A PROJECT REPORT

on

"AUTISM SPRECTRUM DISORDER"

Submitted to

KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER SCIENCE & ENGINEERING

BY

ANTRA AMRIT	21052480
SAURABH SUSHANT	21051335
ROSHAN SISODIA	21052611
AASTHA ANAND	21052126
BRIJIT ADAK	21052410

UNDER THE GUIDANCE OF
PROF. ABHAYA KUMAR SAHOO

SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024

May 2022

A PROJECT REPORT

on

"**AUTISM SPRECTRUM DISORDER**"

Submitted to

KIIT Deemed to be University

In Partial Fulfillment of the Requirement for the Award of

BACHELOR'S DEGREE IN
INFORMATION TECHNOLOGY

BY

ANTRA AMRIT	21052480
SAURABH SUSHANT	21051335
ROSHAN SISODIA	21052611
AASTHA ANAND	21052126
BRIJIT ADAK	21052410

UNDER THE GUIDANCE OF
PROF. ABHAYA KUMAR SAHOO

SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAR, ODISHA -751024

May 2022

KIIT Deemed to be University

School of Computer Engineering

Bhubaneswar, ODISHA 751024

CERTIFICATE

This is certify that the project entitled

“AUTISM SPRECTRUM DISORDER”

submitted by

ANTRA AMRIT	21052480
SAURABH SUSHANT	21051335
ROSHAN SISODIA	21052611
AASTHA ANAND	21052126
BRIJIT ADAK	21052410

is a record of bonafide work carried out by them, in the partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2022-2023, under our guidance.

Date: 08/04/2024

(PROF. ABHAYA KUMAR SAHOO)

Project Guide

Acknowledgments

We are profoundly grateful to PROF. ABHAYA KUMAR SAHOO of Affiliation for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. ....

ANTRA AMRIT
SAURABH SUSHANT
ROSHAN SISODIA
AASTHA ANAND
BRIJIT ADAK

ABSTRACT

This documentation provides a comprehensive overview of Autism Spectrum Disorder (ASD), covering its characteristic, diagnosis, and treatment options . Gain a deeper understanding of this complex neurodevelopmental condition.

Autism Spectrum Disorder (ASD) represents a multifaceted neurodevelopmental condition characterized by difficulties in social communication, interaction, and repetitive behaviors. This paper comprehensively reviews the current understanding of ASD, encompassing its epidemiology, potential etiological factors, and available interventions. Recognizing the heterogeneous nature of ASD presentations, emphasis is placed on the significance of early detection and individualized support strategies tailored to the unique needs of each affected individual. Moreover, the discussion advocates for increased awareness and acceptance within society to foster inclusive environments that promote the well-being and integration of individuals with ASD into various social and educational settings.

Contents

List of Figures

Chapter 1

Introduction:

This presentation provides a comprehensive overview of Autism Spectrum Disorder (ASD), covering its characteristic, diagnosis, and treatment options . Gain a deeper understanding of this complex neurodevelopmental condition.

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition characterized by persistent challenges in social interaction, communication, and restricted or repetitive behaviors. First identified by Leo Kanner and Hans Asperger in the early 20th century, ASD has garnered increasing attention in recent decades due to its prevalence and impact on individuals and families worldwide.

While the exact causes of ASD remain elusive, it is widely understood to be a combination of genetic and environmental factors. Research suggests that certain genetic mutations and prenatal factors may contribute to the development of ASD, though the specific mechanisms are still under investigation.

The spectrum nature of ASD means that individuals can experience a wide range of symptoms and levels of impairment. Some may have relatively mild symptoms and be highly functional, while others may require significant support for daily living. This variability underscores the importance of early diagnosis and tailored interventions to address the unique needs of each individual.

Despite the challenges associated with ASD, many individuals with the condition possess unique strengths and talents. With appropriate support and accommodations, they can lead fulfilling lives and make valuable contributions to their communities.

In this paper, we will delve into the various aspects of Autism Spectrum Disorder, including its diagnostic criteria, prevalence, potential causes, and available interventions. By gaining a deeper understanding of ASD, we can work towards promoting greater awareness, acceptance, and support for individuals on the autism spectrum.

Chapter 2

Basic Concepts/ Literature Review

This section contains the basic concepts about the related tools and techniques used in this project. For research work, present the literature review in this section.

2.1 What is ASD?

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition characterized by challenges with social communication and repetitive behaviors.It is a spectrum disorder, meaning its impact varies widely from person to person.

Diagnosis of ASD involves assessment of social communication, repetitive behaviors, and sensory issues. The evaluation is conducted by a multidisciplinary team, including psychologists, speech therapists, and pediatricians.

Autism was described first by Kanner in 1943. It is a developmental disorder commencing in early childhood. Boys are 4 times more likely than girls to be affected.

Individuals with ASD may exhibit challenges in social interactions, restricted interests, and sensory sensitivities. Understanding these characteristics is crucial for providing appropriate support and interventions.

2.2 CAUSE OF AUTISM?

The cause of autism is unknown and no one particular anatomical, biochemical or genetic disorder has been found in those who suffer from it. The problem appears to lie in that part of the brain responsible for the development of language.

2.3 Educational Strategies

Educators play a crucial role in supporting students with ASD. Implementing visual supports, structured routines, and individualized learning plans can create an inclusive and supportive learning environment.

Family Support
Families of individuals with ASD benefit from educational resources, support groups, and access to specialized services. Building a strong support network is essential for navigating the challenges of ASD.

Medical Interventions Medical Interventions
In some cases, medical interventions such as medication for co-occurring conditions and genetic testing may be recommended. These interventions should be carefully considered and tailored to the individual's needs.

Transition to Adulthood Transition to Adulthood
Supporting individuals with ASD during the transition to adulthood involves vocational training, independent living skills, and access to community resources. This phase requires a holistic approach to ensure long-term success.

Advocacy and Awareness Advocacy and Awareness
Advocacy efforts are crucial for promoting inclusion, access to services, and public awareness of ASD. By raising awareness and advocating for policy changes, we can create a more inclusive society for individuals with ASD.

Research and Innovation
Ongoing research is essential for advancing our understanding of ASD and developing innovative interventions. Collaborative efforts in neuroscience, genetics, and behavioral interventions are driving progress in the field.

Chapter 3

Treatment therapies for autism:

Therapy sessions are meetings with patients with a goal to improve some aspects of their life.Initiating it early may help to improve skills and resolve their symptoms.
Some of them are mentioned below:

Socially Assistive Robotics (SAR):
It refers to the robotics that focuses on developing robots designed to interact with humans in emotionally intelligent manner.
They teach and demonstrate socially beneficial behaviors. This tool assists children in expressing themselves in a better way.

Nutrition therapy:
It is typically conducted by registered dietitians.Getting enough vitamins, minerals, and healthy fats from food is really important. A good diet can help manage some of the symptoms of autism and keep them under control.

Speech Therapy:
It is helpful in curing communication and speech issues.
Children can use sign language or images to communicate with others easily.
It enhances the clarity of sound of the patient.

Occupational Therapy:
The primary goal of occupational therapy is to enable individuals to participate in activities that are meaningful to them.It helps them get better at everyday tasks. In therapy, they focus on what each person needs and what they want to achieve.

Chapter 4

Technology used in Autism:

Technology is helpful for people with autism. It helps them learn and communicate better. For example, if Autistic people have very little focus duration, then they can use pictures and sounds to keep them focused. Some of them are as follows:
Gaming:
That is Learning vocabulary using games

Virtual Reality:
Real world is presented to autistic children with help of Virtual Reality.

Augmented Reality:
Facial expressions can be used to detect the symptoms.

Social Robots:
Social robot are used by therapists for communication,thus Helping children with less involvement.

Chapter 5

Machine Learning Algorithms used in detection survey. (ABIDE dataset is used)

Method	Dataset	Algorithms/ Statistical Techniques	Accuracy	Limitations/ Future scope
Resting-state functional magnetic resonance imaging (rs-fMRI)	ABIDE	SVM Logistic regression, ridge	71.40% 71.79% 71.98%.	Integration of ML classifiers with other ASD clinical features for accurate results
Functional magnetic resonance imaging (fMRI).	ABIDE	Linear Discriminant Analysis (LDA) KNN, SVM, LR	77.7% 73.7% 75.5% 76.6%	Use reinforcement and deep learning for better classification model
Clustering of eye-tracking scan path Figshare data repository	59 children data from French school	K-Means clustering	–	–
Remote eye tracking	Participant information	SVM	88.6%	Use of larger dataset to validate algorithm performance
Resting-state functional magnetic resonance imaging (rs-fMRI)	ABIDE	SVM	Male71.6% Female 93.7% as ASD	–
Human computer interaction	50 autistic children	Fuzzy logic	85%	Levels can be increased, more input parameters can be added
Cluster investigation	SKILL database of autism treatment services	K-Means algorithm	–	Incorporating functional component of challenging behaviour
Machine learning Association Rule (AR) with minimum Redundancy-Maximum-Relevance (mRMR)	Autism Therapy Counselling and Help (CATCH)	Association rule with min support and max confidence	83%	Time reduction in symptoms identification and accurate prediction
Screening tool (questionnaires and home videos)	ADI-R & ADOS	Random Forest	–	Use of screening tool for situation beyond autism
Ensemble model	208 ASD subjects Simons Simplex Collection (SSC)	k- dimensional Clustering	–	Use in other disorders with heterogeneity
Speech Transcripts analysis	TalkBank Eigsti & Nadig dataset	Logistic Regression & Random Forest	75%	conversation with Chat bots or robot assistant to be used

Chapter 6

Overview of Machine Learning Algorithms

The machine learning algorithms have been applied widely in diverse aspects of investigating and detecting autism spectrum disorder.
Various data sources such as Autism Spectrum Quotient and Childhood Autism Rating Scale are available based on different age group that can be leveraged to classify different aspects of Autism.

