**INTRODCTION:**

* Autism is an integral neurodevelopment disorder that affects the functioning of brain. It can occur at any age but usually occur in childhood, mostly children at the age of 2 or 3 have more chances of having ASD.
* It occurs due to the combination of both genetic and environmental factors.
* Autism is not an illness or a disease rather it is a neurological condition in which child is unable to concentrate, think, learn, focus and solve the problems.
* They find difficulty in explaining things through facial expression or by making Gestures, It is fastest growing development disability in all over the world including India.
* Autisum can be caused: Genetic- either of parents is suffering from this disorder, some other member in the family is autistic, and Complications during Pregnancy, child has not received proper and timely vaccination and so on.
* Detecting autism traits through screening tests is very expensive and time consuming. With the advancement of artificial intelligence and machine learning (ML), autism can be predicted at quite early stage.
* This project is to analyze various Machine learning algorithms, used by various researcher like SVM (support Vector Machine), Random Forest, Decision Trees, Logistic Regression and compare the result based on their accuracy and efficiency.

**EXISTING SYSTEM**

Most of the current existing systems uses older machine learning algorithms like logistic regression, Decision Tree etc which are efficient in processing but fails to generate good accuracy. There are some systems which uses Deep learning for autism detection but they require high processing power are a bit complex to use.

**DISADVANTAGES:**

* Low accuracy.
* High complexity.
* Difficult to scale.

**PROPOSED SYSTEM**

With the advancement of artificial intelligence and machine learning (ML), autism can be predicted at quite early stage. The main aim of this project is to analyze various Machine learning algorithms, used by various researcher like SVM (support Vector Machine), Random Forest, Decision Trees, Logistic Regression, and Linear Discriminant Analysis (LDA) and compare the result based on their accuracy and efficiency.

**ADVANTAGES:**

* + High accuracy.
  + Low complexity.
  + Easy to scale.

**LITERATURE SURVEY**

Literature [survey](http://www.blurtit.com/q876299.html) is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy n company strength. Once these things r satisfied, ten next steps are to determine which operating system and language can be used for developing the tool. Once the [programmers](http://www.blurtit.com/q876299.html) start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from [book](http://www.blurtit.com/q876299.html) or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

1. **Anshu Sharma, Dr. Poonam Tanwar “Deep Analysis of Autism Spectrum Disorder Detection Techniques”, 2022**

The authors provide an overview of the existing literature on machine learning techniques for ASD detection, including supervised, unsupervised, and deep learning approaches. They discuss the advantages and limitations of each approach and provide examples of studies that have used these techniques.

The paper also includes a comparative analysis of several studies that have used machine learning techniques for ASD detection. The authors compare the accuracy, sensitivity, specificity, and other performance metrics of these studies and discuss the factors that may contribute to differences in performance.

In addition, the authors discuss the challenges and limitations of using machine learning techniques for ASD detection, including the need for large and diverse datasets, the potential for bias in the data, and the need for interpretability and transparency in the models.

1. **Kazi Shahrukh Omar, Prodipta Mondal, Nabila Shahnaz Khan, Md. Rezaul Karim Rizvi, Md Nazrul Islam, “A Machine Learning Approach to Predict Autism Spectrum Disorder”, “International Conference on Electrical, Computer and Communication Engineering (ECCE)”, 2019**

In present day Autism Spectrum Disorder (ASD) is gaining its momentum faster than ever. Detecting autism traits through screening tests is very expensive and time consuming. With the advancement of artificial intelligence and machine learning (ML), autism can be predicted at quite early stage. Though number of studies have been carried out using different techniques, these studies didn't provide any definitive conclusion about predicting autism traits in terms of different age groups. Therefore, this paper aims to propose an effective prediction model based on ML technique and to develop a mobile application for predicting ASD for people of any age. As outcomes of this research, an autism prediction model was developed by merging Random Forest-CART (Classification and Regression Trees) and Random Forest-ID3 (Iterative Dichotomiser 3) and also a mobile application was developed based on the proposed prediction model. The proposed model was evaluated with AQ-10 dataset and 250 real dataset collected from people with and without autistic traits. The evaluation results showed that the proposed prediction model provide better results in terms of accuracy, specificity, sensitivity, precision and false positive rate (FPR) for both kinds of datasets.

1. **Md. Shahriare Satu , Farha Farida Sathi, Md. Sadrul Arifen, Md. Hanif Ali and Mohammad Ali Moni, “Early Detection of Autism by Extracting Features:A Case Study in Bangladesh”, “International Conference on Robotics,Electrical and Signal Processing Techniques” , 2019**

Autism Spectrum Disorder (ASD) is a neurobehavioral disorder that begins at childhood and exists this whole life. The objective of this work is that to explore significant features of normal and autism of divisional regions in Bangladesh. We collected individual samples of various children from their parents between 16 to 30 months of different residents using Autism Barta apps by web and fieldwork at Savar, Bangladesh. Then, we pre-processed our data and categorized frequent features based on their individual regions. Different tree based techniques such as J48, Logistic Model Tree, Random Forest, Reduced Error Pruned Tree, and Decision Stump were analyzed to find out the best classifier of them. From these classifiers, J48 showed the best outcomes than other classifiers. We extracted 9 rules and associated conditions from J48 decision tree and gathered frequent instances from our data for extracted rules. Finally, 8 within 23 features were required to classify normal and autism of individual regions in Bangladesh. Besides, 4 rules (10 Conditions) for normal and 5 (12 Conditions) rules for autism out of 9 (16 Conditions) rules were extracted from decision tree. This outcomes assist us to find out significant features of autism in Bangladesh. We expect that our work will be helpful things to improve their condition that leads them to a normal life.

1. **Kayleigh K. Hyde, Marlena N. Novack, Nicholas LaHaye, Chelsea Parlett-Pelleriti ,Raymond Anden, Dennis R. Dixon. Erik Linstead. “Applications of Supervised Machine Learning in Autism Spectrum Disorder” Research: “Review Journal of Autism and Developmental Disorders”, 2019.**

Autism spectrum disorder (ASD) research has yet to leverage “big data” on the same scale as other fields; however, advancements in easy, affordable data collection and analysis may soon make this a reality. Indeed, there has been a notable increase in research literature evaluating the effectiveness of machine learning for diagnosing ASD, exploring its genetic underpinnings, and designing effective interventions. This paper provides a comprehensive review of 45 papers utilizing supervised machine learning in ASD, including algorithms for classification and text analysis. The goal of the paper is to identify and describe supervised machine learning trends in ASD literature as well as inform and guide researchers interested in expanding the body of clinically, computationally, and statistically sound approaches for mining ASD data.

1. **Fadi Thabtah, “Machine learning in autistic spectrum disorder behavioral research: A review and ways forward”, “informatics for Health and Social Care”,2018 ,**

Autistic Spectrum Disorder (ASD) is a mental disorder that retards acquisition of linguistic, communication, cognitive, and social skills and abilities. Despite being diagnosed with ASD, some individuals exhibit outstanding scholastic, non-academic, and artistic capabilities, in such cases posing a challenging task for scientists to provide answers. In the last few years, ASD has been investigated by social and computational intelligence scientists utilizing advanced technologies such as machine learning to improve diagnostic timing, precision, and quality. Machine learning is a multidisciplinary research topic that employs intelligent techniques to discover useful concealed patterns, which are utilized in prediction to improve decision making. Machine learning techniques such as support vector machines, decision trees, logistic regressions, and others, have been applied to datasets related to autism in order to construct predictive models. These models claim to enhance the ability of clinicians to provide robust diagnoses and prognoses of ASD. However, studies concerning the use of machine learning in ASD diagnosis and treatment suffer from conceptual, implementation, and data issues such as the way diagnostic codes are used, the type of feature selection employed, the evaluation measures chosen, and class imbalances in data among others. A more serious claim in recent studies is the development of a new method for ASD diagnoses based on machine learning.

1. **Daniel Bone, Somer Bishop, Matthew P. Black, Matthew S. Goodwin, Catherine Lord and Shrikanth S. Narayanan, “Use of machine learning to improve autism screening and diagnostic instruments: effectiveness, efficiency, and multi-instrument fusion”, “Journal of Child Psychology and Psychiatry”, 2016**

Machine learning (ML) provides novel opportunities for human behavior research and clinical translation, yet its application can have noted pitfalls (Bone et al., 2015). In this work, we fastidiously utilize ML to derive autism spectrum disorder (ASD) instrument algorithms in an attempt to improve upon widely used ASD screening and diagnostic tools.

**SYSTEM SPECIFICATIONS:**

# H/W System Configuration:

# Processor - I3/Intel Processor

* RAM - 4GB (min)
* Hard Disk - 160GB
* Key Board - Standard Windows Keyboard
* Mouse - Two or Three Button Mouse
* Monitor - SVGA

**S/W System Configuration:**

* Operating System : Windows 10
* Scripts : JavaScript, Jquery.
* Server side Script : Python
* IDE : PyCharm, Anaconda.
* Packages : Sklearn, Pandas, Numpy, Matplotlib, Seaborn

**REQUIREMENTS ANALYSIS:**

**Non –functional Requirements:**

* **Reliability:** During runtime, the system is always be up to perform the requested task 99.9%of the time except for some abnormal disruptions.
* **Performance:** Numerous user should be able to use the system at all time
* **Maintainability:** Watching and managing the program should be essential and should be oriented in its approach.
* **Portability:** The system should apply to another application efficiently.
* **Scalability:** The system will be adaptable enough at later level to add new functionalities.
* **Flexibility:** Flexibility is the capacity of a system to respond to changes in the system environments and tom adjust to changes in market strategies and laws.

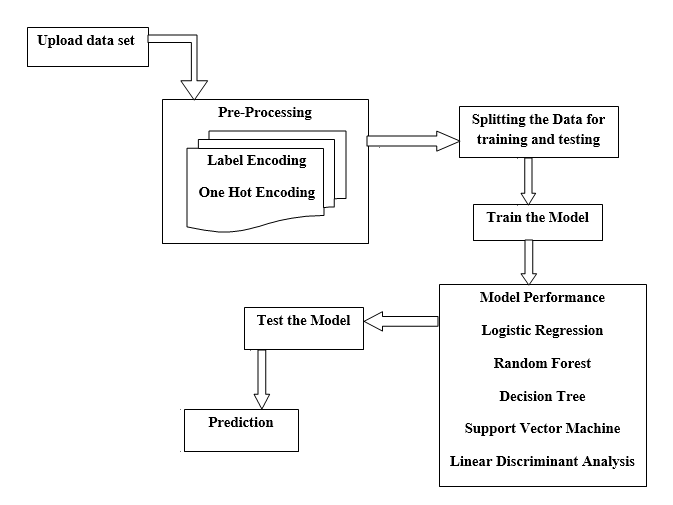
**Functional Requirements**

* Collect the dataset from the internet(Kaggle),which has more than 600 records to find the autism
* In this dataset, ten behavioral features(Q-chat-10) plus other individuals characteristics that have provided to be effective in detecting the ASD cases from controls in behavior science are present.
* This dataset contained some string value parameter based on that we need to convert string to numeric. then plot the graph for analysing the data.
* Split the data into train and test dataset.By making use of trainig data,we are going to train our algorithms namely: Logistic regression, Random forest,SVM classifier.
* Then we are compareing all algorithms based on accuracy.

