AUTISM SPECTRUM DISORDER DETECTION TECHNIQUES USING MACHINE LEARNING

In the partial fulfillment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICATIONS

A project submitted by

NETETI VENKATA SANTOSH JAGADEESH

(Regd. No: 321206420022)

Under the guidance of

Dr. S. JHANSI RANI

Associate Professor

DEPARTMENT OF INFORMATION TECHNOLOGY AND COMPUTER APPLICATIONS

ANDHRA UNIVERSITY COLLEGE OF ENGINEERING,

ANDHRA UNIVERSITY, VISAKHAPATNAM – 530003

(2021-2023)

DEPARTMENT OF INFORMATION TECHNOLOGY AND COMPUTER APPLICATIONS

ANDHRA UNIVERSITY COLLEGE OF ENGINEERING

VISAKHAPATNAM-530003

CERTIFICATE

THIS IS TO CERTIFY THAT THE PROJECT REPORT ENTITLED "AUTISM SPECTRUM DISORDER DETECTION TECHNIQUE USING MACHINE LEARNING" IS THE BONAFIDE WORK CARRIED OUT BY WITH REGD. NO: 321206420022, A STUDENT OF MCA IN AU COLLEGE OF ENGINEERING (A), ANDHRA UNIVERSITY, VISAKHAPATNAM, DURING THE YEAR 2021-2023, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF MASTER OF COMPUTER APPLICATIONS.

Project Guide
DR. S. JHANSI RANI

Head of the Department
Prof. K. NAGESWARA RAO

DECLARATION

I declare that the project report entitled "**AUTISM SPECTRUM DISORDER DETECTION TECHNIQUE USING MACHINE LEARNING**" has been done by me in partial fulfilment of requirement for the award of degree of "**Master of Computer Applications**", during the academic year 2021-2023 under the guidance of "**Dr. S. Jhansi Rani**", department of Information Technology and Computer Applications, AU College of Engineering(A), Andhra University, Visakhapatnam. I, here by declare that this project work has not been submitted to any other universities/institutions for the award of any degree.

NETETI VENKATA SANTOSH JAGADEESH

(321206420022)

ACKNOWLEDGEMENT

We have immense pleasure in expressing earnest gratitude to our Project Guide **Dr. S. Jhansi Rani**, Associate Professor, Andhra University for her inspiring and scholarly guidance. Despite her preoccupation with several assignments, she has been kind enough to spare her valuable time and gave us the necessary counsel and guidance at every stage of planning and constitution of this work. We express sincere gratitude for having up this project work and for helping us graciously throughout the execution of this work.

We express our sincere thanks to **Prof. K. NAGESWARA RAO**, Head of the Department, Computer Applications and Information Technology, Andhra University College of Engineering, Andhra University for his keen interest and providing necessary facilities for this project study.

We express our sincere thanks to **Prof. G. SASIBHUSANA RAO**, Principal, Andhra University College of Engineering for his keen interest and for providing necessary facilities for this project study.

We express our sincere gratitude to **Prof. P.VG.D. PRASAD REDDY**, Vice Chancellor, Andhra University for his keen interest and for providing necessary facilities for this project study.

We extend our sincere thanks to our academic teaching staff and non-teaching staff for their help throughout our studies.

ABSTRACT

ABSTRACT

Autism Spectrum Disorder (ASD) is a neurodevelopmental disorder characterized by persistent challenges in social interaction, communication, and repetitive behaviors. Early detection and intervention play a crucial role in improving the outcomes for individuals with ASD. In recent years, there has been a growing interest in utilizing machine learning techniques to aid in the early diagnosis and detection of ASD. This project aims to develop a novel ASD detection technique using machine learning algorithms.

The proposed technique involves the collection of various data modalities, including behavioral, demographic, and medical information, from individuals suspected of having ASD. These data are combined and preprocessed to extract meaningful features that capture the underlying characteristics associated with ASD. Feature selection and dimensionality reduction techniques are employed to enhance the efficiency and effectiveness of the subsequent machine learning models.

Several machine learning algorithms, such as decision trees, support vector machines, random forests, and deep neural networks, are investigated and compared to identify the most accurate and reliable model for ASD detection. The models are trained on a carefully curated dataset comprising both ASD and typically developing individuals to enable accurate classification.

The evaluation of the developed ASD detection technique involves assessing its performance in terms of accuracy, sensitivity, specificity, and area under the receiver operating characteristic curve (AUC-ROC). Cross-validation and independent testing are conducted to ensure the robustness and generalizability of the proposed technique.

Furthermore, the project focuses on the interpretability of the machine learning models to gain insights into the underlying factors contributing to the classification decisions. This interpretability aspect is crucial for understanding the discriminative features and potential biomarkers associated with ASD, leading to improved understanding and knowledge of the disorder.

Furthermore, the project focuses on the interpretability of the machine learning models to gain insights into the underlying factors contributing to the classification decisions. This interpretability aspect is crucial for understanding the discriminative features and potential biomarkers associated with ASD, leading to improved understanding and knowledge of the disorder.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS/NOTATIONS

ML	MACHINE LEARNING
AI	ARTIFICIAL INTELLIGENCE
SVM	SUPPORT VECTOR MACHINE
KNN	K NEAREST NEIBHOURS
XG BOOST	EXTREAME GRADIENT BOOST

CHAPTER 1
INTRODUCTION

CHAPTER 1
INTRODUCTION

1.1 Overview

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental disorder that affects individuals worldwide. It is characterized by difficulties in social communication and interaction, as well as restricted and repetitive patterns of behavior, interests, or activities. Early detection and diagnosis of ASD are crucial for providing timely interventions and support, leading to improved outcomes for individuals with ASD.

Traditional methods for ASD detection heavily rely on clinical assessments, behavioral observations, and expert evaluations. While these methods have been valuable, they often suffer from subjectivity, time-consuming processes, and potential biases. There is a need for more objective, efficient, and data-driven approaches to enhance the accuracy and accessibility of ASD detection.

Machine learning, a subfield of artificial intelligence, offers promising opportunities for improving ASD detection. By leveraging algorithms that can learn patterns and relationships from data, machine learning techniques have the potential to identify subtle features and variations associated with ASD. This can enable automated screening and provide valuable insights into the underlying factors contributing to ASD.

The proposed project aims to develop an ASD detection technique using machine learning algorithms. By collecting and integrating various data modalities, such as behavioral, demographic, and medical information, the project seeks to capture comprehensive information related to ASD. Feature extraction and selection techniques will be employed to identify the most relevant and informative features for ASD classification.

Several machine learning algorithms, including decision trees, support vector machines, random forests, and deep neural networks, will be investigated and compared to identify the most accurate and reliable model for ASD detection. The developed models will be trained on carefully curated datasets, consisting of both individuals with ASD and typically developing individuals, to ensure accurate classification.

The evaluation of the proposed technique will involve assessing the performance of the machine learning models in terms of accuracy, sensitivity, specificity, and AUC-ROC. Cross-validation techniques will be applied to validate the models' robustness and generalizability.

Furthermore, the interpretability of the models will be emphasized to gain insights into the discriminative features and potential biomarkers associated with ASD.

The successful development of an accurate and interpretable ASD detection technique using machine learning has the potential to revolutionize the field of autism diagnosis and early intervention. By providing an automated and efficient screening tool, this project aims to facilitate early identification, reduce diagnosis time, and improve outcomes for individuals with ASD by enabling timely interventions and support.

1.1 MOTIVATION FOR THE WORK

The main motivation of doing this research is to present a Autism Spectrum Disorder Detection model for the prediction of occurrence of Autistic disease. Further, this research work is aimed towards identifying the best classification algorithm for identifying the possibility of Autism in a patient. This work is justified by performing a comparative study and analysis using three classification algorithms namely Naïve Bayes, Decision Tree, and Random Forest are used at different levels of evaluations. Although these are commonly used machine learning algorithms, The Autism Spectrum Disorder Detection is a vital task involving highest possible accuracy. Hence, the three algorithms are evaluated at numerous levels and types of evaluation strategies. This will provide researchers and medical practitioners to establish a better.

1.2 PROBLEM STATEMENT

The major challenge in ASD is its detection. There are instruments available which can predict ASD but either it are expensive or are not efficient to calculate chance of Autism in human. Furthermore,the traditional method often lack to leverage large-scale data and complex patterns that could enhance diagnostic accuracy. They may struggle to capture subtle features and variations assosciated with ASD across different Individuals. Since we have a good amount of data in today's world, we can use various machine learning algorithms to analyze the data for hidden patterns. The hidden patterns can be used for health diagnosis in medicinal data

CHAPTER 2
LITERARTURE SURVEY

CHAPTER 2
LITERATURE SURVEY

With growing development in the field of medical science alongside machine learning various experiments and researches has been carried out in these recent years releasing the relevant significant papers

1. Machine Learning-Based Methods for Autism Spectrum Disorder Diagnosis and Prognosis: A Systematic Review" by López-Bonilla et al. (2020) This systematic review explores the application of machine learning techniques in ASD diagnosis and prognosis. It provides an overview of various machine learning algorithms used, data modalities employed, and performance metrics evaluated. The review highlights the potential of machine learning in improving ASD detection and emphasizes the need for standardized datasets and evaluation protocols.
2. Deep Learning Approaches for Autism Spectrum Disorder Classification: A Review" by Gilani et al. (2019) This review focuses on deep learning techniques, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), for ASD classification. It discusses the advantages of deep learning in capturing complex patterns from different data modalities and provides insights into the challenges and future directions in the field.