Details of Machine Learning Algorithms

Following classification machine learning algos have been tried for ASD:

I. **Support Vector Machine (SVM):** Support Vector Machine (SVM) is a powerful supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that maximizes the margin between classes, with support vectors defining this boundary. SVM can efficiently handle non-linear data through kernel tricks, but it's sensitive to outliers and requires proper parameter tuning for optimal performance.

II. **Random Forest(RF)**: Random Forest is a versatile ensemble learning method that operates by constructing a multitude of decision trees during training. It excels in both classification and regression tasks, leveraging the wisdom of crowds to make predictions. By averaging or voting on the outputs of multiple trees, it reduces overfitting and enhances generalization performance. Random Forest is robust to noisy data and missing values, and it automatically evaluates feature importance, making it widely applicable in various domains without extensive preprocessing. Random Forest works as follows:

- Choose random N data samples from the training dataset
- Construct the decision tree using chosen data samples
- Select the number of decision trees to be built.
- On arrival of new data sample, Classifier predicts final decision using majority of votes given by decision constructed trees.

III. **Naive Bayes (NB):** Naive Bayes is a simple yet effective probabilistic classifier based on Bayes' theorem with an assumption of independence between features. It's particularly useful for text classification tasks, such as spam detection and sentiment analysis. Despite its simplicity, Naive Bayes often performs well, especially with high-dimensional data, and it's computationally efficient. It works by calculating the probability of a given instance belonging to each class and selecting the class with the highest probability.

Naive Bayes models are robust to irrelevant features and can handle missing data gracefully, making them widely used in various machine learning applications. $P(A|B) = P(B|A) * P(A)/p(B)$.

IV. **Logistic Regression (LR):** Logistic Regression is a widely used statistical method for binary classification, estimating the probability of a binary outcome based on predictor variables. It utilizes the logistic function to map predictions to probabilities, offering simplicity, interpretability, and efficiency in modeling tasks. Despite its linear nature, it's versatile and extends to multiclass classification with suitable techniques.

V. **Decision tree (DT):** Decision tree is a supervised machine learning algorithm used for classification.

VI. **Artificial Neural Network (ANN):** Artificial Neural Networks (ANNs) are versatile machine learning models inspired by the human brain's structure. Comprising interconnected nodes organized in layers, ANNs excel in learning complex patterns from data through forward and backward propagation. They're widely used for tasks like image recognition and natural language processing due to their flexibility and scalability.

Chapter 7

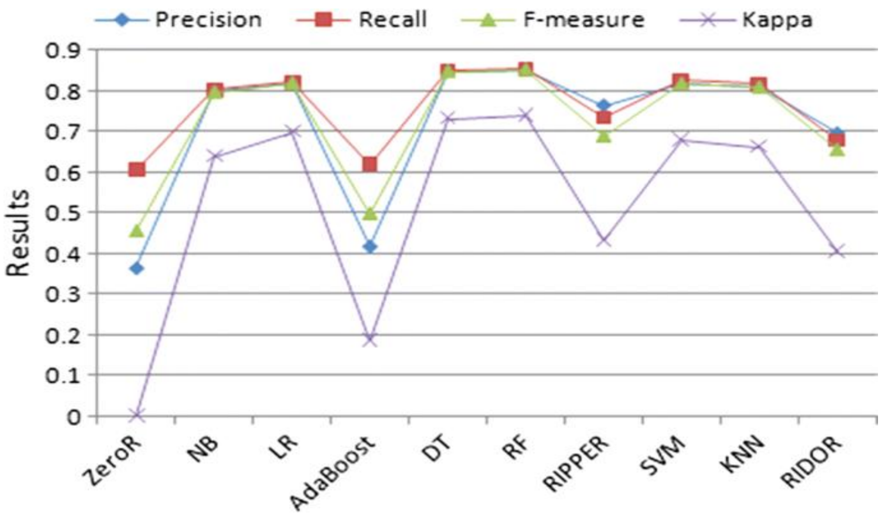
Data processing framework

- **Data sources :**All data sources to have seamless collection of data from existing system without human intervention irrespective of mode of data transfer.
- **Governance and Quality:** All the Data Collected be it Flat file, Hive Tables, SQL Database/Table , Json file etc will be tagged for governance, lineage, attribute dictionary etc to enable to ensure that data sets are tracked, identifiable by business terms, governed and managed. Business users can manage the role driven access to system.
- **Unify and Transform:** Unify all the data uploaded on the basis of auto workflow to ensure no manual intervention. Business users can write new workflow as per their needs. Dynamic data structure will provide flexibility to add more dimension to data in future. To explore the data , transform the data, Aggregate the data etc
- **Feature Engineering:** Leverages data to create new variables that aren't in the training set. It can produce new features for both supervised and unsupervised learning, with the goal of simplifying and speeding up data transformations while also enhancing model accuracy.

Chapter 8

Finally Selected Model

Ranking of the models based on the results of below parameters:

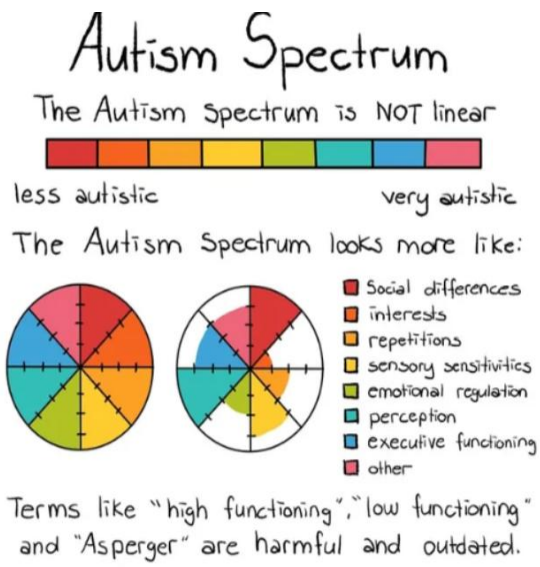


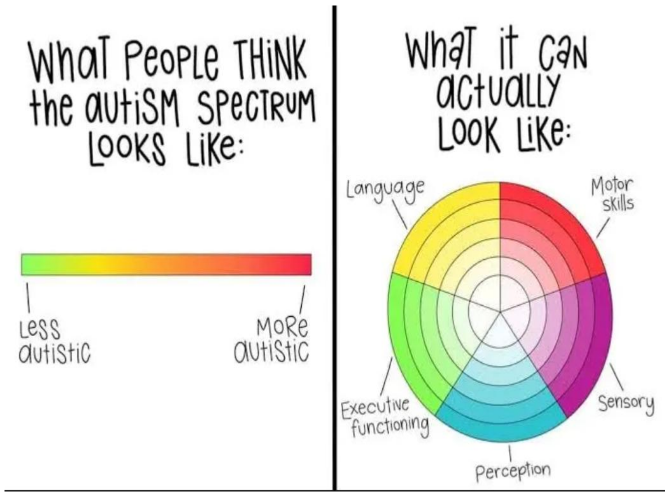
1. Random Forest
2. Decision Tree
3. Logistics regression
4. SVM
5. KNN
6. NB
7. Ripper
8. Ridor
9. AdaBoost
10. ZeroR

It can be concluded that Random Forest and Decision trees are the best models to be selected.

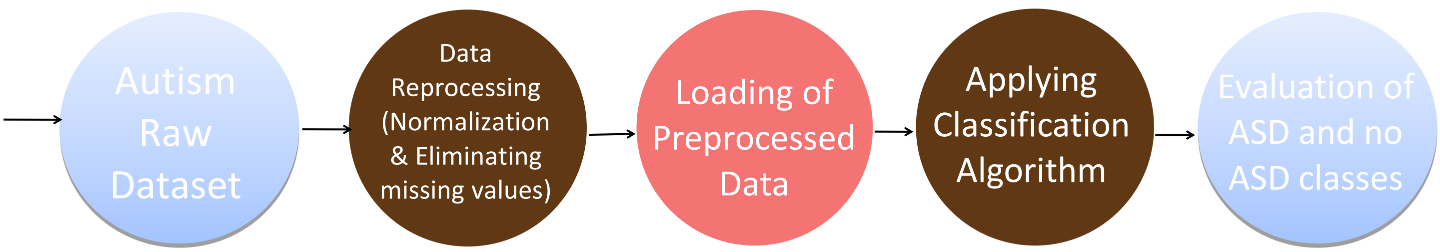
Chapter 9

Diagrams





System Architecture



Chapter 10

Code

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report
from keras.models import Sequential
from keras.layers import Dense
from sklearn.ensemble import RandomForestClassifier
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier

train=pd.read_csv(r"train.csv")
test=pd.read_csv(r"test.csv")
train.head()
```

Out[4]:

	ID	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Sco
0	1	1	0	1	1	1	1	0	1	
1	2	0	0	0	0	0	0	0	0	
2	3	1	1	1	1	1	1	0	0	
3	4	0	0	0	1	0	0	0	0	
4	5	0	0	0	0	1	0	0	0	

5 rows × 22 columns