**SYSTEM DESIGN**

**ARCHITECTURE:**

The architecture of Autism Spectrum Disorder (ASD) involving upload data, preprocessing, split data, train data, test data, and prediction is a machine learning-based approach used to predict the likelihood of a person having ASD based on their behavioral and demographic characteristics.

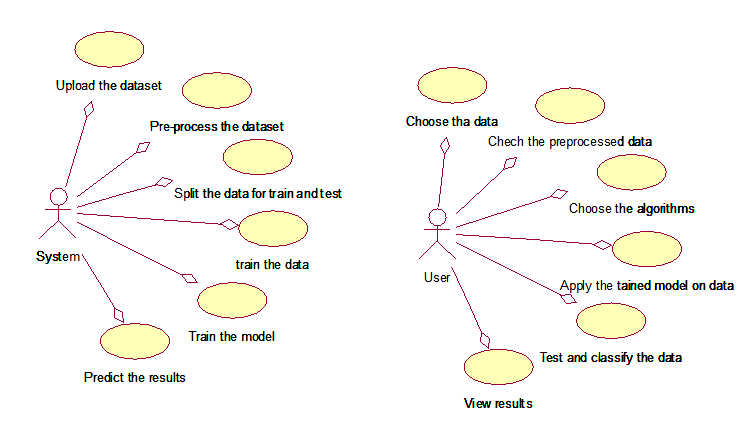


* The first step in this architecture is to upload the data to be used for the prediction. This data may include information such as age, gender, behavioral characteristics, and medical history. The data is typically stored in a database or in a file format such as CSV or Excel.
* The next step is to preprocess the data. This involves cleaning and transforming the data to ensure that it is in a format suitable for analysis. This may involve removing missing values, converting categorical variables to numerical values, and scaling or normalizing the data.
* Once the data has been preprocessed, it is split into two sets: the training set and the test set. The training set is used to train the machine learning model, while the test set is used to evaluate the performance of the model.
* The next step is to train the machine learning model using the training data. This involves selecting an appropriate machine learning algorithm, such as logistic regression or a decision tree, and fitting the algorithm to the training data. The model is then optimized using techniques such as cross-validation to ensure that it is robust and accurate.
* After the model has been trained, it is tested using the test data. This involves using the model to predict the likelihood of ASD for each individual in the test set, and comparing these predictions to the actual diagnoses to evaluate the performance of the model.
* Finally, the model can be used to make predictions on new data. This involves applying the model to new data that has not been used in the training or testing phases to predict the likelihood of ASD for new individuals.
* In summary, the architecture of ASD involving upload data, preprocessing, split data, train data, test data, and prediction is a machine learning-based approach that can be used to predict the likelihood of a person having ASD based on their behavioral and demographic characteristics. This approach can be used to improve the accuracy and efficiency of ASD diagnosis and treatment.

**UML DIAGRAMS**

**USE CASE DIAGRAM:**

A use case diagram is a graphical representation of the interactions between users and a system. In the case of Autism Spectrum Disorder (ASD), a use case diagram could be used to illustrate the various actions that users and the system may take in order to predict the likelihood of ASD based on data.



In this diagram, there are two actors: the user and the system. The user is represented by the stick figure icon, and the system is represented by the box icon.

The use case diagram shows several actions that can be taken by the user and the system, including:

**Upload data:** The user can upload data to the system, which can be used to predict the likelihood of ASD. This action is represented by the "Upload Data" use case.

**Preprocessing:** The system can preprocess the uploaded data to ensure that it is in a format suitable for analysis. This action is represented by the "Preprocessing" use case.

**Split data:** The system can split the preprocessed data into two sets: the training set and the test set. This action is represented by the "Split Data" use case.

**Train data:** The system can train a machine learning model using the training data. This action is represented by the "Train Data" use case.

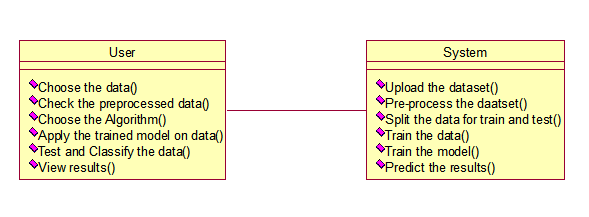
**Test data:** The system can test the trained model using the test data. This action is represented by the "Test Data" use case.

**Prediction:** The system can use the trained model to predict the likelihood of ASD for new data. This action is represented by the "Prediction" use case.

The use case diagram provides a high-level view of the interactions between the user and the system, and can be used to guide the design and development of the ASD prediction system.

**CLASS DIAGRAM :**

A class diagram is a type of UML diagram that shows the structure of a system by depicting its classes, attributes, methods, and relationships. In the context of Autism Spectrum Disorder (ASD), a class diagram could be used to illustrate the classes and relationships involved in the ASD prediction system.



In this diagram, there are two main classes: the User class and the System class. The User class represents the user of the ASD prediction system, while the System class represents the system itself.

The attributes and methods of the User class include:

**Upload Data:** This method allows the user to upload data to the system.

**Preprocess Data:** This method allows the user to preprocess the uploaded data.

**Split Data:** This method allows the user to split the preprocessed data into training and test sets.

**Test Data:** This method allows the user to test the trained model using the test data.

The attributes and methods of the System class include:

**Preprocess Data:** This method preprocesses the uploaded data.

**Split Data:** This method splits the preprocessed data into training and test sets.

**Train Model:** This method trains a machine learning model using the training data.

**Test Model:** This method tests the trained model using the test data.

**Predict:** This method uses the trained model to predict the likelihood of ASD for new data.

There are also several relationships depicted in the class diagram:

* The User class has an association with the System class, indicating that the user interacts with the system to upload, preprocess, and test data.
* The System class has an aggregation relationship with the Preprocessing, Splitting, Training, and Testing classes, indicating that the system uses these classes to perform its functions.
* The Training and Testing classes have a composition relationship with the Model class, indicating that the model is an integral part of the training and testing processes.
* The class diagram provides a detailed view of the classes and relationships involved in the ASD prediction system, and can be used to guide the implementation of the system.

**SEQUENCE DIAGRAM :**

A sequence diagram is a type of UML diagram that shows the interactions between objects in a system over time. In the context of Autism Spectrum Disorder (ASD), a sequence diagram could be used to illustrate the sequence of actions that the user and the system take in order to predict the likelihood of ASD based on data.



In this diagram, there are two main objects: the User object and the System object. The User object represents the user of the ASD prediction system, while the System object represents the system itself.

The sequence diagram shows several interactions between the User and System objects, including:

**Upload Data:** The user uploads data to the system using the "Upload Data" method.

**Preprocessing:** The system preprocesses the uploaded data using the "Preprocessing" method.

**Split Data:** The system splits the preprocessed data into training and test sets using the "Split Data" method.

**Train Model:** The system trains a machine learning model using the training data using the "Train Model" method.

**Test Model:** The system tests the trained model using the test data using the "Test Model" method.

**Prediction:** The system uses the trained model to predict the likelihood of ASD for new data using the "Prediction" method.

The sequence diagram shows the sequence of interactions between the User and System objects, and can be used to guide the implementation and testing of the ASD prediction system.

**Implementation Technologies:**

**PYTHON**

* Python is a versatile, easy-to learn progarmminng language.
* Python is interpreter- You don’t have to compile your program before it is executed.
* Python is platform independent i.e. programs can be implemented on one machine and use them on another machine without any changes.

**Python libraries used:**

* **NUMPY**: which stands for Numerical python, is a library of multidimensional array objects.
* **Pandas :** It gives a set of tools to perform data analysis
* **Matplotlib :** It gives a simulation frame work for 2D plotting.
* **Scikit-learn** : Offers sevral supervised and unsupervised machine learning algorithms via simple python framework.

**Machine learning :**

* Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

ML Algorithm used:

**Random forest classification**:

* Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of **ensemble learning,** which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.
* As the name suggests, ***"Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset."*** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.
* **The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.**

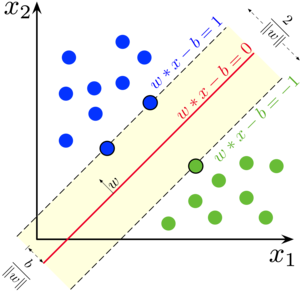
The below diagram explains the working of the Random Forest algorithm:



**Support vector Machine:**

* Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.
* The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.
* SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.
* SVM algorithm can be used for **Face detection, image classification, text categorization,** etc.

Consider the below diagram in which there are two different categories that are classified using a decision boundary or hyperplane:



# Decision Tree Classification Algorithm:

* Decision Tree is a **Supervised learning technique**that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where**internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.**
* In a Decision tree, there are two nodes, which are the **Decision Node** and**Leaf Node.** Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.
* The decisions or the test are performed on the basis of features of the given dataset.
* ***It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.***
* It is called a decision tree because, similar to a tree, it starts with the root node, which expands on further branches and constructs a tree-like structure.
* In order to build a tree, we use the **CART algorithm,** which stands for **Classification and Regression Tree algorithm.**
* A decision tree simply asks a question, and based on the answer (Yes/No), it further split the tree into subtrees.

Below diagram explains the general structure of a decision tree:



# Logistic Regression in Machine Learning:

* Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.
* Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, **it gives the probabilistic values which lie between 0 and 1**.
* Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas **Logistic regression is used for solving the classification problems**.
* In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
* The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
* Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

The below image is showing the logistic function:



**METHODOLOGY:**

Detecting Autism Spectrum Disorder (ASD) using machine learning techniques involves training a model on a dataset of features extracted from patients with ASD and healthy individuals. The following methodology can be used for the task.

