DSCI 6003 - Final Project
Fall 2020

Autism Spectrum Disorder Screening Using Machine Learning

By:
Archana Dasharath Marol

Under the guidance of:
Prof. Travis Millburn

ABSTRACT

Autism Spectrum Disorder (ASD) is a neurodevelopmental disorder in which a person has a lifelong effect on interaction and communication with others. Autism can be diagnosed at any stage in once life, the symptoms usually appears in the first two years of life, so it is said to be a "behavioural disease". The ASD problem starts with childhood and continues to keep going on into adolescence and adulthood.

Due to the rise in use of machine learning techniques in the research dimensions of medical diagnosis, in this project there is an attempt to try some of the techniques such as Naïve Bayes, KNN, Decision Tree and Random Forest for predicting and analysis of ASD using the behavioural traits. The proposed techniques are evaluated on available dataset from the sources; Kaggle. The obtained dataset related to ASD screening contains a total of 704 instances and 21 attributes. After applying various machine learning techniques and doing some pre-processing like handling missing values and one-hot-encoding, results suggest that Random Forest and Decision Tree prediction models work better on the datasets with higher accuracy of more than 90%. Classifiers like Random forests has a feature_importance_ attribute, which is a function that ranks the importance of features according to the chosen classifier hence the accuracy is more. The results are obtained by using Evaluation technique such as Confusion Matrix and ROC-AUC. The GitHub link for the project is also provided in this report.

Table of Contents

Chapter One

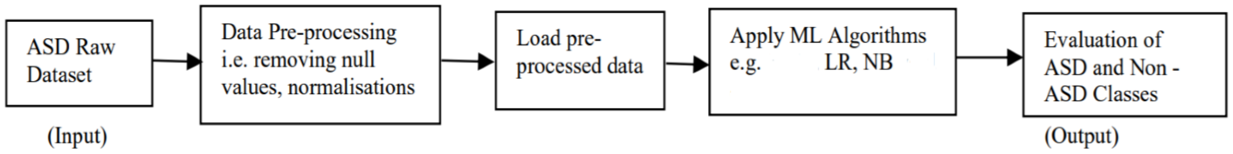
1 INTRODUCTION

Autism spectrum disorder (ASD) is a complex developmental condition that involves challenges in social interaction, speech and nonverbal communication, and repetitive behaviors. Early diagnosis and treatment are important to reducing the symptoms of autism. Early diagnosis also helps in improving the quality of life for people with autism and their families. ASD is diagnosed based on observing how the child talks and acts in comparison to other children of the same age. There is no medical test for autism. Doctors and trained professionals diagnose autism typically by talking with the child and asking questions to the parents and other caregivers.

In ASD, a person's life is usually affected for a person's entire lifetime. In this, both environmental and genetic factors may turn out to be the causing factors for this disease. The symptoms of this problem may be started at the age of three years and may continue for the rest of the patient's life. As mentioned earlier, it is not possible to treat the patient completely, suffering from this disease. But its effects can be reduced if the symptoms are early detected. There is some risk factor which influences ASD like as low birth weight, having old parents or a sibling with ASD and so on. Some social interaction and communication problems like as: Inappropriate laughing and giggling, no sensitivity of pain, not able to make eye contact, no proper response to sound, may not have a wish for cuddling, not able to express their gestures and so on. People with ASD also have difficulty with constrained interests and consistently repetition of behaviors. The specific examples of the types of behaviors are repeating words or phrases much time, the Person will be upset when a routine is going to change, having a little interest in certain matters of the topic like numbers, and facts.

With the rise of application of machine learning-based models in the predictions of various human diseases, their early detection based on various physiological and health parameter now seems possible. This factor motivates to increase interest in the detection and analysis of ASD diseases for the better treatment methodology. There are several other mental disorders whose few symptoms are very similar to those with ASD symptoms, so detection of ASD becomes a bit challenge. Using the above-mentioned symptoms and behavioral traits can be used as variables for building a machine learning algorithm.