```
train.describe()
```

Out[4]:

	ID	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score
count	800.0000	800.000000	800.00000	800.000000	800.00000	800.000000	800.00000	800.000000
mean	400.5000	0.582500	0.28625	0.321250	0.41500	0.457500	0.20875	0.273750
std	231.0844	0.493455	0.45229	0.467249	0.49303	0.498502	0.40667	0.446161
min	1.0000	0.000000	0.00000	0.000000	0.00000	0.000000	0.00000	0.000000
25%	200.7500	0.000000	0.00000	0.000000	0.00000	0.000000	0.00000	0.000000
50%	400.5000	1.000000	0.00000	0.000000	0.00000	0.000000	0.00000	0.000000
75%	600.2500	1.000000	1.00000	1.000000	1.00000	1.000000	0.00000	1.000000
max	800.0000	1.000000	1.00000	1.000000	1.00000	1.000000	1.00000	1.000000

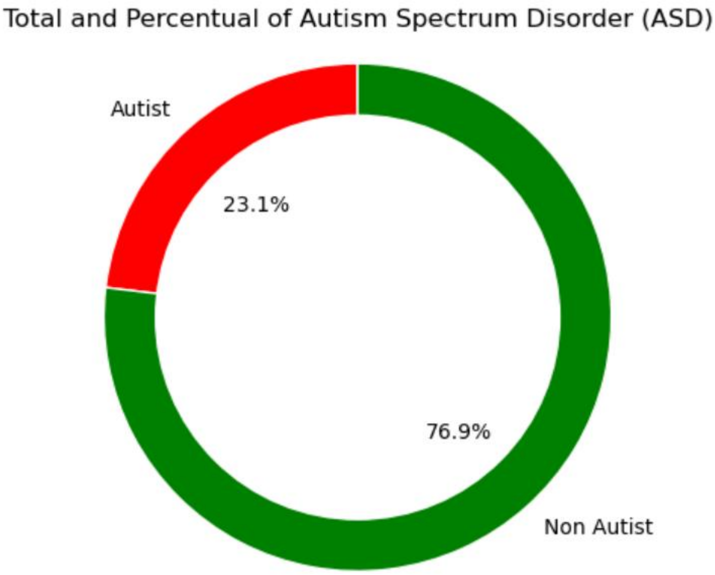
train.isna().sum()

```
Out[5]: ID                 0
        A1_Score           0
        A2_Score           0
        A3_Score           0
        A4_Score           0
        A5_Score           0
        A6_Score           0
        A7_Score           0
        A8_Score           0
        A9_Score           0
        A10_Score          0
        age                0
        gender             0
        ethnicity          0
        jaundice           0
        austim             0
        contry_of_res      0
        used_app_before    0
        result             0
        age_desc           0
        relation           0
        Class/ASD          0
        dtype: int64
```

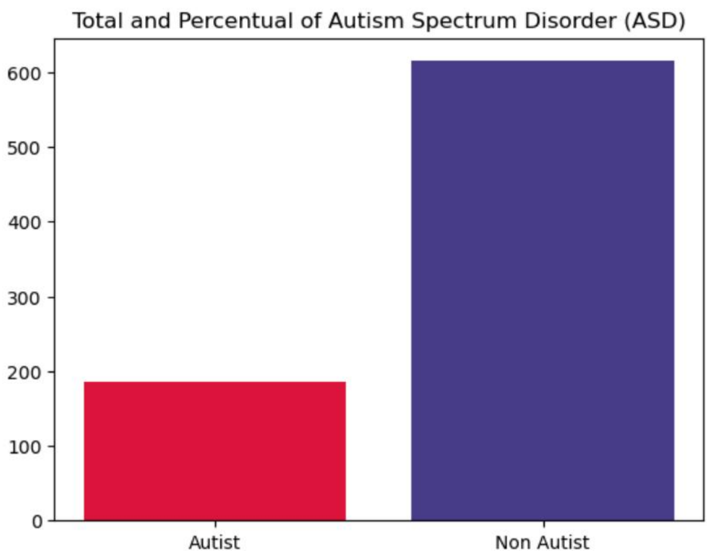
Visualization:

```
autism_map = {1: 'Autist', 0: 'Non Autist'}
autism_colors = ['red', 'green']
total_count = len(train)
autist_count = train['Class/ASD'].sum()
non_autist_count = total_count - autist_count
autist_percent = autist_count / total_count * 100
non_autist_percent = non_autist_count / total_count * 100
fig, ax = plt.subplots()
ax.pie([autist_percent, non_autist_percent], labels=[f'Autist', f'Non Autist'],
 autopct='%1.1f%%', startangle=90, colors=autism_colors, wedgeprops=dict(widt
centre_circle = plt.Circle((0,0),0.80,fc='white')
```

```
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
# Equal aspect ratio ensures that pie is drawn as a circle.
ax.axis('equal')
plt.title('Total and Percentual of Autism Spectrum Disorder (ASD)')
plt.show()
```



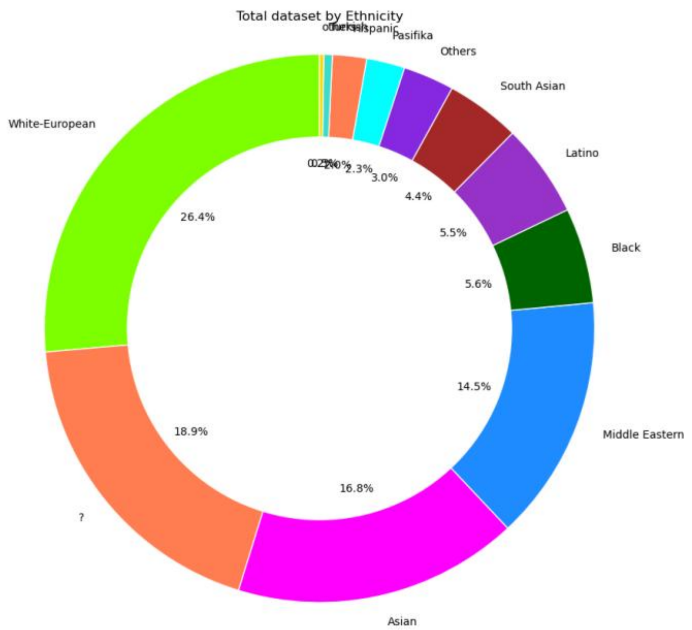

```
autist_count = train['Class/ASD'].sum()
non_autist_count = len(train) - autist_count
fig, ax = plt.subplots()
ax.bar(['Autist', 'Non Autist'], [autist_count, non_autist_count], color=['crimson'
plt.title('Total and Percentual of Autism Spectrum Disorder (ASD)')
plt.show()
```



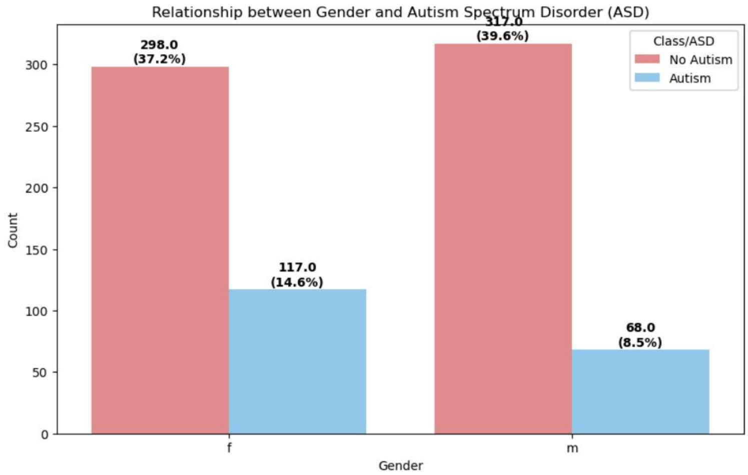

```
train["ethnicity"].value_counts()
```

```
White-European     211
?                  151
Asian              134
Middle Eastern     116
Black               45
Latino              44
South Asian         35
Others              24
Pasifika            18
Hispanic            16
Turkish              4
others               2
Name: ethnicity, dtype: int64
```