   Automatic Detection of Autism Spectrum Disorder Using Machine Learning and EEG Signal Processing: A Systematic Review" by Roshni et al. (2020) This systematic review examines the utilization of electroencephalogram (EEG) signal processing and machine learning algorithms for ASD detection. It discusses the potential of EEG-based features and classifiers for accurate ASD classification and identifies the limitations and future directions for improving EEG-based ASD detection methods.
3. A Review of Machine Learning Applications in Autism Spectrum Disorder: A Systematic Review" by Lajiness-O'Neill et al. (2018) This systematic review provides an overview of machine learning applications in ASD research. It covers various domains, including diagnosis, prediction, treatment, and intervention. The review highlights the potential of machine learning in identifying biomarkers, developing personalized interventions, and improving outcomes for individuals with ASD.

4. Machine Learning in Autism Research: Advancements, Challenges, and Future Directions" by Bone et al. (2020) This article discusses the advancements, challenges, and future directions of machine learning in ASD research. It explores the integration of multimodal data, interpretability of machine learning models, ethical considerations, and the importance of collaboration between researchers and clinicians. The article emphasizes the need for robust and validated machine learning approaches in ASD detection.

These studies highlight the growing interest in utilizing machine learning techniques for ASD detection and classification. They provide insights into the advantages, challenges, and potential future directions in the field. The literature survey demonstrates the potential of machine learning in improving ASD diagnosis, prognosis, and personalized interventions, paving the way for more accurate and efficient ASD detection technique.

CHAPTER 3

METHODOLOGY

CHAPTER 3
METHODOLOGY

3.1 EXISTING SYSTEM

These studies highlight the growing interest in utilizing machine learning techniques for ASD detection and classification. They provide insights into the advantages, challenges, and potential future directions in the field. The literature survey demonstrates the potential of machine learning in improving ASD diagnosis, prognosis, and personalized interventions, paving the way for more accurate and efficient ASD detection techniques.

Psychological tests, such as the Autism Diagnostic Observation Schedule (ADOS) and the Autism Diagnostic Interview-Revised (ADI-R), are commonly used in clinical settings to aid in ASD diagnosis. These tests involve direct observations, interviews with parents or caregivers, and scoring based on predefined criteria. While these assessments provide valuable insights, they are resource-intensive, require trained professionals, and may not always be readily accessible in all healthcare settings.

Furthermore, the existing approaches to ASD detection typically lack the ability to leverage large-scale data and complex patterns that could potentially enhance diagnostic accuracy. They may also be limited in their ability to capture subtle features and variations associated with ASD across different individuals.

Overall, the existing system for ASD detection faces challenges related to subjectivity, time consumption, limited scalability, and potential bias. There is a need for more objective, efficient, and data-driven approaches to improve the accuracy and accessibility of ASD detection.

3.2 PROPOSED SYSTEM

The proposed system aims to develop an innovative autism spectrum disorder (ASD) detection technique utilizing machine learning algorithms. This system seeks to address the limitations of the existing approaches by leveraging the power of data-driven analysis, automation, and objective decision-making. The system implemented by following method.

1. Data Collection and Integration
2. Feauture Extraction and Selection
3. Data Preprocessing
4. Model Evolution and Optimization
5. Interpretability and Explainability

3.2.1 DATA COLLECTION AND INTEGRATION

Various data modalities, including behavioral, demographic, and medical information, are collected from individuals suspected of having ASD. These data sources may include structured assessments, questionnaires, sensor data, and electronic health records. The collected data are integrated and prepared for further analysis.

Initially, we collect a dataset for our ASD Detection System. After the collection of the dataset, we split the dataset into training data and testing data. The training dataset is used for prediction model learning and testing data is used for evaluating the prediction model. For this project, 70% of training data is used and 30% of data is used for testing. The dataset used for this project is Toddler Autism dataset. The dataset consists of 19 Attributes

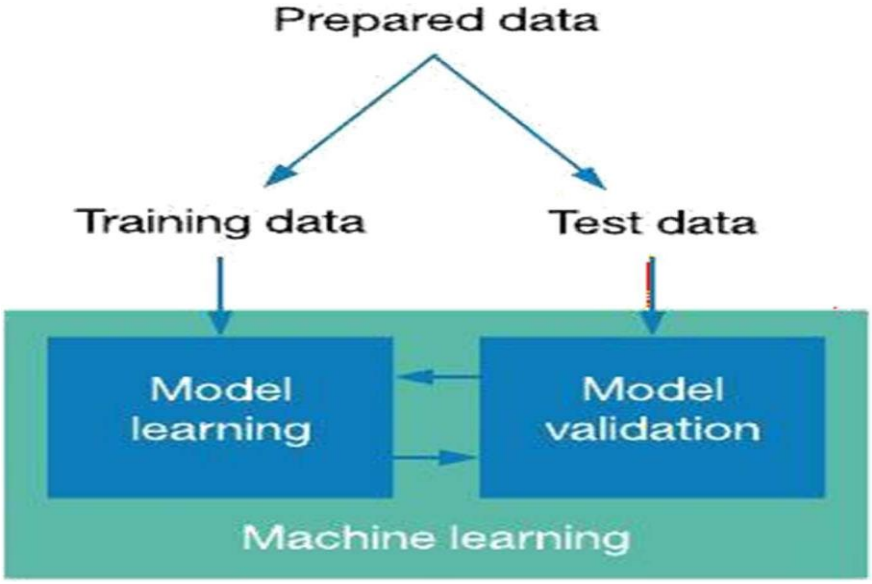
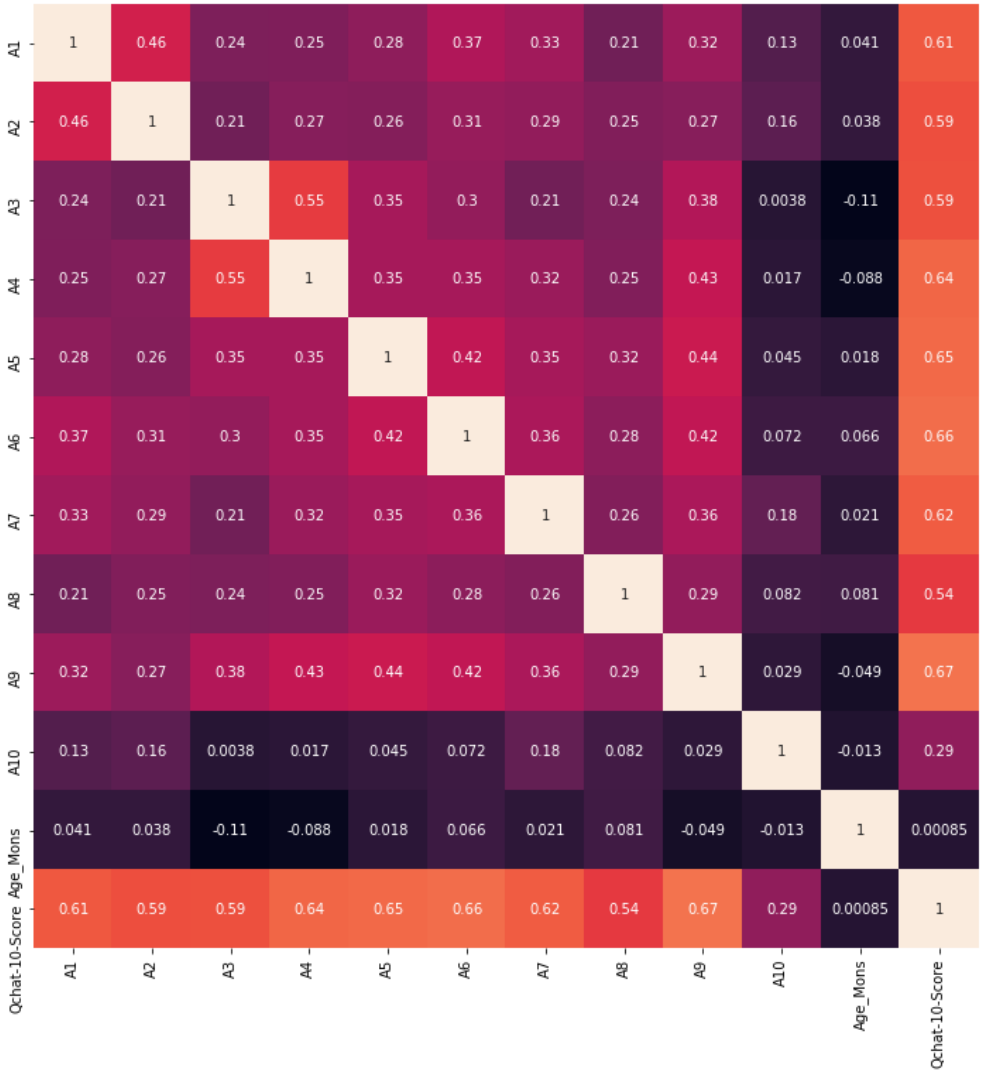


Figure: collection of data

3.2.2 FEAUTURE EXTRACTION AND SELECTION OF ATTRIBUTES

The integrated data undergo preprocessing and feature extraction to capture relevant information and patterns associated with ASD. Feature selection techniques are then applied to identify the most informative and discriminative features, reducing dimensionality and improving model performance.

