

A Project report on

AUTISM SPECTRUM DISORDER PREDICTION

Submitted in partial fulfillment of the requirements

for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

By

D. SUDHARSHAN REDDY (194G1A05B2)

S. SHAHEEN TAJ (194G1A0599)

A. NIKHITA (194G1A0568)

B. VIJAYA (194G1A05C5)

Under the Guidance of

Dr. B. Harichandana, M.Tech, Ph.D

Associate Professor



Department of Computer Science & Engineering

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)

Rotarypuram Village, B K Samudram Mandal, Ananthapuramu - 515701

2022-2023

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

(AUTONOMOUS)

(Affiliated to JNTUA, Accredited by NAAC with 'A' Grade, Approved by AICTE, New Delhi & Accredited by NBA (EEE, ECE & CSE)

Rotarypuram Village, BK Samudram Mandal, Ananthapuramu-515701

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING



Certificate

This is to certify that the Project report entitled **AUTISM SPECTRUM DISORDER PREDICTION** is the bonafide work carried out by **D. Sudharshan Reddy, S. Shaheen Taj, A. Nikhita and B. Vijaya** bearing Roll Number **194G1A05B2, 194G1A0599, 194G1A0568 and 194G1A05C5** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering** during the academic year 2022 - 2023.

Project Guide

Dr. B. Harichandana, M.Tech, Ph.D
Associate Professor

Head of the Department

Mr. P. Veera Prakash, M.Tech, (Ph.D)
Assistant Professor

Date:

External Examiner

Place: Rotarypuram

ACKNOWLEDGEMENTS

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose constant guidance and encouragement crowned our efforts with success. It is a pleasant aspect that we have now the opportunity to express our gratitude for all of them.

It is with immense pleasure that we would like to express our indebted gratitude to our Guide **Dr. B. Harichandana, Associate Professor, Computer Science & Engineering**, who has guided us a lot and encouraged us in every step of the project work. We thank her for the stimulating guidance, constant encouragement and constructive criticism which have made possible to bring out this project work.

We express our deep felt gratitude to **Dr. B. Harichandana, Associate Professor** and **Mrs. S. Sunitha, Assistant Professor**, project coordinators for giving valuable guidance and unstinting encouragement enabled us to accomplish our project successfully in time.

We are very much thankful to **Mr. P. Veera Prakash, Assistant Professor & Head of the Department, Computer Science & Engineering**, for his kind support and for providing necessary facilities to carry out the work.

We wish to convey our special thanks to **Dr. G. Bala Krishna, Principal of Srinivasa Ramanujan Institute of Technology** for giving the required information in doing our project work. Not to forget, we thank all other faculty and non- teaching staff, and our friends who had directly or indirectly helped and supported us in completing our project in time.

We also express our sincere thanks to the Management for providing excellent facilities.

Finally, we wish to convey our gratitude to our families who fostered all the requirements and facilities that we need.

Project Associates

194G1A05B2 SUDHARSHAN REDDY D

194G1A0599 SHAHEEN TAJ S

194G1A0568 NIKHITA A

194G1A05C5 VIJAYA B

DECLARATION

We, Mr D. Sudharshan Reddy with reg no: 194G1A05B2 , Ms S. Shaheen Taj with reg no: 194G1A0599, Ms A. Nikhita with reg no: 194G1A0568 and Ms B. Vijaya with reg no: 194G1A05C5 students of SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY, Rotarypuram hereby declare that the dissertation entitled “**AUTISM SPECTRUM DISORDER PREDICTION**” embodies the report of our project work carried out by us during IV year Bachelor of Technology under the guidance of **Dr. B. Harichandana**, M.Tech, Ph.D. Department of CSE, **SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY**, and this work has been submitted for the partial fulfillment of the requirements for the award of the Bachelor of Technology degree.

The results embodied in this project have not been submitted to any other University or Institute for the award of any Degree or Diploma.

D. SUDHARSHAN REDDY

Reg no: 194G1A05B2

S. SHAHEEN TAJ

Reg no: 194G1A0599

A. NIKHITA

Reg no: 194G1A0568

B. VIJAYA

Reg no: 194G1A05C5

CONTENTS

Page No.

List of Figures		vii
List of Tables		viii
List of Abbreviations		ix
Abstract		x
Chapter 1	Introduction	1
	1.1 Problem Statement	3
	1.2 Objectives	3
Chapter 2	Literature Survey	4
Chapter 3	Methodology	6
	3.1 Machine Learning	7
	3.2 Basic Process of Machine Learning	9
	3.3 Algorithms Used	10
Chapter 4	System Requirements Specification	14
	4.1 Functional Requirements	15
	4.2 Basic Requirements	16
	4.3 Non-Functional Requirements	17
	4.4 Python Libraries	18
	4.5 Hardware Requirements	20
	4.6 Software Requirements	23
Chapter 5	System Analysis and Design	26
	5.1 System Architecture	26
	5.2 UML Diagrams	27

Chapter 6	Implementation	32
	6.1 Dataset	33
	6.2 Data Collection	34
	6.3 Data Pre-Processing	37
	6.3.1 Data Cleaning	37
	6.3.2 Data Integration	37
	6.3.3 Data Reduction	37
Chapter 7	Testing	39
	7.1 Unit Testing	39
	7.2 Integration Testing	39
	7.3 Validation Testing	40
	7.4 System Testing	40
	7.5 Manual Testing	41
	7.6 Performance Testing	41
	7.7 Compatability Testing	41
	7.8 White Box Testing	41
Chapter 8	Results	43
Conclusion		49
References		50

List of Figures

Fig No.	Description	PageNo
Fig.3.1	Types of Machine Learning	7
Fig.3.2	Process of Supervised Learning	8
Fig.3.3	Process of Unsupervised Learning	8
Fig.3.4	Reinforcement Learning	9
Fig.3.5	Basic process of Machine Learning	10
Fig.3.6	Working of Random Forest algorithm	12
Fig.3.7	Working of SVM algorithm	13
Fig.4.1	Types of Requirements in SRS	15
Fig.4.2	Processor	21
Fig.4.3	Hard Disk	22
Fig.4.4	RAM	22
Fig.4.5	Jupyter Notebook icon	23
Fig.4.6	Google Colab icon	24
Fig.4.7	Anaconda Navigator	24
Fig.4.8	Spyder IDE	25
Fig.4.9	Python icon	25
Fig.5.1	System Architecture	26
Fig.5.2	Use Case Diagram	28
Fig.5.3	Class Diagram	29
Fig.5.4	Sequence Diagram	30
Fig.5.5	Activity Diagram	31
Fig.6.1	Dataset	33
Fig.6.2	Importing Libraries	34
Fig.6.3	Loading the Dataset	35
Fig.6.4	Example of Correlation matrix	36
Fig.8.1	Confusion Matrix for SVM	43
Fig.8.2	Confusion Matrix for Random Forest	44
Fig.8.3	Accuracy Results	44
Fig.8.4	Models and their accuracy	45
Fig.8.5	Comparision of Models	45
Fig.8.6	ASD Prediction Web App UI-1	46
Fig.8.7	ASD Prediction Web App UI-2	47
Fig.8.8	Final output page of ASD Prediction	48

List of Tables

Table.No	Title	Page No.
1.1	Details of variables mapping to the Q-Chat-10	2

List of Abbreviations

ASD	Autism Spectrum Disorder
IDE	Integrated Development Environment
SVM	Support Vector Machine
RF	Random Forest
SVN	Support Vector Network
ADOS	Autism Diagnostic Observation Schedule

Abstract

Machine Learning based behavioral analytics are needed to develop accurate prediction models for detecting the risk of autism faster than the traditional diagnostic methods. Autism Spectrum Disorder is a neuro-disorder in which a person has a lifelong effect on interaction and communication with others. Autism can be diagnosed at any stage in once life and is said to be a “behavioral disease”. ASD problem starts with childhood and continues to keep going on into adolescence and adulthood. In the current times, clinical standardized tests are the only methods which are being used, to diagnose ASD. This not only requires prolonged diagnostic time but also faces a steep increase in medical costs. To improve the precision and time required for diagnosis, machine learning techniques are being used to complement the conventional methods. Early detection and treatment are most important steps to be taken to decrease the symptoms of autism spectrum disorder problem and to improve the quality of life of ASD suffering people.

Keywords: Autism Spectrum Disorder, Machine Learning, Behavioural Analytics.

CHAPTER 1

INTRODUCTION

Autism Spectrum Disorder (ASD) is probably one of the major treatable diseases of our time due to the growing number of affected children. Over the past few years, the autistic spectrum disorder has been the subject of increasing scrutiny from the media, physicians and the general public. But more importantly, the main reason behind ASD is not invented yet. Children with Autism Disorder need important backing from the institutional, educational, clinical or medical, and social systems.

The core symptoms and sings for ASD can be reduced with proper and valid behavioral therapy. The measures would be more efficient if it is achieved in proper time. Specially, if the disorder can be diagnosed to the earliest possible time in childhood. Though behavioral therapy is used applying variety of certain strategies, but the efficient and long-term outcome can be ensured with the earliest start only. Nevertheless, though we are aware of that early stage examination, particularly if the disorder is noticed earlier on childhood (before the second year of a child), is maximum effective.

Though various barriers are found there opposing the implementation of proper treatment ensuring early stage in childhood and the very initial limitation would be the timely proper recognition and detection of ASD in a child. In our work, we are also concern about our failure to achieve early detection goals as well as a few issues for some possible solutions.

Machine-learning is a subfield of Artificial Intelligence, that has the potential to substantially enhance the role of computational methods in neuroscience. This is apparent by substantial work that has been carried out in developing machine-learning models. These efforts have resulted in development of machine-learning methods for the detection of ASD. These machine-learning models rely on statistical algorithms, and are suitable for complex problems involving combinatorial explosion of possibilities or non-linear processes where traditional computational models fail in quality or scalability.

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).

Classification Algorithms is one of the most important applications in the field of datamining, which can be useful for decision making in many real-world problems. The purpose of this proposed system is predication of autism spectrum disorder at early stages using different classification techniques. In healthcare field, the important and challenging task is todiagnose health conditions and proper treatment of disease at the early stage. It is very much important to detect it in the primary stage so that doctors can provide better medication to keep itself not turning into a serious matter.Thus there arises a need for predicting autism spectrum disorder at early stage using machine learning algorithms.In this we are working with Random Forest and SVM algorithms.

Table 1.1: Details of variables mapping to the Q-Chat-10

Variable in Dataset	Corresponding Q-chat-10-Toddler Features
A1	Does your child look at you when you call his/her name?
A2	How easy is it for you to get eye contact with your child?
A3	Does your child point to indicate that s/he wants something? (e.g. a toy that is out of reach)
A4	Does your child point to share interest with you? (e.g. pointing at an interesting sight)
A5	Does your child pretend? (e.g. care for dolls, talk on a toy phone)
A6	Does your child follow where you're looking?
A7	If you or someone else in the family is visibly upset, does your child show signs of wanting to comfort them? (e.g. stroking hair, hugging them)
A8	Would you describe your child's first words as:
A9	Does your child use simple gestures? (e.g. wave goodbye)
A10	Does your child stare at nothing with no apparent purpose?

