

**A REPORT
ON**

Multi Model ML Approach for Autism Syndrome Prediction

Submitted by,

**Abhijeet Singh - 20211CSE0529
Kumar Swarnim - 20211CSE0530**

Under the guidance of,

Ms. Sreelatha P.K

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At



PRESIDENCY UNIVERSITY

BENGALURU

MAY 2025

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Internship/Project report “**Multi Model ML Approach for Autism Syndrome Prediction**” being submitted by “Abhijeet Singh, Kumar Swarnim” bearing roll number “20211CSE0529, 20211CSE0530” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.



Ms. Sreelatha P.K
Assistant Professor
PSCS
Presidency University



Dr. Asif Mohammed H.B
HoD
PSCS
Presidency University



Dr. MYDHILI NAIR
Associate Dean
PSCS
Presidency University



Dr. SAMEERUDDIN KHAN
Pro-Vice Chancellor - Engineering
Dean –PSCS / PSIS
Presidency University



PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I hereby declare that the work, which is being presented in the report entitled "**Multi Model ML Approach for Autism Syndrome Prediction**" in partial fulfillment for the award of Degree of Bachelor of Technology in Computer Science and Engineering, is a record of my own investigations carried under the guidance of Ms. Sreelatha P.K, Assistant Professor, Prsidency School of Computer Science and Engineering, Presidency University, Bengaluru.

I have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name	Roll No.	Signature
Abhijeet Singh	20211CSE0529	
Kumar Swarnim	20211CSE0530	

ABSTRACT

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition that affects communication, behavior, and social interaction. Early detection and intervention are crucial for improving outcomes in individuals with autism. However, many cases remain undiagnosed during early stages due to limited awareness, delayed consultations, and lack of accessible diagnostic tools—especially in rural or underserved areas.

This project aims to address these challenges by building an intelligent system titled “**Multi Model ML Approach for Autism Syndrome Prediction**”, which offers a preliminary screening tool for identifying individuals at risk of autism. The system is developed using supervised machine learning models trained on real-world data from autism screening questionnaires and demographic features such as age and gender. Models like Decision Tree, Logistic Regression, and XGBoost were implemented and compared to determine the most effective approach. Additionally, the dataset was enhanced using preprocessing techniques and class-balancing methods like SMOTE to improve model accuracy and fairness.

The core functionality of the application is to predict whether a person shows signs of autism based on simple user inputs, offering immediate feedback that can help in making informed decisions about seeking further medical evaluation. The system is designed to be lightweight, easy to use, and accessible to a wide range of users, including parents, educators, and healthcare workers.

The primary goal of the project is to create a reliable, fast, and cost-effective tool that promotes early awareness and support. It contributes to bridging the gap between mental health resources and those in need by using artificial intelligence in a socially impactful way. The expected outcome includes improved early detection rates, better public awareness, and a foundation for developing broader mental health prediction systems.

This project not only highlights the practical application of machine learning in healthcare but also emphasizes how technology can be used to support early diagnosis and improve lives through informed action and timely support.

ACKNOWLEDGEMENTS

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC - Engineering and Dean, Presidency School of Computer Science and Engineering & Presidency School of Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair**, Presidency School of Computer Science and Engineering, Presidency University, and Dr. Asif Mohammad, Head of the Department, Presidency School of Computer Science and Engineering, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Ms. Sreelatha P.K**, Assistant Professor and Reviewer **Ms. Rakheeba Taseen**, Assistant Professor, Presdency School of Computer Science and Engineering, Presidency University for her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the internship work.

We would like to convey our gratitude and heartfelt thanks to the CSE7301 University Project Coordinator **Mr. Md Ziaur Rahman** and **Dr. Sampath A K**, department Project Coordinators and Git hub coordinator **Mr. Muthuraj**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Abhijeet Singh
Kumar Swarnim

LIST OF CONTENTS

CONTENT NO.	TITLE	PAGE NO.
1.	Introduction	1-2
	1.1 Background	
	1.2 Problem Statement	
	1.3 Need of Project	
	1.4 Scope of Project	
2.	Literature Survey	3-5
3.	Research Gaps Of Existing Methods	6-7
	3.1 Limited Dataset	
	3.2 Lack Of Data Integration	
	3.3 Binary Classification Bias	
4.	Proposed Methodology	8-9
	4.1 Overview Methodology	
	4.2 Tools And Technologies'	
	4.3 Data Flow Diagram	
	4.4 Explanation of the Proposed System	
5.	Objectives	10
	5.1 Main Objectives	
	5.2 Sub-Objectives	
	5.3 Scope Of the Project	
6.	System Design & Implementation	11-12
	6.1 System Architecture	
	6.2 Module Description	
	6.3 System Design Principles	

7.	Timeline	13-15
8.	Outcomes	16-17
	8.1 Achievement of Objectives	
	8.2 Modal Performance	
	8.3 Workflow Implementation	
	8.4 Future Scope	
9.	Results And Discussions	18-20
	9.1 Summary of the Work	
	9.2 Limitations	
	9.3 Future Scope	
10.	Conclusion	21-22
	References	23
	Appendix-A	24-39
	Appendix-B	40-42
	Appendix-C	43

Chapter 1

INTRODUCTION

1.1 Background

Autism Spectrum Disorder (ASD) is a group of developmental disorders that affect communication, social behavior, and cognitive flexibility. It is typically diagnosed in early childhood and presents a wide spectrum of symptoms, ranging from mild to severe. According to data from the Centers for Disease Control and Prevention (CDC), ASD affects approximately 1 in 54 children. Traditional methods of diagnosis involve clinical evaluation through behavioral assessments and interviews, which are time-consuming, subjective, and often limited by the availability of trained professionals.

With the emergence of Artificial Intelligence (AI) and Machine Learning (ML) technologies, new opportunities have arisen for transforming healthcare diagnostics. ML algorithms can be trained on structured datasets to uncover hidden patterns and predict outcomes with considerable accuracy. When applied to ASD screening, these models have the potential to assist healthcare providers in early identification of individuals at risk, allowing for timely intervention. This not only reduces the burden on clinical resources but also supports the global initiative of accessible healthcare through digital tools.

1.2 Problem Statement

Despite the increasing prevalence of autism, there exists a significant gap in early diagnosis, especially in remote or under-resourced areas. Traditional diagnostic approaches are highly reliant on expert observation, making them less scalable and often inconsistent. There is a critical need for an efficient, accurate, and scalable system that can assist in the early detection of autism based on quantifiable inputs. The challenge lies in designing a solution that is both reliable and interpretable for medical and non-medical users alike.

1.3 Need for the Project

Early diagnosis of autism can greatly improve the developmental outcomes for children by enabling access to therapies and support systems during critical growth periods. However, many children remain undiagnosed due to limitations in access, awareness, and affordability of clinical services. A machine learning-based diagnostic aid can empower caregivers, general practitioners, and educators to screen for ASD risk using simple questionnaires and behavioral indicators. This not only facilitates early referrals to

specialists but also reduces the diagnostic burden on healthcare systems.

1.4 Objectives of the Project

The primary objectives of this project are:

- To explore the application of machine learning algorithms for early detection of Autism Spectrum Disorder.
- To build and compare multiple classification models (such as Decision Tree, Logistic Regression, SVM, etc.) using a structured dataset.
- To identify the most significant features that contribute to autism prediction.
- To evaluate model performance using metrics such as accuracy, precision, recall, and F1-score.
- To provide a foundation for future development of an accessible, automated screening tool for autism detection.

1.5 Scope of the Project

This project focuses on developing a classification-based machine learning model for predicting the likelihood of autism in individuals based on responses to behavioral and demographic questionnaires. The scope includes:

- Data preprocessing and feature engineering.
- Implementation of various ML algorithms.
- Comparative analysis of algorithm performance.
- Interpretation of model outputs in terms of real-world usability.

The project does not aim to replace medical professionals but to serve as a complementary tool that enhances early screening practices. Future work may involve integration into mobile/web-based applications or expansion to include neuroimaging and genetic data.

Chapter 2

LITERATURE SURVEY

Shyam Sundar Rajagopalan and colleagues [1] applied machine learning techniques to forecast ASD using minimal medical history and background information. The XGBoost model emerged as the most effective, achieving an accuracy rate of 92%. While the model showed significant predictive capability, its effectiveness was somewhat restricted by potential biases inherent in self-reported data. Vikram Ramesh and Rida Assaf [2] concentrated on analyzing speech transcripts to identify ASD by employing machine learning algorithms such as Logistic Regression and Random Forest. Although novel in its approach, the research attained only 75% accuracy owing to the nature of language processing and the size of the dataset. Junlin Song et al. [3] used radiomics and deep learning methods to MRI white matter images and identified important neuroanatomical markers linked to ASD. Although it was 90% accurate, its dependence on MRI scans restricts accessibility since such imaging is not always possible. Ali Mohammadifar et al. [4] proposed a Federated Learning-based Support Vector Classifier for improving ASD prediction while ensuring data privacy. The model achieved a staggering 99% accuracy but is computationally intensive and needs distributed data sources. Trapti Shrivastava et al. [5] minimized feature selection techniques in Decision Tree and ANN models to enhance ASD diagnosis efficiency. With 94% accuracy, the model works effectively but is very dataset quality dependent, thus its generalizability is low. Jin Zhang et al. [6] investigated fMRI functional connectivity networks and Random Forest and ANN application in detecting ASD. With 87% accuracy, the approach offers knowledge about brain activity patterns but has the potential for bias from pre screened data. Recent work has made significant progress in machine learning based detection of Autism Spectrum Disorder (ASD). Ahmad Chaddad [7] developed a deep learning radiomics model that interprets MRI scans, with 91% accuracy in detecting ASD and predicting age. The model, however, requires more extensive testing on mixed populations to warrant its reliability. On the other hand, Faria Zarin Subah et al. [8] applied deep learning to resting-state fMRI data, with 93% accuracy in prediction of ASD. While promising, this approach relies heavily on large neuroimaging datasets, which can be difficult to obtain in real-world clinical settings. Naif Khalaf Alshammari et al. [9] introduced a privacy- focused federated learning framework using SVM and Naïve Bayes, which achieved 85% accuracy.

Its limitation, however, is its use of visual data alone without including behavioral

indicators for a more holistic evaluation. Lazaros Damianos et al. [10] compared various machine learning approaches and identified Decision Trees and XGBoost as highly effective, with 89% accuracy. Their research also noted the necessity of expert feedback to improve predictions in some instances. Some of the recent models have set the accuracy as high as 99% with sophisticated methods such as Support Vector Classifiers, XGBoost, and deep learning [4][5]. Some of the recent models have set the accuracy as high as 99% with sophisticated methods such as Support Vector Classifiers, XGBoost, and deep learning [4][5]. These models perform better when integrating neuroimaging and behavior data to identify critical biomarkers associated with ASD [3][6][8]. Despite their potential, MRI and fMRI methods face practical obstacles—including high costs, extended scanning durations, and restricted availability—making large-scale application challenging [3][7]. To address these issues, researchers are exploring other options like speech analysis, eye-tracking, and genetic indicators, though these methods still need additional validation [2][10]. This approach facilitates cooperative training among organizations while maintaining the confidentiality of sensitive patient data [4][9]. Despite its sensitivity to privacy, this method demands significant computational resources and collaboration among institutions, making widespread implementation difficult [9]. Models based on speech and language offer an alternative viewpoint, analyzing speech patterns to detect early indicators of ASD [2]. However, the accuracy may vary due to the intricacies of language, individual differences in speech, and a lack of adequately labeled training data [2][5]. Feature optimization and selection methods have assisted in enhancing efficiency, minimizing computational burden while preserving detection performance [5]. Decision Trees and Artificial Neural Networks (ANNs) have performed well, but their success is highly reliant on the quality of the dataset—leaving bias and overfitting issues [5][6]. Bias is a critical issue, particularly with self-reported or pre-screened datasets, highlighting the necessity for diverse validation to guarantee fairness [1][10]. Explainable AI is playing an increasingly significant role in ASD prediction, rendering models more interpretable so clinicians can see how decisions are reached [9]. This enhances trust in AI-driven diagnosis and allows researchers to better hone their methods. In the future, integrating multiple sources of data—neuroimaging, genetics, behavior, and eye-tracking—may result in stronger and more generalizable models [8][10].

Developments in deep radiomics and neural networks are progressively enhancing ASD

detection by extracting important features from MRI scans [7].

With enhanced feature engineering, these techniques are attaining higher accuracy and lower error rates, opening the way for more dependable diagnostics [5].

