A Data Mining Based Approach to Predict Autism Spectrum Disorder Considering Behavioral Attributes

BY

JOYOSHREE GHOSH
ID: 152-15-5589

ATIA SUJANA OYSHI

ID: 151-15-5240

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised by
Shaon Bhatta Shuvo
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By
Shah Mohammad Tanvir Siddiquee
Senior Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

MAY 2019

APPROVAL

This Project titled **"A Data Mining Based Approach to Predict Autism Spectrum Disorder Considering Behavioral Attributes"**, submitted by Joyoshree Ghosh, ID No: 152-15-5589 and Atia Sujana Oyshi, ID No: 151-15-5240 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 03/05/2019.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Tarek Habib — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

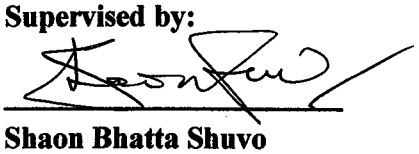
Moushumi Zaman Bonny — **Internal Examiner**
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Swakkhar Shatabda — **External Examiner**
Associate Professor
Department of Computer Science and Engineering
United International University

DECLARATION

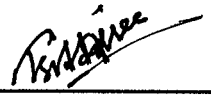
We hereby declare that, this project has been done by us under the supervision of **Shaon Bhatta Shuvo, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



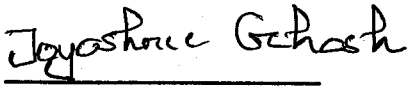
Shaon Bhatta Shuvo
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:



Shah Mohammad Tanvir Siddiquee
Senior Lecturer
Department of CSE
Daffodil International University

Submitted by:



Joyoshree Ghosh
ID: 152-15-5589
Department of CSE
Daffodil International University

Atia Sujana Oyshi
ID: 151-15-5240
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Shaon Bhatta Shuvo**, **Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Data Mining*" to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Prof. Dr. Syed Akhter Hossain, professor, and Head,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

Autism Spectrum Disorder (ASD) is a condition that hinders brain development. It affects a person's way of communicating and behaving. It impacts how a person's way of perceiving and socializing with other people. People with ASD experience different type of symptoms like difficulty with interacting with others, repetitive behaviors, difficulty to function properly in all areas of life. And these symptoms generally occur in early childhood. In this paper, a data mining classification technique was used for the prediction of ASD in adults. Random forest was used as the classifier and accuracy, sensitivity and specificity score 0.9946, 0.9874, 0.9975 were obtained from training set and 0.9571, 0.8571, 0.9821 were obtained from testing set. The dataset used for prediction had 10 behavioral attributes and 10 more individual attributes.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
Introduction

1.1Introduction

ASD is a brain disorder that usually affects a person in his or her childhood and lasts until death. Symptoms are generally noticed to be seen in early childhood when the affected person is a year or two years old. It affects a person's way of interacting with others, way of communicating with others and way of learning. Though this disease is seen early in childhood, many symptoms may not be noticed unless they affect the child's life in significant ways [1]. Those who have ASD generally have problems in interacting with others, or they generally avoid eye contact when other people talk to them. The affected people are seen spending a vast amount of time putting things in order, say the same things over and over again [2].

Data mining is a technology that uses data analysis methods and algorithms for the processing of data. Classification is one kind of data mining task. Classification algorithms are used to build classification models which are used for prediction. The model is first trained with some data which are referred to as training data and then it is tested for prediction on the unseen testing data.

Therefore, the goal of this study is to represent a summary on the performance of Random Forest, a data mining classification technique to predict ASD in a dataset which was collected from the results of effective screening methods.

1.2Motivation

The social and economic influences of ASD and the rising number of Autism patients throughout the earth shows how important it is to develop easily implementable and efficient

screening methods [3]. It is estimated that ASD affects about 1% of people (globally 62.2 million as of 2015) [4] [5]. Males are diagnosed with Autism Spectrum Disorder more than females [6]. Medication can be used to improve the condition of the affected person but it is not very helpful. That is why proper ASD diagnosis is very important.

CHAPTER 2
Background

2.1Related Works

Most studies related to ASD emphasis on genetic attributes. Very few took into account behavioral attributes.

Among them, in a study conducted by Osman Altay & Mustafa Ulas, they used KNN and the LDA algorithm [1]. This study showed if children are affected with ASD by using KNN and LDA classification techniques. The accuracy score was 0.9080 for LDA and 0.8851 for KNN.

In another study SVM, MLP, Decision tree classifier were used which was conducted by V. Pream Sudha and M. S. Vijaya [7]. They took into five monogenic disorders RTT, FXS, TSC, PMS and Timothy syndrome which are related to syndromic ASD, and four different pathogenic gene mutations, namely missense, nonsense, synonymous and frameshift mutations, underlying them. The accuracy were 0.96 for SVM, 0.95 for MLP and 0.98 for Decision Tree. In the study conducted by Wenbo Liu, Zhiding Yu, Bhiksha Raj, Li Yi, Xiaobing Zou & Ming Li showed a new framework for facial recognition and the use of classification techniques for ASD diagnosis [8].

2.2Challenges of The Work

Working with Random Forest classifier is tough as it gives different results different times. Random forest is originally a set of Decision Trees and it randomly selects the height and the split of those trees. So, it gives different results each time the program is run.

To overcome this problem and get a firm score every time, some parameters of Random Forest was set constant. ‘bootstrap’ was set to ‘False’, ‘max_features’ was set to ‘sqrt’ and ‘random_state’ was set to ‘0’.