Attribute or Feature selection includes the selection of appropriate attributes for the prediction system. This is used to increase the efficiency of the system. Various attributes of the patient like gender, age,A1-A10 etc are selected for the prediction. The Correlation matrix is used for attribute selection for this model.



3.2.3 DATA PREPROCESSING

Data pre-processing is an important step for the creation of a machine learning model. Initially, data may not be clean or in the required format for the model which can cause misleading outcomes. In pre-processing of data, we transform data into our required format. It is used to deal with noises, duplicates, and missing values of the dataset. Data pre-processing has the activities like importing datasets, splitting datasets, attribute scaling, etc. Preprocessing of data is required for improving the accuracy of the model.

Figure: Data Pre-processing

3.2.4 MODEL EVOLUTION AND OPTIMIZATION

The performance of the developed machine learning models is evaluated using various metrics, including accuracy, sensitivity, specificity, and area under the receiver operating characteristic curve (AUC-ROC). Cross-validation techniques are applied to assess the generalizability and robustness of the models. Model hyperparameters are tuned and optimized to achieve the best possible performance.

3.2.5 INTERPRETABILITY AND EXPLAINABILITY

The proposed system emphasizes the interpretability of the machine learning models to gain insights into the factors contributing to ASD classification. Interpretability techniques, such as feature importance analysis and model visualization, are employed to provide explanations for the model's decisions. This helps in understanding the underlying features and potential biomarkers associated with ASD.

The proposed system offers the potential to enhance the accuracy, efficiency, and accessibility of ASD detection. By leveraging machine learning algorithms and data-driven approaches, it aims to improve early identification, reduce diagnostic delays, and ultimately improve outcomes for individuals with ASD.

CHAPTER 4

WORKING OF SYSTEM

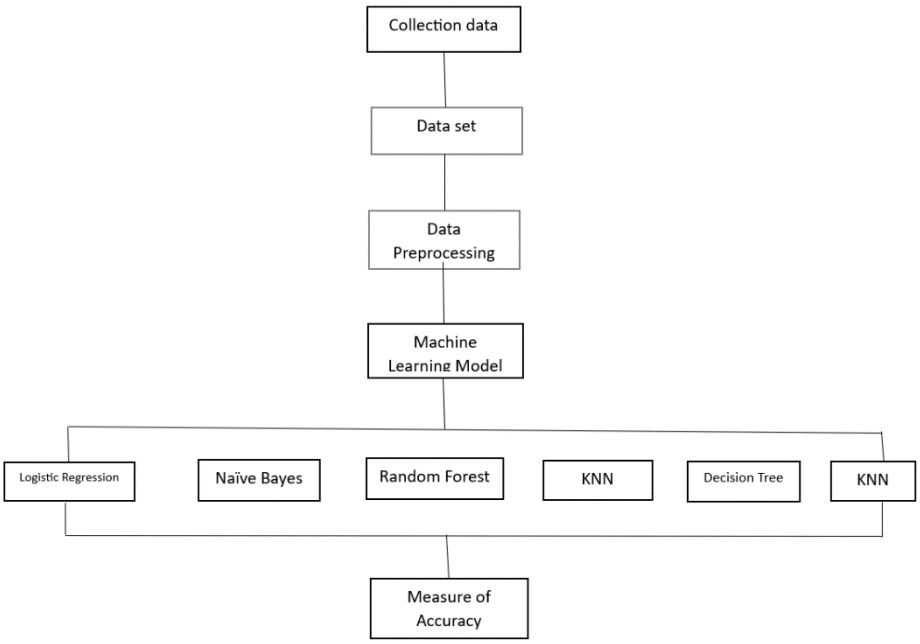
CHAPTER 4
WORKING OF SYSTEM

4.1 SYSTEM ARCHITECTURE

The system architecture gives an overview of the working of the system.

The working of this system is described as follows:

Dataset collection is collecting data which contains patient details. Attributes selection process selects the useful attributes for the prediction of heart disease. After identifying the available data resources, they are further selected, cleaned, made into the desired form. Different classification techniques as stated will be applied on preprocessed data to predict the accuracy of heart disease. Accuracy measure compares the accuracy of different classifiers.



System Architecture

4.2 MACHINE LEARNING

In machine learning, classification refers to a predictive modeling problem where a class label is predicted for a given example of input data.

4.2.1. SUPERVISED LEARNING

Supervised learning is the type of machine learning in which machines are trained using well "labelled" training data, and on the basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher

Supervised learning is a process of providing input data as well as correct output data to the machine learning model. The aim of a supervised learning algorithm is to find a mapping function to map the input variable(x) with the output variable(y)

4.2.2. UNSUPERVISED LEARNING

Unsupervised learning cannot be directly applied to a regression or classification problem because unlike supervised learning, we have the input data but no corresponding output data. The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.

Unsupervised learning is helpful for finding useful insights from the data.

Unsupervised learning is much similar to how a human learns to think by their own experiences, which makes it closer to the real AI.

Unsupervised learning works on unlabeled and uncategorized data which make unsupervised learning more important.

In real-world, we do not always have input data with the corresponding output so to solve such cases, we need unsupervised learning.

4.2.3. REINFORMENT LEARNING

Reinforcement learning is an area of Machine Learning. It is about taking suitable action to maximize reward in a particular situation. It is employed by various software and machines to find the best possible behaviour or path it should take in a specific situation. Reinforcement learning differs from supervised learning in a way that in supervised learning the training data has the answer key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of a training dataset, it is bound to learn from its experience.

4.3 ALGORITHMS

4.3.1. LOGISTIC REGRESSION

Logistic regression is one of the most popular Machine Learning algorithms,which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas logistic regression is used for solving the classification problems.

In Logistic regression, instead of fitting a regression line, we fit an "S"shaped logistic function, which predicts two maximum values.

The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.

Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Adavantages:

Logistic Regression is one of the simplest machine learning algorithms and is easy to implement yet provides great training efficiency in some cases. Also due to these reasons, training a model with this algorithm doesn't require high computation power.

The predicted parameters (trained weights) give inference about the importance of each feature. The direction of association i.e. positive or negative is also given. So we can use Logistic Regression to find out the relationship between the features

This algorithm allows models to be updated easily to reflect new data, unlike Decision Tree or Support Vector Machine. The update can be done using stochastic gradient descent.

Logistic Regression outputs well-calibrated probabilities along with classification results. This is an advantage over models that only give the final classification as results. If a training example has a 95% probability for a class, and another has a 55% probability for the same class, we get an inference about which training examples are more accurate for the formulated problem.

Disadvantages:

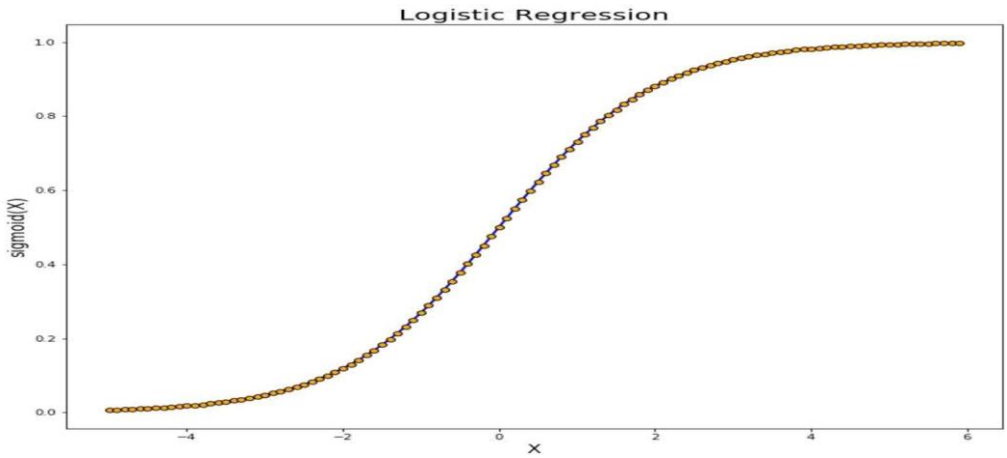
Logistic Regression is a statistical analysis model that attempts to predict precise probabilistic outcomes based on independent features. On high dimensional datasets, this may lead to the model being over-fit on the training set, which means overstating the accuracy of predictions on the training set and thus the model may not be able to predict accurate results on the test set. This usually happens in the case when the model is trained on little training data with lots of features. So on high dimensional datasets, Regularization techniques should be considered to avoid over- fitting (but this makes the model complex). Very high regularization

factors may even lead to the model being under-fit on the training data.