```
# Define colors for each ethnicity
ethnicity_colors = ['chartreuse', 'coral', 'magenta', 'dodgerblue', 'darkgreen', 'd
# Plotting the donut chart for ethnicity
fig, ax = plt.subplots(figsize=(10, 10))
# Extract ethnicity counts from the DataFrame
ethnicity_counts = train['ethnicity'].value_counts()
# Draw the donut chart
ax.pie(ethnicity_counts, labels=ethnicity_counts.index, autopct='%1.1f%%', startang
 colors=ethnicity_colors, wedgeprops=dict(width=0.4, edgecolor='w'))
# Draw a circle at the center to make it a donut chart
centre_circle = plt.Circle((0, 0), 0.70, fc='white')
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
# Equal aspect ratio ensures that pie is drawn as a circle.
ax.axis('equal')
plt.title('Total dataset by Ethnicity')
plt.show()
```

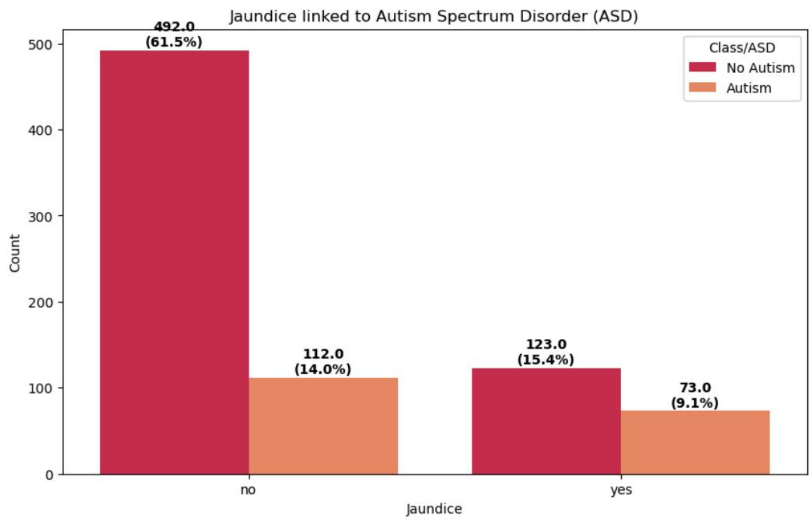


```
# Define colors for each class
class_colors = {0: 'lightskyblue', 1: 'lightcoral'}
# Plotting the relationship between Gender and Class/ASD with count and percentage
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='gender', hue="Class/ASD", data=train, palette=set(class_color
# Change legend labels
ax.legend(title='Class/ASD', labels=['No Autism', 'Autism'])
# Adding annotations with count and percentage
total_count = len(train)
for p in ax.patches:
 height = p.get_height()
 ax.text(p.get_x() + p.get_width() / 2, height + 1, f'{height}\n({height / total
 ha='center', va='bottom', fontweight='bold')
# Setting labels and title
plt.xlabel('Gender')
plt.ylabel('Count')
plt.title('Relationship between Gender and Autism Spectrum Disorder (ASD)')
# Display the plot
plt.show()
```

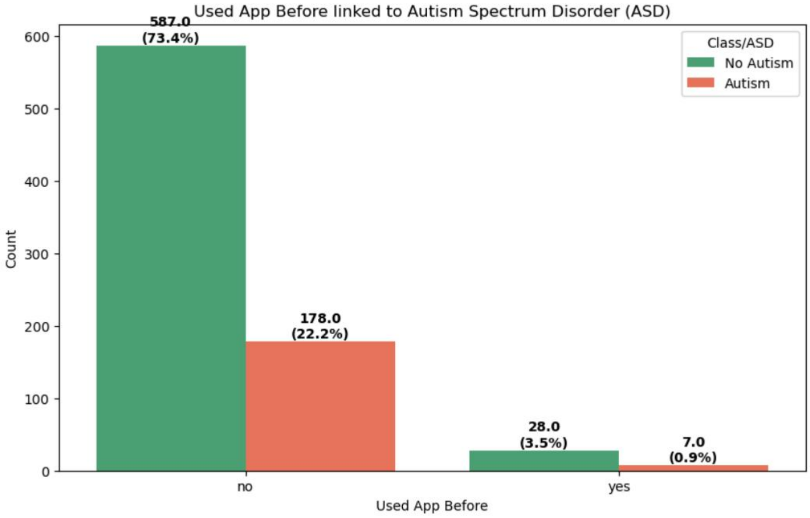



```
# Define colors for each class
jaundice_colors = ['crimson', 'coral']
jaundice_map = {'yes': 'Jaundice', 'no': 'No Jaundice'}
# Plotting the relationship between Jaundice and Class/ASD with count and percentag
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='jaundice', hue='Class/ASD', data=train, palette=jaundice_colo
# Adding annotations with count and percentage
total_count = len(train)
for p in ax.patches:
 height = p.get_height()
 ax.text(p.get_x() + p.get_width() / 2, height + 1, f'{height}\n({height / total
 ha='center', va='bottom', fontweight='bold')
```

```
# Setting labels and title
plt.xlabel('Jaundice')
plt.ylabel('Count')
plt.title('Jaundice linked to Autism Spectrum Disorder (ASD)')
# Change legend labels
ax.legend(title='Class/ASD', labels=['No Autism', 'Autism'])
# Display the plot
plt.show()
```

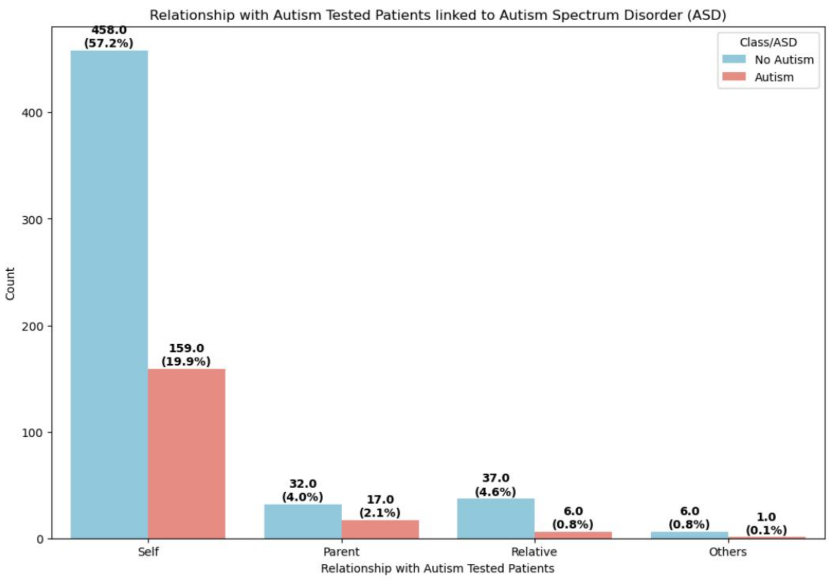



```
# Define colors for each class
used_app_colors = ['mediumseagreen', 'tomato']
used_app_map = {'yes': 'Used App Before', 'no': 'No App Usage'}
# Plotting the relationship between Used App Before and Class/ASD with count and pe
plt.figure(figsize=(10, 6))
ax = sns.countplot(x='used_app_before', hue='Class/ASD', data=train, palette=used_a
# Adding annotations with count and percentage
total_count = len(train)
for p in ax.patches:
 height = p.get_height()
 ax.text(p.get_x() + p.get_width() / 2, height + 1, f'{height}\n({height / total
 ha='center', va='bottom', fontweight='bold')
# Setting labels and title
plt.xlabel('Used App Before')
plt.ylabel('Count')
plt.title('Used App Before linked to Autism Spectrum Disorder (ASD)')
# Change legend labels
ax.legend(title='Class/ASD', labels=['No Autism', 'Autism'])
# Display the plot
plt.show()
```

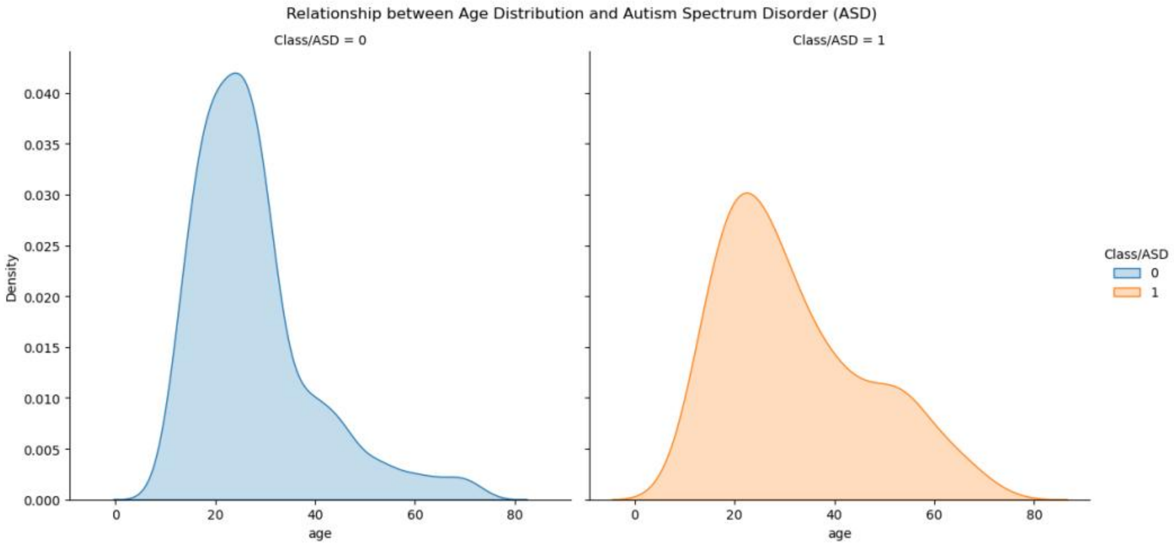


nolo

```
# Define colors for each class
relation_colors = ['skyblue', 'salmon']
relation_map = {'Self': 'Self', 'Parent': 'Parent', 'Relative': 'Relative', 'Others
# Plotting the relationship with Autism Tested Patients and Class/ASD
plt.figure(figsize=(12, 8))
ax = sns.countplot(x='relation', hue='Class/ASD', data=train, palette=relation_colo
# Adding annotations with count and percentage
total_count = len(train)
for p in ax.patches:
 height = p.get_height()
 ax.text(p.get_x() + p.get_width() / 2, height + 1, f'{height}\n({height / total
 ha='center', va='bottom', fontweight='bold')
# Setting labels and title
plt.xlabel('Relationship with Autism Tested Patients')
plt.ylabel('Count')
plt.title('Relationship with Autism Tested Patients linked to Autism Spectrum Disor
# Change legend labels
ax.legend(title='Class/ASD', labels=['No Autism', 'Autism'])
# Display the plot
plt.show()
```