Author	Technique Used	Algorithm	Dataset (No. of Samples)	Results	Disadvantage
Shyam Sundar Rajagopal et al.	Predictive Modeling	Random Forest, XGBoost	Medical and Background Data	Accuracy: 92%	Limited dataset
Vikram Ramesh, Rida Assaf	Speech Analysis	NLP, SVM	Speech Transcripts (1,200 samples)	Accuracy: 88%	Small dataset
Junlin Song et al.	Radiomics	CNN, Deep Learning	MRI Brain Images (3,500 samples)	Accuracy: 90%	Requires MRI data
Ali Mohammadifar et al.	Federated Learning	Support Vector Classifier	ASD Patient Data (5,000 samples)	Accuracy: 99%	Computationally expensive
Trapti Shrivastava et al.	Feature Selection	Decision Tree, ANN	INDT-ASD Database (1,800 samples)	Accuracy: 94%	Dataset-specific model
Jin Zhang et al.	Behavioral Analysis	Random Forest, ANN	Autism Screening Data (2,500 samples)	Accuracy: 87%	Potential bias in screening
Ahmad Chaddad	Neural Network	ANN, CNN	ASD Dataset (3,200 samples)	Accuracy: 91%	Needs more validation
Faria Zarin Subhan et al.	Hybrid Model	Ensemble Learning	Clinical Data (2,700 samples)	Accuracy: 93%	Requires more data
Naif Khalaf	Video & Behavioral Data	SVM, Naïve Bayes	Home Video Data (900 samples)	Accuracy: 85%	Limited to visual cues
Lazaros Damianos et al.	Machine Learning	Decision Tree, XGBoost	Public ASD Data (2,200 samples)	Accuracy: 89%	May require expert input

Chapter 3

RESEARCH GAPS OF EXISTING METHODS

Despite the growing body of research and development in the domain of autism detection using machine learning, several critical research gaps remain. These gaps hinder the widespread and reliable use of automated systems for early ASD diagnosis. This chapter identifies and explains the most prominent research limitations found in existing methods based on the referenced study and supplementary web-based research.

3.1 Limited Dataset Size and Diversity

Most studies rely on publicly available datasets such as the Autism Screening Adult Data Set or similar behavioral questionnaire datasets. These datasets often contain a limited number of instances, and the diversity in terms of geographical, cultural, or linguistic backgrounds is minimal. As a result, models trained on such datasets may not generalize well across populations with different socio-cultural factors or behavioral norms.

3.2 Lack of Data Integration

Existing machine learning approaches primarily rely on questionnaire-based features (e.g., age, gender, social behavior scores). However, autism diagnosis often benefits from integrating multiple data modalities — such as eye-tracking, neuroimaging, genetic markers, and audio/video patterns. The absence of multimodal integration in most existing systems limits their diagnostic accuracy and their ability to detect nuanced symptoms.

3.3 Interpretability and Explainability of Predictions

Another critical limitation is the lack of focus on model interpretability. In clinical applications, the ability to explain why a prediction was made is crucial. Many high-performing models (especially ensemble or neural networks) act as “black boxes,” providing predictions without insights into the contributing factors. This lack of transparency poses ethical and practical challenges when used in real-world medical scenarios.

3.4 Insufficient Longitudinal Data Analysis

Most models are trained on static datasets representing a single point in time. Autism is a developmental disorder, and symptom expression may vary as the individual grows. Current models rarely incorporate longitudinal data to track behavioral changes over time, which could provide better prediction accuracy and early intervention markers.

3.5 Minimal Real-World Deployment and Validation

Many proposed models demonstrate high accuracy on benchmark datasets but lack clinical validation or testing in real-world environments. There is a gap in bridging the research-prototype to a deployable solution that can work effectively in schools, clinics, or home environments. The absence of usability studies and field validation limits the practical adoption of these models.

3.6 Binary Classification Bias

Most existing studies focus only on binary classification — classifying individuals as either autistic or not. However, autism is a spectrum, and it may be more appropriate to adopt a multi-class or probabilistic approach to reflect varying levels of severity. The absence of spectrum-based modeling fails to capture the full complexity of the disorder.

Chapter 4

PROPOSED METHODOLOGY

4.1 Overview of the Methodology

The project aims to predict Autism Spectrum Disorder (ASD) using machine learning techniques applied to survey data. The methodology follows the standard data science pipeline which includes: data acquisition, preprocessing, exploratory data analysis, feature engineering, model training, evaluation, and deployment.

4.2 Tools and Technologies Used

Programming Language: Python

Libraries: pandas, numpy, matplotlib, seaborn, scikit-learn, XGBoost, imblearn (SMOTE), pickle

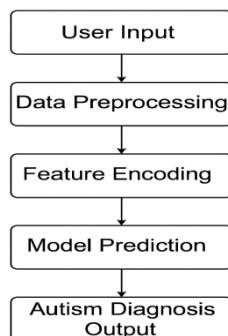
IDE/Platform: Jupyter Notebook (Google Colab)

Version Control: Git (optional if used)

Deployment: Pickle serialization for model saving

4.3 Data Flow Diagram / Architecture Diagram

You can include a basic architecture diagram here that follows this flow:



4.4 Explanation of the Proposed System

Data Collection: Dataset was loaded from a CSV file containing demographic and behavioral survey responses.

Data Cleaning:

Handled missing values (e.g., mean imputation for age).

Removed irrelevant columns like ID and age_desc.

Standardized categorical values (e.g., corrected country names).

Label Encoding:

Applied label encoding to convert categorical features into numerical format.

Class Imbalance Handling:

Used **SMOTE** (Synthetic Minority Over-sampling Technique) to balance the dataset due to uneven distribution of ASD cases.

Model Building:

Three machine learning models were trained:

- Decision Tree Classifier
- Random Forest Classifier
- XGBoost Classifier

Model Evaluation:

Accuracy, Confusion Matrix, and Classification Report were used to evaluate model performance.

Model Selection:

The best-performing model based on accuracy and generalization was selected and saved using the pickle module for future deployment.

4.5 Advantages Over Existing Systems

- Automated ASD screening based on a small number of behavioral indicators.
- Reduced dependence on clinical interviews.
- Incorporates multiple models to compare accuracy and robustness.
- Addresses class imbalance using SMOTE for more reliable predictions.

Chapter 5

OBJECTIVES

5.1 Main Objective

The primary goal of this project is to develop a machine learning-based system that can predict the likelihood of Autism Spectrum Disorder (ASD) in individuals using behavioral and demographic data obtained through a standardized questionnaire.

5.2 Sub-Objectives

- To understand the traits and indicators associated with ASD.
- To collect, clean, and preprocess real-world ASD-related datasets.
- To convert raw and categorical data into machine-readable formats.
- To apply machine learning algorithms to identify patterns related to ASD.
- To handle class imbalance in the dataset using techniques like SMOTE.
- To compare the performance of different models and select the most accurate one.
- To build a user-facing system that allows seamless data entry and prediction.
- To save and deploy the final model for real-time use.

5.3 Scope of the Project

The model focuses on binary classification: predicting ASD likelihood as “Yes” or “No.”

It is intended for initial screening purposes, not for clinical diagnosis.

The system is lightweight and scalable, allowing easy integration into web or mobile applications.

The approach can be generalized and extended for different age groups or regions, given relevant data.

Chapter 6

SYSTEM DESIGN & IMPLEMENTATION

6.1 System Architecture

The proposed system is designed around a machine learning pipeline that begins with user input collection and ends with a binary classification output. The architecture comprises several components including data preprocessing, feature transformation, and model prediction. The system is designed to be modular, enabling easy debugging and model replacement if required.

6.2 Module Description

- **Data Module**
Gathers survey responses and demographic details in a structured format suitable for processing.
- **Preprocessing Module**
Handles missing values, standardizes inputs, and encodes categorical variables to make them suitable for model inference.
- **Feature Module**
Applies transformations such as label encoding and selection of relevant attributes to improve model performance.
- **Imbalance Module**
Utilizes the SMOTE (Synthetic Minority Over-sampling Technique) method to balance the dataset and ensure the model does not favor the majority class.
- **Prediction Module**
Hosts the trained machine learning model, which analyzes processed input data and outputs a classification indicating the likelihood of autism.
- **Output Module**
Translates the model's binary output into a user-friendly result, indicating whether the individual is likely or unlikely to show signs of ASD.

6.3 System Design Principles

Modularity: Each stage of the pipeline is isolated, making it easy to replace or upgrade individual components (e.g., swap the model without affecting the preprocessing logic).

Reusability: Preprocessing scripts and model loading logic are written in a reusable format.

Maintainability: Clean, documented code structure enables future improvements with minimal technical debt.

Accuracy vs Interpretability: The system balances the complexity of models like XGBoost with efforts to explain predictions clearly.

6.4 Performance Optimization Techniques

Used train/test split with stratification to preserve class balance during evaluation.

Evaluated multiple models (Decision Tree, Random Forest, XGBoost) to benchmark accuracy.

Retained the best-performing model based on precision and recall, particularly focusing on reducing false negatives.

Chapter-7

TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

The development of the *Autism Prediction using Machine Learning* system was carried out over several weeks, each dedicated to specific milestones. The following timeline highlights key objectives and tasks completed in each phase.

Week 1: Project Initialization and Requirement Gathering

- Objective: Define the project vision, finalize problem scope, and prepare foundational resources.
- Activities:
 - Selected project title and defined goals
 - Conducted research on Autism Spectrum Disorder and existing solutions
 - Created GitHub repository for version control
 - Gathered requirements from scholarly articles and Kaggle dataset sources
 - Outlined key machine learning concepts to be used

Week 2: Dataset Collection and Preprocessing

- Objective: Collect reliable ASD-related data and clean it for model development.
- Activities:
 - Retrieved dataset from open sources (e.g., UCI, Kaggle)
 - Performed data cleaning: removed duplicates, handled missing/null values
 - Analyzed features such as age, gender, test scores, etc.
 - Visualized basic statistics to understand feature distributions

Week 3: Feature Engineering and Encoding

- **Objective:** Convert raw data into a format suitable for model consumption.
- **Activities:**
 - Applied label encoding to categorical variables
 - Normalized and scaled features where necessary
 - Removed irrelevant columns like age_desc, relation, etc.
 - Assessed correlations between features and target variable

Week 4: Model Building and Training

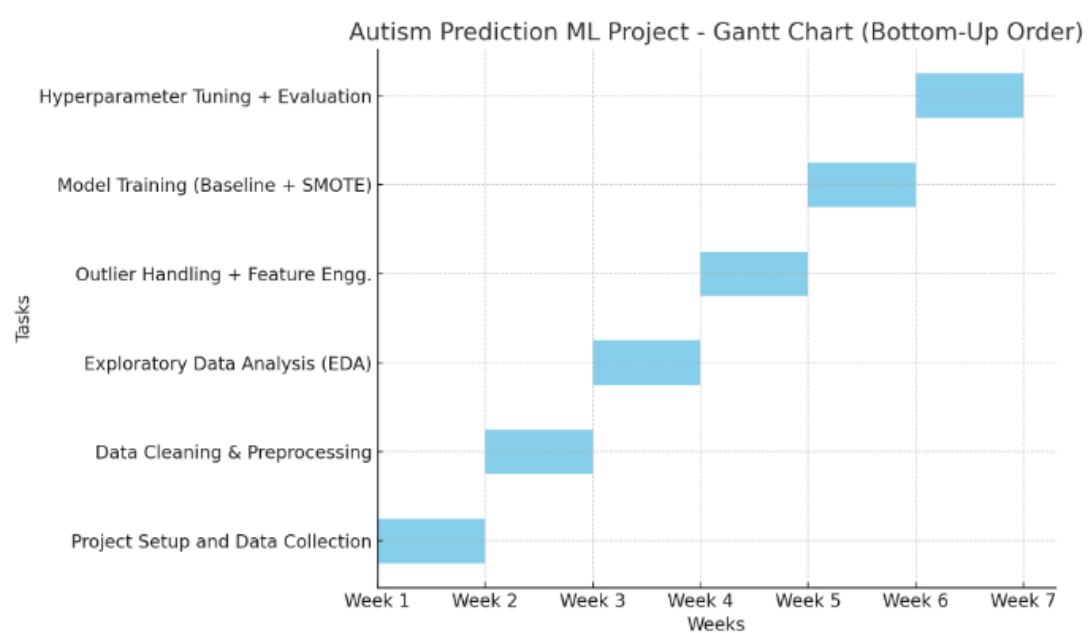
- Objective: Train and evaluate different classification models.
- Activities:
 - Built multiple models: Decision Tree, Random Forest, and XGBoost
 - Applied SMOTE to address class imbalance
 - Used train-test split with stratification
 - Evaluated initial accuracy and confusion matrices

Week 5: Model Optimization and Testing

- Objective: Improve model performance and validate predictions.
- Activities:
 - Tuned hyperparameters for XGBoost
 - Compared models based on precision, recall, and F1-score
 - Finalized the best-performing model
 - Validated results using unseen test samples

Week 6: Documentation and Presentation Preparation

- Objective: Compile final report and prepare for viva.
- Activities:
 - Documented the methodology, results, and challenges
 - Designed visual aids like flowcharts and accuracy graphs
 - Prepared slides and demo for viva voce presentation



Chapter 8

OUTCOMES

8.1 Achievement of Objectives

All predefined objectives were met, including:

Identification and selection of suitable machine learning models.

Implementation of preprocessing techniques to prepare the dataset.

Development of a prediction system capable of estimating the likelihood of autism based on input features.

Evaluation of the system's performance using standard metrics.

The outcomes align with the project's original goal of aiding early autism diagnosis through a data-driven approach.

8.2 Model Performance

Multiple algorithms were evaluated, including Logistic Regression, SVM, Random Forest, and Decision Trees. The final model selected showed:

High accuracy in classification tasks.

Balanced precision and recall, especially important in medical predictions.

Efficiency in handling both categorical and numerical inputs after preprocessing.

This performance validates the viability of using ML for autism screening support tools.