In this project, the behavioral traits that are effective in detecting the ASD cases are used in predicting whether a child has ASD or not. The available dataset from Kaggle source is used to run four different classification algorithms of machine learning, Naïve Bayes, KNN, Decision Tree and Random Forest. The description of the dataset and the algorithms are explained in the next part of this report. The flow of the project is shown below:


ASD Raw Dataset
(Input)
Data Pre-processing i.e. removing null values, normalisations
Load pre-processed data
Apply ML Algorithms e.g. LR, NB
Evaluation of ASD and Non - ASD Classes
(Output)

. Steps in the proposed ASD detection solution

Chapter Two

2 LITERATURE SURVEY

1. Autistic Spectrum Disorder (ASD) is a neurodevelopment disorder characterized by impaired communication, cognitive, and social skills and abilities. Existing screening tools for detection of autism are expensive, cumbersome and time-intensive. The objective of this project is to create a low cost, quick and easy to use diagnostic test for ASD by building a machine learning algorithm that can predict with close to 100% accuracy, whether a person has ASD, based on behavioral traits.

   Machine learning models were developed for five different classification algorithms namely Logistic Regression (LR), Decision Trees (DT), Gaussian Naive Bayes (NB), Support Vector Machines (SVM) and Neural Networks (NN). Coding was done in Python using scikit-learn in Jupyter notebook. The models were trained using ASD screening data from UC Irvine machine learning repository. Data consists of response to questions on behavioral traits, age, gender, ethnicity, family history and if the person had jaundice when born. For evaluating the models, 10-fold cross validation technique was used in which the data is partitioned into ten equal sizes and nine samples were used for training and one for validation. Accuracy score, confusion matrix that describes performance of the model and classification report were generated for each model.

   The models developed have achieved average accuracies of 96.7%(LR), 90.7%(DT), 95.4%(NB), 92.3%(SVM) and 97.4%(NN) with standard deviations of 0.023(LR), 0.036(DT), 0.034(NB), 0.035(SVM) and 0.01(NN) respectively. Neural networks based model is the best with highest possible accuracy and lowest variance. Dropping gender and age from the input feature list improved accuracy which means they are not useful for predicting ASD. Accuracy of models drop if only response to questions i.e. behavioral traits are used for training. Family history, ethnicity and if the person was born with jaundice are important factors to consider along with behavioral traits for ASD screening.

   Neural networks-based machine learning model developed predicts if someone has ASD with 97.4% accuracy, based on answers to behavioral traits questions. People can take this test from the comfort of their home, on the on their computer or mobile phone for initial assessment, before doing more expensive diagnostic tests.

2. Autism Spectrum Disorder (ASD) screening can improve prognosis via early diagnosis and intervention, but lack of time and training can deter pediatric screening. The Modified Checklist for Autism in Toddlers, Revised (M-CHAT-R) is a widely used screener, but requires follow-up questions and error-prone human scoring and interpretation. We consider an automated machine learning (ML) method for overcoming barriers to ASD screening, specifically employing the feed-forward artificial neural network (fANN).
   The fANN technique was applied using archival M-CHAT-R data of 14,995 toddlers (16-30 months, 46.51% male). The 20 M-CHAT-R items were inputs, and ASD diagnosis after follow-up and diagnostic evaluation (i.e., ASD or not ASD) was output. The sample was divided into subgroups by race (i.e., White and Black), sex (i.e., boys and girls), and maternal education (i.e,

below and above 15 years of education completed) to examine subgroup differences. Each subgroup was evaluated for best-performing fANN models.

For the total sample, best results yielded 99.72% correct classification using 18 items. Best results yielded 99.92% correct classification using 14 items for White toddlers and 99.79% correct classification using 18 items for Black toddlers. In boys, best results yielded 99.64% correct classification using 18 items, while best results yielded 99.95% correct using 18 items in girls. For the case when maternal education is 15 years or less (i.e., Associate Degree and below), best results were 99.75% correct classification when using 16 items. Results were essentially the same when maternal education was 16 years or more (i.e., above Associate Degree); that is 99.70% correct classification was obtained using 16 items.

The ML method was comparable to the M-CHAT-R with follow-up items in accuracy of ASD diagnosis, while using fewer items. Therefore, ML may be a beneficial tool in implementing automatic, efficient scoring that negates the need for labor-intensive follow-up as well as circumvents human error, providing an advantage over prior screening methods.

