Detection Of Autistic Spectrum Disorder: Classification

Team Members:

Vaibhav Yadav 22BBS0030

Tanav Kolar 22BCE2862

Marut Kashyap 22BBS0085

Rohit Dubey 22BBS0092

Introduction

Project overview:

Autism, or autism spectrum disorder (ASD), refers to a broad range of conditions characterized by challenges with social skills, repetitive behaviours, speech and nonverbal communication.

Causes and Challenges:

It is mostly influenced by a combination of genetic and environmental factors. Because autism is a spectrum disorder, each person with autism has a distinct set of strengths and challenges. The ways in which people with autism learn, think and problem-solve can range from highly skilled to severely challenged.
Research has made clear that high quality early intervention can improve learning, communication and social skills, as well as underlying brain development. Yet the diagnostic process can take several years.

The Role of Machine Learning:

This dataset is composed of survey results for more than 700 people who filled an app form. There are labels portraying whether the person received a diagnosis of autism, allowing machine learning models to predict the likelihood of having autism, therefore allowing healthcare professionals prioritize their resources.

Objective: To develop and test machine learning models to predict the likelihood of having autism based on the given features.

Project Initialization and Planning Phase

Problem Statement:

Autism spectrum disorder (ASD) is a chronic condition that will impact a person's behaviour and how he socializes with others. ASD appears in the early childhood but unfortunately most children diagnosed with ASD until school. Early diagnosis of ASD is significant for a family and also for the children.

ASD is associated with significant healthcare costs, and early diagnosis can significantly reduce these. Unfortunately, waiting times for an ASD diagnosis are lengthy and procedures are not cost effective. The economic impact of autism and the increase in the number of ASD cases across the world reveals an urgent need for the development of easily implemented and effective screening methods.

Therefore, a time- efficient and accessible ASD screening is imminent to help health professionals and inform individuals whether they should pursue formal clinical diagnosis. The rapid growth in the number of ASD cases worldwide necessitates datasets related to behaviour traits. However, such datasets are rare making it difficult to perform thorough analyses to improve the efficiency, sensitivity, specificity, and predictive accuracy of the ASD screening process.

Project Proposal (Proposed Solution):

Presently, very limited autism datasets associated with clinical, or screening are available and most of them are genetic in nature. Hence, we propose a new dataset related to autism screening of adults that contain 20 features to be utilized for further analysis especially in determining influential autistic traits and improving the classification of ASD cases. In this dataset, we record ten behavioural features (A1-A10-Adult) plus ten individual characteristics that have proved to be effective in detecting the ASD cases from controls in behaviour science.

In this study area, we would like to know how individual characteristics have influence on ASD detection and whether the given individual characteristics are able to effectively predict the ASD cases..

The proposal report aims to revolutionize the early detection of Autism Spectrum Disorder (ASD) using machine learning, enhancing diagnostic efficiency and accuracy. It addresses the current system's limitations, promising faster diagnosis, reduced costs, and improved patient outcomes. Key features include a machine learning-based screening tool and a user-friendly interface for real-time predictions.

Approach	"The project will utilize machine learning techniques to analyse a dataset containing behavioural and individual characteristics associated with ASD. The steps include data preprocessing, feature selection, model training, evaluation, and deployment. The tool will be accessible through a web-based interface for ease of use."
Key Features	• “User-friendly interface for inputting data and receiving predictions." • “Integration of both behavioural and individual characteristics for comprehensive analysis." • “Real-time predictions with high accuracy, sensitivity, and specificity." • “Accessible to both healthcare professionals and parents."

Data Collection and Pre-processing Phase

Data Collection Plan and Raw Data Sources Identified:

The data files used for the development of this model were accessed as freely available on Kaggle.

The data was developed synthetically by using CTGan on Autism Screening on Adults Dataset. The dataset creation, as well as the competition setup, was done by
The dataset creation, as well as the competition setup, was done by Suryansu Dash and Laxman Naik of GDSC CET-Bhubaneswar.

Data Quality Report:

The data was developed synthetically by using CTGan. The dataset contained information about individuals who had opted to take the test. It included factors such as the gender, ethnicity, age, country of residence of the individual as well as whether the individual had a past history of jaundice, and had used the app before.

All these factors were considered to potentially help us identify patterns in individuals suffering with ASD.

The dataset had several missing values in the columns related to ‘age’, ‘ethnicity’ and ‘relation’ which had to be dealt with in the data pre-processing stage.

