims in our research are true. All of the papers and archives that were used in this paper were appropriately cited and referenced. This work or any part of it has never been submitted to another college or academic institution for the purpose of acquiring a degree.

Dedication

This particular gratitude goes to our university's counselors, without whom we would not have been able to accomplish it. Our lecturers were more than just academic advisers; they also offered advice and assistance when we needed it the most.

Abstract

The objective of this study was to develop a technique for calculating the ages of people's brains by analyzing EEG data signals and using machine learning algorithms on a Raspberry Pi. We employed many machine learning techniques, including random forest (RF), Decision Tree Classifier, K Nearest Neighbors Classifier (K-NN), Gaussian Naive Bayes, and Multi-layer Perceptron classifier(MLP). K-NN stands for K-nearest Neighbors, whereas RF stands for Random Forest. We initially implemented our machine learning algorithms on a desktop computer with many bells and whistles, where the dataset was also trained. By applying the Random Forest classifier (RF), we were able to attain 90% accuracy, the maximum feasible. The K-Nearest Neighbors classifier placed second with an accuracy of 87%. The accuracy obtained by the Decision Tree Classifier, the Naive Bayes algorithm, and the MLP algorithm, in order, was 83%, 39%, and 40%, respectively. Our major aim was the creation of an Internet of Things-based device, we tested our data on Raspberry Pi. If in the future, we were to construct, based on our model, a device that rapidly turned EEG brain signals into the participant's brain age, we would be able to significantly improve the quality of our work. In addition, it will aid in the diagnosis of some brain illnesses at an early stage, which has been a struggle up until now.

Keywords: EEG, Brain age, K-NN, RF, Decision Tree, MLP, Naive Bayes, Raspberry Pi

Acknowledgement

First and foremost, we give thanks to the Almighty Allah for allowing us to finish our thesis without any serious setbacks. Secondly, we would like to express our gratitude to our supervisor, Dr. Amitabha Chakraborty sir and our co-supervisor, Dr. Muhammad Iqbal Hossain sir, for their invaluable assistance and advice. They came to our aid whenever we needed. Finally, without our parents' continuous encouragement, we may not be able to achieve our goals. We are currently on the verge of graduation. Thus, thanks to their gracious support and prayers.

Table of Contents

List of Figures

List of Tables

Nomenclature

The following rundown depicts a few images and condensing that will be subsequently utilized inside the body of the report

AI Artificial Intelligence

EEG Electroencephalogram

$K-NN$ K-Nearest Neighbour

ML Machine Learning

MLP Multilayer Perceptron

MLP Multilayer perceptron

NV Naive Bayes

RF Random Forest

RPi Raspberry Pi

RVM Relevance Vector Machine

SVM Support Vector Machines

Chapter 1

Introduction

In certain nations, the birth certificate is signed as soon as the baby is born, and the government retains a record of practically everyone's precise age. In other countries, however, the birth certificate is not signed immediately after the baby is delivered. This permits more accurate social services. As a consequence of this, age assessment based on EEG may give the impression of being unduly expensive. The electroencephalogram (EEG) test, on the other hand, determines not just a person's chronological age but also their brain age in addition to their actual age. Electroencephalography (EEG), which may be categorized as a subfield of neuropsychology due to the ease with which it gathers and evaluates electrical activity, is one of the most widely used methods in the field. It is possible to determine a person's chronological age by moving backward from their date of birth to a later date while simultaneously determining the person's brain age. An individual's age is often regarded as an effort to summarize their advancement through the aging process using a single number. Gray matter, a component of the layer of the brain known as the white matter, becomes less dense as we become older. The sulci that have always been there on its surface will become more noticeable as it continues to develop and deepen. The effects of aging have a variety of diverse manifestations in the minds of different people, despite the fact that the underlying tendency of aging is something that is common to all brains. In a nutshell, the purpose of Brain Age is to quantify the whole of the aging process in a single value. Methods of brain age prediction may forecast a person's age-related decline in cognitive skills, including those linked with dementia, Alzheimer's disease, epilepsy, and other brain illnesses. This includes the loss of cognitive abilities associated with the aging process itself. Age is a factor that contributes to a decline in a person's cognitive skills, such as those listed above. In this article, we propose a portable, real-time data collecting and processing system that is built on Raspberry Pi and uses machine learning to correctly anticipate brain ages based on EEG data. The system is portable and can be used anywhere. Python was the programming language that was used throughout the development of the system. Extensive research has been done in order to develop a technique for the digitization of EEG data. This project is to build an EEG classification system that, through the use of machine learning, will be able to determine an individual's age. The primary computing device that will be used for this endeavor will be a Raspberry Pi(RPi) [12]. This technique makes use of a variety of different models and algorithms for machine learning in order to calculate the age of the brain. It is possible to utilize it in conjunction with live EEG recording equipment in order to get the most accurate results.

1.1 Research Problem

Age has always played a key role in the evolution of human identity throughout human history. It is crucial that we maintain open channels of communication in both our personal and professional life. For example, security, cosmetics, online commerce, and intelligent human-machine interfaces (IHMI) might all benefit from artificial intelligence-enabled age prediction [2]. Verifying the ages of people with Alzheimer's disease and other kinds of dementia is substantially more necessary than for healthy individuals. Depending on the stage of Alzheimer's disease a person is experiencing, the illness's symptoms present in a number of ways. As a result, establishing an accurate age estimate by EEG is crucial for patients with significant diseases [16]. Using EEG to establish a person's age is one method for determining whether or not a person's brain is working appropriately when placed in a particular age group.

Incorporating all of these examinations, including X-rays of the teeth and radiographic evaluation of the clavicles, led to the final conclusion of age [17]. This led to the eventual conclusion of age determination being discovered. When a patient's age is unknown, it is impossible to offer the required degree of emergency medical treatment, which has proven to be a serious obstacle in several instances. Additionally, the human face has been used to assess an individual's age. In addition to undergoing changes in size and shape, the face of a child developing into an adult undergoes a sequence of mathematical alterations. It is exceedingly difficult, if not impossible, to calculate the exact age of an elderly person using this approach. Historically, a person's age could be determined by evaluating the size and shape of their face using neural networks and machine learning algorithms such as NNC, SVC, and LDC. These procedures were utilized [29].

Given that we are attempting to calculate ages based on EEG data, it is crucial that we have a thorough knowledge of these signals. There are four groups of EEG signal frequencies: alpha, beta, gamma, and theta. The lowest frequency is alpha, whereas the highest frequency is theta. The range's lowest frequency is 0.01 Hz, while its maximum frequency exceeds 100 Hz. Using this signal, the surface area of the scalp may be estimated. Each signal has its own individual set of phases and qualities. Electroencephalogram (EEG) data can be handled by eliminating artifacts and filtering scalp data rather than brain data. Utilizing the Fourier Transform, the Wavelet Transform, and Principal Component Analysis, it is possible to distinguish between four separate frequency bands during signal analysis (PCA) The delta frequency is between 1 and 4 hertz, whereas the theta frequency is between 4 and 8 hertz. 8-13 Hz Alpha, beta 13-30Hz [7]. After the feature extraction procedure is complete, it is necessary to categorize the nonstationary signal characteristics that were extracted [3]. In order to acquire a greater understanding of how the brain operates, EEG data may be divided into several categories based on the waveform, heat distribution, and symmetry of the signal. In order for us to proceed with our study, we must thus establish how to use EEG signals to reliably identify an individual's age. We will need to educate ourselves on a vast diversity of distinct methods. Because we will forecast a person's age based on an EEG signal, selecting a dataset provides a significant issue. Numerous websites provide consumers with access to numerous datasets. In conclusion, we chose to utilize the Kaggle dataset since it is feasible to properly determine a person's age using this EEG signal recording [7]. After that, we will need to test a variety of tactics and approaches to determine which ones are the most effective. A distinct kind of machine learning has been implemented for the purpose of age prediction. To prevent overfitting, ENEST regularization has been used for linear regression. Using the SVR method [24], high dimensional training data are optimized before the creation of the regression model. Grouping with RF is performed to create multi-weak learners capable of executing the training data's feature space [24]. Construct a group of grouping learning, optimization, and regularization using XgbTree to provide a generic model. Gausspr Poly is a probabilistic technique that creates a regression model from training data [24]; it is a subset of the Bayesian approach. Electroencephalogram (EEG) waves may be evaluated using various machine learning approaches to calculate an individual's age.

1.2 Research Objective

Our research objective is to build an IoT-based device in the future which will accurately translate EEG signals into brain age with the help of five different machine learning algorithms. The study focuses on neural networks and machine learning techniques to develop a system that can predict an individual's Brain age based on EEG data. With the help of an EEG dataset, we obtained from Kaggle, we started our work. After processing our dataset for predicting age, we will also use five machine learning algorithms to train the dataset. We will try to get the best possible accuracy from the algorithms we will use, using feature scaling and reducing null values. We will try to ascertain the performance of models on different performance measures. Next, we will try to find out one age class for a participant. And after we ascertain which model gives us the best performance, we will test our data in Raspberry PI as our device will be an IoT-based device. And lastly, we are hoping to build a cheap and portable device that can be used in the medical field to detect brain age accurately, which will further lead to the early detection of brain disorders like Alzheimer's disease, Epileptic Seizure, Dementia, etc.

Chapter 2

Background

2.1 Literature Search Process

We must perform an organized literature search with a certain research topic in mind. We discovered our datasets using a search for "predict age" on Kaggle (https://www.kaggle.com/) and "EEG signal" on the IEEE port. Additionally, we searched for papers that had the keyword "Machine Learning" and "Data Analysis" which produced a large number of hits. As a consequence, by providing the search word, we gradually reduced it.

2.2 Related Works

This section will assess previous research on brain age prediction. MRI was once used to predict brain age, but new studies have shown that machine learning and neural algorithms can also be used to predict brain age. We'll look at each technique, as well as the restrictions and problems of algorithms, in order to make an accurate prediction.

A paper worked on EEG signals along with an ML algorithm to estimate Brain age and actual age. They have worked with a set of regression algorithms and features of EEG signals to have a better age estimation. The author mentioned that selecting proper algorithms was a challenge for them. Five regression algorithms including ENET, SVR, RF, XgbTree, and gaussprPoly. A new method is proposed in this paper to predict age and gender using EEG signals. The use of BCI (Brain-Computer interfaces) technology in applications is increasing rapidly. Predicting age and gender is a new approach to BCI. Primarily, a traditional ML algorithm was used to predict age but the problem is that features that are learned using these ML are not appropriate for all applications [27] [5]. Deep learning models extract features that are not possible using traditional extracting. The 'Deep BLSTM-LSTM' framework has been constructed using bidirectional LSTM in this research. Accuracy in predicting age and gender has been found in this paper, respectively 93.69% and 97.5%.

A new detection methodology for ML is offered in this research paper. The author here worked on predicting children's and teenagers' ages. Perhaps a person's brain development is reflected in their consistent prediction error which can be a measure of stability, they looked for if the overestimation or underestimation of age was constant over time. Three machine learning algorithms were used by Marjolein

RF, SVM, and RVM. In this paper, the EEG channel is compared to MRI age estimations. While comparing children and teenagers recordings are more accurate than MRIs. EEG recordings produce RVM accuracy levels of approximately 95%, while MRI accuracy levels ranged from 75% to 95%. link statistical characteristics with the age range in the study effort [10]. The author employed LDA to see the differences in the EEG signal in the process of aging and also employed traditional features to see the differences in the EEG signals of the younger and the elders.
With the rapid growth of IoT, the healthcare field got an extension ranging from wearable sensors to medical equipment connecting clouds in previous monitoring systems. Different paper works with mainly two ways to collect data through IoT devices and send them to the main server. One is, that data is collected from patients and the environment and sent to the cloud. Data is stored there, processed, analyzed, and then systematic decisions are made and this requires more network resources like bandwidth. Edge computing comes then to solve these concerns. Where edge devices can do computational tasks to avoid delay and work fast and then data is sent to the main server [18]. Since many studies and research has been done on implementing EEG signals on IoT devices, in this section we will discuss the latest studies and research on this process.
This paper [38] has used EEG signals for authentication methods of IoT devices. NeuroSky Mindwave headset is used for choosing an adaptive threshold for the authentication key. And a camera to capture hand gestures for controlling authentication systems. This study also proposes a technique based on EEG and hand motions. The NeuroSky MindWave headset was used in this study as a single channel Brain-Computer Interaction (BCI) device. The Raspberry Pi board is an excellent example of an IoT device. To control EEG authentication method operations, a Raspberry Pi camera is used to recognize and categorize hand gestures. The goal of the project is to see whether an EEG authentication method that leverages NeuroSky MindWave as a unique solution for IoT device authentication is beneficial. The user supplies four bits for authentication, as described in the previous section. Each participant entered ten authentication sequences, each of which consisted of a unique four-bit sequence, to assess the utility of the EEG authentication approach.
To improve the interaction between motor disabled people and the environment, this paper [36] has worked with a BCI interface that controls IoT devices using EEG signals. Muse headband is used here for recording EEG signals. K-NN algorithm is used for the classification of signals and classified results are then sent to a cloud server, mosquito, to manage the action of IoT devices.
This author [28] has proposed a framework for automatically predicting age and gender in IoT based healthcare devices.Which can be further implemented in IoT devices in the healthcare sector. For monitoring a patient remotely, using EEG signal monitoring brain signals is a new pavement.For the prediction, wireless EEG sensors are used to collect brain activities. To process the signals,first they are sent to the phone,PC and then again sent to hospitals and emergency centres. Discrete wavelet transform for feature extraction and random forest classifier for brain signal modeling have been performed. They have represented that this proposed algorithm takes less EEG signal to predict age and gender. Author has talked about the challenge that faced while placing electrodes over female candidate's scalps to collect EEG signal. This happened for females hair density. Using Random Forest classifier, optimum accuracy obtained 88.33% for age and 96.66% for gender prediction.

Diagnosis of disease at an early stage from remote is becoming an essential strategy of the healthcare sector. To assist this strategy, a sophisticated diagnosis system to diagnose insomnia is introduced in this paper [35]. Neurosky Mindwave headset is used here to collect EEG signals for diagnosis. Author hopes for this approach to diagnose disease from the remote in future.

From the focus of building an BCI system based on EEG to help daily interactions of paralysed and impaired people,this paper [34] worked on a EEG based device for interpreting eye movement. From each age group, above 30 to 90, 1 to 2 people are used for collecting data. Open BCI Ganglion is used for data acquisition and data were labeled while the subject was seeing the three actions - eyes up,eyes down,blink are used. A deep neural network along with TensorFlow was also built for detecting open and close states of eyes as blinking state is much easier to detect [34].

The author designed a wheelchair and automated home for quadriplegics with BCI and IoT [31]. Signal due to blinking controls the wheelchair. Brain waves can be recorded through EEG using BCI. And in this system a patient in a wheelchair wearing a Neurosky headset, blinking eyes, thinking of moving, EEG signal is collected through Neurosky. Later on, Neurosky is connected to a PC with Bluetooth. MATLAB code, Arduino UNO process signal and produce input to control the movement of the wheelchair. For home automation again EEG signal is provided to switch on or off the lights, fan, and tv and to open, and close the door.

In a study of a prototype of an EEG system for IoT [33], the author develops a device with a reduced number of sensors and a dual-core microcontroller. Sensors for getting EEG signals and microcontrollers for performing signal classification operations and transmitting data simultaneously to IoT devices. The author proposed a method where depending on the power ratio of frequency bands detection of the user's eye states is determined. Then data related to eye states are used to control IoT devices.

EEG signals are mostly used for visually monitoring and studying purposes. From visual representation graphical analysis of signals are useful but at the same time sufficient many times due to the difficulty of analyzing large data. This paper [30] works with decision tree algorithms to mitigate the large data analysis issue. Decision trees are quite easy to understand and analysis of EEG signals also becomes easy as using decision trees, which brain part is active and no occurrence of a pattern is easily understandable. To present the methodology, the author studies a case where two groups of sighted and visually impaired people's ability to identify objects from distance. One issue the author faced after the result of the methodology is that if the trees are larger, then accuracy is 90% but signals are difficult to read otherwise accuracy is average, 45%, and easy to read.

British scientists have discovered that those who are physically and cognitively weak are likewise weak when measured by brain age rather than chronological age. They utilize MRI images to compare a person's physical age with their brain age. They also discovered that persons who move slowly or have a limited lung capacity have a younger brain. The Aging process is complex because it includes organs and tissues. In this paper [25], the author proposed 7 steps for brain aging. They found marginal errors for five years using MRI scans; also it would need to be fine-tuned for accuracy before it could be used in this way. Apart from all of these, MRI scans are now too costly to be employed as a wide world screening tool.