The 'max_features' parameter is the highest number of features Random Forest is approved to try in each tree. Here, 'sqrt' option receives the square root of the total number of features in a separate run.

The 'random_state' parameter produces an easily replicable solution. A constant value of this parameter always gives the same result.

CHAPTER 3
Methodology

3.1Data Collection

The data were collected from UCI machine learning repository [3]. The data were obtained from Fadi Fayez Thabtah. Tabtah F. also has a paper on Autism Spectrum Disorder Screening [9], a mobile application for ASD diagnosis [10] and a paper on machine learning in autistic spectrum disorder behavioral research [11].The dataset had ten behavioral attributes (AQ-10-Adult) plus ten individual attributes [3].

The dataset had 704 samples and 21 attributes. After eliminating duplicate data there were 699 samples. After eliminating the unnecessary attributes there were 699 samples with 19 attributes. Table 3.1 represents the attribute names, types and descriptions of the dataset used in this study.

TABLE 3.1.Attributes of ASD Dataset

Name	Type	Description
A1_Score	Boolean	Answer to the question
A2_Score	Boolean	Answer to the question
A3_Score	Boolean	Answer to the question
A4_Score	Boolean	Answer to the question
A5_Score	Boolean	Answer to the question
A6_Score	Boolean	Answer to the question
A7_Score	Boolean	Answer to the question
A8_Score	Boolean	Answer to the question
A9_Score	Boolean	Answer to the question

A10_Score	Boolean	Answer to the question
Age	Number	Age of the person
Gender	String	Gender of the person
Ethnicity	String	Ethnic group of the person
Jaundice	Boolean	If the person was born with jaundice or not
Autism	Boolean	If any family member has a PDD or not
Country_of_res	String	Country of residence
Used_app_before	Boolean	If the person has used app before
Result	Integer	The total score
Age_desc	String	Age category
Relation	String	Who is taking the test for the person
ASD	Boolean (yes or no)	If the person has ASD or not

3.2Data Preprocessing

Some of the attributes had categorical data. The categorical data were encoded first. They were converted to numerical values.

The attribute named “Result” was the sum total of the scores of questions (A1-A10). And in all the samples, the value of “age_desc” was ’18 and more’. These two attributes were unnecessary. So, the attributes “Result” and “age_desc” was eliminated too. Then duplicate rows were deleted.

Then, the entire dataset was split into training set and testing set. 80% of the data was kept for training and 20% was kept for testing. Feature scaling was applied to put all the data in same range and same scale.

3.3 LDA

Linear Discriminant Analysis (LDA) is used for reducing dimensionality of datasets. Dimensionality reduction is the process of shortening the number of variables and a set of principal variables is obtained [12].

LDA is a dimensionality reduction technique which is applied on linear datasets. The dataset used in this study was linear as well.

So, LDA was used here as well for reducing dimensionality. LDA reduces dimensionality through maintaining as much of the class information as possible. It searches the limits of the clusters of classes and projects the data points on a line in such a manner so that the clusters created remain as separated as possible, with each one of the clusters having a relatively close distance to a particular centroid.

Dimensionality reduction can be done in two techniques. One of them is feature selection and the other one is feature extraction. LDA uses feature extraction technique for reducing dimensionality. LDA separates most of the classes of the dependent variable by extracting new independent variables.

If we consider two class and μ_1 & μ_2 as their mean of samples feature extraction can be mathematically represented as

$$\omega = S_{\omega}^{-1}(\mu_1 - \mu_2)$$

Where, ω is the eigenvector corresponding to the largest eigenvalue of $S_{\omega}^{-1}S_b$.

Here, S_{ω}=S_1+S_2

S_1 and S_2 are the scatter matrix of class 1 and class 2. And the mathematical formula for S_b is,

$$S_b = \frac{1}{C}\sum_{i=1}^{C}(\mu_i - \mu)(\mu_i - \mu)^T$$

Here, T is a threshold.

So, LDA was applied and dimensionality was reduced for better performance. Figure 3.1 shows the Python code snippet of LDA. Figure 3.2 and Figure 3.3 shows training set variables before and after applying LDA respectively. And Figure 3.4 and Figure 3.5 shows test set variables before and after applying LDA respectively.

```
#lda
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis as LDA
lda=LDA(n_components=None)
X_train=lda.fit_transform(X_train,Y_train)
X_test=lda.transform(X_test)
```