Chapter Three

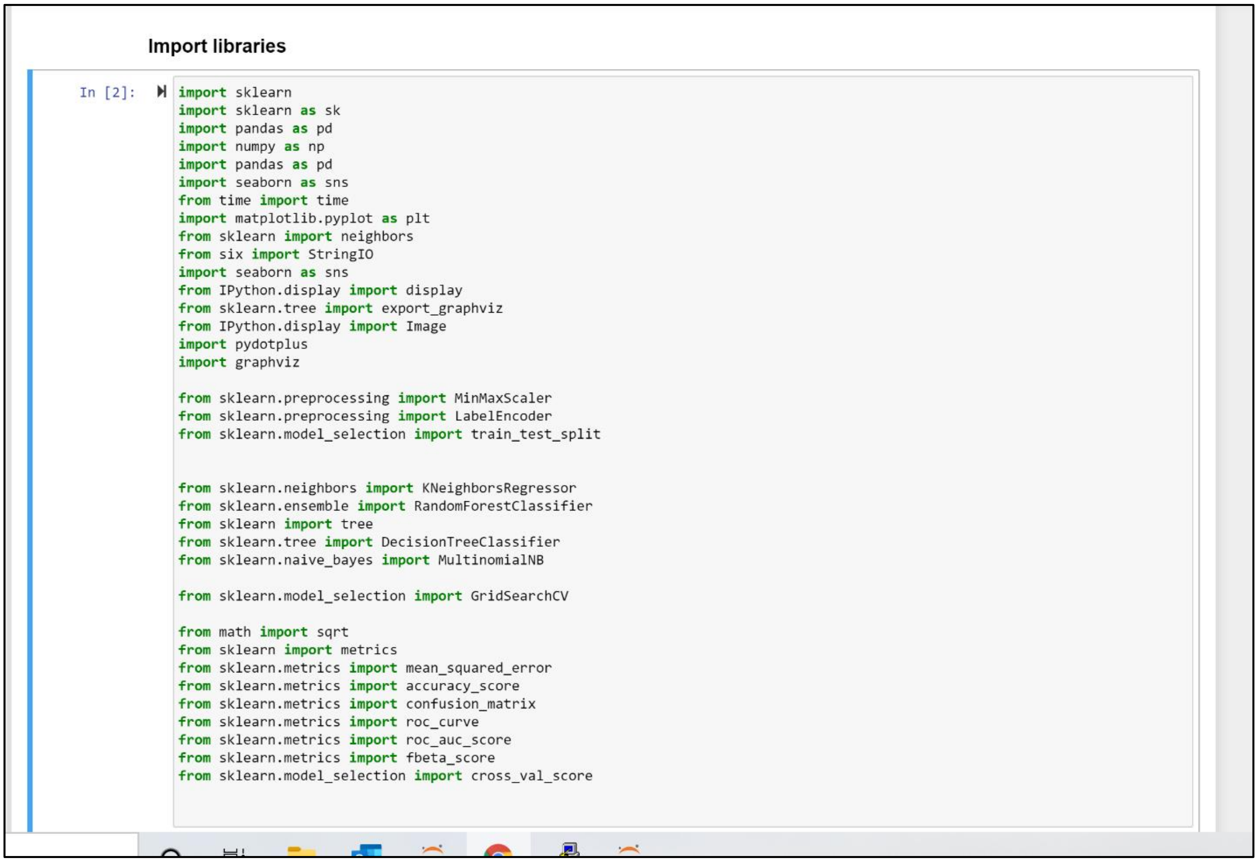
3 SYSTEM REQUIREMENTS

To run this project to perform the machine learning algorithms, below software and libraries are were installed:

For Windows, Python 3.6 was installed with the help of the free version of Anaconda. Jupyter Notebook was used as a tool to run the python code. The libraries required to run this project are:

- NumPy
- Pandas
- Matplotlib
- Seaborn
- Sklearn

The screenshot from the Juputer notebook showing the libraries installed is shown below.



Import libraries

In [2]:

```
import sklearn
import sklearn as sk
import pandas as pd
import numpy as np
import pandas as pd
import seaborn as sns
from time import time
import matplotlib.pyplot as plt
from sklearn import neighbors
from six import StringIO
import seaborn as sns
from IPython.display import display
from sklearn.tree import export_graphviz
from IPython.display import Image
import pydotplus
import graphviz

from sklearn.preprocessing import MinMaxScaler
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split


from sklearn.neighbors import KNeighborsRegressor
from sklearn.ensemble import RandomForestClassifier
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import MultinomialNB

from sklearn.model_selection import GridSearchCV

from math import sqrt
from sklearn import metrics
from sklearn.metrics import mean_squared_error
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import roc_auc_score
from sklearn.metrics import fbeta_score
from sklearn.model_selection import cross_val_score
```