1.1 Problem Statement

Autism spectrum disorder is a neuron developmental disorder that affects a person's interaction, communication and learning skills. Although diagnosis of autism can be done at any age, its symptoms generally appear in the first two years of life and develops through time. Autism patients face different types of challenges such as difficulties with concentration, learning disabilities, mental health problems such as anxiety, depression etc, motor difficulties, sensory problems and many others.

1.2 Objectives

The project aims at predicting the Autism Spectrum Disorder at early stages . It involves following steps.

- Collect the Autism Spectrum Disorder dataset and clean the data using Data Preprocessing Techniques. Data Cleaning is done to remove inaccurate, incomplete and unreasonable data that increases the quality of the data and hence the overall productivity.
- Divide the analyzed data into training and testing sets and train the model using the training data to predict the Autism Spectrum Disorder for given inputs.
- Compare various Algorithms by passing the analyzed dataset through them and calculating the error rate and accuracy for each. Choose the algorithm with the highest accuracy and lowest error rate.

CHAPTER 2

LITERATURE SURVEY

Vaishali R et.al. [1] proposed a method to identify Autism with optimum behavior sets. In this work, an ASD diagnosis dataset with 21 features obtained from the UCI machine learning repository experimented with swarm intelligence based binary firefly feature selection wrapper. The alternative hypothesis of the experiment claims that it is possible for a machine learning model to achieve a better classification accuracy with minimum feature subsets. Using Swarm intelligence based single-objective binary firefly feature selection wrapper it is found that 10 features among 21 features of ASD dataset are sufficient to distinguish between ASD and non-ASD patients.

Suman Raj et.al. [2] proposed that there is an attempt to explore the possibility to use Naïve Bayes, Support Vector Machine, Logistic Regression, KNN, Neural Network and Convolutional Neural Network for predicting and analysis of ASD problems in a child, adolescents, and adults. The proposed techniques are evaluated on publicly available three different non-clinically ASD datasets.

Khondaker.A et.al. [3] proposed a Smart Autism is a cloud based, automated framework for autism screening and confirmation. In developing countries, due to lack of resources and expertise, autism is detected later than early ages which consequently delays timely intervention. Therefore a mobile, interactive and integrated framework is proposed to screen and confirm autism in different age group with 3 layers of assessment process. Firstly, it screens by evaluating the responses of pictorial based screening questionnaire through mobile application. If autism is suspected, then in virtual assessment process, the child watches a video, its reaction is recorded and uploaded to the cloud for remote expert assessment. If autism is still suspected, then the child is referred to the nearest Autism Resource Center (ARC) for actual assessment.

J. A. Kosmicki et.al. [4] supposed a searching method for a least set of traits for autism detection. In this, the authors used a machine learning approach to evaluate the clinical assessment of ASD. The ADOS was performed on the subset of behaviors of children based on the autism spectrum. In this work different

machine learning algorithms were employed, involving stepwise backward feature identification on score sheets from 4540 individuals.

Kaushik Vakadkar et.al. [5] proposed models such as Naïve Bayes (NB), Logistic Regression (LR), and KNN to our dataset and constructed predictive models based on the outcome. The main objective of this paper is to thus determine if the child is susceptible to ASD in its nascent stages, which would help streamline the diagnosis process. Based on the results, Logistic Regression gives the highest accuracy for the selected dataset. These are lengthy and taking up a large amount of time as well as effort.

Dr.R. Surendiran et.al. [6] proposed "Using Machine Learning for Detection of Autism Spectrum Disorder": Prediction of Autism Spectrum Disorder in a child was carried out by using developmental delay, learning disability and speech or other language problems as attributes. Physical activity, premature birth and birth weight was also used to improve the accuracy. They used the 1-away method to predict the severity of Autism Spectrum disorder quite reasonably. The 1-away method improved the accuracy from 54.1% to 90.2%, which is a significant increase. Predicting the severity of ASD was possible with this kind of data set after applying the 1-away method, but the big increase in accuracy does warrant further research. This research could verify whether the 1-away method is actually useful in this case, or should not be used. It could also look into other attributes that are linked better to the severity of ASD.

CHAPTER 3

METHODOLOGY

The system uses machine learning to make predictions of the Autism Spectrum Disorder and Python as the programming language since Python has been accepted widely as a language for experimenting in the machine learning area. Machine learning uses historical data into gain experiences and generate a trained model by training it with the data. This model then makes output predictions. The better the collection of dataset, the better the accuracy of the classifier. It has been observed that machine learning methods such as regression and classification perform better than various statistical models.

3.1 Machine Learning

Machine Learning is undeniably one of the most influential and powerful technologies in today's world. Machine learning is a tool for turning information into knowledge. In the past 50 years, there has been an explosion of data. This mass of data is useless; we analyse it and find the patterns hidden within. Machine learning techniques are used to automatically find the valuable underlying patterns within complex data that we would otherwise struggle to discover.

The hidden patterns and knowledge about a problem can be used to predict future events and perform all kinds of complex decision making. To learn the rules governing a phenomenon, machines have to go through a learning process, trying different rules and learning from how well they perform. Hence, why it's known as Machine Learning.

Basic Terminology

- Dataset: A set of data examples, which contain features important to solving the problem.
- Features: Important pieces of data that help us understand a problem. These are fed into a Machine Learning algorithm to help it learn.
- Model: The representation (internal model) of a phenomenon that a Machine Learning algorithm has learnt. It learns this from the data it is shown during training. The model is the output you get after training an algorithm.

For example, a decision tree algorithm would be trained and produce a decision tree model.

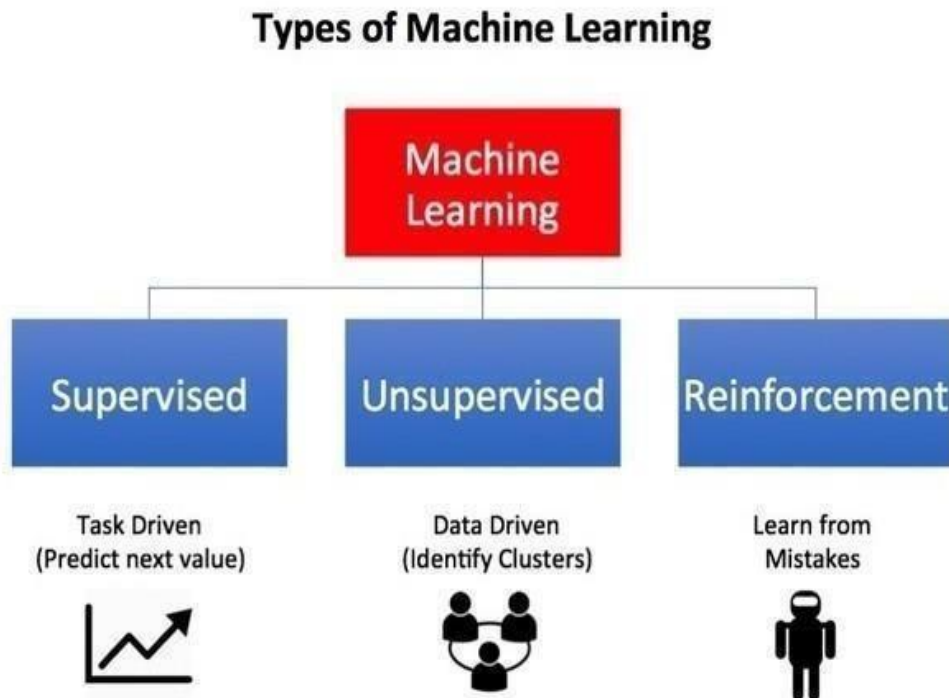


Fig 3.1: Types of Machine Learning

Supervised Learning: It is the most popular paradigm for machine learning. Given data in the form of examples with labels, we can feed a learning algorithm these example-label pairs one by one, allowing the algorithm to predict the label for each example, and giving it feedback as to whether it predicted the right answer or not. Over time, the algorithm will learn to approximate the exact nature of the relationship between examples and their labels. When fully trained, the supervised learning algorithm will be able to observe a new, never before-seen example and predict a good label for it.

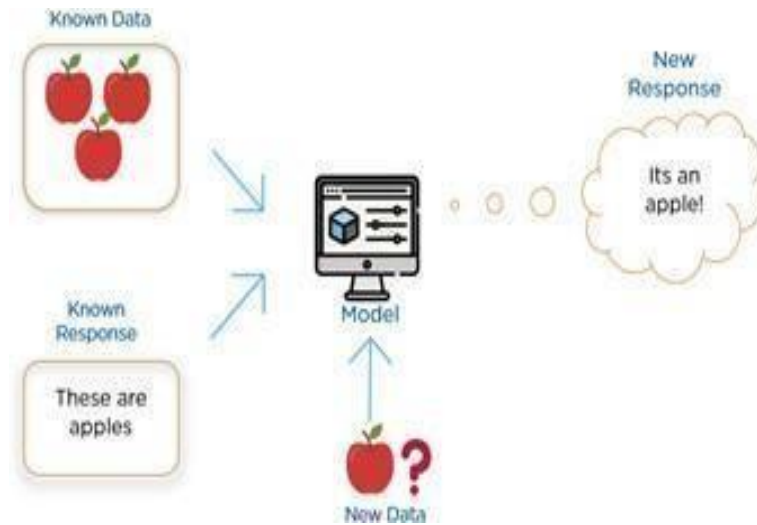


Fig 3.2: Process of Supervised Learning

Unsupervised learning: It is very much the opposite of supervised learning. It features no labels. Instead, the algorithm would be fed a lot of data and given the tools to understand the properties of the data. From there, it can learn to group, cluster, and organize the data in a way such that a human can come in and make sense of the newly organized data. Because unsupervised learning is based upon the data and its properties, we can say that unsupervised learning is data- driven. The outcomes from an unsupervised learning task are controlled by the data and the way it's formatted.

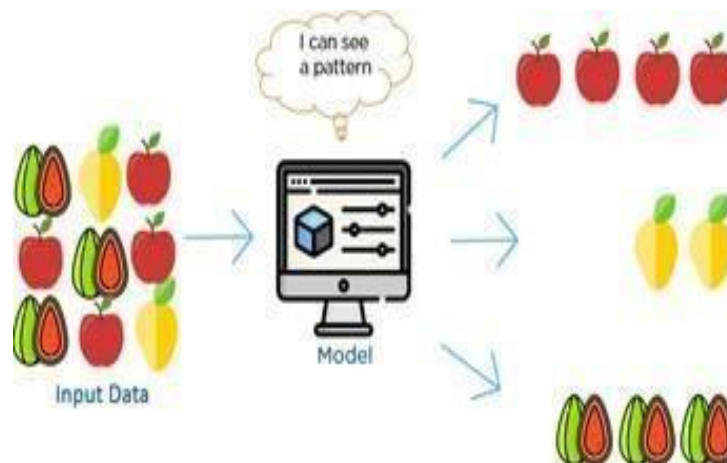


Fig 3.3: Process of Unsupervised Learning

Reinforcement learning: It is fairly different when compared to supervised and unsupervised learning. Reinforcement learning is very behaviour driven. It has influences from the fields of neuroscience and psychology. For any reinforcement learning problem, we need an agent and an environment as well as a way to connect the two through a feedback loop. To connect the agent to the environment, we give it a set of actions that it can take that affect the environment. To connect the environment to the agent, we have it continually issue two signals to the agent: an updated state and a reward (our reinforcement signal for behaviour).

