Autism Prediction

Using Machine Learning

Submitted By

Name: Janani R

Register Number: 125018034

Table of Contents

Abstract

Autism Spectrum Disorder (ASD) is a neurodevelopmental condition that impacts communication, behavior, and social interaction. Early diagnosis and intervention can significantly improve outcomes for individuals with autism. This project explores the use of machine learning models, including Random Forest, Logistic Regression, Support Vector Machine (SVM), and Neural Network (MLP), to predict ASD based on behavioral responses and demographic information. The dataset utilized comprises responses from the Autism Quotient (AQ) screening tool along with other demographic data. This study aims to enhance autism screening through machine learning-based methods, facilitating early diagnosis and access to appropriate interventions.

Introduction

Due to its complexity and wide variety of symptoms, accurate diagnosis of Autism Spectrum Disorder (ASD) is a significant challenge. Early diagnosis and treatments are often beneficial for people with autism spectrum disorder (ASD). Machine learning approaches have proven more useful in the prediction of Autism Spectrum Disorder (ASD) as more and more data is evaluated and significant trends are found. This project seeks to leverage machine learning techniques to enhance the accuracy and efficiency of ASD prediction.

The dataset consists of 800 records, each with 22 features capturing behavioral and demographic information. These features include responses to the AQ-10 screening tool - a widely used instrument for assessing autism-related behaviors and other relevant demographic factors such as age, gender, and family history of autism. Accurate and timely ASD diagnosis is essential for early intervention, making predictive models based on such data highly valuable.

The project methodology covers the stages of data exploration, preprocessing (handling missing values, encoding, and scaling), model training, and evaluation. Techniques for balancing the dataset are also employed to ensure robust model performance.Multiple machine learning models - Random Forest, Logistic Regression, Support Vector Machine (SVM), and Neural Network (MLP) - are trained and evaluated on metrics including accuracy, precision, recall, and F1-score.

Results indicate that the Random Forest classifier achieved the highest accuracy at 90.23%, effectively distinguishing individuals with and without ASD. This outcome underscores the potential of machine learning in ASD screening, providing a more accessible and data-driven diagnostic tool.

Related Work

References to Sources:
The project drew insights from various sources, including:

- **Dataset Link**: The Autism Prediction dataset was sourced from Kaggle
- **Tools & Libraries**: Scikit-Learn, Pandas, Seaborn, Imbalanced-learn, Matplotlib Python libraries were employed for model implementation and data processing.
- **Code Reference**: Kaggle, GeeksforGeeks, Medium, ChatGPT

Background & Methodology

(a)Dataset:

The dataset used in this project contains 800 instances, each representing an individual. It includes 22 features that capture both behavioral responses from the AQ-10 screening tool and demographic data. A breakdown of the features is as follows:

- **ID**: ID of the patient
- **A1_Score to A10_Score**: Behavioral scores from the AQ-10 screening tool, each corresponding to an answer from a set of ten questions designed to assess autism-related behaviors.
- **age**: The individual's age.
- **gender**: The gender of the individual.
- **ethnicity**: The individual's ethnic background.
- **jaundice**: Indicates whether the individual had jaundice at birth (Yes/No).
- **autism**: Whether there is a family history of autism (Yes/No).
- **country_of_res**: The country of residence of the individual.
- **used_app_before**: Indicates if the individual has used an app for autism screening before (Yes/No).
- **result**: The final AQ score derived from the screening tool.
- **age_desc**: Age description, such as "child" or "adult."
- **relation**: The relationship of the person who completed the test on behalf of the individual (e.g., parent, self).