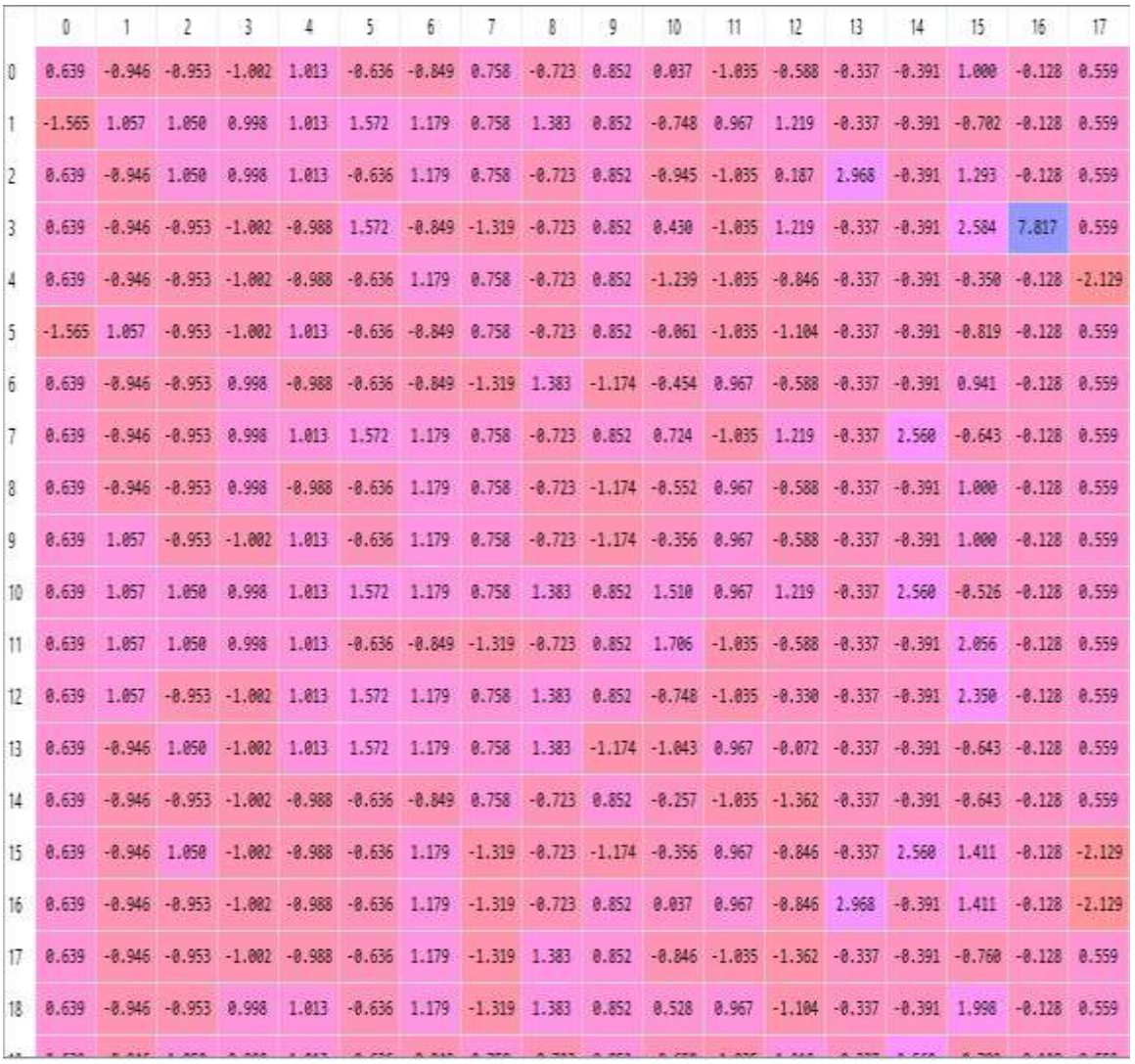
Figure 3.1: Python Code Snippet of LDA

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	0.639	-0.946	-0.953	-1.002	1.013	-0.636	1.179	-1.319	-0.723	0.852	-0.159	-1.035	1.219	-0.337	2.560	-0.702	-0.128	0.559
1	-1.565	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	-1.319	-0.723	-1.174	3.277	0.967	-0.846	-0.337	-0.391	-0.995	-0.128	-2.129
2	0.639	1.057	-0.953	0.998	1.013	1.572	-0.849	-1.319	1.383	0.852	0.037	-1.035	1.219	-0.337	-0.391	-0.643	-0.128	0.559
3	0.639	1.057	1.050	0.998	-0.988	-0.636	1.179	-1.319	-0.723	-1.174	-0.650	0.967	-1.362	-0.337	-0.391	-0.760	-0.128	0.559
4	0.639	1.057	-0.953	0.998	-0.988	-0.636	1.179	0.758	-0.723	-1.174	2.099	0.967	1.219	2.968	-0.391	-0.702	-0.128	0.559
5	-1.565	-0.946	1.050	0.998	-0.988	-0.636	-0.849	0.758	-0.723	0.852	-0.650	0.967	-0.588	-0.337	-0.391	1.000	-0.128	0.559
6	-1.565	1.057	-0.953	-1.002	1.013	-0.636	-0.849	-1.319	-0.723	-1.174	2.295	-1.035	-0.846	-0.337	-0.391	1.117	-0.128	-2.129
7	-1.565	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	-1.319	-0.723	-1.174	0.135	-1.035	1.219	-0.337	-0.391	0.883	-0.128	0.559
8	0.639	-0.946	-0.953	0.998	1.013	1.572	-0.849	0.758	-0.723	0.852	0.135	0.967	-1.362	-0.337	2.560	0.355	-0.128	0.559
9	-1.565	-0.946	1.050	-1.002	1.013	1.572	-0.849	-1.319	1.383	0.852	0.430	0.967	0.445	-0.337	-0.391	-1.054	-0.128	0.559
10	-1.565	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	0.758	-0.723	-1.174	-0.748	0.967	-1.104	-0.337	-0.391	-1.054	-0.128	0.559
11	-1.565	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	-1.319	-0.723	-1.174	-0.257	-1.035	-0.330	-0.337	-0.391	-0.760	-0.128	0.559
12	-1.565	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	0.758	-0.723	0.852	0.724	0.967	1.219	-0.337	-0.391	-0.702	-0.128	-0.785
13	-1.565	-0.946	-0.953	0.998	-0.988	-0.636	-0.849	0.758	-0.723	-1.174	-0.846	0.967	-0.846	-0.337	-0.391	1.411	-0.128	-2.129
14	-1.565	-0.946	1.050	-1.002	-0.988	-0.636	-0.849	-1.319	-0.723	-1.174	2.589	0.967	1.219	-0.337	-0.391	-1.054	-0.128	0.559
15	0.639	-0.946	1.050	0.998	-0.988	-0.636	-0.849	0.758	1.383	0.852	0.332	-1.035	-0.588	-0.337	-0.391	1.000	-0.128	-0.785
16	0.639	1.057	1.050	0.998	-0.988	-0.636	-0.849	0.758	1.383	0.852	0.332	-1.035	-0.330	-0.337	2.560	0.824	-0.128	-0.785
17	0.639	1.057	-0.953	0.998	1.013	1.572	1.179	0.758	1.383	-1.174	0.332	-1.035	-0.330	-0.337	2.560	-0.643	-0.128	0.559
18	0.639	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	0.758	-0.723	-1.174	0.233	-1.035	-0.588	-0.337	-0.391	-0.702	-0.128	0.559

