A project report on

Mental Illness Prediction System

Submitted in partial fulfilment for the award of the degree of

Master of Computer Applications

By

Tusharika Joshi
(19MCA0051)

School of Information Technology & Engineering (SITE)

VIT, Vellore

April, 2020

DECLARATION

I hereby declare that the thesis entitled “Mental Illness Prediction System" submitted by me, for the award of the degree of *Masters of Computer Application* to VIT is a record of Bona fide work carried out by me under the supervisionof Dr.Brindha K.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore
Date :

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled “Mental Illness Prediction System" submitted by **Tusharika Joshi (19MCA0051)**, **School of Information Technology**, VIT, for the award of the degree of *Masters of Computer Application*, is a record of Bona fide work carried out by him / her under my supervision during the period, 08. 08. 2020 to 07.06.2020, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Brindha.K 06.06.2021

Signature of HOD **Signature of the Guide**

Internal Examiner *External Examiner*

ABSTRACT

Mental health has always been taken for granted. The most appreciated body part, the one we are told to use excessively right from our childhood, is the least look forward to. And that is very saddening. With increase in use of artificial intelligence, it’s use in mental health care is also increasing. The cost of treatment of mental illnesses is very high, hence resulting in people never visiting to a professional and not even knowing they have a health issue. Aim of this paper isto build an efficient system for prediction of mental illness, which can further be used on applications or websites that our freely available to everyone, so if they use their symptoms they can self-diagnose to some extent and hence reach out for help if required. This work will be done using different techniques such as, Random forest, Multilayer preceptor etc. And then applying Ensemble learning to the used algorithms, comparing results and to come up with the best one.

ACKNOWLEDGEMENT

It is my pleasure to express with deep sense of gratitude to my guide Prof. Brinda K, Associate professor, SITE, Vellore Institute of Technology, for her constant guidance, continual encouragement, understanding; more than all, she taught me patience in my endeavor. My association with her is not confined to academics only, but it is a great opportunity on my part of work with an intellectual and expert in the field of MCA.

I would like to express my gratitude to G. GOVINDASAMY VISWANATHAN (Chancellor), Dr. ANAND A. SAMUEL (Vice Chancellor), and Dr. BALAKRUSHNA TRIPATHY (Dean), SITE, for providing with an environment to work in and for his inspiration during the tenure of the course.

In jubilant mood I express ingeniously my whole-hearted thanks to all teaching staff and members working as limbs of our university for their not self-centered enthusiasm coupled with timely encouragements showered on me with zeal, which prompted the acquirement of the requisite knowledge to finalize my course study successfully. I would like to thank my parents for their support.

It is indeed a pleasure to thank my friends who persuaded and encouraged me to take up and complete this task. At last, but not least, I express my gratitude and appreciation to all those who have helped me directly or indirectly toward the successful completion of this project.

Place: Vellore Name of the student

Date: Tusharika Joshi(19MCA0051)

EXECUTIVE SUMMARY

Autism is a not so uncommon disease, the reference of the disease can be seen in many movies and TV shows. The system developed is to find the possibility of having Autism in a Toddler, that is small children at very young age. It takes common parameters like, age, gender, ethnicity, etc. in order to predict the output. Dataset for this paper is taken from Developed by Dr Fadi Fayez Thabtah using a mobile app called ASDTests to screen autism in toddlers.. There are mainly two datasets one for training the model, another one for testing it. Algorithms used are Random Forest, MLP Classifier, AdaBoost, SVM, KNN. Also, ensemble learning is applied through stacking. Challenges faced in the paper were due to occurrence of overfitting while using stacking, due to which high accuracy is not achieved.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
ASD	Autism Spectrum Disorder
SVM	Support Vector Machine
DT	Decision Tree
LR	Logistic Regression
NLP	Natural Language Processing
ML	Machine Learning
KNN	K-Nearest Neighbors
AdaBoost	Adaptive Boosting

Chapter 1

INTRODUCTION

1.1. INTRODUCTION

Mental illnesses, the, “it’s just in your head”, illnesses. The “you aren’t taking any medication, are you?”, illnesses. The “do yoga, just think positive.”, illnesses. Brain, a part of our body, one of the most important part of our body and is sadly the most neglected one. Get a fever, you’ll be asked regularly if you’re doing ok. Get Cancer, everyone will send you wishes and would pray for you. Get depression and you’ll get laughed at. Get Autism, Schizophrenia, ADHD, you’ll get laughed at. You will be labelled “just lazy”, “crazy”. But would not be taken to a doctor. Because these are not “get well soon card” illnesses. The awareness and education on mental illness is so low in our society, especially in India.

The rise of Artificial Intelligence could therefore be used as in this field to help patients and doctors. The paper here Aims to do the same. The system built will allow user to predict the possibility of Autism present by entering relevant attributes.

AUTISM

Autism Spectrum Disorder (ASD) is a term used to describe various neurological disorders.

The disorder can be problem while communicating or other social interactions. There can be seen patterns of behavior in Autistic Patients.

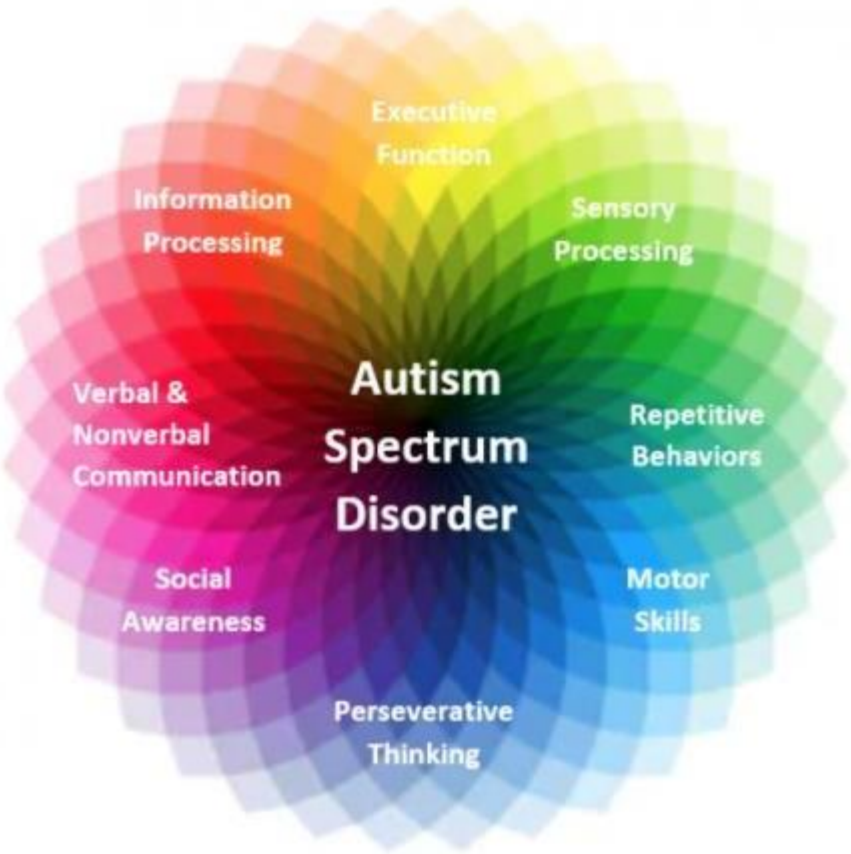


Fig 1.1 Autism Spectrum Disorder

Autism is a condition where patient suffers from a neural behavioral condition. Due to variety and diversity of symptoms it is termed as Autism Spectrum Dysfunction or Autism Spectrum Disorder. An individual autistic person’s symptom can be very different from another person with autism and so on. However, there are 3 basic symptoms that are basic in all types of autism:

- Social interaction challenges
- Deviated responses and difficulty in communication
- Cognitive dysfunction and insensitivity or over sensitivity to certain stimuli including sight, hearing, smell, taste and touch, tendency to engage in repetitive behaviour throughout life on a daily basis. It may involve pacing, head rocking and hand flipping etc

Autism is known to effect one in every 60 individuals with boys at five times more risk to be autistic than girls. What causes autism is still unknown. It is believed that the cocktail of genetic, environmental and psychological factors contributes towards autism.

Autism is broadly divided into three major types:

- Kanner’s Autism, also called Severe Autism.
- Asperger’s Syndrome, also called High Function Autism.
- Rhett Syndrome.
- Childhood Disintegrative Disorder.
- Pervasive development disorder, also called Atypical Autism.

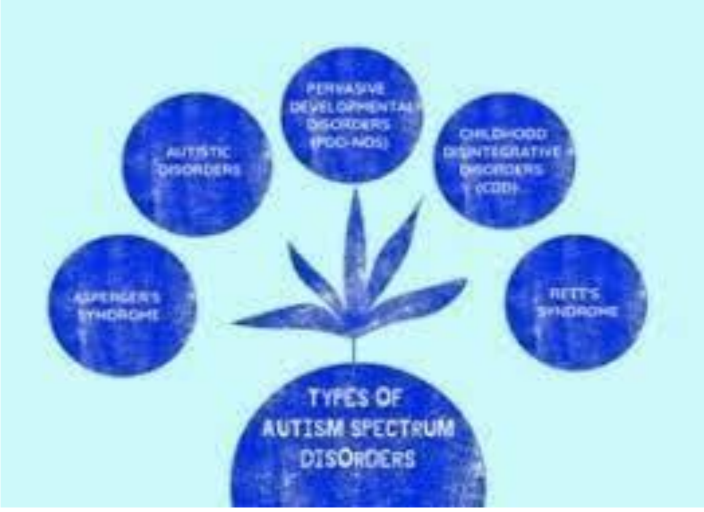

PERVASIVE DEVELOPMENTAL DISORDERS (PDD-NOS)
AUTISTIC DISORDERS
CHILDHOOD DISINTEGRATIVE DISORDERS (CDD)
ASPERGER'S SYNDROME
RETT'S SYNDROME
TYPES OF AUTISM SPECTRUM DISORDERS

Fig 1.2 Types of Autism Spectrum Disorders

1.2 OVERVIEW

The aim of this project, as described in abstract and other five places in this document, is to build an efficient system for detection of early traits of autism or ASD in toddlers. Where required data is basic things like Age, Ethnicity etc. Using various Machine Learning Algorithms. Random Forest, AdaBoost, Multilayer Perceptron, K-Nearest Neighbour, Support Vector Machine. And finally, applying stacking, using all these algorithms as base models.

1.3 OBJECTIVE

Aim of this paper is to build an efficient system for prediction of mental illness, which can further be used on applications or websites that our freely available to everyone, so if they use their symptoms they can self-diagnose to some extentand hence reach out for help if required. This work will be done using different techniques such as, Random forest, Multilayer preceptor etc. And then applying Ensemble learning to the used algorithms, comparing results and to come up with the best one.

Chapter 2

BACKGROUND WORK

2.1 LITERATURE SURVEY

Predictive modelling in e-mental health: A common language framework- This research was done to build a connection between different domains of research on the topic. They first gave an overview of predicting methods used in data mining. Then they proposed the idea of characterizing the analysis in mental health care on 3 dimensions: 1. Time since treatment; 2. Types of available data; 3. Clinical decision. Based on these, they came up with a framework identifying 4 model types which can be used for classification of existing or future work. They used this framework to categorize published research on mental health that used predictive modelling. According to them, single predictors do not provide a basis for prediction.[1]

Application of Data Mining Techniques to Healthcare Data- Top tier research on data mining by comparing it to traditional statistics. Figured out advantages of automated data systems. Description of data mining techniques and algorithms. Three healthcare applications using data mining explained. Found out automated systems offer more advantages over manual ones.[2]

Artificial Intelligence for Mental Health and Mental Illnesses: An Overview- Gave an overview of use of artificial intelligence in mental health. EHR (Electronic Health Records), Brain Imaging etc. were used. Concluded that AI is more promising and efficient than manual methods of diagnosis. A handful of researchers have invested in this field.[3]

Data Mining Algorithms and Techniques in Mental Health: A Systematic Review- Compared different techniques of data mining for common mental illnesses like-depression, dementia, schizophrenia, Alzheimer etc. The most commonly used techniques as figured out are- Naïve Bayes, Random Forest, Support Vector Machine (SVM), Artificial Neural Networks (ANN), Decision Tree (DT), Logistic Regression. At the end they concluded that their study can be useful for further research. They gave percentage of usage of each technique for each individual illness and then also overall percentage.[4]

Deep learning in mental health outcome research: A scoping review- Previous work on Artificial Intelligence in Mental Health was reviewed, and articles were characterized into the following categories: prognosis and diagnosis based on clinical data, assessment of genetics and genomics data to better understand mental health conditions, analysis of vocal and visual expression data for disease detection, and estimation of risk of mental illness using social media data. Concluded that deep learning is improving.[5]

AI in mental health- Reviewed about Artificial Intelligence in mental health. Data mining from social media interactions and digital devices can be used to predict mental health issues, this idea is proposed in this paper. Computer based Natural Language Processing (NLP) is used to work upon proposed idea. Concluded that AI can not only be used in diagnosis but also for treatment.[6]

Artificial Intelligence Approaches to Predicting and Detecting Cognitive Decline in Older Adults: A Conceptual Review- Talking about mental illness in older adults. Called pathological cognition. Talks not only about merits, but also limitations of using Artificial Intelligence for prediction, diagnosis and treatment. 6 categories of studies: sociodemographics, clinical and psychometric assessments, neuroimaging and neurophysiology, electronic health records and claims, novel assessments and genomics.[7]

Machine learning in medicine: A practical introduction- Provide practical guide for implementing machine learning for predictive algorithm. Expression given by using 3 Machine learning models: General Linear Model Regression (GLMs), Single Layer Artificial Neural Network, Support Vector Machines (SVMs), achieved accuracy 0.94 - 0.96, sensitivity 0.97 - 0.99, and specificity 0.85 - 0.94. Maximum accuracy 0.96 and area under the curve 0.97 was achieved using the SVM algorithm. Prediction performance increased marginally to 97% accuracy, 99% sensitivity and 95% specificity.[8]

Machine learning for precision psychiatry- Aimed at introducing psychiatrists and clinicians the use of machine learning in mental health, or health, in general for their practice. support vector machines, modern neural-network algorithms, cross-validation procedures techniques shown. Concluded that using statistical models can increase efficiency of diagnosis, especially with genetic cases.[9]

The utility of artificial intelligence in suicide risk prediction and the management of suicidal behaviors- Using Big Data to generate risk algorithms that can determine included risk factors. At a substantial level, selected algorithms can identify groups that might be at risk or suicide hotspots, which will help inform right resources. Artificial intelligence also has been used to help clinically manage suicide across diagnostic centers, managing medication and delivery of behavioral therapy.[10]

Integrating Artificial and Human Intelligence in Complex, Sensitive Problem Domains: Experiences from Mental Health- Human value for getting meaningful stuff out of Artificial Intelligence techniques. Concluded that AI can give better performance if human feedback is kept in mind too.[11]

Methods in predictive techniques for mental health status on social media: a critical review- Studied 75 papers. Showed the methods of data footnotes for status of mental health, collection of data and management of quality, pre-processing and feature selection, and model selection and verification. Despite increasing interest in the field, they identified some concerning drifts around the construct validity, and a lack of rumination in the methods used for functionalizing and identifying mental health status.[12]

Prediction of Mental Disorder for employees in IT Industry- Took dataset from questionnaire given to IT employees. Different Machine Learning techniques used. Confusion Matrix, true positive, true negative, false positive, false negative. Accuracy, Precision, AUC Score were found. Concluded most of the employees need medical attention.[13]

Data Analytics in Mental Healthcare- Survey paper on Bigdata and Mental health connection. Examined different prediction methods. Naïve Bayes, k-means etc. [14]
Artificial intelligence in prediction of mental health disorders induced by the COVID-19 pandemic among health care workers-The paper aims at early prediction of mental health issues in Health care workers in time of Pandemic. They are to go through 5 phases namely, objective evaluation of the intensity of worker’s exposure to stress,

detailed self-report on stress experienced during the COVID-19 pandemic, Phase 3) designing and development or model based of worker's phycological and neurological experiences, measurement of computation etc. according to reaction of worker. statistics as well as machine learning based analysis of highly diverse sets of data that were obtained from previous phases. [15]

Prediction of Mental Health Problems Among Children Using Machine Learning Techniques- Used 8 Machine learning techniques been compared, dataset consisting of 60 cases. Feature Selection algo reduced the sully attribute dataset. 3 algos, multilayer Perceptron, Multiclass Classifier and LAD Tree and only slight difference was found in full and selected dataset.[16]

Machine learning in mental health: A systematic scoping review of methods and applications- 4 domains selected, Detection and diagnosis, Prognosis, treatment and support, Public health, Research and clinical administration. Depression, schizophrenia, Alzheimer are focused diseases. Algorithms used are SVM, decision tree, neural networks. [17]

Behavioral Modelling for Mental Health using Machine Learning Algorithms- Using support vector machines, decision trees, Naïve Bayes classifier, K-nearest neighbor classifier and logistic regression to identify state of mental health in a target group. Obtained cluster labels used to build classifier.[18]

Machine Learning in Healthcare: A Review- Review of various machine learning algorithms to help lessen the research slit to build an efficient system for medical applications. Results concluded the following algorithms work best for respective diseases: Naive Bayes at 86% accuracy for diagnosis of heart disease, SVM at 96.40% accuracy for breast cancer diagnosis, and CART at 79% accuracy for recognition of diabetes.[19]

Machine learning methods in psychiatry: a brief introduction- Supervised and unsupervised learning are discussed and researched in depth using Linear Regression and k-means clustering.[20]

Chapter 3

TECHNICAL SPECIFICATIONS

3.1 METHODOLOGIES

Artificial Intelligence (AI), is growing rapidly in the growing world. From houses to roads, it's becoming a necessity day by day. We literally have it in our hands in form of hundreds of mobile applications we've downloaded on our smartphones.

The several subtypes of Artificial Intelligence, Machine learning, Neural Networks, Fuzzy Logic, Deep Learning, Natural Language Processing, Computer Vision etc. play a vital role in almost every application and gadget we use.

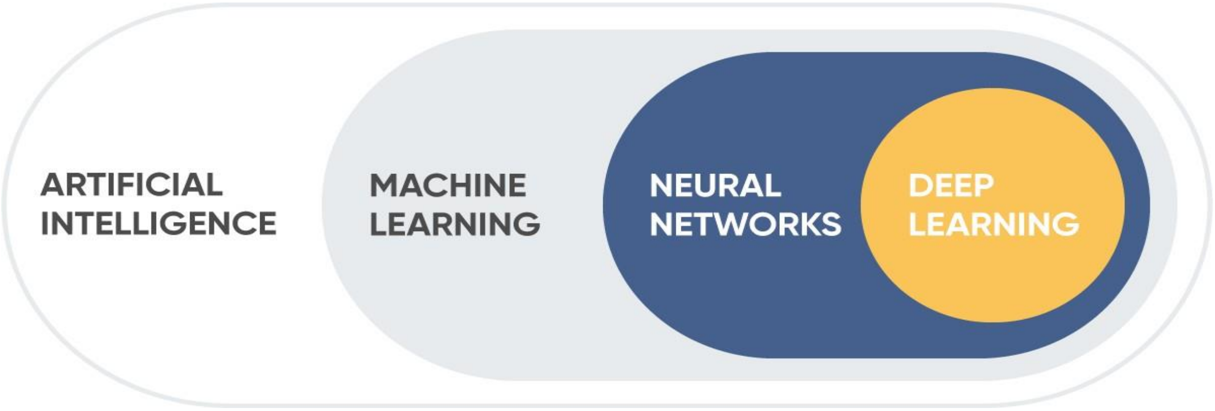


Fig 3.1 Artificial Intelligence

3.1.1 MACHINE LEARNING

Machine Learning (ML) is a subfield of Artificial Intelligence that is widely and widely used. Working of machine learning is quiet simple, provide the algorithm a dataset, train it, get the output based on training.

The use of Machine Learning technologies in predictive modelling is very common, whether be in medical field or any other. Having mainly 3 types of learning techniques, namely Supervised Learning, Unsupervised Learning and Reinforcement/Unsupervised learning.

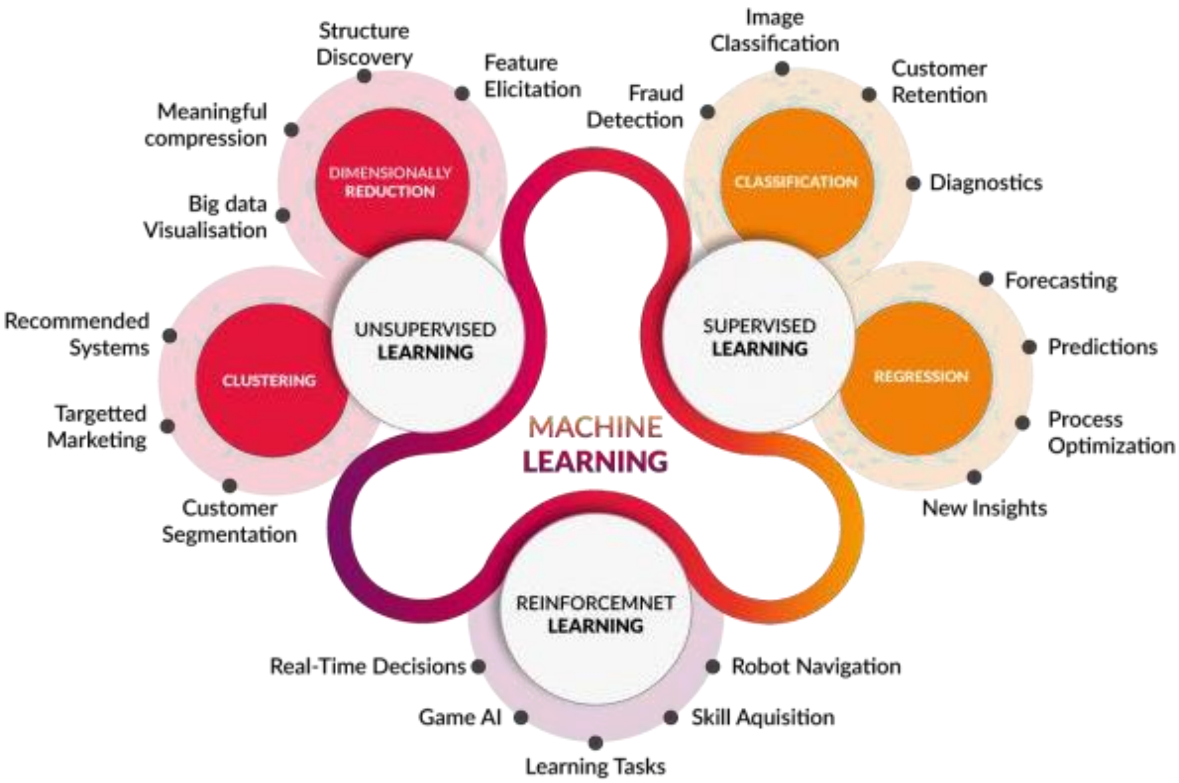


Fig 3.2 Machine Learning

When Machine Learning is applied on data samples, each sample consists of different components, each of them has different features and distinct values. The choice making for choosing the most appropriate techniques for analysis and sampling of data is to be made by the developers and it is a very important choice.