Non linear problems can't be solved with logistic regression since it has a linear decision surface. Linearly separable data is rarely found in real world scenarios. So the transformation of non linear features is required which can be done by increasing the number of features such that the data becomes linearly separable in higher dimensions.

Non-Linearly Separable Data:

It is difficult to capture complex relationships using logistic regression. More powerful and complex algorithms such as Neural Networks can easily outperform this algorithm.



4.3.2 NAÏVE BAYES

Naive Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.It is mainly used in text classification that includes a high-dimensional training dataset.

Naive Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

It is a probabilistic classifier, which means it predicts on the basis of the probability of an object. Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

The Naïve Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

The Naïve Bayes algorithm is comprised of two words Naive and Bayes, Which can be described as :

Naïve:

It is called Naive because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the basis of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.

Bayes:

It is called Bayes because it depends on the principle of Bayes' Theorem.

Baye's Theorm:

Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

The formula for Bayes' theorem is given as:

$$P(A|B)= \frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability:Probability of hypothesis A on the observed event B.
P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

P(A) is Prior Probability: Probability of hypothesis before observing the evidence.

P(B) is Marginal Probability: Probability of Evidence.

Types of Naive Bayes model:

There are three types of Naive Bayes Model, which are given below:

Gaussian:

The Gaussian model assumes that features follow a normal distribution. This means if predictors take continuous values instead of discrete, then the model assumes that these values are sampled from the Gaussian distribution.

Multinominal:

The Multinomial Naïve Bayes classifier is used when the data is multinomial distributed. It is primarily used for document classification problems, it means a particular document belongs to which category such as Sports, Politics, education, etc. The classifier uses the frequency of words for the predictors.

Bernoulli:

The Bernoulli classifier works similar to the Multinomial classifier, but the predictor variables are the independent Booleans variables. Such as if a particular word is present or not in a document. This model is also famous for document classification tasks.

4.3.3 SUPPORT VECTOR MACHINE

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called support vectors, and hence the algorithm is termed as Support Vector Machine.

Support vector machines (SVMs) are powerful yet flexible supervised machine learning algorithms which are used both for classification and regression. But generally, they are used in classification problems. In the 1960s, SVMs were first introduced but later they got refined in 1990. SVMs have their unique way of implementation as compared to other machine learning algorithms. Lately, they are extremely popular because of their ability to handle multiple continuous and categorical variables.

The followings are important concepts in SVM-

Support Vectors :

Data Points that are closest to the hyperplane are called support vectors. Separating line will be defined with the help of these data points.

Hyperplane:

As we can see in the above diagram, it is a decision plane or space which is divided between a set of objects having different classes.

Margin:

It may be defined as the gap between two lines on the closest data points of different classes. It can be calculated as the perpendicular distance from the line to the support vectors. Large margin is considered as a good margin and small margin is considered as a bad margin.

TYPES OF SVM

SVM can be of two types:

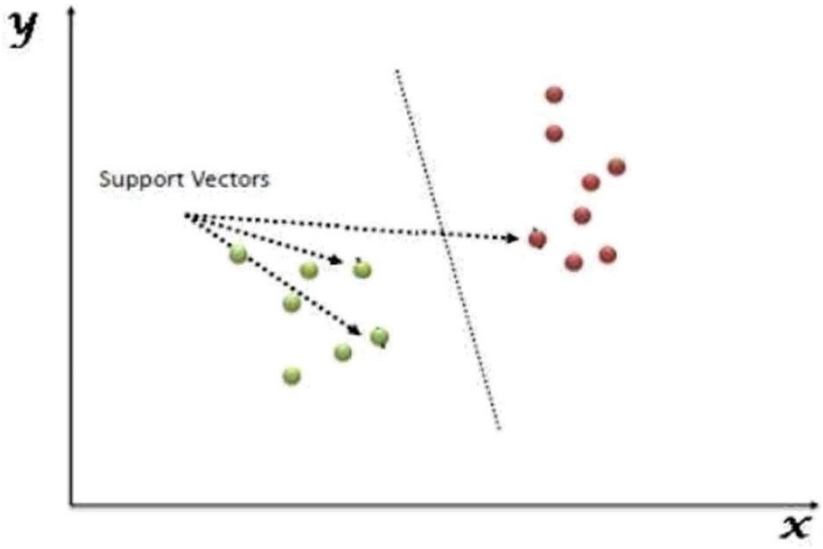
Linear SVM:

Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

Non Linear SVM:

Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non Linear SVM Classifier.

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.



Support vector machine

4.3.4 RANDOM FOREST

Random Forest is a supervised learning algorithm. It is an extension of machine learning classifiers which include the bagging to improve the performance of Decision Tree. It combines tree predictors, and trees are dependent on a random vector which is independently sampled. The distribution of all trees are the same. Random Forests splits nodes using the best among of a predictor subset that are randomly chosen from the node itself, instead of splitting nodes based on the variables. The time complexity of the worst case of learning with Random Forests is O(M(dnlogn)) , where M is the number of growing trees, n is the number of instances, and d is the data dimension.

It can be used both for classification and regression. It is also the most flexible and easy to use algorithm. A forest consists of trees. It is said that the more trees it has, the more robust a forest is. Random Forests create Decision Trees on randomly selected data samples, get predictions from each tree and select the best solution by means of voting. It also provides a pretty good indicator of the feature importance.

Random Forests have a variety of applications, such as recommendation engines, image classification and feature selection. It can be used to classify loyal loan applicants, identify fraudulent activity and predict diseases. It lies at the base of the Boruta algorithm, which selects important features in a dataset.

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.

Assumptions:

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.

Algorithm Steps:

It works in four steps

1. Select random samples from a given dataset.
2. Construct a Decision Tree for each sample and get a prediction result from each Decision Tree.
3. Perform a vote for each predicted result.
4. Select the prediction result with the most votes as the final prediction.

4.3.5 DECISION TREE

Decision Tree is a Supervised learning technique that can be used for both classification and regression problems, but mostly it is preferred for solving classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome. In a Decision Tree, there are two nodes, which are the Decision Node and Leaf Node.

Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches. The decisions or the test are performed on the basis of features of the given dataset. It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions. It is called a Decision Tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure. In order to build a tree, we use the CART algorithm, which stands for Classification and Regression

Tree algorithm. A Decision Tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

The Decision Tree Algorithm belongs to the family of supervised machine learning algorithms. It can be used for both a classification problem as well as for a regression problem.

The goal of this algorithm is to create a model that predicts the value of a target variable, for which the decision tree uses the tree representation to solve the problem in which the leaf node corresponds to a class label and attributes are represented on the internal node of the tree.

There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model. Below are the two reasons for using the Decision Tree:

Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.

The logic behind the decision tree can be easily understood because it shows a tree-like structure.

In Decision Tree the major challenge is to identify the attribute for the root node in each level. This process is known as attribute selection. We have two popular attribute selection measures:

1. INFORMATION GAIN

When we use a node in a Decision Tree to partition the training instances into smaller subsets, the entropy changes. Information gain is a measure of this change in entropy.

Entropy is the measure of uncertainty of a random variable, it characterizes the impurity of an arbitrary collection of examples.

The higher the entropy the more the information content.

2. GINI INDEX

Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified. It means an attribute with lower Gini index should be preferred. Sklearn supports “Gini” criteria for Gini Index and by default, it takes “gini” value.

The most notable types of Decision Tree algorithms are

1. IDichotomiser 3 (ID3):

This algorithm uses Information Gain to decide which attribute is to be used to classify the current subset of the data. For each level of the tree, information gain is calculated for the remaining data recursively.

2. C4.5:

This algorithm is the successor of the ID3 algorithm. This algorithm uses either Information gain or Gain ratio to decide upon the classifying attribute. It is a direct improvement from the ID3 algorithm as it can handle both continuous and missing attribute values.

3.Classification and Regression Tree (CART):

It is a dynamic learning algorithm which can produce a regression tree as well as a classification tree depending upon the dependent variable.

Working:

In a Decision Tree, for predicting the class of the given dataset, the algorithm starts from the root node of the tree. This algorithm compares the values of the root attribute with the record (real dataset) attribute and, based on the comparison, follows the branch and jumps to the next node.

For the next node, the algorithm again compares the attribute value with the other sub-nodes and moves further. It continues the process until it reaches the leaf node of the tree. The complete process can be better understood using the below algorithm:

1. Begin the tree with the root node, says S, which contains the complete dataset

2. Find the best attribute in the dataset using Attribute Selection Measure (ASM)

3. Divide the S into subsets that contains possible values for the best attributes

4. Generate the Decision Tree node, which contains the best attribute

5. Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and call the final node as a leaf node

4.3.6. KNN

K-Nearest Neighbors (KNN) is a popular machine learning algorithm used for classification and regression tasks. In the context of ASD detection, KNN can be employed as a classification algorithm to predict whether an individual is likely to have ASD or not based on their input features.