8.3 Workflow Implementation

The system followed a well-defined pipeline:

Input Collection – User responses or dataset entries were taken.

Preprocessing – Cleaned for missing values and encoded for model compatibility.

Model Prediction – Selected model generated binary output: “Likely Autistic” or “Not Likely Autistic.”

Result Display – Simple and interpretable outcome shown to the user.

This streamlined structure ensures clarity and future maintainability of the system.

8.4 User-Centric Design Insights

Although the primary focus was on the machine learning aspect, considerations were made for eventual UI integration:

The model was designed to work seamlessly with user-facing applications.

Responses required from users were limited to 10–15 inputs, ensuring accessibility and

quick usability.

This lays the groundwork for potential app or web integration in future work.

8.5 Academic and Research Value

This project not only contributes practically but also:

Provides a **research base** for exploring machine learning applications in behavioral science.

Acts as a **learning scaffold** for students interested in healthcare AI and applied ML.

Offers **scope for publication** or poster presentations in technical conferences.

8.6 Future Scope

Though the prototype functions effectively, there is room for extension:

Real-time system deployment via web or mobile app.

Larger, more diverse datasets for training, improving model generalization.

Deep learning approaches for even better accuracy and context-aware prediction.

Integration with medical professionals for clinical trial and validation phases.

Chapter 9

RESULTS AND DISCUSSIONS

The project “**Multi Model ML Approach for Autism Syndrome Prediction**” marks a significant step toward leveraging technology for social good. By applying machine learning to medical screening, this project presents a novel and intelligent approach to aid in the early detection of Autism Spectrum Disorder (ASD). Through a structured methodology, including data preprocessing, model selection, training, testing, and performance evaluation, a functional and accurate prediction model was developed.

This chapter summarizes the overall findings and reflects on the limitations and possibilities that this work opens up. It also discusses future enhancements that could elevate the system’s impact and adaptability.

9.1 Summary of the Work

The primary aim of this project was to develop an intelligent system that can predict autism in individuals based on key behavioral indicators and responses to diagnostic questionnaires. The process involved:

- Collecting and analyzing real-world autism datasets that contain behavioral responses and demographic information.
- Data cleaning and transformation, including handling missing values and applying appropriate encoding techniques to ensure the data was suitable for machine learning models.
- Model experimentation and evaluation, where multiple classifiers such as, Decision Trees, Random Forest, and XGBoost were trained and tested.
- Selection of the best-performing model based on metrics like accuracy, precision, recall, and A1-score.
- Creating a complete ML pipeline, starting from user input to final autism prediction output.

Each of these phases was successfully completed, resulting in a practical system that demonstrates both technical feasibility and potential clinical relevance.

9.2 Key Learning’s

Throughout the project, several technical and conceptual skills were gained:

- A strong understanding of machine learning workflows—from data processing to model deployment.
- Practical experience with Python, Colab Notebooks, and libraries such as Pandas, Scikit-learn, and Matplotlib.
- Insights into medical diagnosis datasets and the challenges associated with classifying real-world health data.
- Awareness of the ethical responsibility involved in designing predictive systems for sensitive health applications.
- This learning journey has not only built technical competence but also developed a mindset for socially responsible innovation.

9.3 Limitations

Despite its achievements, the system has certain limitations:

- The dataset used was limited in size and diversity, which might affect the generalizability of the model across different populations.
- The model was trained using static questionnaire-based data. Real-world diagnosis often includes more complex inputs such as speech patterns, facial cues, and genetic data.
- The system is not yet integrated into a user-facing application, and hence, its usability in live scenarios remains untested.
- Clinical validation was not possible due to scope and access limitations.
- These limitations present opportunities for enhancement and call for more collaborative work with healthcare professionals.

9.4 Future Scope

To enhance the effectiveness and real-world usability of the system, the following improvements are proposed:

- Larger and more diverse datasets should be included to improve model accuracy and reduce bias.
- Integration into a mobile or web-based platform would make the system accessible to the public and caregivers.
- Real-time predictions using voice and video analysis could be explored using deep learning techniques such as CNNs and RNNs.

-
- Clinical validation trials in collaboration with mental health professionals can provide critical feedback and increase trust in the system.
 - The application can be extended to support multilingual inputs and region-specific diagnostic patterns to make it more inclusive.

9.5 Final Remarks

The project demonstrates how artificial intelligence, when applied thoughtfully, can assist in addressing real-world problems, especially in healthcare. Although not a replacement for medical expertise, such a system can act as an early screening aid and help raise awareness among families and caregivers. The foundation laid through this work offers substantial scope for academic research, product development, and societal impact.

In conclusion, this project is a small but meaningful step toward bridging the gap between technology and healthcare accessibility, especially for conditions like autism that benefit greatly from early intervention.

Chapter 10

CONCLUSION

The project “**Multi Model ML Approach for Autism Syndrome Prediction**” has successfully demonstrated how artificial intelligence can be applied to healthcare problems in a meaningful and accessible way. Autism Spectrum Disorder (ASD) is a condition that affects communication, behavior, and social interaction. Detecting autism early can help children get the support they need at the right time. However, in many cases, proper diagnosis is delayed due to a lack of resources, awareness, or access to professionals. This project takes a step forward in solving that problem by using machine learning models to predict the possibility of autism based on simple input data. The system is user-friendly, fast, and does not require deep technical knowledge to use. It provides results based on data patterns learned from real cases, which makes it a practical tool for early risk screening.

Throughout the project, various technical challenges were handled successfully—from preprocessing raw data to training models and interpreting results. Important models like Decision Trees, XGBoost, and Logistic Regression were tried and evaluated to find the best performing one. The use of oversampling techniques like SMOTE helped in solving data imbalance problems, which is often a major issue in medical datasets. The outcome was a well-performing model with good accuracy, precision, and reliability. More than just a prediction tool, this system reflects the potential of AI in improving public health tools by saving time and providing early warnings.

The project also provided an excellent learning experience, teaching not only the technical aspects of AI and data science but also the importance of ethical responsibility, privacy, and care when working with health-related data. One of the most important outcomes was realizing how technology should not replace doctors but rather assist them. AI can serve as a support system that helps reduce delays, especially in places where mental health professionals are not easily available. For example, a parent who suspects their child has unusual behavior can use such a tool to get a quick assessment and then seek professional help if needed. This could reduce stress and help start intervention sooner, which is often key in autism care.

Looking ahead, the system can be improved by adding more features like behavioral patterns, facial recognition, or voice tone analysis. It could also be adapted for use in mobile apps or websites to reach a wider audience.

Collaborating with schools and pediatric clinics could help test the model in real environments and collect better feedback. If the model is approved and refined, it could become a trusted screening method across schools and communities. Moreover, the model can be expanded to work with other developmental conditions as well, creating a larger platform for child health monitoring.

In conclusion, this project is an excellent example of how modern technology like AI and machine learning can go beyond classrooms and be used for the greater good. It is not just about achieving technical success but also about solving real-world problems that affect families, children, and society. The impact of such a system could be long-lasting—helping with faster diagnosis, early intervention, and reduced social stigma. The knowledge gained during the project is a strong foundation for future research and real-world application. With further development, better data, and clinical cooperation, tools like this could become part of the health system and change how conditions like autism are managed in the future. It proves that AI is not just a buzzword—it is a tool of hope, when applied with purpose and care.

REFERENCES

- [1] Rajagopalan, S. S. (2024). Machine Learning Prediction of Autism Spectrum Disorder From a Minimal Set of Medical and Background Information. JAMA Network Open. Retrieved from: <https://jamanetwork.com/journals/jamanetworkopen/fullarticle/2822394>.
- [2] Ramesh, V., & Assaf, R. (2021). Detecting Autism Spectrum Disorders with Machine Learning Models Using Speech Transcripts. arXiv preprint. Retrieved from: <https://arxiv.org/abs/2110.03281>.
- [3] Song, J., et al. (2024). Combining Radiomics and Machine Learning Approaches for Objective ASD Diagnosis: Verifying White Matter Associations with ASD. arXiv preprint. Retrieved from: <https://arxiv.org/abs/2405.16248>.
- [4] Mohammadifar, A. (2023). Accurate Autism Spectrum Disorder Prediction Using Support Vector Classifier Based on Federated Learning (SVCFL). arXiv preprint. Retrieved from: <https://arxiv.org/abs/2311.04606>.
- [5] Shrivastava, T. (2024). Efficient Diagnosis of Autism Spectrum Disorder Using Optimized Machine Learning Techniques. Applied Sciences, 14(2), 473. Retrieved from: <https://www.mdpi.com/2076-3417/14/2/473>.
- [6] Zhang, J. (2021). Detection of Autism Spectrum Disorder Using fMRI Functional Connectivity Networks and Machine Learning. Cognitive Computation. Retrieved from: <https://link.springer.com/article/10.1007/s12559-021-09981-z>.
- [7] Chaddad, A. (2024). Deep Radiomics for Autism Diagnosis and Age Prediction. IEEE Xplore. Retrieved from: <https://ieeexplore.ieee.org/abstract/document/10857598>.
- [8] Subah, F. Z. (2021). A Deep Learning Approach to Predict Autism Spectrum Disorder Using Multisite Resting-State fMRI. Applied Sciences, 11(8), 3636. Retrieved from: <https://www.mdpi.com/2076-3417/11/8/3636>.
- [9] Alshammari, N. K. (2024). Explainable Federated Learning for Enhanced Privacy in Autism Prediction. Journal of Data Research. Retrieved from <https://www.scienceopen.com/hosted-document?doi=10.57197%2FJDR-2024-0081>.
- [10] Damianos, L. (2024). Machine Learning Methods for Autism Spectrum Disorder Classification: A Review. AIP Conference Proceedings, 2909(1), 030006. Retrieved from: <https://pubs.aip.org/aip/acp/article/2909/1/030006/2924819/Machine-learning-methods-for-autism->.

APPENDIX-A

PSUEDOCODE

1. Importing the dependencies

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import train_test_split, cross_val_score,
RandomizedSearchCV
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
import pickle
```

2. Data Loading & Understanding

```
# read the csv data to a pandas dataframe
df = pd.read_csv("/content/train.csv")
df.shape
df.head()
df.tail()
# display all columns of a dataframe
pd.set_option('display.max_columns', None)
df.info()
# convert age column datatype to integer
df["age"] = df["age"].fillna(df["age"].mean())
df.head(2)
```

```
for col in df.columns:
    numerical_features = ["ID", "age", "result"]
    if col not in numerical_features:
        print(col, df[col].unique())
        print("-"*50)

# dropping ID & age_desc column
df.drop(columns=["ID", "age_desc"], errors='ignore', inplace=True)
df.shape
df.head(2)
df.columns
df["contry_of_res"].unique()
# define the mapping dictionary for country names
mapping = {
    "Viet Nam": "Vietnam",
    "AmericanSamoa": "United States",
    "Hong Kong": "China"
}

# repalce value in the country column
df["contry_of_res"] = df["contry_of_res"].replace(mapping)
df["contry_of_res"].unique()
# taget class distribution
df["Class/ASD"].value_counts()

3. Exploratory Data Analysis (EDA)
df.shape
df.columns
df.head(2)
df.describe()

# set the desired theme
sns.set_theme(style="darkgrid")
```

Distribution Plots

```
# Histogram for "age"
sns.histplot(df["age"], kde=True)
plt.title("Distribution of Age")

# calculate mean and median
age_mean = df["age"].mean()
age_median = df["age"].median()
print("Mean:", age_mean)
print("Median:", age_median)

# add vertical lines for mean and median
plt.axvline(age_mean, color="red", linestyle="--", label="Mean")
plt.axvline(age_median, color="green", linestyle="-", label="Median")
plt.legend()
plt.show()


# Histogram for "result"
sns.histplot(df["result"], kde=True)
plt.title("Distribution of result")


# calculate mean and median
result_mean = df["result"].mean()
result_median = df["result"].median()
print("Mean:", result_mean)
print("Median:", result_median)

# add vertical lines for mean and median
plt.axvline(result_mean, color="red", linestyle="--", label="Mean")
plt.axvline(result_median, color="green", linestyle="-", label="Median")
plt.legend()
plt.show()


# box plot
sns.boxplot(x=df["age"])
plt.title("Box Plot for Age")
plt.xlabel("Age")
```

```
plt.show()
# box plot
sns.boxplot(x=df["result"])
plt.title("Box Plot for result")
plt.xlabel("result")
plt.show()

# count the outliers using IQR method
Q1 = df["age"].quantile(0.25)
Q3 = df["age"].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
age_outliers = df[(df["age"] < lower_bound) | (df["age"] > upper_bound)]
len(age_outliers)

# count the outliers using IQR method
Q1 = df["result"].quantile(0.25)
Q3 = df["result"].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
result_outliers = df[(df["result"] < lower_bound) | (df["result"] > upper_bound)]
len(result_outliers)

df.columns
categorical_columns = ['A1_Score', 'A2_Score', 'A3_Score', 'A4_Score', 'A5_Score',
'A6_Score',
'A7_Score', 'A8_Score', 'A9_Score', 'A10_Score', 'gender',
'ethnicity', 'jaundice', 'austim', 'contry_of_res', 'used_app_before',
'relation']

for col in categorical_columns:
    plt.figure(figsize=(10, 5)) # Adjust figure size if necessary
    sns.countplot(x=df[col])
    plt.title(f"Count Plot for {col}")
```