Machine Learning is a set of methods that allows software applications to predict significantly better results without having to be designed specifically. The core assumption of machine learning is to create algorithms that take in input data and utilize statistical analysis to predict output data, with the output data being updated as new data is collected. Machine learning is related to data mining and predictive modelling in terms of the techniques involved. Both must look for specific patterns by date and change the program's actions accordingly. Machine learning is something that many people are familiar with thanks to internet shopping and advertisements that are sorted according to why they buy. This is because referral engines use machine learning to customize ads that are delivered online, almost in real time. In addition to personalized marketing, other known cases in which machine learning is used are fraud detection, spam filtering, threat detection, network maintenance, predictability, and news flow building.

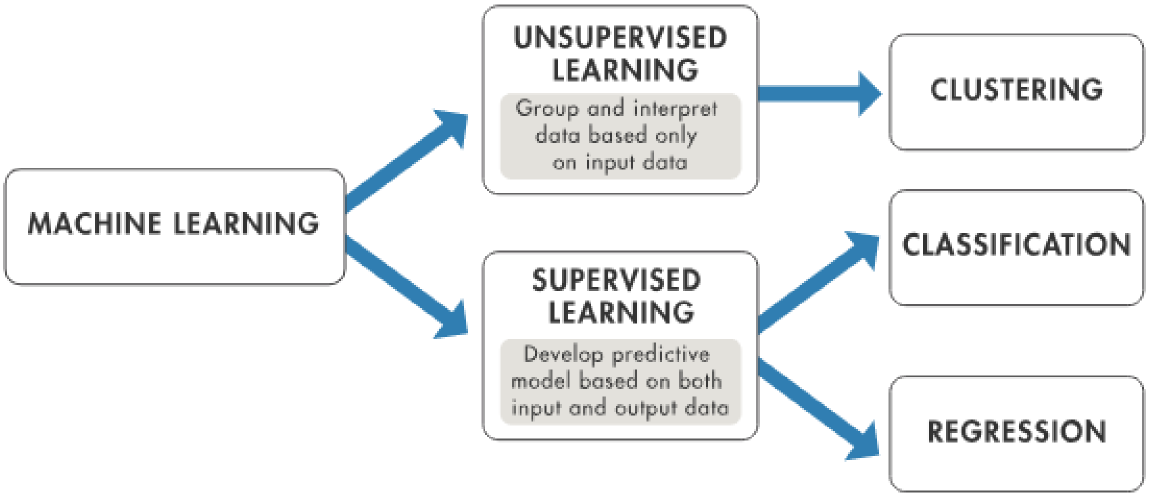

MACHINE LEARNING
UNSUPERVISED LEARNING
Group and interpret data based only on input data
SUPERVISED LEARNING
Develop predictive model based on both input and output data
CLUSTERING
CLASSIFICATION
REGRESSION

Fig 3.3 Supervised Unsupervised Learning

3.1.2 SUPERVISED LEARNING

Supervised learning is when we feed input to the system and also show it the output and have a mapping function, y= f(x) between the input and the output.The aim of model will be to reach to maximum approximation with the mapping function so that when new input is given, it can predict the output correctly.

A mapping function can be anything starting from a linear function, y=x to a complex exponential function, 1/1+e^(-x).

The reason it is called Supervised Learning is because the algorithm is supervised while it trains. The output is already known so whenever it makes a mistake, it is corrected by the supervisor. It keeps on learning until an acceptable amount of accuracy is achieved.

Supervised learning algorithms need a data scientist, or data analyst, who partakes the knowledge of Machine Learning to process the desired input and output information as well as to provide feedback on the precision of the predictions made by the algorithm. Supervised learning consists of two stages:

- Training and fitting the model using dataset.
- Predicting output for new data using the knowledge gained in training stage.

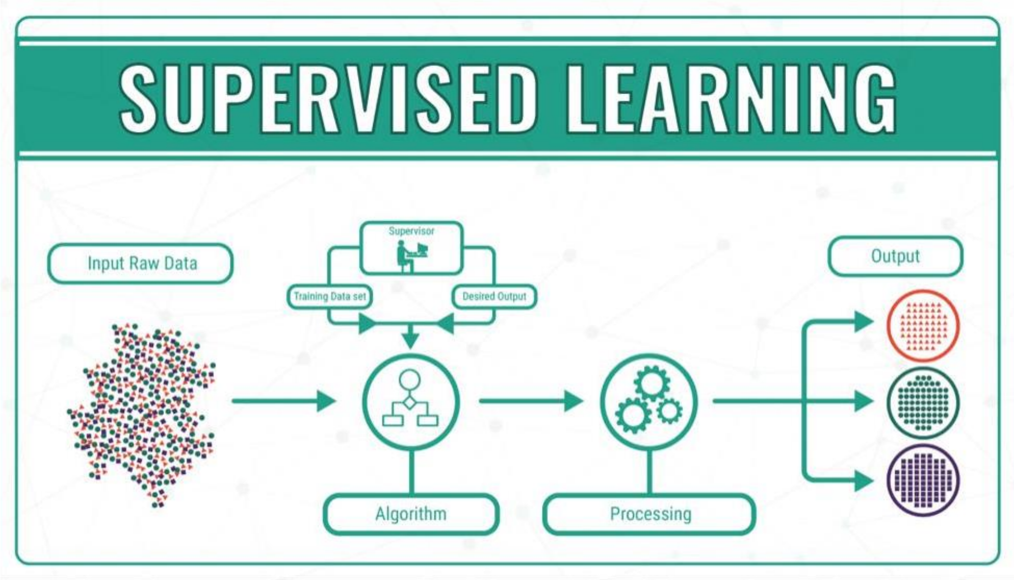
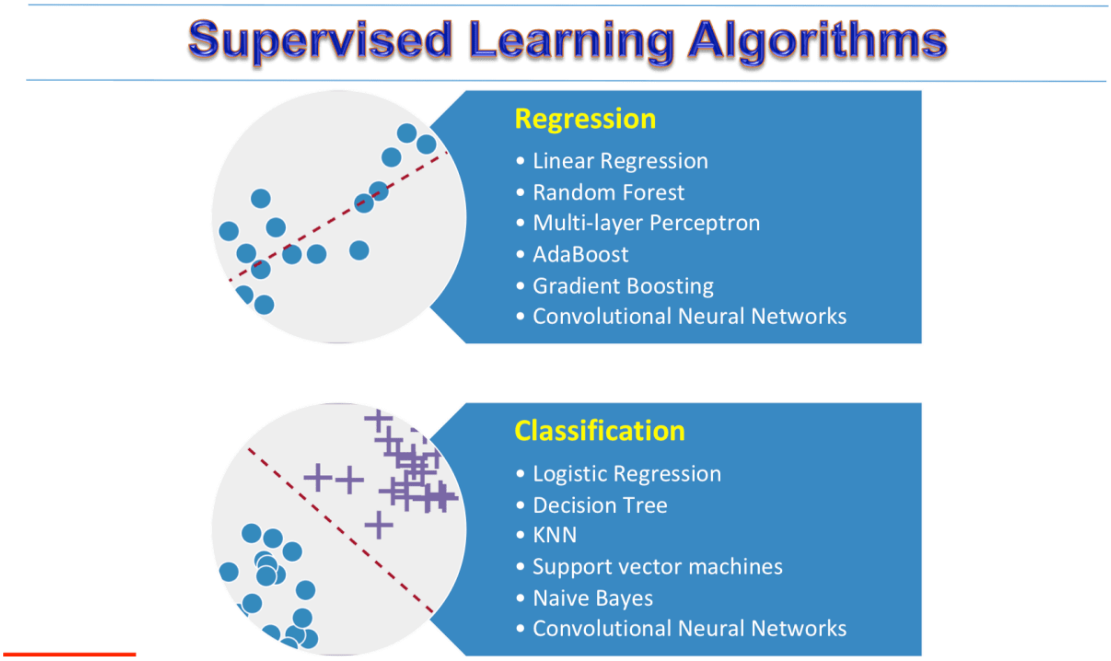


Fig 3.4 Supervised Learning

Supervised learning provides 2 types of Algorithms:

- Classification Algorithms
- Regression Algorithms



Supervised Learning Algorithms

Regression

- Linear Regression
- Random Forest
- Multi-layer Perceptron
- AdaBoost
- Gradient Boosting
- Convolutional Neural Networks

Classification

- Logistic Regression
- Decision Tree
- KNN
- Support vector machines
- Naive Bayes
- Convolutional Neural Networks

Fig 3.5 Supervised Learning Algorithms: Regression & Classification

In this paper we have used Classification Algorithms of Supervised Learning. Namely, K-Nearest Neighbour Classifier (KNN), Support Vector Classifier (SVM), Random Forest Classifier and AdaBoost Classifier.

3.1.3 CLASSIFICATION AND REGRESSION ALGORITHMS

Classification and Regression algorithms both comes under the supervised learning algorithm.

Regression algorithm is basically used for the prediction of the continuous values such as age, Salaries, prices. Etc.

Classification algorithm is basically used for prediction or classification of t he discrete/continuous values such as male female, true and false, yes or no.

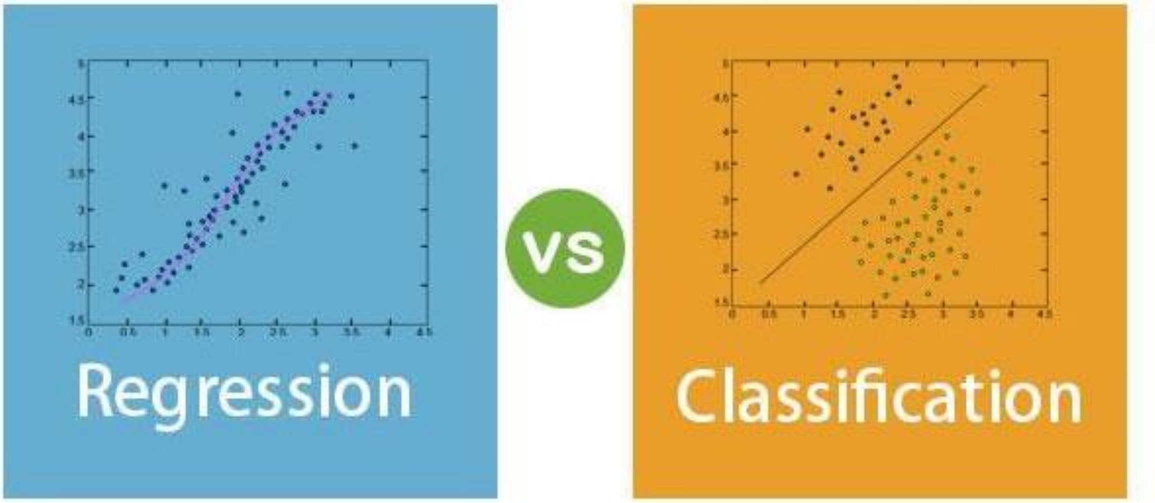


Fig 3.6 Outputs of Classification and Regression Prediction

3.1.4 CLASSIFICATION ALGORITHMS

In the classification process the division of the dataset is done on the basis ofdifferent parameters of the classes. Different Classes are formed after categorizing a data.

3.1.4.1 K-NEAREST NEIGHBOR

This is one of the simplest algorithms in Supervised Learning Approach of machine learning. K-nearest neighbors can be used for both, classification as well as regression problem. KNN is a classification algorithm it is a part of supervised learning. The similarity is being assumed between new and available cases. After that it put it into the most similar category.

Based on the similarity it classifies the data ad also stores all data. It means ifany new data is coming it will be easy for KNN algo to put them into its category. It is non parametric and not making any assumptions.

KNN is usually called a lazy learner because initially it does not learning from training data instead it stores the dataset and when classification is performed at that time it takes any action on data. At initial phase it simply stores the dataset and it any new data arrives at that it performs classification and putting into category which is similar to new data.

The assumption is that similar things lie close to each other. The idea origins with basic mathematics, i.e. if distance between given points, using distance formula, is zero, it implies that the points are same.

The value given to K determines the number of nearest neighbors the algorithm is going to check. For k=1, only a single neighbor would be searched which is, as obvious, not a reliable decision. So, in order to increases precision and accuracy, the value of K is increased according to the dataset.

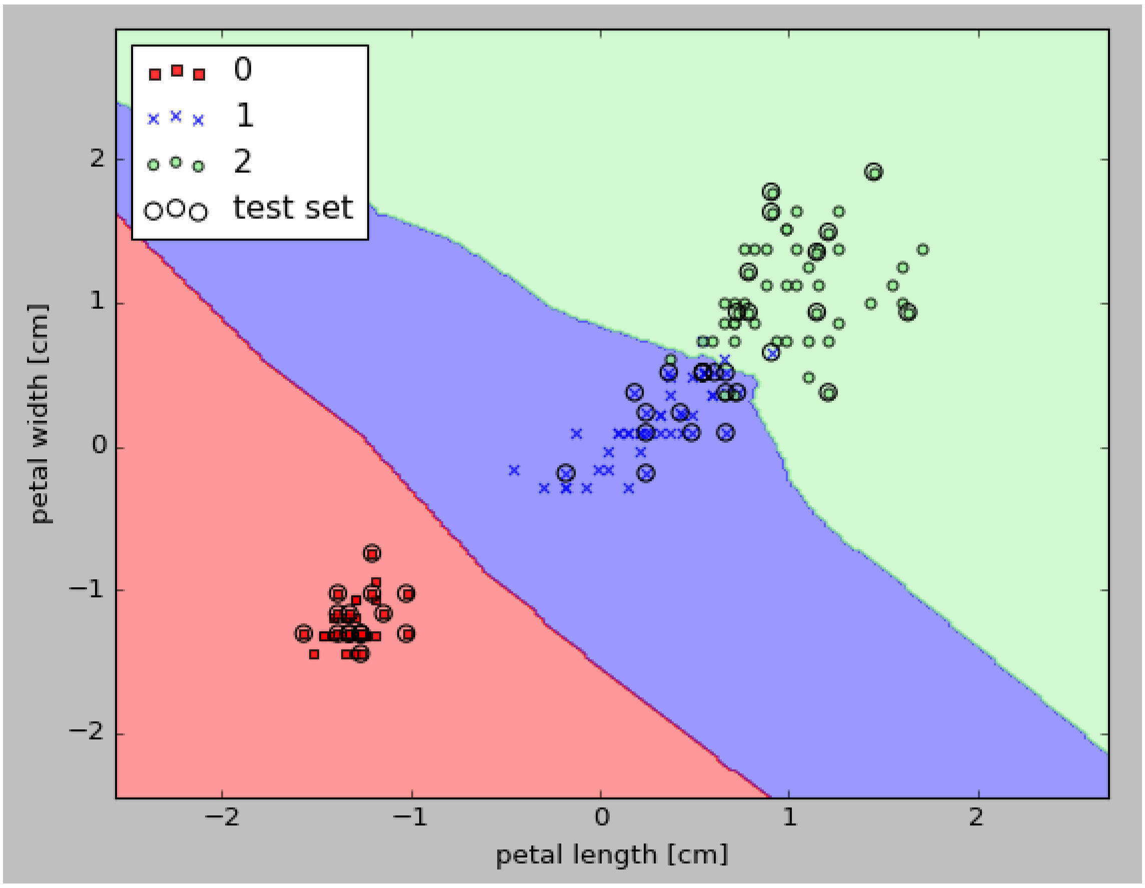


Fig 3.7 K-Nearest Neighbors Output

Working

- Selection of K-neighbors.
- Distance is identified of k numbers of neighbors with the help of Euclidean formulae.
- Selecting the k neighbor according to the value of formulae.
- In the k neighbors, select the data point of each category.
- After that assign that new data point into the category of maximum number of neighbors.
- This is KNN models gets ready.

In the below diagram we have a two categories A and B and new data point is arrived.

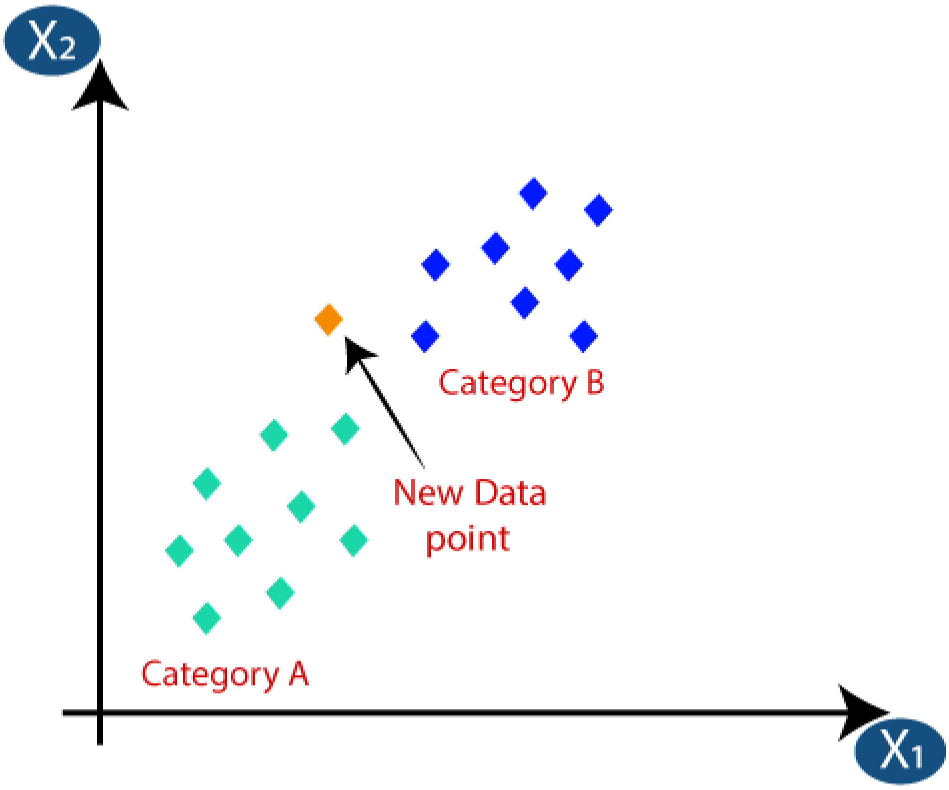


Fig 3.8 New Data Point

Then we choose the k number of neighbors by determining value of k supposing value k=5. Then we will calculate the distance with the help of Euclid distance formula for putting it into same category.

The formula is:

$$\sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$$

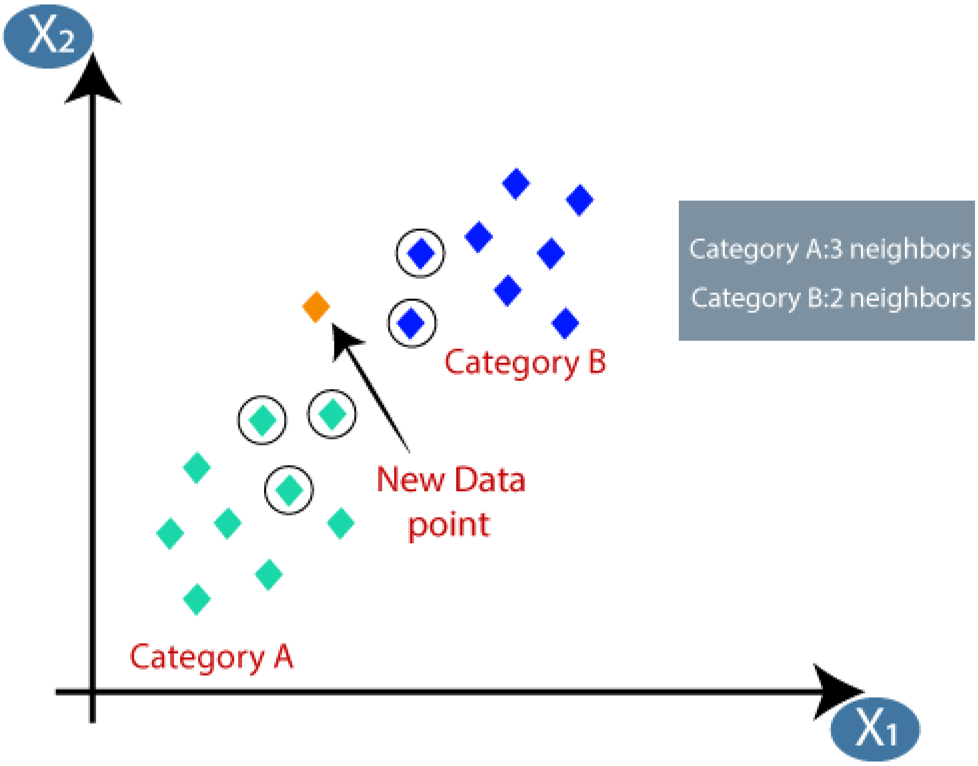


Fig 3.9 Counting of Categories

In the above fig we can see we got three neighbor of category A so we will put that new data point into the category A.