Working Principle of KNN:

KNN is a non-parametric algorithm that makes predictions based on the similarity between data points. The algorithm operates as follows:

1.Training Phase:

a. The algorithm is provided with a labeled dataset, consisting of input feature vectors and their corresponding class labels (ASD or non-ASD).

b. During the training phase, KNN stores the entire training dataset in memory, forming the training set.

2. Prediction Phase:

a. When a new, unlabeled instance is presented for prediction, the algorithm measures the similarity between this instance and the instances in the training set.

b. The similarity is typically computed using distance metrics such as Euclidean distance or cosine similarity.

c. The algorithm identifies the K nearest neighbors (i.e., the K instances with the smallest distances) from the training set to the input instance.

d. The class label of the input instance is determined by majority voting among the K nearest neighbors.

The KNN algorithm has one key parameter

K: The number of nearest neighbors to consider for classification. It is typically chosen based on cross-validation or domain knowledge

CHAPTER 5

EXPERIMENT ANALYSIS

5.1 SYSTEM CONFIGURATION

5.1.1 HARDWARE REQUIREMENTS

Processer	:	Any Update Processer
Ram	:	Min 4GB
Hard Disk	:	Min 100GB

5.1.2 SOFTWARE REQUIREMENTS

Operating System	:	Windows family
Technology	:	Python3.7
IDE	:	Jupiter notebook

5.2 SAMPLE CODE

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC

# Load the dataset
asd = pd.read_csv("../input/Toddler Autism dataset July 2018.csv")
print("Dataset loaded...")
asd.head()
asd.describe()
asd.drop(['Case_No', 'Who completed the test'], axis=1, inplace=True)
asd.columns
asd.dtypes

# Plot countplot and correlation heatmap
sns.countplot(x='Class/ASD Traits ', data=asd)
corr = asd.corr()
plt.figure(figsize=(15, 15))
sns.heatmap(data=corr, annot=True, square=True, cbar=True)

# Plot countplot with hue
sns.countplot(x='Ethnicity', hue='Sex', data=asd)
plt.xticks(rotation=90)
```

```
plt.figure(figsize=(16, 8))
sns.countplot(x='Ethnicity', data=asd)

# Preprocess the data
le = LabelEncoder()
columns = ['Ethnicity', 'Family_mem_with_ASD', 'Class/ASD Traits ', 'Sex', 'Jaundice']
for col in columns:
    asd[col] = le.fit_transform(asd[col])

asd.dtypes
X = asd.drop(['Class/ASD Traits '], axis=1)
Y = asd['Class/ASD Traits ']
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.20, random_state=7)
print('Processed...')

# Train a logistic regression model
logreg = LogisticRegression()
logreg.fit(x_train, y_train)
preds = logreg.predict(x_test)
logreg.score(x_train, y_train)

# Classification report and confusion matrix for logistic regression
print("Logistic Regression:")
print(classification_report(y_test, preds))
print("Confusion Matrix:")
print(confusion_matrix(y_test, preds))

# KNN
knn = KNeighborsClassifier(n_neighbors=27)
knn.fit(x_train, y_train)
pred_knn = knn.predict(x_test)
print("KNN:")
print(classification_report(y_test, pred_knn))
print("Confusion Matrix:")
```

```
print(confusion_matrix(y_test, pred_knn))

# Finding the optimum n_neighbors for KNN
error_rate = []
for i in range(1, 40):
    knn = KNeighborsClassifier(n_neighbors=i)
    knn.fit(x_train, y_train)
    pred = knn.predict(x_test)
    error_rate.append(np.mean(y_test != pred))
plt.figure(figsize=(10, 10))
plt.plot(range(1, 40), error_rate, color='blue', linestyle='dashed', marker='o',
markerfacecolor='red')

# Decision tree
dtree = DecisionTreeClassifier()
dtree.fit(x_train, y_train)
pred_dtree = dtree.predict(x_test)
print("Decision Tree:")
print(classification_report(y_test, pred_dtree))
print("Confusion Matrix:")
print(confusion_matrix(y_test, pred_dtree))