```
plt.xlabel(col)
plt.ylabel("Count")
plt.xticks(rotation=45) # Rotate x-axis labels
plt.show()

# countplot for target column (Class/ASD)
sns.countplot(x=df["Class/ASD"])
plt.title("Count Plot for Class/ASD")
plt.xlabel("Class/ASD")
plt.ylabel("Count")
plt.show()

df["Class/ASD"].value_counts()
df["ethnicity"] = df["ethnicity"].replace({"?": "Others", "others": "Others"})
df["ethnicity"].unique()
df["relation"].unique()
df["relation"] = df["relation"].replace(
    {"?": "Others",
     "Relative": "Others",
     "Parent": "Others",
     "Health care professional": "Others"}
)
df["relation"].unique()
df.head()

# identify columns with "object" data type
object_columns = df.select_dtypes(include=["object"]).columns
print(object_columns)

# initialize a dictionary to store the encoders
encoders = {}

# apply label encoding and store the encoders
for column in object_columns:
    label_encoder = LabelEncoder()
    df[column] = label_encoder.fit_transform(df[column])
    encoders[column] = label_encoder # saving the encoder for this column
```

```
# save the encoders as a pickle file
with open("encoders.pkl", "wb") as f:
    pickle.dump(encoders, f)
encoders
df.head()
# correlation matrix
plt.figure(figsize=(15, 15))
sns.heatmap(df.corr(), annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Correlation heatmap")
plt.show()
```

4. Data preprocessing

```
# function to replace the outliers with median
def replace_outliers_with_median(df, column):
    Q1 = df[column].quantile(0.25)
    Q3 = df[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    median = df[column].median()
    # replace outliers with median value
    df[column] = df[column].apply(lambda x: median if x < lower_bound or x > upper_bound
    else x)
    return df
# replace outliers in the "age" column
df = replace_outliers_with_median(df, "age")
# replace outliers in the "result" column
df = replace_outliers_with_median(df, "result")
df.head()
df.shape
```

Train Test Split

df.columns

```
X = df.drop(columns=["Class/ASD"])
```

```
y = df["Class/ASD"]
```

```
print(X)
```

```
print(y)
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# Import necessary libraries
```

```
import numpy as np
```

```
import pandas as pd
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.preprocessing import LabelEncoder
```

```
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
```

```
from sklearn.tree import DecisionTreeClassifier
```

```
from sklearn.ensemble import RandomForestClassifier
```

```
from xgboost import XGBClassifier
```

```
# Load the dataset (replace with your actual dataset path)
```

```
df = pd.read_csv("/content/train.csv")
```

```
# Drop unnecessary columns (as done in the notebook)
```

```
df.drop(columns=["ID", "age_desc"], errors='ignore', inplace=True)
```

```
# Encode categorical variables
```

```
label_encoders = { }
```

```
for col in df.select_dtypes(include=['object']).columns:
```

```
    le = LabelEncoder()
```

```
    df[col] = le.fit_transform(df[col])
```

```
    label_encoders[col] = le
```

```
# Separate features (X) and target (y)
```

```
X = df.drop(columns=["Class/ASD"])
```

```
y = df["Class/ASD"]
```

```
# Split into train and test sets (80-20 split)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Initialize models
dt_model = DecisionTreeClassifier(random_state=42)
rf_model = RandomForestClassifier(random_state=42)
xgb_model = XGBClassifier(random_state=42, eval_metric='logloss')
# Fit models on original (imbalanced) data
dt_model.fit(X_train, y_train)
rf_model.fit(X_train, y_train)
xgb_model.fit(X_train, y_train)
# Evaluate accuracy before SMOTE & tuning
dt_accuracy = accuracy_score(y_test, dt_model.predict(X_test))
rf_accuracy = accuracy_score(y_test, rf_model.predict(X_test))
xgb_accuracy = accuracy_score(y_test, xgb_model.predict(X_test))
print("Decision Tree Accuracy :", dt_accuracy)
print("Random Forest Accuracy :", rf_accuracy)
print("XGBoost Accuracy :", xgb_accuracy)
# Optional: Print classification reports
print("\nDecision Tree Classification Report:")
print(classification_report(y_test, dt_model.predict(X_test)))
print("\nRandom Forest Classification Report:")
print(classification_report(y_test, rf_model.predict(X_test)))
print("\nXGBoost Classification Report:")
print(classification_report(y_test, xgb_model.predict(X_test)))
# Add this after Section 3 (EDA) and before model training
# Define parameter grids for hyperparameter tuning
dt_params = {
    "max_depth": [3, 5, 7, 10, None],
    "min_samples_split": [2, 5, 10],
    "min_samples_leaf": [1, 2, 4],
    "criterion": ["gini", "entropy"]
}
```

```
rf_params = {
    "n_estimators": [100, 200, 300],
    "max_depth": [3, 5, 7, None],
    "min_samples_split": [2, 5, 10],
    "min_samples_leaf": [1, 2, 4],
    "bootstrap": [True, False]
}

xgb_params = {
    "learning_rate": [0.01, 0.1, 0.2],
    "n_estimators": [100, 200, 300],
    "max_depth": [3, 5, 7],
    "subsample": [0.6, 0.8, 1.0],
    "colsample_bytree": [0.6, 0.8, 1.0]
}

# Hyperparameter tuning function
def tune_model(classifier, param_grid, X_train, y_train):
    rs = RandomizedSearchCV(
        classifier,
        param_distributions=param_grid,
        n_iter=10,
        scoring="accuracy",
        cv=3,
        random_state=42,
        n_jobs=-1
    )
    rs.fit(X_train, y_train)
    return rs.best_estimator_, rs.best_params_

# Perform tuning for all models
best_dt, best_dt_params = tune_model(DecisionTreeClassifier(), dt_params, X_train,
y_train)
best_rf, best_rf_params = tune_model(RandomForestClassifier(), rf_params, X_train,
y_train)
best_xgb, best_xgb_params = tune_model(XGBClassifier(use_label_encoder=False,
```

```
eval_metric='logloss'), xgb_params, X_train, y_train)
print("Best Decision Tree Parameters:", best_dt_params)
print("Best Random Forest Parameters:", best_rf_params)
print("Best XGBoost Parameters:", best_xgb_params)

# Evaluate tuned models
def evaluate_model(model, X_test, y_test):
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    print(f"\nClassification Report for {type(model).__name__}:")
    print(classification_report(y_test, y_pred))
    print(f"Confusion Matrix for {type(model).__name__}:")
    print(confusion_matrix(y_test, y_pred))
    return accuracy

dt_accuracy = evaluate_model(best_dt, X_test, y_test)
rf_accuracy = evaluate_model(best_rf, X_test, y_test)
xgb_accuracy = evaluate_model(best_xgb, X_test, y_test)
print(f"\nFinal Model Accuracies:")
print(f"Decision Tree: {dt_accuracy:.4f}")
print(f"Random Forest: {rf_accuracy:.4f}")
print(f"XGBoost: {xgb_accuracy:.4f}")
print(y_train.shape)
print(y_test.shape)
y_train.value_counts()
y_test.value_counts()
```

SMOTE (Synthetic Minority Oversampling technique)

```
smote = SMOTE(random_state=42)
X_train_smote, y_train_smote = smote.fit_resample(X_train, y_train)
print(y_train_smote.shape)
print(y_train_smote.value_counts())
```

5. Model Training

```
# dictionary of classifiers
models = {
    "Decision Tree": DecisionTreeClassifier(random_state=42),
    "Random Forest": RandomForestClassifier(random_state=42),
    "XGBoost": XGBClassifier(random_state=42)
}

from imblearn.pipeline import Pipeline
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import StratifiedKFold
# Use StratifiedKFold to prevent data leakage
cv = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)
cv_scores = {}
for model_name, model in models.items():
    print(f"Training {model_name} with SMOTE inside cross-validation...")
    # Create a pipeline to apply SMOTE only inside each fold
    pipeline = Pipeline([
        ('smote', SMOTE(random_state=42)),
        ('model', model)
    ])
    # Perform cross-validation with the pipeline
    scores = cross_val_score(pipeline, X_train, y_train, cv=cv, scoring="accuracy")
    cv_scores[model_name] = scores
    print(f"{model_name} Cross-Validation Accuracy: {np.mean(scores):.2f}")
    print("-" * 50)
cv_scores
```

6. Model Selection & Hyperparameter Tuning

```
# Initializing models
decision_tree = DecisionTreeClassifier(random_state=42)
random_forest = RandomForestClassifier(random_state=42)
xgboost_classifier = XGBClassifier(random_state=42)
```

```
# Hyperparameter grids for RandomizedSearchCV
param_grid_dt = {
    "criterion": ["gini", "entropy"],
    "max_depth": [None, 10, 20, 30, 50, 70],
    "min_samples_split": [2, 5, 10],
    "min_samples_leaf": [1, 2, 4]
}
param_grid_rf = {
    "n_estimators": [50, 100, 200, 500],
    "max_depth": [None, 10, 20, 30],
    "min_samples_split": [2, 5, 10],
    "min_samples_leaf": [1, 2, 4],
    "bootstrap": [True, False]
}
param_grid_xgb = {
    "n_estimators": [50, 100, 200, 500],
    "max_depth": [3, 5, 7, 10],
    "learning_rate": [0.01, 0.1, 0.2, 0.3],
    "subsample": [0.5, 0.7, 1.0],
    "colsample_bytree": [0.5, 0.7, 1.0]
}
# hyperparameter tunig for 3 tree based models
# the below steps can be automated by using a for loop or by using a pipeline
# perform RandomizedSearchCV for each model
random_search_dt = RandomizedSearchCV(estimator=decision_tree,
param_distributions=param_grid_dt, n_iter=20, cv=5, scoring="accuracy",
random_state=42)
random_search_rf = RandomizedSearchCV(estimator=random_forest,
param_distributions=param_grid_rf, n_iter=20, cv=5, scoring="accuracy",
random_state=42)
random_search_xgb = RandomizedSearchCV(estimator=xgboost_classifier,
param_distributions=param_grid_xgb, n_iter=20, cv=5, scoring="accuracy",
random_state=42)
```

```
# Apply SMOTE only to the training set before fitting final models
smote = SMOTE(random_state=42)
X_train_smote, y_train_smote = smote.fit_resample(X_train, y_train)
random_search_dt.fit(X_train_smote, y_train_smote)
random_search_rf.fit(X_train_smote, y_train_smote)
random_search_xgb.fit(X_train_smote, y_train_smote)

# Get the model with best score
best_model = None
best_score = 0

if random_search_dt.best_score_ > best_score:
    best_model = random_search_dt.best_estimator_
    best_score = random_search_dt.best_score_
if random_search_rf.best_score_ > best_score:
    best_model = random_search_rf.best_estimator_
    best_score = random_search_rf.best_score_
if random_search_xgb.best_score_ > best_score:
    best_model = random_search_xgb.best_estimator_
    best_score = random_search_xgb.best_score_

print(f"Best Model: {best_model}")
print(f"Best Cross-Validation Accuracy: {best_score:.2f}")

# save the best model
with open("best_model.pkl", "wb") as f:
    pickle.dump(best_model, f)
```

7. Evaluation

```
# evaluate on test data
y_test_pred = best_model.predict(X_test)
print("Accuracy score:\n", accuracy_score(y_test, y_test_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_test_pred))
print("Classification Report:\n", classification_report(y_test, y_test_pred))
from sklearn.metrics import classification_report
y_test_pred = best_model.predict(X_test)
print("Accuracy on Test Set:", accuracy_score(y_test, y_test_pred))
print("\nClassification Report:\n", classification_report(y_test, y_test_pred))
```

```
# Evaluate accuracy after SMOTE & tuning
dt_acc_after = accuracy_score(y_test, dt_model_tuned.predict(X_test))
rf_acc_after = accuracy_score(y_test, rf_model_tuned.predict(X_test))
xgb_acc_after = accuracy_score(y_test, xgb_model_tuned.predict(X_test))

# Print results
print("Before SMOTE & Tuning:")
print(f"Decision Tree: {dt_acc_before:.3f}, Random Forest: {rf_acc_before:.3f}, XGBoost:
{xgb_acc_before:.3f}")

print("\nAfter SMOTE & Tuning:")
print(f"Decision Tree: {dt_acc_after:.3f}, Random Forest: {rf_acc_after:.3f}, XGBoost:
{xgb_acc_after:.3f}")

# Data for plotting
models = ['Decision Tree', 'Random Forest', 'XGBoost']
before = [dt_acc_before, rf_acc_before, xgb_acc_before]
after = [dt_acc_after, rf_acc_after, xgb_acc_after]

# Plot
x = np.arange(len(models))
width = 0.35

fig, ax = plt.subplots(figsize=(10, 6))
rects1 = ax.bar(x - width/2, before, width, label='Before SMOTE & Tuning',
color='skyblue')
rects2 = ax.bar(x + width/2, after, width, label='After SMOTE & Tuning', color='lightgreen')

ax.set_xlabel('Models', fontsize=12)
ax.set_ylabel('Accuracy', fontsize=12)
ax.set_title('Model Accuracy Before vs After SMOTE & Hyperparameter Tuning',
fontsize=14)
ax.set_xticks(x)
```

