Autism Spectrum Disorder screening using Machine Learning

Kavana Manvi Krishnamurthy
Depaul University
Chicago, USA
kmanvikr@depaul.edu

Abstract*—Autism Spectrum Disorder (ASD) is a neurodevelopmental condition characterized by challenges in social communication, repetitive behaviors, and restricted interests. Early diagnosis is essential for improving outcomes, yet conventional diagnostic methods are often time-consuming and costly. This paper presents a machine learning approach for enhancing ASD screening using a novel dataset focused on behavioral indicators and demographic features. The dataset, which includes 704 instances and 20 features, was pre-processed, cleaned, and subjected to feature selection techniques such as Mutual Information and Recursive Feature Elimination to improve classification performance. Several machine learning models, including XGBoost, SVM, Logistic Regression, Random Forest, and MLP, were evaluated using hyperparameter tuning via GridSearchCV, with a focus on sensitivity as the key metric. The evaluation strategy incorporated Stratified K-Fold cross-validation, ensuring balanced class representation. The findings emphasize the importance of sensitivity in the context of ASD screening, highlighting the potential for machine learning models to offer a more accessible, accurate, and scalable tool for early detection. Among the models tested, SVM with the RBF kernel exhibited the best sensitivity, making it the most effective for identifying ASD cases. Additionally, efforts to mitigate overfitting were implemented to ensure robust and generalizable model performance.

Keywords*—Autism Spectrum Disorder (ASD), Machine Learning, Early Detection, Sensitivity, SVM (Support Vector Machine), RBF Kernel, Feature Selection, Hyperparameter Tuning, GridSearchCV, Random Forest, XGBoost, Logistic Regression, MLP (Multilayer Perceptron), Classification, Stratified K-Fold Cross-Validation, Behavioral Indicators, Diagnostic Screening

I. Introduction

Autism Spectrum Disorder (ASD) is a multifaceted neurodevelopmental condition marked by persistent challenges in social communication, restricted interests, and repetitive behaviors. Early identification of ASD is critical, as timely interventions can significantly improve developmental trajectories, reduce long-term healthcare burdens, and enhance quality of life for individuals and their families. Despite its importance, conventional diagnostic pathways often involve prolonged evaluation processes, specialized clinical expertise, and high costs, creating barriers to accessibility and delaying critical support for many.

The increasing global prevalence of ASD has intensified demand for scalable and efficient screening methods. Current diagnostic frameworks rely heavily on subjective clinical assessments, which are resource-intensive and prone to variability. These limitations exacerbate disparities in care, particularly in underserved communities, and highlight the urgent need for standardized, cost-effective tools capable of streamlining early detection.

A significant challenge in advancing ASD screening lies in the scarcity of comprehensive behavioral datasets. Existing research often prioritizes genetic or neurological markers over observable behavioral traits, limiting insights into practical screening applications. To address this gap, this study introduces a novel dataset focused on ASD behavioral indicators in adults, encompassing 20 features derived from validated screening tools and demographic factors. By integrating behavioral metrics such as the *Autism-Spectrum Quotient (AQ-10-Adult)* with individual characteristics, this dataset aims to enhance machine learning models' ability to distinguish ASD cases from neurotypical profiles reliably. Such advancements could pave the way for accessible, objective screening tools that complement clinical evaluations, ultimately bridging the gap between early suspicion and formal diagnosis.

II. Literature review

Machine learning (ML) has become a powerful tool in detecting Autism Spectrum Disorder (ASD), utilizing various algorithms to analyze datasets and identify patterns that may be indicative of the disorder. One significant study by Raj and Masood [1] explored the application of several ML classifiers, including Support Vector Machines (SVM), Naïve Bayes, Convolutional Neural Networks (CNN), and others, to ASD classification. Their research demonstrated that CNN achieved the highest accuracy of 99.53%, highlighting the potential of deep learning models in this domain. Other classifiers such as K-Nearest Neighbors (KNN) and SVM showed competitive performance, with accuracies ranging from 95.75% to 98.11%. This work emphasizes the importance of selecting optimal ML models for maximizing prediction accuracy in ASD detection.

Feature selection is a critical aspect of machine learning-based ASD detection. Thabtah et al. [2] employed various feature selection techniques, including Variable Analysis (Va), Information Gain (IG), and Correlation Attribute Evaluation, to improve the performance of ASD screening. Their findings revealed that the Va method was particularly effective, reducing feature redundancy and enhancing the model's efficiency. This study underscores the need for effective feature reduction to ensure that only the most relevant attributes are considered, thereby improving classification performance while minimizing the computational load.