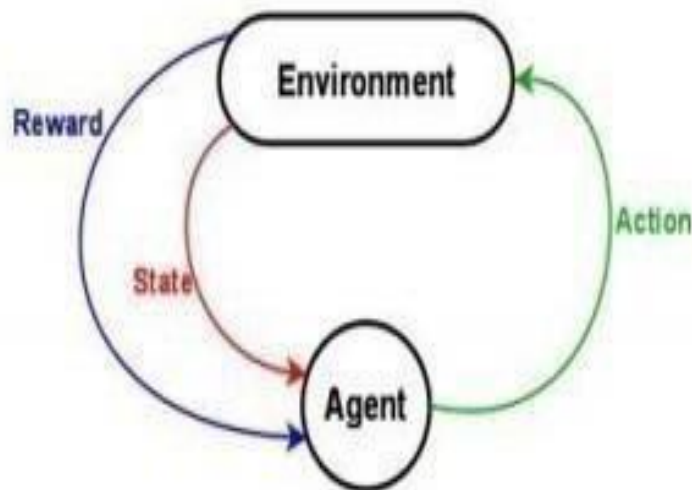


Fig 3.4: Reinforcement Learning

3.2 Basic Process of machine learning

- **Data Collection:** Collect the data that the algorithm will learn from.
- **Data Preparation:** Format and engineer the data into the optimal format, extracting important features and performing dimensionality reduction.
- **Training :** Also known as the fitting stage, this is where the Machine Learning algorithm actually learns by showing it the data that has been collected and prepared.
- **Evaluation:** Test the model to see how well it performs.
- **Tuning:** Fine tune the model to maximize its performance.

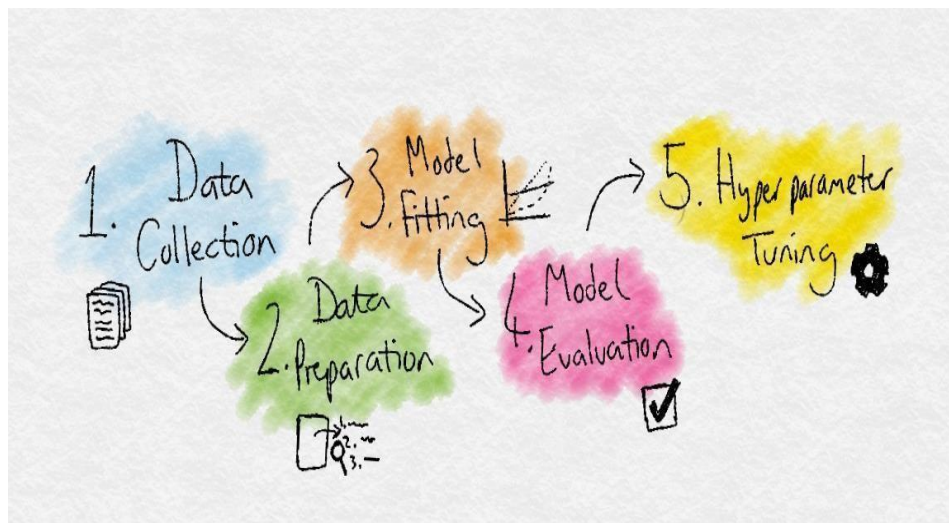


Fig 3.5: Basic process of Machine learning

3.3 Algorithms Used

Machine Learning offers a wide range of algorithms to choose from. These are usually divided into classification, regression, clustering and association. Classification and regression algorithms come under supervised learning while clustering and association comes under unsupervised learning.

- **Classification:** A classification problem is when the output variable is a category, such as —red‖ or —blue‖ or —disease‖ and —no disease‖. Example: Decision Trees
- **Regression:** A regression problem is when the output variable is a real value, such as dollars or weight. Example: Linear Regression
- **Clustering:** A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behavior Example: k means clustering.
- **Association:** An association rule learning problem is where you want to discover rules that describe large portions of your data, such as people that buy X also tend to buy Y.

Random Forest

Random forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because of its simplicity and diversity. It can be used for both classification and regression tasks. Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. One big advantage of random forest is that it can be used for both classification and regression problems, which form the majority of current machine learning systems. Another great quality of the random forest algorithm is that it is very easy to measure the relative importance of each feature on the prediction. Sk-learn provides a great tool for this that measures a feature's importance by looking at how much the tree nodes that use that feature reduce impurity across all trees in the forest.

The hyper parameters in random forest are either used to increase the predictive power of the model or to make the model faster. Python offers some built in random Forest functions which have the following hyper parameters.

i. To increase the predictive power:

- Firstly, there is the `n_estimators` hyper parameter, which is just the number of trees the algorithm builds before taking the maximum voting or taking the averages of predictions.
- Another important hyper parameter is `max_features` , which is the maximum number of features random forest considers to split a node.
- The last important hyper parameter is `min_sample_leaf` . This determines the minimum number of leafs required to split an internal node.

ii To Increase the model's speed:

- The `n_jobs` hyper parameter tells the engine how many processors it is allowed to use. If it has a value of one, it can only use one processor.
- The `random state` hyper parameter makes the model's output replicable. The model will always produce the same results when it has a definite value of random state and if it has been given the same hyper parameters and the same training data.

- The oob score (also called oob sampling), which is a random forest cross validation method. In this sampling, about one-third of the data is not used to train the model and can be used to evaluate its performance. These samples are called the out-of-bag samples.

The working of Random Forest is as follows:

- **Step 1** – First, start with the selection of random samples from a given dataset.
- **Step 2** – Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
- **Step 3** – In this step, voting will be performed for every predicted result.
- **Step 4** – At last, select the most voted prediction result as the final prediction result.

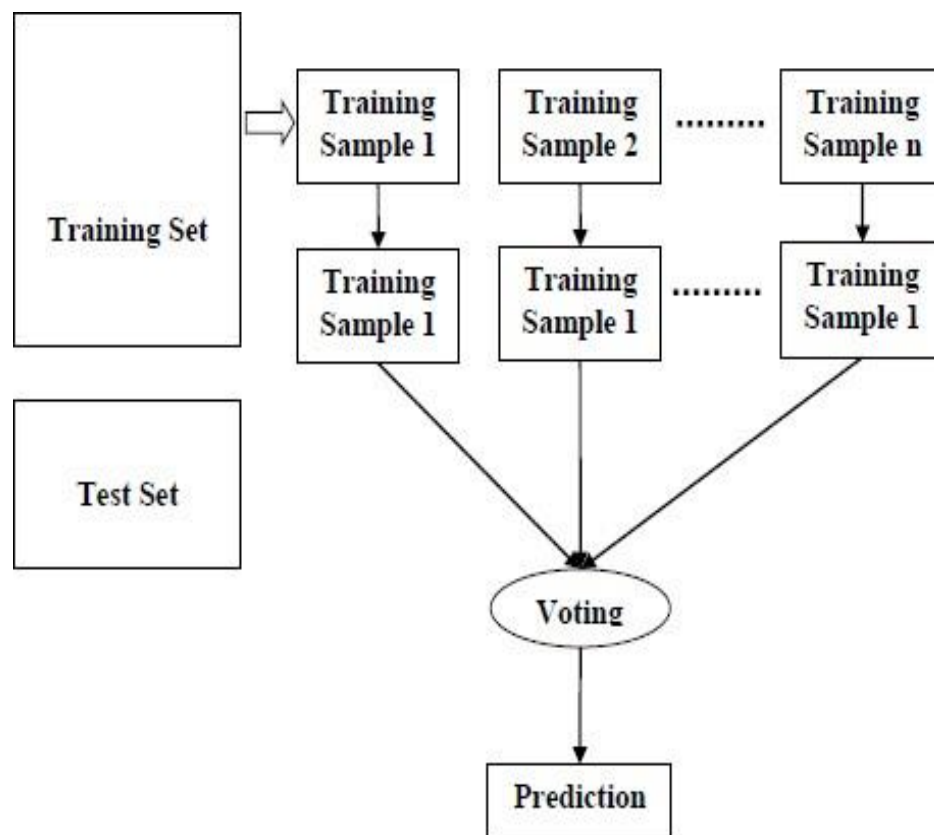


Fig 3.6: Working of Random Forest algorithm

Support Vector Machine

A support vector machine (SVM) is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. A support vector machine is also known as a support vector network (SVN).

The working of SVM is as follows:

- Pre-process the data
- Split the data into attributes and labels
- Divide the data into training and testing sets
- Train the SVM algorithm
- Make some predictions
- Evaluate the results of the algorithm

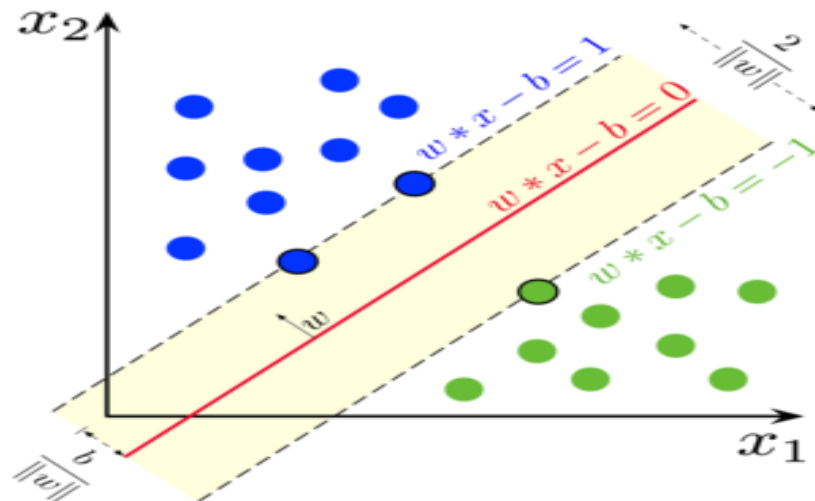


Fig 3.7: Working of SVM algorithm

CHAPTER 4

SYSTEM REQUIREMENTS SPECIFICATION

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and nonfunctional requirements, and may include a set of use cases that describe user interactions that the software must provide. It is very important in a SRS to list out the requirements and how to meet them. It helps the team to save upon their time as they are able to comprehend how are going to go about the project. Doing this also enables the team to find out about the limitations and risks early on.