In this paper [39], the author proposed a smart model called Logic-in-Headbands which is based on Edge Analytics (LiHEA) to decrease latency and bandwidth use by smoothly integrating with consumer-grade EEG headsets. The LiHEA framework was created to prevent transmitting superfluous raw EEG data to the cloud for calculation. They want to capture and analyze brainwave signals directly through wearable devices like EEG headbands. The analytics system of the ultra-edge includes the sensor of EEG in order to collect EEG signals, as well as the capacity to interpret and make appropriate inferences and categorize the data.
The Traumatic Brain injury diagnosis process requires extensive medical setup, this takes much time in diagnosis. To have early detection this paper works with Raspberry Pi to have high performance in computation. For classification authors use CNN and XGBoost models. And to capture EEG signals use MCP3008 ADC [37].
When EEGs are recorded in a laboratory, the data might get corrupted by outside factors such as sound, frequency waves, and the activation of muscles. The applicability of the findings will suffer as a direct consequence of this pollution. In this study, a potential solution to this issue of applicability reduction is presented. Instead, the author of the laboratory report employed Raspberry Pi, which is a gadget that is inexpensive in cost, lightweight, and compact. The outcomes of using Raspberry Pi include minimal costs and dependability. This experiment is carried out with ten subjects ranging in age from 18 to 25 [20].
The experiment consists of running a session on a Raspberry Pi and a PC in the lab. They tested a Raspberry Pi computer on a regular desktop PC in order to determine whether or not the Raspberry Pi could be used to present excitation for EEG testing and get acceptable ERP data [20]. The ERP measurements that are generated need to be checked and produced accurately. It took less than 11.15 minutes on average to get Trigger Two Tone for Raspberry Pi up and running. At this point in time, the Raspberry Pi offers a larger capacity for adaptability than a traditional personal computer.

Paper ID	Work	Used Algorithm	No of Participants	Accuracy	Dataset Availability
10.3389/fnagi.2018.00184	Researches actual age and brain age can be predicted or not using ML framework and EEG signals.	ENET,SVR,RF,XgbTree,gaussprPoly	468 people	MAE:6.87(0.69)	Yes
10.1007/s00521-018-3397-1	Presented a framework that can predict age and gender from EEG signals automatically.	Random Forest Classifier.	60 people. Age range: 6-55	age:88.33% gender:96.66%	Yes
10.1002/hbm.24501	Investigated whether EEG signals can predict age of children or not.	RF,SVM,RVM	Two twin family dataset of 836 and 621 people.	RF:93.9% RVM:95.2%	Yes
10.1109/JSEN.2018.2885582	Used deep neural networks to predict age and gender from EEG signals.	Bidirectional LSTM	60 participants.	Age:93.69% Gender:97.5%	Yes
10.1080/08839514.2018.1451217	Proposed a classification system to predict gender and age from the human face.	Feed Forward ANN	1000 grayscale facial images	95%	Yes
10.1109/JBHI.2021.3083187	Researched that ML algorithm can predict brain age more accurately.	Quadratic Support Vector, Binary Decision Tree	788 people	Quadratic Support vector:95% Binary decision tree:95%	Yes

Figure 2.1: Summary of the Related Papers

After reviewing all of the studies, we can conclude that EEG signals can be used to predict brain age using Machine Learning techniques, including Neural Network methods. These studies are also extremely accurate. However, we can see that these studies used datasets with very small amounts of data. They used data from a hundred or thousand people. We want to create an IoT gadget that can automatically predict age and gender, and we'll need a vast dataset to do it. If our model is built with a limited dataset, it is possible that we may not able to predict a person's brain age accurate. As a result, the IoT device will not correctly anticipate age. Despite the great accuracy of ML systems in detecting brain age, we chose to predict using a big dataset using ML algorithms again.

2.3 Used Algorithm

2.3.1 Supervised Learning

Machine learning is a subcategory of artificial intelligence, and supervised learning is one of the three forms. In order to find patterns, generate classifications, and predict outcomes as exactly as possible, this particular sort of machine learning uses labeled datasets. To get the desired result, it needs a large amount of data to understand patterns and train certain models. The model learns over time by incorporating additional data into the dataset and comparing results, which helps it to become more accurate in delivering outputs. The model's correctness is determined by implementing the loss function and updating itself until the error is sufficiently decreased to the point where it is insignificant. These images demonstrate how Supervised Machine Learning works in action [21] [32].

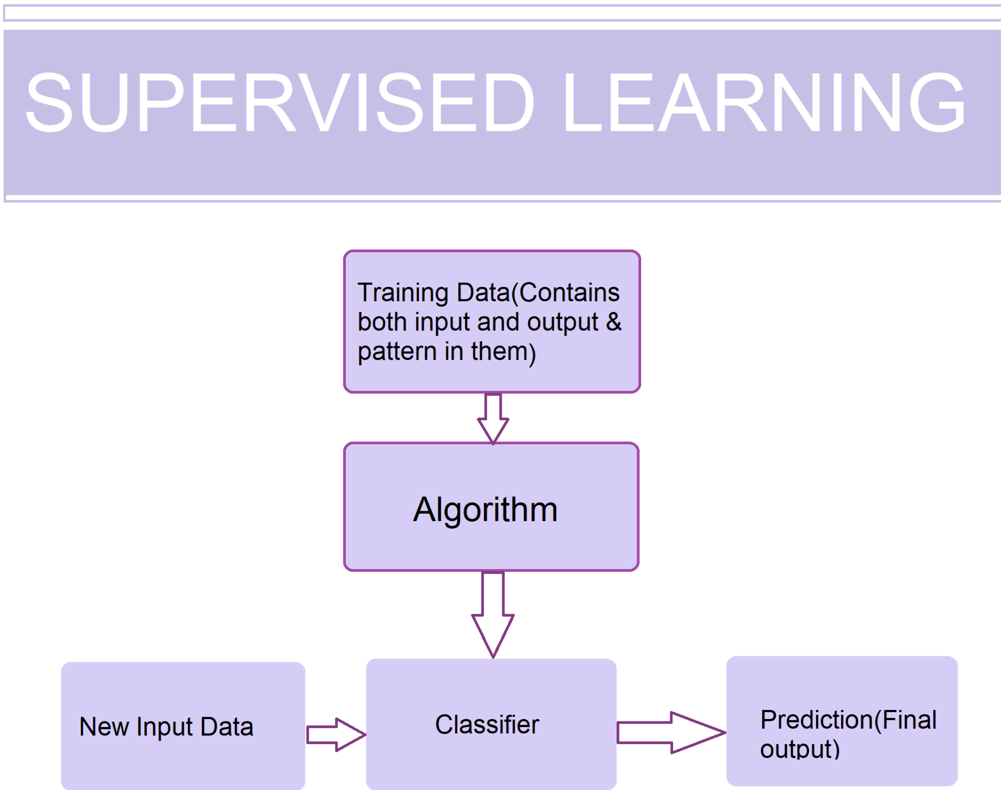

Figure 2.2: Supervised learning algorithms

2.3.2 K-Nearest Neighbor

Machine learning method K-Nearest Neighbor is based on Supervised Learning. In the K-NN technique, a new case or collection of data is compared to the previous ones, then placed in a category that is the most comparable. K-NN is used to store data, and new data points are classified depending on how similar they are to data already stored in K-NN [22]. In order to quickly and accurately classify fresh data, one may use the K-NN technique. K-NN may be used for regression and classification, however, it is most often employed for classification. K-NN does not make any assumptions about the underlying data because of its non-parametric character. K-NN. A lazy learner algorithm is also known because it does not instantly learn from the training set, but instead retains the knowledge and takes action during classification. Classifying new data into a category comparable to the new data is a key part of the K-NN algorithm's training phase [8] [19]. Let's say we have a creature that resembles either a horse or a sheep, but we're unsure if it is. The K-NN method may be used to identify the desired target. If we can find similar characteristics in the new data set, our K-NN model will classify it based on which features are most similar.

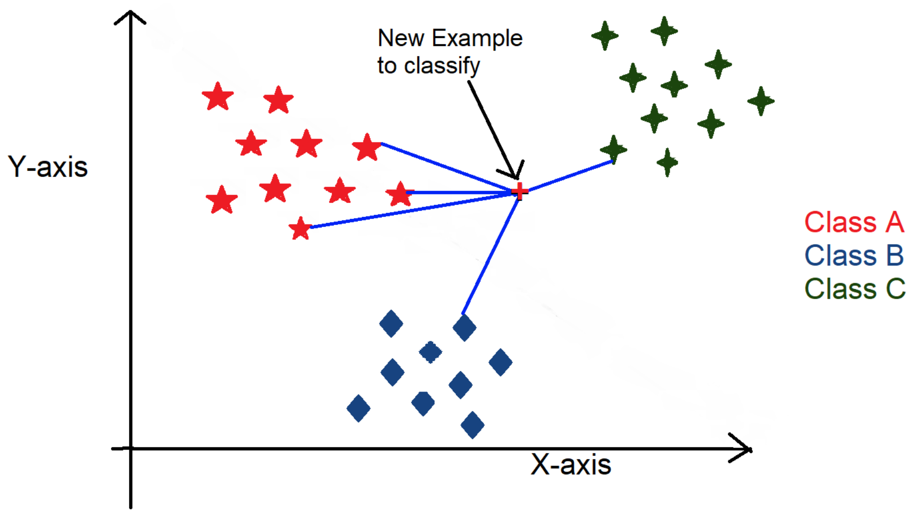

Figure 2.3: K-Nearest Neighbor

2.3.3 Random Forest

The supervised learning approach makes use of Random Forest, a well-known machine learning algorithm. It may be used in machine learning for both classification and regression. To address a particularly challenging assignment, the 10an ensemble learning method is used to improve the overall performance of the model by merging many classifiers. As the name indicates, Random Forest is a classifier that consists of several decision trees trained on distinct subsets of the input dataset and averages their predictions to increase the dataset's projected accuracy. Rather than using a single decision tree, the random forest takes the predictions from each tree and predicts the ultimate conclusion depending on the majority vote. By using a larger forest, we can increase accuracy while minimizing the danger of overfitting [6]. The random forest is built in the first stage by merging the N decision trees, and predictions are generated for each tree in the random forest in the second stage. The steps that will follow are as follows, coupled with an image to demonstrate how the approach works: The first step is to randomly choose K data points from the training set. Constructing decision trees based on the data points selected. Decide how many decision trees you want to generate at this step. Repetition of Steps 1 and 2 is the fourth step. Five: Find predictions for new data points in each decision tree and assign the new data to the category with the most votes [26].

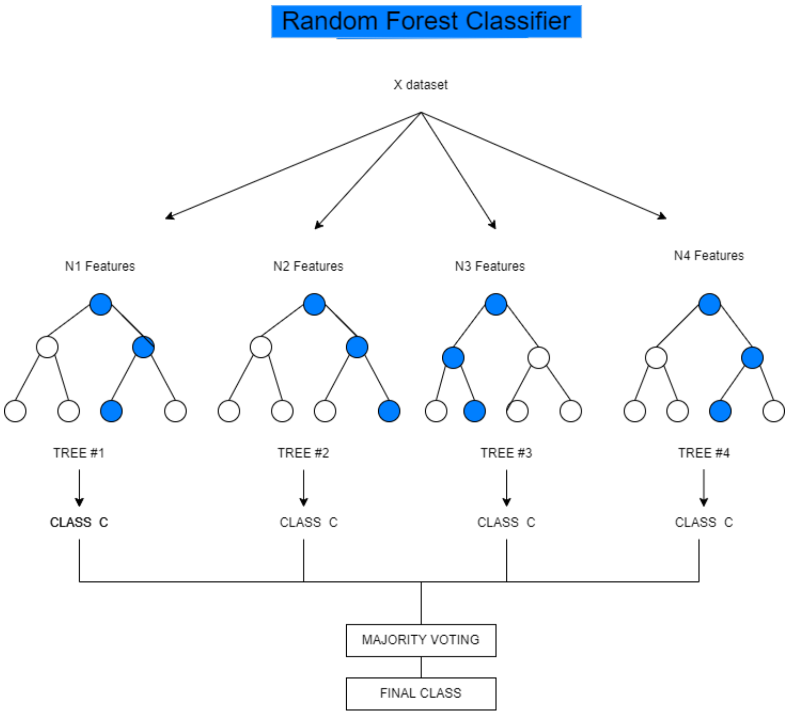

Figure 2.4: Random Forest Algorithm Working Procedure

2.3.4 Naive Bayes

The Naive Bayes algorithm is based on the Bayes theorem and assumes that each pair of features is independent of the others. In a variety of real-world applications, such as document or text classification, spam filtering, and so on, it may be used to provide binary or multi-class classification. The NB classifier is capable of identifying and removing noise from data so that it can provide accurate predictions. One of the key advantages is that, as compared to more sophisticated algorithms, it just takes a little amount of training data to properly estimate the crucial properties. However, because of its strict feature independence assumptions, its performance may suffer. Gaussian, Multinomial, Complement, Bernoulli, and Categorical are the most popular NB classifier variants. Discriminant Analysis: Bayes' Rule is used to fit class densities to datasets using linear discriminant analysis (LDA). A higher-dimensional dataset might be referred to as an extension of Fisher's linear discriminant since it reduces complexity or processing costs. The LDA model provides a Gaussian density to each class based on the assumption that all classes have the same covariance matrix. Regression and ANOVA are both methods that aim to depict a dependent variable as an ordered linear mixture of the independent variables' attributes or measurements [9][42].

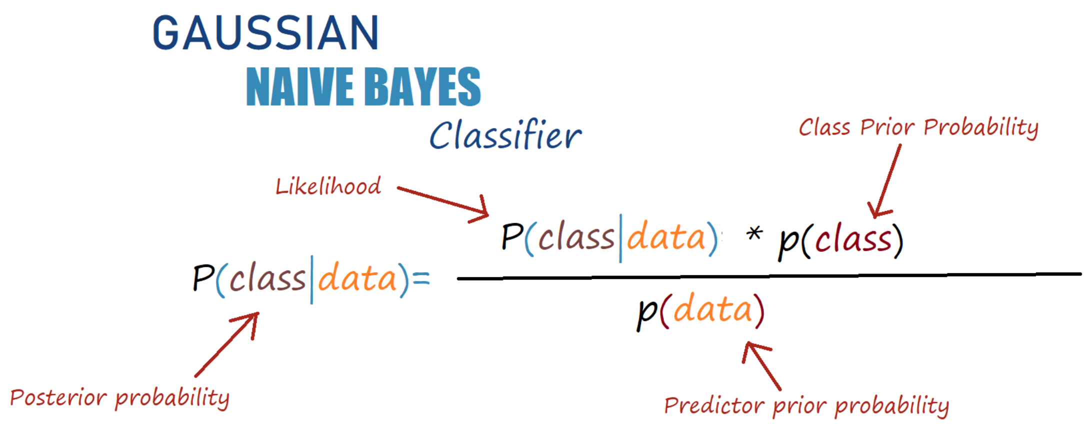

Figure 2.5: Naive Bayes Algorithm Working Procedure

2.3.5 Decision Tree

Decision Trees are a type of Supervised Learning that may solve classification and regression problems but are most commonly used to solve classification problems. It is a tree-structured classifier with core nodes containing dataset properties, branches containing decision rules, and leaf nodes containing the outcome. A decision tree is made up of two nodes: the Decision Node and the Leaf Node. Choice nodes are used to make decisions and have several branches, whereas Leaf nodes are the results of those decisions and have no more branches [41]. The dataset's attributes are used to conduct the evaluations or tests. When you ask it a question, it generates a mental image of all possible answers to help you choose one. It is called a decision tree because, like a tree, it begins with the root node and expands in a tree-like form via consecutive branches. The CART method is used to construct a tree. The CART algorithm is abbreviated as Classification and Regression Tree. A decision tree is just a tree that asks a question and then branches into subtrees based on the response. Because there are so many various types of machine learning algorithms, it is critical to select the optimal one for the dataset and problem at hand while developing a machine learning model [4]. Decision trees are frequently designed to imitate human decision-making abilities, making them simple to comprehend. The rationale of the decision tree is easily grasped due to its tree-like form [41].