Figure 3.2: Training set variables before applying LDA

	0
0	-0.683
1	-3.059
2	2.089
3	-0.561
4	-0.181
5	-1.225
6	-1.457
7	-2.801
8	0.581
9	0.672
10	-2.870
11	-3.105
12	-2.237
13	-2.132
14	-2.402
15	0.871
16	1.476
17	2.849
18	-1.942

Figure 3.3: Training set variables after applying LDA



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	0.639	-0.946	-0.953	-1.002	1.013	-0.636	-0.849	0.758	-0.723	0.852	0.037	-1.035	-0.588	-0.337	-0.391	1.000	-0.128	0.559
1	-1.565	1.057	1.050	0.998	1.013	1.572	1.179	0.758	1.383	0.852	-0.748	0.967	1.219	-0.337	-0.391	-0.702	-0.128	0.559
2	0.639	-0.946	1.050	0.998	1.013	-0.636	1.179	0.758	-0.723	0.852	-0.945	-1.035	0.187	2.968	-0.391	1.293	-0.128	0.559
3	0.639	-0.946	-0.953	-1.002	-0.988	1.572	-0.849	-1.319	-0.723	0.852	0.430	-1.035	1.219	-0.337	-0.391	2.584	7.817	0.559
4	0.639	-0.946	-0.953	-1.002	-0.988	-0.636	1.179	0.758	-0.723	0.852	-1.239	-1.035	-0.846	-0.337	-0.391	-0.350	-0.128	-2.129
5	-1.565	1.057	-0.953	-1.002	1.013	-0.636	-0.849	0.758	-0.723	0.852	-0.061	-1.035	-1.104	-0.337	-0.391	-0.819	-0.128	0.559
6	0.639	-0.946	-0.953	0.998	-0.988	-0.636	-0.849	-1.319	1.383	-1.174	-0.454	0.967	-0.588	-0.337	-0.391	0.941	-0.128	0.559
7	0.639	-0.946	-0.953	0.998	1.013	1.572	1.179	0.758	-0.723	0.852	0.724	-1.035	1.219	-0.337	2.560	-0.643	-0.128	0.559
8	0.639	-0.946	-0.953	0.998	-0.988	-0.636	1.179	0.758	-0.723	-1.174	-0.552	0.967	-0.588	-0.337	-0.391	1.000	-0.128	0.559
9	0.639	1.057	-0.953	-1.002	1.013	-0.636	1.179	0.758	-0.723	-1.174	-0.356	0.967	-0.588	-0.337	-0.391	1.000	-0.128	0.559
10	0.639	1.057	1.050	0.998	1.013	1.572	1.179	0.758	1.383	0.852	1.510	0.967	1.219	-0.337	2.560	-0.526	-0.128	0.559
11	0.639	1.057	1.050	0.998	1.013	-0.636	-0.849	-1.319	-0.723	0.852	1.706	-1.035	-0.588	-0.337	-0.391	2.056	-0.128	0.559
12	0.639	1.057	-0.953	-1.002	1.013	1.572	1.179	0.758	1.383	0.852	-0.748	-1.035	-0.330	-0.337	-0.391	2.350	-0.128	0.559
13	0.639	-0.946	1.050	-1.002	1.013	1.572	1.179	0.758	1.383	-1.174	-1.043	0.967	-0.072	-0.337	-0.391	-0.643	-0.128	0.559
14	0.639	-0.946	-0.953	-1.002	-0.988	-0.636	-0.849	0.758	-0.723	0.852	-0.257	-1.035	-1.362	-0.337	-0.391	-0.643	-0.128	0.559
15	0.639	-0.946	1.050	-1.002	-0.988	-0.636	1.179	-1.319	-0.723	-1.174	-0.356	0.967	-0.846	-0.337	2.560	1.411	-0.128	-2.129
16	0.639	-0.946	-0.953	-1.002	-0.988	-0.636	1.179	-1.319	-0.723	0.852	0.037	0.967	-0.846	2.968	-0.391	1.411	-0.128	-2.129
17	0.639	-0.946	-0.953	-1.002	-0.988	-0.636	1.179	-1.319	1.383	0.852	-0.846	-1.035	-1.362	-0.337	-0.391	-0.760	-0.128	0.559
18	0.639	-0.946	-0.953	0.998	1.013	-0.636	1.179	-1.319	1.383	0.852	0.528	0.967	-1.104	-0.337	-0.391	1.998	-0.128	0.559

Figure 3.4: Test set variables before applying LDA

	0
0	-0.717
1	2.896
2	0.854
3	-0.306
4	-1.210
5	-0.964
6	-0.738
7	1.487
8	-0.880
9	-0.095
10	3.758
11	0.650
12	2.771
13	1.929
14	-1.711
15	-1.431
16	-1.711
17	-0.456
18	1.093

Figure 3.5: Test set variables after applying LDA

3.4 Prediction Algorithm

Data mining is the technique of extracting meaningful information from data [13]. In this study, a data mining algorithm for classification named Random Forest was used. Random Forest is an ensemble method which works through generating a forest of Decision Trees. An ensemble is a series of classifiers. Random Forest combines a series of Decision Tree classifiers.