Another noteworthy study, conducted by Reghunathan et al. [3], focused on applying the Cuckoo Search Algorithm (CSA) to reduce the number of features in ASD classification tasks. The research utilized datasets across different age groups, including children, adolescents, and adults, and tested classifiers such as Logistic Regression (LR), K-Nearest Neighbors (KNN), and Support Vector Machines (SVM). The results showed that CSA was successful in reducing the feature set without sacrificing accuracy, with Logistic Regression achieving the highest accuracy for adult datasets. This study demonstrates the potential of metaheuristic algorithms like CSA in improving the efficiency of machine learning models for ASD classification.

In addition to feature selection, classifiers' ability to adapt to various age groups is an essential factor in enhancing ASD detection systems. Farooq et al. [4] explored the classification of ASD across different age groups, including children and adults, by utilizing a Multilayer Perceptron (MLP) model combined with the ReliefF feature selection method. Their study found that MLP achieved 100% accuracy across all age groups, highlighting the significance of tailored feature selection methods in improving the model's robustness across diverse populations. This research emphasizes the importance of age-specific adaptations when designing ASD detection systems.

Another significant contribution to the field is the work of Subhash and Motagi [5], who focused on classifying ASD across different age groups using publicly available datasets. Their study utilized multiple machine learning models to demonstrate the efficiency of these models in ASD screening. The work highlights how ML models can be applied to non-clinical datasets to predict ASD in various age groups, with promising accuracy levels. This research also showcases the potential of machine learning models to enhance the accessibility of early diagnosis through non-clinical data sources.

Chowdhury and Iraj [6] further explored the potential of ML classifiers in predicting ASD by incorporating various algorithms, including SVM, Logistic Regression, Random Forest, XGBoost, and MLP. The study's findings underscore the potential of these classifiers in early ASD detection, emphasizing that machine learning techniques can significantly improve the accuracy and efficiency of ASD prediction, particularly in identifying behavioral patterns that may be indicative of the disorder. This research highlights the role of machine learning in offering timely and accurate diagnosis of ASD through the analysis of behavioral features.

III. DATASET OVERVIEW

The dataset is designed to address the challenges of ASD diagnosis by offering a structured format that captures influential traits associated with autism. It integrates nominal and continuous variables alongside binary attributes, enabling machine learning models to effectively distinguish between ASD and non-ASD cases. The inclusion of diverse feature types supports robust classification while balancing sensitivity and specificity—key metrics in medical diagnostics. By leveraging this dataset, researchers can develop scalable and efficient screening tools to enhance early detection of ASD and improve access to timely interventions.

The dataset used in this study was obtained from Dr. Fadi Fayez Thabtah of the Department of Digital Technology at Manukau Institute of Technology in Auckland, New Zealand[7]. Dr. Thabtah developed this dataset specifically for autism screening in adults, containing 20 features designed to be utilized for further analysis in determining influential autistic traits and improving the classification of ASD cases. The dataset was collected using a mobile application called ASDTests, which incorporates behavioral indicators from the AQ-10 adult screening method along with individual characteristics.

A. Dataset

The dataset utilized in this study focuses on Autism Spectrum Disorder (ASD) screening and is tailored for classification tasks. Table 1. comprises 704 instances and 21 attributes, encompassing a mix of categorical, binary, and continuous data types. The features include behavioral indicators, demographic information, and screening results, making it highly relevant for medical, health, and social science applications. Notably, the dataset contains missing values, necessitating preprocessing steps to ensure data quality.

The dataset, although extensive, exhibits a significant class imbalance in the target variable (Class/ASD). Specifically, there is a disproportionate distribution, with 515 instances labeled as non-ASD (0) and 189 instances labeled as ASD (1). This imbalance means that the majority class (non-ASD) greatly outweighs the minority class (ASD), which can create challenges for machine learning models. Without proper handling, models may become biased toward the majority class, reducing their ability to accurately detect ASD cases. Therefore, addressing this imbalance is essential to ensure the development of a reliable screening tool that maintains strong sensitivity and specificity across both classifications.

B. Exploration

In the exploratory data analysis phase, I utilized matplotlib to create a series of visualizations that offer insights into the dataset's features. Figure 1 presents bar

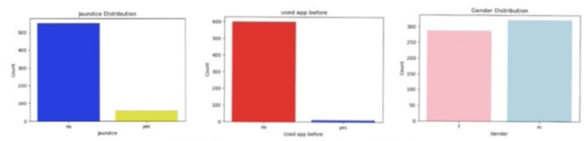

Fig 1. Binary categorical feature distribution

graphs illustrating the distribution of binary categorical variables, including jaundice history, gender, and prior app usage. These graphs reveal the prevalence of each category within these features. Figure 2 focuses on multi-valued categorical attributes, specifically the country of origin and the relationship of the test-taker to the subject. This visualization highlights that the majority of tests were self-administered, and the dataset encompasses participants from 50 different countries, indicating a diverse sample. Figure 3 has histograms for age and result variables. Age variable is skewed. Figure 4 showcases a stacked bar graph representing the responses to the AQ-10 Adult questionnaire (questions A1-A10). This graph provides a clear visual representation of the yes/no answers for each question, allowing for a quick assessment of response patterns across the autism screening items. These visualizations collectively offer a comprehensive overview of the dataset's composition and the distribution of key features relevant to autism screening