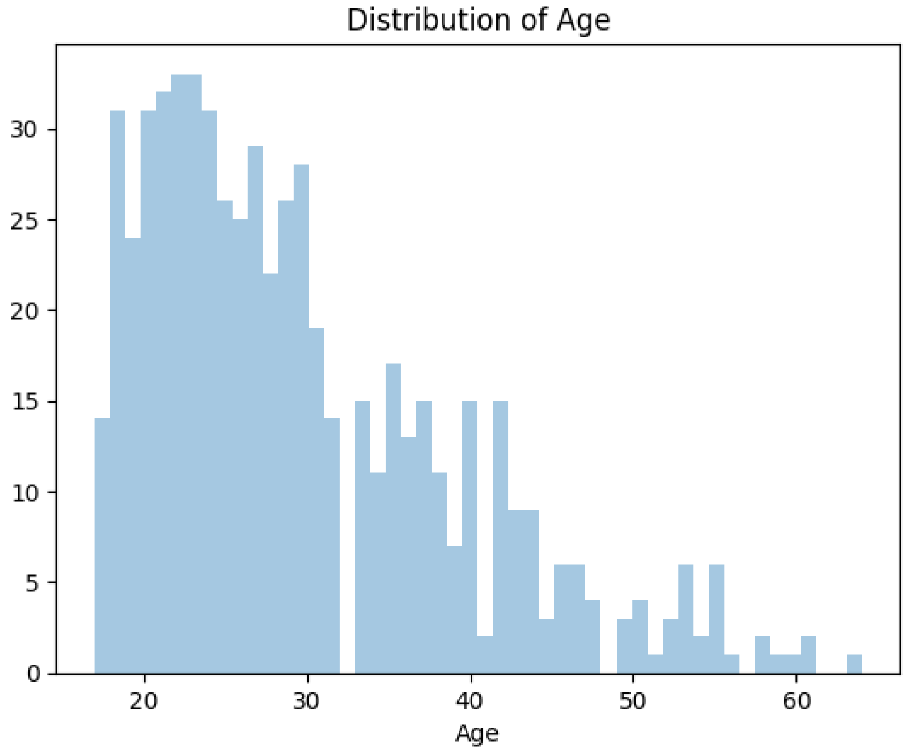
Data Exploration and Pre-processing:

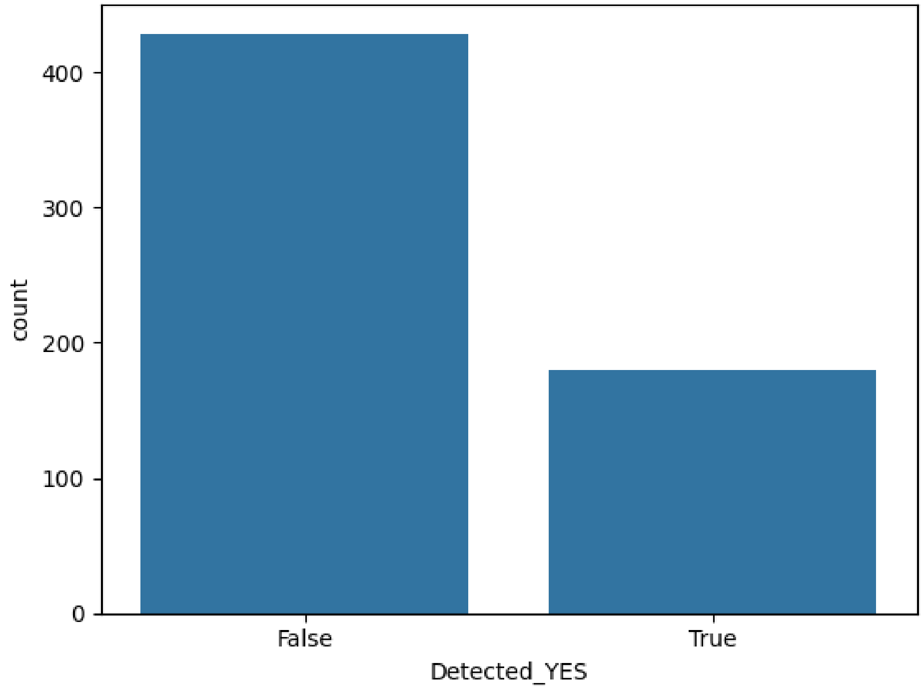
During the development process several data pre-processing methods were utilised to integrate and clean the data and transform it into useful forms.

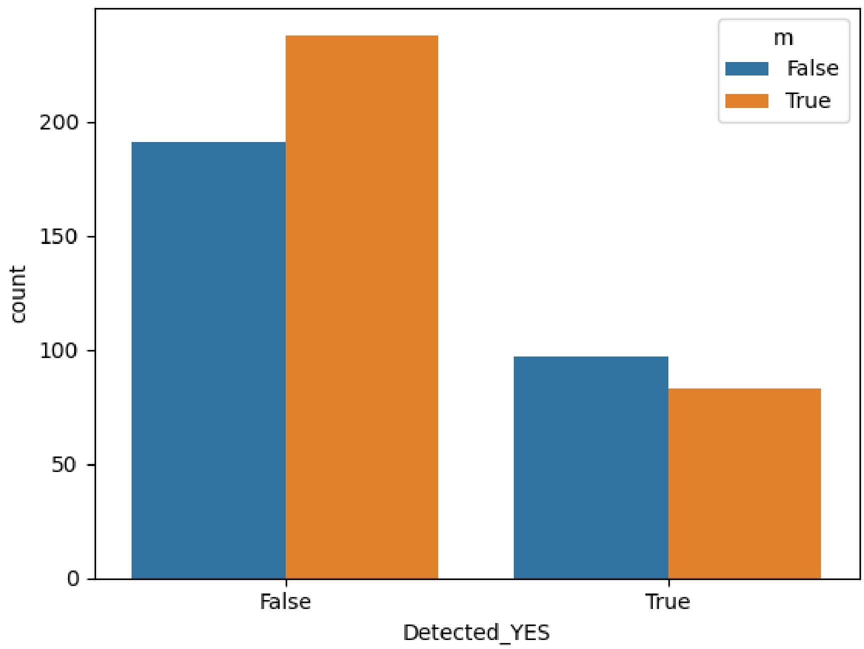
- **Dropping Unwanted Columns:** Unwanted columns such as country of residence, age, relation of the person, ethnicity and whether the person had used the app before or not were dropped to remove data that was unrelated to model building.
- **Handling Categorical Data:** Categorical data was handled by creating dummy variables to replace the columns - ‘gender’, ‘autism’, ‘jaundice’, ‘class/ASD’
- **Filling missing values:** The dataset had a few rows where values of factors such as age, ethnicity and relation were missing. This was tackled by replacing the missing values in the age column by the mean of the values present in the age column.