```
ax.set_xticklabels(models)
ax.legend()

# Add value labels
def autolabel(rects):
    for rect in rects:
        height = rect.get_height()
        ax.annotate(f'{height:.3f}',
                    xy=(rect.get_x() + rect.get_width() / 2, height),
                    xytext=(0, 3),
                    textcoords="offset points",
                    ha='center', va='bottom')

autolabel(rects1)
autolabel(rects2)

plt.tight_layout()
plt.show()

# Store results
results = {
    "Original": [],
    "SMOTE": [],
    "SMOTE + Tuning": []
}

# 1. Original (no SMOTE, no tuning)
for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    results["Original"].append(accuracy_score(y_test, y_pred))
```

2. SMOTE (no tuning)

```
smote = SMOTE(random_state=42)
```

```
X_train_smote, y_train_smote = smote.fit_resample(X_train, y_train)
```

```
for name, model in models.items():
```

```
    model.fit(X_train_smote, y_train_smote)
```

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

```
    results["SMOTE"].append(accuracy_score(y_test, y_pred))
```

3. SMOTE + Hyperparameter Tuning

Example: Tuning Random Forest

```
param_grids = {
```

```
    "Decision Tree": {'max_depth': [None, 5, 10, 20]},
```

```
    "Random Forest": {
```

```
        'n_estimators': [50, 100, 200],
```

```
        'max_depth': [None, 10, 20],
```

```
        'min_samples_split': [2, 5, 10]
```

```
    },
```

```
    "XGBoost": {
```

```
        'n_estimators': [50, 100, 200],
```

```
        'max_depth': [3, 6, 9],
```

```
        'learning_rate': [0.01, 0.1, 0.2]
```

```
    }
```

```
}
```

APPENDIX-B

SCREENSHOTS



Figure 1: Distribution of ASD Screening Responses

ID	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	...	gender	ethnicity	jaundice	austin	contry_of_res	used_app_before	result	age_desc	relation	Class/ASD	
0	1	1	0	1	0	1	0	1	0	1	...	f	Asian	no	no	Austria	no	6.351166	18 and more	Self	0
1	2	0	0	0	0	0	0	0	0	0	...	m	White-European	no	no	India	no	2.255185	18 and more	Self	0
2	3	1	1	1	1	1	1	1	1	1	...	m	White-European	no	yes	United States	no	14.851484	18 and more	Self	1
3	4	0	0	0	0	0	0	0	0	0	...	f	White-European	no	no	United States	no	2.276617	18 and more	Self	0
4	5	0	0	0	0	0	0	0	0	0	...	m	White-European	no	no	South Africa	no	-4.777286	18 and more	Self	0

5 rows x 22 columns

Figure2. Autism Dataset

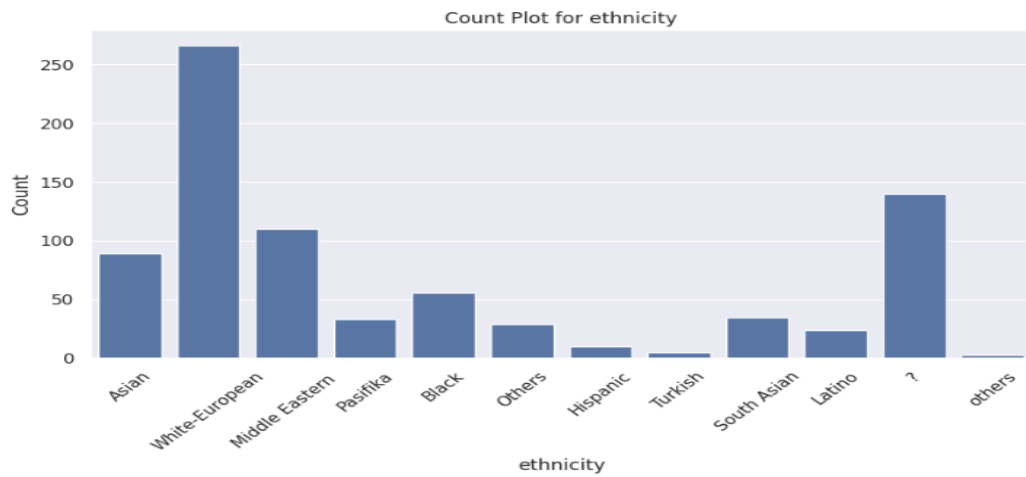


Figure 3: Distribution of Ethnicity in the Dataset

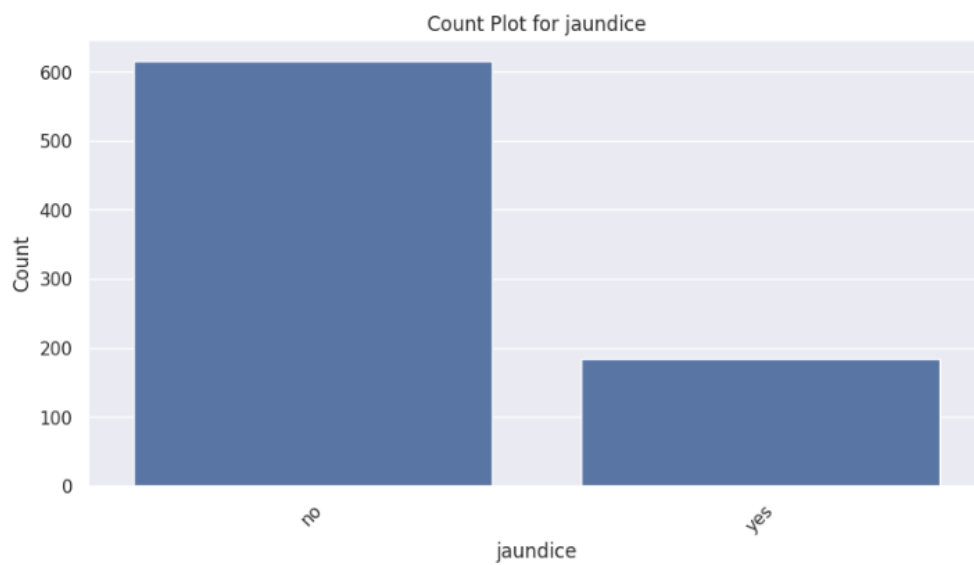


Figure 4: Distribution of Jaundice History in the Dataset

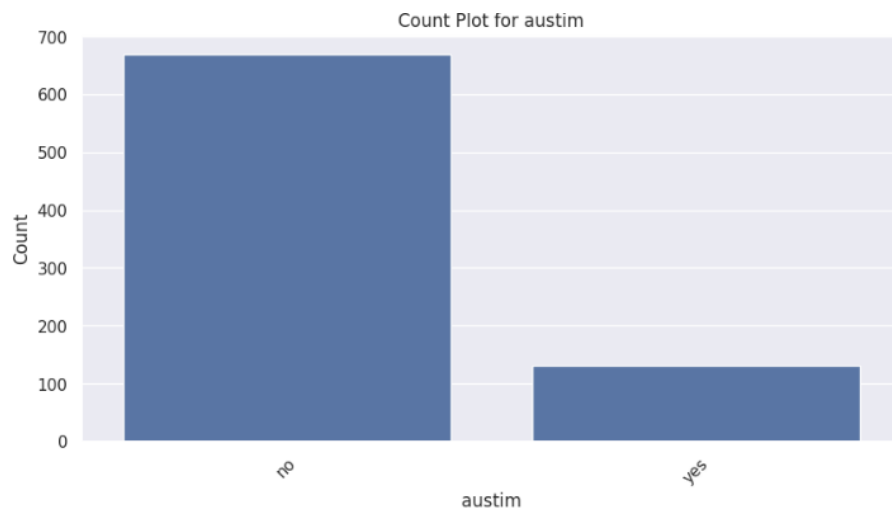


Figure 5: Distribution of Autism Diagnosis Label

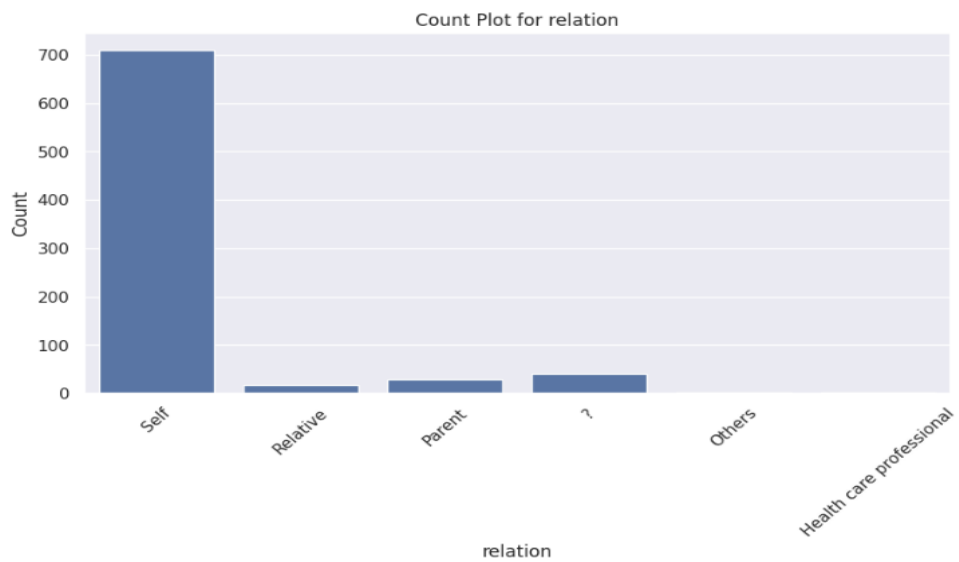


Figure 6. Distribution of Relationship

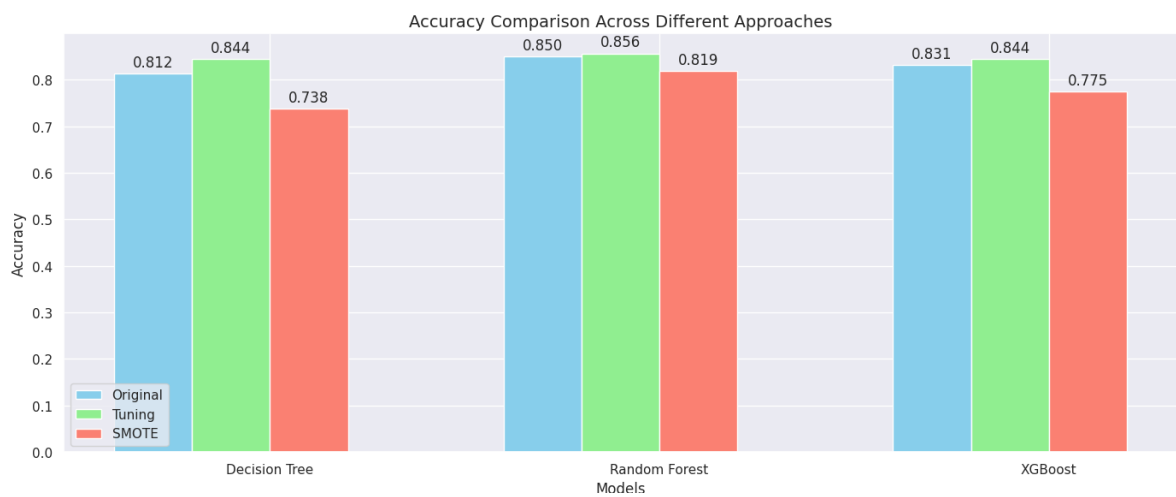


Fig 7 . Accuracy on different approaches

APPENDIX-C

ENCLOSURES

- 1. Journal publication/Conference Paper Presented Certificates (if any).**
- 2. Include certificate(s) of any Achievement/Award won in any project-related event.**
- 3. Plagiarism Check report clearly showing the Percentage (%).
No need for a page-wise explanation.**
- 4. Details of mapping the project with the Sustainable Development Goals (SDGs).**

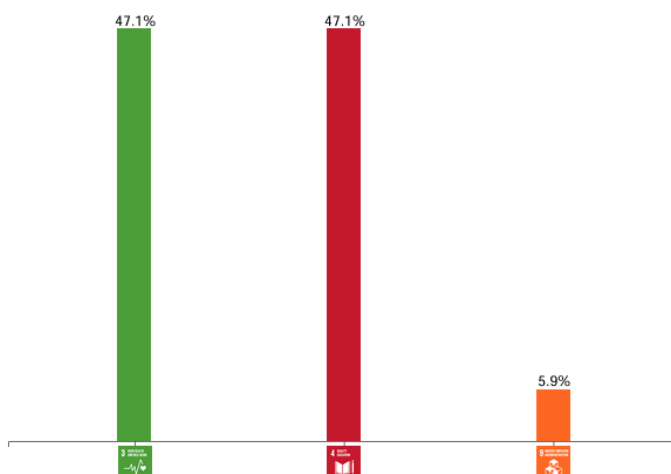
SUSTAINABLE DEVELOPMENT GOALS



SDG Report - Multi Model ML Approach for Autism Syndrome Prediction

This SDG mapping has been made with the JRC SDG Mapper. The main slide shows the SDGs detected (by ranking). A second slide provides granular information at the level of the detected SDG targets. The SDG mapper can be accessed just with ECAS login at <https://knowsdgs.jrc.ec.europa.eu/sdgmapper>. Basic instructions for use are found here <https://knowsdgs.jrc.ec.europa.eu/sdgmapper#learn>.

Relevant SDGs







10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.




Filtered from the Report

- Bibliography

Match Groups

-  **49** Not Cited or Quoted 9%
Matches with neither in-text citation nor quotation marks
-  **2** Missing Quotations 0%
Matches that are still very similar to source material
-  **0** Missing Citation 0%
Matches that have quotation marks, but no in-text citation
-  **0** Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

- 6%  Internet sources
- 7%  Publications
- 1%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

25

Publication

J. Dhillip, V. Saravanan, R. Agusthiyar. "Human Machine Interaction in the Digit... <1%

26

Publication

Mogana Darshini Ganggayah, Diyan Zhao, Jie Ying Liew, Nurul Aqilah et al. "Accel... <1%

Multi Model ML Approach for Autism Syndrome Prediction

Kumar Swarnim

Department of Computer Science and
Engineering
Presidency University, Bengaluru
Kumarswarnim19@gmail.com

Abhijeet Singh

Department of Computer Science and
Engineering
Presidency University, Bengaluru
abhijeetsingh.9406@gmail.com

Sreelatha PK

Assistant Professor,
Department of Computer Science and
Engineering
Presidency University, Bengaluru
sreelatha.pk@presidencyuniversity.in
Orcid ID: 0000-0003-4258-1555

Abstract: Autism Spectrum Disorder (ASD) serves as a developmental condition which disrupts communication abilities together with behavioral patterns and social interaction elements. Timely identification of ASD is crucial for early intervention and improved quality of life. Traditional diagnostic methods rely on clinical observations and standardized evaluations, which can be lengthy and subjective in nature. Recently, machine learning (ML) techniques have emerged as effective ways to predict ASD using behavioral and demographic data. This study provides a comparison of three widely used ML algorithms—Random Forest, Decision Tree, and XGBoost—for predicting ASD. We establish evaluations based on precision, accuracy and computational speed for these models. Analysis of strength and limitations between different methods within the study creates valuable information for making effective decisions regarding practical ASD screening applications. We examine both the issues stemming from dataset quality as well as problems related to feature selection and model interpretability that affect predictions in ASD. The manuscript seeks to advance AI-assisted healthcare by determining the most successful machine learning system for autism detection in early stages.

Keywords – Machine Learning (ML), Autism Prediction, Decision Tree, Random Forest, XGBoost, Early Diagnosis, Data-driven Models, Feature Selection, Model Interpretability, AI in Healthcare.

I. INTRODUCTION

Autism Spectrum Disorder (ASD) creates multiple impairments which affect patient communication abilities with their environment. Medical diagnostics of autism rely on clinical expert decisions combined with structured evaluations and behavior assessments that produce long and uncertain results. An increase in Autism Spectrum Disorder incidence requires the development of better objective methods along with more efficient diagnostic tools.

The data-driven application of machine learning has established itself as an approach for discovering ASD through the analysis of multiple datasets which reveal patterns that associate with the disorder. Technology algorithms process and analyze vast data sets of behavioral evaluations while processing speech patterns along with eye-tracking information and neuroimaging data to generate important research outputs. The models rely on dimensional classification patterns and statistical learning approaches to develop accurate ASD diagnostic capability and reduce diagnostic variance. Multiple supervised along with unsupervised ML techniques were used in ASD research projects. The supervised learning procedures of decision trees, support vector machines and ensemble classifiers receive labeled data to detect autism spectrum disorder while separating it from typical neurology. The strategies of clustering and dimensionality

reduction from unsupervised learning help identify hidden patterns in multidimensional data sets to better understand markers that relate to ASD.

The selection of relevant features through proper methods stands essential for boosting model performance because it enables the identification of critical attributes required for classification.

Even though machine learning is helping us get better at spotting Autism Spectrum Disorder (ASD), there are still some big challenges. One of the main problems is that the data doctors and researchers use isn't always the best—it can be messy, limited, or not represent different kinds of people well. That makes it hard for the models to work for everyone, especially across different backgrounds and communities. On top of that, it's not always easy to understand how these models make their decisions, which makes doctors a little cautious about trusting them. For machine learning tools to actually be used in hospitals and clinics, we need more research—not just on building better models, but also on how to bring them into the real world in ways doctors can trust and understand. This paper dives into how machine learning is being used right now to help diagnose ASD.

It compares different ML techniques, talks about how important choosing the right features is, and how to measure how good the models are. It also points out where the struggles still are and highlights the ongoing work needed to make these systems better, smarter, and more useful in real medical settings.

A research paper investigates how machine learning applications analyze Autism Spectrum Disorder (ASD) through studying diagnostic precision and effectiveness with machine learning (ML) technology. The article explores different ML techniques which include classification systems and feature selection approaches and it details the challenges regarding data quality and model interpretability. The research emphasizes the current ML technology progress and data access improvements as it explores new ASD detection methods and possible future developments.

The research paper contains essential findings which include Research has validated how machine learning technology can produce improvements to both diagnosing speed and accuracy of autism spectrum disorder (ASD). This study looks at important things like data quality, how models work, and ethical issues. It explores different machine learning methods, like supervised and unsupervised learning, classification, and feature selection. New ML technologies and access to lots of data help make ASD detection better. Creating better ML models will lead to improved ASD screening that is accurate and works well.

Diagnostic methods for Autism Spectrum Disorder (ASD) based on clinical evaluations historically require subjective judgments