In a Decision Tree every node represents a test on the value of an attribute, every branch is the result of the test, and the leaves represents classes [14].

However, overfitting can easily occur using Decision Tree classifier. But, Random Forest technique corrects Decision Trees' habit of overfitting. In Random Forest many Decision Trees are generated. And the procedure of classification is carried out on every tree and for each run the tree which gets most amount of the votes gets chosen [15]. Thus, Random Forest is much more efficient than Decision Tree.

Random Forest adds the classification done by the Decision Trees and every one of the tree is constructed by the values from sets of random vectors [16]. Random vectors are used for generating Decision Trees.

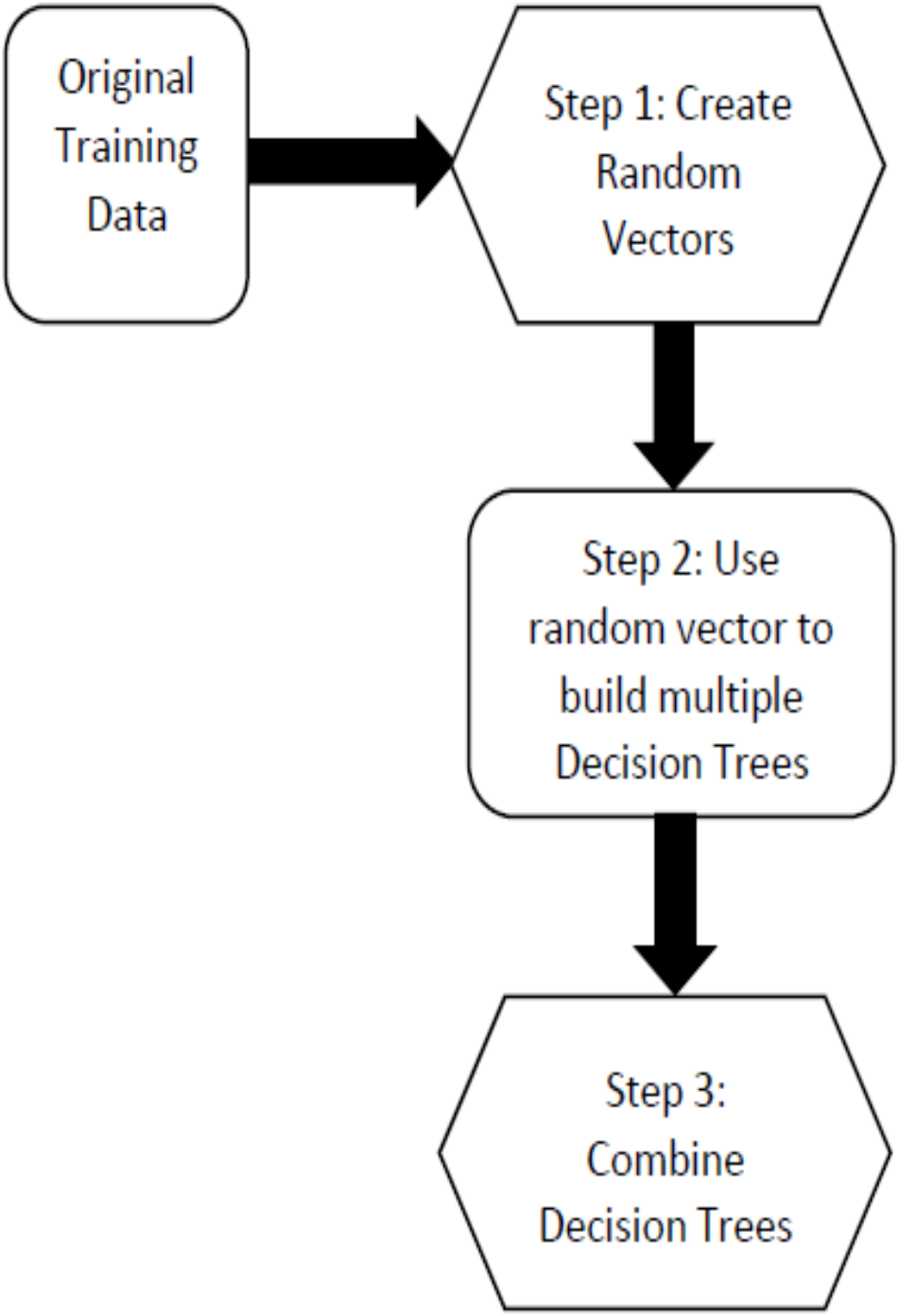


Figure 3.6: Random Forest explained

Figure 3.7 shows the Python code snippet of Random Forest.

```
#fitting classifier to the training set for Random Forest
from sklearn.ensemble import RandomForestClassifier
classifier=RandomForestClassifier(bootstrap='False',max_features='sqrt',random_state=0)
classifier.fit(X_train, Y_train)
```

Figure 3.7: Python Code Snippet of Random Forest

3.5 Tools and Techniques Used for the Study

Spyder IDE: Random Forest was implemented through Spyder IDE. Spyder is a powerful integrated development environment (IDE) for programming in the Python.