Model Development Phase

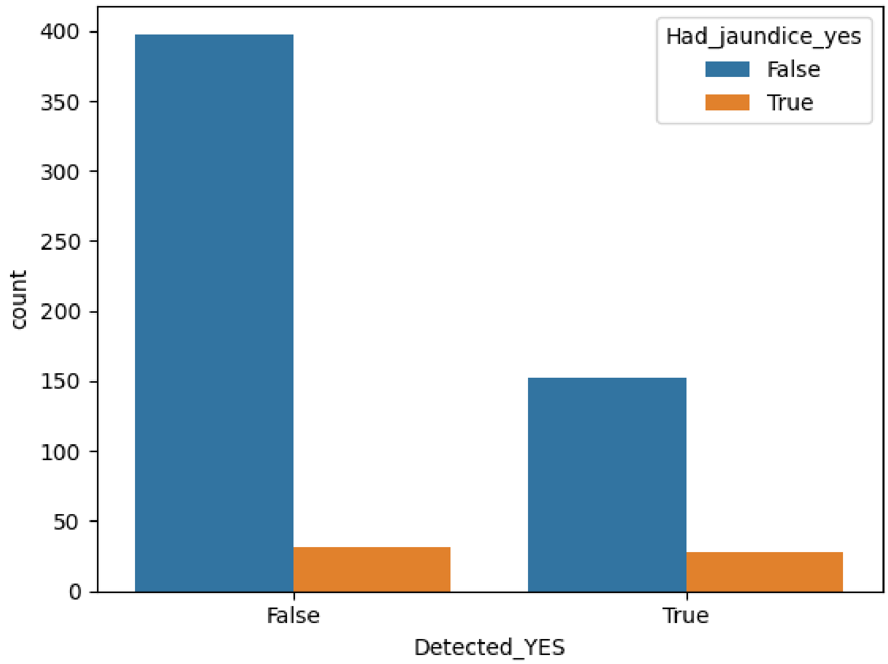
Feature Selection Report: To identify what features are the most relevant for training the model, a series of data visualisation techniques were utilised to find out patterns and relationships between the various input variables.