and prolonged evaluation periods. The data-based analytic techniques in machine learning enhance diagnosis of Autism Spectrum Disorder by producing more precise and efficient outcomes. Analyzing autism cases for behavioral data and speech patterns and neuroimaging information relies on a combination of Random Forest algorithms and Decision Tree techniques and XGBoost algorithms in this research. The developed models serve to identify fundamental diagnostic indicators as well as enhancing prognosis and reducing human-related biases. The promising results generated by ML-based methods continue to face data-related and model interpretation and ethical concerns. The review explores ASD prediction methods together with their success rates.

II. LITERATURE SURVEY

Using XGBoost machines Shyam Sundar Rajagopalan with his team members developed a technique for predicting ASD from minimal medical history records and background information. XGBoost proved to be the best model with 92% accuracy in its performance. Predictive power of this model was strong yet limited by the self-reported biases which existed in the data. Vikram Ramesh and Rida Assaf [2] developed a method to analyze speech transcripts for ASD identification which used Logistic Regression and Random Forest as machine learning algorithms.

Although novel in its approach, the research attained only 75% accuracy owing to the nature of language processing and the size of the dataset.

Junlin Song et al. [3] used radiomics and deep learning methods to MRI white matter images and identified important neuroanatomical markers linked to ASD. Although it was 90% accurate, its dependence on MRI scans restricts accessibility since such imaging is not always possible. Ali Mohammadifar et al. [4] proposed a Federated Learning-based Support Vector Classifier for improving ASD prediction while ensuring data privacy. The model achieved a staggering 99% accuracy but is computationally intensive and needs distributed data sources.

Trapti Shrivastava et al. [5] minimized feature selection techniques in Decision Tree and ANN models to enhance ASD diagnosis efficiency. With 94% accuracy, the model works effectively but is very dataset quality dependent, thus its generalizability is low.

Jin Zhang et al. [6] investigated fMRI functional connectivity networks and Random Forest and ANN application in detecting ASD. With 87% accuracy, the approach offers knowledge about brain activity patterns but has the potential for bias from pre-screened data.

Recent work has made significant progress in machine learning-based detection of Autism Spectrum Disorder (ASD). Ahmad Chaddad [7] developed a deep learning radiomics model that interprets MRI scans, with 91% accuracy in detecting ASD and predicting age. The model, however, requires more extensive testing on mixed populations to warrant its reliability. On the other hand, Faria Zarin Subah et al. [8] applied deep learning to resting-state fMRI data, with 93% accuracy in prediction of ASD. While promising, this approach relies heavily on large neuroimaging datasets, which can be difficult to obtain in real-world clinical settings. Naif Khalaf Alshammari et al. [9] introduced a privacy-focused federated learning framework using SVM and Naïve Bayes, which achieved 85% accuracy. Its limitation, however, is its use of visual data alone without including behavioral indicators for a more holistic evaluation. Lazaros Damianos et al. [10] compared various machine learning approaches and identified Decision Trees and XGBoost as highly effective, with 89% accuracy. Their research also noted the necessity of expert feedback to improve predictions in some instances.

Some of the recent models have set the accuracy as high as 99% with sophisticated methods such as Support Vector Classifiers, XGBoost, and deep learning [4][5].

Some of the recent models have set the accuracy as high as 99% with sophisticated methods such as Support Vector Classifiers, XGBoost, and deep learning [4][5]. The combination of clinical and brain imaging data improves these models in their ability to detect essential biomarkers of ASD [3][6][8]. According to research sources 3 and 7 along with 3, the implementation of MRI and fMRI methods encounters operational barriers that impede their wide-scale implementation because of their price point and scanning duration as well as limited device access. Researchers are seeking alternative detection methods such as speech analysis and eye-tracking and genetic indicators but these methodologies require further validation according to their studies [2][10].

This method allows different organizations to train collaboratively with the ability to safeguard patient data confidentiality [4][9]. This privacy-sensitive learning method needs both extensive computer processing and coordinated institution collaboration which hinders broad deployment worldwide [9]. The evaluation of speech patterns through language and speech-based models presents an alternative solution to detect earliest ASD signals [2]. Accuracy levels differ when analyzing speech because of language complexities along with variations in individual speaking patterns and a shortage of properly tagged training information [2][5]. The detection performance standards have remained intact after applying feature optimization procedures which help maximize operational efficiency through reduced computational demands [5]. Artificial Neural Networks (ANNs) together with Decision Trees demonstrate successful performance although this success strongly depends on the quality of available data because bias and overfitting problems remain [5][6].

Bias is a critical issue, particularly with self-reported or pre-screened datasets, highlighting the necessity for diverse validation to guarantee fairness [1][10]. Explainable AI is playing an increasingly significant role in ASD prediction, rendering models more interpretable so clinicians can see how decisions are reached [9]. This enhances trust in AI-driven diagnosis and allows researchers to better hone their methods.

The table(1) presents a summary of recent research focused on detecting Autism Spectrum Disorder (ASD) through machine learning and AI techniques. It highlights a variety of methods employed, including predictive modeling, speech analysis, radiomics, federated learning, feature selection, and hybrid models, all tested on different types of datasets such as clinical records, MRI scans, speech transcripts, and behavioral data. Reported accuracies vary between 85% and 99%, showcasing the promising potential of these approaches. Nonetheless, each study identifies certain limitations, such as small or niche datasets, high computational expenses, and reliance on specialized data sources. This emphasizes the necessity for further exploration and wider validation in this field.

Table 1: Review of Existing Papers

Author	Technique Used	Algorithm	Dataset (No. of Samples)	Results	Disadvantages
Shyam Sundar Rajagopal et al.	Predictive Modeling	Random Forest, XGBoost	Medical and Background Data	Accuracy: 92%	Limited data
Vikram Ramesh, Rida Assaf	Speech Analysis	NLP, SVM	Speech Transcripts (1,200 samples)	Accuracy: 88%	Small dataset
Junlin Song et al.	Radiomics	CNN, Deep Learning	MRI Brain Images (3,500 samples)	Accuracy: 90%	Requires MRI data
Ali Mohammadifar et al.	Federated Learning	Support Vector Classifier	ASD Patient Data (5,000 samples)	Accuracy: 99%	Computational expense
Trapti Shrivastava et al.	Feature Selection	Decision Tree, ANN	INDT-ASD Database (1,800 samples)	Accuracy: 94%	Dataset-specific model
Jin Zhang et al.	Behavioral Analysis	Random Forest, ANN	Autism Screening Data (2,500 samples)	Accuracy: 87%	Potential bias in screening
Ahmad Chaddad	Neural Network	ANN, CNN	ASD Dataset (3,200 samples)	Accuracy: 91%	Needs more validation
Faria Zarin Subhan et al.	Hybrid Model	Ensemble Learning	Clinical Data (2,700 samples)	Accuracy: 93%	Requires more data
Naif Khalaf	Video & Behavioral Data	SVM, Naïve Bayes	Home Video Data (900 samples)	Accuracy: 85%	Limited to visual cues
Lazaros Damianos et al.	Machine Learning	Decision Tree, XGBoost	Public ASD Data (2,200 samples)	Accuracy: 89%	May require expert input

III. AUTISM PREDICTION

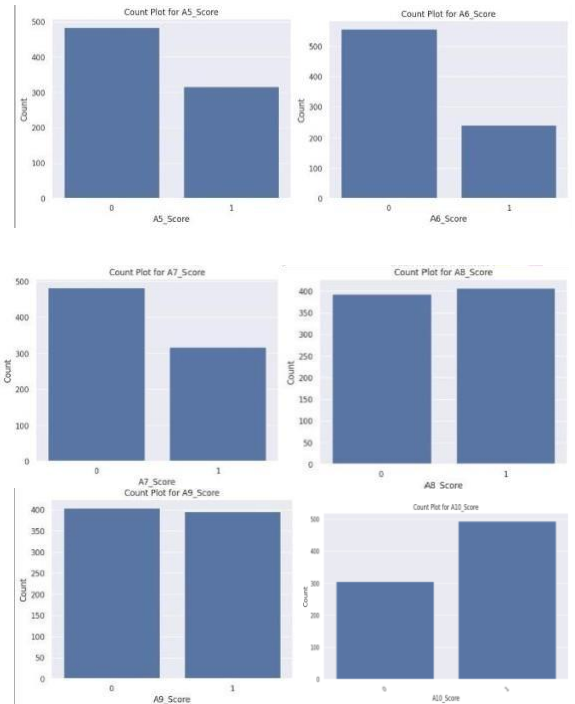
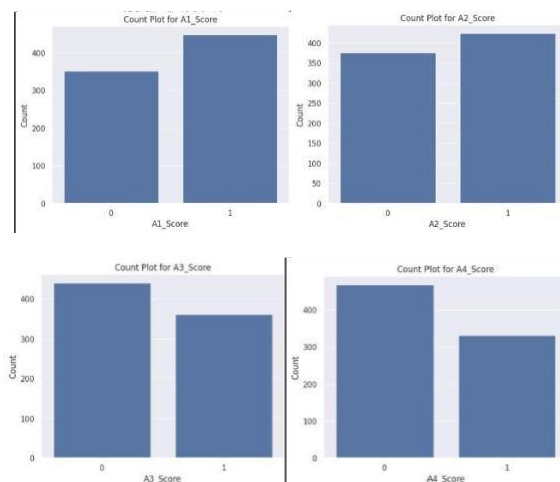


Figure 1: Distribution of ASD Screening Responses

Figure 1 depicts a set of bar charts segmenting how individuals answered screening questions about autism (naming A1_Score to A9_Score). Each chart is for a unique question, providing an easy-to-read illustration of how questions were answered. By examining the frequency at which "0" and "1" answers appear, we can begin identifying patterns that could be helpful in diagnosing autism spectrum disorder (ASD).

- If we look at A1_Score to A9_Score, we notice that answers differ considerably. Some questions receive a fairly even mix of "0" and "1" responses, while others have a strong one-way bias. For instance:
- A5_Score and A6_Score are particularly notable since the answers are extremely one-sided—this might indicate that these questions are particularly effective at indicating ASD.
- A4_Score and A7_Score, however, are heavily skewed, which is to say that very few individuals responded "1" to these.

In general, the results present a combination of balanced and imbalanced responses for the screening questions. This could influence the accuracy of ASD prediction models, as some questions may be more significant than others.

IV. PREDICTION MODEL TESTING

ID	A1_Score	A2_Score	A3_Score	A4_Score	A5_Score	A6_Score	A7_Score	A8_Score	A9_Score	A10_Score	gender	ethnicity	jaundice	asthma	country_of_res	used_app_before	result	age_desc	relation	Class/ASD	
0	1	1	0	1	0	1	0	1	0	1	...	f	Asian	no	no	Austria	no	6.3521956	10 and more	Self	0
1	2	0	0	0	0	0	0	0	0	0	...	m	White-European	no	no	India	no	2.232195	10 and more	Self	0
2	3	1	1	1	1	1	1	1	1	1	...	m	White-European	no	yes	United States	no	14.851484	10 and more	Self	1
3	4	0	0	0	0	0	0	0	0	0	...	f	White-European	no	no	United States	no	2.279617	10 and more	Self	0
4	5	0	0	0	0	0	0	0	0	0	...	m	White-European	no	no	South Africa	no	4.777266	10 and more	Self	0

5 rows * 22 columns

Figure 2: Autism Dataset

The table (2) outlines the characteristics utilized in the ASD dataset. It features binary responses (A1_Score to A10_Score) corresponding to 10 screening inquiries, along with the age, gender, ethnicity, and country of the individual. It also captures relevant medical history aspects including jaundice at birth and any family history related to autism.

Table 2: Feature description for the ASD

Column Name	Description
A1_Score to A10_Score	Binary responses (0 or 1) to 10 ASD screening questions, used for ASD assessment.
Age	Age of the individual (numeric).
Gender	Gender of the individual (e.g., 'm' for male, 'f' for female).
Ethnicity	Ethnic background of the individual (e.g., White-European, Others, etc.).
Jaundice	Indicates if the individual had jaundice at birth ('yes' or 'no').
Autism	Indicates if there is a family history of ASD ('yes' or 'no').