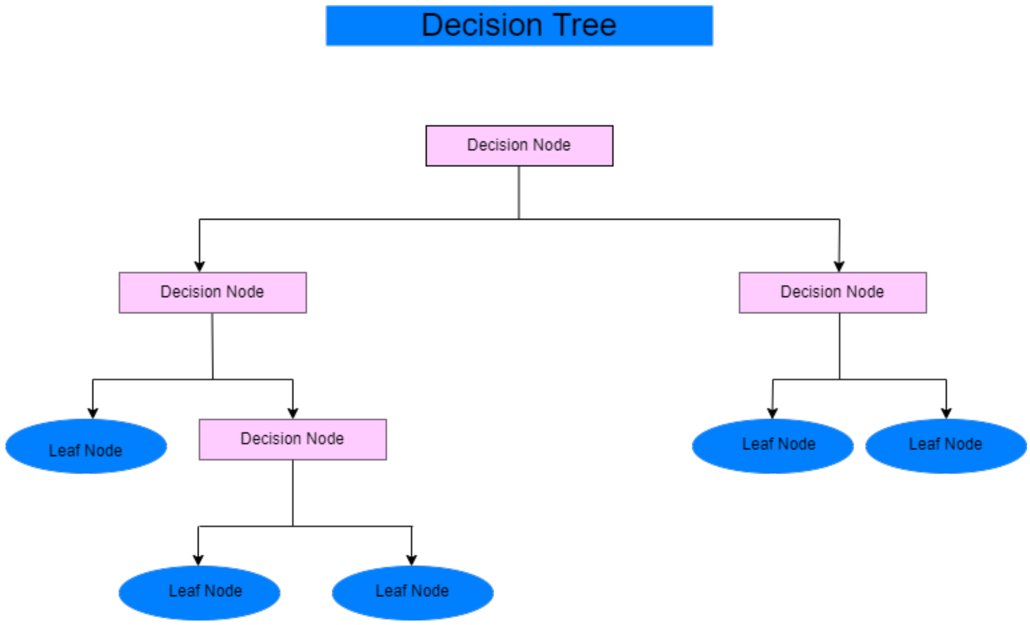

Figure 2.6: Decision Tree Working Procedure

2.3.6 Multilayer Perceptron

A case of a feed-forward neural network is a multilayer perceptron, and each layer of MLP is a linked layer. Three layers are accessible in MLP - input, output, and hidden layer. The input layer accepts inputs that are to be processed. The output layer does categorization and prediction. Hidden layers exist between the input and output layers. The number of buried layers is unpredictable. The feed-forward network direction of data flow is from the input layer towards the output, in the same way as data flows in MLP. Using a backpropagation learning model, MLP becomes learned. Problems that are not individual linearly receive solutions with MLP. Nevertheless, primarily MLP is used for recognition, classification difficulties, and forecasting and approximation [1].

2.4 Used Device

2.4.1 Raspberry Pi

The Raspberry Pi Foundation collaborated with Broadcom to create a compact, single-board computer known as a Raspberry Pi. The Raspberry Pi was created in order to teach basic computer science concepts to students. Due to the cheap cost, modular, and open design of its components (modularity and openness), it has become more popular in a wide range of applications. In addition to using a mouse and keyboard, the Raspberry Pi may be connected to a television or computer display. The Raspberry Pi can do all of the functions of a desktop computer, including accessing the web, creating spreadsheets, and playing music and video [14]. Learning coding, computer science, and building hardware and industrial projects like home automation and implementing edge computing are just some of the uses for this useful technology. There is a range of projects that make use of Raspberry Pi's interactivity and interactivity with the outside world.

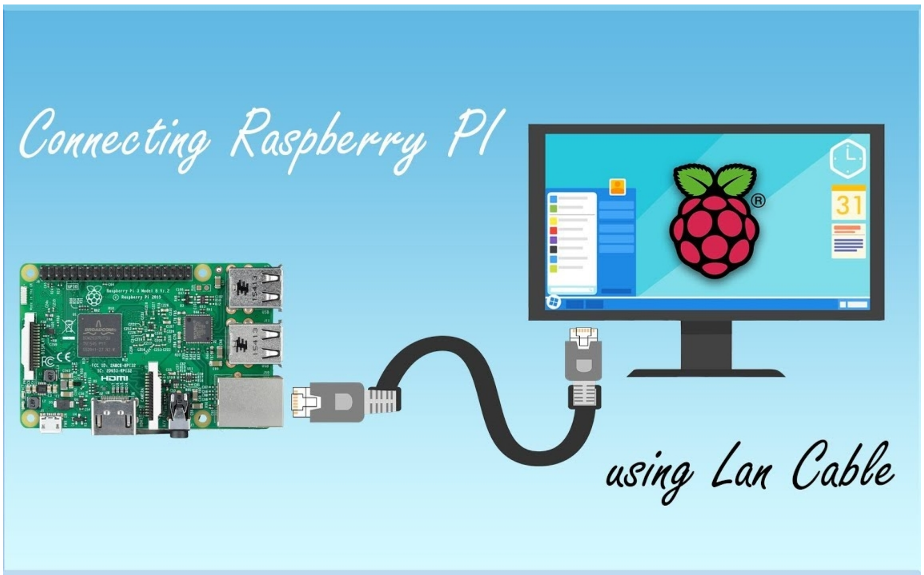

Figure 2.7: Connecting Raspberry Pi[11]

2.4.2 EEG Headset

In real-world contexts, the DSI wireless EEG headsets are built for speed and convenience of use, comfort, and mobility. The use of EEG in research, neurofeedback, brain-computer interfaces, brain age prediction, and other applications is being revolutionized by this world-leading dry electrode technology.

Figure 2.8: Wearable EEG Headset[40]

Chapter 3

Methodology

3.1 Proposed Approach

The aim of our research is to investigate age prediction in numerous ages. so as to try and do the research, we've got to extract data from Kaggle initially so to analyze how different types of electroencephalogram signals will establish human brain age. We need a sufficient dataset for our prediction to perform different algorithms. On the basis of different events' effects on the EEG signal, we have to process the dataset. After processing the dataset, EEG feature extraction will be required. EEG feature is a stage in which we look for common features in EEG samples. Then in feature reduction, comparing some features that are not very different in variations or highly linked with other features, some features will be eliminated. Next, to find out the better performing algorithm we need to apply it to predict age. And finally, we have to compare the results of algorithms for the best algorithm. when preprocessing the information we want to label the datasets into different train and check sets. we are going to use the plaything to coach our model victimization in different supervised learning algorithms and realizing the most effective technique by testing the accuracy of our model. Finally, for the implementation part, we will use raspberry pi. We will deploy our training model in raspberry pi. Will make thresholds for younger, middle, and older age. Taking EEG signals as input from the dataset and our model will detect age class spontaneously.

The entire process can be divided into 3 steps and they are described below -
1. Data preprocessing: In this stage, we format the data from Kaggle, and we conjointly format the electroencephalogram signal data so as to arrange the datasets for more analysis.
2. Training the model: In this degree, we educate our version on the use of diverse supervised gadget mastering algorithms.
3. Testing the model: In this level,We will select the best algorithm for the Raspberry Pi, install it, and test the model using our data. .

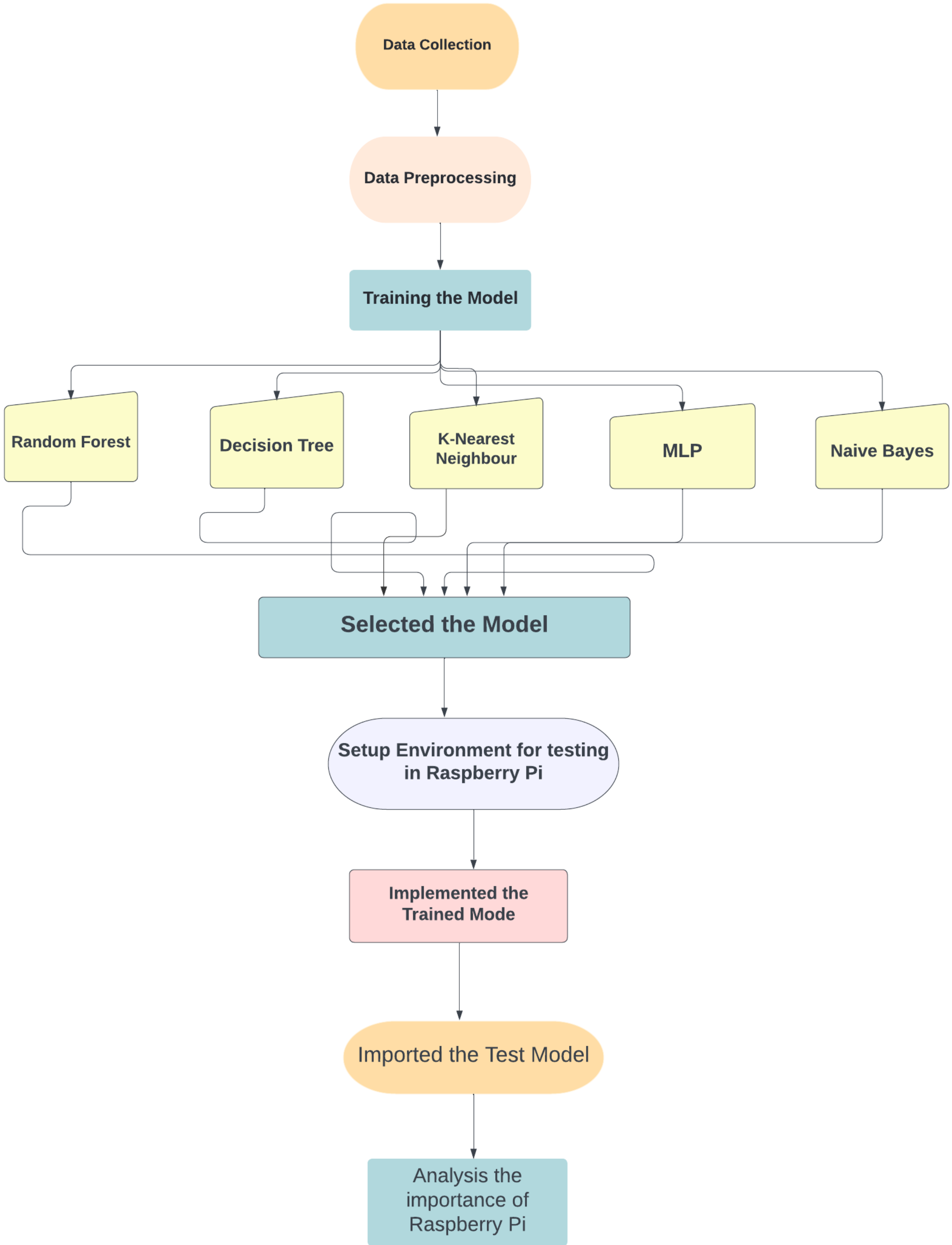

Figure 3.1: The Flowchart of the Proposed Research

3.2 Dataset

We studied the "TUH Abnormal EEG Corpus" dataset for our dissertation. The original dataset is separated into normal and abnormal subsets. The primal purpose of the original dataset is to distinguish between normal and abnormal EEG signals. We, however, are interested in studying normal EEG alone. The purpose of our study is to predict a patient's age from the person's normal EEG signal. Therefore we utilize the dataset containing EEG signals of normal patients only. We used different Machine Learning algorithms such as Decision Trees, Random Forests, K Nearest Neighbors, Logistic Regression, Gaussian Naive Bayes, etc.

Source

The dataset was downloaded from this link: https://www.kaggle.com/datasets/ayurgo/data EEG-age-v1. There are two folders—train (1171 files) and eval (126 files) inside the data EEG agev1 parent folder. The dataset provided in the link contains the data from normal EEG signals exclusively.

Data Conversion

The original data consists of EEG signal in .edf format and the patient's information in .txt format. It was then converted from .edf to .csv for patients having age information, embedding the age into the corresponding .csv files.

Dataset Description

The data contains 1297 patients' recordings of slightly various lengths. The datasets from the train folder were first merged and then split into 80% training and 20% test data. In each folder there are .csv files where the first row contains the age, the second row contains the EEG channels' names used in the recording, and the signal data follows from the third row. The dataset contained the following columns or variables which are also illustrated in Figure 3.7 below: 'EEG FPl REF', 'EEG FP2-REF', 'EEG F3-REF', 'EEG F4-REF', 'EEG C3-REF', 'EEG C4-REF', 'EEG P3-REF', 'EEG P4-REF', 'EEG O1-REF', 'EEG O2-REF', 'EEG F7-REF', 'EEG F8-REF', 'EEG T3-REF', 'EEG T4-REF', 'EEG TS-REF', 'EEG T6-REF', 'EEG Al-REF', 'EEG A2-REF', 'EEG FZ-REF', 'EEG CZ-REF', 'EEG PZ-REF', 'EEG ROC-REF', 'EEG LOC-REF', 'EEG EKGl-REF', 'EMG-REF', 'EEG 26-REF', 'EEG 27-REF', ' EEG 28-REF', 'EEG 29-REF', 'EEG 30-REF', 'EEG Tl-REF', 'EEG T2-REF', 'IBI', 'BURSTS', 'SUPPR', 'Age'.

In [49]:

```
all_ds.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 599997 entries, 0 to 599999
Data columns (total 36 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   EEG FP1-REF   599997 non-null  float64
 1   EEG FP2-REF   599997 non-null  float64
 2   EEG F3-REF    599997 non-null  float64
 3   EEG F4-REF    599997 non-null  float64
 4   EEG C3-REF    599997 non-null  float64
 5   EEG C4-REF    599997 non-null  float64
 6   EEG P3-REF    599997 non-null  float64
 7   EEG P4-REF    599997 non-null  float64
 8   EEG O1-REF    599997 non-null  float64
 9   EEG O2-REF    599997 non-null  float64
 10  EEG F7-REF    599997 non-null  float64
 11  EEG F8-REF    599997 non-null  float64
 12  EEG T3-REF    599997 non-null  float64
 13  EEG T4-REF    599997 non-null  float64
 14  EEG T5-REF    599997 non-null  float64
 15  EEG T6-REF    599997 non-null  float64
 16  EEG A1-REF    599997 non-null  float64
 17  EEG A2-REF    599997 non-null  float64
 18  EEG FZ-REF    599997 non-null  float64
 19  EEG CZ-REF    599997 non-null  float64
 20  EEG PZ-REF    599997 non-null  float64
 21  EEG ROC-REF   599997 non-null  float64
 22  EEG LOC-REF   599997 non-null  float64
 23  EEG EKG1-REF  599997 non-null  float64
 24  EMG-REF       599997 non-null  float64
 25  EEG 26-REF    599997 non-null  float64
 26  EEG 27-REF    599997 non-null  float64
 27  EEG 28-REF    599997 non-null  float64
 28  EEG 29-REF    599997 non-null  float64
 29  EEG 30-REF    599997 non-null  float64
 30  EEG T1-REF    599997 non-null  float64
 31  EEG T2-REF    599997 non-null  float64
 32  IBI           599997 non-null  float64
 33  BURSTS        599997 non-null  float64
 34  SUPPR         599997 non-null  float64
 35  Age           599997 non-null  int64
dtypes: float64(35), int64(1)
memory usage: 169.4 MB
```

Figure 3.2: Various columns of the dataset including their types

3.3 Data Pre-processing

Sample Dataset

00000021_s004_t000	8/21/2020 5:27 AM	Microsoft Excel C...	69,292 KB
00000039_s005_t000	8/21/2020 5:28 AM	Microsoft Excel C...	86,757 KB
00000296_s002_t000	8/21/2020 5:28 AM	Microsoft Excel C...	78,697 KB
00000312_s002_t000	8/21/2020 5:28 AM	Microsoft Excel C...	80,454 KB
00000355_s003_t000	8/21/2020 5:28 AM	Microsoft Excel C...	84,643 KB
00000361_s003_t001	8/21/2020 5:28 AM	Microsoft Excel C...	75,559 KB
00000407_s002_t001	8/21/2020 5:29 AM	Microsoft Excel C...	76,619 KB
00000417_s004_t000	8/21/2020 5:29 AM	Microsoft Excel C...	94,157 KB
00000436_s002_t001	8/21/2020 5:29 AM	Microsoft Excel C...	81,599 KB
00000466_s002_t000	8/21/2020 5:29 AM	Microsoft Excel C...	85,189 KB
00000466_s003_t000	8/21/2020 5:29 AM	Microsoft Excel C...	90,349 KB
00000564_s002_t001	8/21/2020 5:30 AM	Microsoft Excel C...	72,959 KB
00000592_s004_t000	8/21/2020 5:30 AM	Microsoft Excel C...	89,194 KB
00000592_s005_t000	8/21/2020 5:30 AM	Microsoft Excel C...	79,308 KB
00000649_s003_t000	8/21/2020 5:30 AM	Microsoft Excel C...	74,180 KB
00000739_s002_t000	8/21/2020 5:30 AM	Microsoft Excel C...	70,559 KB
00000870_s009_t000	8/21/2020 5:30 AM	Microsoft Excel C...	64,393 KB
00000870_s010_t001	8/21/2020 5:31 AM	Microsoft Excel C...	75,643 KB
00000883_s015_t001	8/21/2020 5:31 AM	Microsoft Excel C...	72,600 KB
00000894_s004_t001	8/21/2020 5:31 AM	Microsoft Excel C...	81,061 KB
00000929_s003_t002	8/21/2020 5:31 AM	Microsoft Excel C...	72,037 KB
00000929_s005_t000	8/21/2020 5:31 AM	Microsoft Excel C...	66,316 KB

Figure 3.3: Sample Dataset from Kaggle

Merging the dataset

We need to reduce the size of our dataset since it might reach 100 GB or more in size. It was decided to reduce the dataset by using a separate csv file for each age group. We randomly chose roughly 20,000 rows from each csv file, using the sample function. Our goal column, on the other hand, has been set up. As stated in the first row of the csv file, based on the specified age number. As a result, the row has been deleted from the dataset and replaced with a target column, and the target value has been set for each one. We must combine all of the various datasets since we must labor for 50 years.