```
# Plotting the relationship between age distribution and Class/ASD
plt.figure(figsize=(12, 8))
g = sns.FacetGrid(train, col='Class/ASD', hue='Class/ASD', height=6)
g.map(sns.kdeplot, 'age', fill=True)
g.add_legend()
# Setting labels and title
plt.subplots_adjust(top=0.9)
g.fig.suptitle('Relationship between Age Distribution and Autism Spectrum Disorder
# Display the plot
plt.show()
```



Preprocessing:

```
cat = {'ethnicity':'category',
 'gender':'category',
 'jaundice':'category',
 'austim':'category',
 'contry_of_res':'category',
 'used_app_before':'category',
 'age_desc':'category',
 'relation':'category'}
test = test.astype(cat)
train = train.astype(cat)

cat_columns = ['ethnicity', 'gender', 'jaundice', 'austim', 'contry_of_res', 'used_
for col in cat_columns:
 train[col] = train[col].cat.codes
 test[col] = test[col].cat.codes

print(train.dtypes)
```

```
ID                    int64
A1_Score              int64
A2_Score              int64
A3_Score              int64
A4_Score              int64
A5_Score              int64
A6_Score              int64
A7_Score              int64
A8_Score              int64
A9_Score              int64
A10_Score             int64
age                 float64
gender                 int8
ethnicity              int8
jaundice               int8
austim                 int8
contry_of_res          int8
used_app_before        int8
result              float64
age_desc               int8
relation               int8
Class/ASD             int64
dtype: object
```

Drop unuseful data

```
train=train.drop('ID', axis=1)
train=train.drop('relation', axis=1)

test=test.drop('ID', axis=1)
test=test.drop('relation', axis=1)
```

Split Data

```
X = train.drop('Class/ASD', axis=1)
y = train['Class/ASD']
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state=42)
```

Standard Scaler

```
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_val = scaler.transform(X_val)
X_test = scaler.fit_transform(test)
```

Classification

10.1.Random Forest

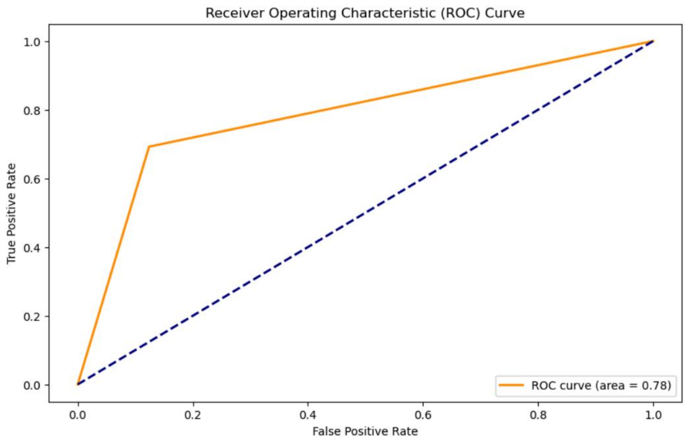
```
classifier = RandomForestClassifier()

# Train the classifier on the training data
classifier.fit(X_train, y_train)

# Make predictions on the test data
y_pred = classifier.predict(X_val)

# We can predict for the test dataset given using below code
# y_pred = classifier.predict(X_test)

# Confusion Matrix
conf_matrix = confusion_matrix(y_val, y_pred)
print("Confusion Matrix:")
print(conf_matrix)

# Accuracy
accuracy_rf = accuracy_score(y_val, y_pred)
print("Accuracy:", accuracy_rf)

# Classification Report
class_report = classification_report(y_val, y_pred)
print("Classification Report:")
print(class_report)
```

```
Confusion Matrix:
[[106  15]
 [ 12  27]]
Accuracy: 0.83125
Classification Report:
              precision    recall  f1-score   support

           0       0.90      0.88      0.89       121
           1       0.64      0.69      0.67        39

    accuracy                           0.83       160
   macro avg       0.77      0.78      0.78       160
weighted avg       0.84      0.83      0.83       160
```

ıpport as score
y_pred)

```
# ROC Curve
fpr, tpr, thresholds = roc_curve(y_val, y_pred)
plt.figure(figsize=(10, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area = %0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc="lower right")
plt.show()
```

ROC-AUC Score: 0.784170375079466

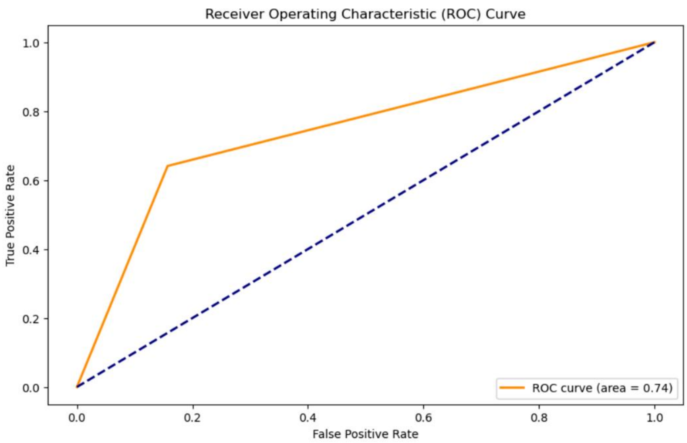


10.2.Decision Tree

```
dtree = DecisionTreeClassifier(criterion="gini")

# Train the classifier on the training data
dtree.fit(X_train, y_train)

# Make predictions on the test data
y_pred = dtree.predict(X_val)

# We can predict for the test dataset given using below code
# y_pred = dtree.predict(X_test)

conf_matrix = confusion_matrix(y_val, y_pred)
print("Confusion Matrix:")
print(conf_matrix)

# Accuracy
accuracy_dt = accuracy_score(y_val, y_pred)
print("Accuracy:", accuracy_dt)

# Classification Report
class_report = classification_report(y_val, y_pred)
print("Classification Report:")
print(class_report)
```

```
Confusion Matrix:
[[102  19]
 [ 14  25]]
Accuracy: 0.79375
Classification Report:
              precision    recall  f1-score   support

           0       0.88      0.84      0.86       121
           1       0.57      0.64      0.60        39

    accuracy                           0.79       160
   macro avg       0.72      0.74      0.73       160
weighted avg       0.80      0.79      0.80       160
```

```
from sklearn.metrics import precision_recall_fscore_support as score
precision_dt,recall_dt,fscore_dt,support_dt=score(y_val, y_pred)

# ROC-AUC Score
roc_auc = roc_auc_score(y_val, y_pred)
print("ROC-AUC Score:", roc_auc)

# ROC Curve
fpr, tpr, thresholds = roc_curve(y_val, y_pred)
plt.figure(figsize=(10, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area = %0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc="lower right")
plt.show()
```



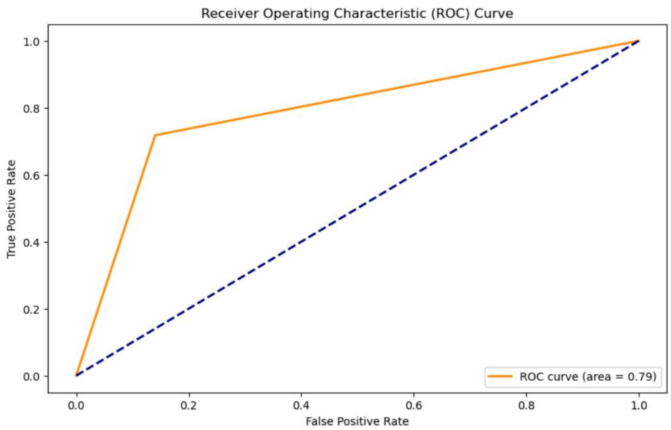
10.3.ANN model

```
classifier = Sequential()

# Adding the input layer and the first hidden layer
classifier.add(Dense(units=20, activation='relu', input_dim=X_train.shape[1]))

# Adding the second hidden layer
classifier.add(Dense(units=15, activation='relu'))

# Adding the output layer
classifier.add(Dense(units=1, activation='sigmoid'))

# Compile the ANN
classifier.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

# Train the ANN on the training data
classifier.fit(X_train, y_train, batch_size=32, epochs=50)
```

```
# Make predictions on the validation data
y_predict = classifier.predict(X_val)
y_predict = (y_predict > 0.5)