The target variable is **Class/ASD**, which is a binary value where 0 represents no autism and 1 indicates a positive autism diagnosis.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800 entries, 0 to 799
Data columns (total 22 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   ID               800 non-null    int64
 1   A1_Score         800 non-null    int64
 2   A2_Score         800 non-null    int64
 3   A3_Score         800 non-null    int64
 4   A4_Score         800 non-null    int64
 5   A5_Score         800 non-null    int64
 6   A6_Score         800 non-null    int64
 7   A7_Score         800 non-null    int64
 8   A8_Score         800 non-null    int64
 9   A9_Score         800 non-null    int64
 10  A10_Score        800 non-null    int64
 11  age              800 non-null    float64
 12  gender           800 non-null    object
 13  ethnicity        800 non-null    object
 14  jaundice         800 non-null    object
 15  austim           800 non-null    object
 16  contry_of_res    800 non-null    object
 17  used_app_before  800 non-null    object
 18  result           800 non-null    float64
 19  age_desc         800 non-null    object
 20  relation         800 non-null    object
 21  Class/ASD        800 non-null    int64
dtypes: float64(2), int64(12), object(8)
memory usage: 137.6+ KB
```

(b)Data Preprocessing:

Data preprocessing is a critical step in preparing our dataset for modeling. It includes handling missing values, encoding categorical variables, dealing with class imbalance, and scaling features to ensure effective model performance.

1.Handling Missing Values:

The dataset did not have any missing values, so no imputation or removal of instances was necessary.

2.Handling Categorical Values:

To ensure uniformity in categorical data, replace inconsistent labels such as “ ? ” and “ others " with a unified label “Others”. Next, drop the ID column, as it does not contribute to predictive modeling.

3.Encoding Categorical Features:

Utilize LabelEncoder to convert categorical variables into numeric format, enabling the machine learning models to interpret these features. The following columns are encoded: gender, ethnicity, jaundice, austim, contry_of_res, used_app_before, age_desc, and relation.

4.Dealing with Class Imbalance:

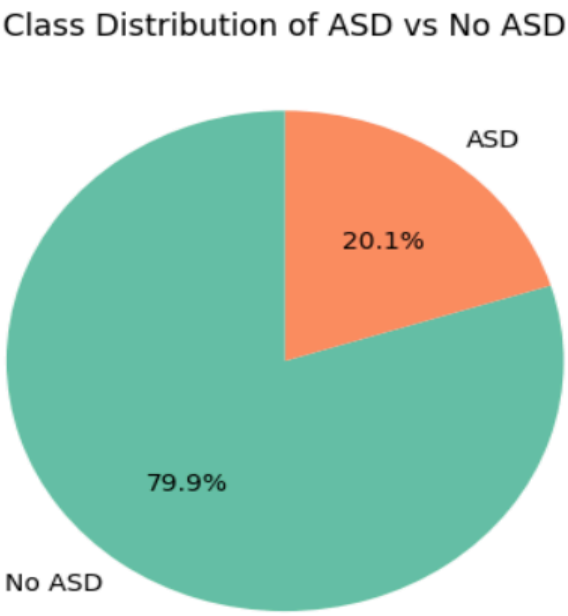
The dataset exhibits class imbalance, which can negatively impact model performance. To address this, we apply the Synthetic Minority Over-sampling Technique (SMOTE), generating synthetic samples of the minority class to achieve a balanced dataset.

5.Feature Scaling:

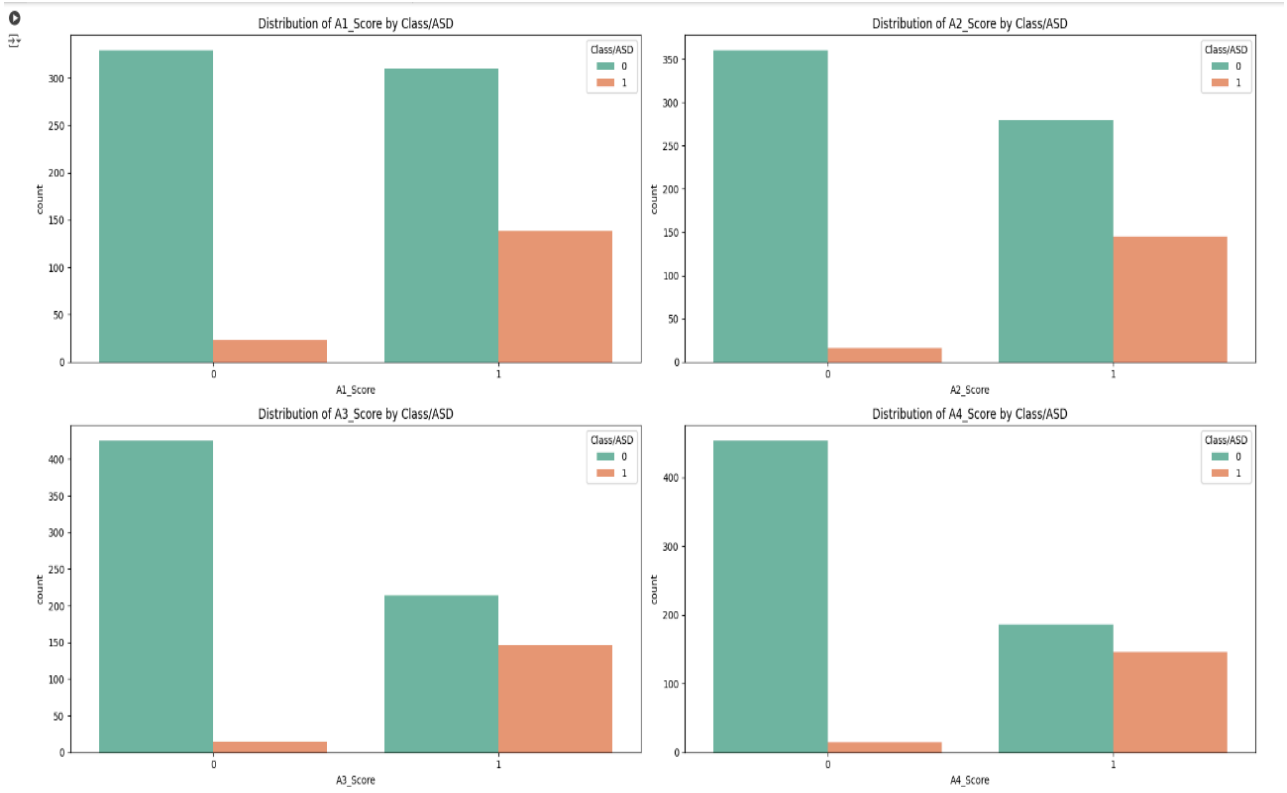
Feature scaling is performed to normalize the range of independent variables. This is particularly important for models sensitive to feature magnitudes, such as Support Vector Machines and Neural Networks. We use StandardScaler to standardize features by removing the mean and scaling to unit variance.

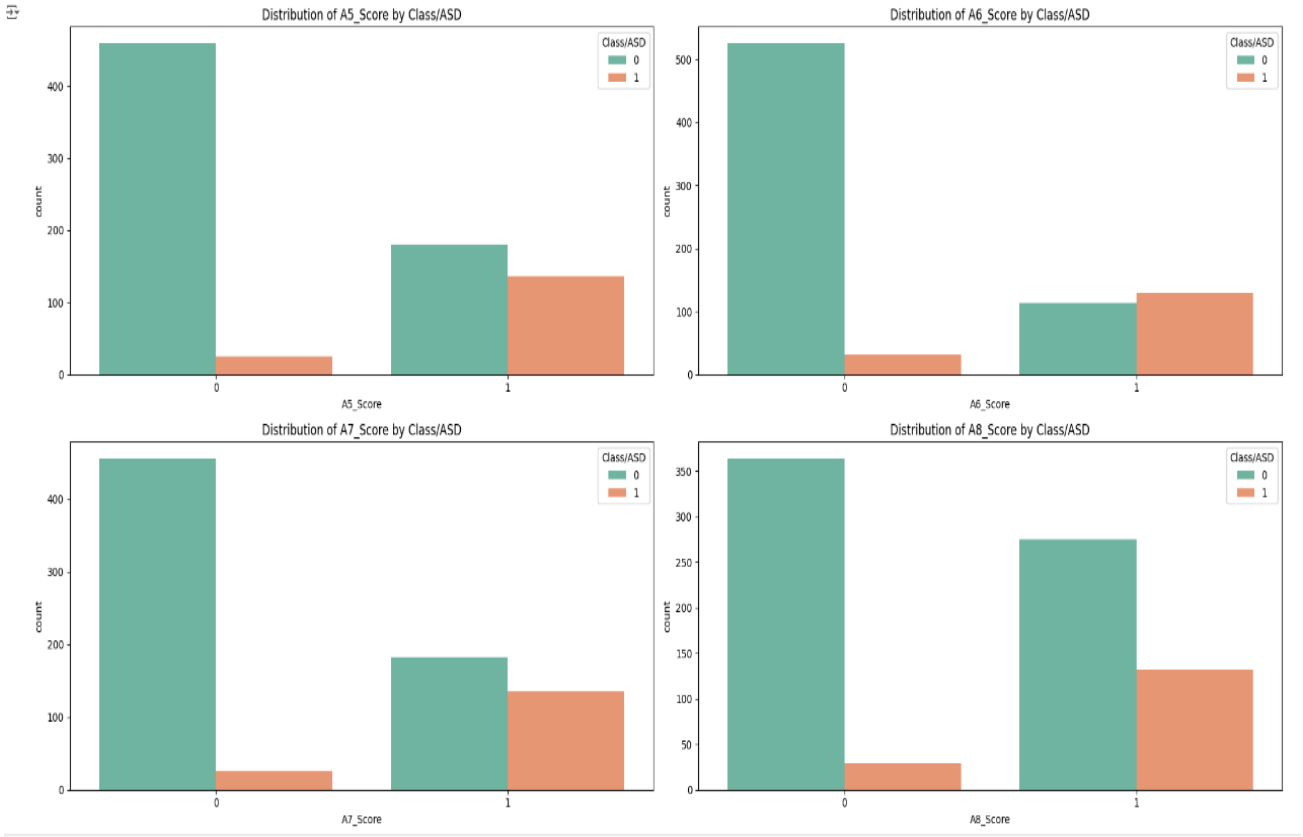
(c)EDA:

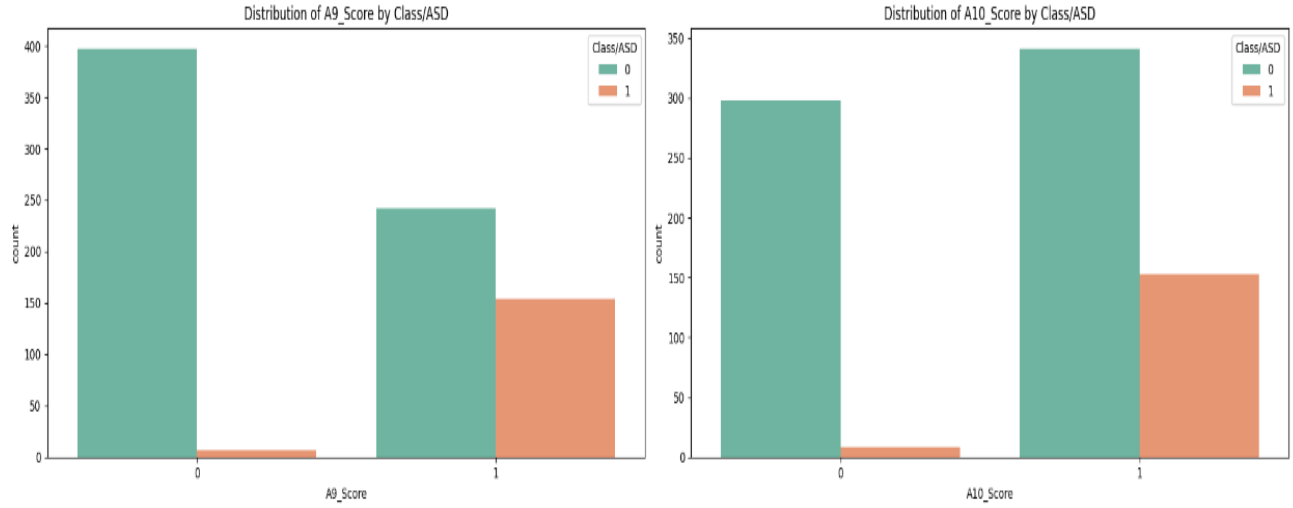
Visualization of the target variable - Class ASD



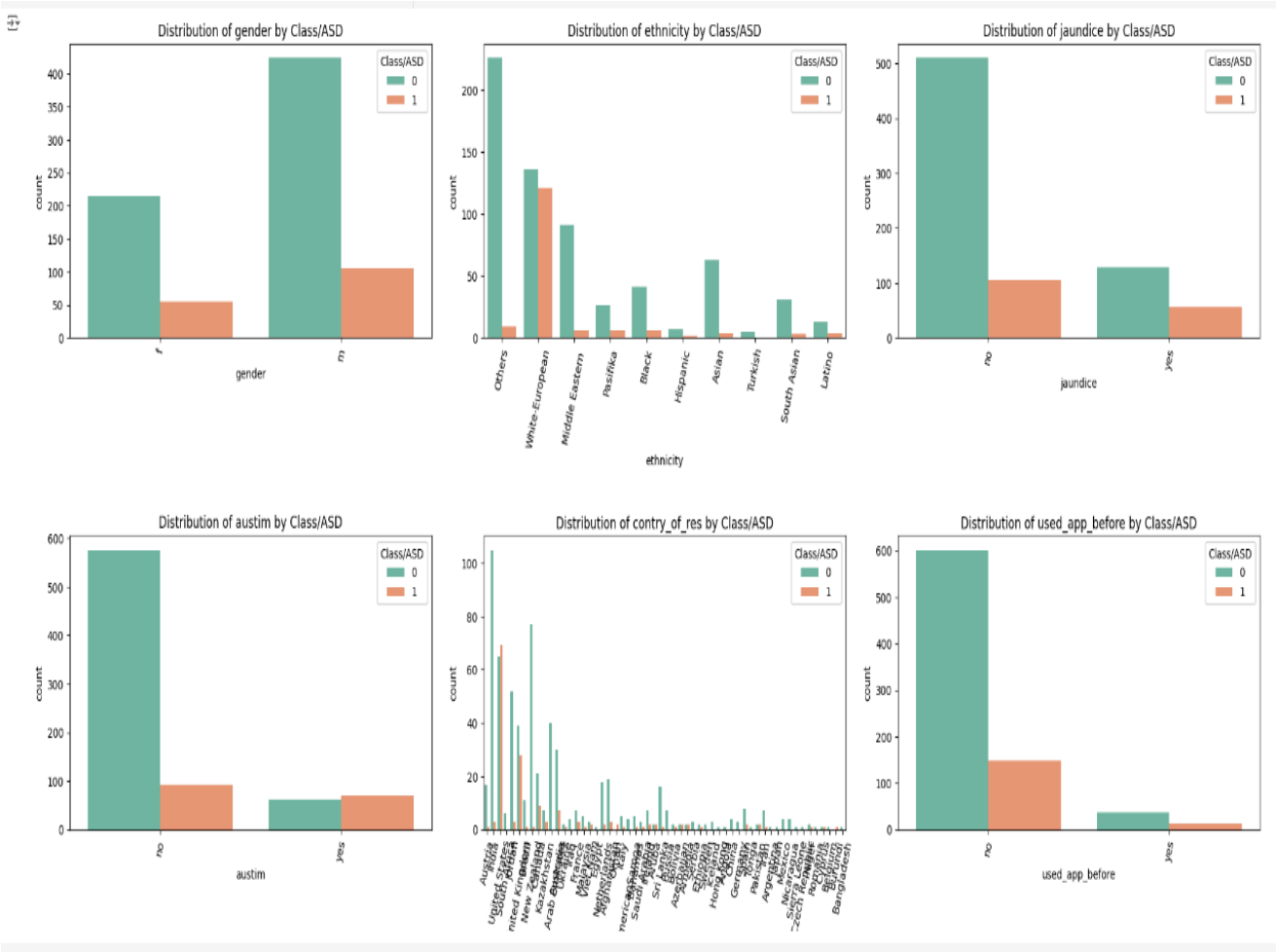
Visualization of Numerical Columns

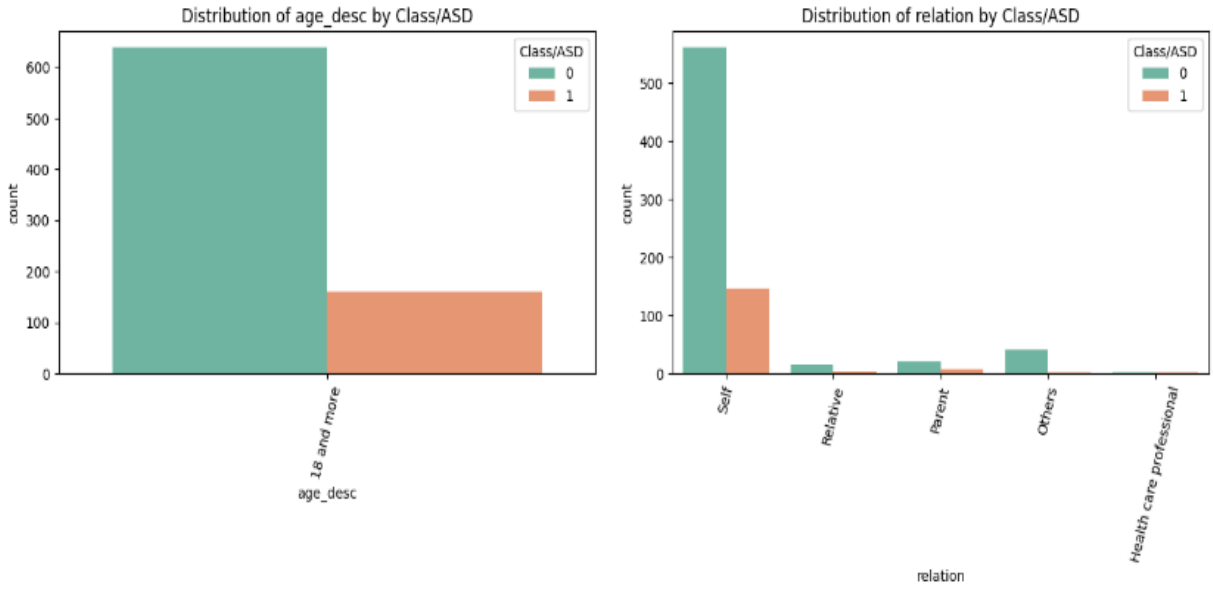



Distribution of A5_Score by Class/ASD
Distribution of A6_Score by Class/ASD
Distribution of A7_Score by Class/ASD
Distribution of A8_Score by Class/ASD
Class/ASD
0
1
count
A5_Score
A6_Score
A7_Score
A8_Score

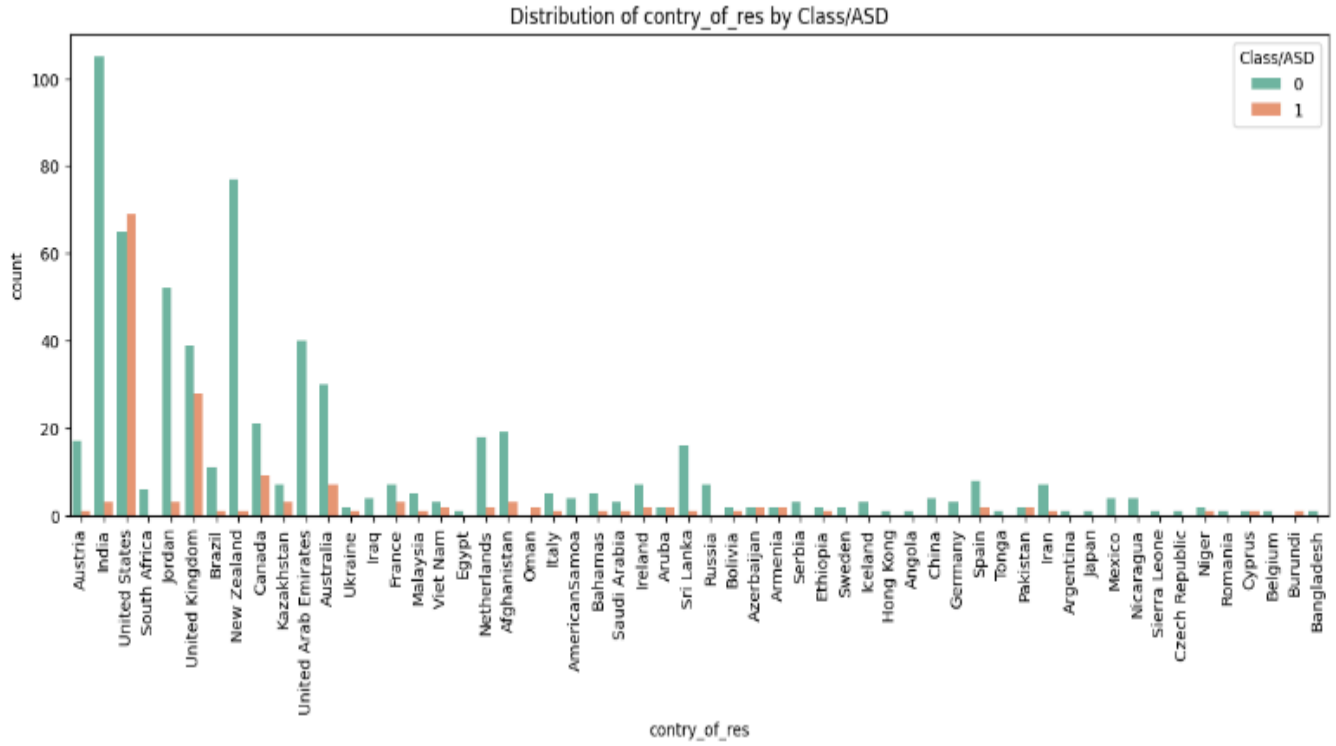

Distribution of A9_Score by Class/ASD
Distribution of A10_Score by Class/ASD
Class/ASD
0
1
count
A9_Score
A10_Score

Visualization of Categorical Columns

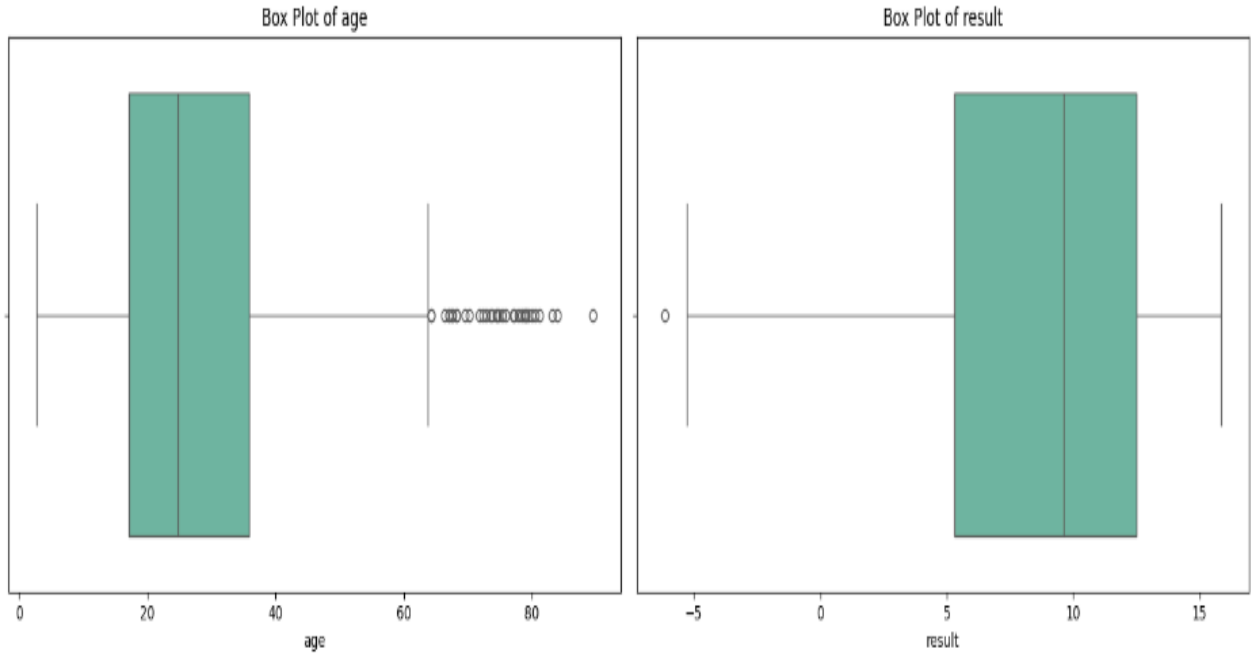




Visualization of Country of Residence



Visualization of Continuous Data



(d) Experimental Design:

1. **Data Loading and Initial Exploration**: Load the dataset and perform an initial exploration to understand its structure and identify any missing values or inconsistencies.
2. **Data Cleaning**: Address any inconsistencies by replacing non-uniform values and dropping non-essential columns to refine the dataset.
3. **Data Visualization**: Visualize the class distribution and analyze features to detect patterns that may enhance model insights.
4. **Feature Encoding, Class Balancing, and Standardization**: Encode categorical features, address class imbalance using SMOTE, and standardize numerical features to prepare the data for modeling.
5. **Data Splitting**: Split the dataset into training and validation sets, with 80% allocated for training and 20% for validation, to support an unbiased model evaluation.