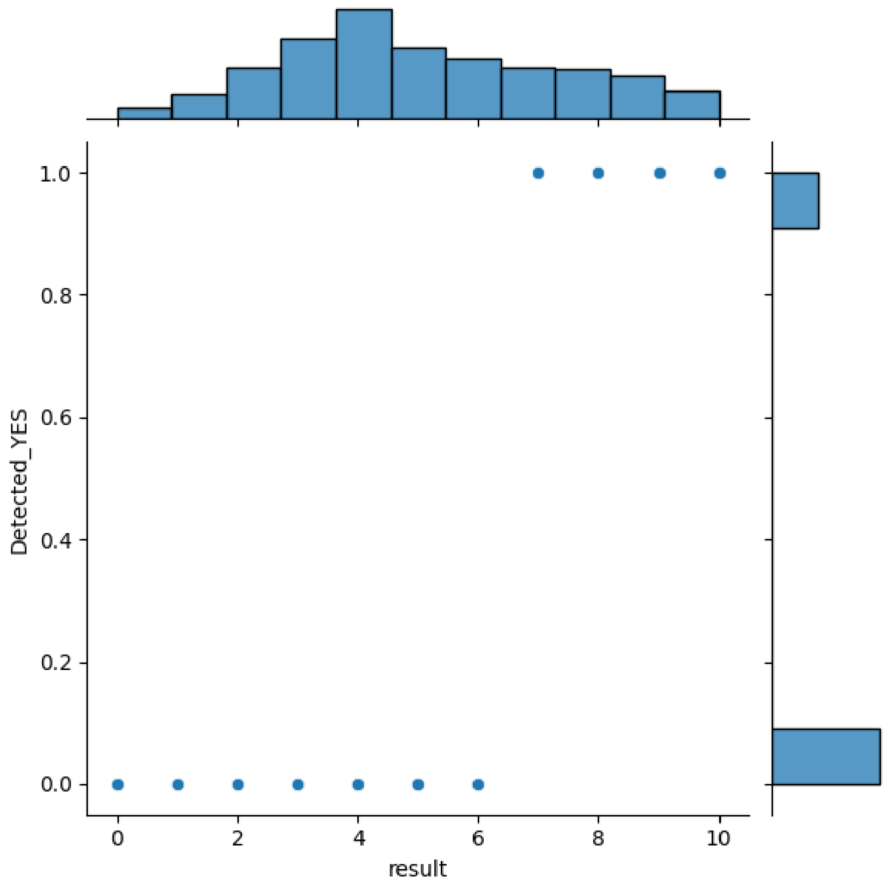




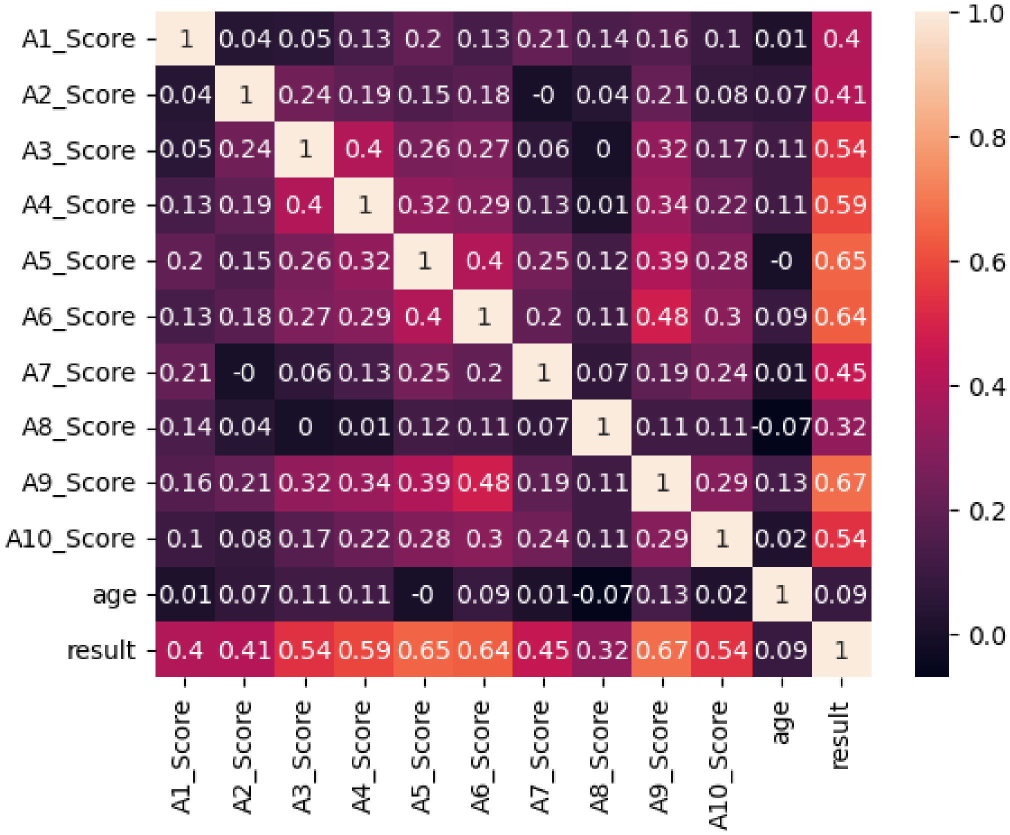
m
False
True
200
150
100
50
0
count
False
True
Detected_YES



Had_jaundice_yes
False
True
400
350
300
250
200
150
100
50
0
count
False
True
Detected_YES



Detected_YES
result
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10



A1_Score
A2_Score
A3_Score
A4_Score
A5_Score
A6_Score
A7_Score
A8_Score
A9_Score
A10_Score
age
result
1 0.04 0.05 0.13 0.2 0.13 0.21 0.14 0.16 0.1 0.01 0.4
0.04 1 0.24 0.19 0.15 0.18 -0 0.04 0.21 0.08 0.07 0.41
0.05 0.24 1 0.4 0.26 0.27 0.06 0 0.32 0.17 0.11 0.54
0.13 0.19 0.4 1 0.32 0.29 0.13 0.01 0.34 0.22 0.11 0.59
0.2 0.15 0.26 0.32 1 0.4 0.25 0.12 0.39 0.28 -0 0.65
0.13 0.18 0.27 0.29 0.4 1 0.2 0.11 0.48 0.3 0.09 0.64
0.21 -0 0.06 0.13 0.25 0.2 1 0.07 0.19 0.24 0.01 0.45
0.14 0.04 0 0.01 0.12 0.11 0.07 1 0.11 0.11 -0.07 0.32
0.16 0.21 0.32 0.34 0.39 0.48 0.19 0.11 1 0.29 0.13 0.67
0.1 0.08 0.17 0.22 0.28 0.3 0.24 0.11 0.29 1 0.02 0.54
0.01 0.07 0.11 0.11 -0 0.09 0.01 -0.07 0.13 0.02 1 0.09
0.4 0.41 0.54 0.59 0.65 0.64 0.45 0.32 0.67 0.54 0.09 1
1.0
0.8
0.6
0.4
0.2
0.0

Based on the above visualisations, we take the following parameters to train the model:

Feature	Description	Selected (Yes/No)	Reasoning
A1_Score	Response to screening question 1	Yes	Critical for assessing autism-related symptoms.
A2_Score	Response to screening question 2	Yes	Critical for assessing autism-related symptoms
A3_Score	Response to screening question 3	Yes	Critical for assessing autism-related symptoms
A4_Score	Response to screening question 4	Yes	Critical for assessing autism-related symptoms
A5_Score	Response to screening question 5	Yes	Critical for assessing autism-related symptoms
A6_Score	Response to screening question 6	Yes	Critical for assessing autism-related symptoms
A7_Score	Response to screening question 7	Yes	Critical for assessing autism-related symptoms
A8_Score	Response to screening question 8	Yes	Critical for assessing autism-related symptoms
A9_Score	Response to screening question 9	Yes	Critical for assessing autism-related symptoms

A10_Score	Response to screening question 10	Yes	Critical for assessing autism-related symptoms
age	Age of the individual	Yes	Important demographic information.
gender	Gender of the individual	Yes	Important demographic information.
ethnicity	Ethnicity of the individual	No	Not relevant
jaundice	History of jaundice	Yes	Relevant medical history.
autism	Family history of autism	Yes	Relevant family medical history.
contry_of_res	Country of residence	No	Not relevant
used_app_before	Prior use of autism screening app	No	Not relevant
result	Outcome of the screening test	Yes	Imporant to predict
age_desc	Age description (e.g., child, adult)	No	Not relevant
relation	Relationship of the respondent to	No	Not relevant

	the individual screened		
Class/ASD	ASD diagnosis outcome	Yes	Target variable for prediction.