* **Data collection:** Collect a large dataset of individuals with ASD and healthy individuals. The dataset should include a range of features such as demographic information, medical history, cognitive and behavioral assessments, and genetic information.
* **Data pre-processing:** Clean and preprocess the data by removing any missing or inconsistent values, normalizing the data, and encoding categorical variables.
* **Feature selection:** Select a subset of relevant features for the model by analyzing the correlation between each feature and the target variable.
* **Model selection:** Choose an appropriate machine learning algorithm for the task, such as logistic regression, decision trees, or support vector machine.
* **Model training:** Train the model using the selected features and algorithm. Use a portion of the dataset for training and the remaining data for validation to evaluate the performance of the model.
* **Model tuning:** Optimize the model's hyperparameters to improve its performance. Use techniques such as cross-validation or grid search to find the optimal values for the hyperparameters.
* **Model evaluation:** Evaluate the performance of the model on a separate testing dataset. Use metrics such as accuracy, precision, recall, and F1-score to assess the model's performance.
* **Deployment:** Deploy the model in a real-world setting, such as a clinical setting, and monitor its performance over time.
* **Continuous improvement:** Continuously monitor and improve the model's performance by collecting new data, retraining the model, and updating the model's hyperparameters.
* To develop an effective predictive model,Q-chat-10 data set used.
* The dataset has age, gender, ethnicity and Autism. specific Screening questions focus on different Domains such as attention to detail, attention switching, communication, imagination, and social interaction.

|  |  |
| --- | --- |
| Variable in Dataset | Corresponding Q-Chat-10 for children features. |
| A1\_Score | Does your child look at you when you call his/her name? |
| A2\_Score | How easy is it for you to make eye contact with your child? |
| A3\_Score | Does your child point to indicate that s/he wants something? |
| A4\_Score | Does your child point to share an interest with you? |
| A5\_Score | Does your child pretend? |
| A6\_Score | Does your child follow where you’re looking? |
| A7\_Score | Would you describe your child’s first words as? |
| A8\_Score | Does your child use simple gestures? |
| A9\_Score | Does your child stare at nothing words no apperent purpose? |
| A10\_Score |  |

**FUTUR SCOPE OF DEVELOPMENT:**

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental disorder that affects social interaction, communication, and behavior. Machine learning techniques have shown promising results in aiding the diagnosis and treatment of ASD. Here are some potential future scopes of using machine learning techniques for ASD.

* **Early Detection:** Machine learning algorithms can analyze large datasets of behavioral and physiological data to identify patterns that may indicate early signs of ASD. With early detection, interventions can be started earlier, leading to improved outcomes for children with ASD.
* **Personalized Treatment:** Machine learning techniques can be used to develop personalized treatment plans for individuals with ASD based on their unique behavioral and genetic profiles. This can lead to more effective treatment outcomes and improved quality of life for those with ASD.
* **Predictive Modeling:** Machine learning algorithms can be used to predict the likelihood of an individual developing ASD based on various risk factors, such as genetics, environmental factors, and behavioral patterns. This can aid in early intervention and prevention efforts.
* **Social Skills Training:** Machine learning algorithms can be used to analyze social interaction patterns and provide feedback to individuals with ASD to improve their social skills. This can be done through the use of virtual reality environments or other computer-based training programs.
* **Assistive Technology:** Machine learning algorithms can be used to develop assistive technology for individuals with ASD, such as speech recognition systems and communication aids. This can improve communication and overall quality of life for those with ASD.

Overall, machine learning techniques hold great potential in aiding the diagnosis and treatment of ASD. With continued research and development, we can expect to see significant advancements in this field in the near future.

**CODE SNIPPETS:**

**Module :1**

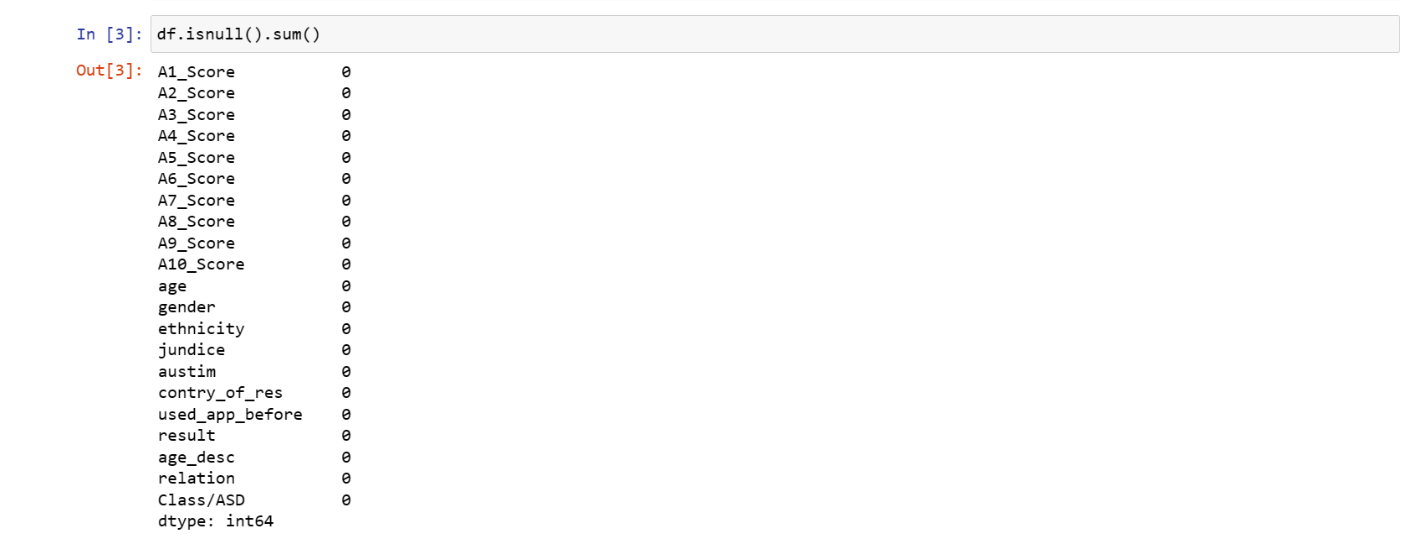
**Data Reading:**

1. df = pd.read\_excel("C:/Users/Admin/Autism child data.xlsx")

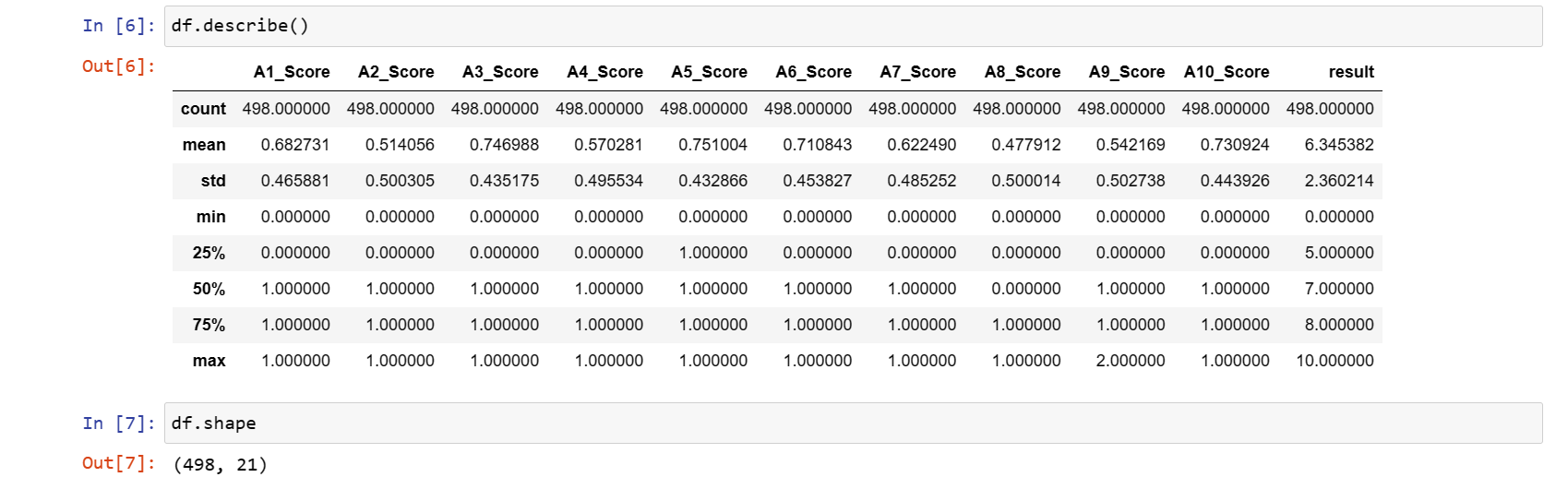
df.head()



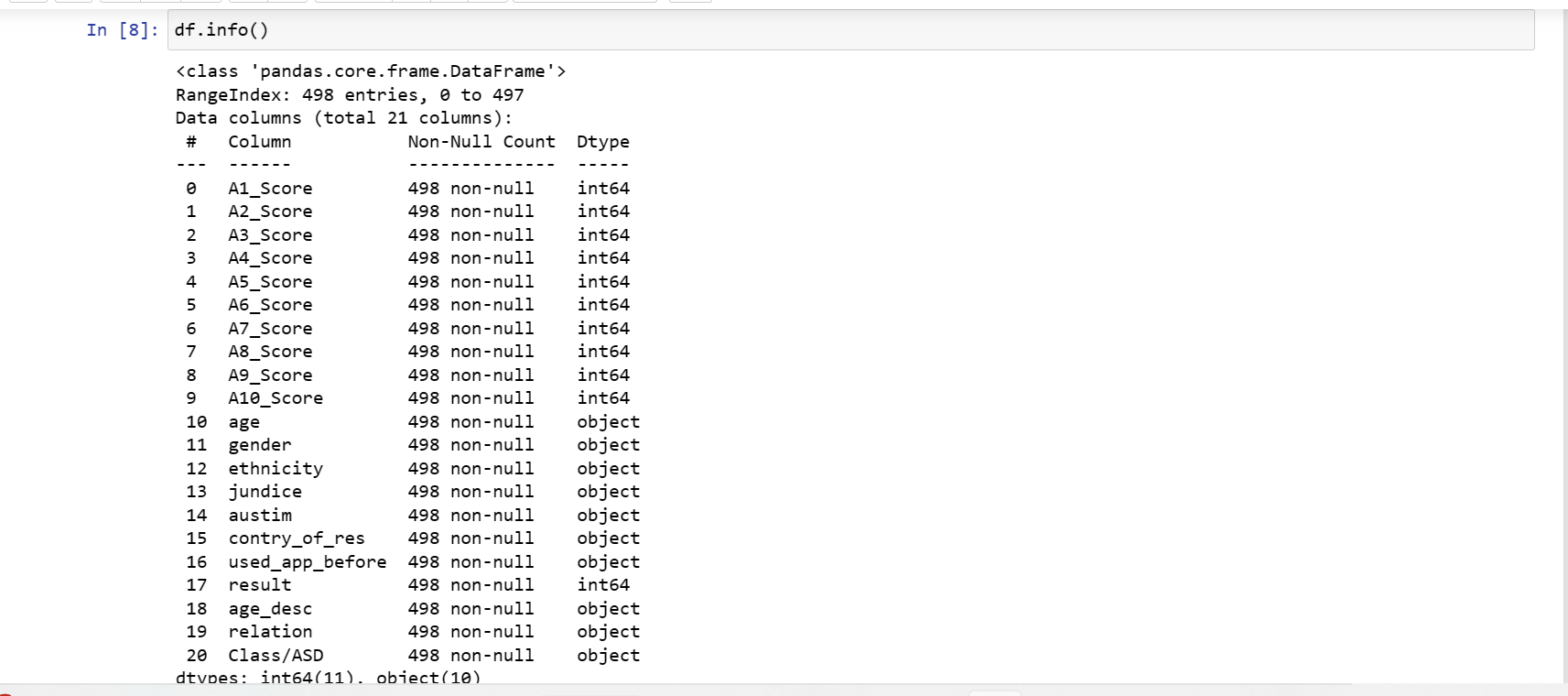
1. df.isnull().sum()