6. **Model Training and Evaluation**: Train Random Forest, Logistic Regression, SVM, and Neural Network classifiers on the training set and evaluate each model's performance on the validation set using accuracy, precision, recall, and F1-score metrics.

(e) Machine Learning Models Used:

1. Random Forest (RF)

Working:
Random Forest builds multiple decision trees using randomly selected subsets of data and features, creating a "forest" of decision trees. Each tree independently makes predictions, and the final output is based on a majority vote (classification) or average (regression) of all tree outputs. This ensemble approach reduces overfitting and improves accuracy by combining multiple weak learners to form a strong model.

Advantages:

- **High Accuracy**: By combining multiple trees, Random Forest improves overall accuracy and reduces overfitting.
- **Versatile**: Works well for both classification and regression tasks.
- **Handles Missing Data**: Can effectively handle missing values and maintains performance.

- **Reduces Overfitting**: Ensemble approach helps in reducing overfitting compared to individual decision trees.

Limitations:

- **Computationally Expensive**: Building multiple trees requires significant computational resources and memory.
- **Lack of Interpretability**: Harder to interpret than single decision trees, as predictions are the result of multiple trees.
- **Longer Prediction Time**: Due to many trees, prediction may take longer, which may be a drawback for real-time applications.

Use Cases:

- **Medical Diagnosis**: For predicting diseases or medical conditions, such as breast cancer detection.
- **Customer Churn Prediction**: Used in banking and telecom to identify customers likely to leave.
- **Fraud Detection**: Widely used in financial services to detect fraudulent transactions.

2. Logistic Regression (LR)

Working:
Logistic Regression is a linear model for binary classification that calculates probabilities using the logistic (sigmoid) function. It computes a weighted sum of the input features, applies the sigmoid function to output probabilities between 0 and 1, and uses a threshold (often 0.5) to classify instances.

Advantages:

- **Simple and Interpretable**: Easy to implement and understand, and its coefficients can offer insights into the importance of features.