Python: Python is an open source programming language which is used in many different applications. There are many tools that efficiently work with Python. And several of those tools are built for data science.

NumPy: NumPy is a module for Python. It has many useful features for operations on n-arrays and matrices.

Pandas: Pandas is another Python module and it contains data structures and tools that are designed for data analysis operations.

Matplotlib: Matplotlib is a library for the Python programming language which is used for plotting. Matplotlib is used to draw line graphs, pie charts, histograms and other useful graphs.

Scikit-learn: It is a library for Machine Learning which is also made in Python. It is used for implementing machine learning algorithms.

3.6 Performance Evaluation

There are many metrics to evaluate a classification model's performance. However, in this case, three classification performance measures were used for evaluating the model's performance. Accuracy, Sensitivity and Specificity are the three measures. Formulas for calculating these measures are shown below.

Accuracy= (TP+TN)/ (TP + TN + FP + FN)

Sensitivity= TP/ (TP + FN)

Specificity= TN/ (TN + FP)

Here, TP stands for True Positive, TN stands for True Negative, FP stands for False Positive and FN stands for False Negatives.

True positives are the outcomes of the model's correct predictions of the positive classes and true negatives are the outcomes when the model correctly predicts the negative classes. On the other hand, false positives are the outcomes of the model's incorrect predictions of the positive classes and false negatives are the results of incorrect predictions of the negative classes.

TP, TN, FP & FN can be obtained from confusion matrix. A confusion matrix is a table which is another performance measure of classification models. Table 3.2 shows what each rows and columns of confusion matrix represents. Figure 3.8 and Figure 3.9 shows the Python code snippet and the output of confusion matrix for training dataset and Figure 3.10 and Figure 3.11 shows the Python code snippet and the output of confusion matrix for testing dataset.

TABLE 3.2 Confusion Matrix

	Predicted: No	Predicted : Yes
Actual : No	TN	FP
Actual : Yes	FN	TP

```
#confusion metrix
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(Y_train,Y_pred)
```

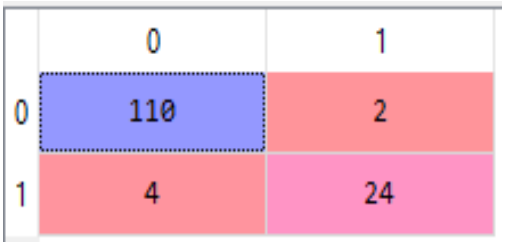
Figure 3.8: Python Code Snippet of Confusion Matrix (training set)

	0	1
0	399	1
1	2	157

Figure 3.9: Confusion Matrix (training set)

```
#confusion metrix
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(Y_test,Y_pred)
```

Figure 3.10: Python Code Snippet of Confusion Matrix (testing set)



	0	1
0	110	2
1	4	24

Figure 3.11: Confusion Matrix (testing set)

3.6 Workflow

The total methodology can be expressed through a workflow diagram. The figure Below is the workflow diagram.

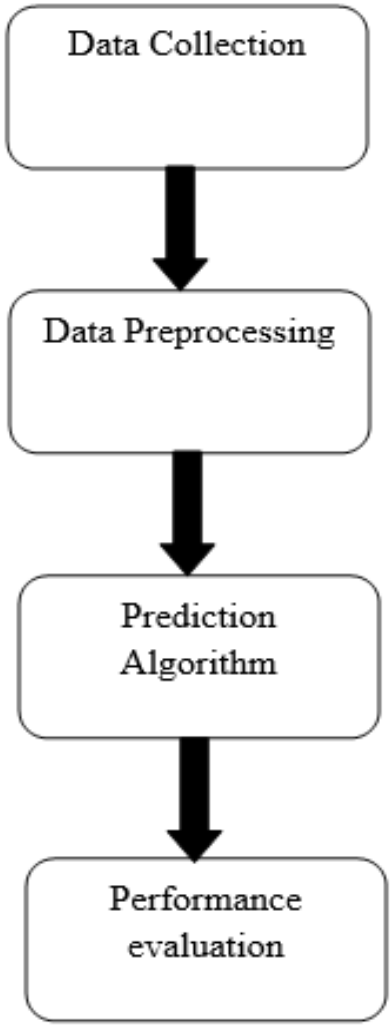


Figure 3.12: Workflow Diagram

CHAPTER 4
Results

4.1 Result

Table 2 & Table 3 shows the prediction results obtained from classification via the Random Forest classifier. Table 2 shows the values of TP, TN, FP and FN. Accuracy, Sensitivity and Specificity are shown in table 3. Figure 4.1, shows the chart representation of Accuracy, Sensitivity and Specificity and Figure 4.2 shows the Receiver Operating Characteristics (ROC) curve. The training dataset had 559 samples and the testing dataset had 140 samples.