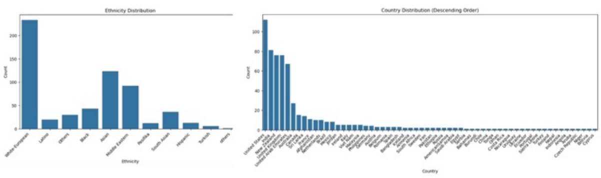

Fig 2. Categorical multi value feature distribution

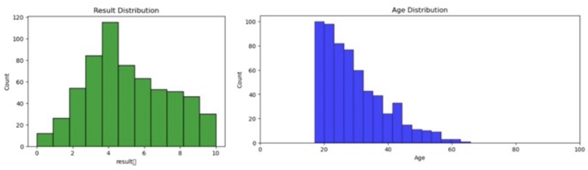

Fig 3. Numerical feature distribution

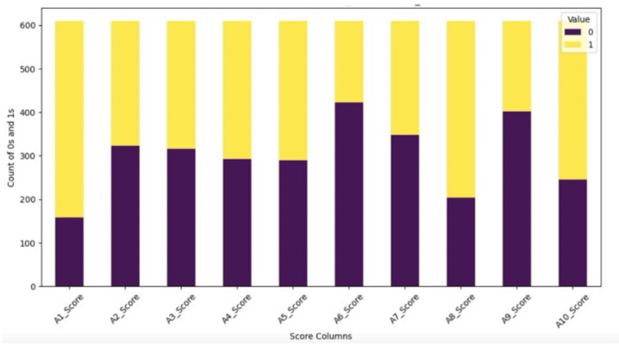

Fig 4. Distribution of 0s and 1s in A1_Score to A10 Score

IV. Methodology

A. Data Cleaning

After the text edit has been completed, the paper is ready for the In preparation for analysis, I conducted data cleaning steps to enhance the dataset's quality. This process involved correcting spelling errors in column names to ensure consistency. To address missing values, I employed strategic approaches tailored to each feature. For the 'age' variable, I imputed missing entries with the mode, preserving the dataset's overall age distribution. Notably, I made the decision to remove instances with missing ethnicity information, as these were predominantly associated with non-autism cases. This choice was motivated by the goal of refining the model's focus on autism-specific characteristics, potentially improving sensitivity while minimizing bias. After these steps, the refined dataset comprised 609 rows and 18 columns, providing a robust foundation for subsequent analysis and model development. Notably, the "result" column was removed from the dataset due to its high correlation (0.83) with the target variable "Class/ASD", as revealed in Figure 5's correlation matrix. This decision was made to prevent data leakage and ensure the model's generalizability by focusing on more independent predictive features. These preprocessing steps collectively aim to optimize the dataset for machine learning applications while maintaining the integrity of the information relevant to autism screening. This careful curation of the data aims to balance the preservation of valuable information with the removal of potentially confounding factors, setting the stage for more accurate autism screening predictions.

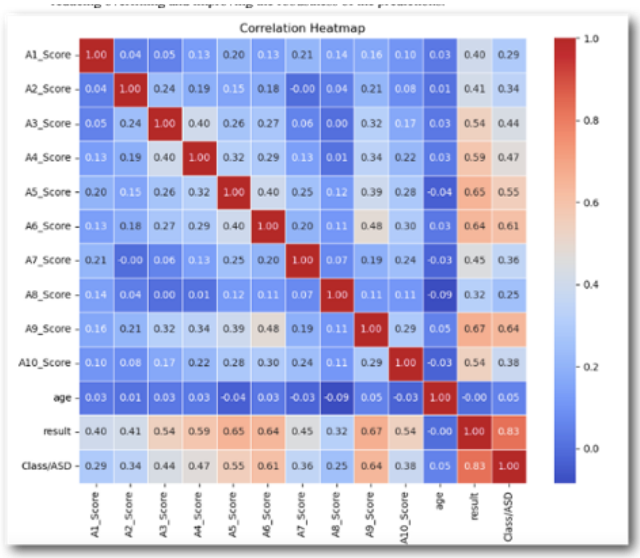

Figure 5. Correlation Matrix

B. Data Pre-processing

Converting categorical variables to binary values (0/1) is a crucial pre-processing step, particularly for algorithms like logistic regression and neural networks. This transformation allows these models to interpret categorical data numerically, enabling them to perform calculations and make predictions effectively. For instance, converting "yes/no" responses to 0/1 allows logistic regression to assign appropriate weights to these features. In the context of Support Vector Machines (SVM), this binary representation helps in creating optimal decision boundaries. For ensemble methods like XGBoost, binary encoding facilitates efficient splitting decisions in tree construction.