	EEG FP1-REF	EEG FP2-REF	EEG F3-REF	EEG F4-REF	EEG C3-REF	EEG C4-REF	EEG P3-REF	EEG P4-REF	EEG O1-REF	EEG O2-REF	...	EEG 28-REF	EEG 29-REF	EEG 30-REF	EEG T1-REF	EEG T2-REF	PHOTIC-REF	IBI	BUF
0	-4.421	-17.544	-12.204	-32.650	8.396	-18.612	43.949	4.886	-2.285	5.649	...	-28.378	9.464	-1.217	-17.849	-13.882	0.0	0.0	
1	3.360	33.878	19.382	8.854	45.017	1.224	45.017	5.191	20.145	24.112	...	36.167	37.845	17.704	32.047	22.892	0.0	0.0	
2	-47.757	-28.836	-17.239	-2.591	20.298	-7.016	31.894	3.360	-0.912	7.022	...	-8.541	50.968	15.262	-13.577	-12.814	0.0	0.0	
3	-33.413	-15.560	17.093	25.333	-17.392	7.785	-46.078	-8.694	-4.116	-17.239	...	-32.498	-33.871	-5.032	3.971	-4.879	0.0	0.0	
4	24.112	17.704	5.954	15.262	19.077	7.633	-10.067	11.448	-2.743	6.412	...	41.812	0.614	-0.607	14.652	22.892	0.0	0.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
599995	27.012	16.483	13.126	0.004	0.766	-4.727	-3.048	0.614	-7.473	-4.879	...	16.483	4.428	2.598	0.766	-11.288	0.0	0.0	
599996	22.281	24.723	1.377	0.766	-10.983	-11.135	-18.917	-18.765	-21.206	-22.885	...	-3.811	-7.779	-37.533	-2.591	-4.116	0.0	0.0	
599997	-18.460	-25.784	2.903	-3.811	9.006	4.734	4.123	8.091	4.734	3.055	...	17.398	-30.667	-4.116	26.859	3.818	0.0	0.0	4300
599998	-3.659	-5.337	-4.269	-0.912	12.973	2.445	21.671	9.159	17.551	14.347	...	37.388	24.265	-18.917	8.243	-6.100	0.0	0.0	
599999	8.243	12.668	-2.591	1.835	-0.759	-5.184	-13.272	-18.917	-15.560	-17.239	...	-5.795	28.232	-9.457	-16.476	-8.236	0.0	0.0	

600000 rows × 37 columns

Figure 3.4: After mergeing the dataset

Cleaning and Scaling Data

We imported the dataset first, then processed it for future usage. Then we looked for eliminated any zero or null values using is null function. With the help of data analyzing phase such heat map, box plot since The "PHOTIC-REF' column was removed since practically all of the entries were zero-valued. In addition, there were 36 columns. The first 35 columns were labeled as features and the last one as the goal. Then we looked for and eliminated any duplicate values. The first 35 features were of data type float64, whereas the target was of data type int64.

Null value

```
EEG FP1-REF     3
EEG FP2-REF     3
EEG F3-REF      3
EEG F4-REF      3
EEG C3-REF      3
EEG C4-REF      3
EEG P3-REF      3
EEG P4-REF      3
EEG O1-REF      3
EEG O2-REF      3
EEG F7-REF      3
EEG F8-REF      3
EEG T3-REF      3
EEG T4-REF      3
EEG T5-REF      3
EEG T6-REF      3
EEG A1-REF      3
EEG A2-REF      3
EEG FZ-REF      3
EEG CZ-REF      3
EEG PZ-REF      3
EEG ROC-REF     3
EEG LOC-REF     3
EEG EKG1-REF    3
EMG-REF         3
EEG 26-REF      3
EEG 27-REF      3
EEG 28-REF      3
EEG 29-REF      3
EEG 30-REF      3
EEG T1-REF      3
EEG T2-REF      3
PHOTIC-REF      3
IBI             3
BURSTS          3
SUPPR           3
Age             0
dtype: int64
```

Figure 3.5: Null value

For cleaning these null value we used is null function

Standardizing the Data

we used StandardScaler to scale the dataset and modify it so that its distribution has a mean value of 0 and a standard deviation of 1.

Data Classification

In our study, we split the dataset in an 6:2:2 ratio, with 60% of the data used to train the model, 20% used as a test set, and the remaining 20% used to verify our models.

Chapter 4

Implementation and Results

4.1 Workplan

Based on EEG readings, our study aims to use machine learning techniques to estimate the age of the human brain. Here, we'll go through the model's implementation and the outcomes we've gotten as a consequence of the model. Our model was trained using Google Colab and Visual Studio code and ready the model for Raspberry pi.

4.1.1 Applying Algorithms

We made use of five different classification methods while we were in the training phase of the dataset. These algorithms are the Random Forest Algorithm, the K-NN Algorithm, the Naive Bayes Theorem, and the Decision Tree Algorithm. In order to improve the reliability of our classification algorithms, we decided to make use of the criteria that are both the most often used and the most widely acknowledged. In addition to the F1 score, our models make use of the accuracy and precision matrices in order to evaluate how well different machine learning approaches perform. As a first step in the categorizing process, we separated our data into two distinct groups. When the model was being evaluated, just 20 percent of the data that was used to train it was taken into consideration. Eighty percent of our data was used in the process of training. We performed training sessions using a strategy known as test train split, during which we practiced putting each of the five different strategies into practice. Our demonstration train is comprised of two sets, each consisting of eight individual carriages. A number of different algorithms' accuracy scores are presented for your consideration in Table 4.1 below. Different Algorithms.

4.2 Selecting Trained Model Best on Performance for Raspberry Pi

4.2.1 Random Forest Classifier

Age prediction is going to be covered in this part utilizing a variety of different ML algorithms and EEG signal data. In order to get our dataset ready, we used a range of different approaches to data analysis. When working with categorized data, accuracy is of the utmost importance; hence, we started with the Random forest classifier.[15]

Classification Report and Confusion of Random Forest

We can see in fig 4.1 the values of precision, recall, and f1-score in the table from the report. When all forecasts are correct, precision is the number of cases in which all predictions are correct. At the age of 21, we can achieve a level of accuracy of 85%. After then, recall is defined as the proportion of those who say "yes" to the question. For those under 21, our recall rate is 89 percent . A rate of 87 percent for the same age group obtained an F1 score. Regarding classifier accuracy and recall, the value provided by f1 is the harmonic mean. Recall and accuracy are the focus of the F1 score. The relative usefulness of two separate classifiers is valued herein. Precision, f1-score, and percent come in at 100% and 99percent, respectively, at 22. We get 99% for all three at the age of 23. Precision, recall, and f1-score are all over 98% for people in their mid-twenties, which is very accurate. Precision is 47percent, recall is 54percent, and the f1-score is 50percent for a 25-year-old. We use this method to estimate the worth of a person's life until they reach 70. The accuracy rate is 89percent, the recall rate is 88percent, and the f1-score rate is 90percent on average.

	precision	recall	f1-score
21	0.85	0.89	0.87
22	1.00	0.99	1.00
23	0.99	0.99	0.99
24	0.99	0.97	0.98
25	0.47	0.54	0.50
26	0.69	0.76	0.72
27	0.90	0.69	0.78
28	0.93	0.80	0.86
29	0.99	0.93	0.96
30	0.94	0.91	0.93
31	0.99	1.00	0.99
32	0.99	0.95	0.97
33	1.00	0.98	0.99
34	0.99	0.96	0.98
35	0.84	0.92	0.88
36	0.63	0.80	0.70
37	0.59	0.61	0.60
38	0.56	0.62	0.59
39	0.97	0.72	0.83
40	0.99	1.00	0.99
41	0.97	0.85	0.91
42	1.00	1.00	1.00
43	1.00	1.00	1.00
44	0.96	0.92	0.94
45	0.72	0.50	0.59
46	0.55	0.70	0.62
47	1.00	0.94	0.97
48	0.99	0.98	0.99
49	0.99	0.97	0.98
50	1.00	0.95	0.97
51	1.00	1.00	1.00
52	0.99	1.00	0.99
53	0.98	0.99	0.99
54	0.99	1.00	0.99
55	0.96	0.99	0.97
56	0.98	1.00	0.99
57	0.87	0.97	0.92
58	0.92	0.97	0.94
59	0.98	1.00	0.99
60	0.97	1.00	0.98
61	0.62	0.59	0.61
62	0.97	0.93	0.95
63	0.97	0.83	0.89
64	0.93	0.89	0.91
65	0.98	0.96	0.97
66	1.00	0.92	0.96
67	0.63	0.63	0.63
68	0.99	0.97	0.98
69	0.97	0.82	0.89
70	0.54	0.67	0.60
accuracy			0.90
macro avg	0.89	0.88	0.88
weighted avg	0.91	0.90	0.90

Figure 4.1: Implementation in Random Forest

4.2.2 Confusion Matrix

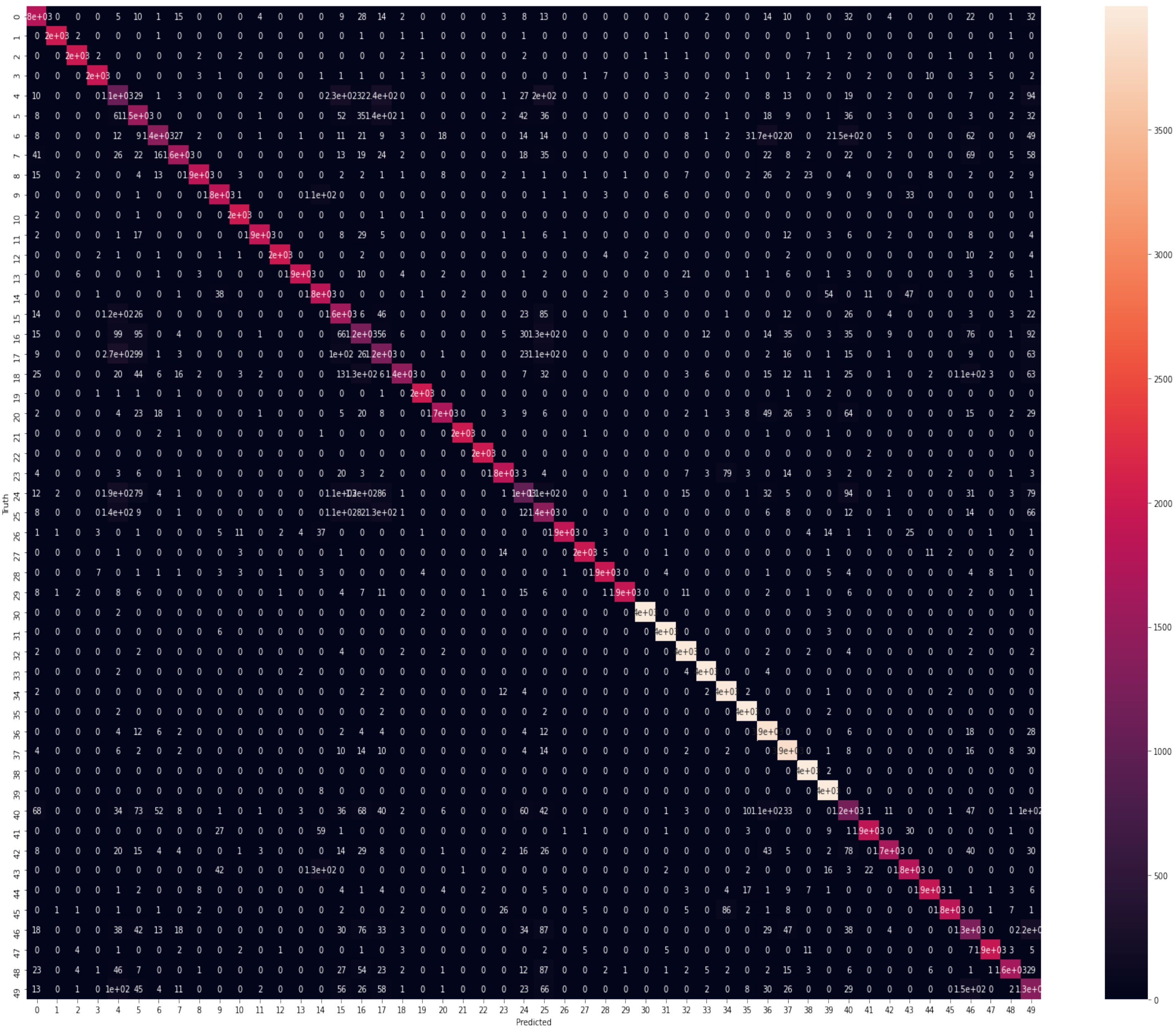

Figure 4.2: Confusion Matrix of Random Forest Classifier

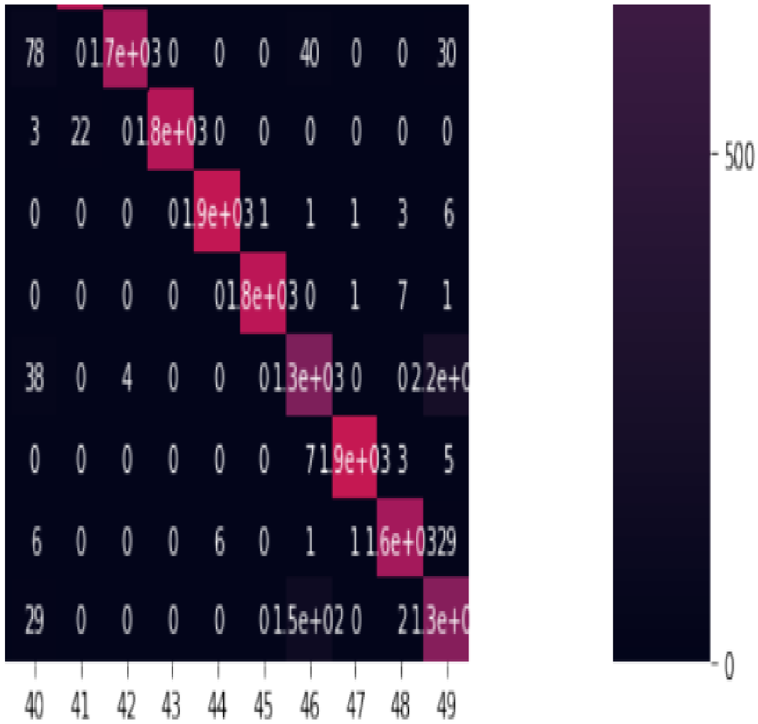

Figure 4.3: Sample of Confusion Matrix

As we are showing the confusion matrix for random forest 50 ages, it is quite hard to analyze all ages from one confusion matrix. That's why we have taken a portion from the whole picture of the confusion matrix. As we have done this confusion matrix for Random Forest Classifier, from this confusion matrix we can easily understand the performance of our model. It will help- us to visualize and summarize the performance of our model. From our picture we can see that for age 65(In confusion matrix the number 44), gives 1.9e+03 times the actual age 65 but it gives one time 66,one time 67,one time 68, three times 69 and six times 70. For age 66(In confusion matrix the number 45), gives 1.8e+03 times the actual age 65 but it gives four times 63, one time 68, seven times 69 and one times 70.