A SRS can also be defined as a detailed description of a software system to be developed with its functional and non-functional requirements. It may include the use cases of how the user is going to interact with the software system. The software requirement specification document is consistent with all necessary requirements required for project development. To develop the software system we should have a clear understanding of Software system. To achieve this we need continuous communication with customers to gather all requirements.

A good SRS defines how the Software System will interact with all internal modules, hardware, and communication with other programs and human user interactions with a wide range of real life scenarios. It is very important that testers must be cleared with every detail specified in this document in order to avoid faults in test cases and its expected results.

Qualities of SRS

- Correct
- Unambiguous
- Complete
- Consistent
- Ranked for importance and/or stability
- Verifiable
- Modifiable
- Traceable



Fig 4.1: Types of Requirements in SRS

Some of the goals an SRS should achieve are to:

- Provide feedback to the customer, ensuring that the IT Company understands the issues the software system should solve and how to address those issues.
- Help to break a problem down into smaller components just by writing down the requirements.
- Speed up the testing and validation processes.
- Facilitate reviews.

4.1 Functional Requirements

A Functional Requirement is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputs to the software system, its behaviour, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. In software engineering and systems engineering, a Functional Requirement can range from the high-level abstract statement of the sender's necessity to detailed mathematical functional requirement specifications.

Benefits of Functional Requirements:

- Helps you to check whether the application is providing all the functionalities that were mentioned in the functional requirement of that application
- A functional requirement document helps you to define the functionality of a system or one of its subsystems.
- Functional requirements along with requirement analysis help identify missing requirements. They help clearly define the expected system service and behavior.
- Errors caught in the Functional requirement gathering stage are the cheapest to fix.
- Support user goals, tasks, or activities.

4.2 Basic Requirements

1. Data collection: The dataset used in this project is the data collected from reliable websites and merged to achieve the desired data set. The source of our dataset is Kaggle machine learning website.

2. Data Preprocessing: The purpose of preprocessing is to convert raw data into a form that fits machine learning. Structured and clean data allows a data scientist to get more precise results from an applied machine learning model. The technique includes data formatting, cleaning, and sampling. Here, data preprocessing focuses on finding the attributes with null values or invalid values and finding the relationships between various attributes as well. Data Preprocessing also helps in finding out the impact of each parameter on the target parameter. To preprocess our datasets we used EDA methodology. All the invalid and null values were handled by removing that record or giving the default value of that particular attribute based on its importance.

3. Dataset splitting: A dataset used for machine learning should be partitioned into two subsets — training and test sets. We split the dataset into two with a split ratio of 80% i.e., in 100 records 80 records were a part of the training set and remaining 20 records were a part of the test set.

4. Model training: After a data scientist has preprocessed the collected data and split it into train and test can proceed with a model training. This process entails —feeding the algorithm with training data. An algorithm will process data and

output a model that is able to find a target value (attribute) in new data an answer you want to get a predictive analysis. The purpose of model training is to develop a model. We trained our model using the random forest algorithm. On training the model it predicts the yield on giving the other attributes of the dataset as input.

5. Model evaluation and testing: The goal of this step is to develop the simplest model able to formulate a target value fast and well enough. A data scientist can achieve this goal through model tuning. That's the optimization of model parameters to achieve an algorithm's best performance.

4.3 Non-Functional Requirements

Non-Functional Requirement (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system.

Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. They specify the criteria that can be used to judge the operation of a system rather than specific behaviour. They may relate to emergent system properties such as reliability, response time and store occupancy. Non-functional requirements arise through the user needs, because of budget constraints, organizational policies, the need for interoperability with other software and hardware systems or because of external factors such as:- Product Requirements, Organizational Requirements, User Requirements, Basic Operational Requirement, etc.

Benefits of Non-Functional Requirements:

- The nonfunctional requirements ensure the software system follows legal and compliance rules.
- They ensure the reliability, availability, and performance of the software system.
- They ensure good user experience and ease of operating the software.

4.4 Python Libraries

Normally, a library is a collection of books or is a room or place where many books are stored to be used later. Similarly, in the programming world, a library is a collection of precompiled codes that can be used later on in a program for some specific well-defined operations. Other than pre-compiled codes, a library may contain documentation, configuration data, message templates, classes, and values, etc.

A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don't need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, etc.

Working of Python Library

As is stated above, a Python library is simply a collection of codes or modules of codes that we can use in a program for specific operations. We use libraries so that we don't need to write the code again in our program that is already available. But how it works. Actually, in the MS Windows environment, the library files have a DLL extension (Dynamic Load Libraries). When we link a library with our program and run that program, the linker automatically searches for that library. It extracts the functionalities of that library and interprets the program accordingly. That's how we use the methods of a library in our program. We will see further, how we bring in the libraries in our Python programs.

Python standard library

The Python Standard Library contains the exact syntax, semantics, and tokens of Python. It contains built-in modules that provide access to basic system functionality like I/O and some other core modules. Most of the Python Libraries are written in the C programming language. The Python standard library consists of more than 200 core modules. All these work together to make Python a high-level programming language. Python Standard Library plays a very important role. Without it, the programmers can't have access to the functionalities of Python. But other than this, there are several other libraries in Python that make a programmer's life easier. Let's have a look at some of the commonly used libraries:

1.Pandas: Pandas are an important library for data scientists. It is an opensource machine learning library that provides flexible high-level data structures and a variety of analysis tools. It eases data analysis, data manipulation, and cleaning of data. Pandas support operations like Sorting, Re-indexing, Iteration, Concatenation, Conversion of data, Visualizations, Aggregations, etc.

2.Numpy: The name “Numpy” stands for “Numerical Python”. It is the commonly used library. It is a popular machine learning library that supports large matrices and multi-dimensional data. It consists of in-built mathematical functions for easy computations. Even libraries like TensorFlow use Numpy internally to perform several operations on tensors. Array Interface is one of the key features of this library.

3.Scikit-learn: It is a famous Python library to work with complex data. Scikit-learn is an open-source library that supports machine learning. It supports variously supervised and unsupervised algorithms like linear regression, classification, clustering, etc. This library works in association with Numpy and SciPy.

4.Seaborn: Seaborn is one of an amazing library for visualization of the graphical statistical plotting in Python. Seaborn provides many color palettes and defaults beautiful styles to make the creation of many statistical plots in Python more attractive.

5. Streamlit: Streamlit is a open source python library for building fast, performant and beautiful data apps. It is a great tool to develop interactive apps quickly for demonstrating a solution. It turns data scripts into shareable web apps in minutes. No front-end experience required. Streamlit is an open-source python framework for building web apps for Machine Learning and Data Science. We can instantly develop web apps and deploy them easily using Streamlit. Streamlit allows you to write an app the same way you write a python code. Streamlit makes it seamless to work on the interactive loop of coding and viewing results in the web app. Streamlit builds upon three simple principles.

- Embrace Scripting
- Weave in interaction
- Deploy instantly

We can build a streamlit in 4 simple steps:

1. Make sure you have python 3.6+ installed in your system.

Install streamlit using PIP:

```
$pip install streamlit
```

2. Create a python script by naming it `hello.py`, enter the following code and save it.

```
Import streamlit as st
```

```
st.title('Hello World')
```

```
st.write('Welcome to my first app.')
```

3. Run the app by running the following command in a terminal:

```
$streamlit run hello
```

we can install the streamlit by using following command:

```
$pip install source streamlit
```

Use of Libraries in Python Program

As we write large-size programs in Python, we want to maintain the code's modularity. For the easy maintenance of the code, we split the code into different parts and we can use that code later ever we need it. In Python, modules play that part. Instead of using the same code in different programs and making the code complex, we define mostly used functions in modules and we can just simply import them in a program wherever there is a requirement. We don't need to write that code but still, we can use its functionality by importing its module. Multiple interrelated modules are stored in a library. And whenever we need to use a module, we import it from its library. In Python, it's a very simple job to do due to its easy syntax. We just need to use **import**.

4.5 Hardware Requirements

The hardware requirements include the requirements specification of the physical computer resources for a system to work efficiently. The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. The Hardware Requirements are listed below:

- 1. Processor:** A processor is an integrated electronic circuit that performs the calculations that run a computer. A processor performs arithmetical, logical,

input/output (I/O) and other basic instructions that are passed from an operating system (OS). Most other processes are dependent on the operations of a processor. A minimum 1 GHz processor should be used, although we would recommend S2GHz or more. A processor includes an arithmetical logic and control unit (CU), which measures capability in terms of the following:

- Ability to process instructions at a given time
- Maximum number of bits/instructions
- Relative clock speed



Fig 4.2: Processor

2. Hard Drive: A hard drive is an electro-mechanical data storage device that uses magnetic storage to store and retrieve digital information using one or more rigid rapidly rotating disks, commonly known as platters, coated with magnetic material. The platters are paired with magnetic heads, usually arranged on a moving actuator arm, which reads and writes data to the platter surfaces. Data is accessed in a random-access manner, meaning that individual blocks of data can be stored or retrieved in any order and not only sequentially. HDDs are a type of non-volatile storage, retaining stored data even when powered off. 32 GB or higher is recommended for the proposed system.



Fig 4.3: Hard Disk

3. Memory (RAM): Random-access memory (RAM) is a form of computer data storage that stores data and machine code currently being used. A random-access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory. In today's technology, random-access memory takes the form of integrated chips. RAM is normally associated with volatile types of memory (such as DRAM modules), where stored information is lost if power is removed, although non-volatile RAM has also been developed. A minimum of 2 GB RAM is recommended for the proposed system.



Fig 4.4: RAM

4.6 Software Requirements

The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client's point of view.

1.Jupyter Notebook: The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use. The Jupyter Notebook combines three components:

- **The notebook web application:** An interactive web application for writing and running code interactively and authoring notebook documents.
- **Kernels:** Separate processes started by the notebook web application that runs users' code in a given language and returns output back to the notebook web application. The kernel also handles things like computations for interactive widgets, tab completion and introspection.
- **Notebook documents:** Self- contained documents that contain a representation of all content visible in the notebook web application, including inputs and outputs of the computations, narrative text, equations, images, and rich media representations of objects. Each notebook document has its own kernel.



Fig 4.5: Jupyter Notebook icon

2.Google Colab: colab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks that you create can be simultaneously edited by your team members - just the way you edit documents in Google Docs. Colab supports many popular machine learning libraries which can be easily loaded in your notebook.



Fig 4.6: Google Colab icon

3.Anaconda Navigator: Anaconda is an open-source distribution of the Python and R programming languages for data science that aims to simplify package management and deployment. Package versions in Anaconda are managed by the package management system, conda, which analyzes the current environment before executing an installation to avoid disrupting other frameworks and packages.