Selection of K in KNN

- In this we take a assumed value there is particular method for selecting the value of K.
- We should take a higher value of k because lower value such as 1 or 2 are noisy and the outliers of the model can be affected.
- Prefer large value.

Advantages:

- Simple and easy to use.
- Versatile, can be used for both, classification and regression.

Disadvantages:

- Gets slower with increase in fields of data point

3.1.4.2 SUPPORT VECTOR MACHINE(SVM)

Support vector machine is use for classification as well as regression analysis. Support vector machine is suitable for non-linear dataset and helps to reduce the mis categorization rate. The major objective of support vector machine is to find the minimum distance among classes and maximized distance with other classes.

Support Vector Machine goal is to generate the best line or the decision boundary which will help the new data to be at the correct point category. And the best decision boundary is also known as hyperplane.

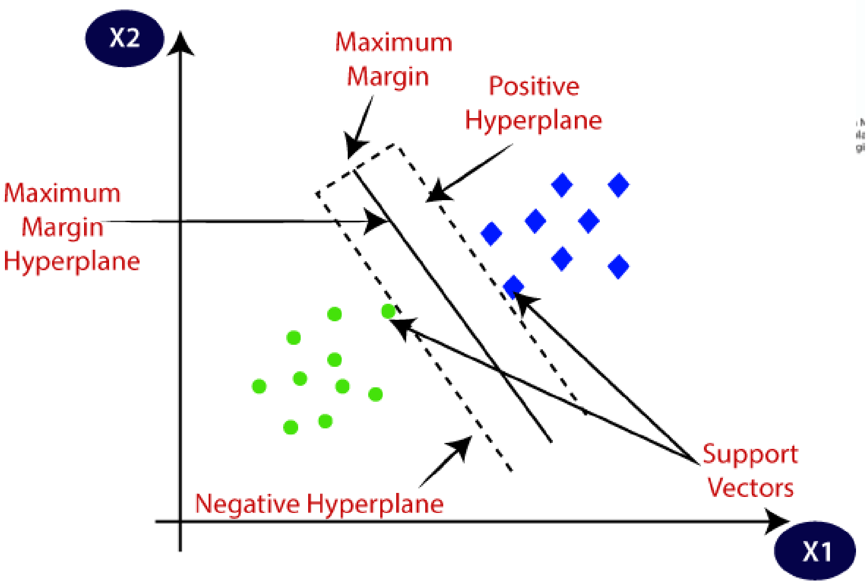


Fig 3.10 Support Vector Machine Model

TYPES OF SUPPORT VECTOR MACHINE ALGORITHM

- Linear Support Vector Machine

  A dataset helps in the classification of the data into two classes by using a single straight line and such data is defined as linearly separable data.

- Non-Linear Support Vector Machine

  A dataset cannot be classified by using a straight line because data is given in cluster form and such data is defined as non-linearly separable data.

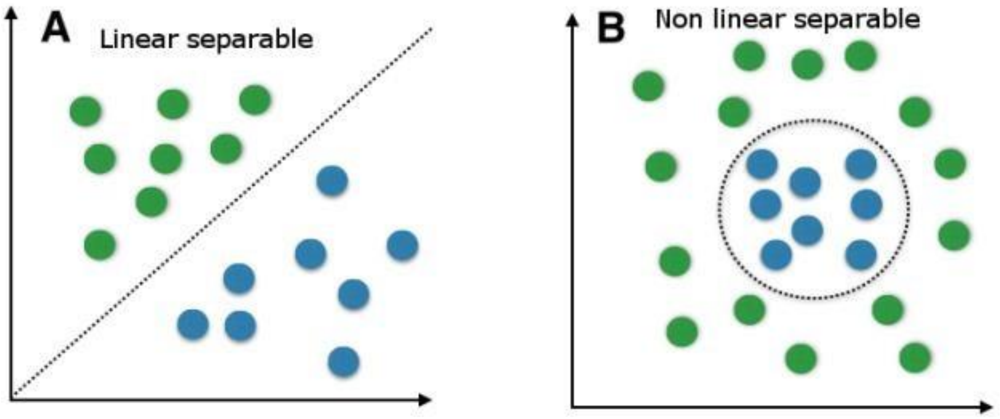


Fig 3.11 Linear and Non-linear separable

Working of Support Vector Machine:

In this there are different scenario how SVM works and identify hyper line.

- Case 1: Here, in the fig, two different classes were taken represented as stars and circles and three hyper lines were drawn

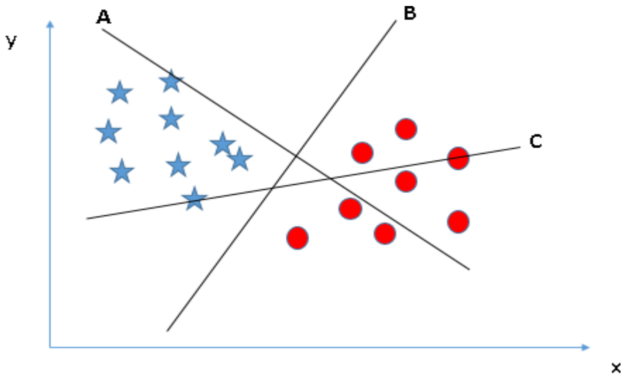


Fig 3.12 Case I

Here A, B, C are hyper line drawn in which we will select B hyper line as it divides the classes perfectly.

- Case 2: In below fig there are 3 hyper line that divides the classes.

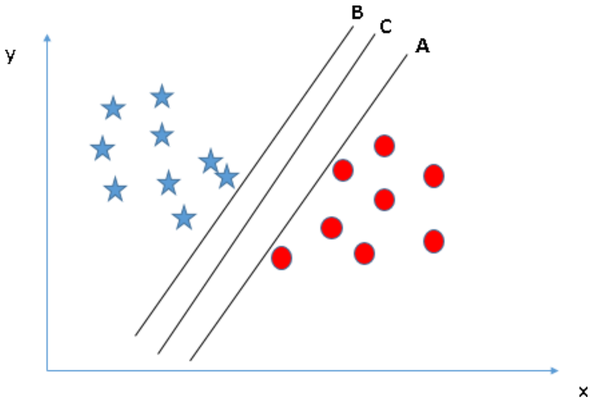


Fig 3.13 Case II

Now we will choose the hyper line which has maximized distance from the classes. Here distance is called Margin. We will choose hyper line C as it has high margin as compared to A and B. if we select line which lower margin than it will be misclassification.

- Case 3: Here two different hyper line were drawn as shown below.

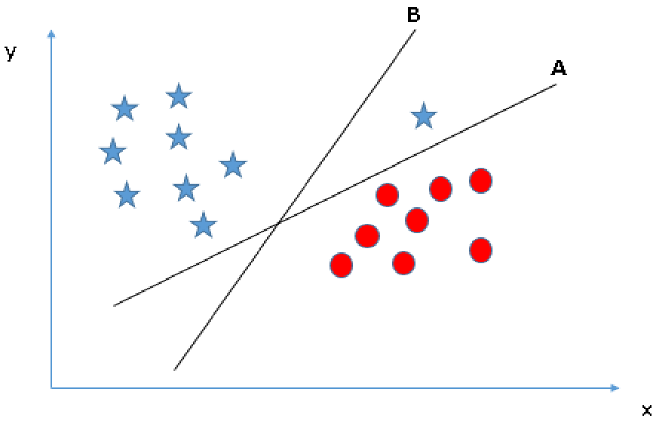


Fig 3.14 Case III

In the above fig we can see that hyper line B has maximum distance from the classes but it is not classifying the classes correctly instead of taking B SVM will consider hyper line A which classify properly.

- Case 4: In below fig we are not able to draw a hyper line.

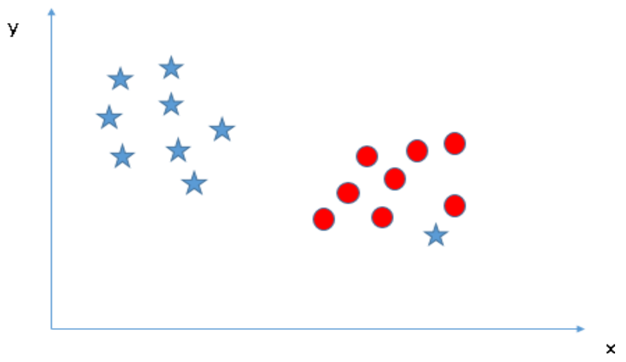


Fig 3.15 Case IV

In the above fig we cannot draw hyper line because there is an outlier of star class in the circle. But SVM does not have any issue it simply ignores the outlier and draws a hyper line as shown in below fig.

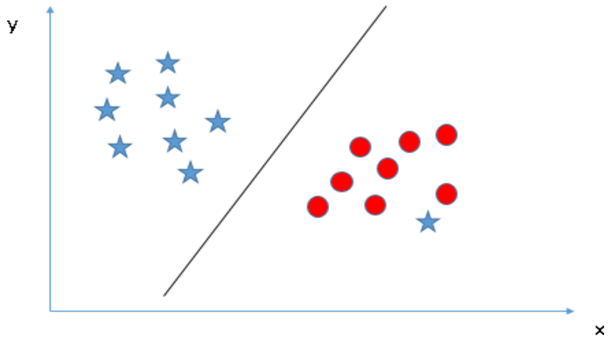


Fig 3.16 Case V

- Case 5: In the Fig we cannot make a hyper line as we have different form of classes.

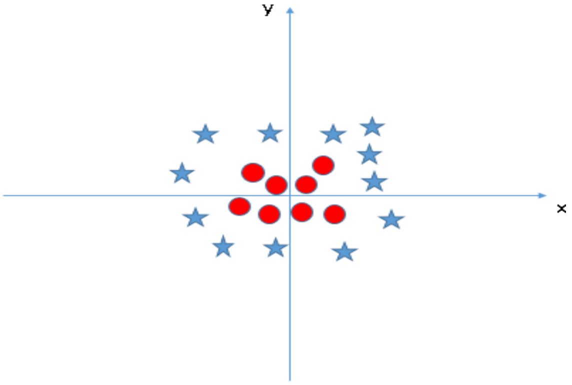


Fig 3.17 Case VI

In the above fig we x axis and y axis perpendicular on each other. Here SVM solves the problem it takes the formulae z=x^2 +y^2 and consider the data which are near the origin. After applying the formulae, the plotting will appear differently as shown in fig 9.

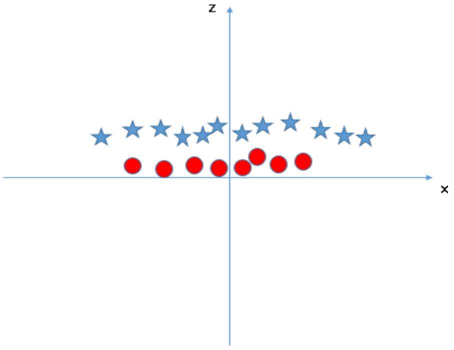


Fig 3.18 Case VII (End Result)

Advantages

- Adaptable to a variety of situations.
- Excellent for non-linear issues.
- Outliers aren't a factor.
- SVM works effectively with margins.

Disadvantages

- It is required to use feature scaling.
- Not Well Known
- Complicated to comprehend
- Does not perform well if dataset have targeted classes and overlapping.

KERNEL SUPPORT VECTOR MACHINE

Non-Linear data works well in Support Vector Machine using a Kernel trick. The purpose and function of a Kernel is to map the low – dimensional input space and it helps to transform into a higher dimensional space.
Radial Basis Function Kernel- Classification of datasets which are not separated on the terms of linear basis so therefore the RBF kernel is most widely used kernel concept which helps in solving the problems.

It gives the best performance with the various parameters used and the results of the RBF kernel are better than the other kernels and there are very less chances of error value in training datasets than any other kernel.

3.1.5 ENSEMBLE LEARNING

Combining hypothesis from multiple algorithms to produce, better, much accurate results is called Ensemble Learning.

The individual machine learning models, when not are able to achieve the desired accuracy, are called weak learners because of their failure to achieve the accuracy.

There are three main ensemble techniques available with respective aiming:

- Bagging: for decreasing variance
- Boosting: for reducing bias
- Stacking: for improving prediction

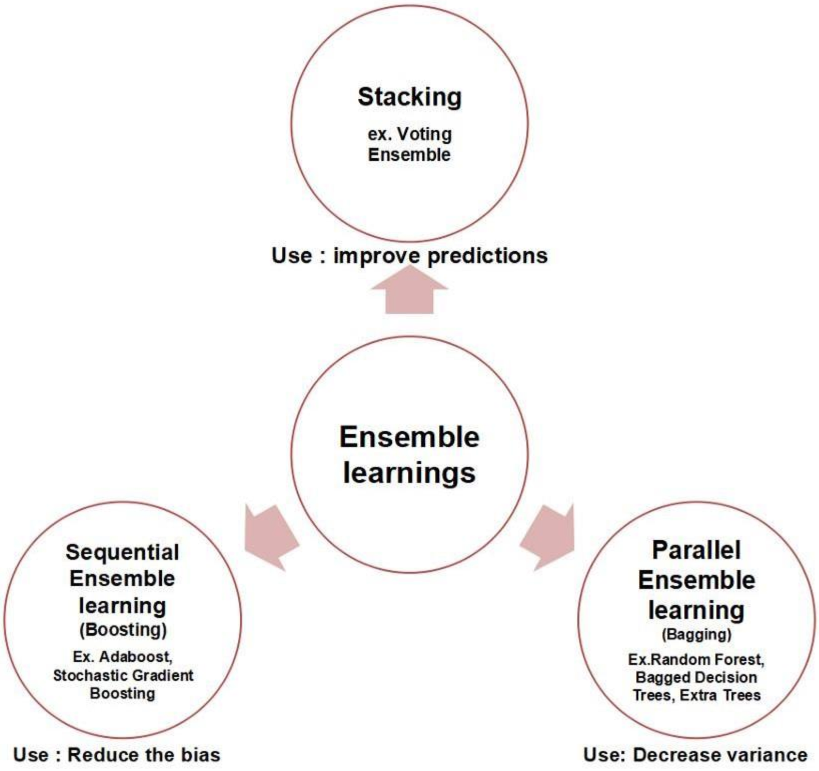


Fig 3.19 Ensemble Learning Approaches

These can be further classified into two groups or categories:

- Sequential Ensemble Methods (Example, AdaBoost): Base learners are dependent upon each other.
- Parallel Ensemble Methods (Example, Random Forest): There is no dependency between base learners, they are independent of each other.

3.1.5.1 BAGGING

Bagging or Bootstrap Aggregation works by reducing variance in the algorithm. This is approached by training same type of algorithm with different subsets of dataset. For example, we can train n trees on different subsets of data, then find

$$f(x) = 1/M \sum_{m=1}^{M} f_m(x)$$

final results by either voting or averaging the outcomes.

Voting is used for classification and averaging is used for regression.

Random Forest is an example of Bagging.

Advantages

- It helps to reduce the over fitting of data.
- It is also suitable for large amount of data.
- There is no effect of missing values on bagging.

Disadvantages

- There is only on disadvantage of bagging is that it gives the final result based on the mean value of subtree

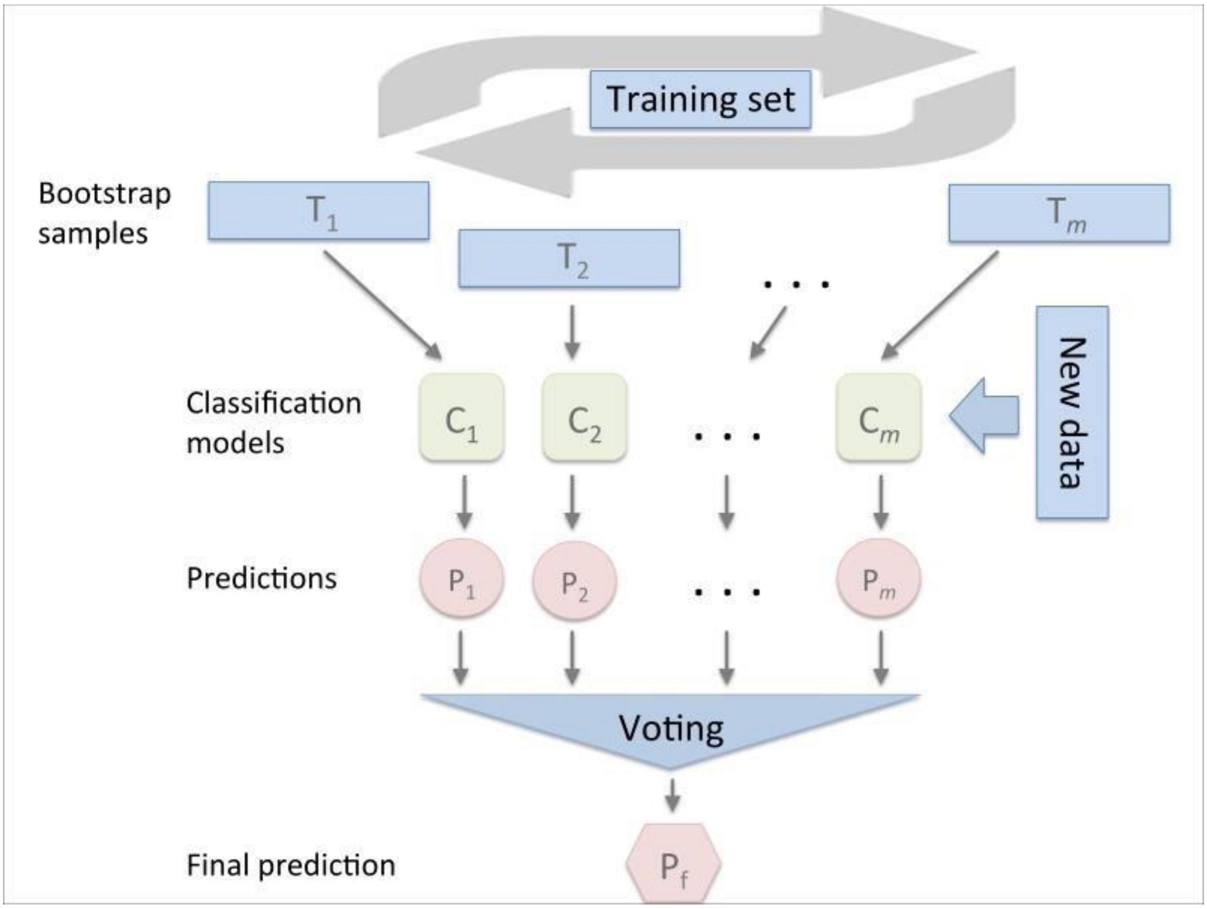


Fig 3.20 Bagging in Ensemble Learning

3.1.5.2 BOOSTING

Boosting method is working same as bagging. In boosting it sequentially fit the different weak learners it handles those in adaptive way: in boosting it focuses on giving importance to observation into the dataset that were mishandled by previous models. Every new model is focused on observation to be fit up so that we get strong learner at the end of the process.

The main purpose of boosting is to avoid the tendency of not taking the full information form the data which may later changes the results if data changes. AdaBoost (Adaptive Boosting) is an example of boosting.

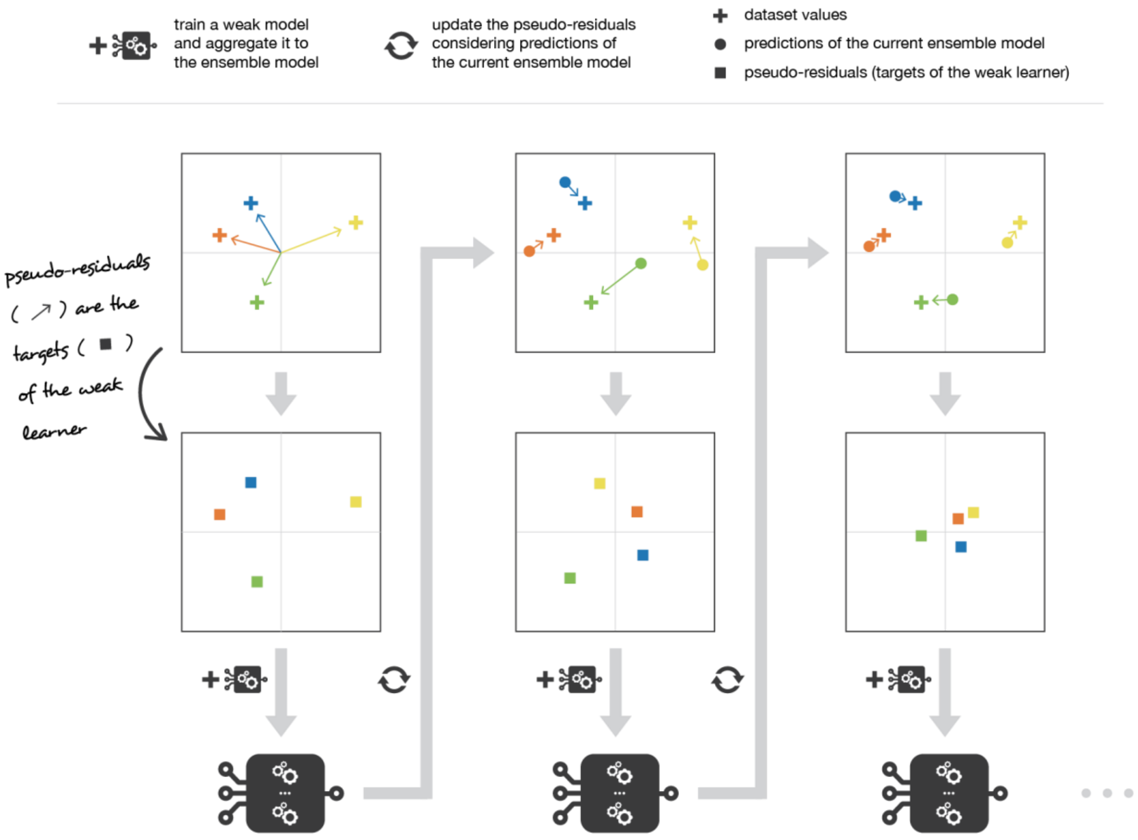

train a weak model and aggregate it to the ensemble model
update the pseudo-residuals considering predictions of the current ensemble model
dataset values
predictions of the current ensemble model
pseudo-residuals (targets of the weak learner)
pseudo-residuals (↗) are the targets (■) of the weak learner

Fig 3.21 Boosting in Ensemble learning

Gradient Tree Boosting is a generalized boosting used for loss functions. The difference in algorithms used in classification and regression is the loss function.