- **Efficient for Linearly Separable Data**: Performs well on datasets with a clear separation between classes.
- **Less Prone to Overfitting**: Especially when used with fewer features or regularization techniques.

Limitations:

- **Limited to Linear Boundaries**: Cannot handle complex non-linear relationships without transformation.
- **Sensitive to Outliers**: Outliers can skew the results if not handled.
- **Binary Output**: Generally suited for binary classification, although it can be extended to multiclass problems with techniques like One-vs-All.

Use Cases:

- **Customer Classification**: Used for predicting whether a customer will buy a product or churn.
- **Credit Scoring**: Commonly used in finance to estimate the likelihood of loan default.
- **Disease Prediction**: Used in healthcare to predict the presence or absence of a condition.

3. Support Vector Machine (SVM)

Working:
SVM aims to find the hyperplane that best separates data points of different classes with the maximum margin, using only the critical "support vectors." For non-linear cases, SVM can apply kernel functions to transform data into higher dimensions, making linear separation possible. The model includes a soft margin to allow some misclassifications, balancing margin maximization and error minimization.

Advantages:

- **Effective in High-Dimensional Spaces**: Works well for datasets with a large number of features.
- **Versatile with Kernels**: The use of different kernel functions (linear, polynomial, radial) allows SVM to model non-linear relationships.
- **Robust to Overfitting**: SVM focuses on finding the margin, which reduces the risk of overfitting, especially with a well-chosen kernel.

Limitations:

- **Computationally Intensive**: Training an SVM is resource-intensive, especially with large datasets.
- **Sensitive to Choice of Kernel**: The performance heavily relies on the kernel chosen, which can be complex to tune.
- **Less Effective for Noisy Data**: Outliers can negatively affect the model, especially in cases with overlapping classes.

Use Cases:

- **Text Classification**: Widely used in NLP for spam detection and sentiment analysis.
- **Image Classification**: Used for facial recognition and handwriting recognition tasks.