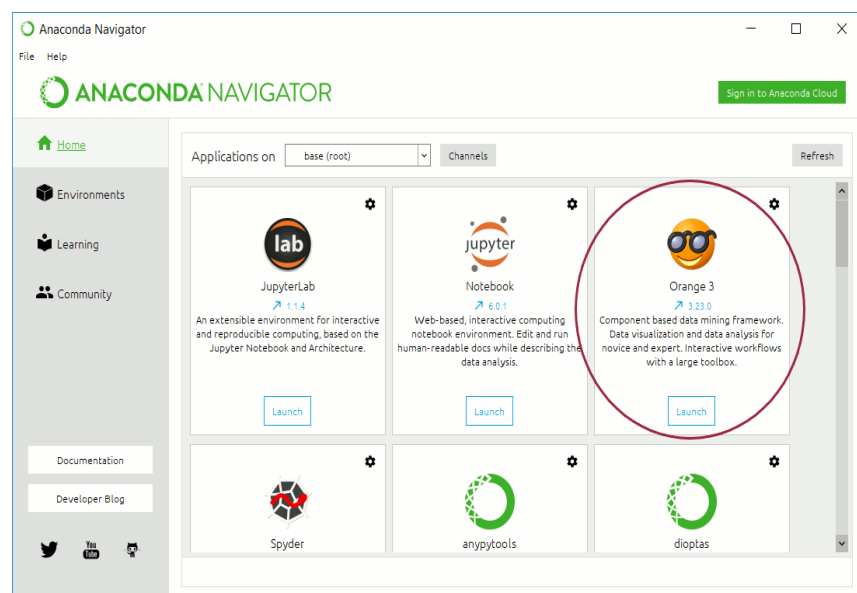


Fig 4.7: Anaconda Navigator

4.Spyder: Spyder, the Scientific Python Development Environment, is a free integrated development environment (IDE) that is included with Anaconda. It includes editing, interactive testing, debugging, and introspection features.



Fig 4.8: Spyder IDE

5.Python: It is an object-oriented, high-level programming language with integrated dynamic semantics primarily for web and app development. It is extremely attractive in the field of Rapid Application Development because it offers dynamic typing and dynamic binding options. Python is relatively simple, so it's easy to learn since it requires a unique syntax that focuses on readability. Developers can read and translate Python code much easier than other languages. In turn, this reduces the cost of program maintenance and development because it allows teams to work collaboratively without significant language and experience barriers. Additionally, Python supports the use of modules and a package, which means that programs can be designed in a modular style and code can be reused across a variety of projects.



Fig 4.9: Python icon

CHAPTER 5

SYSTEM ANALYSIS AND DESIGN

Systems development is a systematic process which includes phases such as planning, analysis, design, deployment, and maintenance. System Analysis is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose. Analysis specifies what the system should do.

System Design is a process of planning a new business system or replacing an existing system by defining its components or modules to satisfy the specific requirements. Before planning, you need to understand the old system thoroughly and determine how computers can best be used in order to operate efficiently. System Design focuses on how to accomplish the objective of the system.

5.1 System Architecture

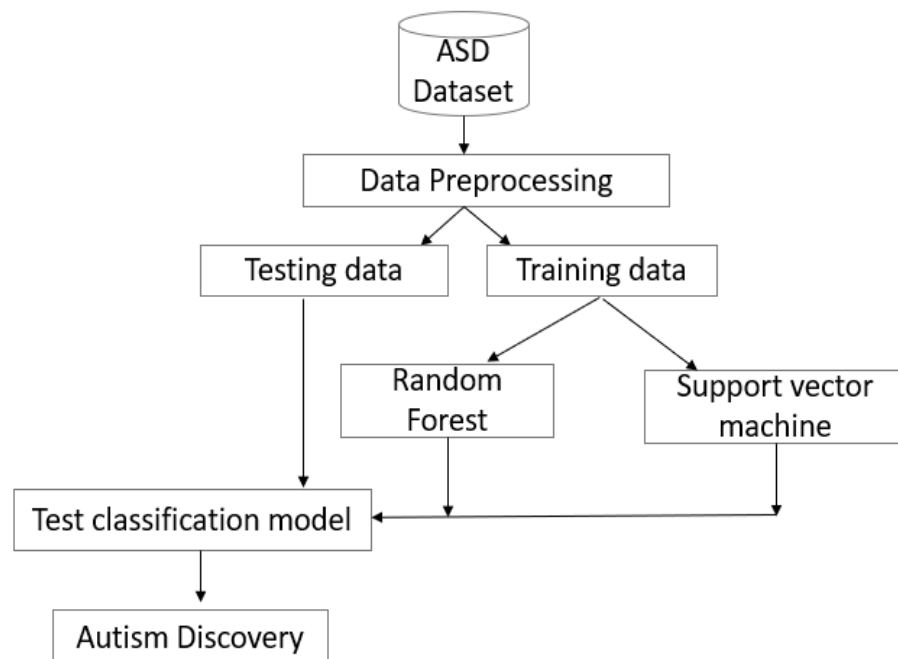


Fig 5.1: System Architecture

Architecture diagrams can help system designers and developers visualize the high-level, overall structure of their system or application for the purpose of ensuring the system meets their users' needs. They can also be used to describe patterns that are used throughout the design. It's somewhat like a blueprint that can be used as a guide for the convenience of discussing, improving, and following among a team.

5.2 UML DIAGRAMS:

UML represents Unified Modelling Language. UML is an institutionalized universally useful showing dialect in the subject of article situated programming designing. The fashionable is overseen, and become made by way of, the Object management Group.

The goal is for UML to become a regular dialect for making fashions of item arranged PC programming. In its gift frame UML is contained two noteworthy components: a Meta-show and documentation. Later on, a few type of method or system can also likewise be brought to; or related with, UML. The Unified Modeling Language is a popular dialect for indicating, Visualization, Constructing and archiving the curios of programming framework, and for business demonstrating and different non-programming frameworks. The UML speaks to an accumulation of first-rate building practices which have verified fruitful in the showing of full-size and complicated frameworks. The UML is a essential piece of creating gadgets located programming and the product development method. The UML makes use of commonly graphical documentations to specific the plan of programming ventures.

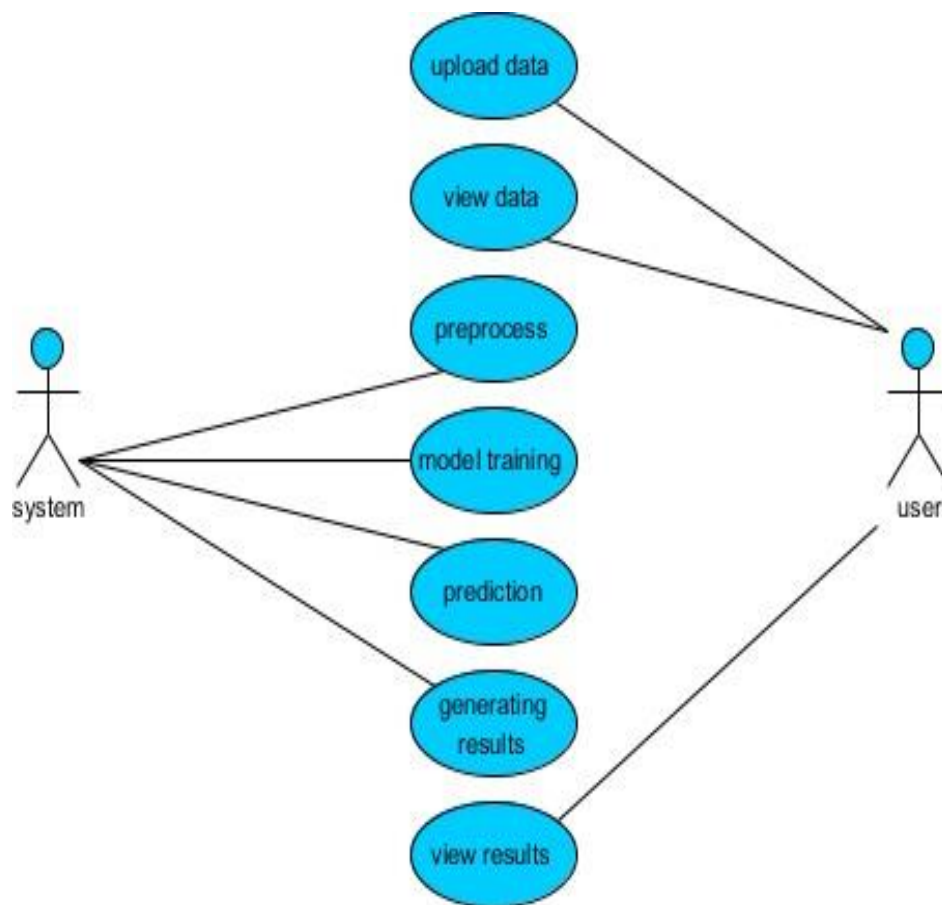
GOALS:

The Primary goals inside the plan of the UML are as in step with the subsequent:

1. Provide clients a prepared to utilize, expressive visual showing Language on the way to create and change massive models.
2. Provide extendibility and specialization units to make bigger the middle ideas.
3. Be free of specific programming dialects and advancement manner.
4. Provide a proper cause for understanding the displaying dialect.
5. Encourage the improvement of OO gadgets exhibit.