3.1.5.3 STACKING

Stacking is an ensemble learning technique that combines results of various classifiers and/or regressors or neural network, called base-models or level 0 models, and use the output as an input in another classifier or regressor algorithm, called meta-model or level 1 model.

Unlike bagging, the base models are all different in Stacking.

There must be at least two base models, that will be fit on the original trainingdata and then their predictions will be combined. Meta model has the job to find the most suitable way to combine the predictions of base models. Input for meta model is predictions by base models as well as the expected output so the meta model trains and fit on that as a dataset.

Stacking can also embed other boosting techniques, like you can use Random Forest or AdaBoost as base model or meta model.

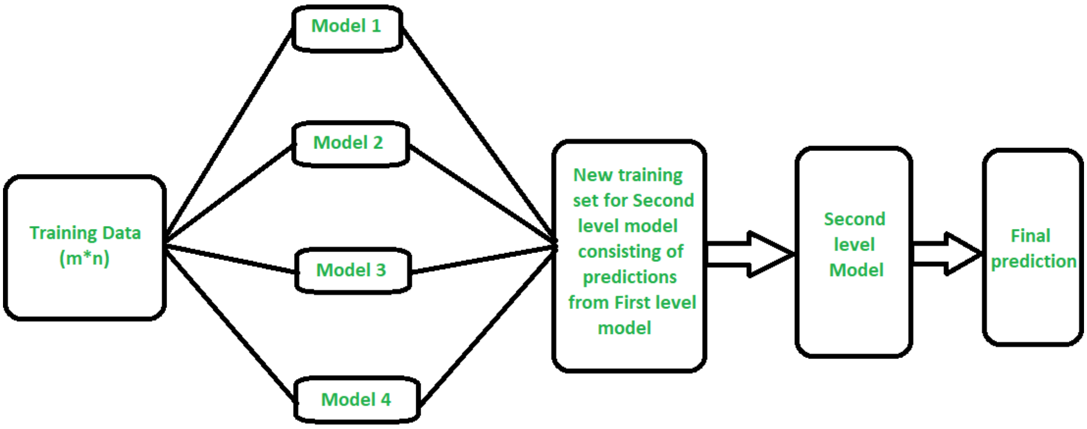

Model 1
Model 2
Training Data (m*n)
Model 3
Model 4
New training set for Second level model consisting of predictions from First level model
Second level Model
Final prediction

Fig 3.22 Stacking in Ensemble Learning

3.1.6 ENSEMBLE ALGORITHMS

3.1.6.1 RANDOM FOREST CLASSIFIER

Random forest algorithm is a part of the supervised learning technique and also one of the most popular algorithms. In machine learning, Random forest can be used for regression problems as well as classification problems. Random forest found on the idea of ensemble learning, bagging, which helps in improving the performance of the model with the help of combination of several decision trees to solve difficult problems.

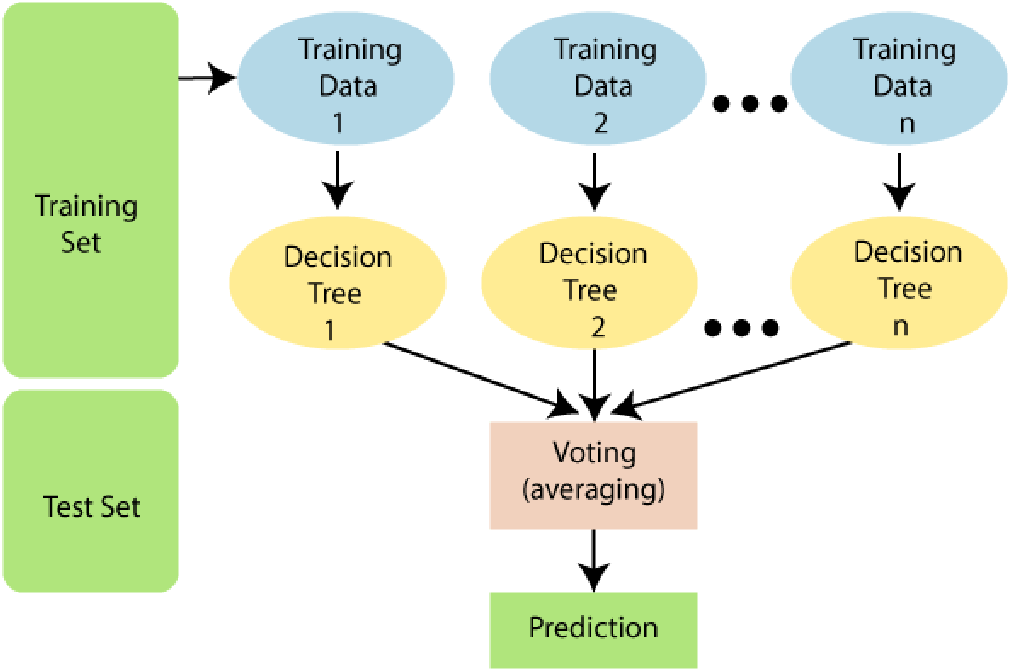


Fig 3.33 Random Forest Prediction model

Uses of Random Forest

- Less time consumption in training than other given algorithms.
- The output predicted accuracy is high and also for high range dataset it runs effectively.
- Accuracy is maintained in a condition of large data missing.

Working of Random Forest

- N number of datapoints are selected from the training set.
- Decision tree is constructed related to the selected points of data i.e., subsets.
- Choose any number of the decision tree you want to build.
- Repetition of first 2 step will continue.
- Prediction of each decision tree builds a new data point and it is assigned to the majority vote points category.

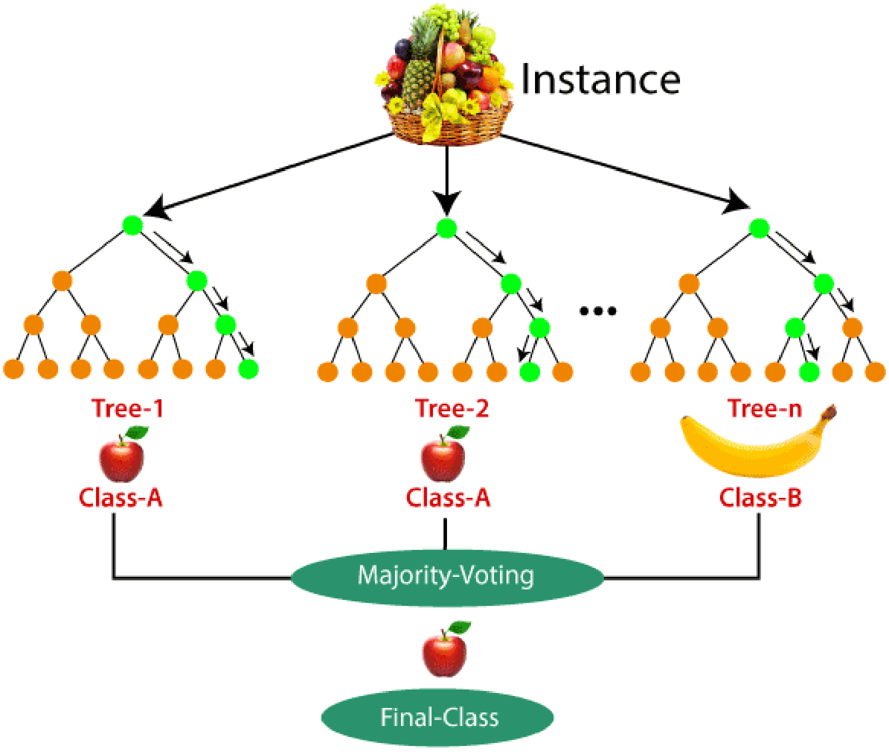


Fig 3.34 Working of random forest algorithm

Four main sectors of Random forest where is most used.

- Banking
- Marketing
- Land Use
- Medicine

Advantages

- Powerful
- Accurate
- Good performance on many problems including non-linear.

Disadvantages

- No interpretability
- Overfitting can easily occur
- We need to choose the number of trees every time.

3.1.6.2 ADABOOST CLASSIFIER

To increase and improve the ensemble techniques in machine learning referred as boosting. This technique endeavors to generate a strong classifier from a given number of weak classifiers.

With the help of this algorithm, we can prepare our data and also it helps us in performance boosting decision tree.

Boosting is a process in which we build our model from the dataset which further helps in prediction of models when there is large number of models. By applying we can do prediction of those models easily. With the help of this model, we can predict weak learners easily

AdaBoost is very successful model as well as modern algorithm which perfectly suitable for binary classes.

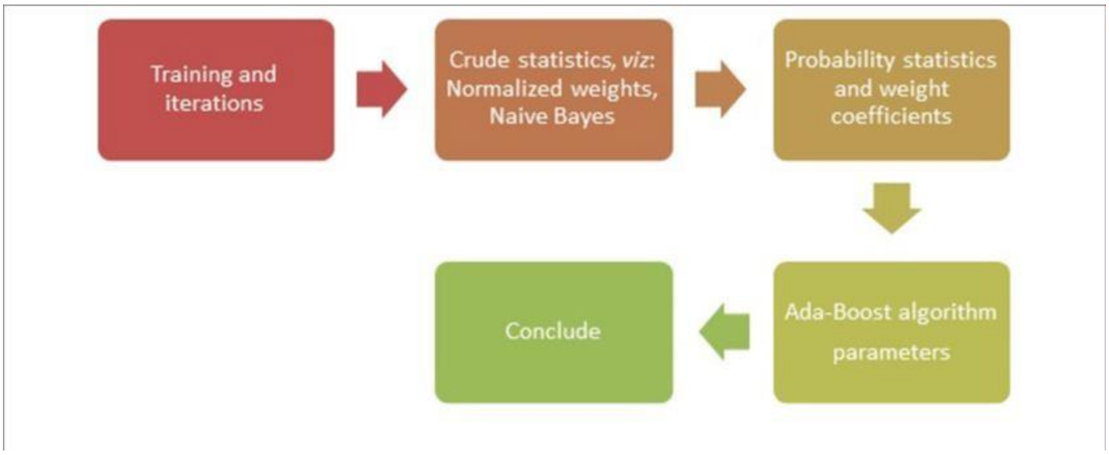

Training and iterations
Crude statistics, *viz*: Normalized weights, Naive Bayes
Probability statistics and weight coefficients
Ada-Boost algorithm parameters
Conclude

Fig 3.35 Flowchart of AdaBoost Algorithm

AdaBoost stands for the Adaptive Boosting and this is very popular and successful of all other algorithms in the present of all the techniques and also highly focuses on the combination of the large and different number of weak classifiers so that they should be able to build or generate one strong classifier.

A single classifier is not able to make predictions accurately of a class object, it’s better to group all the weak classifiers and also help each weak classifier progressively understanding and learning from the different wrong classified objects of others. By using this method, a strong model will be built.

Decision tress or logistic regression types of classifier can be used for your basics and to perform operations on the method.

The random guessing is not a better way to perform the methods. Weak classifiers are better than the guessing work. AdaBoost is applicable on the top of all the classifiers and this also helps in learning the shortcomings and also helps in proposing a more accurate model. Because of this is called as the “best out-of-the-box classifier”.

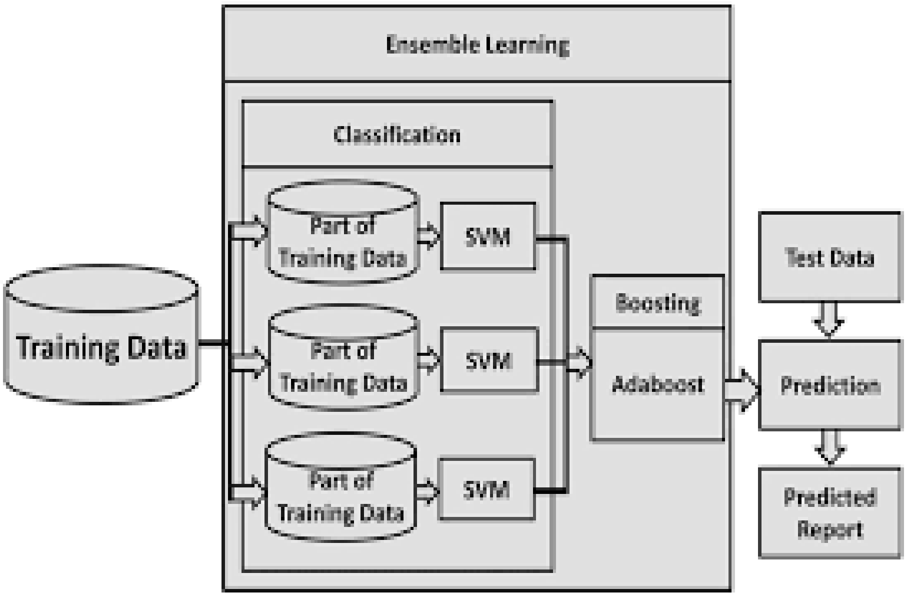


Fig 3.36 Forecasting Using Ensemble Learning

All the feeble models are included successive way, and furthermore prepared by utilizing the weighted preparing information.

There are no further enhancements made on the preparation dataset and the procedure proceeds until and except if the pre-set number of a client parameter have been made.

At last, after the completion of the process the hoard of weak learners is left with a stage value.

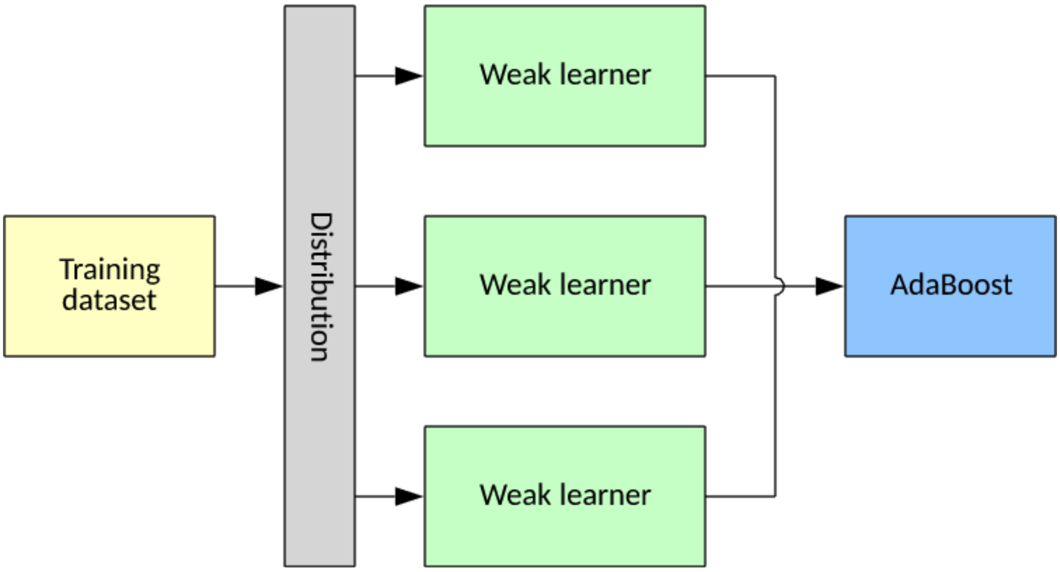


Fig 3.37 Ensemble AdaBoost

AdaBoost Principle

Fitting of weak learners in a sequential manner i.e. these models are moderately finer than random guessing (Can be considered as small decision trees) on frequently customize version of the data.

To generate the final prediction, the prediction from the whole of them are then merged through encumbered plurality vote. The data moderation at each so-called boosting iteration contains requesting weights w1, w2, …, w N, the beginning of the iterations starts from by generally training a weak learner on the original data. For each consecutive iteration, the representative weights are separately improved and the learning algorithm is again applied to the reweighted data.

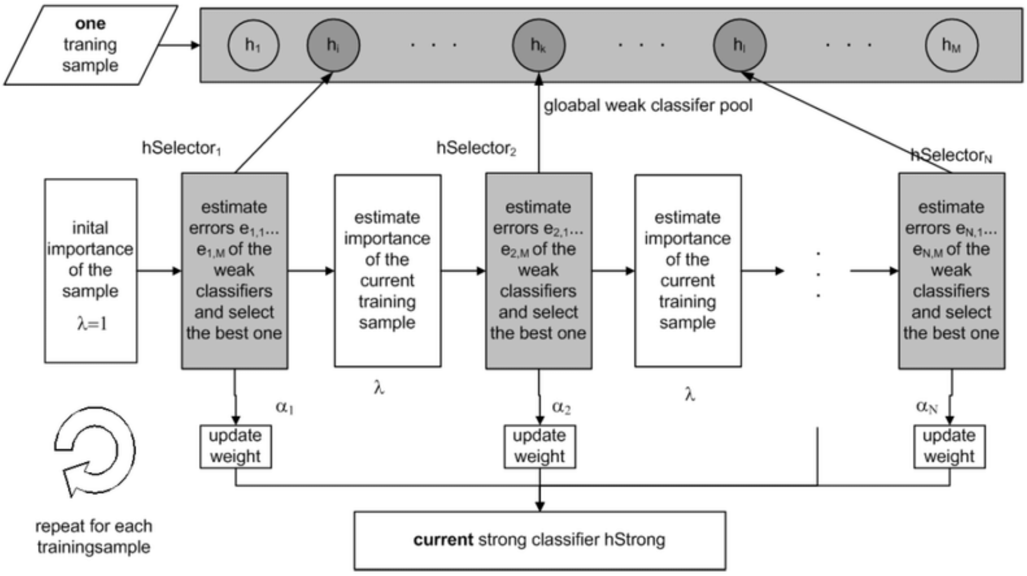

one traning sample
h_1
h_i
h_k
h_l
h_M
gloabal weak classifer pool
hSelector$_1$
hSelector$_2$
hSelector$_N$
inital importance of the sample λ=1
estimate errors $e_{1,1}$... $e_{1,M}$ of the weak classifiers and select the best one
estimate importance of the current training sample
λ
estimate errors $e_{2,1}$... $e_{2,M}$ of the weak classifiers and select the best one
estimate importance of the current training sample
λ
estimate errors $e_{N,1}$... $e_{N,M}$ of the weak classifiers and select the best one
α_1
α_2
α_N
update weight
update weight
update weight
repeat for each trainingsample
current strong classifier hStrong

Fig 3.38 Principles of AdaBoost for feature selection

Advantages

- Best utilization of the weak classifiers for cascading.
- Various algorithms for classification can be used as weak classifiers.
- High degree of precision is there in AdaBoost
- Weight of each classifier is fully considerable; this algorithm is relatable to Random Forest Algorithm and bagging algorithm.
- AdaBoost helps to enhance accuracy of weak classifiers and also makes it flexible for the use.
- AdaBoost also uses SVM algorithm so that it's become easier to use with less need for twisting parameters.
- AdaBoost is not vulnerable to misclassification through there is no tangible proof for this.
- It uses the use cases in the images and in text and also the classification of the images as well.
- The extension of the AdaBoost is done beyond classification.
- The learning process slows down due to stage wise estimation of the parameters.

- Basically, used in Facial Recognition systems so that it should be able to identify the face on the screen.
- This is the first algorithm of boosting the binary algorithm and also a successful one.

Disadvantages

- AdaBoost performs number of iterations and the set is poor numbers of weak classifiers and which can be purposeful using the cross-validation.
- Classification accuracy decreases when the data leads to imbalance.
- Training consumes a more time and time wasting, to perform better solutions its best to cut the point at each other for the reselection.
- AdaBoost algorithm is highly tactful to Noisy data and deviation so if the AdaBoost algorithm is used then it is extremely approved to abolish them.
- XG Boost is faster than the AdaBoost algorithm whereas it works slower than this.

AdaBoost Working

Step 1: The weighted samples are based on top of the training data which is defined as weak classifier. It is important to classify the weights of each sample data correctly and properly. Defines a decision stump and also gives equal weights of the sample.

Step 2: Each variable is created through the decision stump and how well it defines the classification of samples for their target or goal classes. There can be various sample which may be correct or incorrect.

Step 3: The samples which are classified incorrectly are assigned with the more weight in the next decision stump. The accuracy also helps in assigning the weights based on the classifiers i.e., represented as high accuracy = high weight.

Step 4: The iteration continues from the second step up to the whole data points to check whether they have been correctly classified, or also checks the iteration level at its maximum should have reached.

3.1.7 LINEAR REGRESSION

Prediction analysis of linear regression is performed on the basis of statistical method, it is also the popular and easiest algorithm in machine learning.

Parameters such as age, sales, product price, salary, etc., helps in the prediction of the continuous number values or variables.

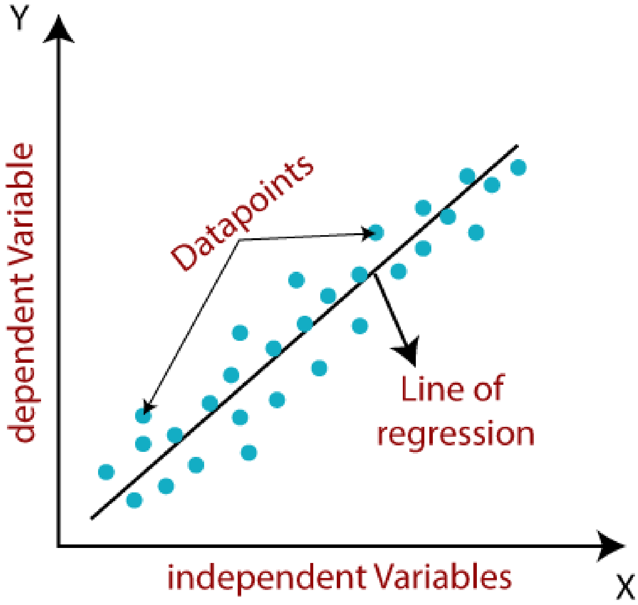


Fig 3.39 Representation of linear regression algorithm

Linear relationship is shown between the two variables i.e., an independent variable presented by X variable, dependent variable represented by Y variable. And the values change is done according to the value of X variable which stands for independent variable.