1. df.describe()
2. df.shape

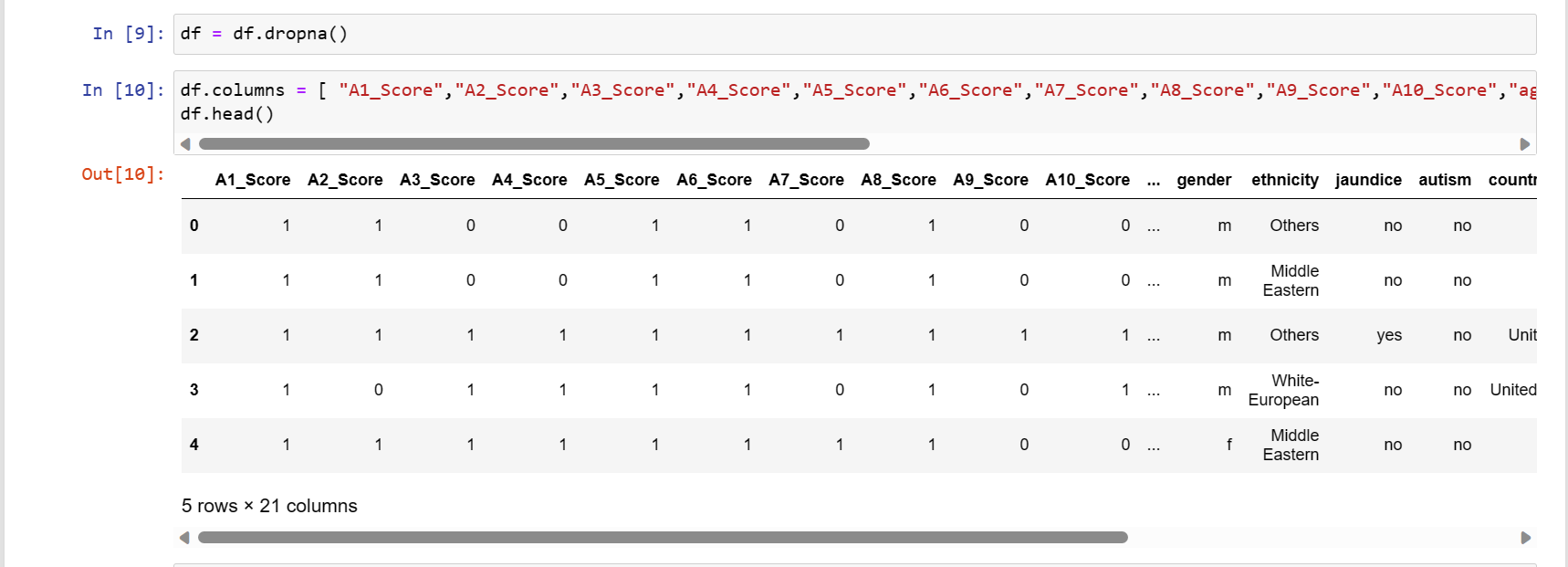


1. df.info()



1. df = df.dropna()
2. df.columns = [ "A1\_Score","A2\_Score","A3\_Score","A4\_Score","A5\_Score","A6\_Score","A7\_Score","A8\_Score","A9\_Score","A10\_Score","age","gender","ethnicity","jaundice","autism","country\_of\_res","used\_app\_before","result","age\_desc","relation","Class\_ASD",]

df.head()



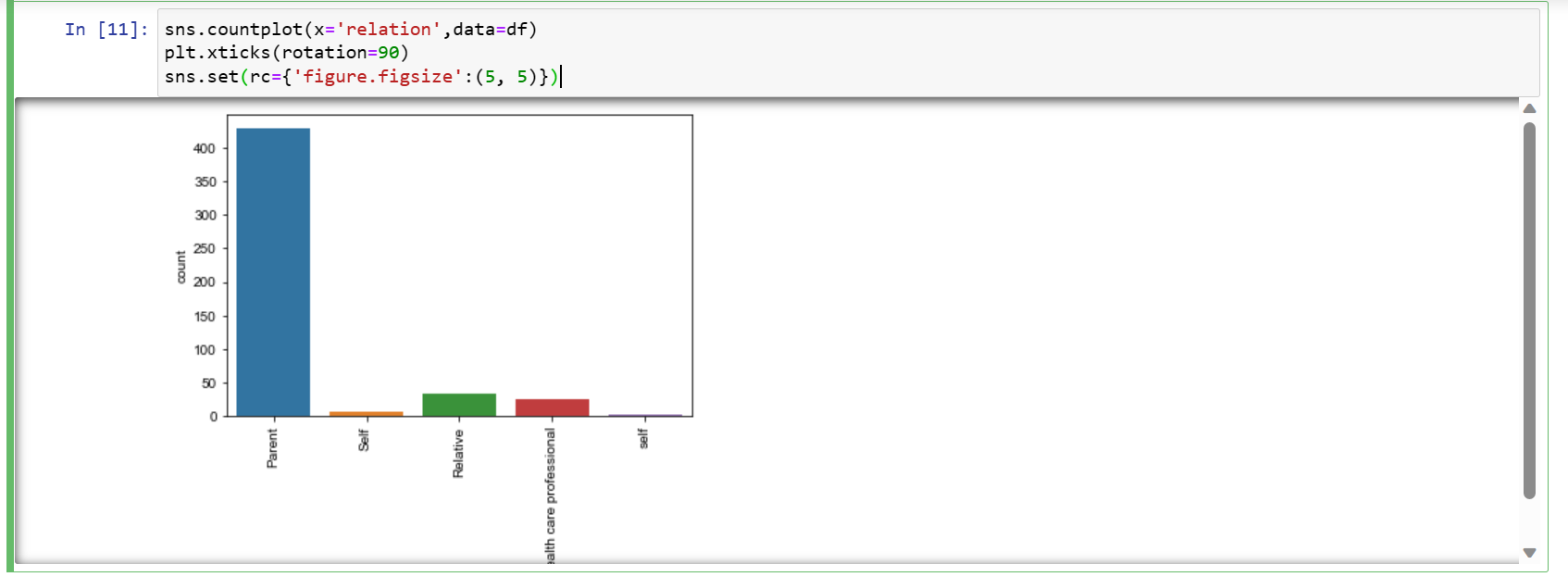
**Module :2**

**Graphical Representation:**

1. sns.countplot(x='relation',data=df)

plt.xticks(rotation=90)

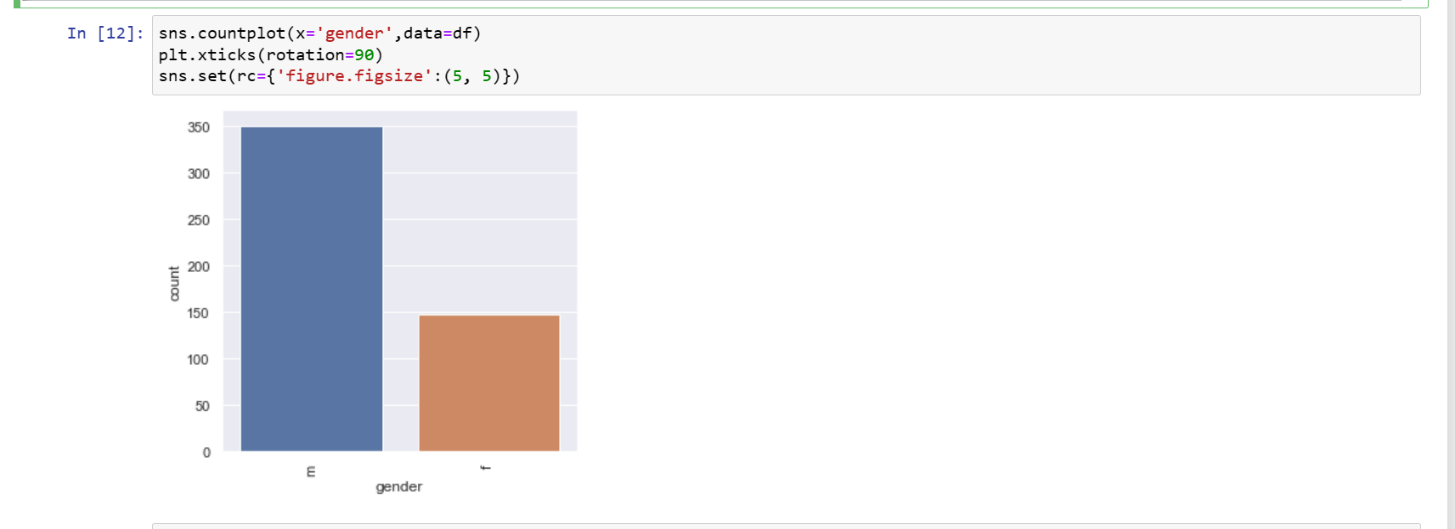
sns.set(rc={'figure.figsize':(5, 5)})



1. sns.countplot(x='gender',data=df)

plt.xticks(rotation=90)

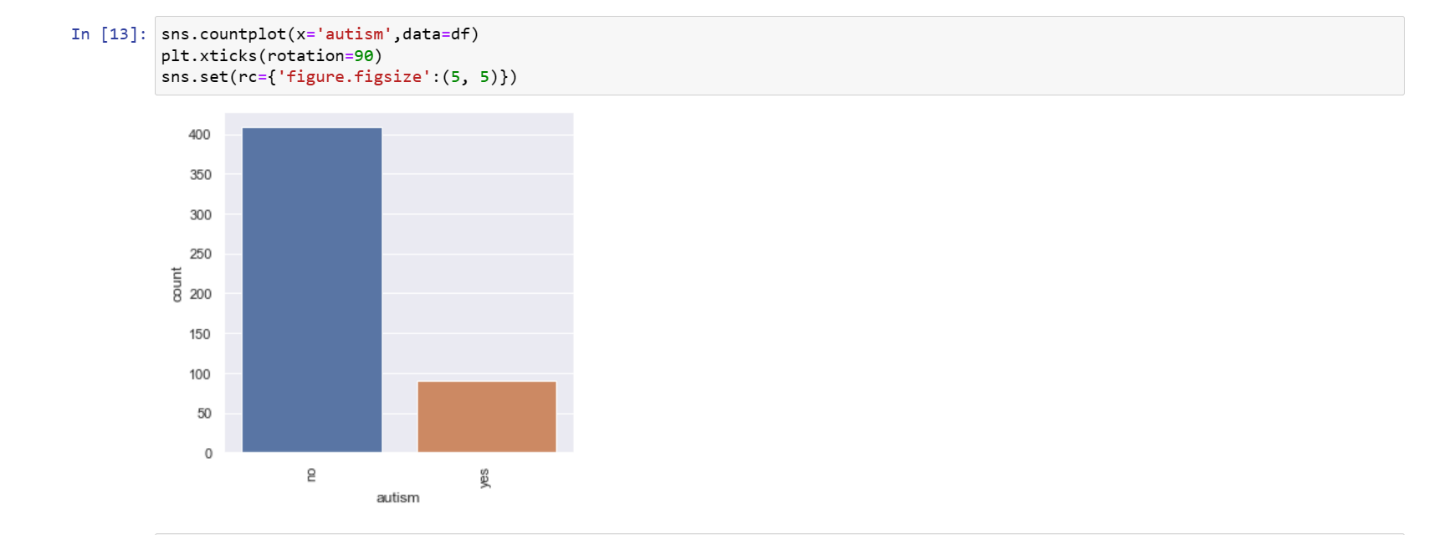
sns.set(rc={'figure.figsize':(5, 5)})