- **Bioinformatics**: Common in identifying proteins and gene classification.

4. Neural Network (MLP)

Working:
Multilayer Perceptron (MLP) is a type of artificial neural network(ANN) which consists of an input layer, hidden layers, and an output layer, where each layer applies a weighted sum and an activation function to its inputs. Through forward propagation, data flows from input to output, and backpropagation adjusts weights by minimizing prediction errors using gradient descent. Neural networks are powerful for learning complex, non-linear relationships and are effective for high-dimensional data.

Advantages:

- **Non-linear Modeling**: Can capture complex patterns in data by using multiple layers and non-linear activation functions.
- **Adaptable and Scalable**: Effective for large, complex datasets, especially with deep architectures.
- **Feature Engineering**: Learns its features from data, reducing the need for manual feature engineering.

Limitations:

- **Requires Large Datasets**: Neural networks need large amounts of labeled data to generalize well.
- **High Computational Cost**: Training can be computationally expensive, often requiring specialized hardware (e.g., GPUs).
- **Prone to Overfitting**: Especially in the absence of sufficient data or regularization techniques.

Use Cases:

- **Image and Video Recognition**: Popular in fields like computer vision for object detection and image classification.
- **Natural Language Processing (NLP)**: Used for machine translation, sentiment analysis, and text generation.
- **Speech Recognition**: Widely applied in voice-activated systems and automated speech transcription.

(f) Tools and Environment:

- Python programming language was used, along with Google Colab for the development environment.
- Key libraries included Scikit-Learn, Pandas, Seaborn, Matplotlib and Imbalanced-learn.

Results & Discussions

Model Accuracies:

Model	Accuracy(%)
Random Forest Classifier	90.23
Neural Network (MLP)	87.8
SVM	87.5
Logistic Regression	82.8

In evaluating the performance of the various machine learning models, each model demonstrated different strengths and limitations. Here is an in-depth discussion of the results for each model.