ROC Curve of Random Forest

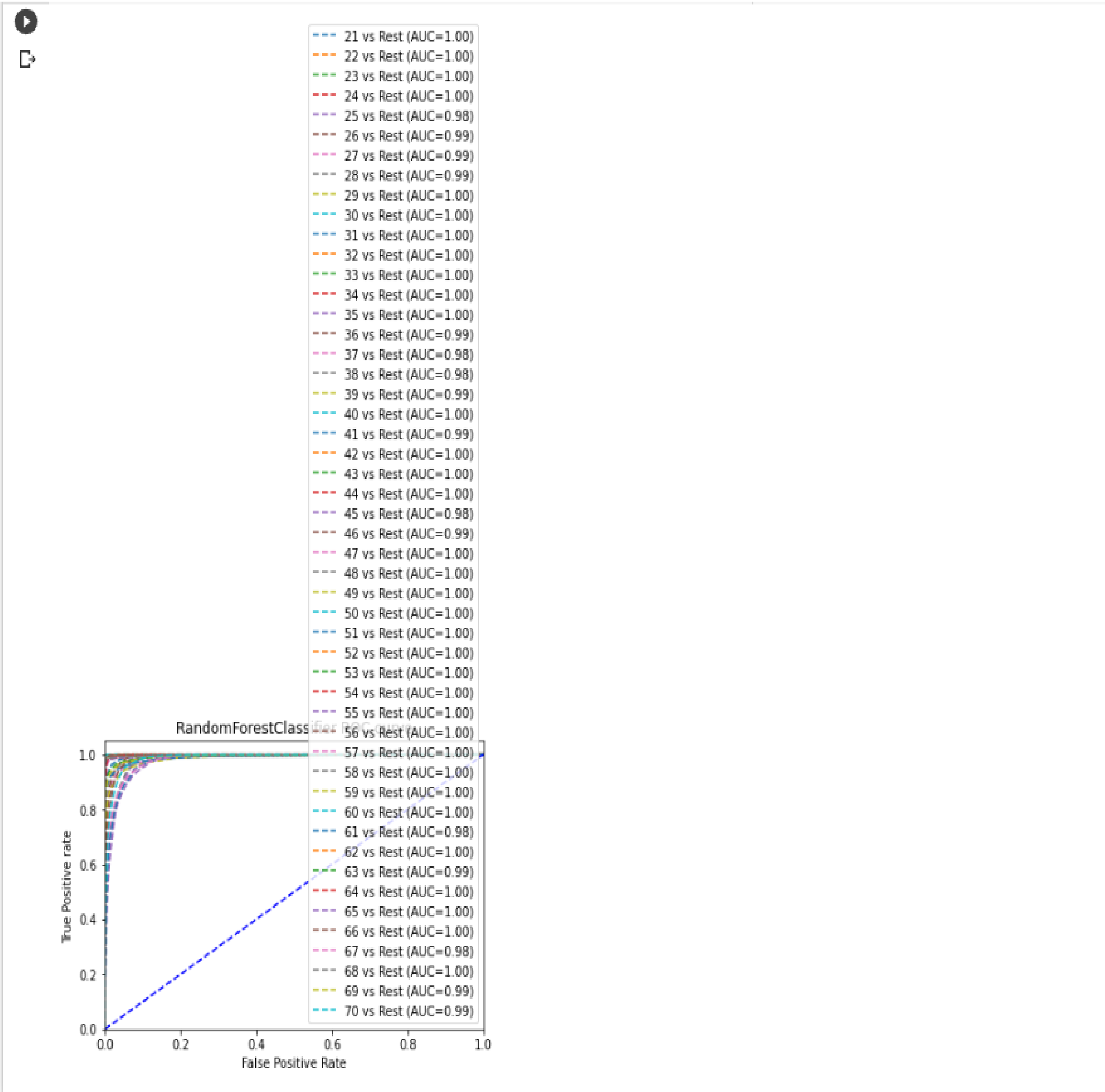

Figure 4.4: ROC Curve of Random Forest

Each individual age forecast may be readily shown by using the ROC curve in Decision Tree software. For example, we can see from the roc curve that ages 59 and 60 both provide 1.00 accuracy, age 63 provides.99 accuracy etc.

4.2.3 Decision Tree Classifier

Classification Report of Decision Tree:

	precision	recall	f1-score	support
21	0.69	0.62	0.66	2000
22	0.99	0.99	0.99	2000
23	0.98	0.97	0.97	2000
24	0.98	0.97	0.97	2000
25	0.30	0.31	0.30	2000
26	0.50	0.45	0.47	2000
27	0.67	0.65	0.66	2000
28	0.70	0.66	0.68	2000
29	0.94	0.93	0.94	2000
30	0.87	0.87	0.87	2000
31	0.99	0.99	0.99	2000
32	0.94	0.93	0.94	2000
33	0.98	0.99	0.99	2000
34	0.98	0.96	0.97	2000
35	0.78	0.78	0.78	2000
36	0.44	0.44	0.44	2000
37	0.31	0.30	0.31	2000
38	0.37	0.37	0.37	2000
39	0.74	0.69	0.72	2000
40	0.98	0.98	0.98	2000
41	0.84	0.82	0.83	2000
42	1.00	1.00	1.00	2000
43	0.99	0.99	0.99	2000
44	0.93	0.91	0.92	2000
45	0.42	0.42	0.42	2000
46	0.42	0.41	0.42	2000
47	0.98	0.97	0.97	2000
48	0.98	0.98	0.98	2000
49	0.98	0.97	0.98	2000
50	0.94	0.94	0.94	2000
51	0.99	1.00	1.00	4000
52	0.99	1.00	0.99	4000
53	0.97	0.99	0.98	4000
54	0.96	0.98	0.97	4000
55	0.97	0.99	0.98	4000
56	0.98	0.99	0.99	4000
57	0.82	0.93	0.87	4000
58	0.84	0.94	0.88	4000
59	0.99	0.99	0.99	4000
60	0.97	0.99	0.98	4000
61	0.39	0.37	0.38	2000
62	0.89	0.88	0.89	2000
63	0.83	0.79	0.81	2000
64	0.83	0.81	0.82	2000
65	0.95	0.95	0.95	2000
66	0.97	0.96	0.97	2000
67	0.36	0.36	0.36	2000
68	0.97	0.96	0.97	2000
69	0.78	0.77	0.78	2000
70	0.35	0.34	0.35	2000
accuracy			0.83	120000
macro avg	0.81	0.80	0.81	120000
weighted avg	0.83	0.83	0.83	120000

Figure 4.5: Implementation in Decision Tree

In the table below Fig 4.5, we can see the accuracy, recall, and f1-score values from the report. Precision is the overall number of true positives if all forecasts are correct. We get 69 percent accuracy for age 21. Following that, the proportion of true yes outcomes is defined as recall, and recall correlates to genuine positives. For age 21, we have a recall of 62percent. And the f1-score we earned at a rate of 66 percent for the same age. f1 returns a result that is the harmonic mean of a classifier's precision and recall. The F1-score is centered on a single value of recall and accuracy. It considers the relative usefulness of two distinct classifiers. Then, at age 22, the accuracy, recall, and f1-score scores are all 99%. When we look at age 23, precision is 98% and recall and f1-score are both 97%. Age 24 has values of 98% accuracy, 97% recall, and f1-score. 30% accuracy, 31% recall, and 30% f1-score at age 25. This method calculates value up to the age of 70. And the typical values we acquire are 81% accuracy, 80% recall, and 81% f1-score.

4.2.4 Confusion Matrix

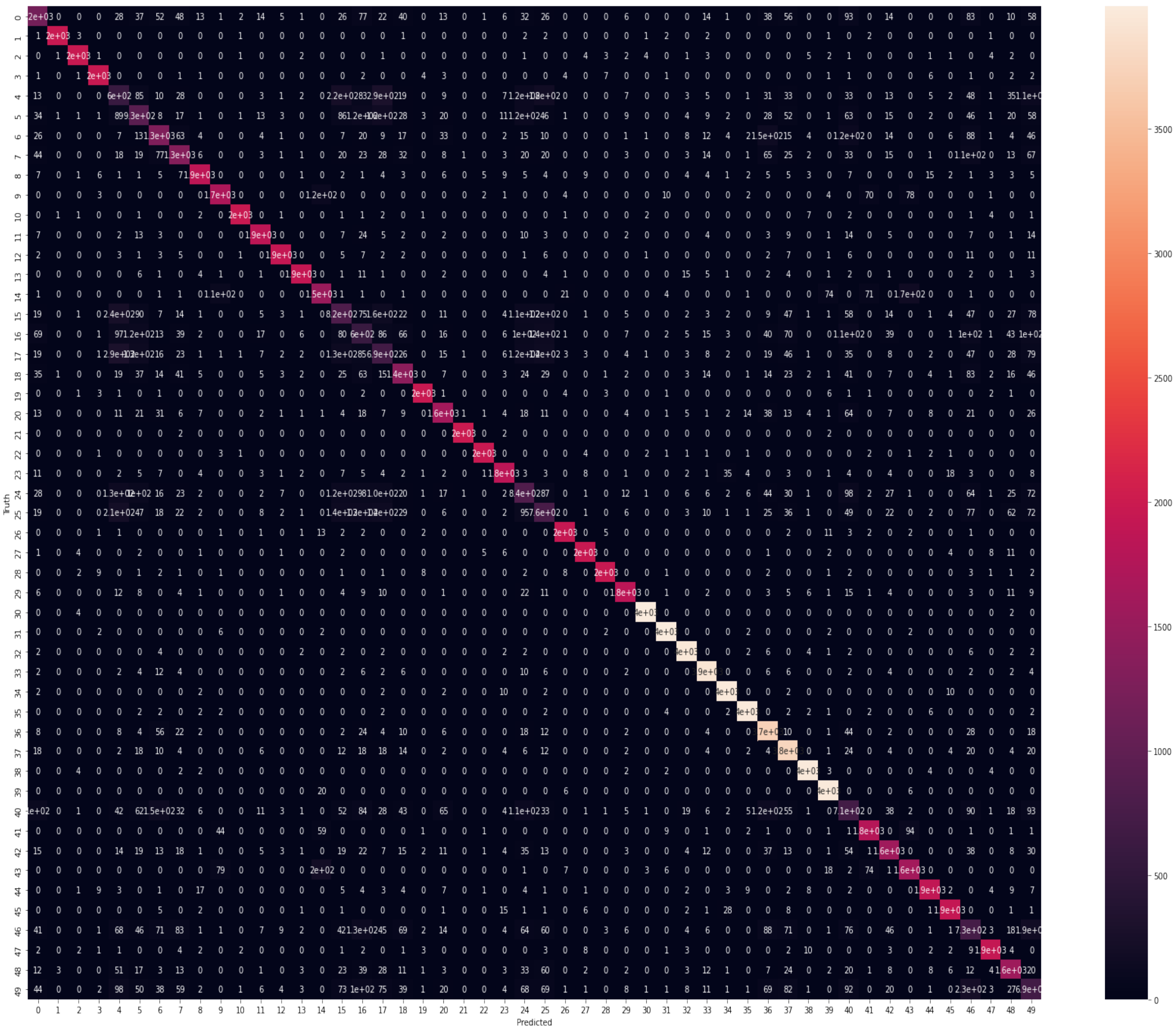

Figure 4.6: Confusion Matrix of Decision Tree

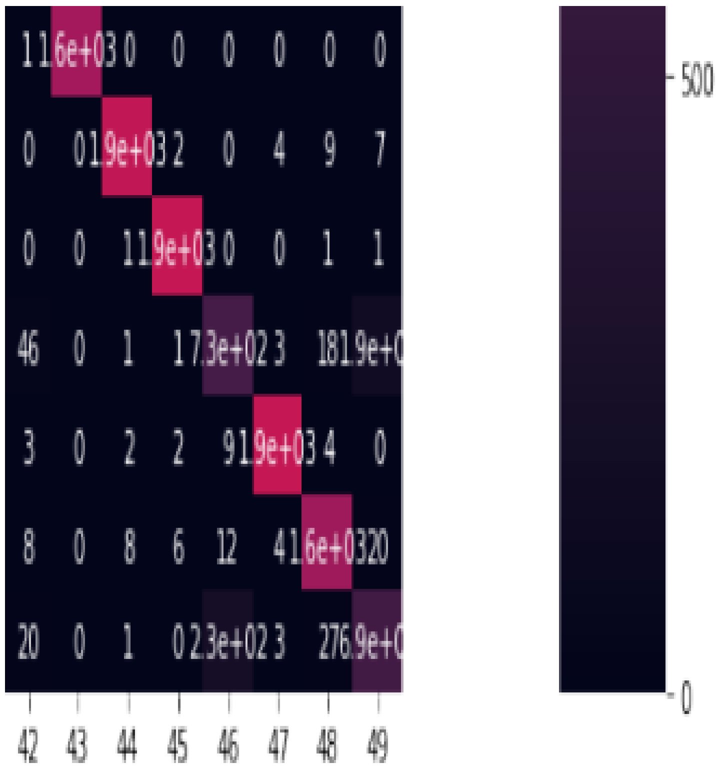

Figure 4.7: Sample of Confusion Matrix

It is rather difficult to do an analysis of all ages using only one confusion matrix due to the fact that we are displaying the confusion matrix for a decision tree that has 50 ages. Due to this reason, we have only extracted a small part of the overall image of the confusion matrix. Because we have constructed this confusion matrix for the decision tree classifier, we are now in a position to quickly comprehend the performance of our model by referring to this confusion matrix. It will be of great use to us in visualizing and analyzing the results of our model. From our picture we can see that for age 66(In confusion matrix the number 45), gives 1.9e+03 times the actual age 66 but it gives two time 66,four times 68,nine time 69, seven times 70. For age 68(In confusion matrix the number 47), gives 1.9e+03 times the actual age 68 but it gives four times 69.

ROC Curve of Decision Tree

We are able to readily see the forecast of each individual age by using the roc curve that corresponds to the Decision Tree. For instance, using the roc curve, we can see that ages 59 and 60 both offer an accuracy of 1.00, ages 57 and 61 both give an accuracy of 0.96, and age 61 gives the lowest accuracy of 0.67 etc.

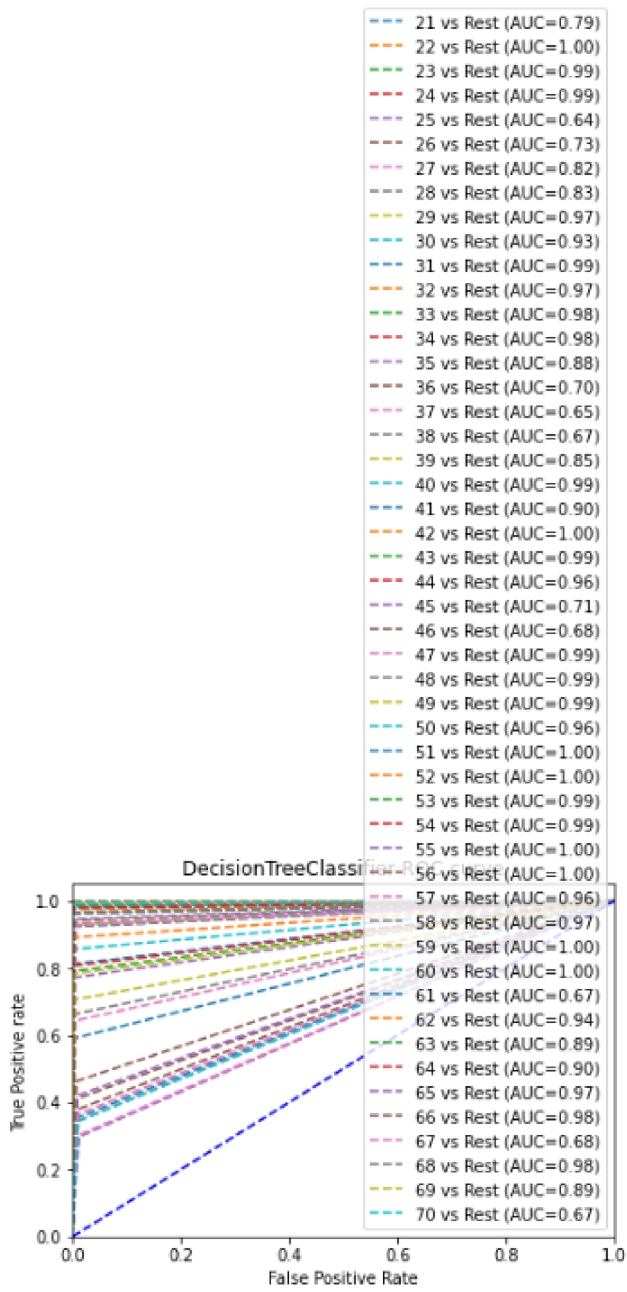

Figure 4.8: ROC Curve of Decision Tree

4.2.5 K-NN classifier

Classification Report of KNN: The accuracy, recall, and f1-score values in the table

	precision	recall	f1-score	support
21	0.91	0.58	0.71	2000
22	1.00	1.00	1.00	2000
23	1.00	1.00	1.00	2000
24	0.99	1.00	0.99	2000
25	0.31	0.39	0.35	2000
26	0.73	0.55	0.63	2000
27	0.75	0.72	0.74	2000
28	0.92	0.67	0.77	2000
29	1.00	0.98	0.99	2000
30	0.89	0.88	0.89	2000
31	1.00	1.00	1.00	2000
32	0.97	0.94	0.95	2000
33	0.99	0.99	0.99	2000
34	0.99	0.98	0.99	2000
35	0.83	0.78	0.81	2000
36	0.45	0.71	0.55	2000
37	0.46	0.38	0.42	2000
38	0.41	0.41	0.41	2000
39	0.90	0.74	0.81	2000
40	0.99	1.00	1.00	2000
41	0.88	0.89	0.88	2000
42	1.00	1.00	1.00	2000
43	1.00	1.00	1.00	2000
44	0.96	0.95	0.95	2000
45	0.54	0.52	0.53	2000
46	0.46	0.39	0.42	2000
47	1.00	0.96	0.98	2000
48	0.97	0.99	0.98	2000
49	1.00	0.99	0.99	2000
50	0.98	0.97	0.98	2000
51	1.00	1.00	1.00	4000
52	0.99	1.00	1.00	4000
53	0.99	1.00	1.00	4000
54	1.00	0.99	0.99	4000
55	0.99	0.99	0.99	4000
56	1.00	1.00	1.00	4000
57	0.89	0.94	0.92	4000
58	0.83	0.96	0.89	4000
59	1.00	1.00	1.00	4000
60	0.96	1.00	0.98	4000
61	0.56	0.36	0.44	2000
62	0.91	0.90	0.90	2000
63	0.89	0.87	0.88	2000
64	0.84	0.86	0.85	2000
65	0.99	0.99	0.99	2000
66	0.99	0.99	0.99	2000
67	0.42	0.53	0.47	2000
68	1.00	0.99	0.99	2000
69	0.82	0.80	0.81	2000
70	0.38	0.49	0.43	2000
accuracy			0.87	120000
macro avg	0.85	0.84	0.84	120000
weighted avg	0.87	0.87	0.87	120000