Types of Linear Regression

- Simple Linear Regression: Prediction of the numeric dependent variable value is done on the basis of single or one independent variable which is called as a simple linear regression.

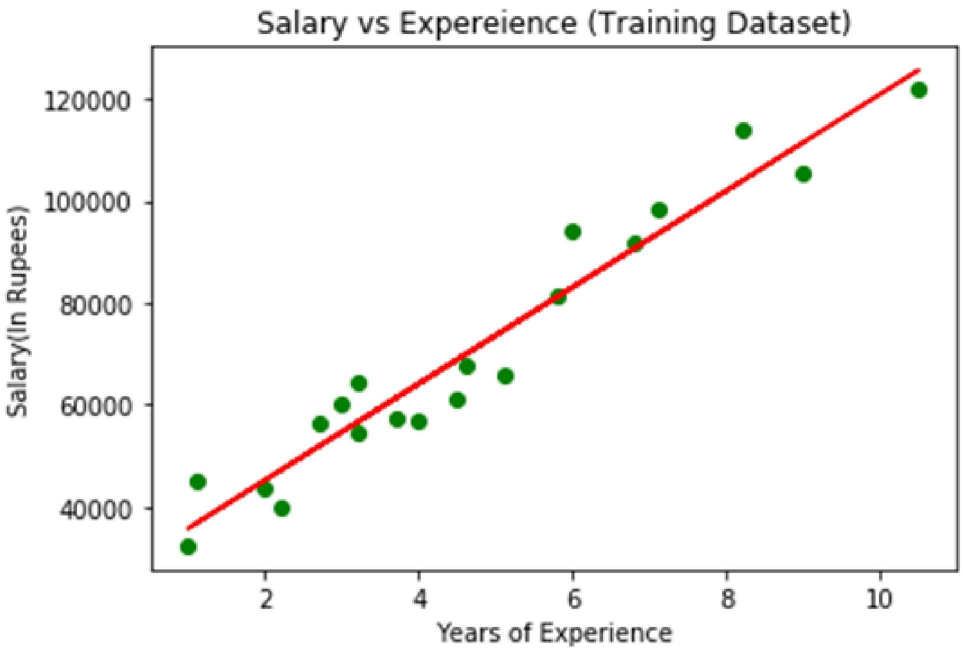


Fig 3.40 Prediction of simple linear regression

- Multiple Line Regression: Prediction of the numeric dependent variable value is done on the basis of more than two or three independent variable which is called as a multiple linear regression.
-

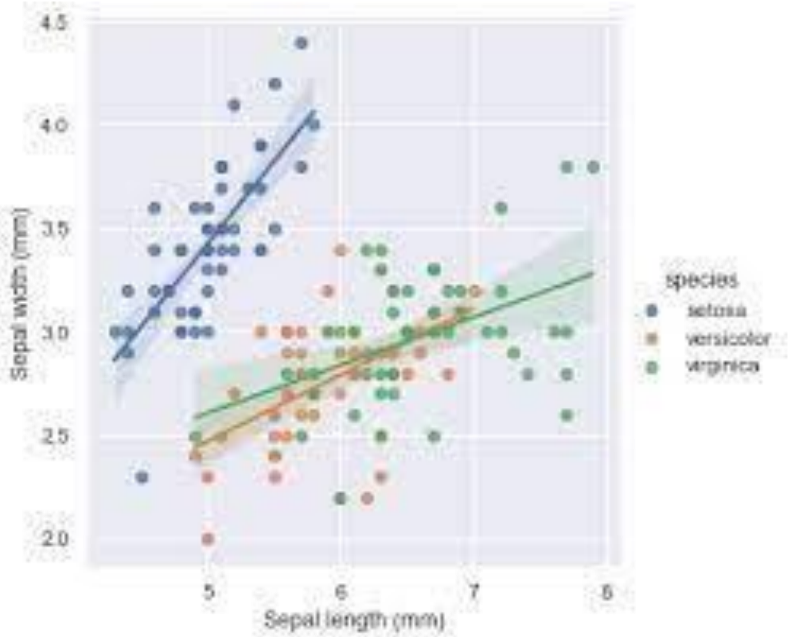


Fig 3.41 Prediction of multiple line regression

Advantages

- Work well irrespective of the dataset size.

- Gives information about the relevance of features.

Disadvantages

- The assumptions linear regression

3.1.7 NEURAL NETWORKS

Deep Learning, a subfield of machine learning in which the algorithms are inspired by the structure of the human brain, is built on the foundation of neural networks. Neural networks take in data, train themselves to recognize patterns in it, and then predict the output for a new batch of data that is comparable.

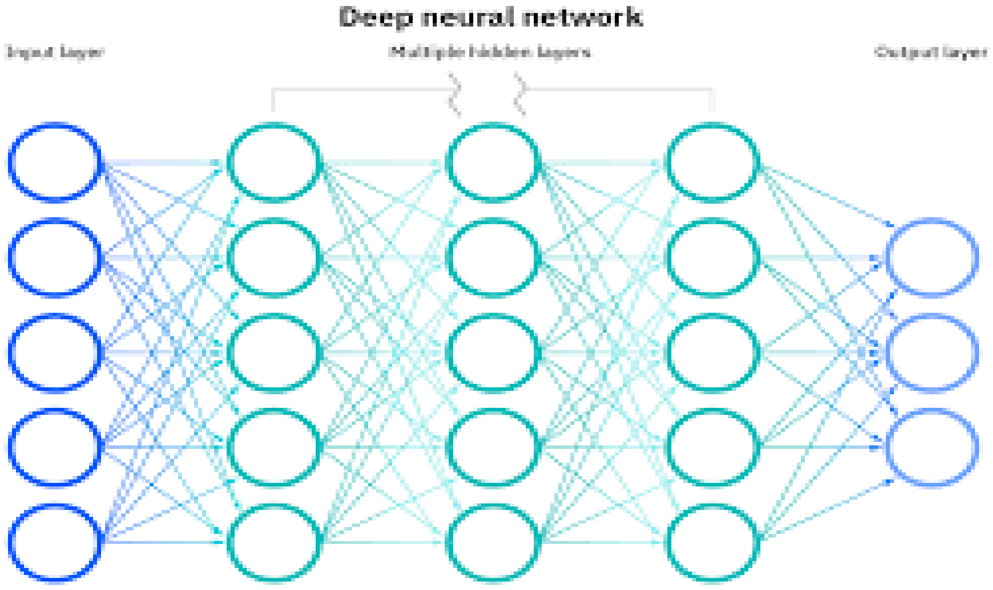


Fig 3.42 Representation of a Deep Neural Network

3.1.7.1 NEURONS

Layers of neurons make up neural networks. The network's core processing units are these neurons. The input layer is the first layer, and it receives the data. Our final output is predicted by the output layer. Between the visible and hidden layers are the hidden layers, which do the majority of the computations necessary by our network. Through channels, neurons in one layer communicate with neurons in the next layer. Weight is a numerical value applied to each of these channels. The inputs are multiplied by the weights, and the sum is given to the neurons in the hidden layer as input.

$$a_n^L = \left[\sigma\left(\sum_m \theta_{nm}^L \left[\cdots\left[\sigma\left(\sum_j \theta_{kj}^2 \left[\sigma\left(\sum_i \theta_{ji}^1 x_i + b_j^1\right)\right] + b_k^2\right)\right]\cdots\right]_m + b_n^L\right)\right]_n$$

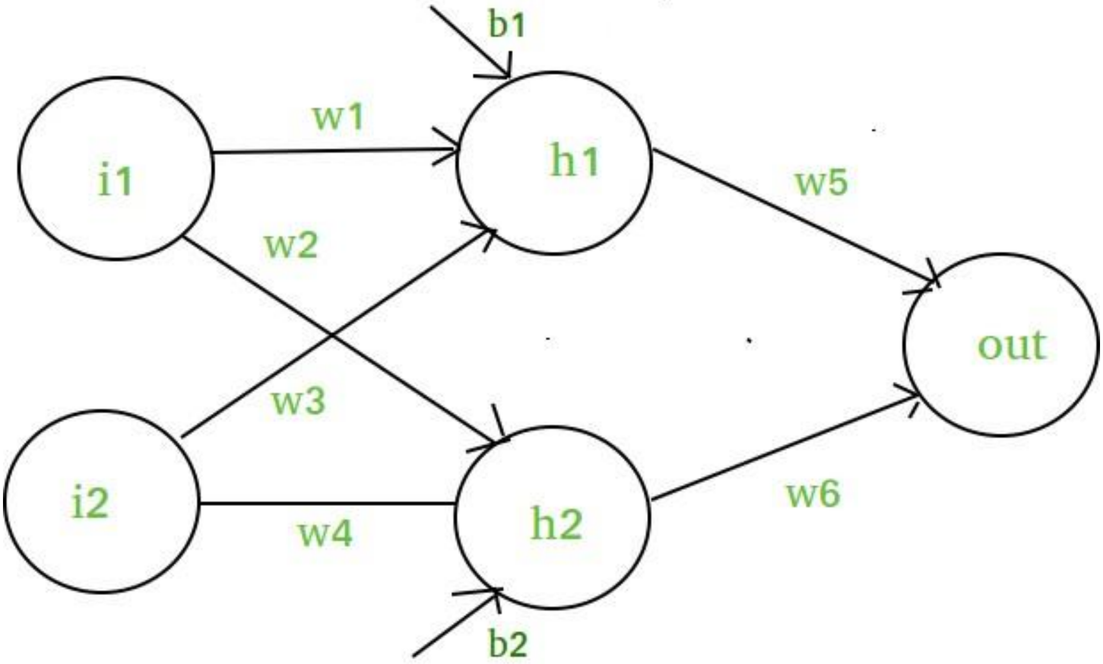


Fig 3.43 Weights and biases added to Activation Functions in hidden layer

Each of these neurons has a numerical value known as bias, which is then added to the input sum. This value is then passed through the activation function, which is a threshold function. The output of the activation function decides whether or not a certain neuron is stimulated. Through the channels, an active neuron communicates info to the neurons of the following layer. Data is disseminated over the network in this manner. Forward propagation is the term for this.

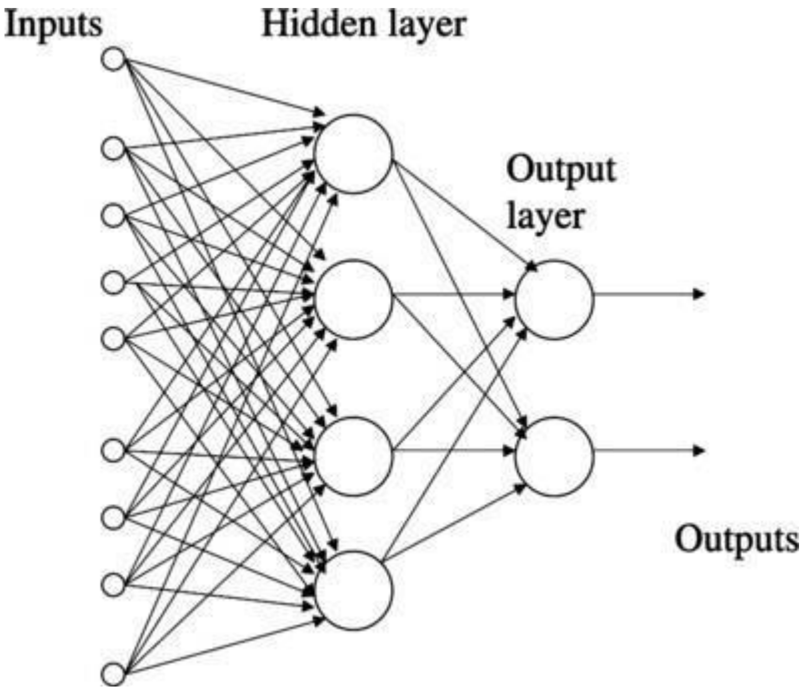


Fig 3.44 Forward Propagation in Neural Network

The neuron with the greatest value fires in the output layer and decides the output. The numbers are essentially probabilities. During the training phase, the output is also sent to the network, which checks the projected outputs and continues to train until it achieves the desired precision and accuracy. The amount of the error indicates how far we've gone wrong, while the sign indicates whether the results are greater or lower than predicted. This data is then sent backwards across our network. Backward propagation is the term for this.

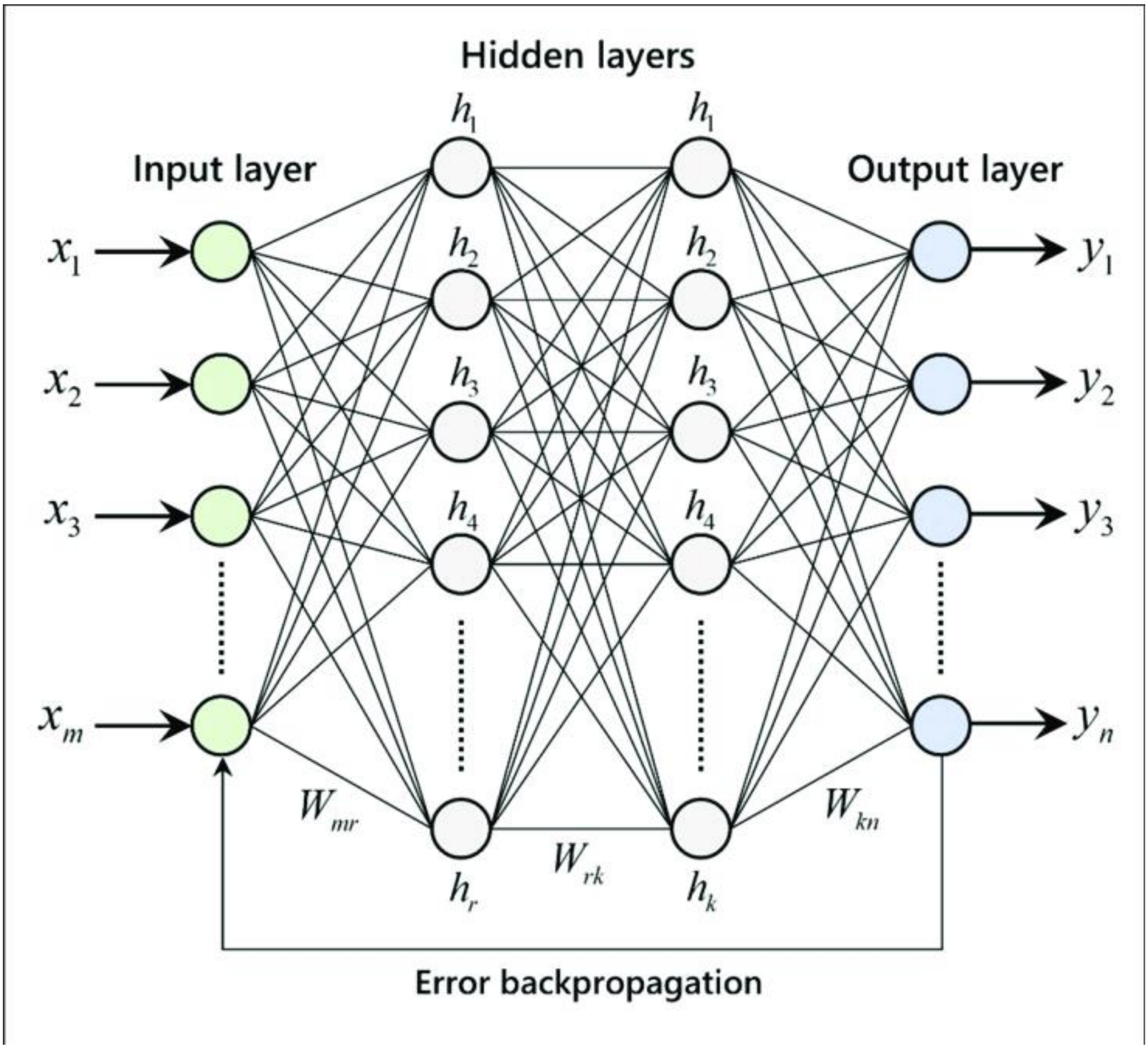


Fig 3.45 Backward Propagation in Neural Network

The weights are now modified based on this information. With several inputs, this cycle of forward and backward propagation is repeated recursively. This process is repeated until the weights are given in such a way that the network can properly anticipate the output in the majority of circumstances. Our training program has come to a conclusion

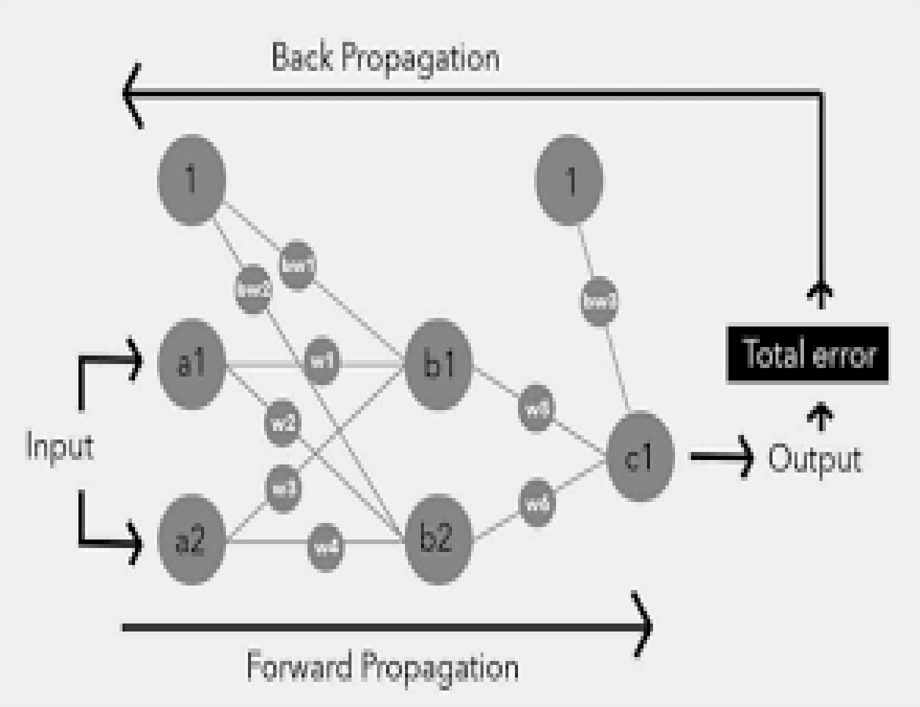


Fig 3.46 The cycle of forward and backward propagation

3.1.7.2 PERCEPTRON

A perceptron, basically is a single layer neural network. was a hardware initially , developed by Frank Rosenblatt in 1957 at the Laboratory of Cornell Aeronautics. The device’s inspiration was human brain.

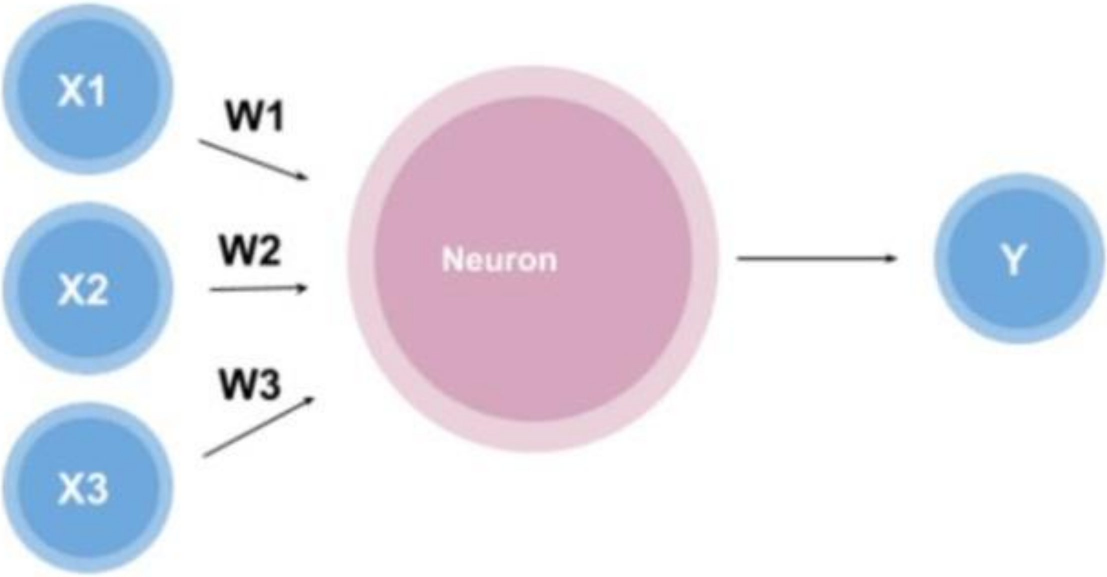


Fig 3.47 A perceptron

3.1.7.3 MULTILAYER PERCEPTRON

A multi-layered perceptron is simply a perceptron with multiple layers and therefore multiple outputs at initial layers. The working is same as a neural network.

To apply a multilayer perceptron, we have a classifier called MlpClassifier, in python.