TABLE 4.1 VALUES OF TP,TN,FP & FN

	TP	**TN**	**FP**	**FN**
Training Dataset	157	399	1	2
Testing Dataset	24	110	2	4

TABLE 4.2 PERFORMANCE MEASURES

	Accuracy	**Sensitivity**	**Specificity**
Training Dataset	0.9946	0.9874	0.9975
Testing Dataset	0.9571	0.8571	0.9821

4.2 Result Analysis

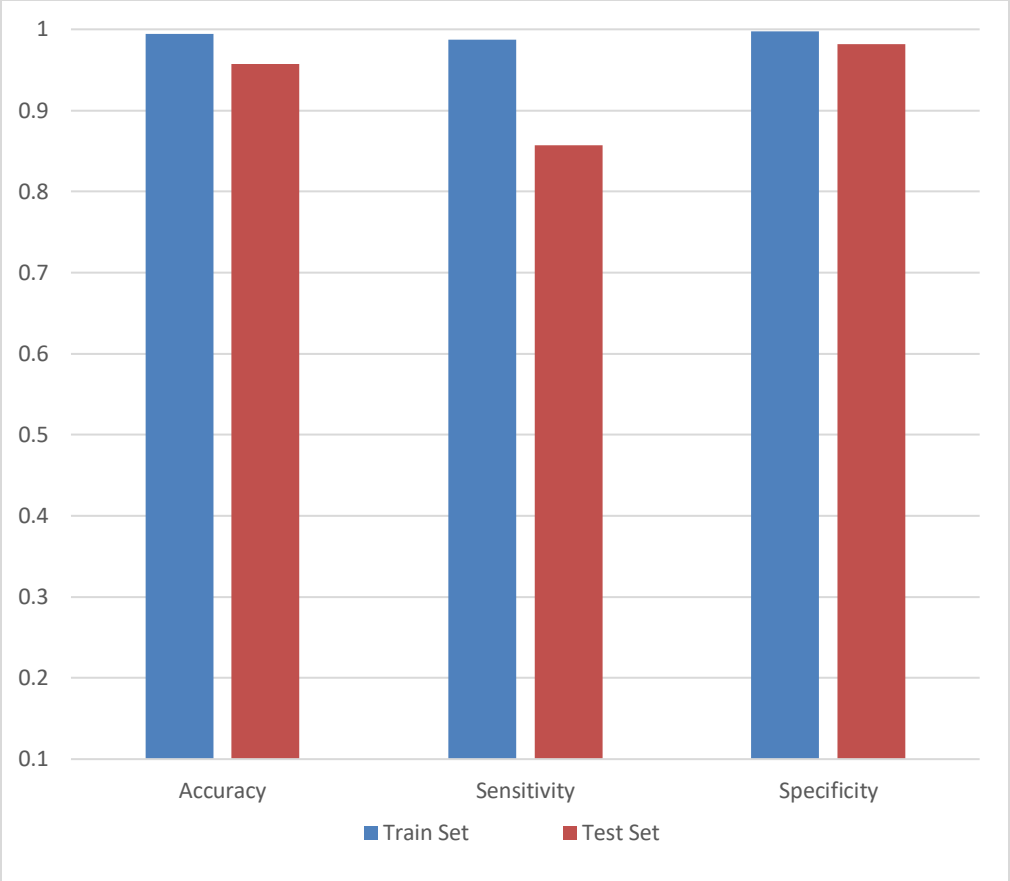


Figure 4.1: Chart Representing the results

We can see from the chart above the difference between training and testing set results are very small. So, it can be said that no overfitting has occurred. And the scores are good.

Receiver Operating Characteristics (ROC) curve is one of the most important evaluation metrics for evaluating classification model's performance. The curve is plotted with TPR is on y-axis and FPR is on x-axis where TPR is the true positive rate and FPR is false positive rate. Formulas for calculating TPR & FPR are given below.

TPR= TP/ (TP + FN)

FPR= FP/ (TN + FP)

The greatest classifying model should give a result which is pointed in the upper left corner of the ROC space which represents there are no false negatives.

Figure 4.2 shows the Python code snippet and Figure 4.3 shows the output of ROC curve.

```
# fpr, tpr
Train= np.array([0.0025, 0.9874])
Test= np.array([0.0178, 0.8571])

# plotting
plt.scatter(Train[0], Train[1], label = 'Train', facecolors='black',edgecolors='orange', s=400)
plt.scatter(Test[0], Test[1], label = 'Test', facecolors='orange', edgecolors='orange', s=400)

plt.plot([0, 1], [0, 1], 'k--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.0])
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('Receiver operating characteristic curve')
plt.legend(loc='lower center')
plt.show()
```