# Evaluation of other models
models = [
    ('Logistic Regression:', LogisticRegression()),
    ('Naive Bayes :', GaussianNB()),
    ('SVM :', SVC())
]
for name, model in models:
    model.fit(x_train, y_train)
    pred = model.predict(x_test).astype(int)
    print(name, accuracy_score(y_test, pred))
    print("Confusion Matrix:")
    print(confusion_matrix(y_test, pred))
```

5.3 DATA SET DETAILS

Of the 19 attributes available in the dataset,14 attributes are considered for the prediction of the output.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
1	Case_No	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Age_Mons	Qchat-10-	Sex	Ethnicity	Jaundice	Family_mem_with_A		Class/ASD	Who completed the test		
2	1	0	0	0	0	0	0	1	1	0	1	28	3	f	middle eas	yes	no		No	family member		
3	2	1	1	0	0	0	1	1	0	0	0	36	4	m	White Eurc	yes	no		Yes	family member		
4	3	1	0	0	0	0	0	1	1	0	1	36	4	m	middle eas	yes	no		Yes	family member		
5	4	1	1	1	1	1	1	1	1	1	1	24	10	m	Hispanic	no	no		Yes	family member		
6	5	1	1	0	1	1	1	1	1	1	1	20	9	f	White Eurc	no	yes		Yes	family member		
7	6	1	1	0	0	1	1	1	1	1	1	21	8	m	black	no	no		Yes	family member		
8	7	1	0	0	1	1	1	0	0	1	0	33	5	m	asian	yes	no		Yes	family member		
9	8	0	1	0	0	1	0	1	1	1	1	33	6	m	asian	yes	no		Yes	family member		
10	9	0	0	0	0	0	0	1	0	0	1	36	2	m	asian	no	no		No	family member		
11	10	1	1	1	0	1	1	0	1	1	1	22	8	m	south asia	no	no		Yes	Health Care Professional		
12	11	1	0	0	1	0	1	1	0	1	1	36	6	m	Hispanic	yes	yes		Yes	family member		
13	12	1	1	1	1	0	1	1	1	0	1	17	8	m	middle eas	yes	no		Yes	family member		
14	13	0	0	0	0	0	0	0	0	0	0	25	0	f	middle eas	yes	no		No	family member		
15	14	1	1	1	1	0	0	1	0	1	1	15	7	f	middle eas	yes	no		Yes	family member		
16	15	0	0	0	0	0	0	0	0	0	0	18	0	m	middle eas	no	no		No	family member		
17	16	1	1	1	0	1	0	1	1	0	1	12	7	m	black	no	no		Yes	family member		
18	17	0	0	0	0	0	0	0	0	0	0	36	0	m	middle eas	no	yes		No	family member		
19	18	1	1	1	0	1	1	1	1	0	1	12	8	f	middle eas	yes	no		Yes	family member		
20	19	1	0	0	0	1	0	0	0	0	1	29	3	f	middle eas	no	no		No	family member		
21	20	1	1	1	0	1	0	1	1	0	1	12	7	f	black	no	no		Yes	family member		
22	21	1	0	0	1	1	1	1	1	1	0	36	7	m	middle eas	no	no		Yes	family member		
23	22	1	0	1	1	1	1	1	0	1	0	36	7	m	middle eas	no	no		Yes	family member		
24	23	1	0	1	1	0	1	0	1	1	1	36	7	m	Native Ind	yes	yes		Yes	Health Care Professional		
25	24	1	1	1	0	1	1	0	1	1	0	36	7	m	middle eas	yes	yes		Yes	family member		
26	25	1	1	1	1	1	1	1	1	1	0	22	9	m	White Eurc	no	no		Yes	family member		
27	26	0	0	0	0	0	0	0	0	0	0	24	0	f	middle eas	no	no		No	family member		
28	27	1	1	0	1	1	1	1	1	0	1	36	8	m	middle eas	no	no		Yes	family member		
29	28	1	1	1	1	1	1	1	1	1	1	35	10	m	Others	yes	no		Yes	family member		
30	29	0	0	1	1	1	0	1	1	1	1	36	7	m	middle eas	no	yes		Yes	family member		

Dataset Attribute

Input Dataset Attributes

A1-A10 (value 1 : yes; value 0 : No)

Age_mon (age in months)

Qchat-10 Score (Number of Yes answered questions from A1-A10)

Sex (value 1:Male ; value 0: Female)

Ethnicity (Which part of the world they came from)

Jaundice (value 1: yes; value 0: No)

Family_mem_with_ASD(value 1 :yes ; value 0 : No)

Class/ASD Traits(value 1: yes; value 0 : No)

5.4 PERFORMANCE ANALYSIS

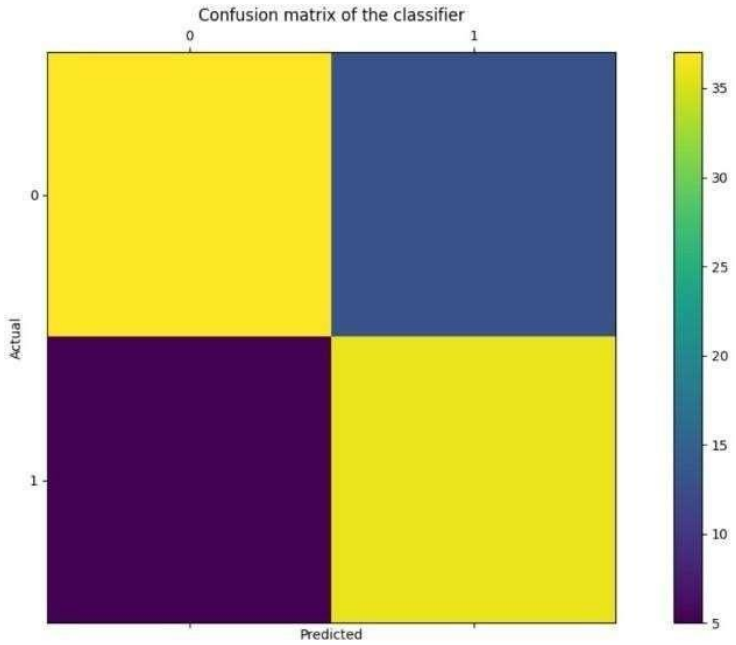
In this project, various machine learning algorithms like SVM, Naive Bayes, Decision Tree, Random Forest, Logistic Regression, KNN are used to predict ASD. Toddlers ASD dataset, has a total of 19 attributes, out of those only 14 attributes are considered for the prediction of ASD. Various attributes of the patient like gender, age, A1-A10 Questions, age, Qchat-10 score, jaundice are considered for this project. The accuracy for individual algorithms has to measure and whichever algorithm is giving the best accuracy, that is considered for the heart disease prediction. For evaluating the experiment, various evaluation metrics like accuracy, confusion matrix, precision, recall, and f1-score are considered.

Accuracy- Accuracy is the ratio of the number of correct predictions to the total number of inputs in the dataset. It is expressed as:

$$\text{Accuracy} = (TP + TN) / (TP+FP+FN+TN)$$

Confusion Matrix- It gives us a matrix as output and gives the total

Performace of the system.



Confusion Matrix

Where,

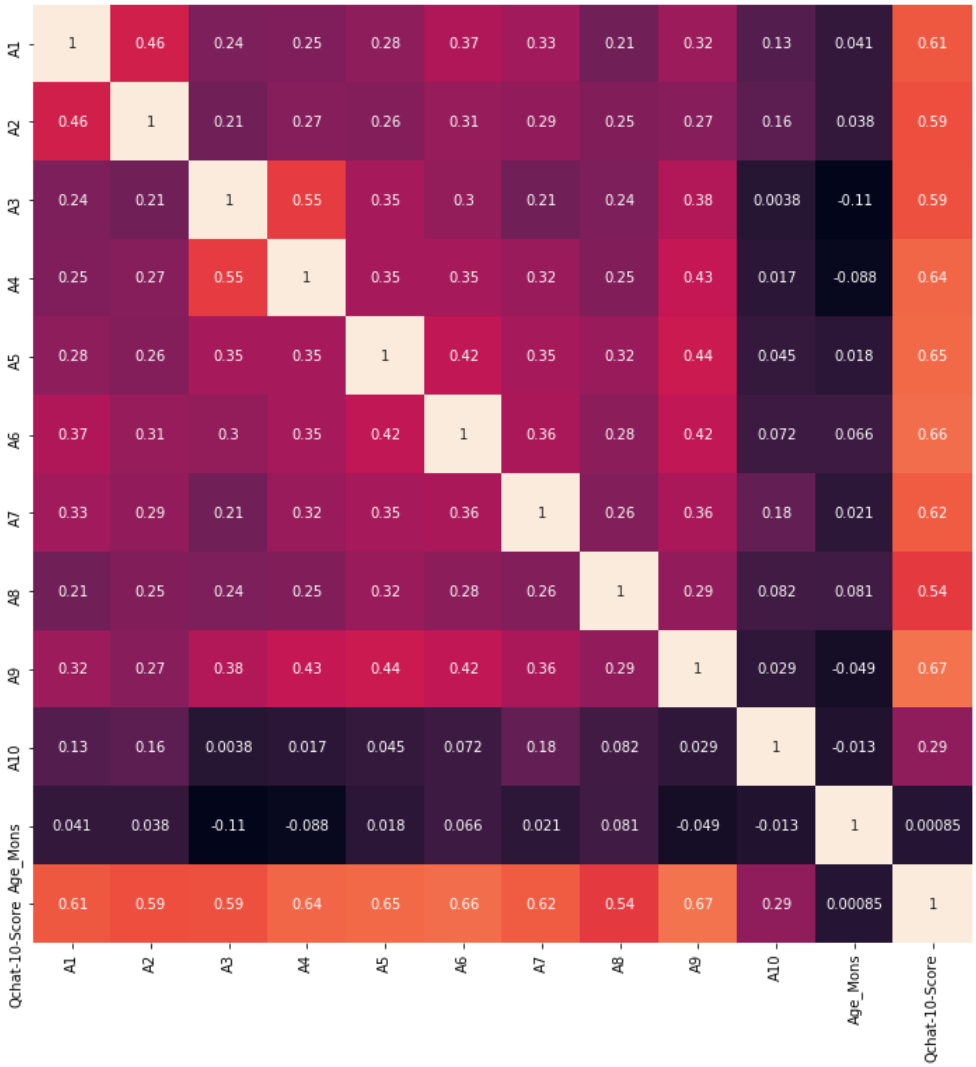
TP: True Positive

FP: False Positive

FN: False Negitive

TN : True Negitive

Correlation Matrix: The correlation matrix in machine learning is used for feature selection. It represents dependency between various attributes:



Correlation matrix

Precision- It is the ratio of correct positive results to the total number of positive results predicted by the system.

Recall-It is the ratio of correct positive results to the total number of positive results predicted by the system.

F1 Score-It is the harmonic mean of Precision and Recall. It measures the test accuracy. The range of this metric is 0 to 1.

5.5 PERFORMANCE MEASURE

```
Logistic Regression: 0.957345971563981
Naive Bayes        : 0.981042654028436
SVM                : 0.981042654028436
KNN                : 0.957345971563981
Descion Tree       : 0.8672985781990521
```

5.6 RESULT

5.6.1 INPUTS AND OUTPUTS

Input 1:

```
prediction = clf.predict([[1,0,0,0,1,0,1,0,0,0,36,5,1,1,0,0]])
```

Output 1:

```
Yes, You have ASD
```

Input 2:

```
prediction = clf.predict([[1,0,0,1,0,0,1,0,0,0,36,3,1,1,0,0]])
```

Output 2:

```
No, You dont have ASD
```

CHAPTER 6
CONCLUSIONAND FUTURE ENHANCEMENT

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

Autism is a major dissease in India and throughout the world, application of promising technology like machine learning to the initial prediction of ASD will have a profound impact on society. The early prognosis of ASD can aid in making decisions on lifestyle changes in high-risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine. The number of people facing ASD is on a raise each year. This prompts for its early diagnosis and treatment. The utilization of suitable technology support in this regard can prove to be highly beneficial to the medical fraternity and patients. In this paper, the six different machine learning algorithms used to measure the performance are SVM, Decision Tree, Random Forest, Naïve Bayes, Logistic Regression,and KNN applied on the dataset.

The expected attributes leading to heart disease in patients are available in the dataset which contains 19 features and 17 important features that are useful to evaluate the system are selected among them. If all the features taken into the consideration then the efficiency of the system the author gets is less. To increase efficiency, attribute selection is done. In this n features have to be selected for evaluating the model which gives more accuracy. The correlation of some features in the dataset is almost equal and so they are removed. If all the attributes present in the dataset are taken into account then the efficiency decreases considerably.

All the seven machine learning methods accuracies are compared based on which one prediction model is generated. Hence, the aim is to use various evaluation metrics like confusion matrix, accuracy,

precision, recall, and f1-score which predicts the disease efficiently. Comparing all seven the extreme DECISION TREE classifier gives the highest accuracy of 86%.

CHAPTER 7

APPENDIX

CHAPTER 7
APPENDIX

PYTHON

Python is an interpreted, high-level, general purpose programming language created by Guido Van Rossum and first released in 1991, Python's design philosophy emphasizes code Readability with its notable use of significant White space. Its language constructs and object oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

SKLEARN

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistent interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

NUMPY

NumPy is a library for the python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric, was originally created by Jim with contributions from several other developers. In 2005, Travis created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open source software and has many contributors.

MATPLOTLIB

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a statemachine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged.

SEABORN

Seaborn is a Python data visualization library based on matplotlib. It provides a high- level interface for drawing attractive and informative statistical graphics. Seaborn is a library in Python predominantly used for making statistical graphics. Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

REFERENCE

1. López-Bonilla, J. M., Cuevas, E., Travieso, C. M., & Orozco-Arroyave, J. R. (2020). Machine Learning-Based Methods for Autism Spectrum Disorder Diagnosis and Prognosis: A Systematic Review. Frontiers in Psychology, 11, 1566.

2. Gilani, S. Z., Rooney, K., & Sharda, B. (2019). Deep Learning Approaches for Autism Spectrum Disorder Classification: A Review. Frontiers in Psychiatry, 10, 500.

3. Roshni, M. S., Sadasivan, R. S., Nair, M. S., & Soman, K. P. (2020). Automatic Detection of Autism Spectrum Disorder Using Machine Learning and EEG Signal Processing: A Systematic Review. International Journal of Information Technology, 12(2), 381-388.

4. Lajiness-O'Neill, R., Beaulieu, L., Titus, J. B., & Asp, E. (2018). A Review of Machine Learning Applications in Autism Spectrum Disorders: A Systematic Review. Review Journal of Autism and Developmental Disorders, 5(3), 240-253.

5. Bone, D., Bishop, S. L., Black, M. P., Goodwin, M. S., Lord, C., & Narayanan, S. S. (2020). Machine Learning in Autism Research: Advancements, Challenges, and Future Directions. Nature Partner Journals Digital Medicine, 3, 41.

6. Chang, Y. C., Tsai, C. H., & Lai, M. C. (2019). Autism and Machine Learning: A Review of the Literature. Frontiers in Psychiatry, 10, 517.

7. Wall, D. P., Kosmicki, J., DeLuca, T. F., Harstad, E., Fusaro, V. A., & Use, S. (2012). Use of machine learning to shorten observation-based screening and diagnosis of autism. Translational Psychiatry, 2(4), e100.

8. So, J. R., & Nam, H. S. (2019). Machine Learning Approaches for Autism Spectrum Disorder Detection and Classification: A Review. International Journal of Environmental Research and Public Health, 16(5), 894.

9. Ghorai, S., & Bhowmik, M. (2020). Machine Learning-Based Autism Spectrum Disorder Detection: A Review. IEEE Access, 8, 13921-13937.

10. Ji, X., & Huang, L. (2020). Machine Learning Techniques for Autism Spectrum Disorder Diagnosis: A Review. Frontiers in Psychology, 11, 1433.

11. Shetty, A. C., Jose, J., & Manjunath, K. N. (2020). Machine Learning Approaches for Autism Spectrum Disorder Diagnosis: A Review. Journal of Medical Systems, 44(10), 186.

12. Liang, S., Shang, Y., & Wang, Y. (2020). Machine Learning Techniques for Autism Spectrum Disorder Screening and Diagnosis: A Review. Journal of Healthcare Engineering, 2020, 8886952.

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 07, July-2022

Detection of Autism Spectrum Disorder using Machine Learning

R. Bhuvaneshwari
Department of Information Technology
Madras Institute of Technology, Anna University
Chennai, India

Pranusha S Bavan
Department of Information Technology Madras Institute of Technology, Anna University
Chennai, India

Dr. P. Lakshmi Harika
Department of Information Technology Madras Institute of Technology, Anna University
Chennai, India

N. Mathubaala
Department of Information Technology Madras Institute of Technology, Anna University
Chennai, India

Dr. M R Sumalatha
Professor
Department of Information Technology Madras Institute of Technology,
Anna University Chennai, India

***Abstract*—Autistic Spectrum Disorder (ASD) is a severe neuro- logical condition that affects the entire brain system and whichin turn impacts the cognitive, emotional, social, and physical health of the individual. They experience difficulty in socializing and communicating with others. They are always in need of support either from parents, relatives, or friends to guide them. Unfortunately, there is no cure for autism but early detection can help in better treatment. A person's behavioural behaviours can be used to diagnose autism disorder. This method of diagnosisis time-consuming and ineffective for early detection of autism.. Therefore, there is a need for time-efficient and low-cost ASD screening to help individuals to decide whether they should undergo a clinical diagnosis and seek treatment. Therefore, we propose a machine learning-based, time-efficient solution to detect autism.**

I. INTRODUCTION

Autism is one of the serious issues for humankind. Which affects the overall behavior of a person. It will also affect the emotional, cognitive, social, and physical health of an indi- vidual. It can be witnessed in individuals irrespective of their age (toddler, child, teenager, adults, and senior citizens). The autism screening tests are both time- and money-consuming.A technique based on machine learning is proposed to help the person to decide whether to get a formal clinical diagnosis or not based on the prediction/prognosis of the Machine Learning model. ASD is not curable but early detection is helpful to determine better treatment methodology. This proves to be of great help and can significantly reduce healthcare costs.

II. RELATED WORK

1. Simple neural network models have been used by Madhura Ingalhalikar, Sumeet Shinde, Arnav Karmarkar, Archith Rajan, Dr. Rangaprakash, and Gopikrishna Deshpande (2021) to categorise their models. When opposed to sophisticated models, neural networks have made it easier to achieve greater accuracy on harmonised data. It was crucial to use ablation analysis to describe the most discriminative sub-networks that were directly linked to the clinical markers of autism.

2. Five classification algorithms were utilised by Md. Fazle Rabbi, S. M. Mahedy Hasan, Arifa Islam Champa, and Md. Asif Zaman (2021) to identify autism in youngsters, and the best accurate model was identified by comparing them. Comparing various evaluation metrics, CNN algorithm outperformed all other algorithms. The dataset used consists of 2940 images of children.

3. Classification Techniques have been utilised by M.
S. Mythili and A. R. Mohamed Shanavas (2014) to research ASD. This paper's primary goals were to identify autism and its severity degrees. SVM and neural networks were two of the classification algorithms employed. WeKA tools and fuzzy techniques were also employed to examine the social interaction and conduct of the students.

4. In order to identify autism, J. A. Kosmicki1, V. Sochat,
M. Duda, and D.P. Wall (2015) utilised a strategy of searching for the smallest collection of features. To assess the clinical assessment of ASD, the authors employed a machine learning methodology. The ADOS was applied to children's behaviour that fell inside the autism spectrum. In this research, eight distinct machine learning algorithms from ADOS's four modules were applied. Stepwise backward feature recognition on score sheets from 4540 people was another aspect of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 07, July-2022

the study. With an overall accuracy of 98.27 percent and 97.66 percent, respectively, it used 9 out of the 28 behaviours from module 2 and it had employed 12 out of the 28 behaviours from module 3 to detect an ASD risk.