1. sns.countplot(x='autism',data=df)

plt.xticks(rotation=90)

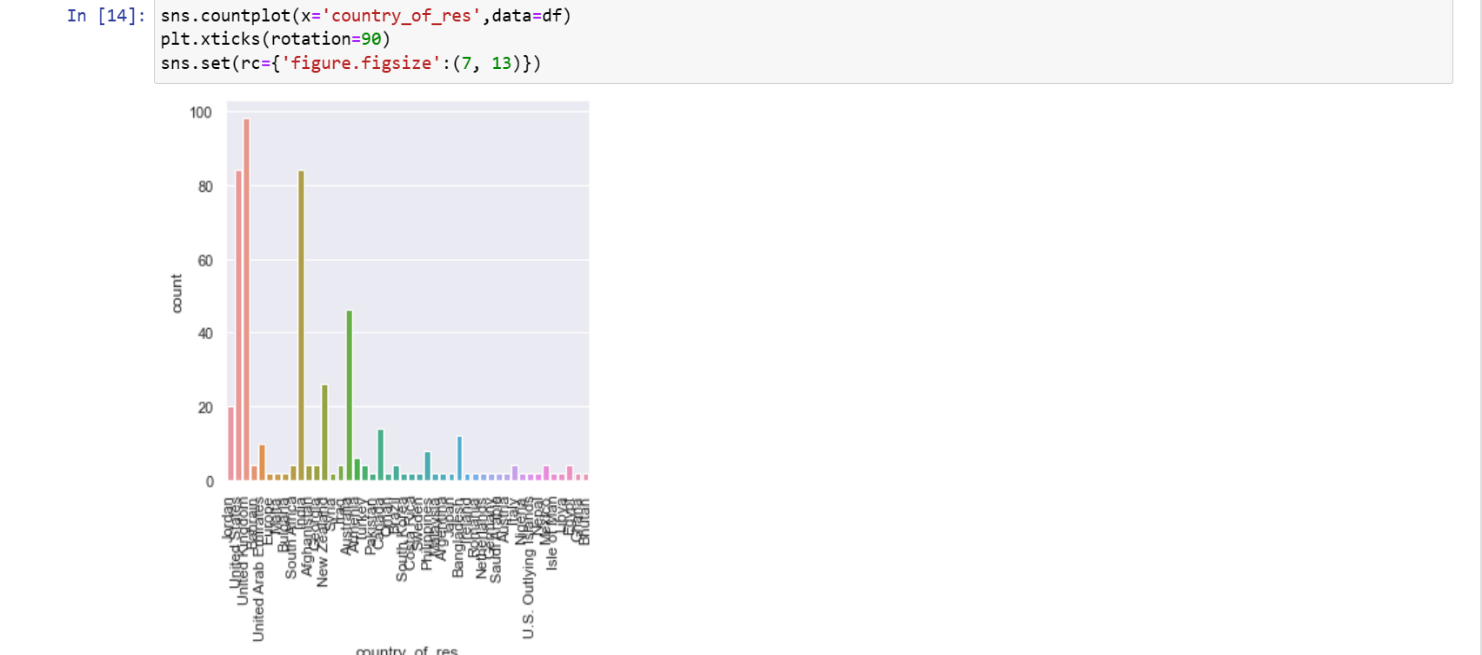
sns.set(rc={'figure.figsize':(5, 5)})



1. sns.countplot(x='country\_of\_res',data=df)

plt.xticks(rotation=90)

sns.set(rc={'figure.figsize':(7, 13)})



1. sns.countplot(x='Class\_ASD',data=df)

plt.xticks(rotation=90)



1. sns.countplot(x='jaundice',data=df)

plt.xticks(rotation=90)

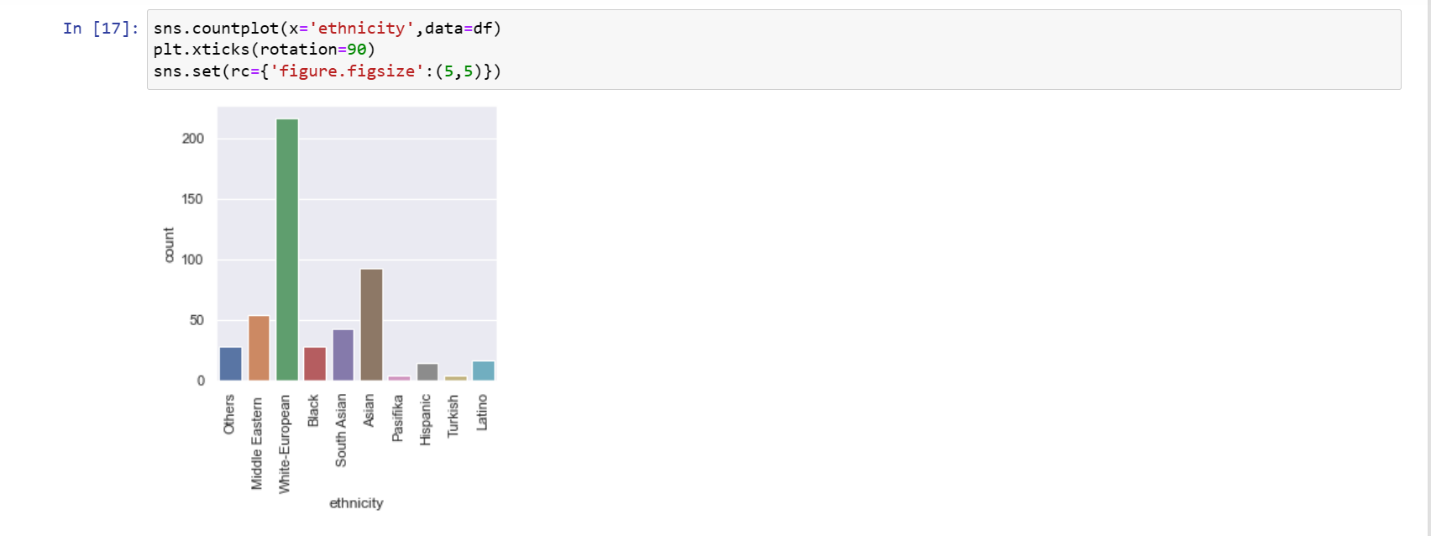
sns.set(rc={'figure.figsize':(4, 4)})



1. sns.countplot(x='ethnicity',data=df)

plt.xticks(rotation=90)

sns.set(rc={'figure.figsize':(5,5)})



**Module :3**

**Data Preprocessing:**

1. df = df[df['age']!='?']
2. org\_data = df[[ "A1\_Score", "A2\_Score", "A3\_Score", "A4\_Score", "A5\_Score", "A6\_Score", "A7\_Score", "A8\_Score", "A9\_Score", "A10\_Score", "age", ]]
3. label\_encoded\_data = df[["gender", "autism", "jaundice", "Class\_ASD"]]

label\_encoded\_data["gender"] = label\_encoded\_data["gender"].apply(lambda x: 1 if x == "m" else 0)

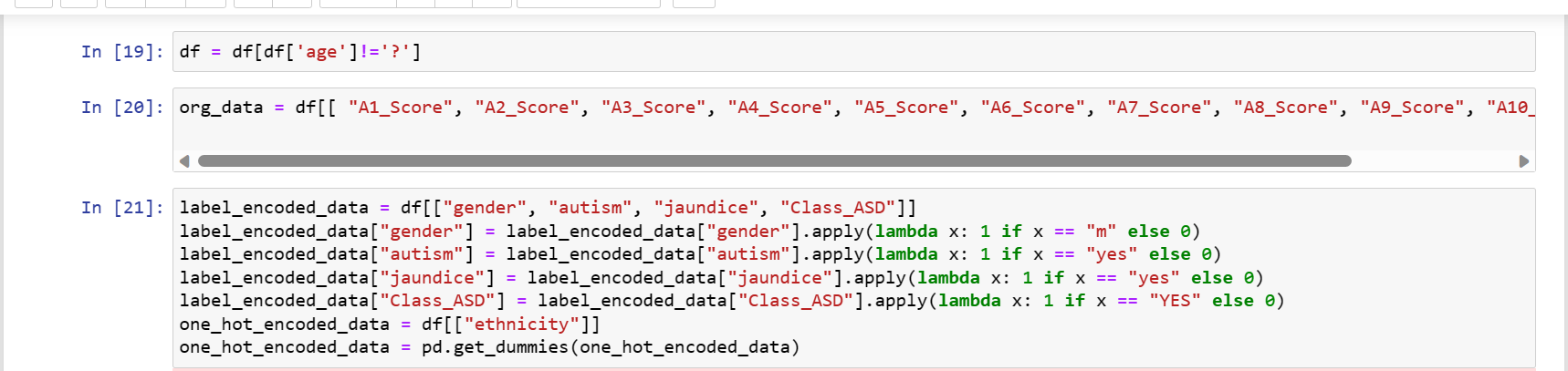
label\_encoded\_data["autism"] = label\_encoded\_data["autism"].apply(lambda x: 1 if x == "yes" else 0)

label\_encoded\_data["jaundice"] = label\_encoded\_data["jaundice"].apply(lambda x: 1 if x == "yes" else 0)

label\_encoded\_data["Class\_ASD"] = label\_encoded\_data["Class\_ASD"].apply(lambda x: 1 if x == "YES" else 0)

one\_hot\_encoded\_data = df[["ethnicity"]]

one\_hot\_encoded\_data = pd.get\_dummies(one\_hot\_encoded\_data)



**Module :4**

**Date Train and Test**

1. fixed\_data = pd.concat([org\_data,label\_encoded\_data,one\_hot\_encoded\_data],axis=1)

fixed\_data.head()

1. X = fixed\_data.drop(columns=['Class\_ASD'])

y = fixed\_data[['Class\_ASD']]