Chapter Four

4 DATA EXPLORATION AND DATA PREPROCESSING

About Dataset:

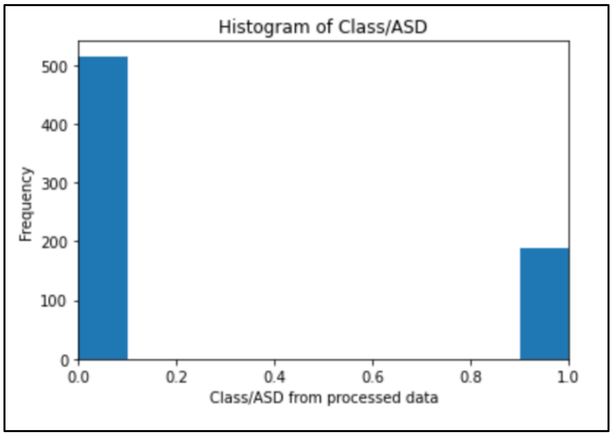
A dataset related to autism screening of adults that contained 21 and 704 instances.

The dataset has ten behavioural features (AQ-10-Adult) plus ten individuals characteristics that have proved to be effective in detecting the ASD. This data contains the following attributes:

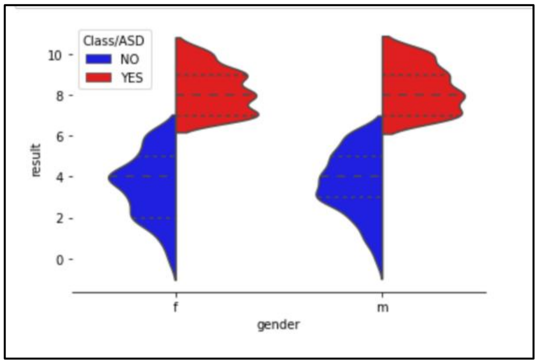
- **age**: *number* (Age in years).
- **gendar**: *String* [Male/Female].
- **ethnicity**: *String* (List of common ethnicities in text format).
- **Born with jaundice**: *Boolean* [yes or no].
- **Family member with PDD**: *Boolean* [yes or no].
- **Who is completing the test**: *String* [Parent, self, caregiver, medical staff, clinician ,etc.].
- *Country of residence *: *String* (List of countries in text format).
- *Used the screening app before *: *Boolean* [yes or no] (Whether the user has used a screening app)
- **Screening Method Type**: *Integer* [0,1,2,3] (The type of screening methods chosen based on age category (0=toddler, 1=child, 2= adolescent, 3= adult).
- **Question 1-10 Answer**: *Binary* [0, 1] (The answer code of the question based on the screening method used).
- **Screening Score**: *Integer* (The final score obtained based on the scoring algorithm of the screening method used. This was computed in an automated manner).

Data Exploration:

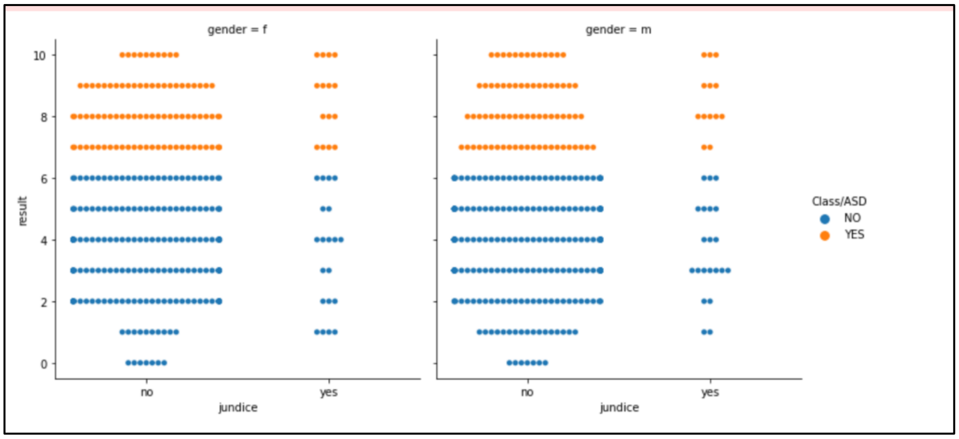
Among 704, number of observations that have autism are 189 and rest do not have autism.