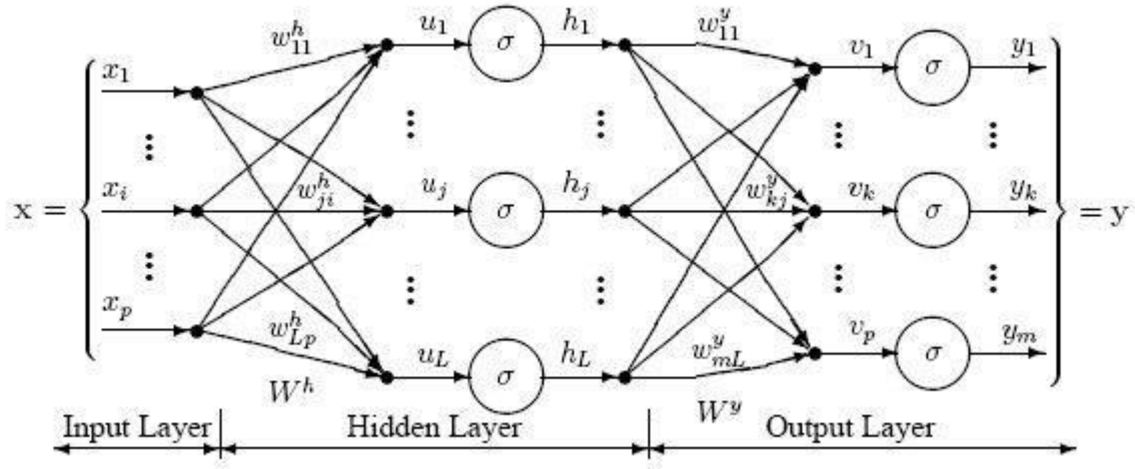


Fig 3.48 A multi-layered Perceptron

Working

- Like in a neuron or a perceptron, input is fed to the input layer as a dot product of original input points and their respected weights.
- The sum goes through various activation functions in the hidden layer.
- Activation function decides whether a particular neuron will be passed ahead or not.
- Repeat above 2 steps until output layer is reached.
- After reaching output layer, outputs are cross checked from expected outputs.
- If the estimation of most of the neurons is correct that final output is predicted otherwise back propagation is used to send them back and train again.
- Above steps repeats until desired accuracy is received.

3.1.8 PYTHON

Python is general purpose language and also a high-level language.

It can be used for

- Console app
- Desktop application
- Web app
- Mobile app

- Internet of things
- Machine learning

Very simple and straight forward syntax. Python is case sensitive. It is based on OOPS concepts. The code is typed dynamically and in specified format.

No use of braces instead of this indentation is used. No need to declare variables before using. No compiler is used instead interpreter is used.

Features of Python

- Python codes are reliable
- Python manages memory automatically by itself
- Python has different and large number of libraries
- It is both OOP based and procedural
- Its interpreter is interactive which used for checking python commands.
- Dependency on platform is not required.

6. Importance of Python in Machine Learning

- Python is the most preferred language for machine learning.
- The programs in machine learning are not performed implicitly of a simply making computer task.
- Python has very clear syntax and easy to understand.
- Python can be written with other languages and also it can be used as an extension for other app.
- It is easy to use for scripting as well for automation.
- Python is widely used and considerable for the machine learning.
- The python language is bit slower than the other languages but is very good at

handling the capacity of the data.

- The python language is more capable for the interaction with all the kind of third-party platforms and languages.

- The python language is also popular in the analytics domain and also an open source language.

3.1.8.1 PYTHON AND MACHINE LEARNING

Machine learning is a part of computer science it is widely being used.

ML provided better understanding of data. It is also considered as type of artificial intelligence and the raw data patterns are being extracted by using the methods and the other algorithms. The main key is to focus that how to allow computer systems to help in learning the new experiences without the human intervention and with implicitly programming.

3.1.8.2 ADVANTAGES OF ML WITH PYTHON

Doing machine learning plays important role regarding the making of the decisions and the decisions based on the data with great efficiency and better margin scale.

Deep leaning and the machine learning are used to the main key point knowledge and the information from the given data to operate various real-world task and solve problems. Machines helps in taking data-driven decisions and the start the process automatically.

Data driven decision can be use directly instead of applying programming logic because some program cannot be made naturally.

Machine learning came into use because it helps to solve and understand the real-life situation easily and with great efficiency.

ML Model with Python

The Programs in PC are said to the take in assignments from the experience E as for certain undertakings T of various classes and the exhibition is estimated by P. The improvement of the experience E with the help of the performance at tasks and the P is as measured by it.

Basically, the main three parameters are being focused and the main components of any learning algorithm, which consists of the name i.e. (E) for experience, (T) for task and (P) for performance.

Machine Learning is a field of learning algorithms of the Artificial Intelligence

- Helps in improving their performance (P)

- Helps in executing the some or the different task (T)

- With the time increases the experience also increases (E)

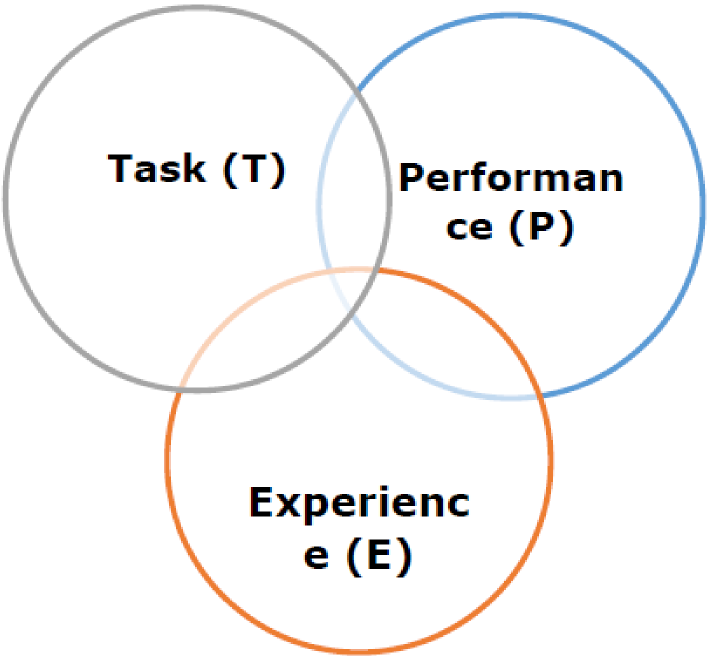


Fig 3.49 Machine learning model using python

Challenges in Machine Learning Using Python

- Data Quality – The biggest challenge in machine learning is to have a good data quality. When the low quality of data is used it leads to many problems which are related to the processing of the data and the extraction of the features.

- Time-demolishing task – A lot of time consumption is done because of the acquisition of the data and the retrieval and the extraction of the features.

- Less experience people – As this technology is in its early age so there are less people that have specialization.

- Formulating Business problems with no clear objectives – When there is a well-defined goal for business but when the objective is not clear then it's a key challenging for machine learning and also that the more maturity is required in the machine learning.

- Issues of overfitting and underfitting - The model is underfitted and overfitted does not helps in performing and presenting the problem very well manner.

- Curse of dimensionality – The datapoints with too many features can be a real barrier or obstacle in the machine learning model with the python.
- Deployment difficulty – Sometimes data may be too complex that it may be difficult for models of machine learning.

Various applications of Machine Learning using Python

- Fraud Detection
- Error detection and prevention
- Fraud Prevention
- Weather forecasting and prediction
- Recommendation of products to customer in online shopping
- Speech synthesis
- Sentiment analysis
- Speech Recognition
- Customer Segmentation
- Emotion’s analysis
- Object Recognition
- Stock market analysis and prediction

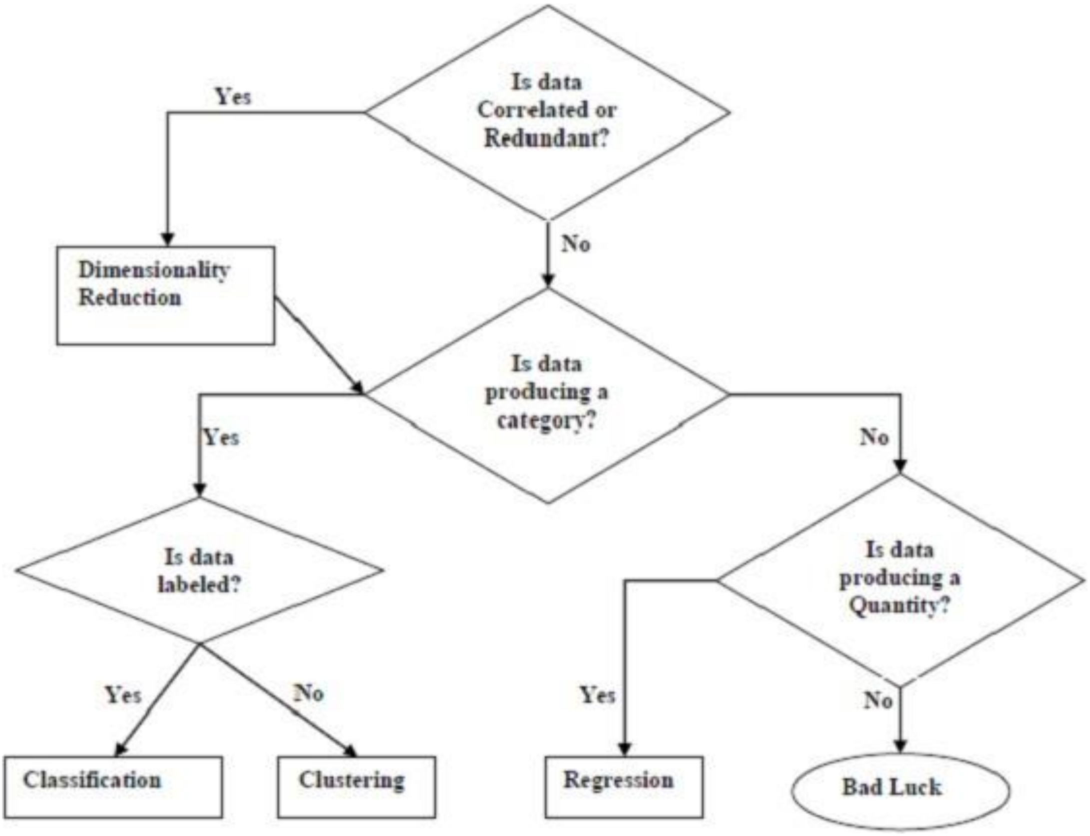


Fig 3.50 Various tasks is suitable for Machine leaning problems.

Different types of methods used in Machine Leaning Using Python

- Semi-Supervised Learning
- Based on human supervision
- Reinforcement Learning
- Unsupervised learning

Chapter 4

DESIGN APPROACHES AND USE CASE DIAGRAM

4.1 ACTIVITY DIAGRAM

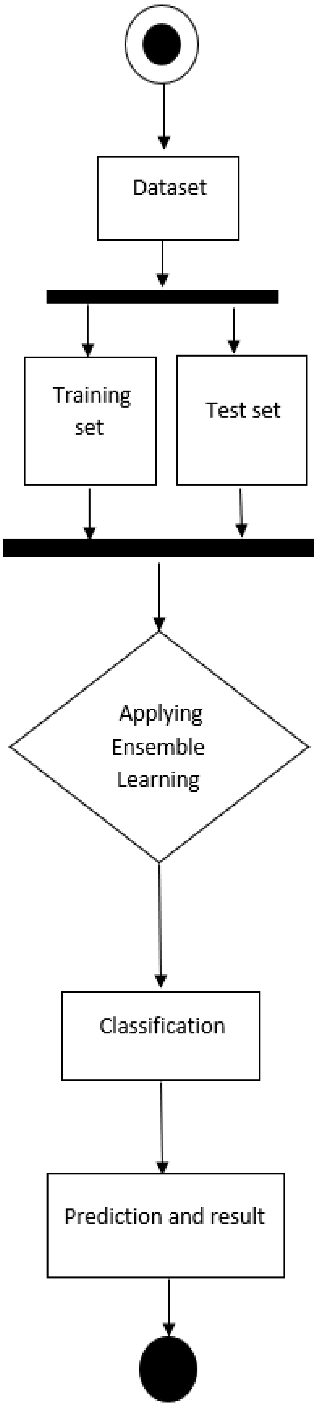


Fig 4.1 Activity Diagram

4.2 ARCHITECTURE DIAGRAM

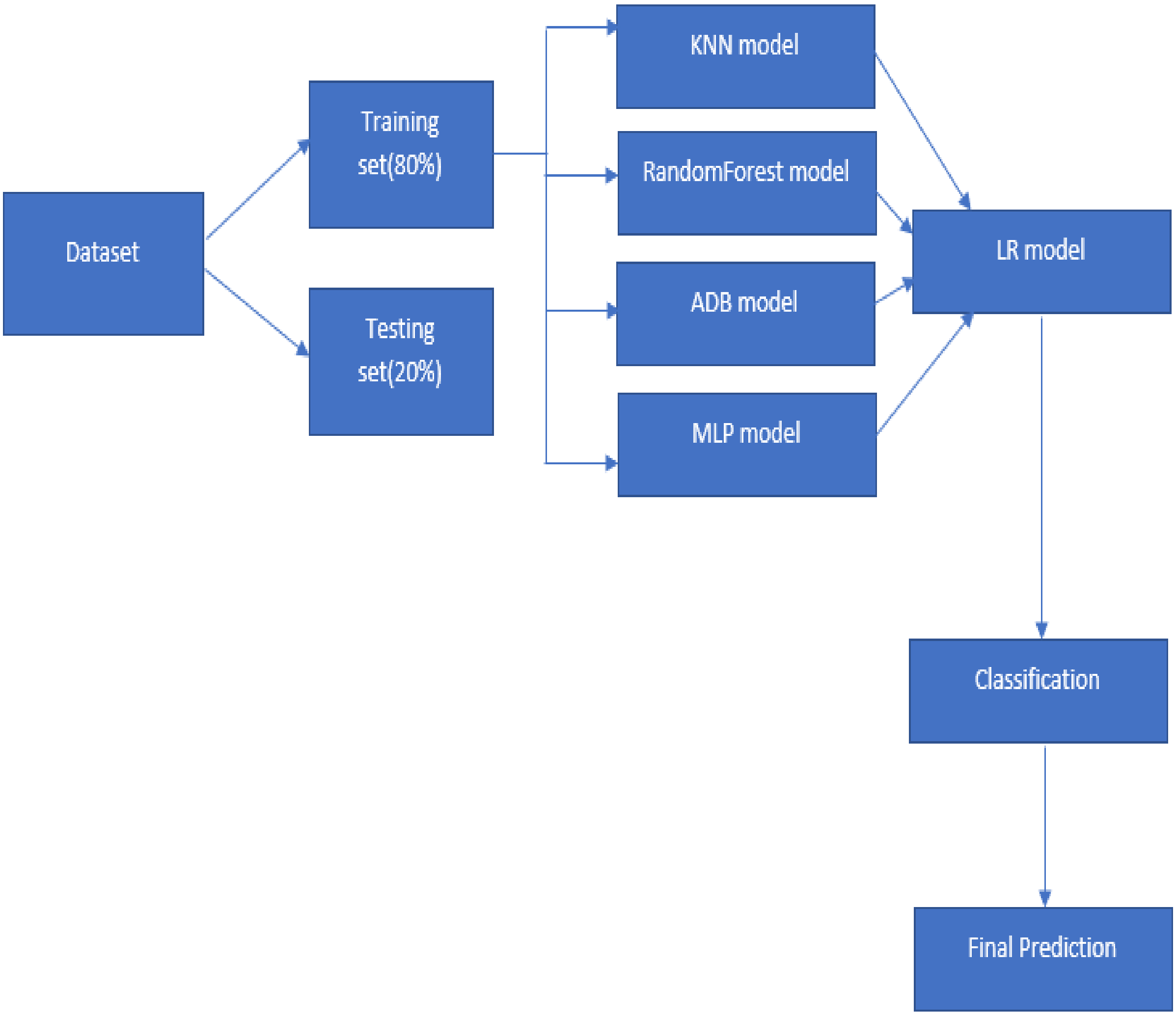


Fig 4.2 Architecture Diagram

4.3 DATA FLOW DIAGRAM

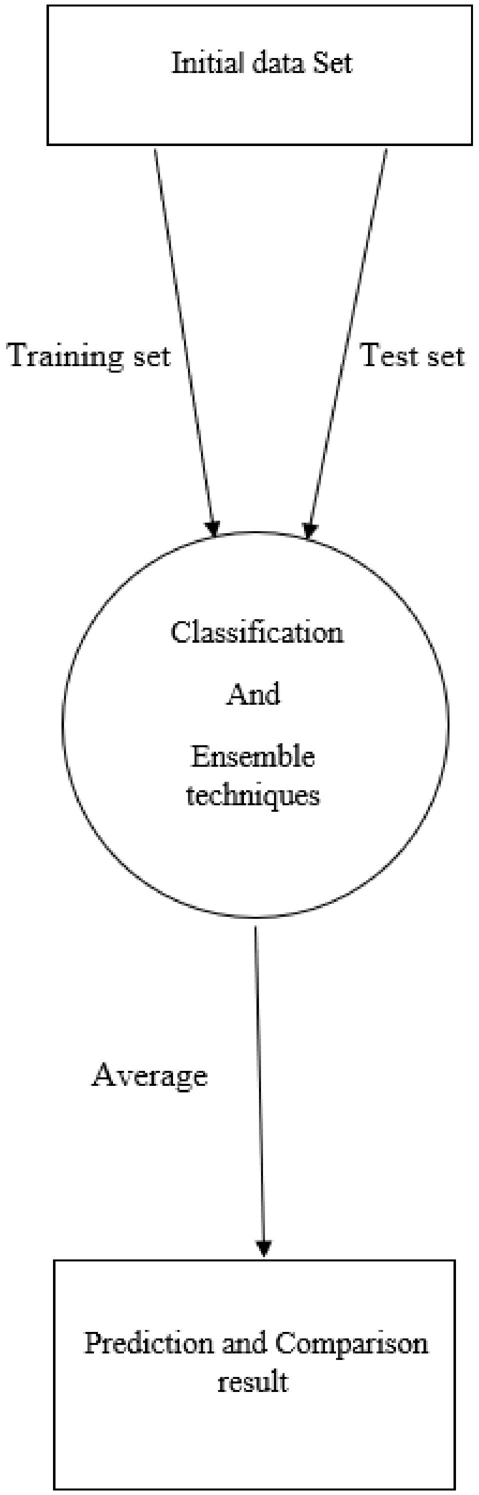


Fig 4.4 Data Flow Diagram (DFD)

Chapter 5

MODULE DISCRIPTION

5.1 UNDERSTANDING DATASET

Two datasets are taken for Autism. In this module, understanding things like:

- How attributes are related to each other.
- Their dependency on one another.
- Whether we need a particular attribute

Autism(https://www.kaggle.com/zohebabai/predicting-early-asd-traits-of-toddlers/data?select=Toddler+Autism+dataset+July+2018.csv)

- in toddlers- This dataset is used to gather information about toddlers in order to further predict the outcome.
- In adults- this dataset was used as a reference to understand ASD symptoms.

5.2 PROCESSING AND MANIPULATING DATA

Datasets are vast and we don’t need all that’s in there, so we process and manipulate it according to our need. For example, by removing columns, adding new columns, filling null values, etc.

5.3 SPLITTING THE DATASET

Data split into two sets, i.e., training set and testing set in ratio of 80% and 20% respectively. Using train_test_split() function of sci-kit learn.

5.4 CLASSIFICATION OF DATA

In this module, classification is of data will be done using 5 Algorithms:

- K-nearest neighbor
- Support Vector Machine
- Random Forest
- Multi-layer perceptron
- AdaBoost Classifier

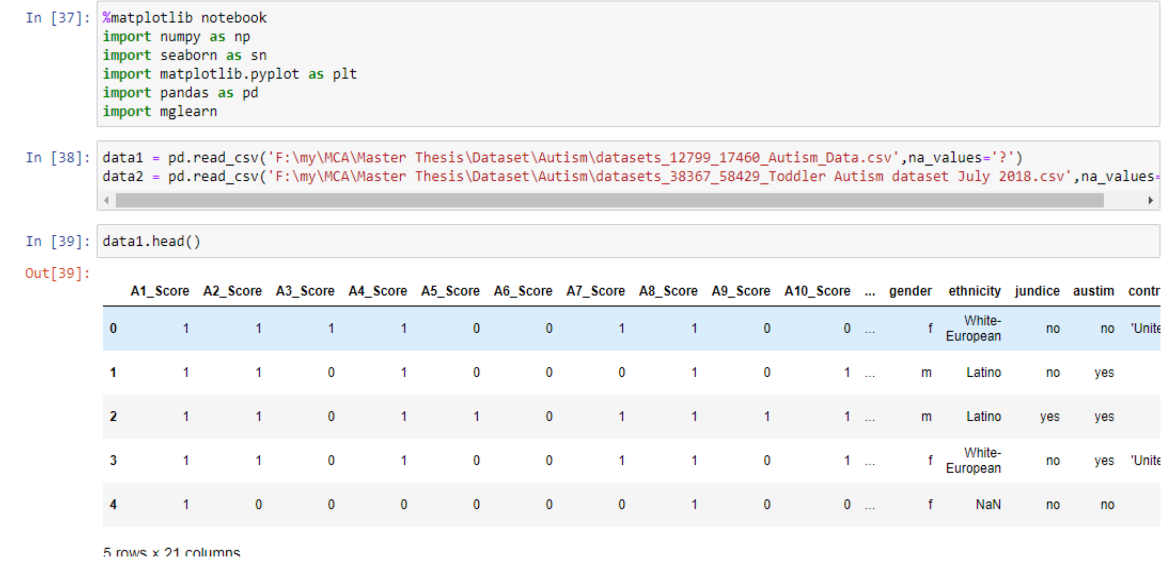
And testing the accuracy of prediction of Autism using testing data.

5.5 APPLYING ENSEMBLE LEARNING

Applying Stacking (type of ensemble learning), using Stacking classifier from mlxtend and sklearn and using Linear regression as meta-Model.