# We can predict for the test dataset given using below code
# y_predict = classifier.predict(X_test)

Epoch 1/50
20/20 [==============================] - 1s 2ms/step - loss: 0.6286 - accuracy: 0.6344
Epoch 2/50
20/20 [==============================] - 0s 2ms/step - loss: 0.4911 - accuracy: 0.8047
Epoch 3/50
20/20 [==============================] - 0s 2ms/step - loss: 0.4120 - accuracy: 0.8453
Epoch 4/50
20/20 [==============================] - 0s 2ms/step - loss: 0.3655 - accuracy: 0.8672
Epoch 5/50
20/20 [==============================] - 0s 2ms/step - loss: 0.3372 - accuracy: 0.8813
Epoch 6/50
20/20 [==============================] - 0s 2ms/step - loss: 0.3170 - accuracy: 0.8859
Epoch 7/50
20/20 [==============================] - 0s 2ms/step - loss: 0.3051 - accuracy: 0.8906
Epoch 8/50
20/20 [==============================] - 0s 2ms/step - loss: 0.2966 - accuracy: 0.8891
Epoch 9/50
20/20 [==============================] - 0s 2ms/step - loss: 0.2900 - accuracy: 0.8891
Epoch 10/50
20/20 [==============================] - 0s 2ms/step - loss: 0.2849 - accuracy: 0.8938
Epoch 11/50
20/20 [==============================] - 0s 2ms/step - loss: 0.2803 - accuracy: 0.8922
Epoch 12/50
20/20 [==============================] - 0s 2ms/step - loss: 0.2764 - accuracy: 0.8922
Epoch 13/50
...
20/20 [==============================] - 0s 2ms/step - loss: 0.2006 - accuracy: 0.9156
Epoch 50/50
20/20 [==============================] - 0s 2ms/step - loss: 0.1975 - accuracy: 0.9172
5/5 [==============================] - 0s 2ms/step

# Evaluate the model
conf_matrix = confusion_matrix(y_val, y_predict)
print("Confusion Matrix:")
print(conf_matrix)

accuracy_ANN = accuracy_score(y_val, y_predict)
print("Accuracy:", accuracy_ANN)

class_report = classification_report(y_val, y_predict)
print("Classification Report:")
print(class_report)
```

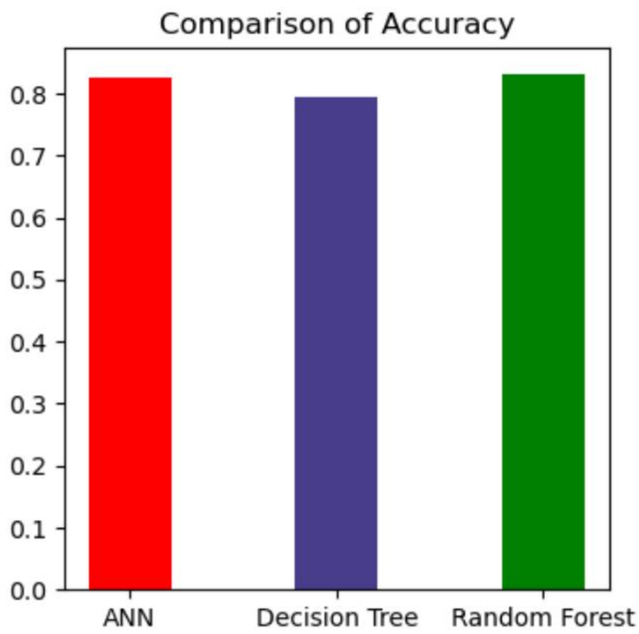
```
Confusion Matrix:
[[104  17]
 [ 11  28]]
Accuracy: 0.825
Classification Report:
              precision    recall  f1-score   support

           0       0.90      0.86      0.88       121
           1       0.62      0.72      0.67        39

    accuracy                           0.82       160
   macro avg       0.76      0.79      0.77       160
weighted avg       0.84      0.82      0.83       160
```

```
from sklearn.metrics import precision_recall_fscore_support as score
precision_ANN,recall_ANN,fscore_ANN,support_ANN=score(y_val, y_predict)

roc_auc = roc_auc_score(y_val, y_predict)
print("ROC-AUC Score:", roc_auc)

# ROC Curve
fpr, tpr, thresholds = roc_curve(y_val, y_predict)
plt.figure(figsize=(10, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area = %0.2f)' % roc_auc)
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc="lower right")
plt.show()
```

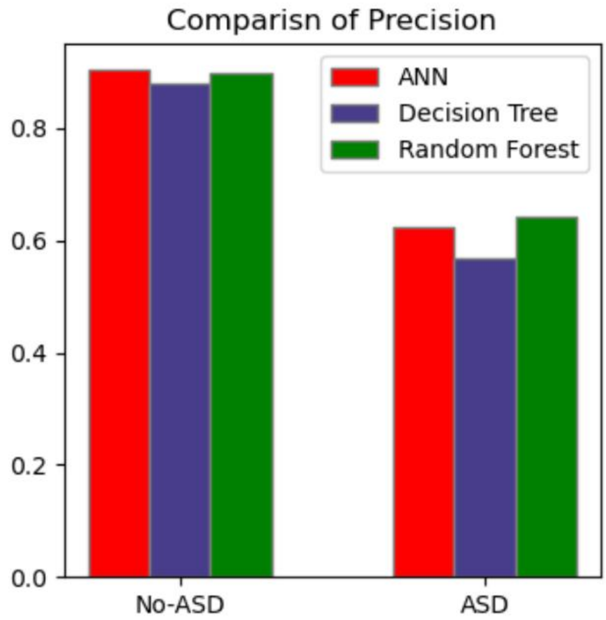


Comparison of ANN, Decision Tree and Randoom Forest

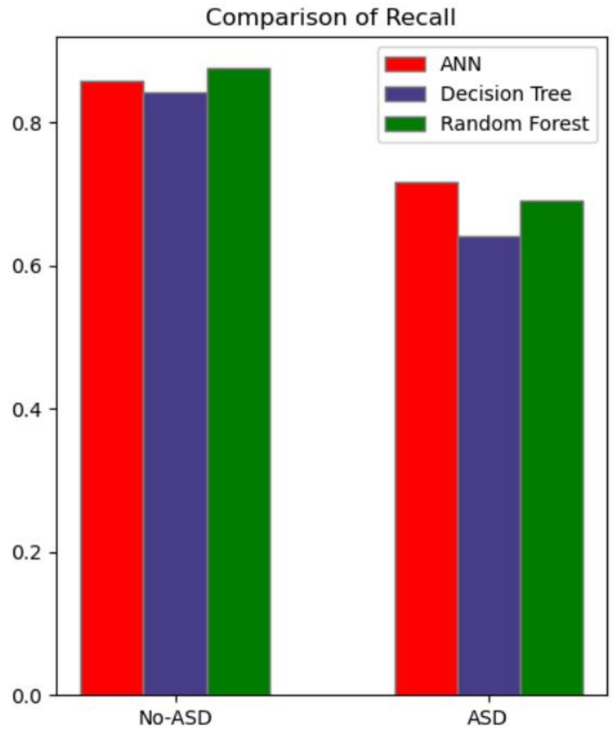
```
#Comparison of Accuracy
fig, ax2 = plt.subplots(figsize = (4, 4))
ax2.bar(['ANN', 'Decision Tree', 'Random Forest'], [accuracy_ANN, accuracy_dt, accuracy_rf],
color=['red', 'darkslateblue', 'green'], width = 0.4)

plt.title('Comparison of Accuracy')
plt.show()
```



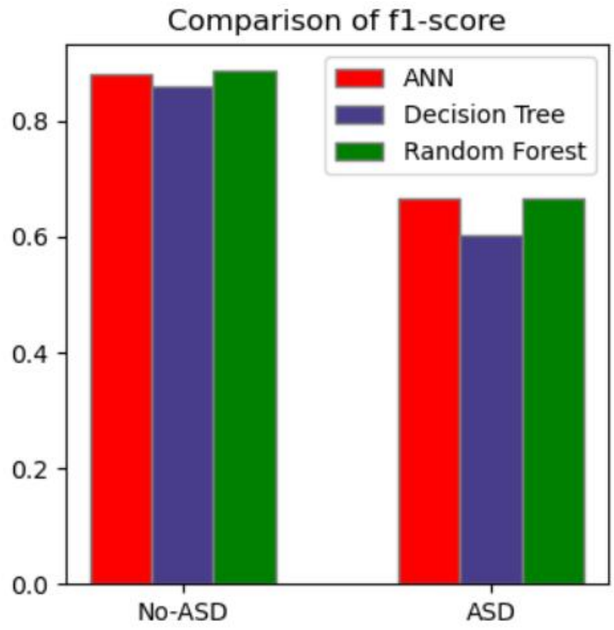

```
#Comparison of Precision
barWidth = 0.2
fig = plt.subplots(figsize =(4, 4))

# Set position of bar on X axis
br1 = np.arange(len(precision_ANN))
br2 = [x + barWidth for x in br1]
br3 = [x + barWidth for x in br2]

# Make the plot
plt.bar(br1, precision_ANN, color ='r', width = barWidth,
        edgecolor ='grey', label ='ANN')
plt.bar(br2, precision_dt, color ='darkslateblue', width = barWidth,
        edgecolor ='grey', label ='Decision Tree')
plt.bar(br3, precision_rf, color ='g', width = barWidth,
        edgecolor ='grey', label ='Random Forest')