Model Selection Report:

Model	Description	Hyperparameters	Performance Metric (e.g., Accuracy, F1 Score)
SVM	It finds the optimal hyperplane that best separates the data into different classes. It is particularly effective in high-dimensional spaces and is known for its robustness against overfitting, especially in cases where the number of dimensions exceeds the number of samples.		97.54%
KNN	Classifies based on nearest neighbors; adapts well to data patterns, effective for local variations in loan approval criteria.		96.72%
Logistic Regression	Logistic regression is a statistical model used for binary classification that predicts the probability of an outcome by fitting data to a logistic curve.		98.3%

Decision Tress	Simple tree structure; interpretable, captures non-linear relationships, suitable for initial insights into loan approval patterns.		95.3%
Random Forest	Ensemble of decision trees; robust, handles complex relationships, reduces overfitting, and provides feature importance for loan approval prediction.		96.2%

<u>Initial Model Training Code, Model Validation and Evaluation Report:</u>

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=21)
```

Logistic Regression

```
lgr = LogisticRegression()
```

```
lgr.fit(X_train,y_train)
```

```
lgr.score(X_test,y_test)
```

```
accuracy_lr = lgr.score(X_test,y_test)*100
```

```
accuracy_lr
```

```
print(classification_report(y_true=y_test,y_pred=pred))
```

SVC

```
svm = SVC(kernel = 'rbf',random_state=0)
svm.fit(X_train,y_train)
```

```
SVC
SVC(random_state=0)
```

```
svm.score(X_train,y_train)
```

```
0.997946611909651
```

```
svm.score(X_test,y_test)
```

```
0.9754098360655737
```

```
accuracy_SVM = svm.score(X_test,y_test)*100
```

```
accuracy_SVM
```

```
97.54098360655738
```

```
pred = svm.predict(X_test)
```

```
print(classification_report(y_true=y_test,y_pred=pred))
```

KNN

In [73]:
```
knn = KNeighborsClassifier(n_neighbors=5, metric = 'minkowski',p=2)
```

In [74]:
```
knn.fit(X_train,y_train)
```

Out[74]:
```
▾ KNeighborsClassifier
KNeighborsClassifier()
```

In [75]:
```
y_pred = knn.predict(X_test)
```

In [76]:
```
accuracy_KNN = accuracy_score(y_test, y_pred)*100
accuracy_KNN
```

Out[76]: 96.72131147540983

In [77]:
```
print(classification_report(y_true=y_test,y_pred=y_pred))
```

Random Forest