Chapter 6

IMPLEMENTATION AND RESULTS



```
In [37]: %matplotlib notebook
         import numpy as np
         import seaborn as sn
         import matplotlib.pyplot as plt
         import pandas as pd
         import mglearn
```

```
In [38]: data1 = pd.read_csv('F:\my\MCA\Master Thesis\Dataset\Autism\datasets_12799_17460_Autism_Data.csv',na_values='?')
         data2 = pd.read_csv('F:\my\MCA\Master Thesis\Dataset\Autism\datasets_38367_58429_Toddler Autism dataset July 2018.csv',na_values=
```

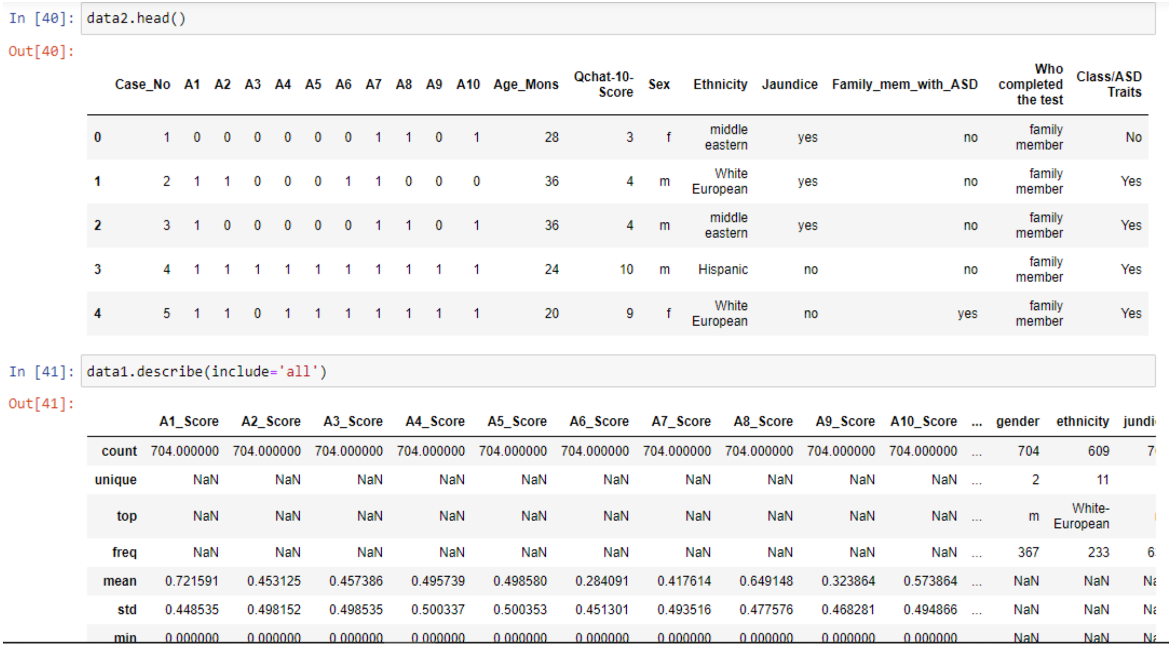
```
In [39]: data1.head()
```

Out[39]:

	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	A10_Score	...	gender	ethnicity	jundice	austim	contr
0	1	1	1	1	0	0	1	1	0	0	...	f	White-European	no	no	'Unite
1	1	1	0	1	0	0	0	1	0	1	...	m	Latino	no	yes	
2	1	1	0	1	1	0	1	1	1	1	...	m	Latino	yes	yes	
3	1	1	0	1	0	0	1	1	0	1	...	f	White-European	no	yes	'Unite
4	1	0	0	0	0	0	0	1	0	0	...	f	NaN	no	no	

5 rows x 21 columns

Fig 6.1 First five values of Dataset1



```
In [40]: data2.head()
```

Out[40]:

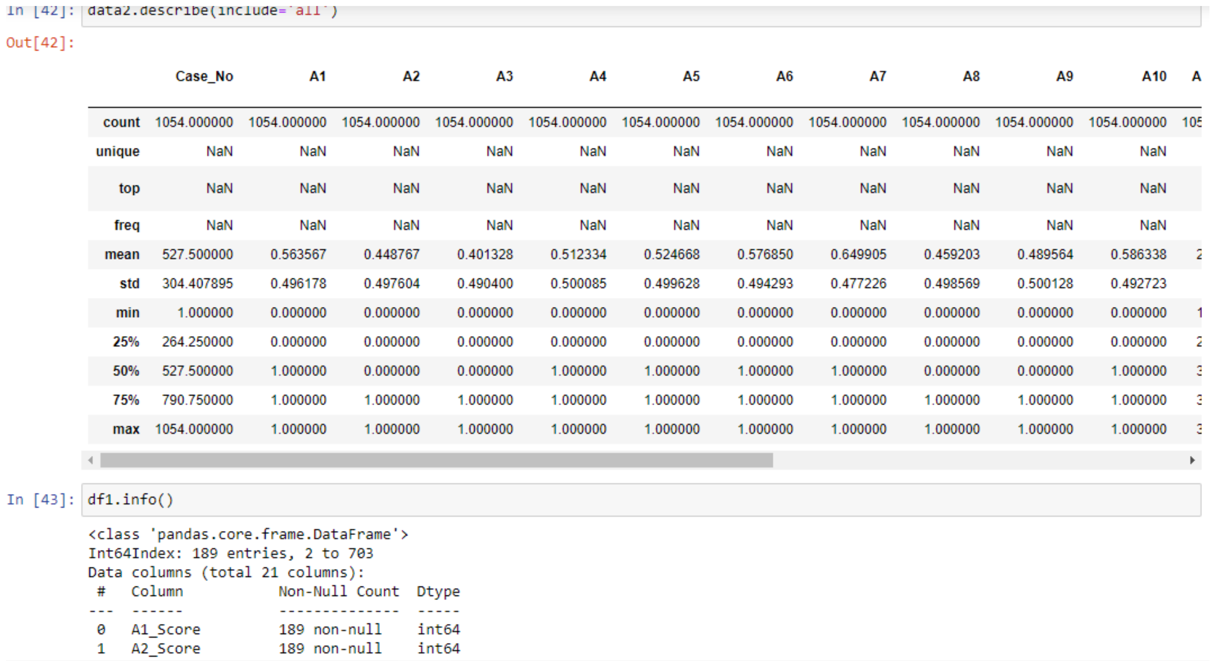
	Case_No	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	Age_Mons	Qchat-10-Score	Sex	Ethnicity	Jaundice	Family_mem_with_ASD	Who completed the test	Class/ASD Traits
0	1	0	0	0	0	0	0	1	1	0	1	28	3	f	middle eastern	yes	no	family member	No
1	2	1	1	0	0	0	1	1	0	0	0	36	4	m	White European	yes	no	family member	Yes
2	3	1	0	0	0	0	0	1	1	0	1	36	4	m	middle eastern	yes	no	family member	Yes
3	4	1	1	1	1	1	1	1	1	1	1	24	10	m	Hispanic	no	no	family member	Yes
4	5	1	1	0	1	1	1	1	1	1	1	20	9	f	White European	no	yes	family member	Yes

```
In [41]: data1.describe(include='all')
```

Out[41]:

	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	A10_Score	...	gender	ethnicity	jundi
count	704.000000	704.000000	704.000000	704.000000	704.000000	704.000000	704.000000	704.000000	704.000000	704.000000	...	704	609	7
unique	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	2	11	
top	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	m	White-European	
freq	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	367	233	6
mean	0.721591	0.453125	0.457386	0.495739	0.498580	0.284091	0.417614	0.649148	0.323864	0.573864	...	NaN	NaN	Na
std	0.448535	0.498152	0.498535	0.500337	0.500353	0.451301	0.493516	0.477576	0.468281	0.494866	...	NaN	NaN	Na
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000		NaN	NaN	Na

Fig 6.2 First five values of Dataset2

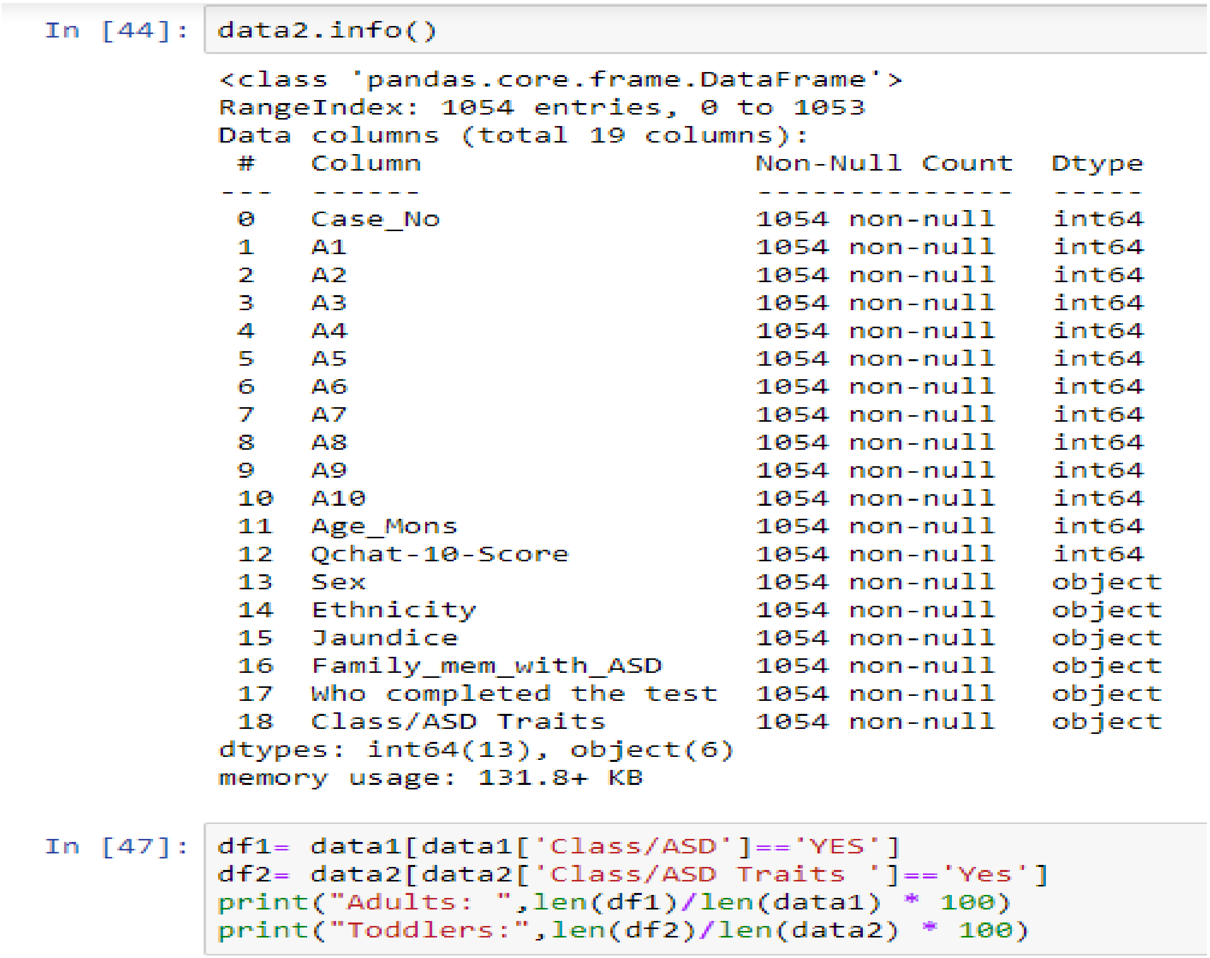


```
In [42]: data2.describe(include='all')
Out[42]:
```

	Case_No	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A
count	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	1054.000000	105
unique	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
top	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
freq	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
mean	527.500000	0.563567	0.448767	0.401328	0.512334	0.524668	0.576850	0.649905	0.459203	0.489564	0.586338	2
std	304.407895	0.496178	0.497604	0.490400	0.500085	0.499628	0.494293	0.477226	0.498569	0.500128	0.492723	
min	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	1
25%	264.250000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	2
50%	527.500000	1.000000	0.000000	0.000000	1.000000	1.000000	1.000000	1.000000	0.000000	0.000000	1.000000	3
75%	790.750000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	3
max	1054.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	3

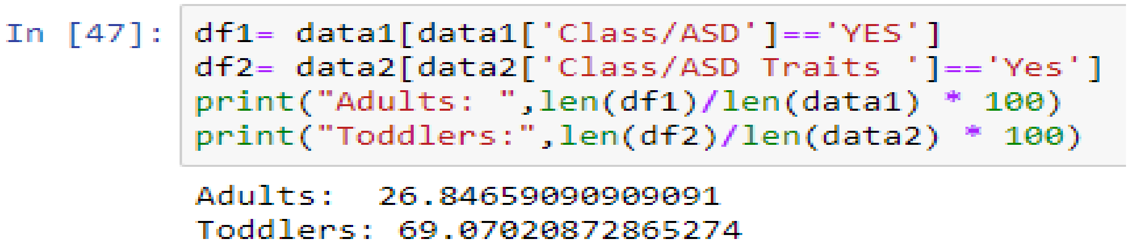
```
In [43]: df1.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 189 entries, 2 to 703
Data columns (total 21 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   A1_Score         189 non-null    int64
 1   A2_Score         189 non-null    int64
```

Fig 6.3 Description of Dataset1



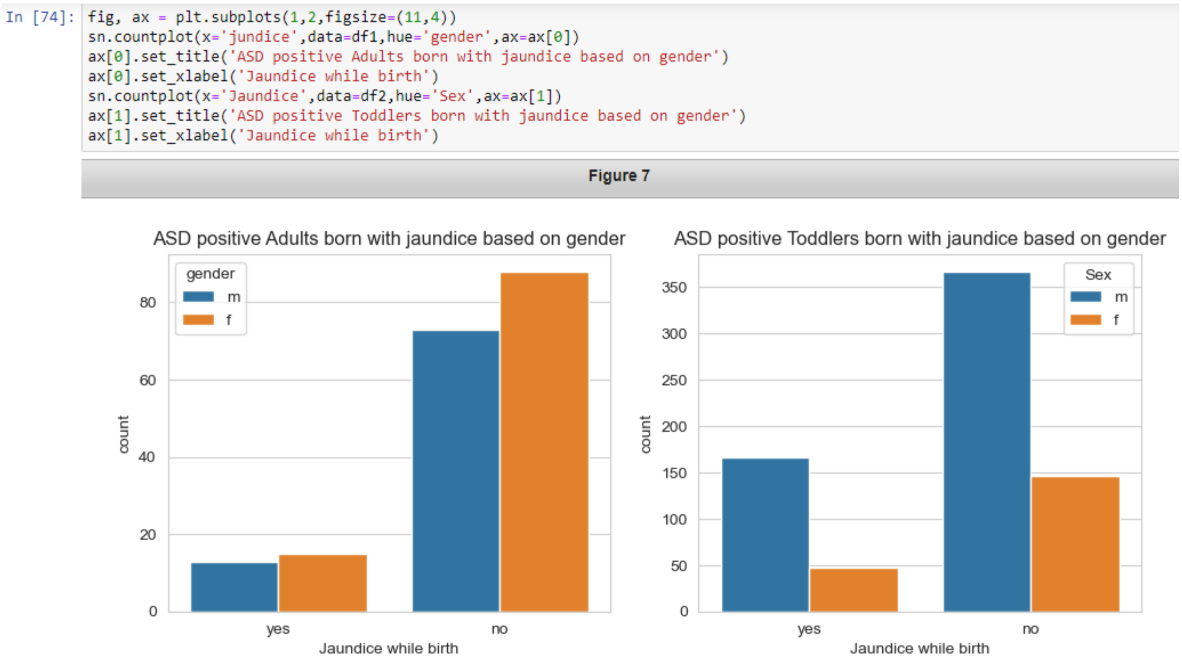
```
In [44]: data2.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1054 entries, 0 to 1053
Data columns (total 19 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Case_No                 1054 non-null   int64
 1   A1                      1054 non-null   int64
 2   A2                      1054 non-null   int64
 3   A3                      1054 non-null   int64
 4   A4                      1054 non-null   int64
 5   A5                      1054 non-null   int64
 6   A6                      1054 non-null   int64
 7   A7                      1054 non-null   int64
 8   A8                      1054 non-null   int64
 9   A9                      1054 non-null   int64
 10  A10                     1054 non-null   int64
 11  Age_Mons                1054 non-null   int64
 12  Qchat-10-Score          1054 non-null   int64
 13  Sex                     1054 non-null   object
 14  Ethnicity               1054 non-null   object
 15  Jaundice                1054 non-null   object
 16  Family_mem_with_ASD     1054 non-null   object
 17  Who completed the test  1054 non-null   object
 18  Class/ASD Traits        1054 non-null   object
dtypes: int64(13), object(6)
memory usage: 131.8+ KB

In [47]: df1= data1[data1['Class/ASD']=='YES']
         df2= data2[data2['Class/ASD Traits ']=='Yes']
         print("Adults: ",len(df1)/len(data1) * 100)
         print("Toddlers:",len(df2)/len(data2) * 100)
```

Fig 6.4 Description of dataset2



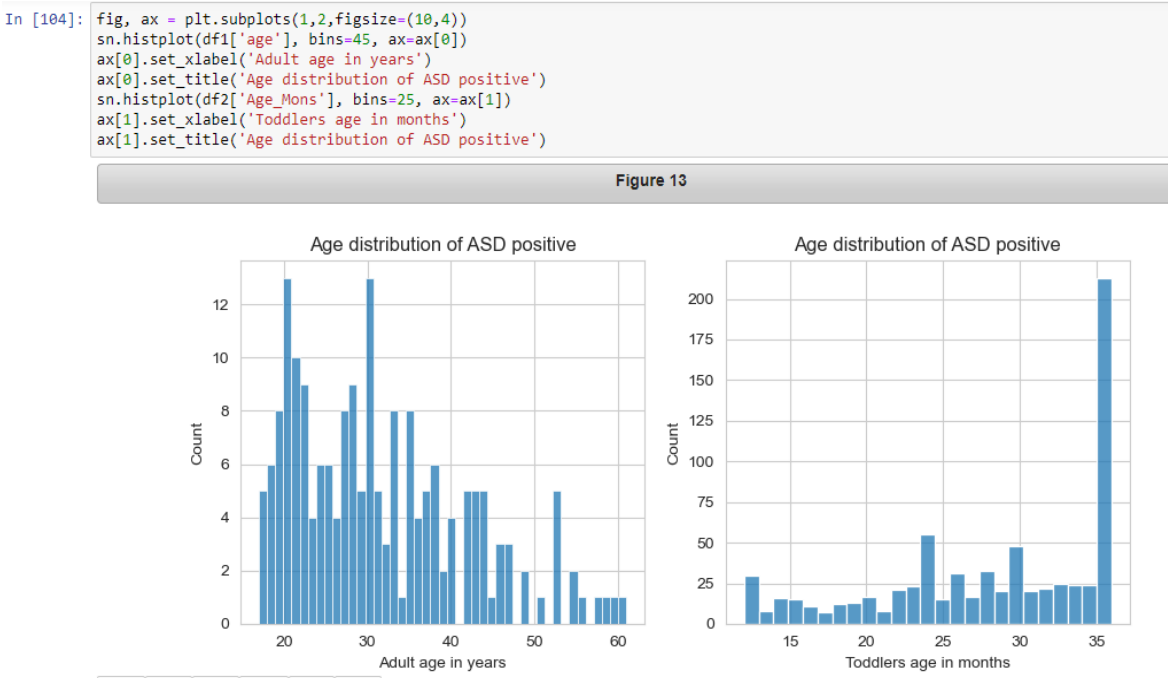
```
In [47]: df1= data1[data1['Class/ASD']=='YES']
         df2= data2[data2['Class/ASD Traits ']=='Yes']
         print("Adults: ",len(df1)/len(data1) * 100)
         print("Toddlers:",len(df2)/len(data2) * 100)

         Adults:  26.84659090909091
         Toddlers: 69.07020872865274
```

Fig 6.5 Presence of Autism in Adults and Toddlers



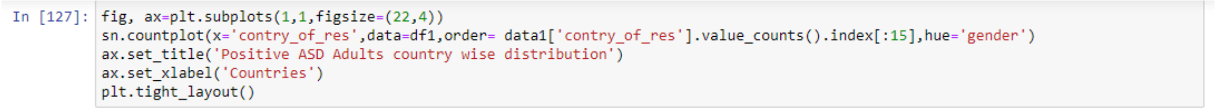
```
In [74]: fig, ax = plt.subplots(1,2,figsize=(11,4))
         sn.countplot(x='jundice',data=df1,hue='gender',ax=ax[0])
         ax[0].set_title('ASD positive Adults born with jaundice based on gender')
         ax[0].set_xlabel('Jaundice while birth')
         sn.countplot(x='Jaundice',data=df2,hue='Sex',ax=ax[1])
         ax[1].set_title('ASD positive Toddlers born with jaundice based on gender')
         ax[1].set_xlabel('Jaundice while birth')
```

Fig 6.6 Impact of Gender



```
In [104]: fig, ax = plt.subplots(1,2,figsize=(10,4))
          sn.histplot(df1['age'], bins=45, ax=ax[0])
          ax[0].set_xlabel('Adult age in years')
          ax[0].set_title('Age distribution of ASD positive')
          sn.histplot(df2['Age_Mons'], bins=25, ax=ax[1])
          ax[1].set_xlabel('Toddlers age in months')
          ax[1].set_title('Age distribution of ASD positive')
```

Fig 6.7 Impact of Age



```
In [127]: fig, ax=plt.subplots(1,1,figsize=(22,4))
          sn.countplot(x='contry_of_res',data=df1,order= data1['contry_of_res'].value_counts().index[:15],hue='gender')
          ax.set_title('Positive ASD Adults country wise distribution')
          ax.set_xlabel('Countries')
          plt.tight_layout()
```