# Adding Xticks
plt.title('Comparisn of Precision')
plt.xticks([r + barWidth for r in range(len(precision_ANN))], ['No-ASD', 'ASD'])
plt.legend()
plt.show()
```



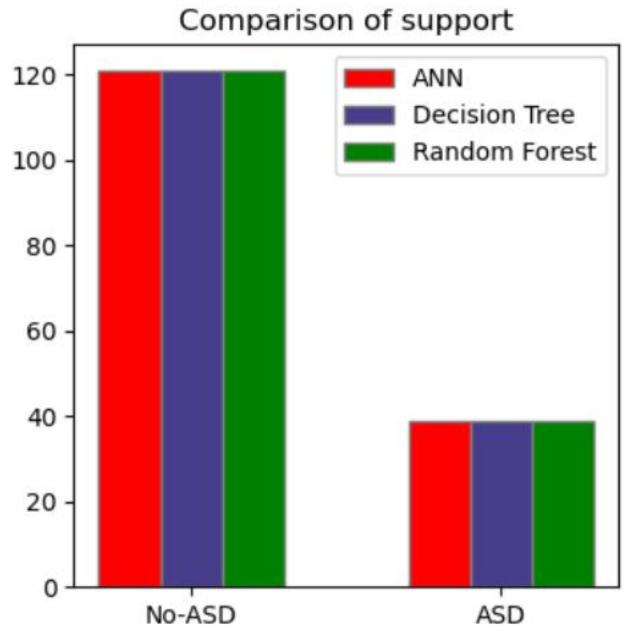

```
#Comparison of Recall
barWidth = 0.2
fig = plt.subplots(figsize =(5, 6))

# Set position of bar on X axis
br1 = np.arange(len(recall_ANN))
br2 = [x + barWidth for x in br1]
br3 = [x + barWidth for x in br2]

# Make the plot
plt.bar(br1, recall_ANN, color ='r', width = barWidth,
        edgecolor ='grey', label ='ANN')
plt.bar(br2, recall_dt, color ='darkslateblue', width = barWidth,
        edgecolor ='grey', label ='Decision Tree')
plt.bar(br3, recall_rf, color ='g', width = barWidth,
        edgecolor ='grey', label ='Random Forest')

# Adding Xticks
plt.title('Comparison of Recall')
plt.xticks([r + barWidth for r in range(len(precision_ANN))], ['No-ASD', 'ASD'])

plt.legend()
plt.show()
```




```
#Comparison of fscore
barWidth = 0.2
fig = plt.subplots(figsize =(4, 4))

# Set position of bar on X axis
br1 = np.arange(len(fscore_ANN))
br2 = [x + barWidth for x in br1]
br3 = [x + barWidth for x in br2]

# Make the plot
plt.bar(br1, fscore_ANN, color ='r', width = barWidth,
        edgecolor ='grey', label ='ANN')
plt.bar(br2, fscore_dt, color ='darkslateblue', width = barWidth,
        edgecolor ='grey', label ='Decision Tree')
plt.bar(br3, fscore_rf, color ='g', width = barWidth,
        edgecolor ='grey', label ='Random Forest')

# Adding Xticks
plt.title('Comparison of f1-score')
plt.xticks([r + barWidth for r in range(len(precision_ANN))], ['No-ASD', 'ASD'])

plt.legend()
plt.show()
```




```
#Comparison of support
barWidth = 0.2
fig = plt.subplots(figsize =(4, 4))

# Set position of bar on X axis
br1 = np.arange(len(support_ANN))
br2 = [x + barWidth for x in br1]
br3 = [x + barWidth for x in br2]

# Make the plot
plt.bar(br1, support_ANN, color ='r', width = barWidth,
        edgecolor ='grey', label ='ANN')
plt.bar(br2, support_dt, color ='darkslateblue', width = barWidth,
        edgecolor ='grey', label ='Decision Tree')
plt.bar(br3, support_rf, color ='g', width = barWidth,
        edgecolor ='grey', label ='Random Forest')

# Adding Xticks
plt.title('Comparison of support')
plt.xticks([r + barWidth for r in range(len(precision_ANN))], ['No-ASD', 'ASD'])