5. Fadi Thabtah has suggested an ASD screening methodthat makes use of machine learning adaption and the DSM-5 (2017). In this article, the researcher discussed the benefits and drawbacks of the ASD Machine Learning categorization. He has also made an effort to call attention to the problems with current ASD screening techniques and the way they consistently rely on the DSM-IV rather than the DSM-5 manual.

III. SYSTEM DESIGN AND ARCHITECTURE

A. SYSTEM DESIGN

The architecture follows the following flow:

• Collection of Dataset The Autism Screening Datasets are used and include age groupings for adults, toddlers, and children. Datasets were gathered via the UCI Repository and Kaggle.

• Data Pre-processing The raw data will be cleaned by data pre-processing.

• Model development and Evaluation After pre-processing, the dataset gets split into testing and training sets. Multiple classifiers are developed using prominent machine learning algorithms (Decision Tree, Naive Bayes, KNN and SVM). In the training phase, the training data is given to train the classifier. In the testing phase, class predictions are made on the test dataset. The classifiers are evaluated based on their performances in diagnosing autism. The test data is used for model evaluation to evaluate a model based on its performance and accuracy. The accuracies of the classification algorithms are compared. Using Voting classifier, a hybrid ensemble machine learning model is developed.
Accuracy and recall are the metrics that are computed during the model evaluation process. The performance metric accuracy is the ratio of correctly predicted observationsto all observations. Recall is the ratio of correctly predicted positive positive observations to all positive observations(output label 1). The voting classifier is evaluatedusing RepeatedStratifiedKfold cross validation. From cross validation, accuracy scores and recall scores are obtained.

• Model deployment The model which has the highest accuracy is then deployed using Flask. In the frontend theuser has to input their basic form details and according to the age category, the user attempts the ASD Screening test. Atthe backend, the user's input is pre-processed. The model is loaded from the pickle file and performs prediction with the processed input. The model's prediction is displayed at the user's front end.

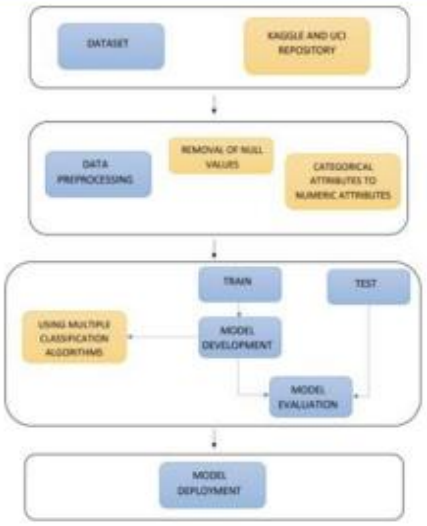


Fig. 1. System Architecture

IV. PROPOSED WORK

A. Pre-processing

Three datasets for different age categories of Adult, Chil- dren and Toddlers are used. The datasets are obtained from Kaggle and UCI repository. There are nearly 20 attributes in each of the dataset which consist of categorical, continuous, and binary values. The Class/ASD output label indicates if a person has ASD (1) or not (0). Firstly, the raw data is pre- processed. Unnecessary columns are dropped, column names renamed so that they are uniform across all datasets., Null values removed. Repository.

B. Encoding

Encoding is performed on categorical values to convert string values into numerical values. For this purpose, Label Encoder is used. The label encoder is saved as a pickle filefor further use. This pickle file is further used at the backend to encode the input obtained from the user.

C. Model development

After pre-processing, the dataset is split into testing set and training set. A train test split ratio of 80:20 is used. Multiple classification algorithms are used to develop the model. The classification algorithms used are K-nearest neighbours, Sup- port Vector Machine, Decision Tree, and Na¨ıve bayes. Using these classification algorithms, a hybrid ensemble machine learning model is developed using Voting classifier. Both hardvoting and soft voting methods are used.

The Voting Classifier is an estimator that combines repre- sentations of many classification techniques along with indi- vidual confidence weights. The Voting estimator, which was created by integrating various classification models, is a pow- erful meta-classifier that

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 07, July-2022

effectively counteracts the limitations of the individual classifiers on a given dataset. Voting classifier assigns a class label to a record based on a majority vote and weights applied to the class or class probabilities.

D. Model Evaluation

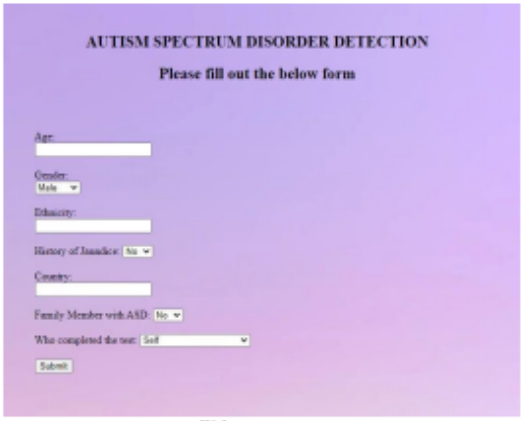
The Voting classifier is evaluated using RepeatedStrati- fiedKFold cross validation that repeats Stratified K-Fold n times. Three repeats of stratified 10-fold cross-validation is performed. KFold: Split dataset into k consecutive folds. Stratified: The folds are made by preserving the percentage of samples for each class. Repeats: Number of times cross- validator needs to be repeated.

The metrics that are calculated are: Accuracy and recall The average of all the accuracies and recall scores are computed and the mean accuracy and recall score is produced.

E. Model Deployment

The flask webapp starts with the form page that asks the user to enter the required input. After the user submits the input through a form, the user attempts the ASD Screening test. After user submits the test, backend receives the input data. The user's input is first processed before performing the prediction. The encoder is loaded from the pickle file and is applied over the user's input. The model is loaded from the pickle file and performs prediction with the processed input. The model's prediction, the result of whether the user is autistic or not, and the class of the prediction are computed and displayed at the user's front end.

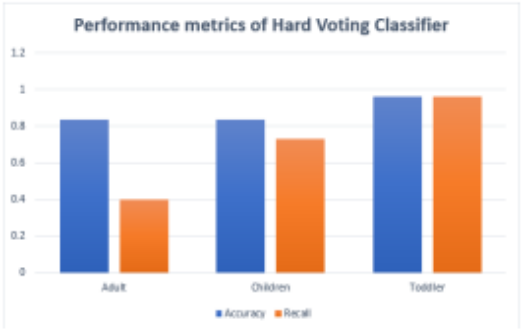
Fig. 2.



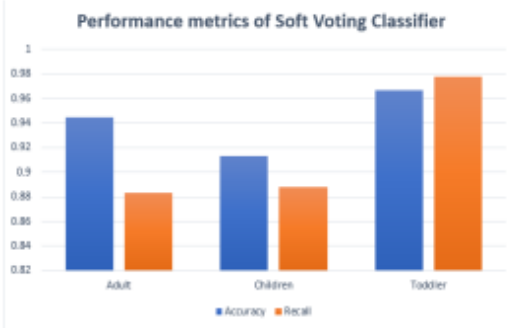
Web page

V. RESULTS

The performance metrics of the Hard Voting Classifier obtained are: accuracy of 83.76% for adults, accuracy of 84.48% for children, and accuracy of 96.39% for toddlers. The Recall values of the Hard Voting classifiers are 39.88% for adults, 73.11% for children, 96.11% for toddlers.



The performance metrics of the Soft Voting Classifier ob- tained are: accuracy of 94.45% for adults, 91.30% for children and 96.71% for toddlers. The Recall values of the Soft Voting classifiers are 88.35% for adults, 88.77% for children, 97.85% for toddlers.



In comparison, the Soft Voting Classifier has better perfor-mance metrics than Hard Voting Classifier.

VI. CONCLUSION

It is inferred that Soft Voting classifier performs better than Hard Voting classifier. On comparing the ensemble model with the classification algorithms it is found that, for adult dataset, Naive Bayes classifier has slightly higher accuracy than the ensemble model. For other datasets, Soft Voting classifier has the maximum accuracy than other algorithms. If the person is found to be autistic, they are advised to seek a proper clinical diagnosis and are recommended support institutions that provide help to autistic people. A list of organizations such as schools for kids with special needs and other relevant institutions for all age groups are suggested.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 07, July-2022

REFERENCES

[1] M. Ingalhalikar, S. Shinde, A. Karmarkar, A. Rajan, D. Rangaprakash and G. Deshpande, "Functional Connectivity-Based Prediction of Autism on Site Harmonized ABIDE Dataset," in IEEE Transactions on Biomedical Engineering, vol. 68, no. 12, pp. 3628-3637, Dec. 2021,doi: 10.1109/TBME.2021.3080259.

[2] M. F. Rabbi, S. M. M. Hasan, A. I. Champa and M. A. Zaman, "A Convolutional Neural Network Model for Early-Stage Detection of Autism Spectrum Disorder," 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), 2021, pp. 110-114, doi: 10.1109/ICICT4SD50815.2021.9397020.

[3] Mythili, M. S., and AR Mohamed Shanavas. "An Analysis of students' performance using classification algorithms." IOSR Journal of ComputerEngineering 16.1 (2014): 63-69.

[4] Kosmicki, J. A., Sochat, V., Duda, M., & Wall, D. P. (2015). Searching for a minimal set of behaviors for autism detection through feature selection-based machine learning. Translational psychiatry, 5(2), e514- e514.

[5] Thabtah, Fadi. "Autism spectrum disorder screening: machine learning adaptation and DSM-5 fulfillment." Proceedings of the 1st International Conference on Medical and health Informatics 2017. 2017.

[6] Vaishali, R., and R. Sasikala. "A machine learning based approach to classify autism with optimum behaviour sets." International Journal of Engineering & Technology 7.4 (2018): 18.

[7] Vakadkar, Kaushik, Diya Purkayastha, and Deepa Krishnan."Detection of Autism Spectrum Disorder in Children Using Machine Learning Techniques." SN Computer Science 2.5 (2021): 1-9.

[8] Cavus, Nadire, Abdulmalik A. Lawan, Zurki Ibrahim, Abdullahi Dahiru, Sadiya Tahir, Usama Ishaq Abdulrazak, and Adamu Hussaini. "A systematic literature review on the application of machine-learning models in behavioral assessment of autism spectrum disorder." Journal of Personalized Medicine 11, no. 4 (2021): 299.

[9] S.B. Shuvo, J. Ghosh and A. S. Oyshi, "A Data Mining Based Approach to Predict Autism Spectrum Disorder Considering Behavioral Attributes," 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1-5, doi: 10.1109/ICCCNT45670.2019.8944905.

[10] J. Baio, "Prevalence of autism spectrum disorders: Autism and developmental disabilities monitoring network, united states, 2008. morbidity and mortality weekly report.61." Centers for Disease Control and Prevention, 2012.

[11] S. E. Bryson, L. Zwaigenbaum, and W. Roberts, "The early detection of autism in clinical practice," Paediatrics & child health, vol. 9, no. 4, pp. 219– 221, 2004.

[12] F. Thabtah and D. Peebles, "A new machine learning model based on induction of rules for autism detection," and "A complete guide to the random forest algorithm," Built In, vol. 16, 2019.

[13] Haishuai Wang, Li LiLianhua Chi, Ziping Zhao, "Autism Screening Using Deep Embedding Representation," International Conference on Computational Science, Lecture Notes in Computer Science, vol 11537, pp. 160-173, jun 2019.

[14] Muhammad Nazrul Islam, Kazi Shahrukh Omar, Prodipta Mondal, Nabila Shahnaz Khan, "A Machine Learning Approach to Predict Autism Spectrum Disorder," International Conference on Electrical, Computer and Commmunication Engineering, feb 2019.