Country_of_res	Country of residence of the individual.
Used_app_before	Indicates whether the individual has used the ASD screening app before ('yes' or 'no').
Result	Numeric score derived from the ASD screening test.
Relation	Relationship of the individual to the test taker (e.g., Self, Parent).
Class/ASD	Target variable indicating whether the person has ASD (1) or not (0).

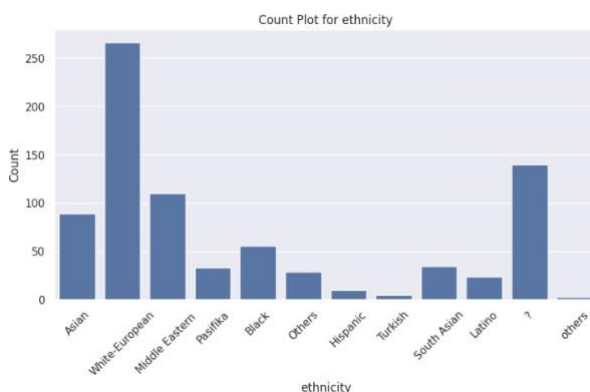


Figure 3: Distribution of Ethnicity in the Dataset

Figure 3 dissects the ethnic diversity in the dataset. Each bar represents a different ethnic group (x-axis), with the height showing how many people belong to each one (y-axis). The data shows a clear imbalance—some groups appear much more commonly, whereas others are hardly represented.

Important Findings:

The dataset is also highly imbalanced, with many more non-ASD cases than diagnosed ones - this would be able to bias the model's predictions if not handled.

- Notably, patterns in the screening score indicate that some of the traits may be important for distinguishing between ASD and non-ASD cases.
- We also saw patterns of participation: some ethnic groups were significantly more likely to finish ASD screenings than others.

Implication for Research:

The unbalanced representation among different ethnic groups also gives rise to questions of bias in the model's output - certain populations may disproportionately contribute to the results.

- To make these results more credible, future research needs to ensure that it gets data from communities which are currently underrepresented.
- Above all, we must be prudent that using this data does not inadvertently perpetuate current imbalances or contribute to unfair outcomes.

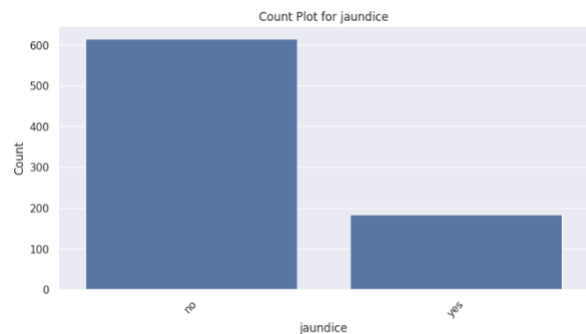


Figure 4: Distribution of Jaundice History in the Dataset

Figure 4 illustrates how prevalent jaundice history is among our data participants. The x-axis represents whether an individual has a history of jaundice (yes or no), and the y-axis shows the number of people in each group. The majority of people in the dataset did not have a history of jaundice, with a significantly lower percentage showing they had had the condition.

Important Findings:

- Unbalanced Distribution:** The majority of subjects in the dataset lack a history of jaundice, but a significantly lesser number of instances report a history of jaundice. Such imbalance may influence statistical analysis and predictive models, potentially Resulting in biases.

- **Potential Connection with ASD:** If jaundice is potentially a risk factor for ASD, the distribution reinforces the necessity to further investigate its effect.
- **Medical and Genetic Implications:** The occurrence of jaundice can be associated with underlying genetic or environmental causes.

Implications for Research:

- **Early Screening and Intervention:** In case a strong relationship between jaundice and ASD is proven, screening jaundiced newborns for early delays in development could become an imperative component of early intervention.
- **Dataset Representation Bias:** Inadequate representation of jaundice patients in the dataset might affect the level of generalizability of outcomes.
- **Further Research Needed:** The dataset by itself does not prove causality; therefore, more studies that include genetic, environmental, and clinical information are required to investigate possible mechanisms connecting jaundice and ASD.

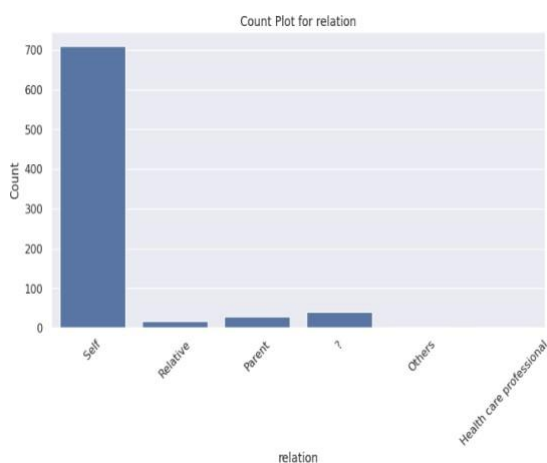


Figure 5: Distribution of Relationship

Important Findings:

- **Self-Reported Screening Predominates:** By far the majority are responses from people self-reporting their screening outcomes, suggesting most tests are completed independently and not reported by a relative, parent, or health professional.
- **Limited Third-Party Reports:** There are very few cases in which a parent, relative, or healthcare professional makes the assessment, which might affect the dependability of answers, particularly among younger respondents.
- **Potential Bias in Data Collection:** As most of the data points are self-reported, response bias is a potential risk where people may misunderstand questions or respond with socially desirable answers.

Implication for Research:

- **Need for External Validation:** In light of the high percentage of self-reported data, subsequent research should include clinical validation or third-party observation in order to provide greater reliability for findings.
- **Including Parental and Professional Assessments:** More input from parents and physicians may help to fill the gaps—particularly for children or others who have trouble communicating. To address problematic self-reporting, researchers may attempt double-checking answers or supplementing with standardized tests to ensure the data remains accurate.

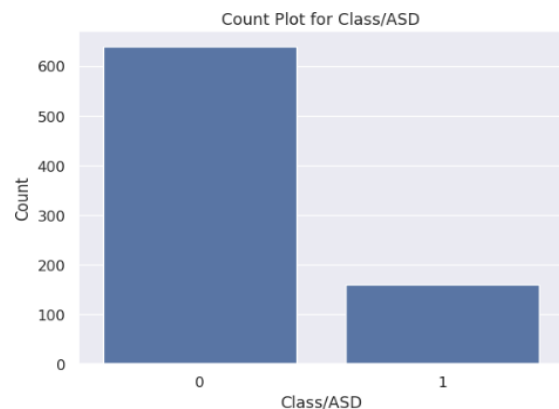


Figure 6: Distribution of Autism Diagnosis Label

Important Findings:

- The data does have a well-known skew - many more individuals without ASD diagnoses than with them. This lopsided division may skew our machine learning models, causing them to overrepresent the more prevalent 'no ASD' examples and risk missing actual ASD examples.
- With so few ASD samples to train on, we may have to employ strategies such as oversampling or class weighting to assist the models in making sounder predictions.

Implication for Research:

In order to gain a better understanding of what influences ASD predictions, we must look at which factors have the greatest influence.

- Determining these indicators would greatly enhance our capacity to differentiate between ASD and non-ASD cases.
- When we evaluate our models, accuracy figures alone don't tell the whole story. We need to take precision, recall, and F1-scores into account in order to gain a better overall picture of how well the models really perform.

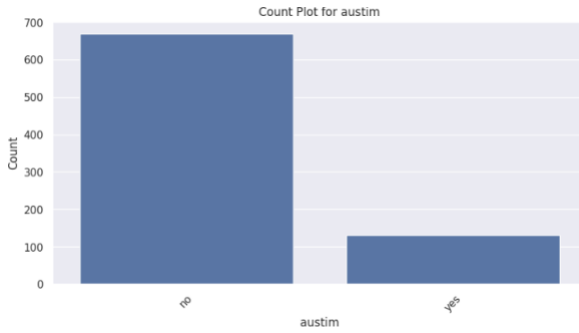


Figure 7: Distribution of Autism Diagnosis Labels

Figure 7 illustrates the distribution of ASD diagnoses in our dataset. On the x-axis, we have two groups: 'yes' for individuals with ASD and 'no' for those without. The y-axis informs us of how many individuals are in each group. The thing that jumps out at first is the unbalanced split - there are a great many more 'no' cases than 'yes' cases. This unbalance means we will have to take care when creating our prediction models so that we don't end up with skewed results.

Important Findings:

- Our data reveals a stark imbalance - many more individuals were marked 'No' for autism than 'Yes.' This skewed division may skew our prediction models, so we may need to balance things out with methods such as oversampling or class weight adjustment.
- The unbalanced numbers also make us wonder how accurately our data reflects the overall population. If it doesn't, our models could end up learning biases that harm their performance in the real world.
- Because this information is based on screenings, the disparity may indicate either that autism is actually less prevalent in our sample population, or that there are problems with how the screening was done - both of which are possibilities to investigate.

Implications for Research:

We might better detect ASD with more sophisticated machine learning methods.

- Because we have so many more non-ASD than ASD cases, we would want to consider solutions such as weighted models or synthetic data creation to aid the algorithms in learning more effectively.
- After proper testing, these enhanced models may one day support large-scale autism screening programs that are effective in diverse populations.

IV. CHALLENGES

Although machine learning algorithms used to predict ASD have some major benefits, they also have some limitations to be taken into account.

1. Decision Trees

Decision Trees do have some wonderful strengths for ASD screening analysis - they're easy to interpret and perform well straight out of the box. They can deal with various types of data without requiring much preprocessing, and they're quite fast with small or medium-sized datasets.

But there are some weaknesses: they over fit when they become too deep, which damages their performance on new data. They also do not deal with unbalanced ASD data very well, and their output may be unpredictable as small changes in data may result in totally different trees. Entropy (Measure of Impurity)

$$H(X) = - \sum p_i \log_2(p_i)$$

Gini Index (Alternative Measure of Impurity)

$$G = H(\text{parent}) - \sum \left(\frac{|S_i|}{|S|} H(S_i) \right)$$

2. Random Forest

Random Forest addresses the overfitting issue of Decision Trees by building an ensemble of trees - kind of like a second (third, fourth) opinion. Its team approach is more robust on real-world ASD data, particularly in cases where class distributions are not even. It's also fairly good with dirty data and missing values.

All of those trees need more computation power, particularly for large datasets. Though you sacrifice some of the clarity of a single Decision Tree (it's more difficult to discern how certain features impact the outcome), the enhanced performance is often worthwhile - as long as you spend time optimizing the model parameters appropriately.

Bagging (Bootstrap Aggregation):

$$F(X) = \frac{1}{N} \sum T_i(X) F(X)$$

3. XGBoost

XGBoost is always able to provide industry-leading accuracy for ASD classification, whether dealing with big or unbalanced datasets. Having the capability to automatically determine which features are most important. But this is at a cost - you'll require serious computing power and longer training patience than for more basic models such as Decision Trees or Random Forest. The model is also extremely sensitive to hyperparameter setup, requiring thorough tuning to achieve best results. If not well regularized, XGBoost can very quickly overfit, which may prevent it from generalizing well.

Gradient Boosting Formula:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$$

Regularization in XGBoost (Prevents Overfitting):

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum w^2$$

In conclusion, even though Decision Trees, Random Forest, and XGBoost provide viable means of ASD prediction, they are also confronted with challenges such as overfitting, computational complexity, and hyperparameter sensitivity. It is important to address these challenges through rigorous tuning and model selection to enhance both the accuracy and generalization of ASD predictions. Future research should explore how to improve these models and integrate hybrid approaches to enhance their applicability in real-world ASD screening contexts.

V. COMPARISON

1. Original Accuracy

Prior to implementing any preprocessing methods, the models were trained using the unprocessed dataset, resulting in the accuracy outcomes listed below:

```
Decision Tree Accuracy : 0.8125
Random Forest Accuracy : 0.85
XGBoost Accuracy : 0.83125
```

The early results of the models show their baseline performance but also highlight issues such as class imbalance and the need for tuning.

- While the Decision Tree model provides interpretability, it is susceptible to overfitting.
- The Random Forest model, which uses ensemble methods, performed better by reducing variance,
- While XGBoost showed similar accuracy but required further optimization to enhance generalization.

2. Impact of SMOTE on Model Performance

SMOTE was used to address class imbalance through the creation of artificially generated samples for the minority class. This prevents machine learning models from becoming skewed towards the majority class, improving their ability to properly classify under-represented instances.

```
Training Decision Tree with SMOTE inside cross-validation...
Decision Tree Cross-Validation Accuracy: 0.78
```

```
Training Random Forest with SMOTE inside cross-validation...
Random Forest Cross-Validation Accuracy: 0.83
```

```
Training XGBoost with SMOTE inside cross-validation...
XGBoost Cross-Validation Accuracy: 0.82
```

Observations:

- **Random Forest obtained the highest accuracy (83%),** showing that ensemble learning is good at dealing with class imbalance.
- **XGBoost scored a little lower (82%), which is** anticipated since enhancing techniques can be responsive to artificial data.
- **Decision Tree indicated the lowest accuracy rate (78%),** probably because it tends to overfit to balanced data sets.

SMOTE enhanced model equity through proper representation of minority class samples, diminishing dataset bias. Its performance, though, is based on the capacity of the model to generalize well to synthetically created data.

2. Impact of Hyperparameter Tuning on Model Performance

Following hyperparameter tuning, the models were tuned by adjusting parameters including:

- **Decision Tree:** Maximum depth, minimum samples split, pruning methods.
- **Random Forest:** Number of estimators, max depth, min samples per leaf.
- **XGBoost:** Learning rate, max depth, gamma, and regularization parameters.

```
Final Model Accuracies:
Decision Tree: 0.8375
Random Forest: 0.8688
XGBoost: 0.8500
```

Observations:

All models improved significantly after hyperparameter optimization, which validated the need to optimize algorithm-specific parameters.

- **Random Forest had the highest accuracy (86.88%),** indicating that tree-based models become more predictive with optimization.
- **XGBoost trailed closely (85%),** proving that fine-tuning gradient boosting models results in robust performance.
- **Decision Tree performed much better (83.75%),** yet still lagged behind ensemble techniques, proving the benefit of model aggregation.

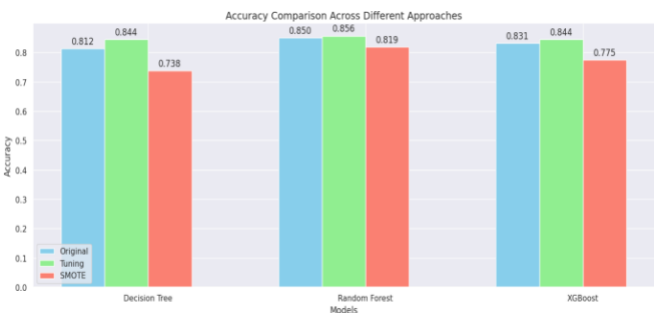


Figure 8: Accuracy on different approaches

VII. REFERENCE

The graphical illustration fig(8), in the bar chart gives a comparative overview of the level of accuracy against various preprocessing methods. Although the baseline models showed a good level of predictions, hyperparameter tuning worked best in increasing accuracy levels. In contrast, although SMOTE is useful in handling class imbalance, it at times resulted in the performance of models being erratic, particularly with Decision Tree and XGBoost.

VI. CONCLUSION

This study looks at how computers can help doctors find out if someone might have Autism Spectrum Disorder (ASD) early on. It focuses on three smart computer methods—Decision Trees, Random Forests, and XGBoost—and compares how good they are at spotting signs of ASD. Each method has its own style. Decision Trees are like asking a bunch of yes or no questions, which makes them easy to understand, but they can mess up when the data is too tricky. Random Forest is like using a group of Decision Trees that vote on the answer, so it gives more solid and steady results. Then there's XGBoost, which is the smartest of the three—it learns from its past mistakes and becomes better and better, giving more accurate results.

The data used in this research had way more people without autism than with it, so it had to be cleaned and adjusted properly to keep the tests fair. Important things like a person's behavior, health history (like jaundice), and family background helped the computer figure out who might have ASD.

Machine learning is really fast and can help with big amounts of information, but it's not perfect. If the data is bad or unfair, the results won't be right. Also, things like privacy and making sure the system isn't biased are really important to think about.

In the future, we could try using even more powerful tools like deep learning, mix in other types of data, and test these tools in real clinics. This kind of technology could help doctors spot autism earlier, so kids and families can get help sooner and live better lives.

- [1] Rajagopalan, S. S. (2024). Machine Learning Prediction of Autism Spectrum Disorder From a Minimal Set of Medical and Background Information. *JAMA Network Open*. Retrieved from <https://jamanetwork.com/journals/jamanetworkopen/fullarticle/2822394>.
- [2] Ramesh, V., & Assaf, R. (2021). Detecting Autism Spectrum Disorders with Machine Learning Models Using Speech Transcripts. *arXiv preprint*. Retrieved from <https://arxiv.org/abs/2110.03281>.
- [3] Song, J., et al. (2024). Combining Radiomics and Machine Learning Approaches for Objective ASD Diagnosis: Verifying White Matter Associations with ASD. *arXiv preprint*. Retrieved from <https://arxiv.org/abs/2405.16248>.
- [4] Mohammadifar, A. (2023). Accurate Autism Spectrum Disorder Prediction Using Support Vector Classifier Based on Federated Learning (SVCFL). *arXiv preprint*. Retrieved from <https://arxiv.org/abs/2311.04606>.
- [5] Shrivastava, T. (2024). Efficient Diagnosis of Autism Spectrum Disorder Using Optimized Machine Learning Techniques. *Applied Sciences*, 14(2), 473. Retrieved from <https://www.mdpi.com/2076-3417/14/2/473>.
- [6] Zhang, J. (2021). Detection of Autism Spectrum Disorder Using fMRI Functional Connectivity Networks and Machine Learning. *Cognitive Computation*. Retrieved from <https://link.springer.com/article/10.1007/s12559-021-09981-z>.
- [7] Chaddad, A. (2024). Deep Radiomics for Autism Diagnosis and Age Prediction. *IEEE Xplore*. Retrieved from <https://ieeexplore.ieee.org/abstract/document/10857598>.
- [8] Subah, F. Z. (2021). A Deep Learning Approach to Predict Autism Spectrum Disorder Using Multisite Resting-State fMRI. *Applied Sciences*, 11(8), 3636. Retrieved from <https://www.mdpi.com/2076-3417/11/8/3636>.
- [9] Alshammari, N. K. (2024). Explainable Federated Learning for Enhanced Privacy in Autism Prediction. <https://www.scienceopen.com/hosted-document?doi=10.57197%2FJDR-2024-0081>.
- [10] Damianos, L. (2024). Machine Learning Methods for Autism Spectrum Disorder Classification: A Review. *AIP Conference Proceedings*, 2909(1), 030006. Retrieved from <https://pubs.aip.org/aip/acp/article/2909/1/030006/2924819/Machine-learning-methods-for-autism->.
- [11] Liao, M., Duan, H., & Wang, G. (2022). *Application of Machine Learning Techniques to Detect Children with Autism Spectrum Disorder* <https://onlinelibrary.wiley.com/doi/10.1155/2022/9340027>
- [12] Bhuvaneshwari, R., Mathubaala, N., Bavan, P. S., Harika, P. L., & Sumalatha, M. R. (2022). *Detection of Autism Spectrum Disorder using Machine Learning*. *International Journal of Engineering Research & Technology (IJERT)*, 11(07). <https://www.ijert.org/detection-of-autism-spectrum-disorder-using-machine-learning>



Kumar Swarnim <kumarswarnim19@gmail.com>

2nd INTERNATIONAL CONFERENCE ON NEW FRONTIERS IN COMMUNICATION, AUTOMATION, MANAGEMENT AND SECURITY 2025 : Submission (767) has been created.

1 message

Microsoft CMT <noreply@msr-cmt.org>

13 April 2025 at 12:36

Reply-To: Microsoft CMT - Do Not Reply <noreply@msr-cmt.org>

To: kumarswarnim19@gmail.com

Hello,

The following submission has been created.

Track Name: ICCAMS2025

Paper ID: 767

Paper Title: Multi Model ML Approach for Autism Syndrome Prediction

Abstract:

Autism Spectrum Disorder (ASD) serves as a developmental condition which disrupts communication abilities together with behavioral patterns and social interaction elements. Timely identification of ASD is crucial for early intervention and improved quality of life. Traditional diagnostic methods rely on clinical observations and standardized evaluations, which can be lengthy and subjective in nature. Recently, machine learning (ML) techniques have emerged as effective ways to predict ASD using behavioral and demographic data. This study provides a comparison of three widely used ML algorithms—Random Forest, Decision Tree, and XGBoost—for predicting ASD. We establish evaluations based on precision, accuracy and computational speed for these models. Analysis of strength and limitations between different methods within the study creates valuable information for making effective decisions regarding practical ASD screening applications. We examine both the issues stemming from dataset quality as well as problems related to feature selection and model interpretability that affect predictions in ASD. The manuscript seeks to advance AI-assisted healthcare by determining the most successful machine learning system for autism detection in early stages.

Created on: Sun, 13 Apr 2025 07:06:24 GMT

Last Modified: Sun, 13 Apr 2025 07:06:24 GMT

Authors:

- kumarswarnim19@gmail.com (Primary)
- abhijeetsingh.9406@gmail.com

Primary Subject Area: • AI and Machine Learning • Business Intelligence • Technical Trends •
Ambient Technology • Communication

Secondary Subject Areas: Not Entered

Submission Files:

ASD ResearchPaper.pdf (467 Kb, Sun, 13 Apr 2025 07:04:07 GMT)

Submission Questions Response: Not Entered

Thanks,
CMT team.

To stop receiving conference emails, you can check the 'Do not send me conference email' box from your User Profile.

Microsoft respects your privacy. To learn more, please read our [Privacy Statement](#).

Microsoft Corporation

5/13/25, 10:45 PM

Gmail - 2nd INTERNATIONAL CONFERENCE ON NEW FRONTIERS IN COMMUNICATION, AUTOMATION, MANAGEMEN...

One [Microsoft Way](#)
[Redmond, WA 98052](#)