```
rand_forest = RandomForestClassifier(random_state=42)
```

```
rand_forest.fit(X_train,y_train)
```

```
y_pred = rand_forest.predict(X_test)
```

```
accuracy_RF = rand_forest.score(X_test,y_pred)*100
```

```
accuracy_RF
```

```
print(classification_report(y_true=y_test,y_pred=y_pred))
```

```
from sklearn.metrics import f1_score
f1 = f1_score(y_test, y_pred)
f1
```

Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix
SVM	precision recall f1-score support 0 0.97 1.00 0.98 87 1 1.00 0.91 0.96 35 accuracy 0.98 122 macro avg 0.98 0.96 0.97 122 weighted avg 0.98 0.98 0.98 122	97.54%	[[87 0 [0 35
KNN	precision recall f1-score support 0 0.99 0.97 0.98 87 1 0.92 0.97 0.94 35 accuracy 0.97 122 macro avg 0.95 0.97 0.96 122 weighted avg 0.97 0.97 0.97 122	96.72%	[[84 3] [1 34]
Logistic Regression	precision recall f1-score support 0 0.98 1.00 0.99 41 1 1.00 0.95 0.97 20 accuracy 0.98 61 macro avg 0.99 0.97 0.98 61 weighted avg 0.98 0.98 0.98 61	98.3%	[[41 0] [1 19]]
Decision Trees	precision recall f1-score support 0 0.98 0.98 0.98 41 1 0.95 0.95 0.95 20 accuracy 0.97 61 macro avg 0.96 0.96 0.96 61 weighted avg 0.97 0.97 0.97 61	95.3%	[[40 1] [1 19]]
Random Forest	precision recall f1-score support 0 0.98 0.98 0.98 41 1 0.95 0.95 0.95 20 accuracy 0.97 61 macro avg 0.96 0.96 0.96 61 weighted avg 0.97 0.97 0.97 61	96.72%	[[40 1] [1 19]]

Model Optimization and Tuning Phase

<u>Hyper parameter Tuning Documentation:</u>

Model	**Tuned Hyperparameters**	**Optimal Values**

KNN	param_grid_knn = {
	'n_neighbors': [3, 5, 7, 9],
	'weights': ['uniform', 'distance']
	}
	grid_search_knn = GridSearchCV(estimator=KNeighborsClassifier(), param_grid=param_grid_knn, cv=5, scoring='accuracy')
	grid_search_knn.fit(X_train, y_train)
	best_knn = grid_search_knn.best_estimator_
		print(f'Best parameters for KNN: {grid_search_knn.best_params_}')
		print(f'Accuracy for KNN: {accuracy_score(y_test, best_knn.predict(X_test)):.4f}')
		Best parameters for KNN: {'n_neighbors': 7, 'weights': 'uniform'}
		Accuracy for KNN: 0.9672

Logistic Regression	# Hyper Parameter tuning
	param_grid_lr = {
	'C': [0.1, 1, 10, 100],
	'solver': ['liblinear', 'lbfgs']
	}
	grid_search_lr = GridSearchCV(estimator=LogisticRegression(), param_grid=param_grid_lr, cv=5, scoring='accuracy')
	grid_search_lr.fit(X_train, y_train)
	best_lr = grid_search_lr.best_estimator_
		print(f'Best parameters for Logistic Regression: {grid_search_knn.best_params_}')
		print(f'Accuracy for Logistic Regression: {accuracy_score(y_test, best_knn.predict(X_test)):.4f}')
		Best parameters for Logistic Regression: {'max_depth': None, 'min_samples_split': 2, 'n_estimators': 100
		Accuracy for Logistic Regression: 0.972

SVM	param_grid_svm = {
	'C': [0.1, 1, 10, 100],
	'kernel': ['linear', 'rbf']
	}
	grid_search_svm = GridSearchCV(estimator=SVC(), param_grid=param_grid_svm, cv=5, scoring='accuracy')
	grid_search_svm.fit(X_train, y_train)
	best_svm = grid_search_svm.best_estimator_
		print(f'Best parameters for SVM: {grid_search_svm.best_params_}')
		print(f'Accuracy for SVM: {accuracy_score(y_test, best_svm.predict(X_test)):.4f}')
		Best parameters for SVM: {'C': 0.1, 'kernel': 'linear'}
		Accuracy for SVM: 1.0000

Decision Trees	param_grid_dt = {
	'max_depth': [None, 10, 20, 30],
	'min_samples_split': [2, 5, 10]
	}
	grid_search_dt = GridSearchCV(estimator=DecisionTreeClassifier(), param_grid=param_grid_dt, cv=5, scoring='accuracy')
	grid_search_dt.fit(X_train, y_train)