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

**Fig 5.2: Use Case Diagram**

Class Diagram:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

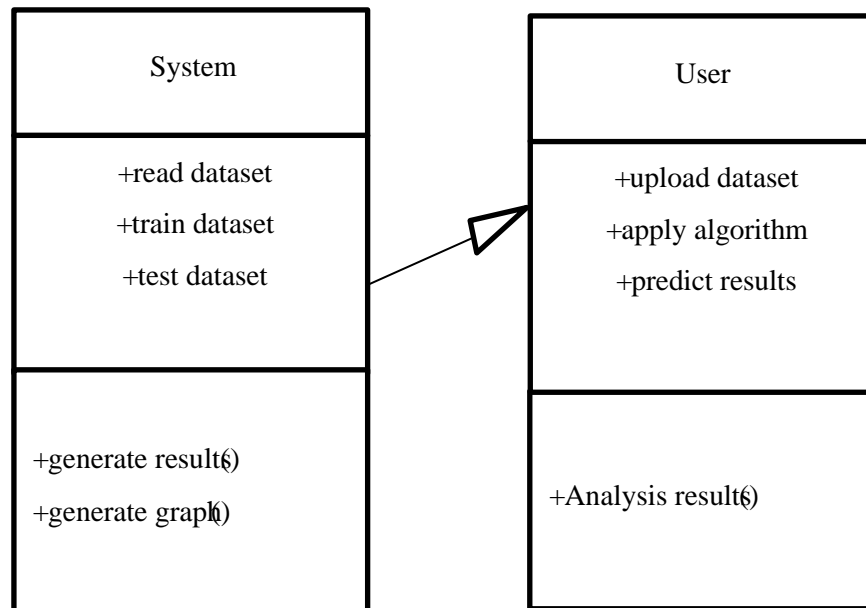


Fig 5.3 : Class Diagram

SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

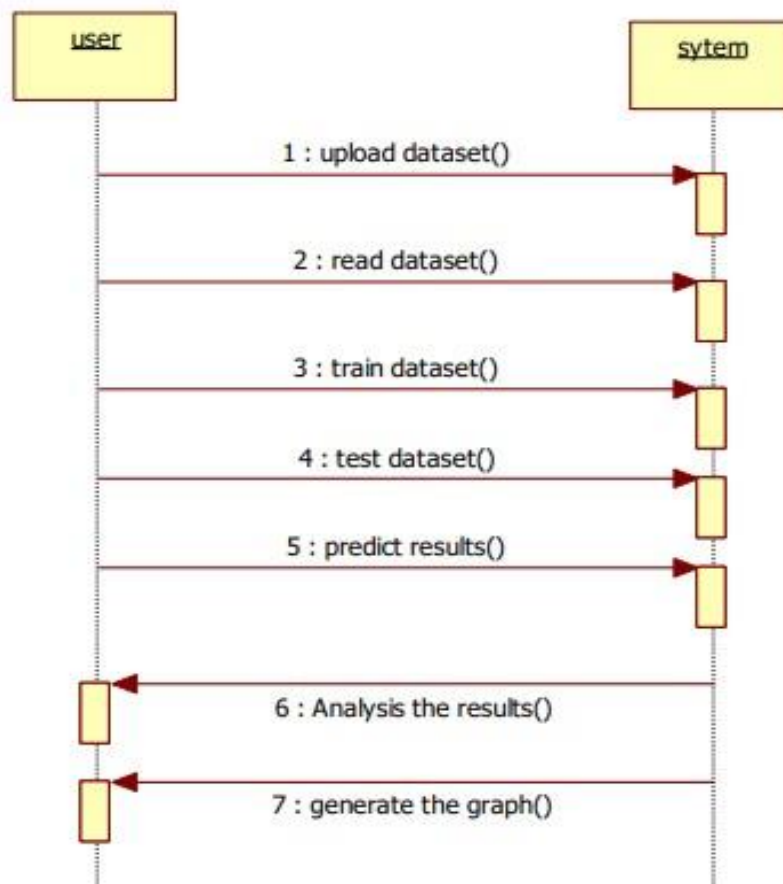


Fig 5.4: Sequence Diagram

ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

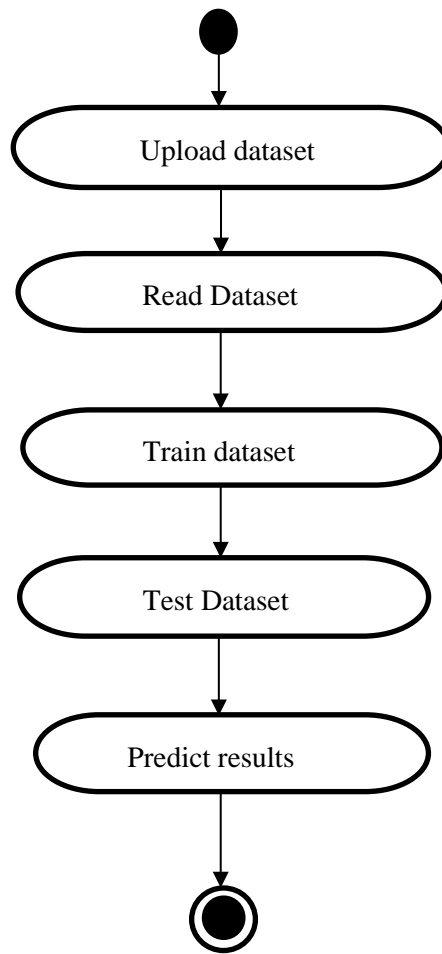


Fig. 5.5: Activity Diagram

CHAPTER 6

IMPLEMENTATION

The project Autism Spectrum Disorder Prediction using Machine Learning is developed to predict Autism Spectrum Disorder, as we all know in competitive environment of economic development the mankind has involved so much that he/she is not concerned about health according to research there are 40% peoples how ignores about general disease which leads to harmful disease later.

The Project “Autism Spectrum Disorder Prediction” is implemented using python completely. Even the interface of this project is done using python's library interface. The user interface is developed by using Streamlit library. Here first the user needs to give the symptoms that are asked. For more accurate result the user needs to enter all the given symptoms(attributes values), then the system will provide the accurate result.

This prediction is basically done with the help of 2 algorithms of machine learning such as Random Forest, SVM. From these two algorithms we are calculating the accuracy after calculating we get to know that Random Forest gives the most accuracy comparing to the other algorithms. Based on the symptoms that user have entered this model will calculate the accuracy and displays the result by a user interface.

The project is designed user friendly and predicts the result based on the user input . Autism Spectrum Disorder Prediction using machine learning, If the user wants to know whether he/she have any Autism Spectrum Disorder or the future prediction of that they need to open and give details. The user has to provide the symptoms which he/she is going through based on which we have several algorithms which predict the disease and also displays the percentage of accuracy.

The user needs to enter all the columns of symptoms to get the accurate result, Data collection and dataset preparation This will involve collection of dataset from Kaggle machine learning website, then pre-processing is applied on dataset which will remove all the unnecessary data and extract important features from data.

6.1 Dataset

Machine Learning depends heavily on data. It's the most crucial aspect that makes algorithm training possible. It uses historical data and information to gain experiences. The better the collection of the dataset, the better will be the accuracy. The first step is Data Collection. For this project, we require the dataset “**Autistic Spectrum Disorder Screening Data for Toddlers**” which is available in Kaggle machine learning website. The source for the dataset is <https://www.kaggle.com/datasets/fabdelja/autism-screening-for-toddlers>. In this dataset we record ten behavioural features (Q-Chat-10) plus other individuals characteristics that have proved to be effective in detecting the ASD cases from controls in behaviour science.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Case_No	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Age_Mon	Qchat-10	Sex	Ethnicity	Jaundice	Family_m	Who com	class
2	1	0	0	0	0	0	0	1	1	0	1	28	3	f	middle ea	yes	no	family me	No
3	2	1	1	0	0	0	1	1	0	0	0	36	4	m	White Eur	yes	no	family me	Yes
4	3	1	0	0	0	0	0	1	1	0	1	36	4	m	middle ea	yes	no	family me	Yes
5	4	1	1	1	1	1	1	1	1	1	1	24	10	m	Hispanic	no	no	family me	Yes
6	5	1	1	0	1	1	1	1	1	1	1	20	9	f	White Eur	no	yes	family me	Yes
7	6	1	1	0	0	1	1	1	1	1	1	21	8	m	black	no	no	family me	Yes
8	7	1	0	0	1	1	1	0	0	1	0	33	5	m	asian	yes	no	family me	Yes
9	8	0	1	0	0	1	0	1	1	1	1	33	6	m	asian	yes	no	family me	Yes
10	9	0	0	0	0	0	0	1	0	0	1	36	2	m	asian	no	no	family me	No
11	10	1	1	1	0	1	1	0	1	1	1	22	8	m	south asia	no	no	Health Ca	Yes
12	11	1	0	0	1	0	1	1	0	1	1	36	6	m	Hispanic	yes	yes	family me	Yes
13	12	1	1	1	1	0	1	1	1	0	1	17	8	m	middle ea	yes	no	family me	Yes
14	13	0	0	0	0	0	0	0	0	0	0	25	0	f	middle ea	yes	no	family me	No
15	14	1	1	1	1	0	0	1	0	1	1	15	7	f	middle ea	yes	no	family me	Yes
16	15	0	0	0	0	0	0	0	0	0	0	18	0	m	middle ea	no	no	family me	No
17	16	1	1	1	0	1	0	1	1	0	1	12	7	m	black	no	no	family me	Yes
18	17	0	0	0	0	0	0	0	0	0	0	36	0	m	middle ea	no	yes	family me	No
19	18	1	1	1	0	1	1	1	1	0	1	12	8	f	middle ea	yes	no	family me	Yes
20	19	1	0	0	0	1	0	0	0	0	1	29	3	f	middle ea	no	no	family me	No
21	20	1	1	1	0	1	0	1	1	0	1	12	7	f	black	no	no	family me	Yes
22	21	1	0	0	1	1	1	1	1	1	0	36	7	m	middle ea	no	no	family me	Yes

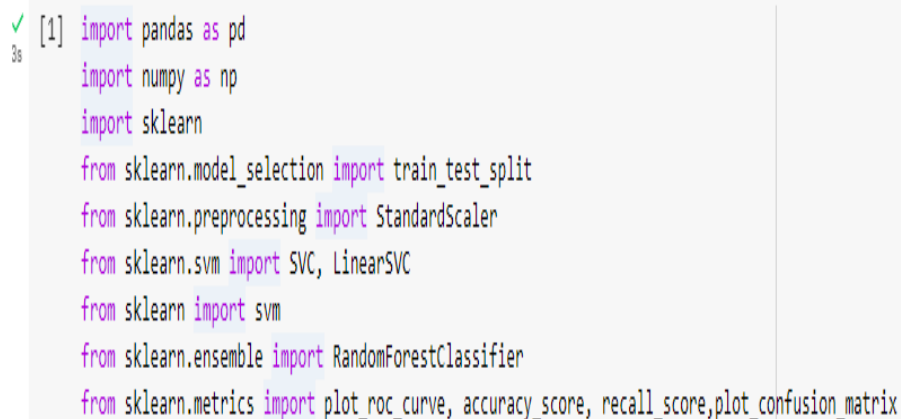
Fig 6.1: Dataset

CSV file:

The dataset used in this project is a .csv file. In computing, a comma separated values (CSV) file is a delimited text file that uses a comma to separate values. A CSV file stores tabular data (numbers and text) in plaintext. Each line of the file is a data record. Each record consists of one or more fields, separated by commas. The use of the comma as a field separator is the source of the name for this file format. CSV is a simple file format used to store tabular data, such as a spreadsheet or database. Files in the CSV format can be imported to and exported from programs that store data in tables, such as Microsoft Excel or Open Office Calc.

6.2 Data Collection

In Python there are many predefined libraries we can use them directly just by importing them into our code. As we import the libraries we can use them when they are needed and using can be done easily. These can be used just by calling them by these we can reduce the writing of larger amount of codes and we can save the time. Some of modules imported from the Python libraries are shown below.