1. X\_train, X\_test, y\_train, y\_test = train\_test\_split( X, y, test\_size=0.3, random\_state=42)
2. X\_test.shape
3. y\_train.shape



1. lr=LogisticRegression()

lr.fit(X\_train,y\_train)

y\_pred=lr.predict(X\_test)

lraccuracy=accuracy\_score(y\_test, y\_pred)

lgiraccuracy

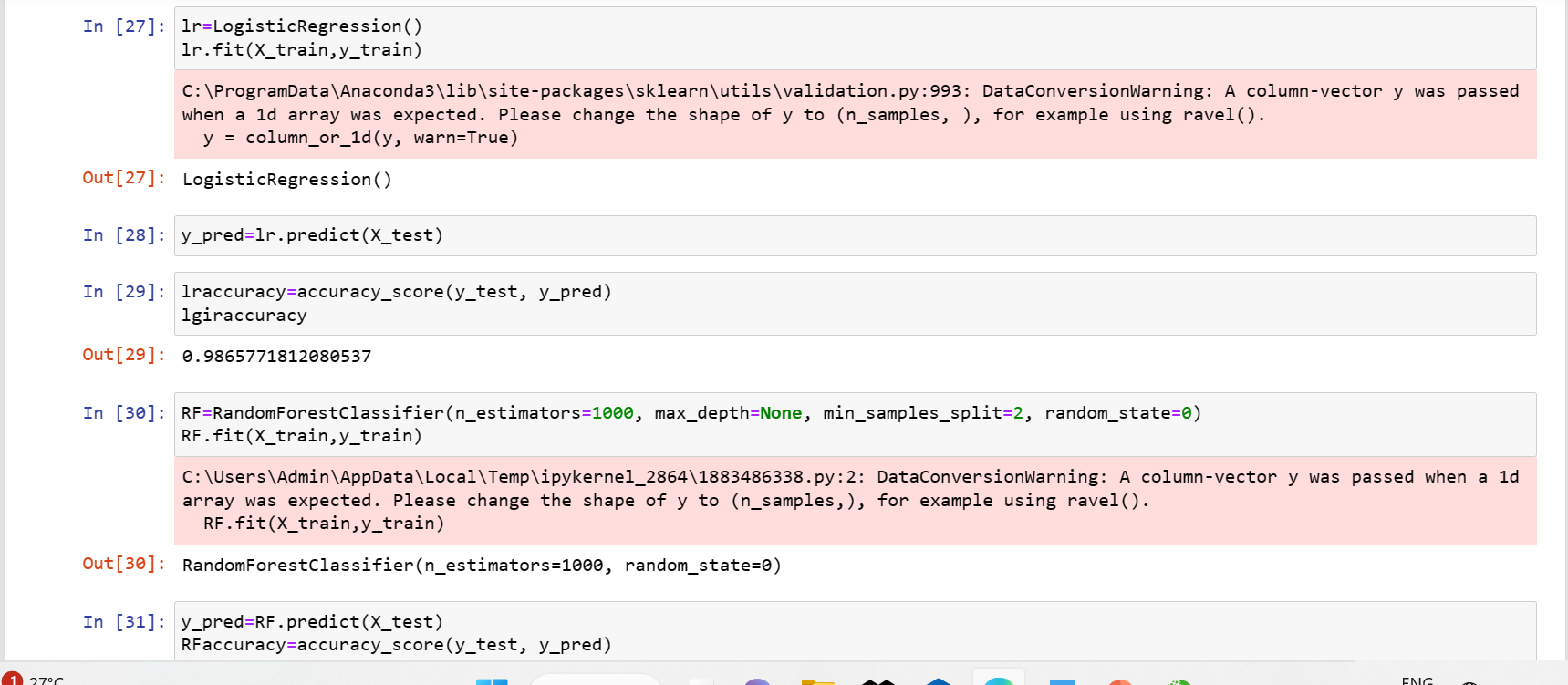
1. RF=RandomForestClassifier(n\_estimators=1000, max\_depth=None, min\_samples\_split=2, random\_state=0)

RF.fit(X\_train,y\_train)

y\_pred=RF.predict(X\_test)

RFaccuracy=accuracy\_score(y\_test, y\_pred)

RFaccuracy



1. DT=DecisionTreeClassifier(max\_depth=None, min\_samples\_split=2, random\_state=0)

DT.fit(X\_train,y\_train)

y\_pred=DT.predict(X\_test)

DTaccuracy=accuracy\_score(y\_test, y\_pred)

DTaccuracy

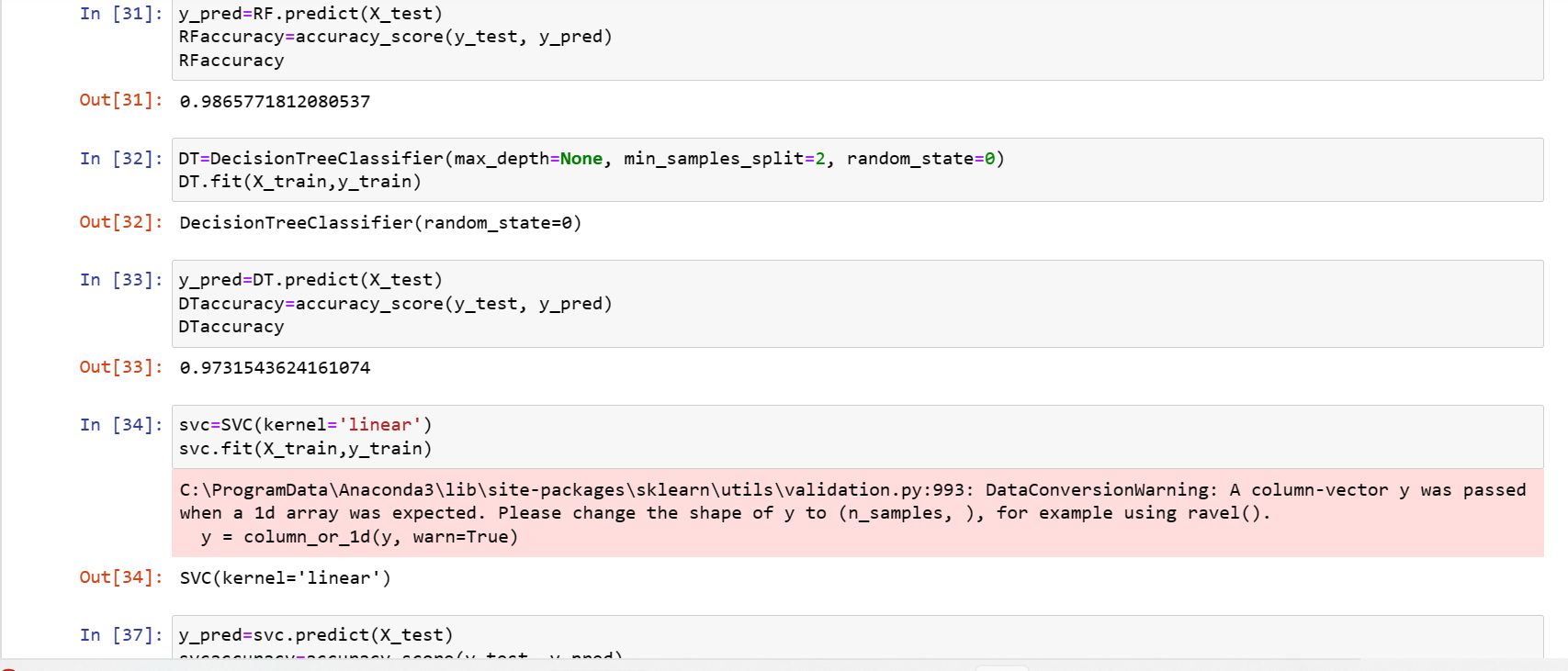
1. svc=SVC(kernel='linear')

svc.fit(X\_train,y\_train)

y\_pred=svc.predict(X\_test)

svcaccuracy=accuracy\_score(y\_test, y\_pred)

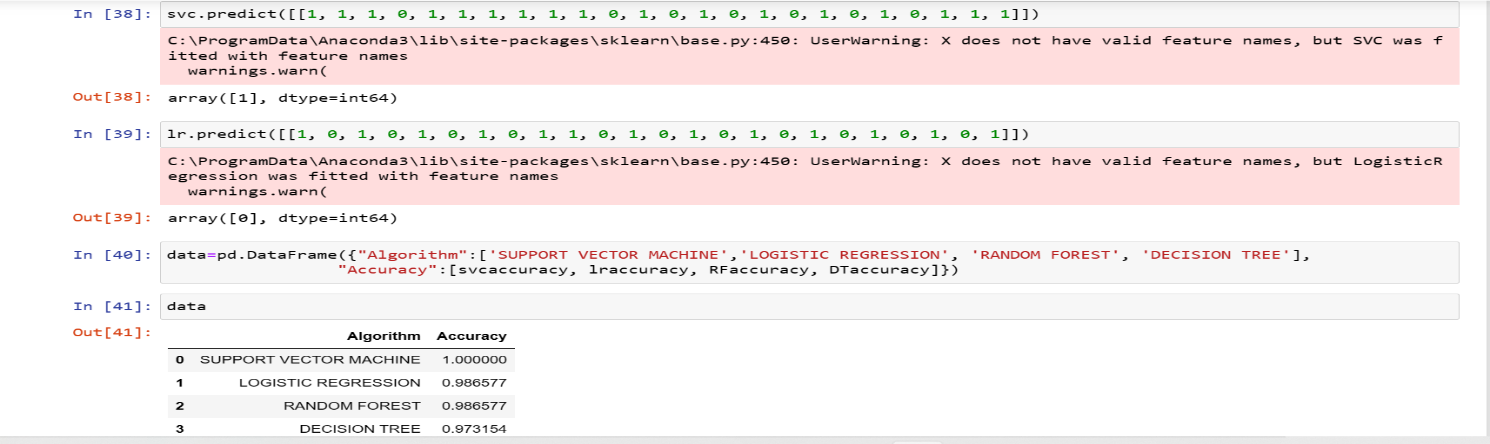
svcaccuracy



1. svc.predict([[1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1]])
2. lr.predict([[1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1]])
3. data=pd.DataFrame({"Algorithm":['SUPPORT VECTOR MACHINE','LOGISTIC REGRESSION', 'RANDOM FOREST', 'DECISION TREE'],

"Accuracy":[svcaccuracy, lraccuracy, RFaccuracy, DTaccuracy]})

Data



1. x=data['Algorithm']

y=data['Accuracy']

sns.barplot(x, y, data=data)