Figure 4.9: Implementation in K-Nearest Neighbor

may be found in the report Fig.4.7. Precision is the overall number of true positives if all forecasts are correct. We get 91% accuracy for age 21. Following that, the proportion of true yes outcomes is defined as recall, and recall correlates to genuine positives. For age 21, we find a recall of 58%. And the f1-score we earned at a rate of 71% for the same age. f1 returns a result that is the harmonic mean of a classifier's precision and recall. The F1-score is centered on a single value of recall and accuracy. It considers the relative usefulness of two distinct classifiers. Then, at age 22, the precision, recall, and f1-score scores are all 100%. If we look at age

23, the accuracy, recall, and f1-score are all 97%. Age 24 has values of 100% recall, 99% accuracy, and f1-score. For age 25, accuracy is 31%, recall is 39percent, and the f1-score is 35%. This method calculates value up to the age of 70. And the typical numbers we get are 85% accuracy, 84% recall, and 87% f1-score rate.

4.2.6 Confusion Matrix

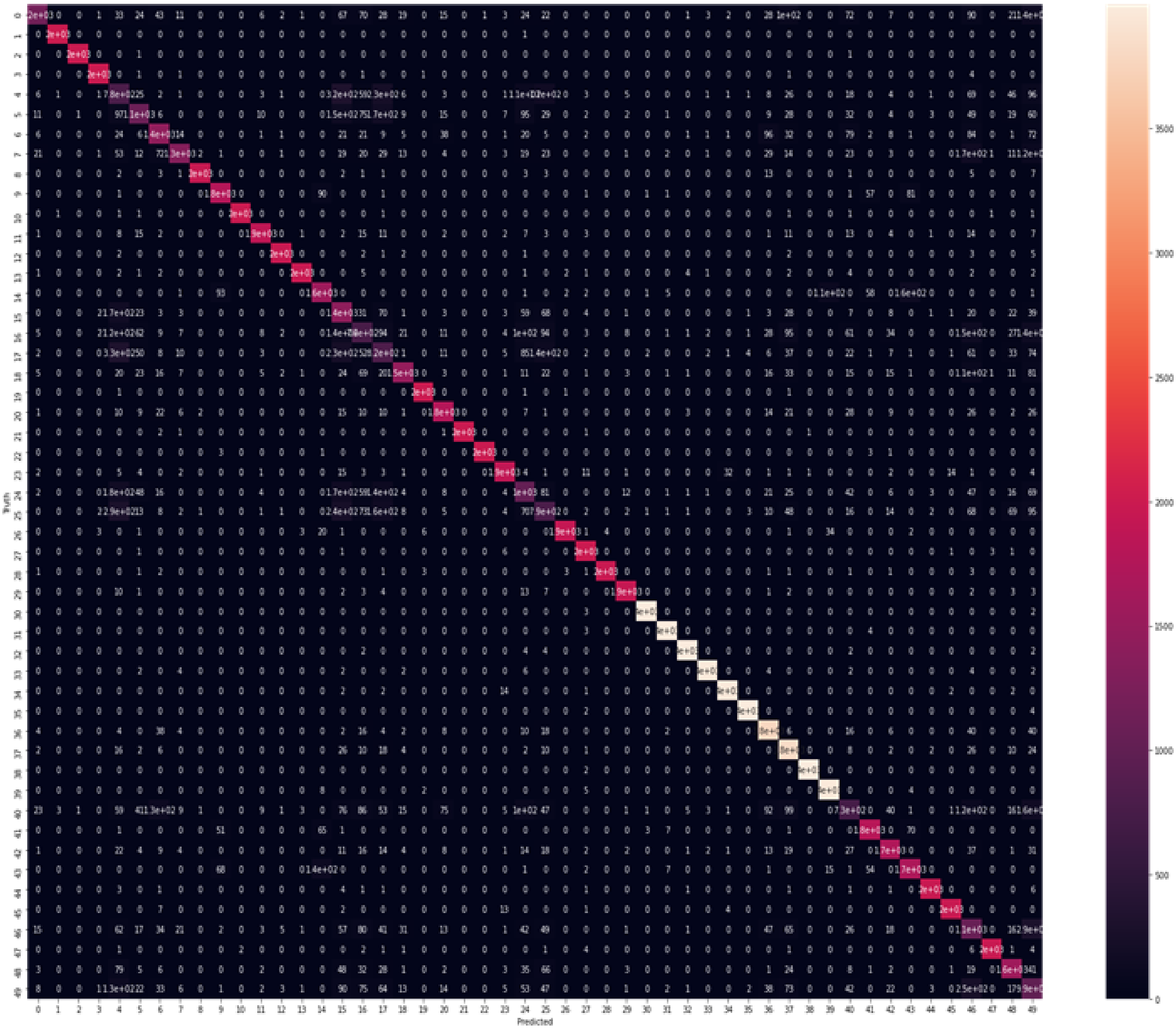

Figure 4.10: Confusion Matrix of K-NN

Portion of Confusion Matrix

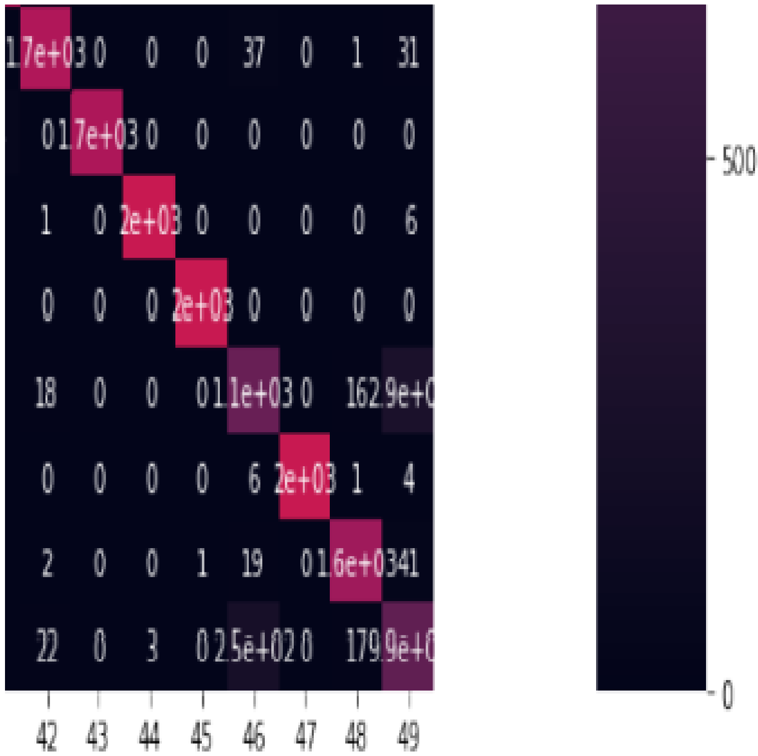

Figure 4.11: Sample of Confusion Matrix

Because we are showing the confusion matrix for a KNN with 50 ages, it is challenging to do an examination of all ages with only one confusion matrix. As a result, we were only able to capture a tiny portion of the whole confusion matrix picture. KNN model performance may now be simply analyzed using this KNN confusion matrix that has been built. We may utilize it to see and understand our model's output much better. From our picture we can see that for age 63(In confusion matrix the number 42), gives 1.7e+03 times the actual age 63 but it gives thirty seven time 67,one time 69,thirty one time 70. For age 68(In confusion matrix the number 47), gives 2e+03 times the actual age 68 but it gives one time 69 and four times 70.

ROC Curve of K-NN

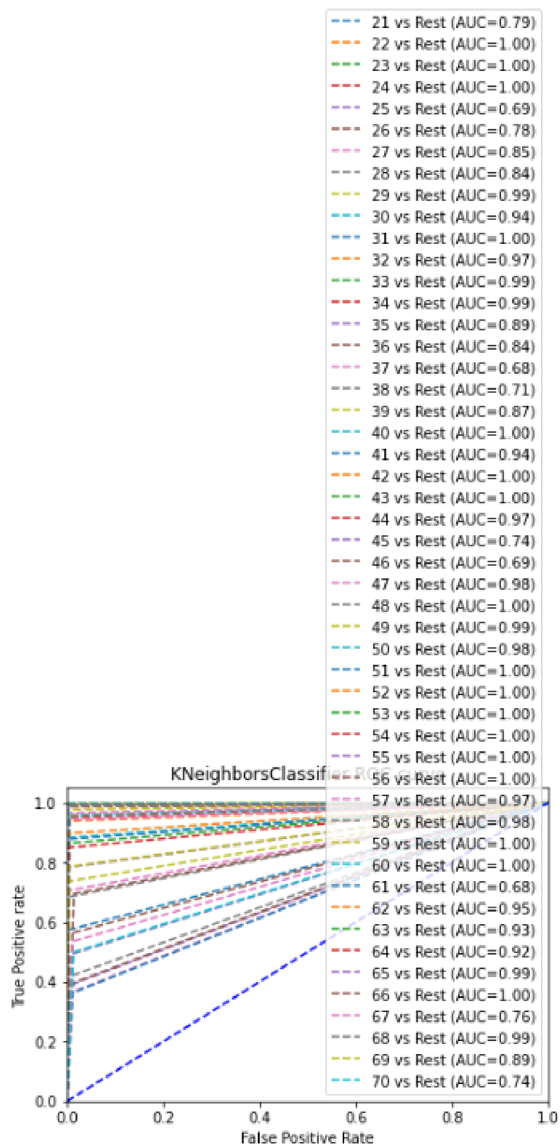

Figure 4.12: ROC Curve of K-NN

Using the roc curve for KNN we can easily visualize the prediction of each individual age.For example, using the roc curve we can see that the age 60 gives individually 1.00 accuracy ,Age 57 gives .97 accuracy ,0.68 accuracy comes from 37 etc.

Naive Bayes

Classification Report of Naive Bayes: We can see the values of precision, recall, and f1-score in the table from the report. The precision of a model is measured by the total number of predictions that come true or accurate positives. We can get a 22 percent level of accuracy for those who are 21 years old. After then, recall is defined as the proportion of those who say "yes" to the question. For those under 21, we receive a recall rate of 23 percent. We got an f1-score increase of 22 percent for the same age group. The harmonic mean of a classifier's accuracy and recall is measured by f1. The focus of the F1 score is on a single recall and accuracy value. A comparison of the two classifiers' utility is taken into account. The precision score drops to 40%, the recall to 17%, and the f1-score jumps to 24% at 22. At 23, accuracy, recall, and f1-score are all 47percent. Age 24 has an accuracy of 22%, recall of 6%, and an f1-score of 10%. At 25, a person's f1-score is 4% higher than it should be. As a result, we can estimate the value up to age 70. The accuracy rate is 39%, the recall rate is 40%, and the f1-score rate is 38% on average.

	precision	recall	f1-score	support
21	0.22	0.23	0.22	1026
22	0.40	0.17	0.24	1022
23	0.47	0.47	0.47	1016
24	0.22	0.06	0.10	969
25	0.12	0.02	0.04	1043
26	0.19	0.37	0.25	1006
27	0.55	0.01	0.01	997
28	0.56	0.53	0.54	988
29	0.30	0.22	0.26	1023
30	0.35	0.27	0.30	1002
31	0.36	0.10	0.16	1011
32	0.94	0.89	0.92	982
33	0.36	0.39	0.38	997
34	0.77	0.72	0.74	1002
35	0.49	0.65	0.56	1009
36	0.19	0.74	0.30	1008
37	0.15	0.17	0.16	976
38	0.00	0.00	0.00	1007
39	0.00	0.00	0.00	1004
40	0.36	0.66	0.47	1004
41	0.02	0.00	0.00	1031
42	0.64	0.64	0.64	1023
43	0.80	0.81	0.81	1019
44	0.32	0.23	0.27	980
45	0.08	0.04	0.05	985
46	0.12	0.19	0.15	950
47	0.55	0.36	0.43	993
48	0.49	0.47	0.48	1026
49	0.64	0.55	0.59	995
50	0.28	0.36	0.32	966
51	0.36	0.48	0.41	2080
52	0.33	0.30	0.32	1993
53	0.37	0.48	0.42	2055
54	0.39	0.33	0.36	1981
55	0.50	0.79	0.61	1967
56	0.75	0.84	0.79	1995
57	0.23	0.26	0.24	1970
58	0.40	0.35	0.38	1962
59	0.43	0.68	0.53	1978
60	0.63	0.89	0.74	1992
61	0.11	0.04	0.05	998
62	0.58	0.45	0.51	1000
63	0.81	0.75	0.78	1024
64	0.42	0.41	0.41	997
65	0.37	0.42	0.40	980
66	0.35	0.38	0.37	1046
67	0.24	0.28	0.26	966
68	0.29	0.15	0.20	974
69	0.32	0.07	0.12	962
70	0.15	0.19	0.17	1020
accuracy			0.40	60000
macro avg	0.38	0.38	0.36	60000
weighted avg	0.39	0.40	0.38	60000

Figure 4.13: Implementation in Naive Bayes

Confusion Matrix:

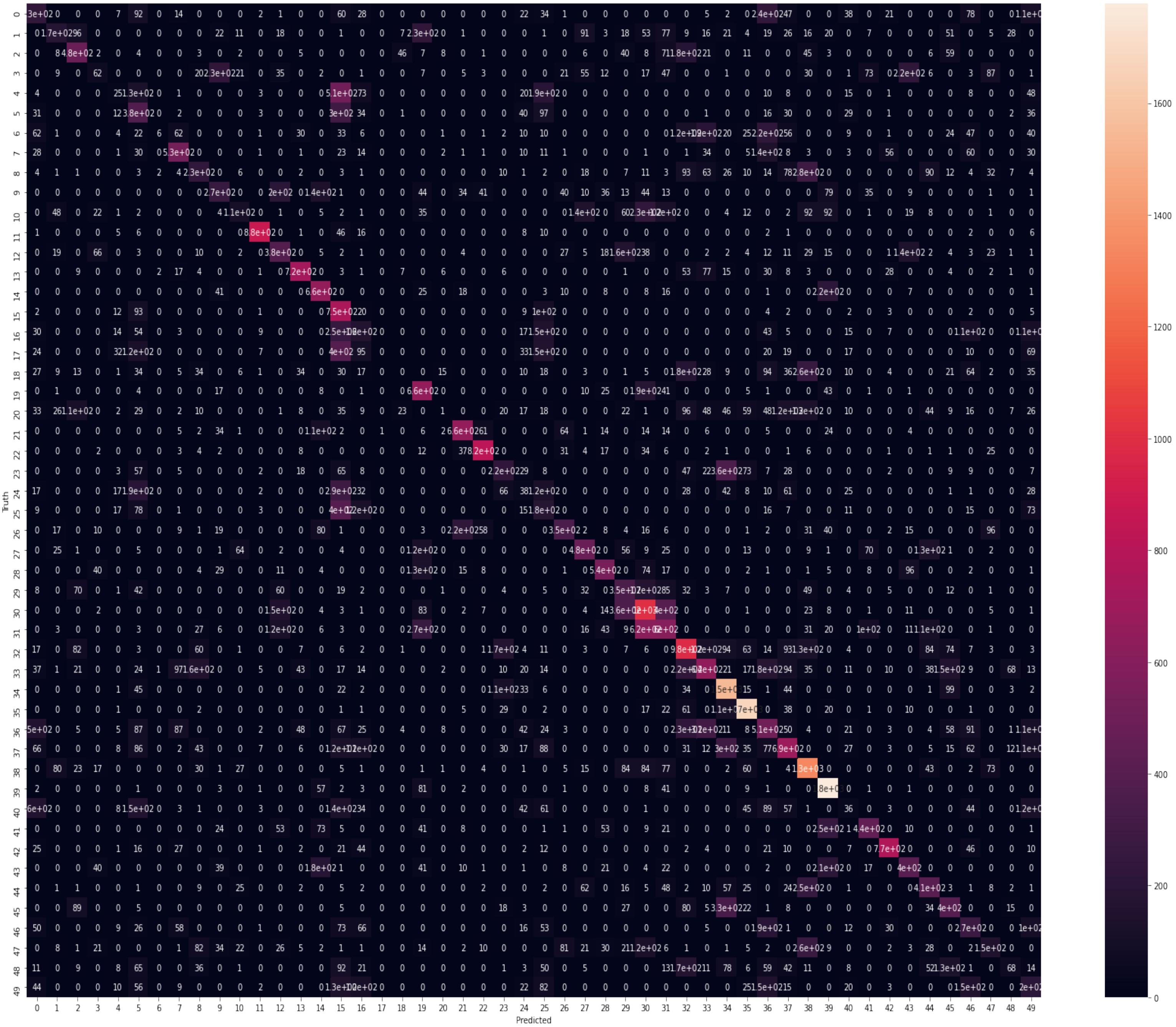

Figure 4.14: Confusion Matrix of Naive Bayes

Portion of Confusion Matrix

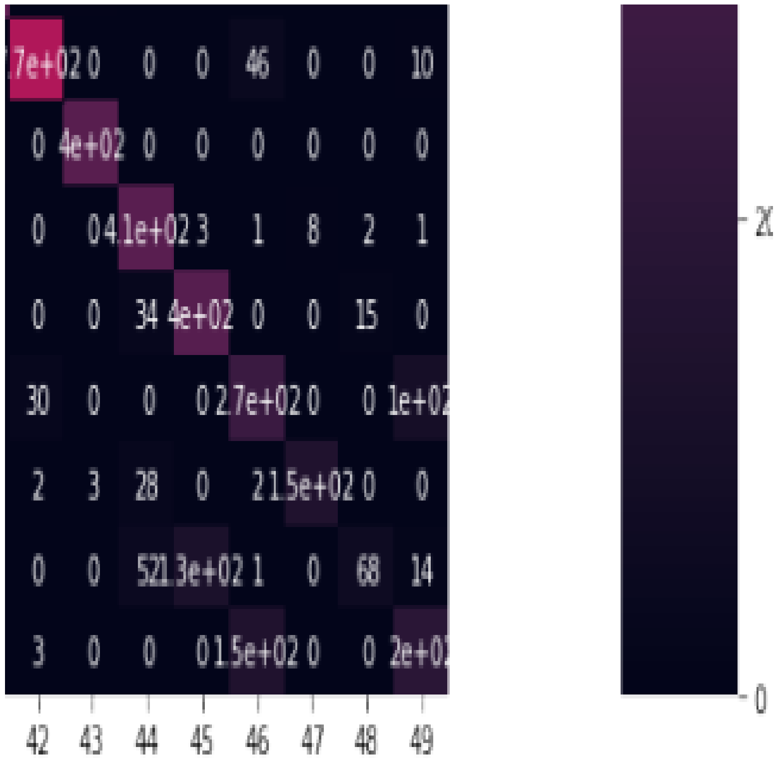

Figure 4.15: Sample of Confusion Matrix

Because we are showing the confusion matrix for a naive bayes with 50 ages, it is hard to do an analysis of all ages with only one confusion matrix. Because of this, we have only taken a small piece of the confusion matrix's overall picture. Now that we have made this confusion matrix for the naive Bayes, we can quickly understand how well our model works by looking at this confusion matrix. It will help us a lot when we look at the results of our model and try to figure out what they mean. From our picture we can see that for age 63(In confusion matrix the number 42), gives 7e+0 times the actual age 63 but it gives fourty six times 66,One time 69,Ten times 70. For age 66(In confusion matrix the number 45), gives 4.1e+02 times the actual age 66 but it gives five times 69.

Multilayer Perceptrons Network

Classification Report of Multilayer Perceptrons: The table in the report provides the values for accuracy, recall, and f1-score. Precision is the total number of true positives in which every forecast is correct. For age 21, we achieve 10% accuracy. Next, the proportion of true affirmative responses is defined as recall, and recall correlates to genuine positives. We have a 1percent recall rate for age 21. And the f1-score we obtained at the same age at a rate of 1%. f1 delivers a number that represents the harmonic mean of the accuracy and recall of a classifier. F1-score focuses on a single recall and accuracy value. It compares the relative usefulness of two distinct classifiers. Then, at age 22, precision scores are 43 percent, memory scores are 20%, and f1-score scores are 27 percent. At age 23, the accuracy rate is 49%, the recall rate is 58%, and the f1-score is 53%. There are 58% accuracy, 45 percent recall, and 51% f1-score for individuals aged 24. Age 25 has a 10% precision, 13% recall, and 11% f1-score. This is how we determine value up to age 70. And the typical values we receive are 42% accuracy rate, 41% recall rate, and 39% f1-score rate.

Confusion Matrix and Implementation in Multilayer Perceptrons:

	precision	recall	f1-score	support
21	0.22	0.23	0.22	1026
22	0.40	0.17	0.24	1022
23	0.47	0.47	0.47	1016
24	0.22	0.06	0.10	969
25	0.12	0.02	0.04	1043
26	0.19	0.37	0.25	1006
27	0.55	0.01	0.01	997
28	0.56	0.53	0.54	988
29	0.30	0.22	0.26	1023
30	0.35	0.27	0.30	1002
31	0.36	0.10	0.16	1011
32	0.94	0.89	0.92	982
33	0.36	0.39	0.38	997
34	0.77	0.72	0.74	1002
35	0.49	0.65	0.56	1009
36	0.19	0.74	0.30	1008
37	0.15	0.17	0.16	976
38	0.00	0.00	0.00	1007
39	0.00	0.00	0.00	1004
40	0.36	0.66	0.47	1004
41	0.02	0.00	0.00	1031
42	0.64	0.64	0.64	1023
43	0.80	0.81	0.81	1019
44	0.32	0.23	0.27	980
45	0.08	0.04	0.05	985
46	0.12	0.19	0.15	950
47	0.55	0.36	0.43	993
48	0.49	0.47	0.48	1026
49	0.64	0.55	0.59	995
50	0.28	0.36	0.32	966
51	0.36	0.48	0.41	2080
52	0.33	0.30	0.32	1993
53	0.37	0.48	0.42	2055
54	0.39	0.33	0.36	1981
55	0.50	0.79	0.61	1967
56	0.75	0.84	0.79	1995
57	0.23	0.26	0.24	1970
58	0.40	0.35	0.38	1962
59	0.43	0.68	0.53	1978
60	0.63	0.89	0.74	1992
61	0.11	0.04	0.05	998
62	0.58	0.45	0.51	1000
63	0.81	0.75	0.78	1024
64	0.42	0.41	0.41	997
65	0.37	0.42	0.40	980
66	0.35	0.38	0.37	1046
67	0.24	0.28	0.26	966
68	0.29	0.15	0.20	974
69	0.32	0.07	0.12	962
70	0.15	0.19	0.17	1020
accuracy			0.40	60000
macro avg	0.38	0.38	0.36	60000
weighted avg	0.39	0.40	0.38	60000

Figure 4.16: Implementation in Multilayer Perceptrons

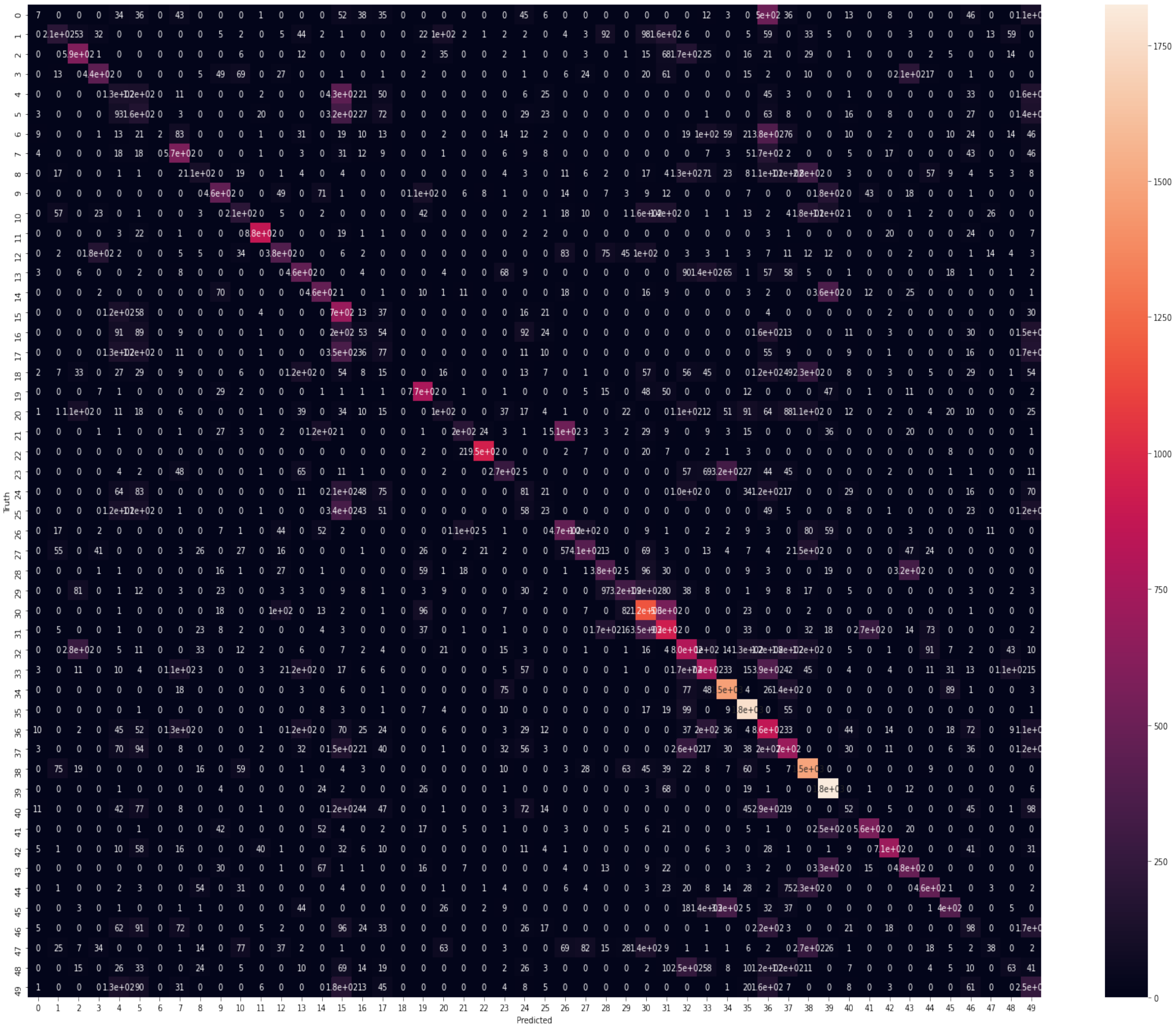

Figure 4.17: Confusion Matrix of Multilayer Perceptrons

Portion of Confusion Matrix

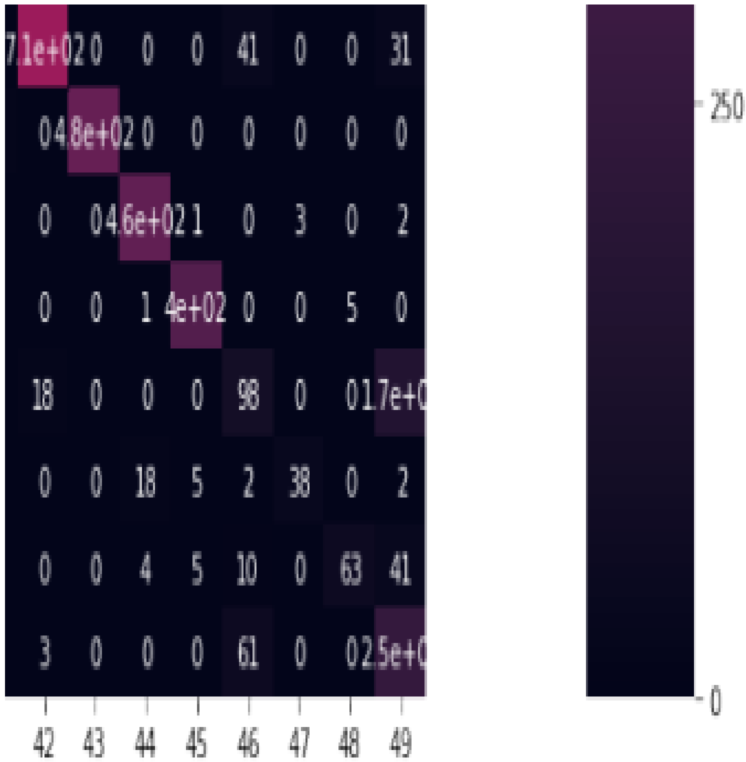

Figure 4.18: Sample of Confusion Matrix

Using a single confusion matrix for an MLP with 50 ages, it is difficult to examine all ages at the same time. Consequently, we only got a glimpse of the whole image of the confusion matrix. This MLP confusion matrix may now be used to evaluate the performance of MLP models. Using it, we'll be able to better comprehend our model's output. From our picture we can see that for age 63(In confusion matrix the number 42), gives 7.1e+02 times the actual age 63 but it gives fourty one times 67,thirty one time 70. For age 65(In confusion matrix the number 44), gives 4.6e+02 times the actual age 65 but it gives one time 69 and four times 70.

4.3 Comparison Between Different Algorithms

Algorithm	Accuracy
Random Forest Classifier	0.90
Decision tree classifier	0.83
K Nearest Neighbors Classifier	0.87
Naive Bayes	0.40
MLP	0.40

Table 4.1: Accuracy Score of Different Algorithms

In order to determine which approaches were most effective, we'll look at the rates of false negatives and false positives for each method, as well as their accuracy, precision, recall, and f-measure. The histogram in Fig.4.19 below shows the comparability of all methods.

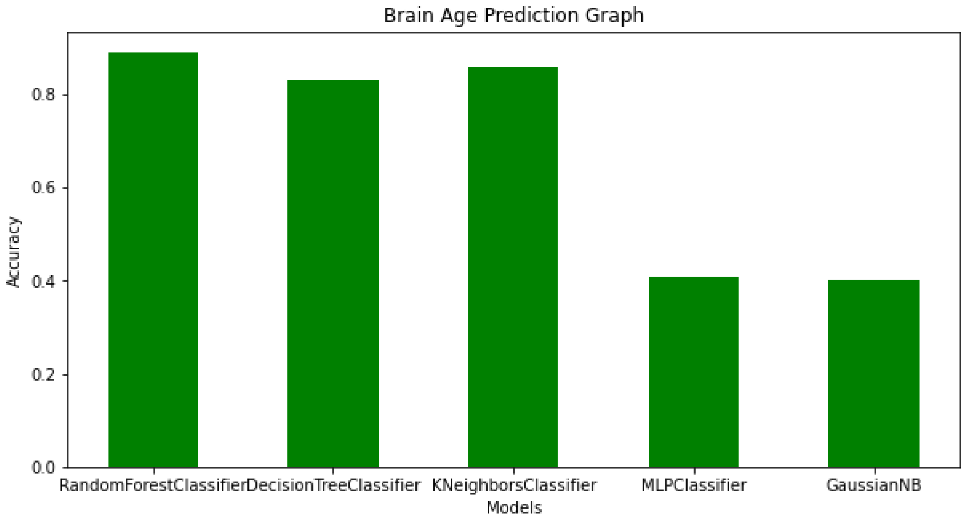

Figure 4.19: Comparability of All Algorithms (Accuracy)

We can see from the graph above that 90 % is the most accurate. Therefore, it is evident that the results we came up with after evaluating our dataset using 15 different methodologies are, for the most part, similar. Random forest classifier, Nave Bayes, Decision Tree and K-Nearest Neighbor are the most successful approaches.