1. Random Forest: Achieving the highest accuracy at 90.23%, the Random Forest classifier demonstrated superior predictive power for ASD. This model's ensemble approach combines multiple decision trees to capture complex relationships in the data, minimizing overfitting risks while balancing model interpretability. Additionally, its robustness with both categorical and numerical features makes it well-suited to the dataset, positioning Random Forest as an effective model for distinguishing between individuals with and without ASD.

2. Neural Network (MLP): The MLP classifier achieved an accuracy of 87.8%, performing well due to its layered architecture, which allows it to learn intricate patterns in the dataset. While MLP's performance is comparable to SVM, it benefits from the flexibility to model non-linear relationships inherent in the data.Training neural networks can also require more computational power and time, which can be a limitation for larger datasets or those requiring extensive tuning.

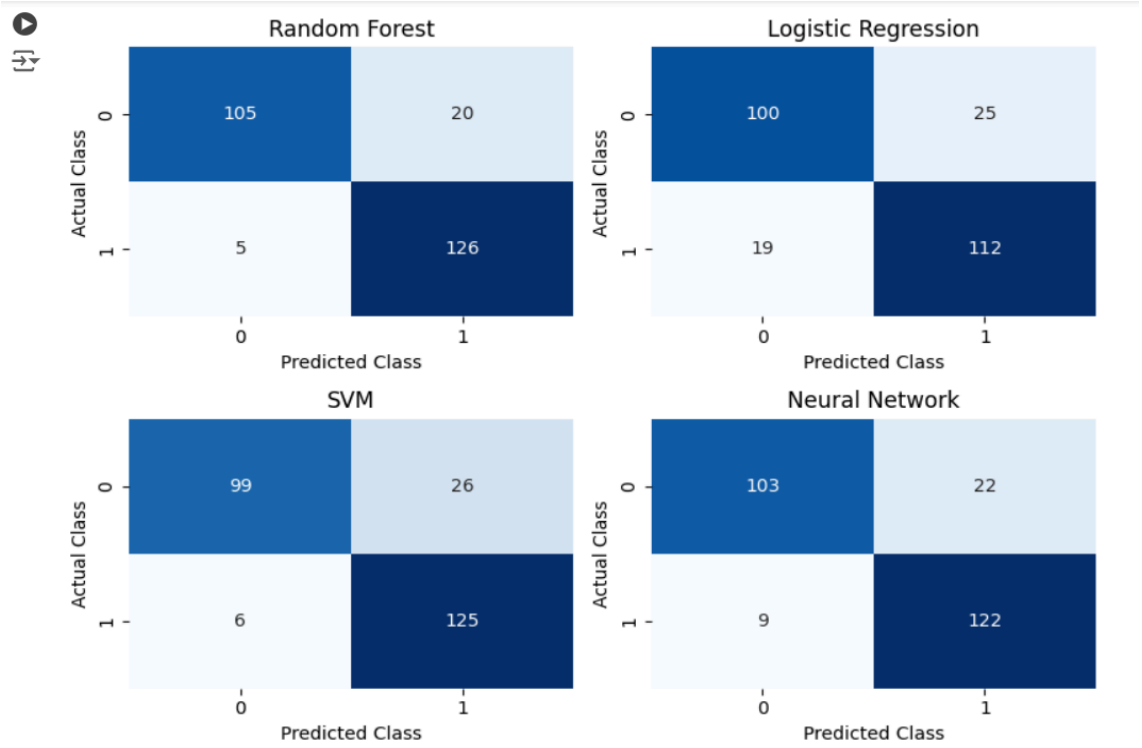
3. Support Vector Machine (SVM): With an accuracy of 87.5%, the SVM model performed closely to MLP. Known for its ability to manage high-dimensional spaces, SVM leverages optimal hyperplane selection to distinguish ASD-related patterns effectively. While highly reliable, SVM models can be computationally expensive, especially with complex kernels.

4. Logistic Regression: Scoring an accuracy of 82.8%, Logistic Regression, did not perform as well as the other models. Although it is simple and interpretable, Logistic Regression may struggle with datasets where the relationship between features and the target variable is nonlinear. While less flexible than the other models, it remains a useful baseline model due to its efficiency and ease of interpretation.

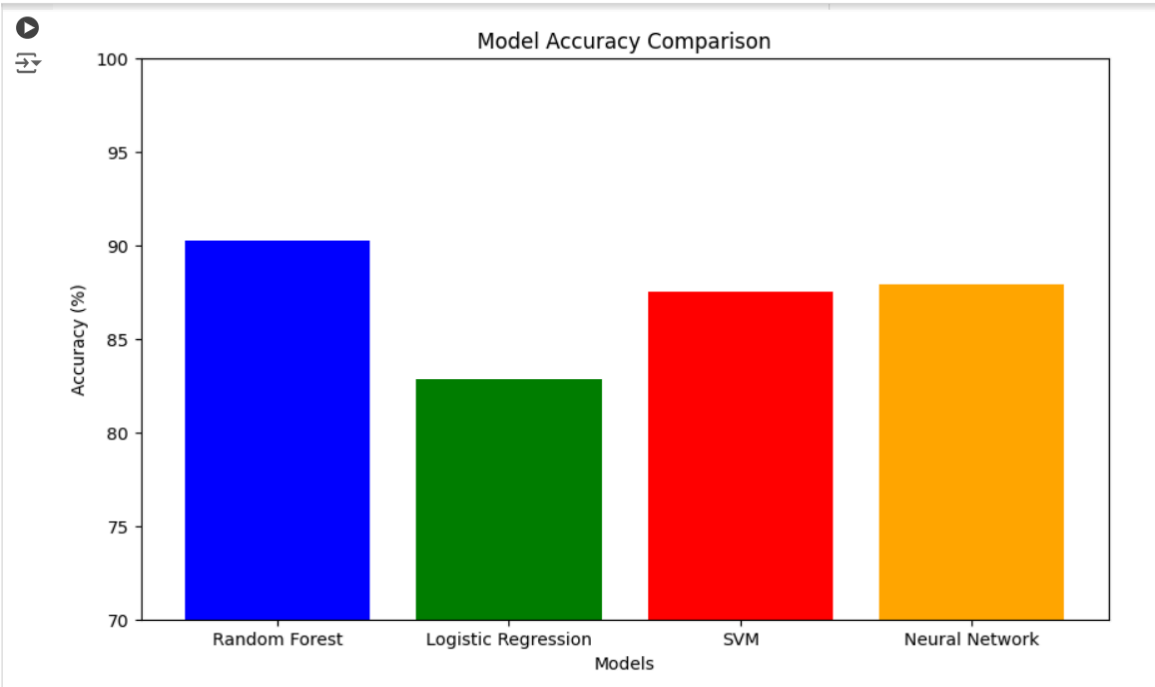
In summary, Random Forest led with the highest accuracy, highlighting its strength in managing diverse data types and complex relationships. MLP and SVM followed, showing promising predictive power, while Logistic Regression, though less accurate, provided a practical baseline. This ranking illustrates how ensemble and non-linear models can enhance ASD prediction, particularly with data complexity.

Figures and Tables:

Confusion matrices



Comparison using Bar Chart



Learning Outcome

(a)Google Colab Link:

This project was developed and executed in Google Colab, which provided a robust environment for experimentation and model development. Here's the link to the Colab notebook where all steps, from data preprocessing to model evaluation, are documented:

https://colab.research.google.com/drive/1sMXuVH17oJAL5rhEoU4m4FJu3OQgFXrc?usp=sharing

(b) GitHub Repository:

The repository contains the code scripts and visualizations.

https://github.com/Janani-Raju/SASTRA_CSE425_MLE

(c) Key Learnings:

This project presented a unique opportunity to gain in-depth experience in autism prediction using machine learning.

Key takeaways include:

- Practical understanding of handling imbalanced data using SMOTE to improve model performance.
- Insights into the strengths and limitations of various machine learning algorithms, allowing informed decisions about model selection.
- Skills in data visualization for analyzing feature distributions and understanding model performance through metrics.
- Beyond technical skills, this project offered a valuable perspective on the implications of machine learning in healthcare.

Conclusion

This project demonstrated the feasibility of predicting Autism Spectrum Disorder (ASD) using machine learning models that leverage behavioral and demographic data. Through training and comparing various models, the Random Forest classifier emerged as the most accurate for this dataset, achieving an accuracy of 90.23%. The project presented a systematic approach to data processing, model selection and evaluation, resulting in reliable predictive capabilities that underscore the potential of machine learning in ASD prediction.

One key advantage of this approach is its comprehensive methodology, which utilizes various machine learning techniques to predict ASD and highlights the effectiveness of ensemble learning (Random Forest) in identifying complex patterns. However, the model's accuracy is heavily dependent on the quality, diversity, and size of the dataset used. If the data lacks representation across various demographics or includes biases, the model may not generalize well to broader populations or underrepresented groups. This limitation suggests the need for further validation on larger, more diverse datasets to enhance the model's applicability across different demographic segments.