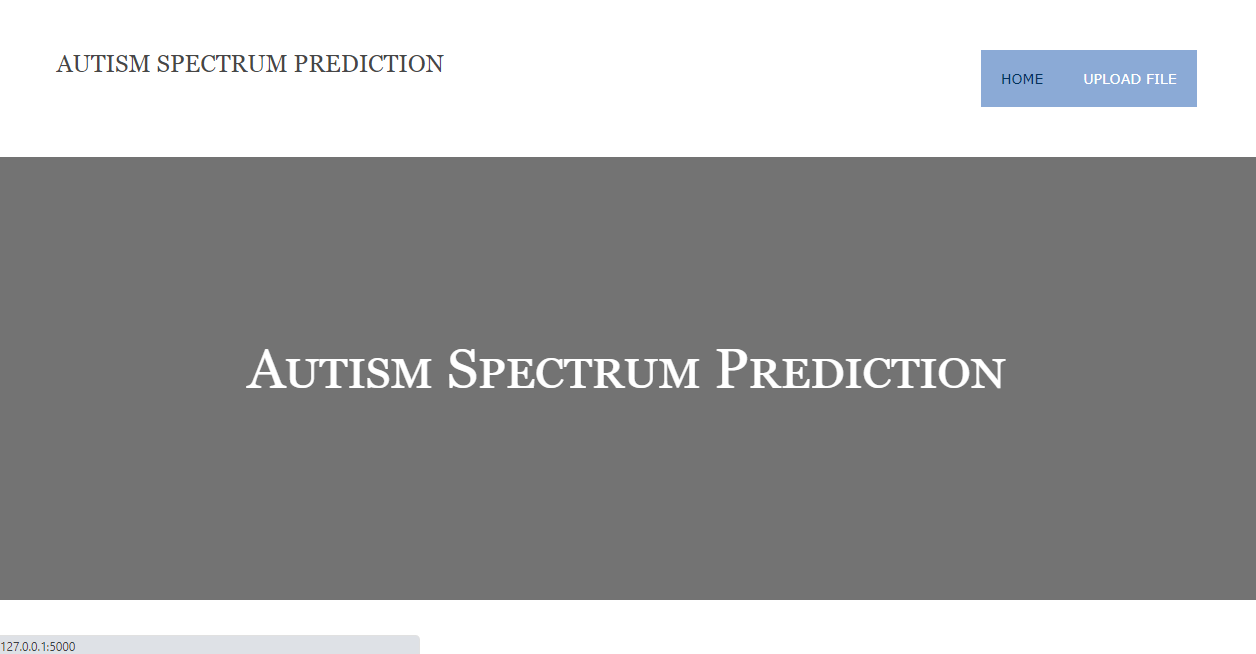
plt.xticks(rotation=90)

sns.set(rc={'figure.figsize':(4, 4)})