Importing Necessary Libraries


```
[1] import pandas as pd
import numpy as np
import sklearn
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC, LinearSVC
from sklearn import svm
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import plot_roc_curve, accuracy_score, recall_score, plot_confusion_matrix
```

Fig 6.2: Importing Libraries**Exploratory Data Analysis**

It is an approach to analyzing datasets to summarize their main characteristics, often with visual methods. It is also about knowing your data, gaining a certain amount of familiarity with the data, before one starts to extract insights from it. The idea is to spend less time coding and focus more on the analysis of data itself. After

the data has been collected, it undergoes some processing before being cleaned and EDA is then performed. After EDA, go back to processing and cleaning of data, i.e., this can be an iterative process. Use the cleaned dataset and knowledge from EDA to perform modelling and reporting. Exploratory data analysis is generally cross-classified in two ways. First, each method is either non-graphical or graphical. And second, each method is either univariate or multivariate (usually just bivariate). It is a good practice to understand the data first and try to gather as many insights from it. EDA is all about making sense of data in hand. EDA can give us the following:

- Preview data
- Check total number of entries and column types using in built functions. It is a good practice to know the columns and their corresponding data types.
- Check any null values.
- Check duplicate entries.
- Plot count distribution of categorical data.

Using various built in functions, we can get an insight of the number of values in each column which can give us information about the null values or the duplicate data. We can also find the mean, standard deviation, minimum value and the maximum value. This is the basic procedure of EDA. To get a better insight of the data that is being used, we can plot graphs like the correlation matrix which is one of the most important concept which gives us a lot of information about how variables (columns) are related to each other and the impact each of them have on the other.

```
✓ [2] #Loading the dataset to pandas DataFrame  
0s toddlers_df = pd.read_csv("ASD Dataset.csv")  
type(toddlers_df)  
  
pandas.core.frame.DataFrame
```

Fig 6.3: Loading the Dataset

A few other graphs like box plot and distribution graphs can be plotted too.

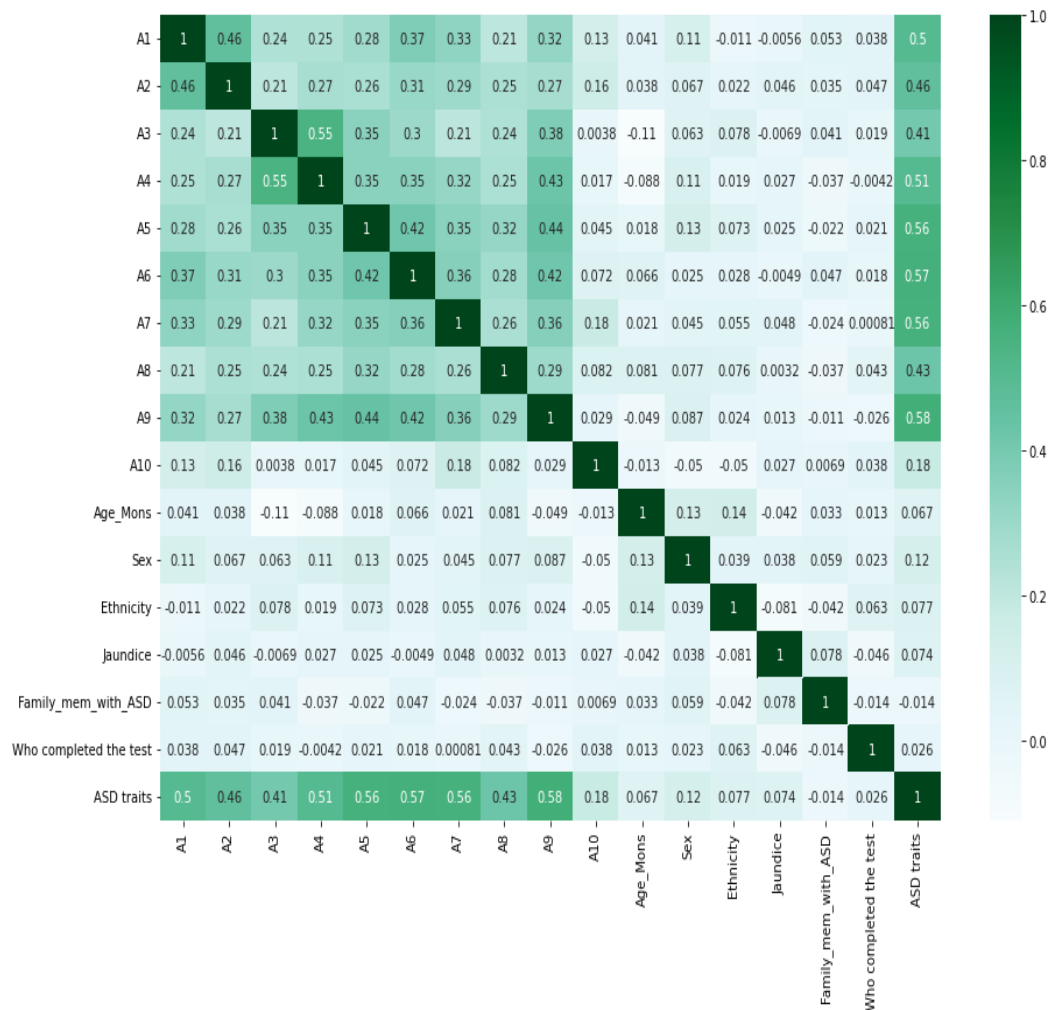


Fig 6.4: Example of Correlation matrix

Dark shades represent positive correlation while lighter shades represent negative correlation. A correlation matrix is a statistical technique used to evaluate the relationship between two variables in the dataset. The matrix is a table in which every cell contains a correlation coefficient, where 1 is considered as a strong relationship between variables, 0 is considered as a neutral and -1 is not a strong relationship.

6.3 Data Pre-Processing

Data Pre-Processing is a Data Mining method that entails converting raw data into a format that can be understood. Real-world data is frequently inadequate, inconsistent, and/or lacking in specific activities or trends, as well as including numerous inaccuracies. This might result in low-quality data collection and, as a result, low-quality models based on that data. Preprocessing data is a method of resolving such problems. Machines do not comprehend free text, image, or video data; instead, they comprehend 1s and 0s. So putting on a slideshow of all our photographs and expecting our machine learning model to learn from it is probably not going to be adequate. Data Pre-processing is the step in any Machine Learning process in which the data is changed, or encoded, to make it easier for the machine to parse it. In other words, the algorithm can now easily interpret the data's features. Data Pre-processing can be done in four different ways. Data cleaning/cleaning, data integration, data transformation, and data reduction are the four categories.

6.3.1 Data Cleaning:

Data in the real world is frequently incomplete, noisy, and inconsistent. Many bits of the data may be irrelevant or missing. Data cleaning is carried out to handle this aspect. Data cleaning methods aim to fill in missing values, smooth out noise while identifying outliers, and fix data discrepancies. Unclean data can confuse data and the model. Therefore, running the data through various Data Cleaning/Cleansing methods is an important Data Pre-processing step.

6.3.2 Data Integration:

It is involved in a data analysis task that combines data from multiple sources into a coherent data store. These sources may include multiple databases. Do you think how data can be matched up?? For a data analyst in one database, he finds Customer_ID and in another he finds cust_id, How can he be sure about them and say these two belong to the same entity. Databases and Data warehouses have Metadata (It is the data about data) it helps in avoiding errors.

6.3.3 Data Reduction :

Because data mining is a methodology for dealing with large amounts of data. When dealing with large amounts of data, analysis becomes more difficult. We employ a data reduction technique to get rid of this. Its goal is to improve storage

efficiency while lowering data storage and analysis expenses.

Dimensionality Reduction

A huge number of features may be found in most real-world datasets. Consider an image processing problem: there could be hundreds of features, also known as dimensions, to deal with. As the name suggests, dimensionality reduction seeks to minimize the number of features but not just by selecting a sample of features from the feature set, which is something else entirely Feature Subset Selection or feature selection.

Numerosity Reduction

Data is replaced or estimated using alternative and smaller data representations such as parametric models (which store only the model parameters rather than the actual data, such as Regression and Log-Linear Models) or non- parametric approaches.

CHAPTER -7

TESTING

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. Testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements. It involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration.

It involves identifying bug/error/defect in a software without correcting it. Normally professionals with a quality assurance background are involved in bugs entification. Testing is performed in the testing phase. We have different types of testing like unit testing, integration testing, validation testing, system testing, manual testing.

7.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration.

7.2 Integration Testing

Integration tests are designed to test integrated software components to determine if they actually run as one program Testing is event driven and is more

concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

As a rule, system testing takes, as its input, all the "integrated" software components that have successfully passed integration testing and also the software system itself integrated with any applicable hardware system. System testing is a more limited type of testing; it seeks to detect defects both within the "inter assemblages" and also within the system as a whole. System testing is performed on the entire system in the context of a Functional Requirement Specification (FRS) or System Requirement Specification (SRS).

7.3 Validation Testing

An engineering validation test (EVT) is performed on first engineering prototypes, to ensure that the basic unit performs to design goals and specifications. It is important in identifying design problems and solving them as early in the design cycle as possible, is the key to keeping projects on time and within budget. Too often, product design and performance problems are not detected until late in the product development cycle-when the product is ready to be shipped. The old adage holds true: It costs a penny to make a change in engineering, a dime in production and a dollar after a product is in the field.

Verification is a Quality control process that is used to evaluate whether or not a product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase. Verification can be in development, scaleup, or production. This is often an internal process. Validation is a Quality assurance process of establishing evidence that provides a high degree of assurance that a product, service, or system accomplishes its intended requirements, This often involves acceptance of fitness for purpose with end users and other product stakeholders.

7.4 System Testing

System testing of software or hardware is testing conducted on a complete,

integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

As a rule, system testing takes, as its input, all the "integrated" software components that have successfully passed integration testing and also the software system itself integrated with any applicable hardware system. System testing is a more limited type of testing; it seeks to detect defects both within the "inter assemblages" and also within the system as a whole. System testing is performed on the entire system in the context of a Functional Requirement Specification (FRS) or System Requirement Specification (SRS).

7.5 Manual Testing

Manual testing is the process of manually testing software for defects. It requires a tester to play the role of an end user whereby they use most of the application's features to ensure correct behavior. To guarantee completeness of testing, the tester often follows a written test plan that leads them through a set of important test cases.

Advantages:

- Low-cost operation as no software tools are used.
- Most of the bugs are caught by manual testing.
- Humans observe and judge better than the automated tools.

7.6 Performance Testing

It works fine with moderate internet speed. The connection is secured and user details are stored in a secured manner. The switch from one screen to another is quick and smooth. The inputs from users are taken correctly and response is recorded quickly.

7.7 Compatibility Testing

This application is compatible with all the browsers enabled with javascript. It is compatible with all the mobile devices and desktop.

7.8 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

CHAPTER 8

RESULTS

In this section, results are analyzed using classification technique SVM and Random Forest Algorithms. The classification technique implementation were performed using Matplotlib Library. Here the dataset contains various attributes of health-related data and the class name. Here we divided the data into two parts which consists of training data and other is validating data. With the given training data, we trained an SVM model and Random Forest model. Then using validating data we validated the models and plotted a confusion matrix and also found out the 87.2% accuracy through SVM model and 93.83% accuracy through Random Forest model.

The Confusion matrix is used to indicate the quality of the classifier for correct prediction. The count value in the confusion matrix shows the number of correct and incorrect classifier predictions. The top row of the confusion matrix gives the predicted positive events with true positive and bottom row corresponds to no events with true negatives.