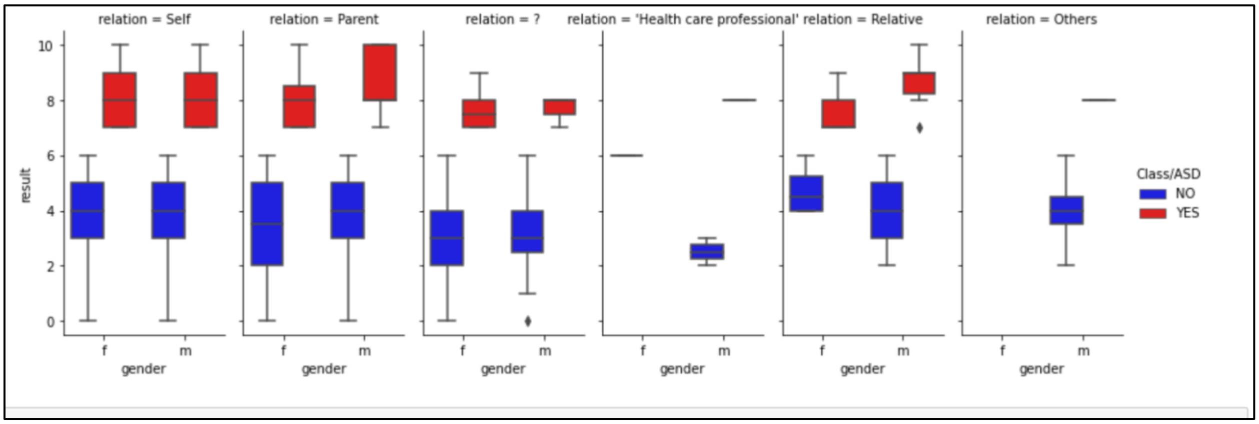
The graph below show the Male and Female split of two classes:



The graph shows the comparison of two classes which patients having jaundice, and which do not have jaundice:



This graph shows the box plot that shows the number of people who has answered the questionaries, whether it's answered by self, parents, relative and so on. It also shows the classes split and gender split.



Data Pre-processing:

Data pre-processing is a technique in which the raw data is transformed into a meaningful and understandable format. Real-world data is commonly incomplete and inconsistent because it contains lots of errors and null values. A good, pre-processed data always yields to a better result than the results obtained from data that is not pre-processed.

Various Data pre-processing methods are used to handle incomplete and inconsistent data:

1. **Dropna**

There were a few missing values in the data set. Before droping every row that's missing data, I make sure not to bias the data in any way. We need to make sure that there does not appear to be any sort of correlation to what sort of data has missing fields. If there were then one have to try and go back and fill that data in.

2. **Sklearn MinMaxScaler**

The data is transformed by scaling each feature to a given range. This estimator scales and translates each feature individually such that it is in the given range on the training set between zero and one.

3. **LabelEncoder**

Some of the variables in the dataset were object datatype, those variables were converted to numeric so that they can be considered in the algorithm. The object datatypes are also useful variables that need to be converted to numeric for sklearn algorithm to work and given results. Something called LabelEncoder is available in sklearn libraries that can be used for this task

4. **One-Hot-encoding**

One-hot encoding creates a "dummy" variable for each possible category of each non-numeric feature. For example, assume someFeature has three possible entries: A, B, or C. We then encode this feature into someFeature_A, someFeature_B and someFeature_C.

Additionally, as with the non-numeric features, I need to convert the non-numeric target label, 'Class/ASD' to numerical values for the learning algorithm to work. Since there are only two possible categories for this label ("YES" and "NO" to Class/ASD), I can avoid using one-hot encoding and simply encode these two categories as 0 and 1, respectively. In code cell below, I will implement the following:

Use pandas.get_dummies() to perform one-hot encoding on the 'features_minmax_transform' data.

Convert the target label 'asd_raw' to numerical entries.

Set records with "NO" to 0 and records with "YES" to 1.