plt.legend()
plt.show()
```


Comparison of support
ANN
Decision Tree
Random Forest
120
100
80
60
40
20
0
No-ASD
ASD

Chapter 11

Result comparison with different Algos

Below are the charts of all the algorithms based on different parameters:

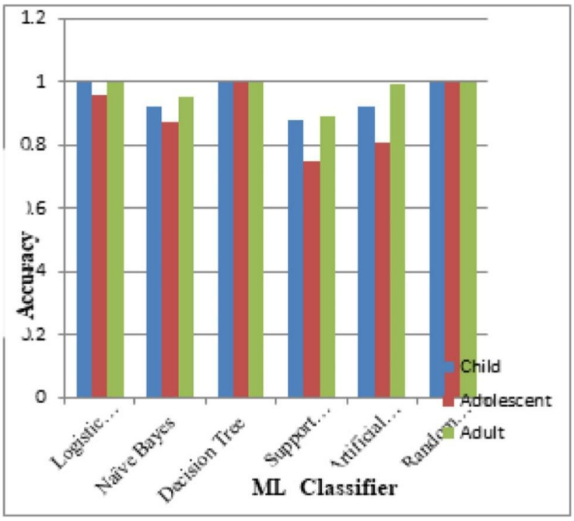


Fig. 5. Accuracy graph of classifiers.

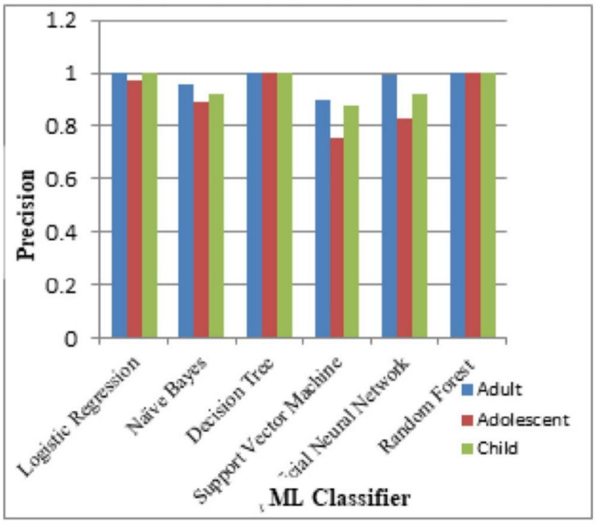


Fig. 6. Precision graph of classifiers.

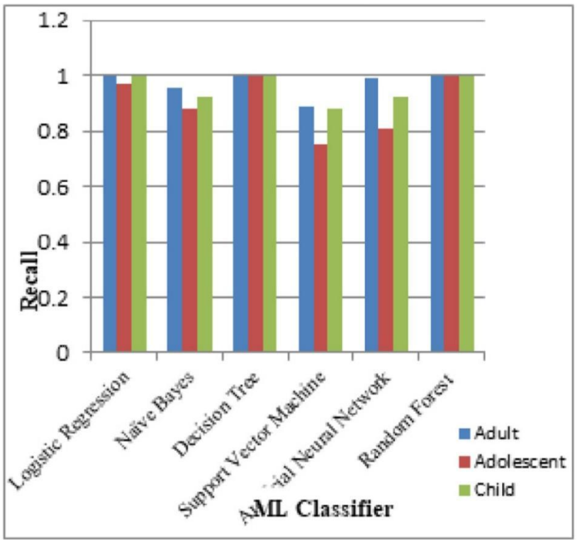


Fig. 7. Recall graph of classifiers.

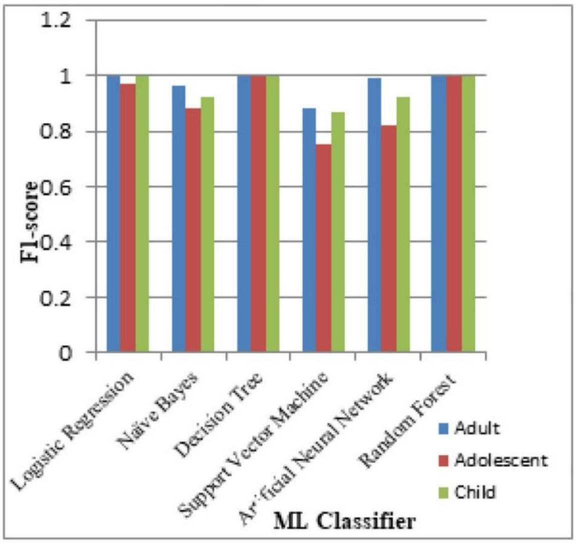


Fig. 8. F1-score graph of classifiers.

Chapter 12

12.1 Design Standards

The dataset which we are using in research purpose is from UCI Machine Learning Repository made available by Fadi Fayez Thabtah.The Attributes used for our model training are mentioned below:

i). A1_Score to A10_Score - Score based on Autism Spectrum Quotient (AQ) 10 item screening tool
ii). age - Age of the patient in years
iii). gender - Gender of the patient
iv). ethnicity - Ethnicity of the patient
v). jaundice - Whether the patient had jaundice at the time of birth
vi). autism - Whether an immediate family member has been diagnosed with autism
vii). country_of_res - Country of residence of the patient
viii) .used_app_before - Whether the patient has undergone a screening test before
ix). result - Score for AQ1-10 screening test
x). age_desc - Age of the patient
xi). Class/ASD - Classified result as 0 or 1. Here 0 represents No and 1 represents Yes. This is the target column, and during submission submit the values as 0 or 1 only.

12.2 Coding Standards

Followings are the attributes that are taken to train our model:

- Is able to hear sound that others can't.
- Targeting on broad things than small details.
- Pursue the discussions of people in a community group.
- If can effectively switch between available actions.
- Having no idea how to chitchat with peers.
- Fine for day-today small chats.
- Finding hard to understand emotions while reading a book.
- Fond of playing role plays as part of pre-school education.
- Recognize their experience by observing facial expressions.
- Hard to make new friends.
- Background('White-European', 'South Asian','Asian', 'Middle Eastern 'Pasifika','Hispanic', 'Turkish','Latino', 'Black','Others', 'Unknown').
- Born with jaundice .
- Family member with PDD.
- Country.
- Familiar with the screening app .
- Score.
- Age_desc.
- If attempted the test before.
- Class/ASD.

12.3 Testing Standards

i. Accuracy=(TP+TN)/(TP+TN+FP+FN)
ii. Precision= TP/(TP+FP)
iii. Recall = TP/(TP+FN)
iv. F1-Score =2(Recall*Precision)/(Recall+Precision)

1. **Random Forest:**

Classification Report:

	precision	recall	f1-score
0	0.91	0.88	0.90
1	0.67	0.72	0.69
accuracy			0.84

2. **Artificial neural network:**

Classification Report:

	precision	recall	f1-score
0	0.90	0.85	0.88
1	0.61	0.72	0.66
accuracy			0.82

3. **Decision Tree:**

Classification Report:

	precision	recall	f1-score
0	0.88	0.84	0.86
1	0.57	0.64	0.60
accuracy			0.79

Chapter 13

Conclusion and Future Scope

Conclusion

In conclusion, autism spectrum disorder (ASD) is a complex neurodevelopmental condition that affects individuals in diverse ways. Through this presentation, we've explored the key characteristics, diagnostic criteria, prevalence, and potential causes of autism. It's crucial to recognize that each person with autism is unique, with their own strengths and challenges.

While there is still much to learn about autism, increased awareness, acceptance, and support are essential for improving the lives of individuals on the spectrum. Early intervention and personalized therapies can significantly enhance outcomes and help individuals with autism reach their full potential.

As we continue to advance our understanding of autism, let us strive for a society that embraces neurodiversity, promotes inclusivity, and provides equal opportunities for individuals of all abilities. Together, we can create a more compassionate and supportive world for everyone, including those with autism.

Future Scope

As we move forward, the future holds promising developments in the understanding, diagnosis, and treatment of autism spectrum disorder (ASD). Here are some areas of future scope in the field:

Early Detection and Intervention: Advances in neuroimaging, genetic testing, and behavioral assessments may lead to earlier detection of autism, allowing for timely intervention and improved outcomes. Research into early intervention strategies, such as targeted therapies and parent training programs, shows promise in maximizing developmental potential.

Personalized Medicine: The shift towards personalized medicine may revolutionize autism treatment by tailoring interventions to the individual's unique genetic, biological, and behavioral profile. Precision medicine approaches, including pharmacogenomics and targeted therapies, hold potential for optimizing treatment efficacy and minimizing side effects.

Neuroscience and Brain-Computer Interfaces: Continued research into the neurobiology of autism may uncover novel insights into the underlying neural mechanisms of the disorder. Advancements in neuroimaging techniques, such as functional magnetic resonance imaging (fMRI) and electroencephalography (EEG), could facilitate the development of brain-computer interfaces (BCIs) for augmentative communication and assistive technologies.

Artificial Intelligence and Data Analytics: The integration of artificial intelligence (AI) and machine learning algorithms may enhance our ability to analyze large-scale datasets, identify patterns, and predict outcomes in autism research. AI-powered tools for early screening, diagnostic decision support, and personalized treatment planning have the potential to revolutionize clinical practice and improve patient care.

Digital Health Solutions: The proliferation of mobile health technologies and wearable devices presents opportunities for remote monitoring, real-time data collection, and personalized intervention delivery in autism management. Virtual reality (VR) and augmented reality (AR) applications may also offer innovative approaches for social skills training, sensory integration therapy, and behavior modification.

Community Support and Advocacy: The growing emphasis on community-based support services, inclusive education practices, and employment opportunities for individuals with autism reflects a broader societal shift towards promoting neurodiversity and social inclusion. Advocacy efforts aimed at raising awareness, reducing stigma, and fostering acceptance are essential for creating a more inclusive society.

By embracing interdisciplinary collaboration, leveraging technological innovations, and prioritizing the needs and voices of individuals with autism and their families, we can work towards a future where every person on the autism spectrum is empowered to thrive and fulfill their potential.

References

- American Psychiatric Association. (2013). Diagnostic and Statistical Manual of Mental Disorders (5th ed.). Arlington, VA: American Psychiatric Publishing.
- Baio, J., Wiggins, L., Christensen, D. L., et al. (2018). Prevalence of Autism Spectrum Disorder Among Children Aged 8 Years — Autism and Developmental Disabilities Monitoring Network, 11 Sites, United States, 2014. MMWR. Surveillance Summaries, 67(6), 1–23.
- Happé, F., & Ronald, A. (2008). The 'fractionable autism triad': a review of evidence from behavioural, genetic, cognitive and neural research. Neuropsychology Review, 18(4), 287–304.
- Lai, M. C., Lombardo, M. V., & Baron-Cohen, S. (2014). Autism. The Lancet, 383(9920), 896–910.
- National Institute of Mental Health. (2019). Autism Spectrum Disorder. Retrieved from https://www.nimh.nih.gov/health/topics/autism-spectrum-disorders-asd/index.shtml
- Volkmar, F. R., & McPartland, J. C. (2014). From Kanner to DSM-5: Autism as an Evolving Diagnostic Concept. Annual Review of Clinical Psychology, 10, 193–212.
- World Health Organization. (2018). Autism Spectrum Disorders. Retrieved from https://www.who.int/news-room/fact-sheets/detail/autism-spectrum-disorders

Individual Contribution

BRIJIT ADAK -> Lead and Coding
ANTRA AMRIT -> Planning and design
SAURABH SUSHANT -> Coding and Documentation
ROSHAN SISODIA -> Planning and design
AASTHA ANAND -> Research and Documentation

AUTISM SPRECTRUM DISORDER

ORIGINALITY REPORT

46%	37%	21%	38%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to University of South Australia Student Paper	4%
2	Anita Vikram Shinde, Dipti Durgesh Patil. "A Multi-Classifier-Based Recommender System for Early Autism Spectrum Disorder Detection using Machine Learning", Healthcare Analytics, 2023 Publication	4%
3	www.coursehero.com Internet Source	4%
4	Submitted to University of North Texas Student Paper	3%
5	Submitted to KIIT University Student Paper	3%
6	Submitted to University of Western Ontario Student Paper	3%
7	Submitted to Toronto Business College Student Paper	2%
8	Submitted to University of Auckland Student Paper	

		2%
9	fastercapital.com Internet Source	2%
10	Submitted to University of Salford Student Paper	1%
11	www.autism-resources.org Internet Source	1%
12	www.betterfamilyhealth.org Internet Source	1%
13	codingwithalex.com Internet Source	1%
14	Submitted to The University of the West of Scotland Student Paper	1%
15	blog.usejournal.com Internet Source	1%
16	Submitted to Queen Mary and Westfield College Student Paper	1%
17	Pramod Gupta, Anupam Bagchi. "Chapter 8 Machine Learning", Springer Science and Business Media LLC, 2024 Publication	1%
18	www.kaggle.com	

	Internet Source	1 %
19	www.geeksforgeeks.org Internet Source	1 %
20	Submitted to BPP College of Professional Studies Limited Student Paper	1 %
21	www.dremio.com Internet Source	1 %
22	deepnote.com Internet Source	1 %
23	Submitted to Texas A&M University, San Antonio Student Paper	1 %
24	Submitted to Coventry University Student Paper	1 %
25	Submitted to King's College Student Paper	1 %
26	Submitted to St. Petersburg College Student Paper	1 %
27	Submitted to Kean University Student Paper	<1 %
28	Submitted to Liberty University Student Paper	<1 %

29	Alexandros Laios, Evangelos Kalampokis, Marios Evangelos Mamalis, Constantine Tarabanis et al. "RoBERTa-Assisted Outcome Prediction in Ovarian Cancer Cytoreductive Surgery Using Operative Notes", Cancer Control, 2023 Publication	<1%
30	Submitted to Brunel University Student Paper	<1%
31	Submitted to Yavapai College Student Paper	<1%
32	ijrpr.com Internet Source	<1%
33	Submitted to Federation University Student Paper	<1%
34	datascience.stackexchange.com Internet Source	<1%
35	Submitted to SASTRA University Student Paper	<1%
36	www.mdpi.com Internet Source	<1%
37	Submitted to Queen's University of Belfast Student Paper	<1%
38	rhqff.gch2020.eu Internet Source	<1%

39	xavier.org.au Internet Source	<1 %
40	www.hilarispublisher.com Internet Source	<1 %
41	www.intechopen.com Internet Source	<1 %
42	Submitted to AUT University Student Paper	<1 %
43	Submitted to Griffth University Student Paper	<1 %
44	Submitted to Liverpool John Moores University Student Paper	<1 %
45	Submitted to Two Oceans Graduate Institute Student Paper	<1 %
46	docshare.tips Internet Source	<1 %
47	www.putchildrenfirst.org Internet Source	<1 %
48	Submitted to University of Dundee Student Paper	<1 %
49	gist.github.com Internet Source	<1 %
50	huggingface.co Internet Source	

<1%

51	towardsdatascience.com Internet Source	<1%

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On		