Figure 4.2: Python Code snippet of ROC Curve

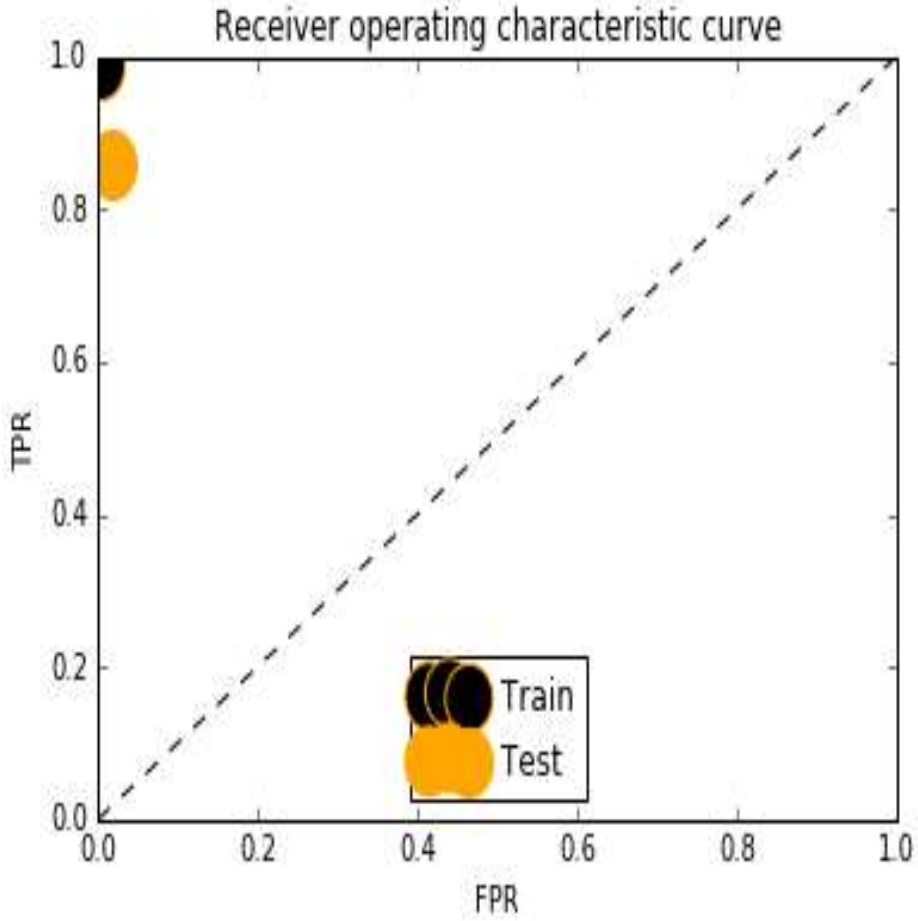


Figure 4.3: ROC Curve

Based on the ROC curve it can be said that both the results from training set and testing set is satisfying and the performance was good.

CHAPTER 5
Conclusion

5.1 Summary of The Study & Conclusion

ASD diagnosis is very important and proper diagnosis is very important. But without any clinical or genetic tests ASD diagnosis is possible based solely on behavioral attributes. In this study it was tried to predict ASD in adults using Random Forest classifier. The data was collected from UCI machine learning repository which were collected from the results of a test that included effective screening methods. As most ASD dataset are genetic in nature, this dataset was exceptional because of having behavioral features.

The purpose of the study was the betterment of prediction of ASD in adults based on behavioral attributes. Random Forest algorithm was used for building the prediction model and the model was quite efficient as the scores of Sensitivity, Specificity & Accuracy were good.

5.2 Future Implementations

Accuracy, sensitivity, specificity were calculated from the values of TP, TN, FP and NP which were used for evaluating the model's performance. Both training and testing dataset's performance was evaluated. From the performance measures, it can be said that the classification model's performance was good. So applications for diagnosis of ASD can be developed based on this study and it can be helpful in diagnosis of ASD affected patients. Without doing any clinical tests this applications can be helpful for patients as they can find out whether they have ASD or not only by answering some questions.

REFERENCES

[1] Osman Altay & Mustafa Ulas. (2018). Prediction of the Autism Spectrum Disorder Diagnosis with Linear Discriminant Analysis Classifier and K-Nearest Neighbor in Children.

[2] Medlineplus.gov. (2019). Autism Spectrum Disorder: MedlinePlus. [Online] Available at: https://medlineplus.gov/autismspectrumdisorder.html [Accessed 10 Feb. 2019].

[3] Archive.ics.uci.edu. (2019) UCI Machine Learning Repository: Autism Screening Adult Data Set. [Online] Available at: https://archive.ics.uci.edu/ml/datasets/Autism+Screening+Adult# [Accessed 10 Feb. 2019].

[4] American Psychiatric Association (2013). "Autism Spectrum Disorder. 299.00 (F84.0)". Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition (DSM-5). Arlington, VA: American Psychiatric Publishing. pp. 50–59.

[5] GBD 2015 Disease and Injury Incidence and Prevalence, Collaborators. (8 October 2016)

[6] Comer, Ronald J. (2016). Fundamentals of Abnormal Psychology. New York: Worth Macmillan Learning. p. 457.

[7] V. Pream Sudha and M. S. Vijaya. (2018). Machine Learning-Based Model for Identification of Syndromic Autism Spectrum Disorder

[8] Wenbo Liu, Zhiding Yu, Bhiksha Raj, Li Yi, Xiaobing Zou & Ming Li. (2015). Efficient Autism Spectrum Disorder Prediction with Eye Movement: A Machine Learning Framework

[9] Thabtah, F. (2017, May). Autism Spectrum Disorder Screening: Machine Learning Adaptation and DSM-5 Fulfillment. In Proceedings of the 1st International Conference on Medical and Health Informatics 2017 (pp. 1-6). ACM.

[10] Thabtah, F. (2017). ASDTests. A mobile app for ASD screening. www.asdtests.com [accessed December 20th, 2017].

[11] Thabtah, F. (2017). Machine Learning in Autistic Spectrum Disorder Behavioural Research: A Review. To Appear in Informatics for Health and Social Care Journal. December, 2017 (in press)

[12] Roweis, S. T.; Saul, L. K. (2000). "Nonlinear Dimensionality Reduction by Locally Linear Embedding". Science. 290 (5500): 2323–2326.

[13] Uma Ojha & Dr. Savita Goel. (2017). A Study on Prediction of Breast Cancer Recurrence Using Data Mining Techniques.

[14] Jiawei Han and Micheline Kamber, “Data Mining Concepts and Techniques”, third edition, Morgan Kaufmann Publishers an imprint of Elsevier. p. 18,333

[15] Divyansh Kaushik & Karamjit Kaur. (2016). Application of Data Mining for High Accuracy Prediction of Breast Tissue Biopsy Results

[16] Introduction to Data Mining by by Pang-Ning Tan, Michael Steinbach, Vipin Kumar. p. 290.