	best_dt = grid_search_dt.best_estimator_
		print(f'Best parameters for Decision Tree: {grid_search_dt.best_params_}')
		print(f'Accuracy for Decision Tree: {accuracy_score(y_test, best_dt.predict(X_test)):.4f}')
		Best parameters for Decision Tree: {'max_depth': None, 'min_samples_split': 2}
		Accuracy for Decision Tree: 1.0000

Random Forest	param_grid_rf = {
	'n_estimators': [100, 200, 300],
	'max_depth': [None, 10, 20],
	'min_samples_split': [2, 5, 10]
	}
	grid_search_rf = GridSearchCV(estimator=RandomForestClassifier(), param_grid=param_grid_rf, cv=5, scoring='accuracy')
	grid_search_rf.fit(X_train, y_train)
	best_rf = grid_search_rf.best_estimator_
		print(f'Best parameters for Random Forest: {grid_search_rf.best_params_}')
		print(f'Accuracy for Random Forest: {accuracy_score(y_test, best_rf.predict(X_test)):.4f}')
		Best parameters for Random Forest: {'max_depth': None, 'min_samples_split': 2,
		Accuracy for Random Forest: 1.0000

Performance Metrics Comparison Report:

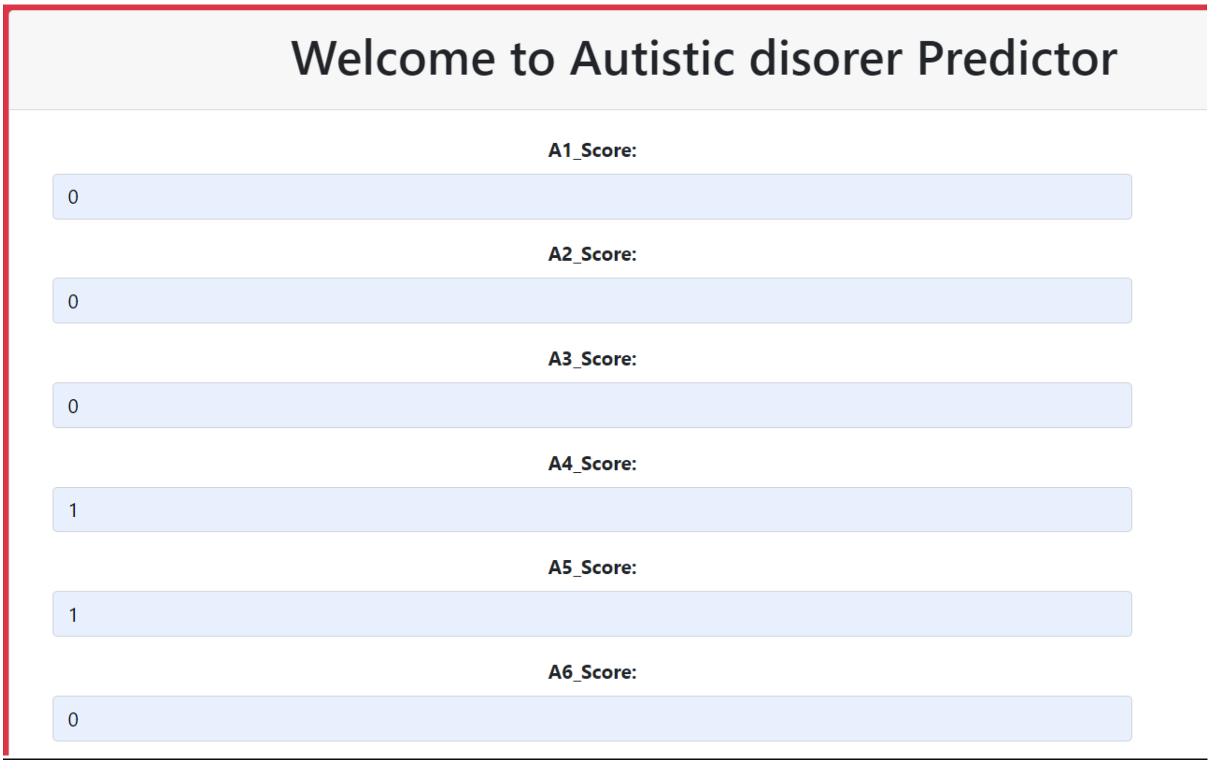
Model	Optimized Metric				
		precision	recall	f1-score	support
KNN	0	0.99	0.97	0.98	87
	1	0.92	0.97	0.94	35
	accuracy			0.97	122
	macro avg	0.95	0.97	0.96	122
	weighted avg	0.97	0.97	0.97	122
		precision	recall	f1-score	support
Logistic Regression	0	0.98	1.00	0.99	41
	1	1.00	0.95	0.97	20
	accuracy			0.98	61
	macro avg	0.99	0.97	0.98	61
	weighted avg	0.98	0.98	0.98	61
		precision	recall	f1-score	support
SVM	0	0.97	1.00	0.98	87
	1	1.00	0.91	0.96	35
	accuracy			0.98	122
	macro avg	0.98	0.96	0.97	122
	weighted avg	0.98	0.98	0.98	122
		precision	recall	f1-score	support
Decision Trees	0	0.98	0.98	0.98	41
	1	0.95	0.95	0.95	20
	accuracy			0.97	61
	macro avg	0.96	0.96	0.96	61
	weighted avg	0.97	0.97	0.97	61
		precision	recall	f1-score	support
Random Forest	0	0.98	0.98	0.98	41
	1	0.95	0.95	0.95	20
	accuracy			0.97	61
	macro avg	0.96	0.96	0.96	61
	weighted avg	0.97	0.97	0.97	61

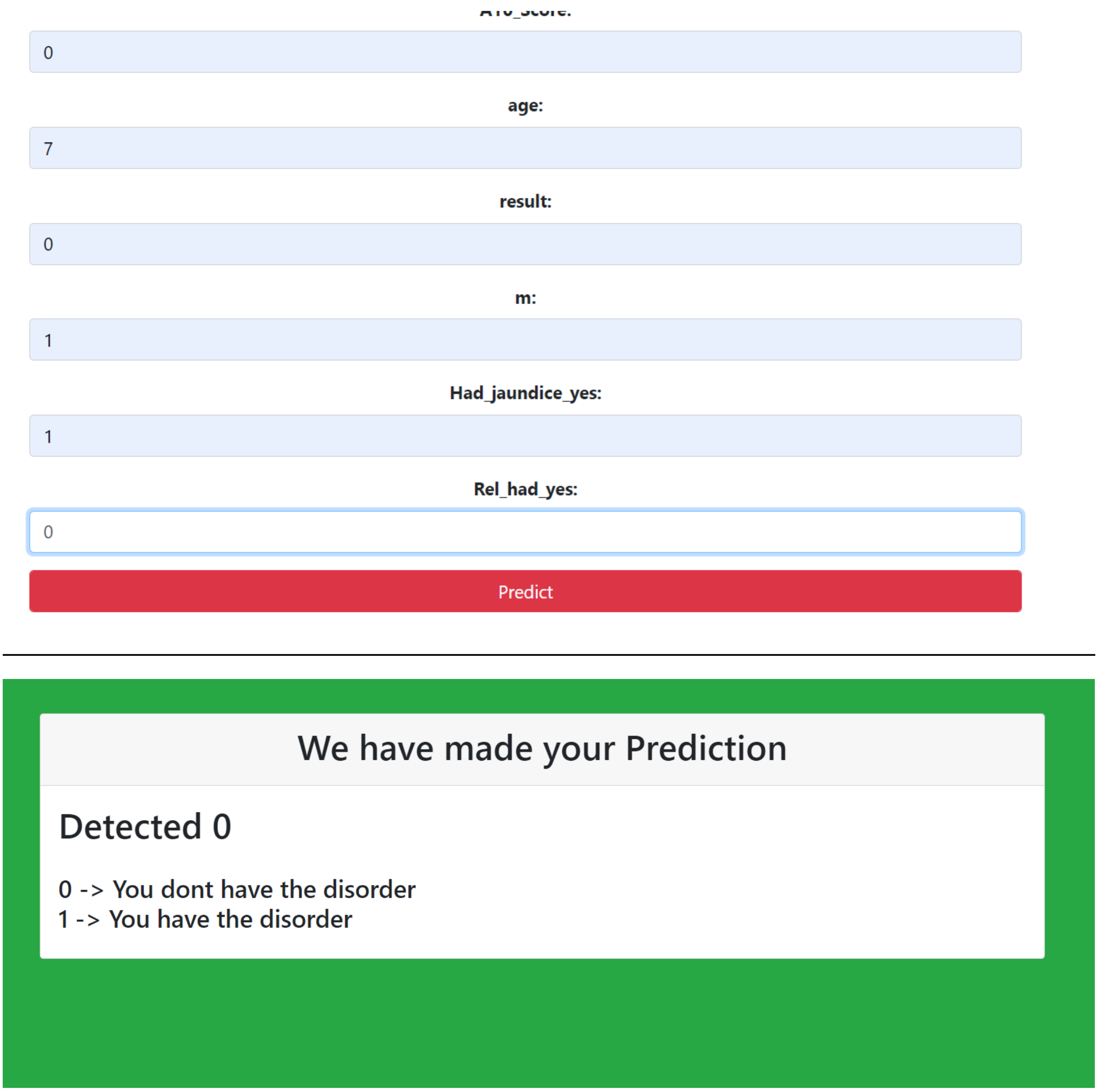
Final Model Selection Justification:

Final Model	Reasoning
Random Forest	The Random Forest model was selected for its superior performance, exhibiting high accuracy during hyperparameter tuning. Its ability to handle complex relationships, minimize overfitting, and optimize predictive accuracy aligns with project objectives, justifying its selection as the final model.

Results

Output Screenshots:




0
age:
7
result:
0
m:
1
Had_jaundice_yes:
1
Rel_had_yes:
0
Predict
We have made your Prediction
Detected 0
0 -> You dont have the disorder
1 -> You have the disorder

Advantages & Disadvantages

<u>Advantages:</u>

1. **Accuracy and Performance**: Random Forests generally provide high accuracy due to their ability to handle non-linear relationships and interactions between features.
2. **Robustness**: They are less prone to overfitting compared to individual decision trees because they aggregate the results of multiple trees.
3. **Feature Importance**: Random Forests can provide insights into the importance of different features, helping in understanding which factors contribute most to the predictions.
4. **Versatility**: They can handle both categorical and numerical data, making them suitable for diverse datasets.
5. **Scalability**: Random Forests are efficient and scalable to large datasets, which is beneficial for the autism prediction dataset that might have a considerable number of features and records.

<u>Disadvantages:</u>

1. **Complexity**: The model can become complex and difficult to interpret, especially when the number of trees and features is large.