4.4 Implementation in Raspberry Pi

4.4.1 Limitations

1. Raspberry PI can not train large datasets due to resource constraints, that's why we needed to train our dataset on a desktop. And we only tested the dataset in Raspberry PI. 2. The Raspberry Pi's RAM cannot be upgraded. The Pi's key components, including the 512 MB RAM, are soldered to the motherboard. When used as a small server, the Pi consumes around 100 MB of RAM. Because the RAM's storage is so limited, it cannot process large datasets. That is why we had to reduce

the size of the dataset in order to test it on the Raspberry Pi. 3. It can not do real-time execution. That is why could not train our dataset in the Raspberry PI, we had to train it using a machine learning algorithm on a computer. 5. Since the raspberry pi doesn't have any internal storage it requires a micro SD card to work as internal storage. Since the raspberry pi doesn't have any internal storage it requires a micro SD card to work as internal storage. We have used a 128 GB SD card which was difficult to implement. Not only that, as SD cards are not as fast, it lacks performance. So, this increases the boot time of the board and the read/write speed of the raspberry pi [13]. 6. Raspberry PI can not be run on Windows OS and needs to be run on Linux. Windows operating system is the most user-friendly also many apps are available for Windows OS because of the ".exe" format support. We have alternative apps available on Linux, but many popular software developers use the .exe formats. Although the ARM is a greatly efficient and low-powered architecture, it is not x86 and hence any binaries that are compiled to run on x86 cannot run on the Pi. The entire GNU/Linux distributions have been gathered for the ARM architecture and new ones are appearing all of the time. There are only a few applications that require x86. Moreover, we have done all our work on Windows OS except for testing the dataset in the raspberry PI part. It was difficult transitioning from one OS to another as we are not used to Linux [23]. 7. It is not a substitute for a computer, and the processor is not as fast. It is time-consuming to download and install software i.e.; unable to do any complex multitasking.

4.4.2 Implementation

1. Creating an Environment for Brain Age Prediction: We must create an environment in which we can execute the tests for each age. That is why we have linked our Raspberry Pi to the internet so that we may install the tools required to create our testing environment. In Raspberry Pi, we installed a Python environment as well as the essential libraries for our testing environment. For example, we have NumPy, pandas, matplotlib, and other tools installed for testing.

2. Importing The Trained Model Into Raspberry Pi: When importing the trained model into the Raspberry Pi, we have to keep in mind that the trained model memory size should be kept to a minimum. That's why we employed Pickle's write-back feature. We transformed our trained model, which consumes very little memory (about 1.37 MB), using this Pickles write-back method (1,440,162 bytes). The transformed trained model was then transferred to the Raspberry Pi using a data wire. After transferring the model, we used Pickle read back to function to load the trained model.
3.Testing the model in Raspberry Pi:
As we are going to test the individual age, we have to make the sample of each person.We have randomly selected a person's EEG signals information by using simple function.We have made 10 different people EEG signals information from our testing dataset(Randomly selected the 10 rows from testing dataset).
In the testing phase we have done the testing for ten people both in Raspberry pi and pc.Our pc and Raspberry pi both predicted the same age for ten people and it takes same ammount of time

	EEG FP1-REF	EEG FP2-REF	EEG F3-REF	EEG F4-REF	EEG C3-REF	EEG C4-REF	EEG P3-REF	EEG P4-REF	EEG O1-REF	EEG O2-REF	...	EEG 26-REF	EEG 27-REF	EEG 28-REF	EEG 29-REF	EEG 30-REF	EEG T1-REF	EEG T2-REF
327070	-9.915	-12.814	-25.784	-14.187	-18.154	-21.359	-28.530	-20.443	-22.732	-28.836	...	-58.590	-40.432	-45.315	-46.383	-49.130	-21.664	-17.697
388108	-12.661	-1.217	-6.100	2.750	-0.454	2.140	5.649	0.309	7.633	9.006	...	54.782	49.137	45.475	65.006	63.022	-5.642	8.548
115397	2.445	-23.342	12.821	-8.236	10.380	-2.591	-3.353	-5.947	-10.373	-6.100	...	-21.969	7.633	-11.593	19.840	763.247	19.687	-15.103
30211	-25.021	-13.577	-20.901	1.529	-2.591	6.107	0.004	3.513	0.614	5.344	...	-41.043	-16.476	-20.596	-10.220	21.518	-20.443	0.614
212782	-20.443	-15.103	-1.522	1.835	3.666	-3.659	-2.285	-0.759	-7.321	-11.135	...	114.139	247.806	89.115	109.256	88.657	-10.525	-4.727
68868	-8.389	-5.184	0.614	-6.100	5.344	0.004	1.987	0.004	1.377	7.175	...	35.404	133.060	2.140	41.965	10.837	0.919	17.704
316544	11.448	-37.380	11.295	-15.255	2.445	-10.373	-9.304	-7.931	-8.694	-8.236	...	14.042	34.794	-8.999	10.227	19.687	21.213	-20.901
536475	29.300	25.638	18.314	13.736	12.363	9.922	9.311	5.039	6.870	7.022	...	-32.040	-7.779	-28.988	-46.688	-50.808	7.938	10.380
299915	-31.124	-22.579	6.260	3.055	15.568	9.922	19.382	10.227	2.903	7.938	...	-20.138	24.570	-20.291	2.445	-19.985	0.614	19.840
581267	15.873	12.821	15.720	11.753	4.123	14.347	-8.084	10.227	-5.490	-5.337	...	-25.326	19.230	-31.582	-56.301	-15.255	14.652	9.922

Figure 4.20: Sample of Ten people Brain age prediction

```
The Brain Age for 10 people in Raspberry Pi:
[53 59 32 24 42 67 52 64 50 69]
Total time taken to predict the 10 age: 0.0 minutes 0.00 seconds
```

Figure 4.21: Prediction result and timing for ten sample

```
The Brain Age for 10 people in PC:
[53 59 32 24 42 67 52 64 50 69]
Total time taken to predict the 10 age: 0.0 minutes 0.00 seconds
```

Figure 4.22: Prediction result and timing for ten sample

Chapter 5

Conclusion

This study presents an architecture that is based on machine learning and that may be used to automatically determine a person's brain age by utilizing an Internet of Things device. We were required to train our model with a massive amount of data, and the accuracy results imply that our model may be used to automatically predict brain age on a device with limited resources such as the Raspberry Pi. By employing the Random Forest Classifier, we were able to get the highest possible accuracy of 90%. The accuracy of the K-NN algorithm, the Decision Tree Algorithm, the Gaussian Naive Bayes algorithm, and the Multilayer Perceptron Algorithm, respectively, is 87%, 83%, 40%, and 40%. During the course of this research, we came up with the most accurate algorithm for estimating a person's age that is the Random Forest algorithm, and we used Raspberry Pi to put it into action. The development of a device based on our model that is capable of instantaneously converting EEG brain signals into the participant's brain age will be the focus of our next project. Additionally, it will be helpful in the early diagnosis of any potential mental problems. That may be able to aid in the research of brain disorders and even possibly may save lives. So we are hoping that we will be able to contribute to the medical field even a little.

5.0.1 Future Work

Despite our best efforts, there are still some issues with our system.
A few drawbacks:
1. We did not use the whole dataset since we randomly picked a certain number of rows for each age group; however, we plan to use the entire dataset in the future.
2.This is only a rudimentary neural network and machine learning method, but we want to use a more advanced neural network in the future.
3.We're just looking at the performance of various algorithms on a Raspberry Pi, but in the future, we want to create an IOT-based architecture and put it to use in the real world.

Aiming for more precision in our model, we seek to find additional important factors or traits that are connected with EEG brain age prediction

Bibliography

[1] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[2] C. Kolski and E. Le Strugeon, "A review of intelligent human-machine interfaces in the light of the arch model," *International Journal of Human-Computer Interaction*, vol. 10, no. 3, pp. 193–231, 1998.

[3] J. S. Kwon, B. F. O'Donnell, G. V. Wallenstein, *et al.*, "Gamma frequency–range abnormalities to auditory stimulation in schizophrenia," *Archives of general psychiatry*, vol. 56, no. 11, pp. 1001–1005, 1999.

[4] L. Rokach and O. Maimon, "Top-down induction of decision trees classifiers-a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 35, no. 4, pp. 476–487, 2005.

[5] A. Randolph, S. Karmakar, and M. Jackson, "Towards predicting control of a brain-computer interface," 2006.

[6] N. Horning *et al.*, "Random forests: An algorithm for image classification and generation of continuous fields data sets," in *Proceedings of the International Conference on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences, Osaka, Japan*, vol. 911, 2010.

[7] D. P. Subha, P. K. Joseph, R. Acharya U, C. M. Lim, *et al.*, "Eeg signal analysis: A survey," *Journal of medical systems*, vol. 34, no. 2, pp. 195–212, 2010.

[8] O. Kramer, "Dimensionality reduction by unsupervised k-nearest neighbor regression," in *2011 10th international conference on machine learning and applications and workshops*, IEEE, vol. 1, 2011, pp. 275–278.

[9] C. Sammut and G. I. Webb, *Encyclopedia of machine learning*. Springer Science & Business Media, 2011.

[10] L. R. M. d. Paiva, A. A. Pereira, M. F. S. d. Almeida, G. L. Cavalheiro, S. T. Milagre, and A. d. O. Andrade, "Analysis of the relationship between eeg signal and aging through linear discriminant analysis (lda)," *Revista Brasileira de Engenharia Biomedica*, vol. 28, no. 2, pp. 155–168, 2012.

[11] M. Richardson and S. Wallace, *Getting started with raspberry PI.* " O'Reilly Media, Inc.", 2012.

[12] "Diagnostic and statistical manual of mental disorders," *Am Psychiatric Assoc (5th ed.)*, vol. 21, pp. 591–643, 2013.

[13] V. Vujović and M. Maksimović, "Raspberry pi as a wireless sensor node: Performances and constraints," in *2014 37th international convention on information and communication technology, electronics and microelectronics (MIPRO)*, IEEE, 2014, pp. 1013–1018.

[14] M. Sharma, N. Agarwal, and S. Reddy, "Design and development of daughter board for usb-uart communication between raspberry pi and pc," in *International Conference on Computing, Communication & Automation*, IEEE, 2015, pp. 944–948.

[15] A. Chaudhary, S. Kolhe, and R. Kamal, "An improved random forest classifier for multi-class classification," *Information Processing in Agriculture*, vol. 3, no. 4, pp. 215–222, 2016.

[16] B. Dubois, H. Hampel, H. H. Feldman, *et al.*, "Preclinical alzheimer's disease: Definition, natural history, and diagnostic criteria," *Alzheimer's & Dementia*, vol. 12, no. 3, pp. 292–323, 2016.

[17] A. Schmeling, R. Dettmeyer, E. Rudolf, V. Vieth, and G. Geserick, "Forensic age estimation: Methods, certainty, and the law," *Deutsches Ärzteblatt International*, vol. 113, no. 4, p. 44, 2016.

[18] A. Ukil, S. Bandyoapdhyay, C. Puri, and A. Pal, "Iot healthcare analytics: The importance of anomaly detection," in *2016 IEEE 30th international conference on advanced information networking and applications (AINA)*, IEEE, 2016, pp. 994–997.

[19] A. Bronshtein, "A quick introduction to k-nearest neighbors algorithm," *Noteworthy-The Journal Blog*, 2017.

[20] J. W. Kuziek, A. Shienh, and K. E. Mathewson, "Transitioning eeg experiments away from the laboratory using a raspberry pi 2," *Journal of neuroscience methods*, vol. 277, pp. 75–82, 2017.

[21] A. Li, J. Wu, and Z. Liu, "Market manipulation detection based on classification methods," *Procedia Computer Science*, vol. 122, pp. 788–795, 2017.

[22] R. Shaw, "Top 10 machine learning algorithms for beginners," *URl: https://www. kdnuggets. com/2017/10/top-10-machine-learning-algorithmsbeginners. html*, 2017.

[23] S. Tanwar, P. Patel, K. Patel, S. Tyagi, N. Kumar, and M. S. Obaidat, "An advanced internet of thing based security alert system for smart home," in *2017 international conference on computer, information and telecommunication systems (CITS)*, IEEE, 2017, pp. 25–29.

[24] O. Al Zoubi, C. Ki Wong, R. T. Kuplicki, *et al.*, "Predicting age from brain eeg signals—a machine learning approach," *Frontiers in aging neuroscience*, vol. 10, p. 184, 2018.

[25] J. H. Cole, R. E. Marioni, S. E. Harris, and I. J. Deary, "Brain age and other bodily 'ages': Implications for neuropsychiatry," *Molecular psychiatry*, vol. 24, no. 2, pp. 266–281, 2019.

[26] N. Donges, "A complete guide to the random forest algorithm," *Built In*, vol. 16, 2019.

[27] B. Kaur, D. Singh, and P. P. Roy, "Age and gender classification using brain–computer interface," *Neural Computing and Applications*, vol. 31, no. 10, pp. 5887–5900, 2019.

[28] ——, "Age and gender classification using brain–computer interface," *Neural Computing and Applications*, vol. 31, no. 10, pp. 5887–5900, 2019.

[29] M. M. Vandenbosch, D. van't Ent, D. I. Boomsma, A. P. Anokhin, and D. J. Smit, "Eeg-based age-prediction models as stable and heritable indicators of brain maturational level in children and adolescents," *Human brain mapping*, vol. 40, no. 6, pp. 1919–1926, 2019.

[30] N. S. Bastos, B. P. Marques, D. F. Adamatti, and C. Z. Billa, "Analyzing eeg signals using decision trees: A study of modulation of amplitude," *Computational Intelligence and Neuroscience*, vol. 2020, 2020.

[31] D. Devasia, T. Roshini, N. S. Jacob, S. M. Jose, and S. Joseph, "Assistance for quadriplegic with bci enabled wheelchair and iot," in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, IEEE, 2020, pp. 1220–1226.

[32] W. Khan, M. A. Ghazanfar, M. A. Azam, A. Karami, K. H. Alyoubi, and A. S. Alfakeeh, "Stock market prediction using machine learning classifiers and social media, news," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–24, 2020.

[33] F. Laport, A. Dapena, P. M. Castro, F. J. Vazquez-Araujo, and D. Iglesia, "A prototype of eeg system for iot," *International journal of neural systems*, vol. 30, no. 07, p. 2 050 018, 2020.

[34] I. A. Pap, S. Oniga, and A. Alexan, "Machine learning eeg data analysis for ehealth iot system," in *2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, IEEE, 2020, pp. 1–4.

[35] V. V. Vardhan, U. Venkatesh, and S. Yadav, "Signal processing based autonomous sensory meridian response to treat insomnia," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, IEEE, 2020, pp. 1173–1176.

[36] K. O. Chicaiza and M. E. Benalcázar, "A brain-computer interface for controlling iot devices using eeg signals," in *2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM)*, IEEE, 2021, pp. 1–6.

[37] N. S. Dhillon, A. Sutandi, M. Vishwanath, M. M. Lim, H. Cao, and D. Si, "A raspberry pi-based traumatic brain injury detection system for single-channel electroencephalogram," *Sensors*, vol. 21, no. 8, p. 2779, 2021.

[38] A. R. Elshenaway and S. K. Guirguis, "Adaptive thresholds of eeg brain signals for iot devices authentication," *IEEE Access*, vol. 9, pp. 100 294–100 307, 2021.

[39] T. Tazrin, Q. A. Rahman, M. M. Fouda, and Z. M. Fadlullah, "Lihea: Migrating eeg analytics to ultra-edge iot devices with logic-in-headbands," *Ieee Access*, vol. 9, pp. 138 834–138 848, 2021.

[40] E. Dry and S.-d. EEG-Electroencephalogram, "The wet eeg cap & differences between water-based, saline and gel eeg caps,"

[41] GeeksforGeeks, *Decision tree*. [Online]. Available: https://www.geeksforgeeks.org/decision-tree/.

[42] S. Prabhakaran, *How naive bayes algorithm works? (with example and full code): Ml+*. [Online]. Available: https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/.