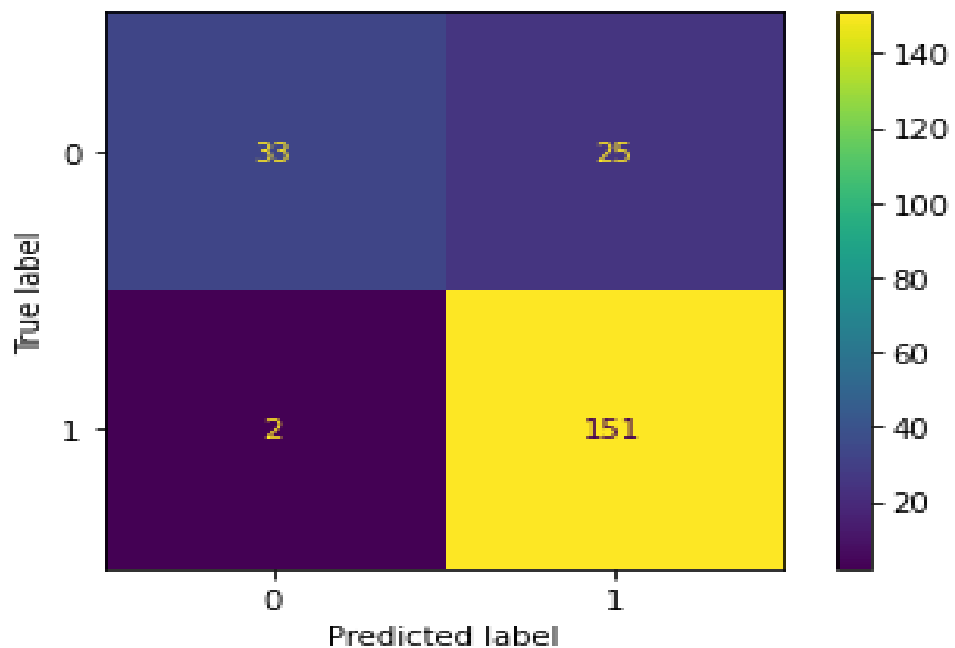


Fig 8.1: Confusion Matrix for SVM

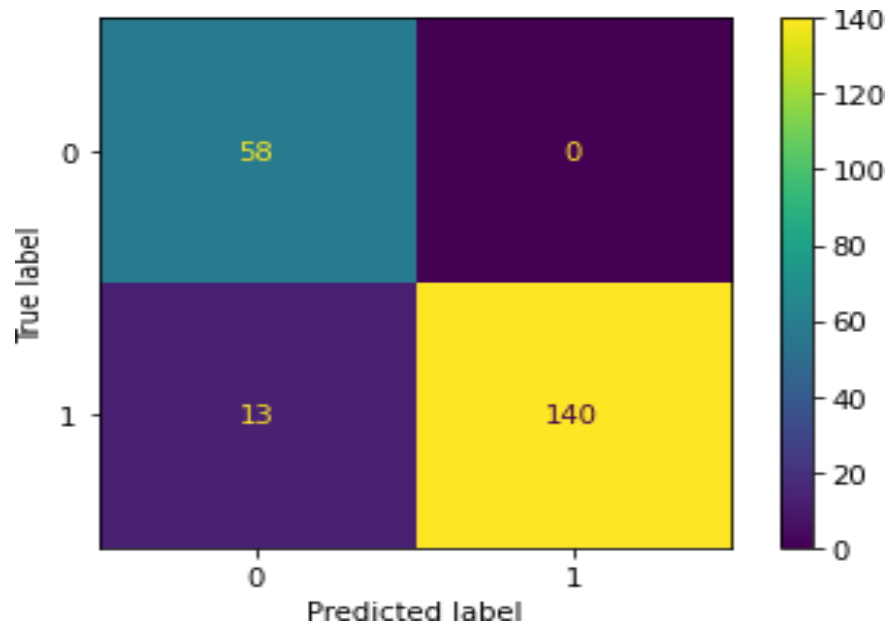


Fig 8.2: Confusion Matrix for Random Forest

```
#SVM Model
x_test_prediction=sv.predict(X_test.values)
test_data_accuracy=accuracy_score(x_test_prediction,y_test)
print('Accuracy for Test Data: ',test_data_accuracy)

Accuracy for Test Data: 0.8720379146919431

#Random Forest Model
x_test_prediction=rf.predict(X_test.values)
test_data_accuracy=accuracy_score(x_test_prediction,y_test)
print('Accuracy for Test Data: ',test_data_accuracy)

Accuracy for Test Data: 0.9383886255924171
```

Fig 8.3: Accuracy Results

	ML Model	Train Accuracy	Test Accuracy
0	SVM	0.865	0.872
1	Random Forset	0.948	0.938

Fig 8.4: Models and their accuracy

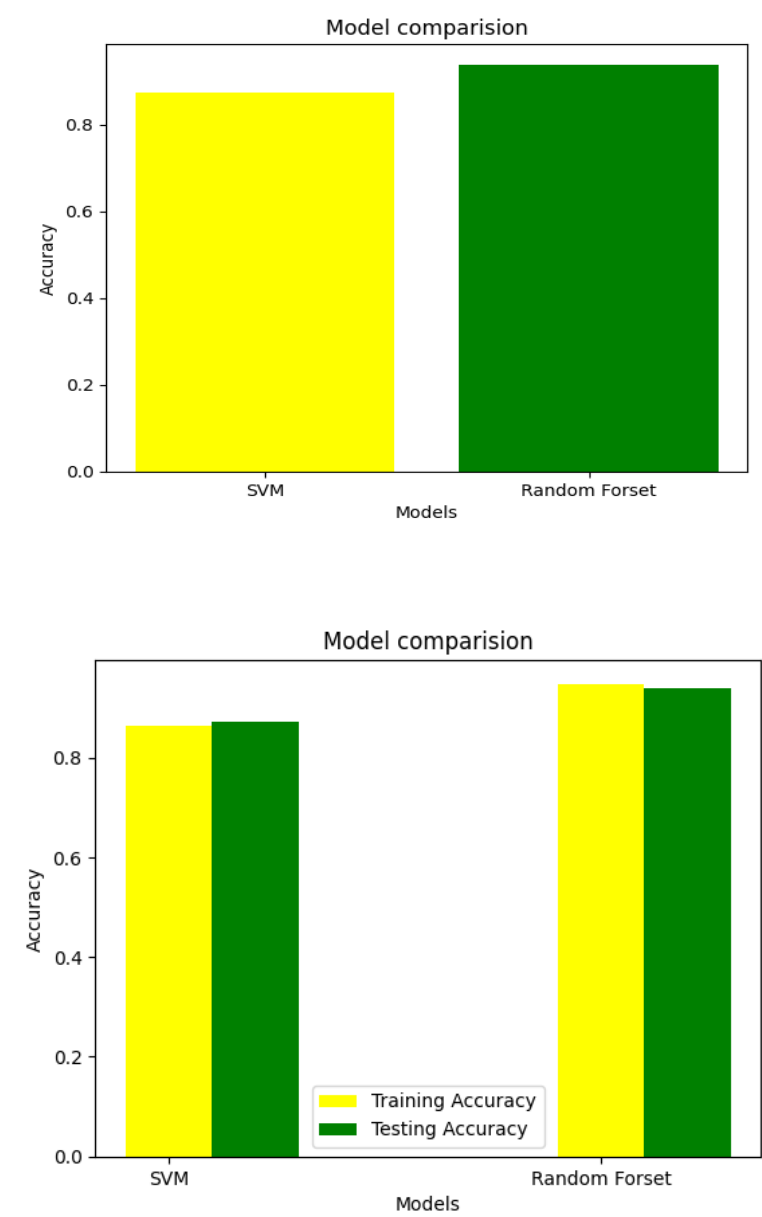


Fig 8.5: Comparision of Models



ASD Prediction Web Application

Autism Spectrum Disorder

Does your child look at you when you call his/her name?

Is it easy for you to get eye contact with your child?

Does your child point to indicate that s/he wants something? (e.g. a toy that is out of reach)

Fig 8.6: ASD Prediction Web App UI -1

The above figure represents the Initial web page of the Autism Spectrum Disorder Prediction model. Here we are required to give the inputs in order to predict the Result.

YES

Does your child follow where you're looking?

NO

If you or someone else in the family is visibly upset, does your child show signs of wanting to comfort them? (e.g. stroking hair, hugging them)

NO

Would you describe your child's first words?

NO

Does your child use simple gestures? (e.g. wave goodbye)

YES

Does your child stare at nothing with no apparent purpose?

YES

Age_Mons

12 36

36

Fig 8.7: ASD Prediction Web App UI -2

In the above figure we are required to give the inputs for the given questions. Bases on those inputs the model will predict the result, whether the person is ASD or non ASD.

The screenshot displays a web application interface for ASD prediction. It features several input fields with yellow labels: 'Age_Mons' (a slider set to 28), 'Gender' (a dropdown menu set to 'Male'), 'Ethnicity' (a dropdown menu set to 'Asian'), 'Jaundice' (a dropdown menu set to 'NO'), 'Family_mem_with_ASD' (a dropdown menu set to 'YES'), and 'who Completed The Test' (a dropdown menu set to 'Family member'). A red box highlights the 'ASD Test Result' button. Below the button, a green box displays the prediction: 'Patient is ASD'.

Fig 8.8: Final output page of ASD Prediction

The above figure represents the required inputs from the user and after giving the inputs it predicts whether the user has ASD or not. The above figure represents the user has ASD.

CONCLUSION

In this work, different machine learning techniques were used for the prediction of Autism Spectrum Disorder. A comparison on the accuracy for a particular data set was performed by using ML techniques. We have used Support Vector Machine (SVM) and Random Forest Classifier algorithms. From the above results, it is inferred that out of all models considered and its performance, Random Forest model is most accurate that gives a good prediction accuracy percentage(93.83%).

REFERENCES

- [1].Khondaker.A, Sharmistha.B, Md. Anwar, Evdokia.A, Jessica B, Shaheen, Mohammad,” Smart Autism - A mobile, interactive and integrated framework for screening and confirmation of autism” IEEE-2016.
- [2].Suman Raj and Sarfaraj Masood,‘Analysis and Detection of Autism Spectrum Disorder Using Machine Learning Techniques’(2019).
- [3].Vaishali, R., and R. Sasikala. “A machine learning based approach to classify Autism with optimum behaviour sets. (2018)” International Journal of Engineering & Technology.
- [4].J. A. Kosmicki, V. Sochat, M. Duda, and D. P. Wall. (2015) “Searching for a minimal set of behaviors for autism detection through feature selection-based machine learning”.
- [5].Kaushik Vakadkar , Diya Purkayastha , Deepa Krishnan (2021). “Detection of Autism Spectrum Disorder in Children Using Machine Learning Techniques.