5. **train_test_split:**

   The whole dataset has been split into two parts i.e. one part is training the dataset and the other one is testing dataset with a ratio of 80:20 respectively

Chapter Five

5 MACHINE LEARNING MODELS

The following supervised learning models were applied in this project which are available in sklearn library

1. K-Nearest Neighbors (KNeighbors)
2. Gaussian Naive Bayes (GaussianNB)
3. Decision Trees
4. Random Forest

K-Nearest Neighbors (KNeighbors)

The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. KNN captures the idea of similarity.

Naïve Bayes (NB)

A naive Bayes classifier is a supervised learning algorithm. It is a generative model and is based on joint probability distribution. The Naive Bayes concept based on independence assumptions. It calculates the posterior probability for a dataset using the prior probability and likelihood.

Decision Trees

It looks like an inverted tree. The most important variable is assigned as a root node, each internal node represents a decision, and the leaf node/terminal node has the final class of the outcome. Each branch represents a response (yes/no).

Random Forest

Random forest is an ensemble of decision tree algorithms. It is an extension of bootstrap aggregation (bagging) of decision trees and can be used for classification and regression problems. In bagging, a number of decision trees are created where each tree is created from a different bootstrap sample of the training dataset. A bootstrap sample is a sample of the training dataset where a sample may appear more than once in the sample, referred to as sampling with replacement.

Hyperparameter tuning:

Finding k values for KNN: Often choosing k is tricky, we cannot discard KNN until we've tried different values of k. Hence, a for loop was used to run KNN with K values ranging from 3 to 20 and see if K makes a substantial difference. For k=3 gave a good results and hence, it was used for implementing the KNN

Choosing hyperparameters for decision tree: In order to create decision trees that will generalize to new problem well, we can tune hyperparameters of the tree. Hyperparameters in decision tree: maximum depth, max number of samples to split. This was achieved by using the Grid Search technique in sklearn, which gives the optimized hyperparameters from the list of options provided by the coder.

Chapter Six

6 EVALUATION METHODS

Test Accuracy Score

In classification problem, this function computes subset accuracy: the set of labels predicted for a sample must *exactly* match the corresponding set of labels in y_true.

Mean absolute error

Absolute difference between the actual or true values and the values that are predicted. Absolute difference means that if the result has a negative sign, it is ignored.

Error = True values – Predicted values

MAE takes the average of this error from every sample in a dataset and gives the output.

ROC-AUC

When we need to check or visualize the performance of the multi - class classification problem, we use AUC (Area Under the Curve) and ROC (Receiver Operating Characteristics) curve. ROC is a probability curve and AUC represent degree or measure of separability. It tells how much model is capable of distinguishing between classes.

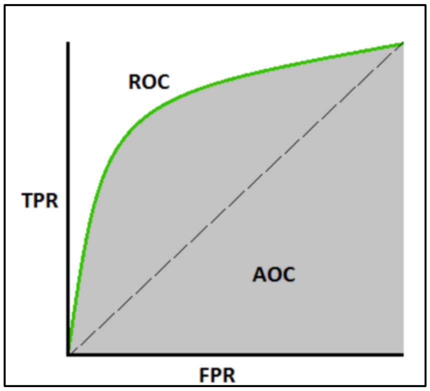
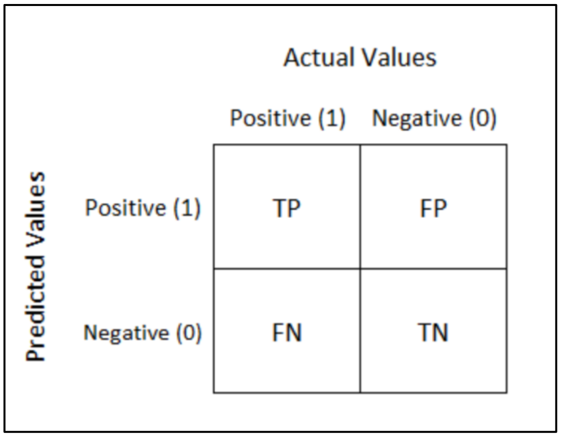


Fig : ROC-AUC

Confusion_matrix

Well, it is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values.



		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Fig: Confusion Matrix

True Positive: You predicted positive and it’s true.

True Negative: You predicted negative and it’s true.

False Positive: (Type 1 Error): You predicted positive and it’s false.

False Negative: (Type 2 Error): You predicted negative and it’s false.

Metrics:

Accuracy is the measures how often the classifier makes the correct prediction. It’s the ratio of the number of correct predictions to the total number of predictions (the number of test data points).