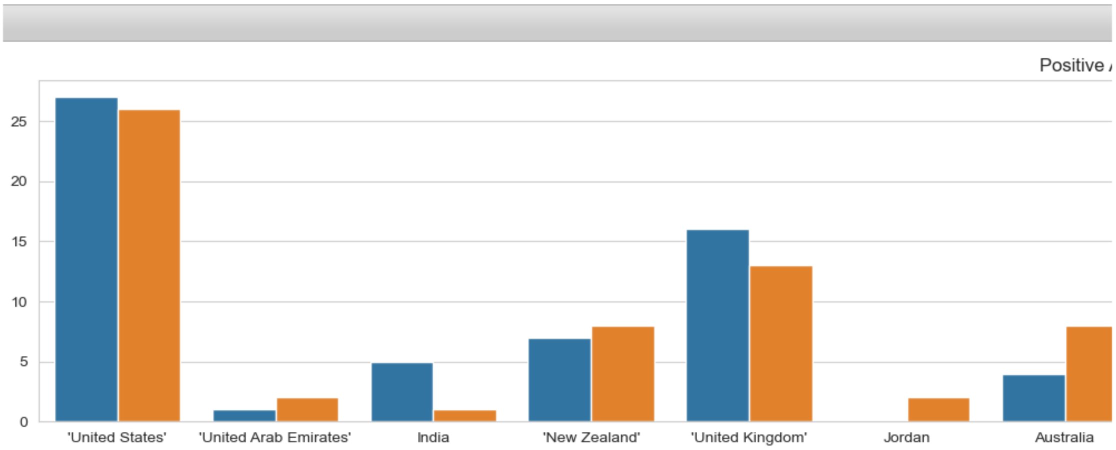
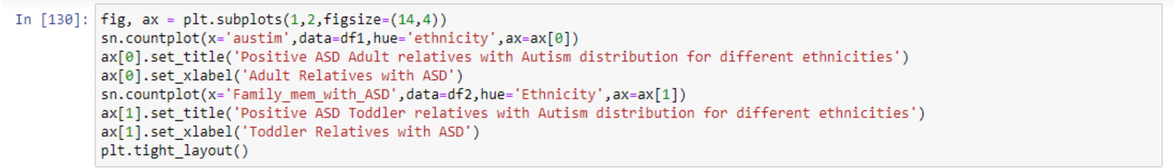


Fig 6.8 Impact of Country



```
In [130]: fig, ax = plt.subplots(1,2,figsize=(14,4))
          sn.countplot(x='austim',data=df1,hue='ethnicity',ax=ax[0])
          ax[0].set_title('Positive ASD Adult relatives with Autism distribution for different ethnicities')
          ax[0].set_xlabel('Adult Relatives with ASD')
          sn.countplot(x='Family_mem_with_ASD',data=df2,hue='Ethnicity',ax=ax[1])
          ax[1].set_title('Positive ASD Toddler relatives with Autism distribution for different ethnicities')
          ax[1].set_xlabel('Toddler Relatives with ASD')
          plt.tight_layout()
```

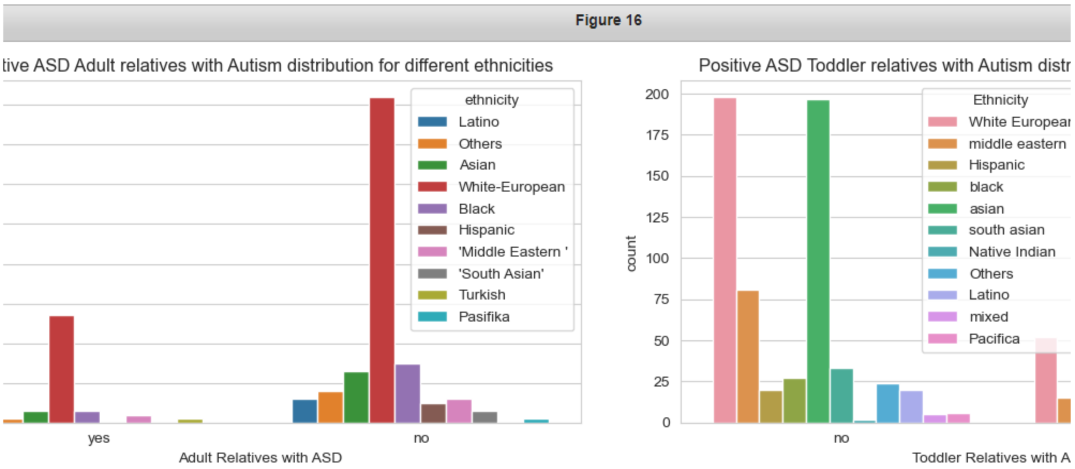
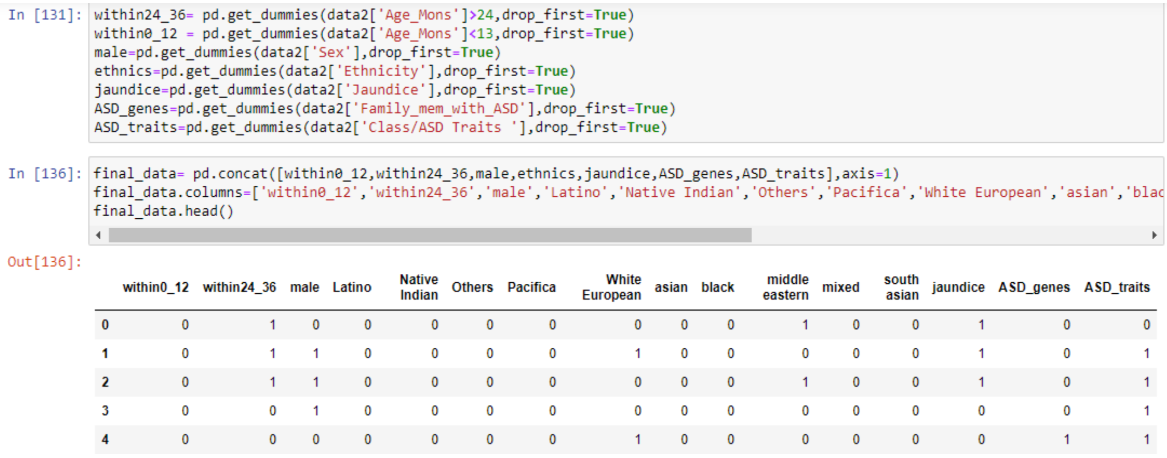


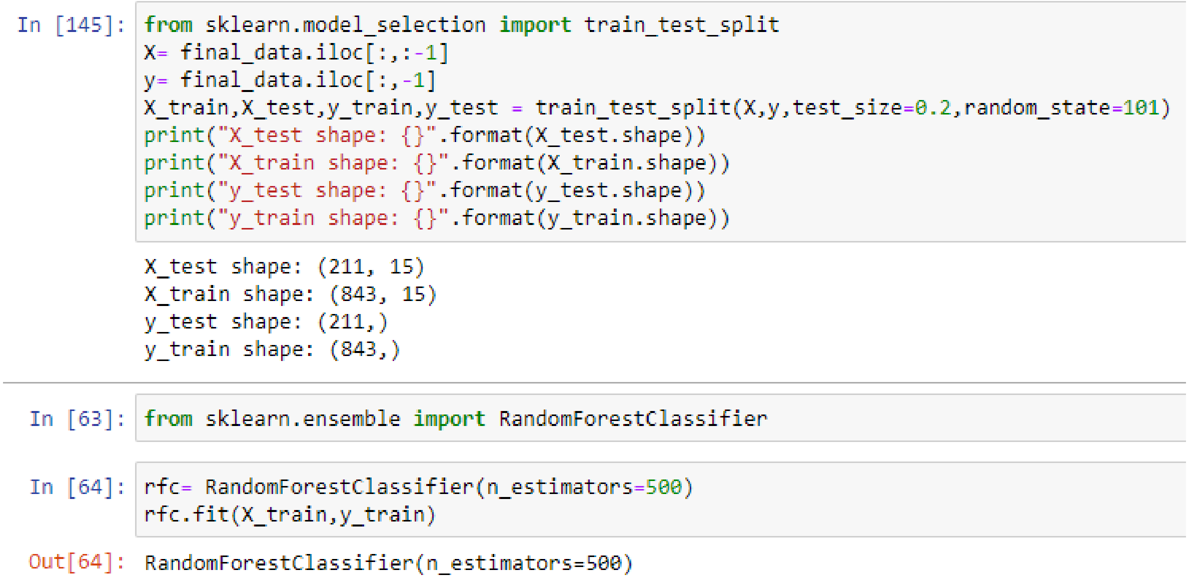
Fig 6.9 Impact of Ethnicity



```
In [131]: within24_36= pd.get_dummies(data2['Age_Mons']>24,drop_first=True)
          within0_12 = pd.get_dummies(data2['Age_Mons']<13,drop_first=True)
          male=pd.get_dummies(data2['Sex'],drop_first=True)
          ethnics=pd.get_dummies(data2['Ethnicity'],drop_first=True)
          jaundice=pd.get_dummies(data2['Jaundice'],drop_first=True)
          ASD_genes=pd.get_dummies(data2['Family_mem_with_ASD'],drop_first=True)
          ASD_traits=pd.get_dummies(data2['Class/ASD Traits '],drop_first=True)

In [136]: final_data= pd.concat([within0_12,within24_36,male,ethnics,jaundice,ASD_genes,ASD_traits],axis=1)
          final_data.columns=['within0_12','within24_36','male','Latino','Native Indian','Others','Pacifica','White European','asian','blac
          final_data.head()

Out[136]:
```

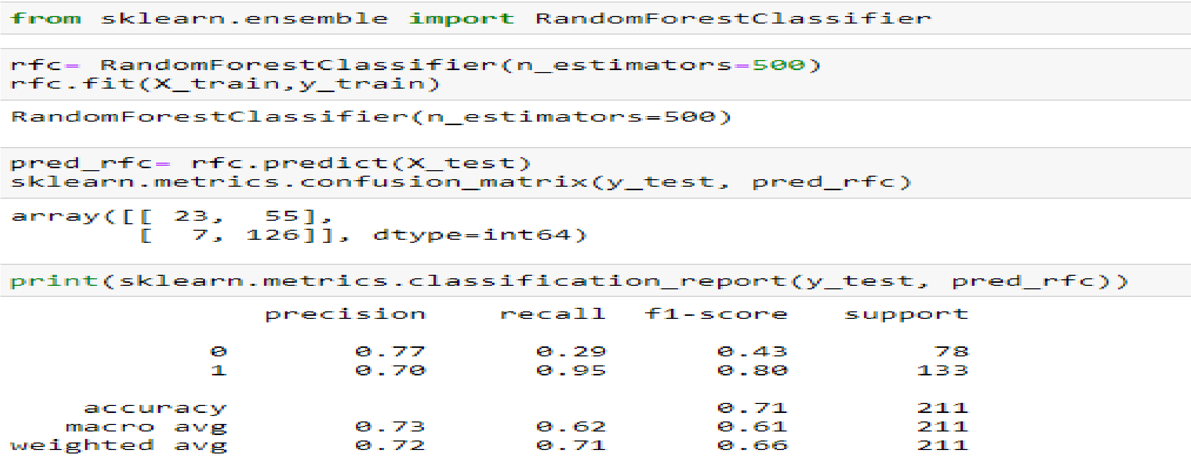
	within0_12	within24_36	male	Latino	Native Indian	Others	Pacifica	White European	asian	black	middle eastern	mixed	south asian	jaundice	ASD_genes	ASD_traits
0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0
1	0	1	1	0	0	0	0	1	0	0	0	0	0	1	0	1
2	0	1	1	0	0	0	0	0	0	0	1	0	0	1	0	1
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1

Fig 6.10 Final dataset



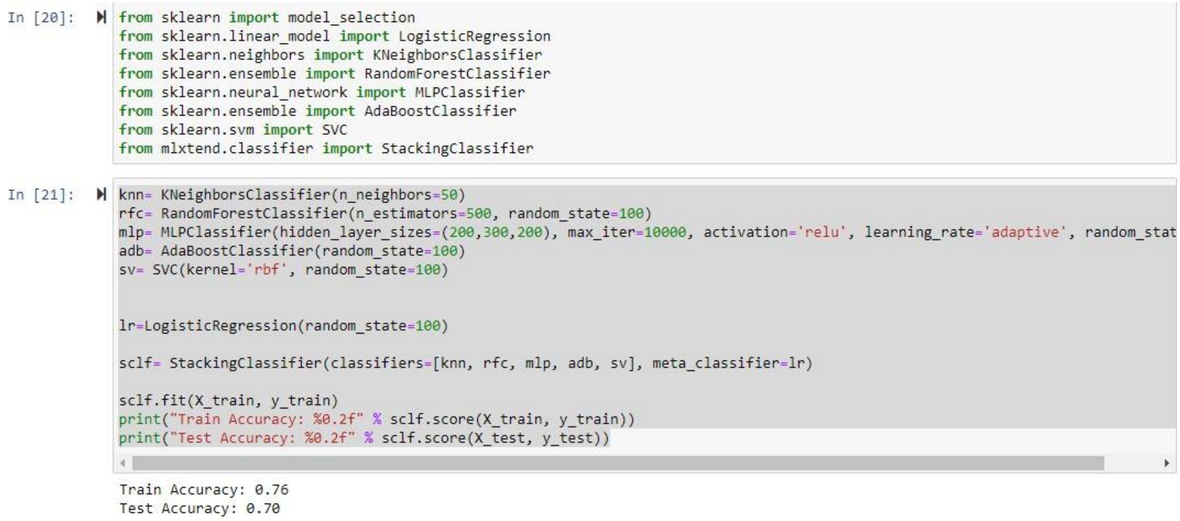
```
In [145]: from sklearn.model_selection import train_test_split
          X= final_data.iloc[:,:-1]
          y= final_data.iloc[:,-1]
          X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=101)
          print("X_test shape: {}".format(X_test.shape))
          print("X_train shape: {}".format(X_train.shape))
          print("y_test shape: {}".format(y_test.shape))
          print("y_train shape: {}".format(y_train.shape))

          X_test shape: (211, 15)
          X_train shape: (843, 15)
          y_test shape: (211,)
          y_train shape: (843,)

In [63]: from sklearn.ensemble import RandomForestClassifier

In [64]: rfc= RandomForestClassifier(n_estimators=500)
         rfc.fit(X_train,y_train)

Out[64]: RandomForestClassifier(n_estimators=500)
```

Fig 6.10 Splitting Dataset



```
from sklearn.ensemble import RandomForestClassifier

rfc= RandomForestClassifier(n_estimators=500)
rfc.fit(X_train,y_train)

RandomForestClassifier(n_estimators=500)

pred_rfc= rfc.predict(X_test)
sklearn.metrics.confusion_matrix(y_test, pred_rfc)

array([[ 23,  55],
       [  7, 126]], dtype=int64)

print(sklearn.metrics.classification_report(y_test, pred_rfc))

              precision    recall  f1-score   support

           0       0.77      0.29      0.43        78
           1       0.70      0.95      0.80       133

    accuracy                           0.71       211
   macro avg       0.73      0.62      0.61       211
weighted avg       0.72      0.71      0.66       211
```

Fig 6.11 Random Forest



```
In [20]:  from sklearn import model_selection
          from sklearn.linear_model import LogisticRegression
          from sklearn.neighbors import KNeighborsClassifier
          from sklearn.ensemble import RandomForestClassifier
          from sklearn.neural_network import MLPClassifier
          from sklearn.ensemble import AdaBoostClassifier
          from sklearn.svm import SVC
          from mlxtend.classifier import StackingClassifier
```

```
In [21]:  knn= KNeighborsClassifier(n_neighbors=50)
          rfc= RandomForestClassifier(n_estimators=500, random_state=100)
          mlp= MLPClassifier(hidden_layer_sizes=(200,300,200), max_iter=10000, activation='relu', learning_rate='adaptive', random_stat
          adb= AdaBoostClassifier(random_state=100)
          sv= SVC(kernel='rbf', random_state=100)


          lr=LogisticRegression(random_state=100)

          sclf= StackingClassifier(classifiers=[knn, rfc, mlp, adb, sv], meta_classifier=lr)

          sclf.fit(X_train, y_train)
          print("Train Accuracy: %0.2f" % sclf.score(X_train, y_train))
          print("Test Accuracy: %0.2f" % sclf.score(X_test, y_test))

          Train Accuracy: 0.76
          Test Accuracy: 0.70
```

Fig 6.12 Stacking

Chapter 7

CONCLUSION

The individual results of the algorithms weren’t as accurate as that after applying Stacking. And for Stacking library and algorithm:

- StackingClassifier algorithm from mxtend library of python produced 76% train and 70% test accuracy.
- StackingCVClassifier algorithm from mxtend library of python produced 75% train 71% test accuracy.
- StackingClassifier algorithm from sklearn library of python produced 74% train 66% test accuracy.

Chapter 8

FUTURE SCOPE

In future the model can be used in various system, web or mobile applications for early Autism detection in Toddlers.

REFERENCES

[1] Shatte, A. B. R., Hutchinson, D. M., & Teague, S. J. (2019). Machine learning in mental health: a scoping review of methods and applications. Psychological Medicine, 1–23. doi:10.1017/s0033291719000151

[2] Croat Med J. 2020 Jun; 61(3): 279–288. doi: 10.3325/cmj.2020.61.279 PMCID: PMC7358693 PMID: 32643346 Artificial intelligence in prediction of mental health disorders induced by the COVID-19 pandemic among health care workers Krešimir Ćosić,1 Siniša Popović,1 Marko Šarlija,1 Ivan Kesedžić,1 and Tanja Jovanovic2

[3] Psychiatry Investig. 2019 Apr; 16(4): 262–269. Published online 2019 Apr 8. doi: 10.30773/pi.2018.12.21.2 PMCID: PMC6504772 PMID: 30947496 Review of Machine Learning Algorithms for Diagnosing Mental Illness Gyeongcheol Cho,1 Jinyeong Yim,2 Younyoung Choi,3 Jungmin Ko,4 and Seoung-Hwan Lee5

[4] Becker, D., van Breda, W., Funk, B., Hoogendoorn, M., Ruwaard, J., & Riper, H. (2018). Predictive modeling in e-mental health: A common language framework. Internet Interventions, 12, 57–67.

[5] Obenshain, M. K. (2018). Application of Data Mining Techniques to Healthcare Data. Infection Control & Hospital Epidemiology, 25(08), 690–695. doi:10.1086/502460

[6] Alonso, S. G., de la Torre-Díez, I., Hamrioui, S., López-Coronado, M., Barreno, D. C., Nozaleda, L. M., & Franco, M. (2018). Data Mining Algorithms and Techniques in Mental Health: A Systematic Review Journal of Medical Systems (2018) 42:161 https://doi.org/10.1007/s10916-018-1018-

[7] Artificial Intelligence for Mental Health and Mental Illnesses: an Overview(2017) Sarah Graham1,2 & Colin Depp1,2,3 & Ellen E. Lee1,2,3 & Camille Nebeker4 & Xin Tu1,2 & Ho-Cheol Kim5 & Dilip V. Jeste1,2,6,7

[8] Sarah A. Graham PhD , Ellen E. Lee MD , Dilip V. Jeste M.D , Ryan Van Patten PhD , Elizabeth W. Twamley PhD , Camille Nebeker EdD, MS , Yasunori Yamada PhD , Ho-Cheol Kim PhD , Colin A. Depp PhD , Artificial Intelligence Approaches to Predicting and Detecting Cognitive Decline in Older Adults: A Conceptual Review, Psychiatry Research (2019), doi: https://doi.org/10.1016/j.psychres.2019.112732

[9] Su, C., Xu, Z., Pathak, J., & Wang, F. (2020). Deep learning in mental health outcome research: a scoping review. Translational Psychiatry, 10(1). doi:10.1038/s41398-020-0780-3

[10] Machine learning in medicine: a practical introduction Jenni A. M. Sidey-Gibbons1 and Chris J. Sidey-Gibbons2,3,4* Fonseka, T. M., Bhat, V., & Kennedy, S. H. (2019).

https://doi.org/10.1186/s12874-019-0681-4

[11] The utility of artificial intelligence in suicide risk prediction and the management of suicidal behaviors. Australian & New Zealand Journal of Psychiatry, 000486741986442. doi:10.1177/0004867419864428

[12] De Choudhury, M., & Kiciman, E. (2018). Integrating Artificial and Human Intelligence in Complex, Sensitive Problem Domains: Experiences from Mental Health. AI Magazine, 39(3), 69–80. doi:10.1609/aimag.v39i3.2815

[13] Methods in predictive techniques for mental health status on social media: a critical review Stevie Chancellor1 ✉ and Munmun De Choudhury2 (2020) https://doi.org/10.1038/s41746-020-0233-7

[14] Prediction of Mental Disorder for employees in IT Industry Sandhya P, Mahek Kantesaria (2019)

[15] Kamran Ul haq, A., Khattak, A., Jamil, N., Naeem, M. A., & Mirza, F. (2020). Data Analytics in Mental Healthcare. Scientific Programming, 2020, 1–9. doi:10.1155/2020/2024160

[16] Kavakiotis, I., Tsave, O., Salifoglou, A., Maglaveras, N., Vlahavas, I., & Chouvarda, I. (2018). Machine Learning and Data Mining Methods in Diabetes Research. Computational and Structural Biotechnology Journal, 15, 104–116.

[17] Prediction of Mental Health Problems Among Children Using Machine Learning Techniques (2019) Ms. Sumathi M.R., Research Scholar, Dept. of Computer Science, Bharathiar University, Coimbatore, India Dr. B. Poorna, Principal,S.S.S. Jain College, T.Nagar,Chennai, India Digital Object Identifier (DOI) : 10.14569/IJACSA.2016.070176

[18] Machine learning in mental health: A systematic scoping review of methods and applications Adrian B. R. Shatte*ab Delyse M. Hutchinsonbcde Samantha J. Teagueb (2019) ARTICLE

[19] Srividya, M., Mohanavalli, S., & Bhalaji, N. (2018). Behavioral Modeling for Mental Health using Machine Learning Algorithms. Journal of Medical Systems, 42(5)

[20] Shatte, A. B. R., Hutchinson, D. M., & Teague, S. J. (2019). Machine learning in mental health: a scoping review of methods and applications. Psychological Medicine, 1–23. doi:10.1017/s0033291719000151

[21] Shailaja, K., Seetharamulu, B., & Jabbar, M. A. (2018). Machine Learning in Healthcare: A Review. 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA). doi:10.1109/iceca.2018.8474918

[22] Zhou, Z., Wu, T.-C., Wang, B., Wang, H., Tu, X. M., & Feng, C. (2020). Machine learning methods in psychiatry: a brief introduction. General Psychiatry, 33(1), e100171. doi:10.1136/gpsych-2019-100171