2. **Computational Cost**: Training a Random Forest can be computationally expensive and time-consuming, particularly with large datasets.
3. **Memory Usage**: They can require significant memory for storing multiple trees and their associated data.
4. **Parameter Tuning**: There are several hyperparameters that need to be fine-tuned (e.g., number of trees, depth of trees), which can be challenging and require extensive cross-validation.
5. **Bias in Small Datasets**: In cases where the dataset is not sufficiently large, Random Forests may not perform as well and could still overfit despite their robustness.

Conclusion

The development of an autism prediction model using Random Forests demonstrates significant promise due to its high accuracy and robust performance. By leveraging the ensemble learning technique, the model effectively handles non-linear relationships and complex interactions within the dataset, providing reliable predictions. Despite challenges such as computational complexity and the need for hyperparameter tuning, the advantages of improved accuracy, robustness against overfitting, and insights into feature importance make Random Forests an excellent choice for this application. This model can substantially aid early diagnosis and personalized interventions for individuals with autism, enhancing their developmental outcomes and quality of life.

Future Scope

1. Early Diagnosis

- **Improved Screening**: Early diagnosis of autism spectrum disorder (ASD) is crucial for effective intervention. Prediction models can help in identifying signs of autism at a younger age than current methods. This can be particularly beneficial as early intervention is linked to better long-term outcomes.
- **Access and Outreach** With predictive models, screening can be made more accessible and widespread, including in underserved or remote areas where traditional diagnostic resources might be limited.

2. Personalized Interventions

- **Customized Treatment Plans**: Predictive models can help in understanding individual differences within the autism spectrum. This allows for more personalized and targeted treatment and educational plans tailored to each child's specific needs and strengths.
- **Dynamic Adjustments**: Models can assist in monitoring progress and adjusting interventions dynamically based on real-time data and predictive analytics, making treatments more responsive to the individual's development.

3. Healthcare Improvements

- **Cost Efficiency**: By improving early diagnosis and personalizing interventions, predictive models can potentially reduce the long-term costs associated with autism care, including the costs of delayed interventions and ineffective treatments.
- **Scalability**: Automated and scalable prediction tools can be deployed in various settings, from pediatric clinics to educational institutions, facilitating broader and more cost-effective screening and intervention processes.

4. Research Advancements

- **Data-Driven Insights**: Large datasets used for training prediction models can reveal new patterns and insights about autism. This can lead to a deeper understanding of the disorder's causes, progression, and response to treatments.

Appendix

GitHub & Project Demo Link:

GITHUB :

https://github.com/vaibhavyada/Autistic-Spectrum-disorder-classification

VIDEO :

https://youtu.be/4GStaGiFi1o?si=Nf31MLCRmzT6MQTB