Precision tells us what proportion of messages we classified as spam, actually were spam. It is a ratio of true positives (words classified as spam, and which are actually spam) to all positives(all words classified as spam, irrespective of whether that was the correct classification), in other words it is the ratio of

[True Positives/(True Positives + False Positives)]

Recall (sensitivity) tells us what proportion of messages that actually were spam were classified by us as spam. It is a ratio of true positives(words classified as spam, and which are actually spam) to all the words that were actually spam, in other words it is the ratio of

[True Positives/(True Positives + False Negatives)]

We can use F-beta score as a metric that considers both precision and recall:

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{\left(\beta^2 \cdot precision\right) + recall}$$

In particular, when β=0.5, more emphasis is placed on precision. This is called the **F**$_{0.5}$ score

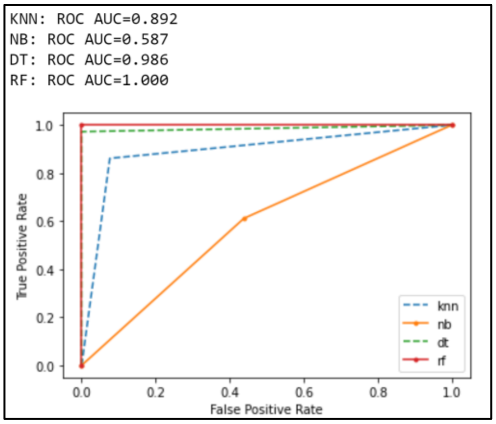
Chapter Seven

7 RESULTS

As the dataset was meant for academic purpose and the features in the dataset that have proved to be effective in detecting the ASD, the results obtained by running the machine learning model turned out to give very good accuracy specially for Decision tree and Random Forest. The table below shows the Metrics results for all four machine learning models. Random forest won over all the four models.

	Accuracy Score	Sensitivity	Specificity	Precision	F-Beta
KNN	0.90	0.88	0.91	0.78	0.80
NB	0.57	0.61	0.56	0.32	0.35
DT	0.95	0.97	0.98	0.97	0.98
RF	0.99	0.98	0.97	0.96	0.98

An important task when performing supervised learning on a dataset like the autistic data we study here is determining which features provide the most predictive power. By focusing on the relationship between only a few crucial features and the target label we simplify our understanding of the phenomenon, which is most always a useful thing to do. In the case of this project, that means we wish to identify a small number of features that most strongly predict whether an individual has ASD or not. Choose a scikit-learn classifier like random forests that has a feature_importance_ attribute, which is a function that ranks the importance of features according to the chosen classifier hence the accuracy is more in RF and DT. The comparison is also shown in the ROC-AUC plot for all 4 models.



```
KNN: ROC AUC=0.892
NB: ROC AUC=0.587
DT: ROC AUC=0.986
RF: ROC AUC=1.000
```

Chapter Eight

8 CONCLUSION

After exploring ASD dataset with different kind of learning algorithms, I can come to a conclusion that all of my model work extremely well with the data. I have used different metric such as accuracy, AUC score and F-score to measure the performance of my models. It looks like all of the metric indicated an almost perfect classification of the ASD cases. Apart from the reasons I mentioned in the first paragraph of the Results section, I also think the reason of this high performances with different models is the fact that only one of the feature is predominant over all others which I have shown and confirmed with the description about features mentioned in the second paragraph of the Result section.

Even though the results are good, we cannot say it is a optimum model as the dataset was very small. So, we need to have access to more larger datasets to build a better and accurate model. However, the models can serve as benchmark models for any machine learning researcher/practitioner who will be interested to explore this dataset further. With this fact in mind, I think this are very well-developed model that can detect ASD in individuals with certain given attributes. The GitHub link for this project is: https://github.com/Archanam5282/Autism-Spectrum-Disorder-Screening-Using-Machine-Learning

REFERENCES

https://www.kaggle.com/faizunnabi/autism-screening

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6579619/

https://csef.usc.edu/History/2018/Projects/J1508.pdf

https://arxiv.org/abs/2009.14499

https://www.psychiatry.org/patients-families/autism/what-is-autism-spectrum-disorder#:~:text=Autism%20spectrum%20disorder%20(ASD)%20is,are%20different%20in%20each%20person.