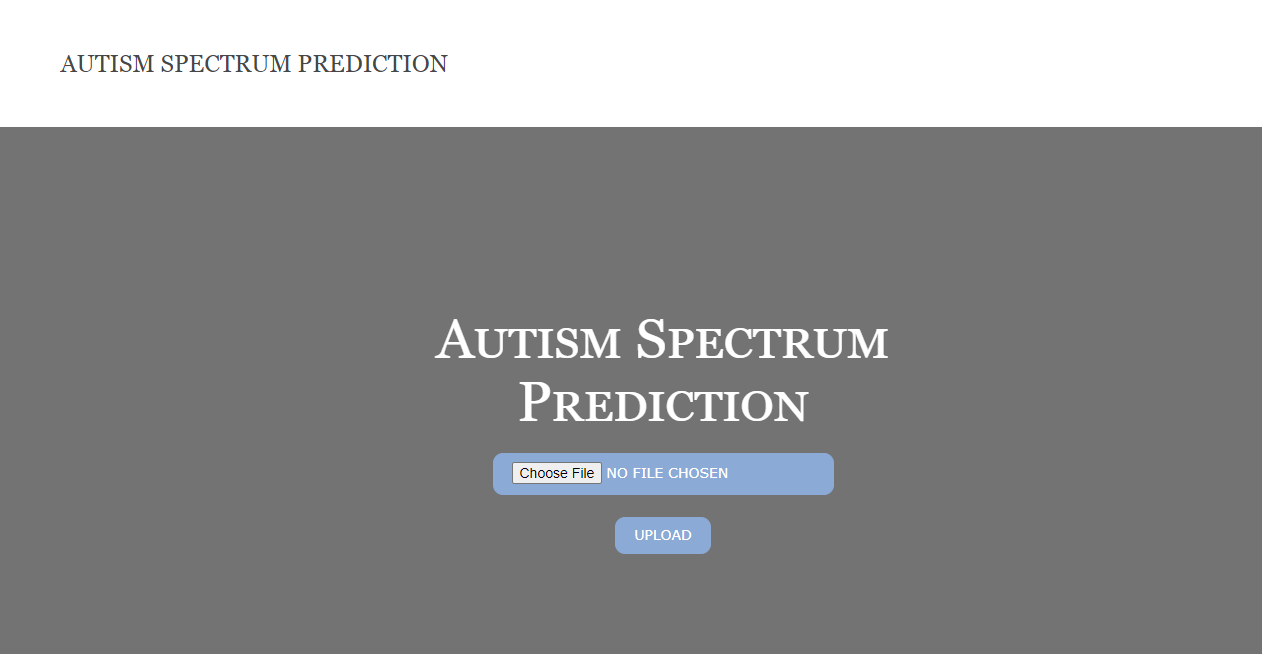


**Results**

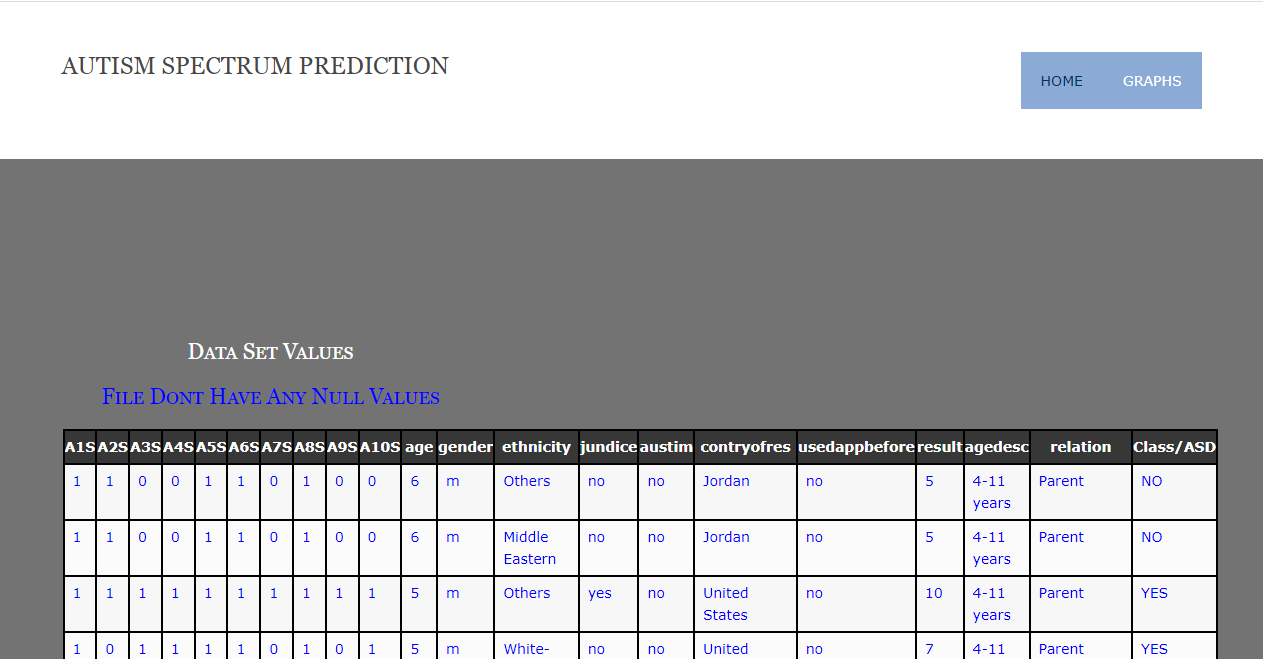
**Home:**

****

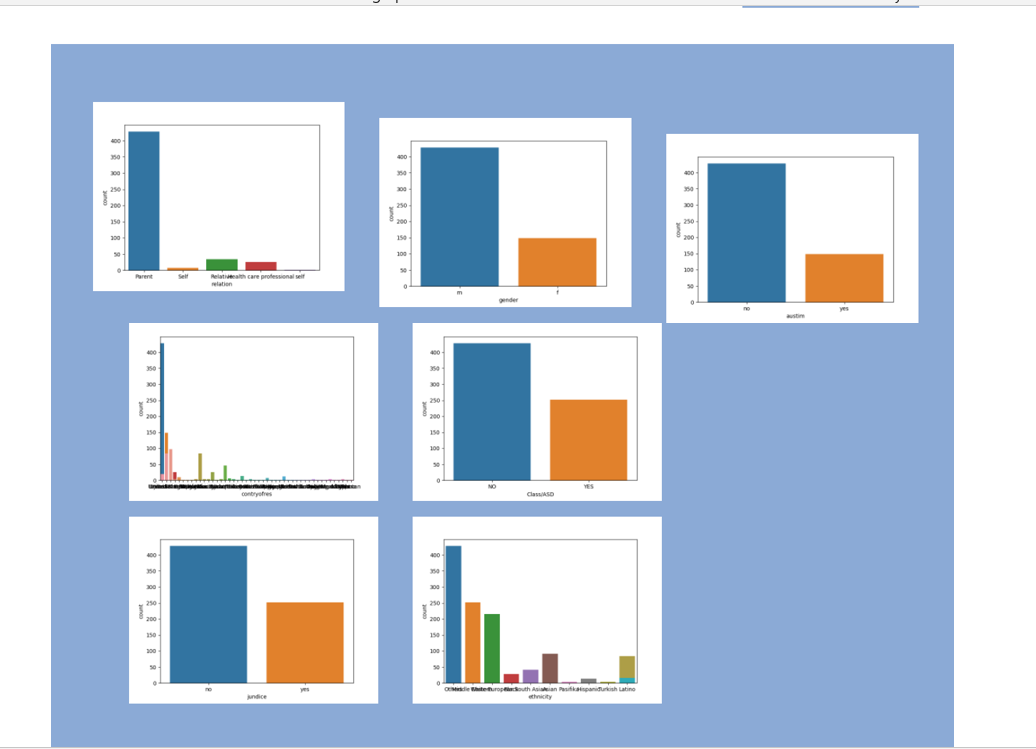
**Upload File:**

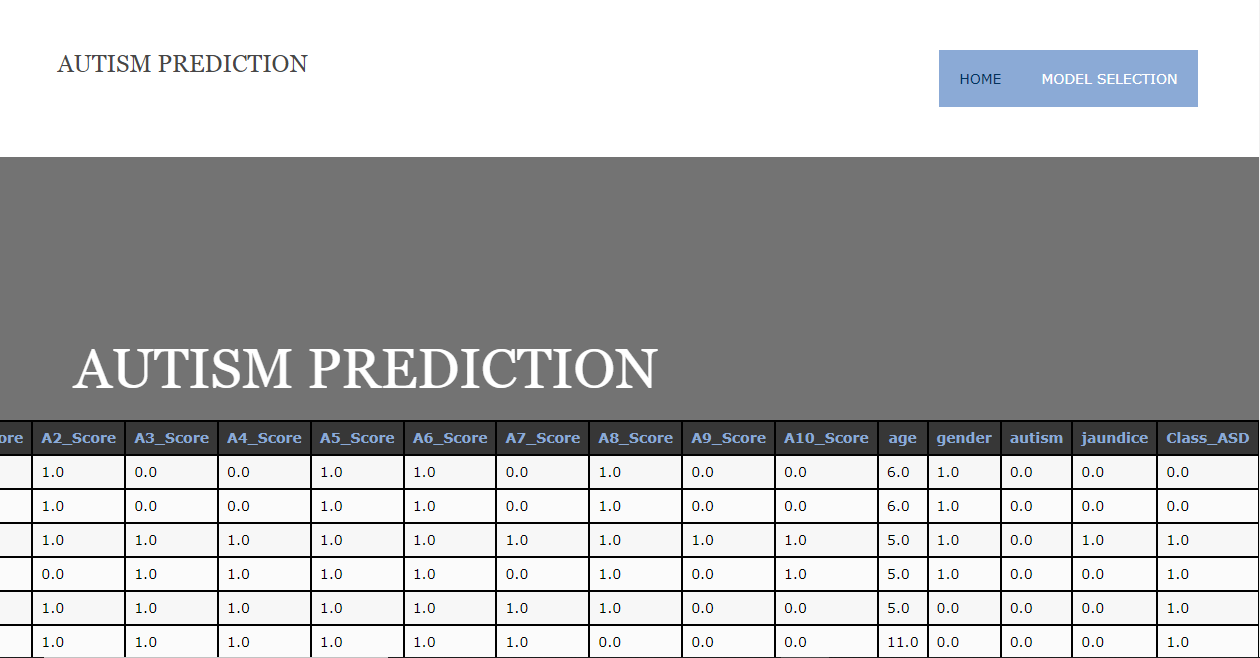
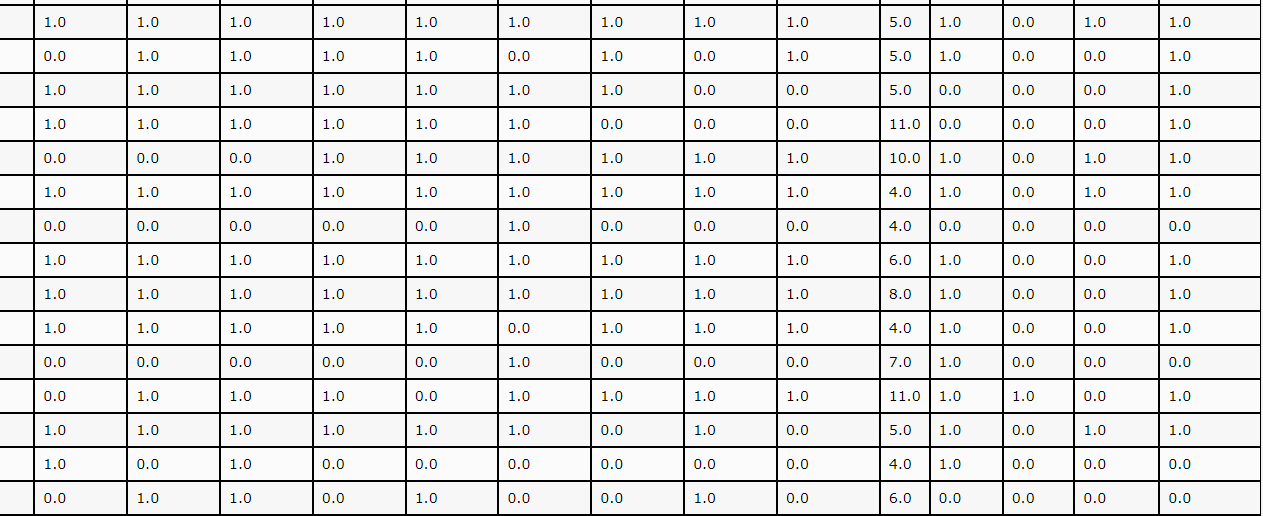
****

**Uploaded File:**

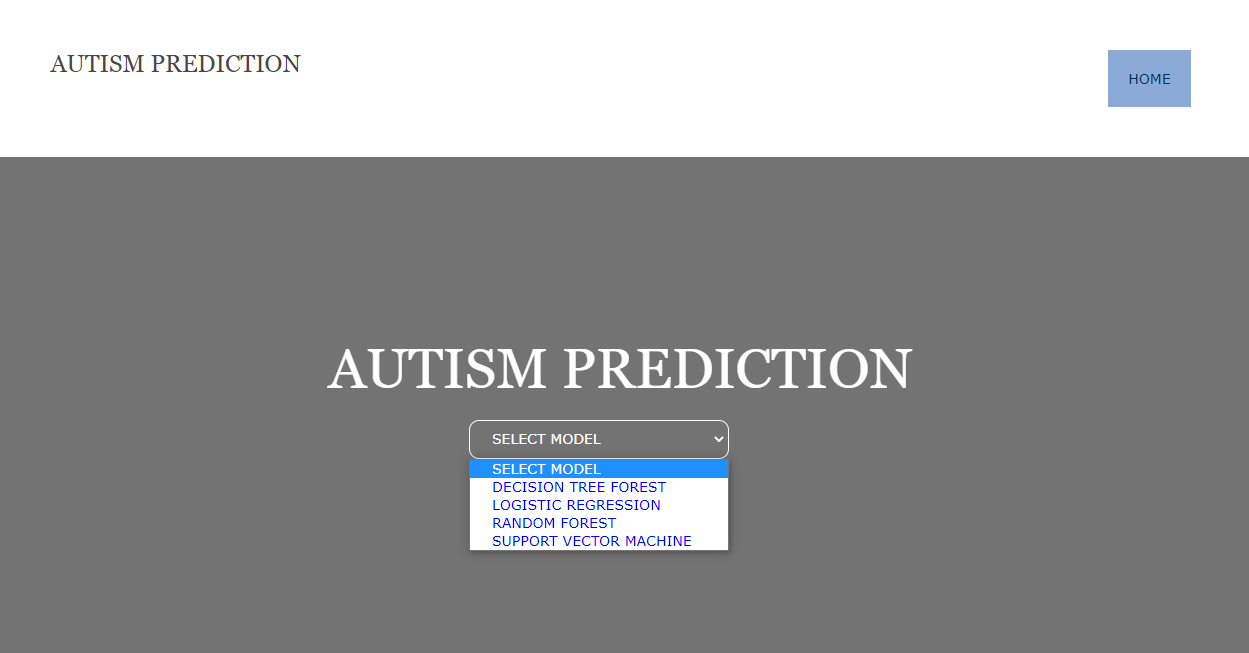


**Graphs:**

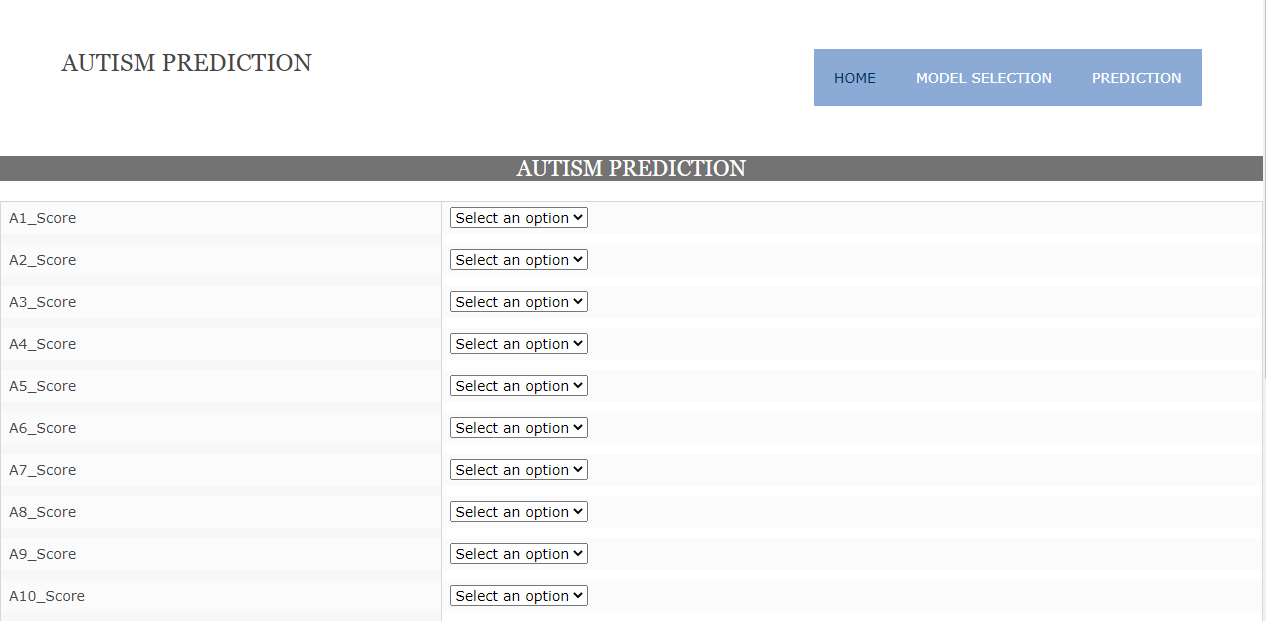
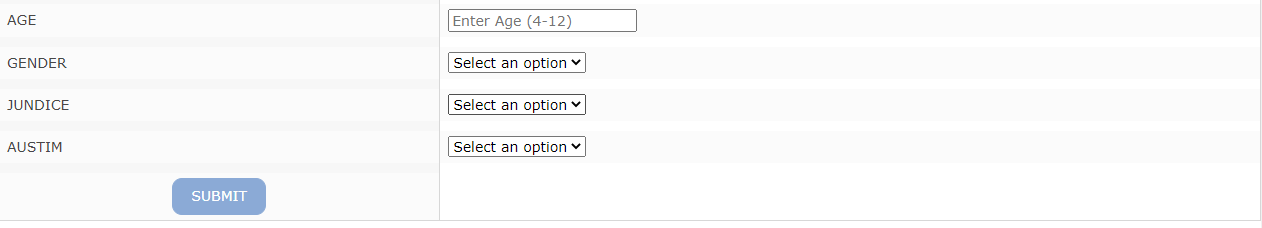


**Preprocessed data:** 

**Model Selection:**



**Prediction:**

**Output-1:**



**Output-2:**