One-hot encoding is essential for categorical features with multiple distinct values, as it prevents the introduction of arbitrary ordinal relationships. This technique creates separate binary columns for each category, which is particularly beneficial for algorithms that assume numerical relationships between values. In logistic regression and neural networks (MLP), one-hot encoding ensures that each category is treated independently, allowing the model to learn specific weights for each category. For tree-based models like Random Forest and XGBoost, one-hot encoding helps in creating more meaningful split points. In SVM, it allows for better hyperplane separation in the higher-dimensional space created by the encoding. Normalizing numerical features through techniques like Min-Max scaling is crucial for many machine learning algorithms. This process scales all numerical features to a common range (typically 0 to 1), which is particularly important for gradient-based optimization algorithms used in logistic regression, neural networks, and SVM. Normalization prevents features with larger scales from dominating the learning process, ensuring that all features contribute proportionally to the model's decisions. For distance-based algorithms like SVM, normalized features ensure that distances are computed fairly across all dimensions. In the case of tree-based models like Random Forest and XGBoost, while they are less sensitive to feature scaling, normalized features can still improve the interpretability of feature importances and ensure consistent performance across different scales of input data.

C. *Feature selection*

Feature selection is a critical step in improving the performance and interpretability of machine learning models, particularly for Autism Spectrum Disorder (ASD) classification tasks. For this study, three distinct feature selection methods were applied using Python libraries such as SelectKBest, RFECV, and RandomForestClassifier. These methods were chosen to capture non-linear relationships, optimize feature importance, and enhance diagnostic accuracy.

Mutual Information (MI), implemented via SelectKBest with mutual_info_classif, was used to identify features with strong non-linear dependencies on the target variable. MI is highly effective for behavioral and clinical data, as it quantifies the information gain between features and ASD diagnosis. This method ensures that only the most relevant features are retained, minimizing redundancy and improving model generalization.

Recursive Feature Elimination (RFE), applied using RFECV with a linear Support Vector Machine (SVM), iteratively removes the least predictive features until an optimal subset is achieved. RFE leverages the feature importance scores computed by SVM to refine the feature set. This approach is particularly suited for high-dimensional datasets, ensuring that the selected features contribute maximally to classification accuracy.

Lastly, Conditional Mutual Information Maximization (CMIM) was combined with Random Forest (RandomForestClassifier) to select features based on their importance scores. This method enhances diagnostic accuracy by focusing on features that provide unique information about the target variable while reducing dimensionality. CMIM is especially beneficial for datasets *with mixed feature types, as it effectively handles both continuous and categorical attributes.*

These feature selection techniques collectively address the challenges of ASD data analysis, ensuring robust model performance while maintaining sensitivity and specificity in predictions.

D. *Machine Learning Algoritms*

Headings, XGBoost is a powerful ensemble learning algorithm that builds decision trees sequentially to minimize errors and improve prediction accuracy. It is particularly effective for ASD screening as it handles class imbalance well, captures complex feature interactions, and includes built-in regularization to prevent overfitting. In the code, the xgboost Python library is used, with hyperparameters such as max_depth, learning_rate, and subsample tuned using GridSearchCV to optimize model performance.

SVM is a supervised learning algorithm that finds the optimal hyperplane to separate classes in a high-dimensional space. It is useful for ASD screening because it effectively handles non-linear relationships using kernels like RBF and works well with small or imbalanced datasets. The implementation leverages the scikit-learn library, with hyperparameters such as C (regularization strength), gamma (kernel coefficient), and class_weight tuned via GridSearchCV to improve classification accuracy.

Logistic Regression is a linear model used for binary classification tasks, making it a simple yet interpretable algorithm for ASD screening. It is particularly useful as a baseline model and for understanding the impact of individual features on predictions. The code uses the LogisticRegression class from scikit-learn, and hyperparameters like C (regularization strength) and solver (optimization algorithm) are tuned using grid search to enhance model performance.

Random Forest is an ensemble learning method that combines multiple decision trees to improve prediction accuracy and reduce overfitting. It is highly effective for ASD screening as it handles mixed data types, identifies important features, and performs well even with missing values. The implementation uses the RandomForestClassifier from scikit-learn, with hyperparameters such as n_estimators, max_depth, and min_samples_split tuned through grid search to maximize diagnostic accuracy.

MLP is a type of artificial neural network capable of capturing complex non-linear relationships between features, making it suitable for ASD screening tasks involving intricate behavioral patterns. It adapts well to diverse datasets by learning from data patterns without explicit feature engineering. The code uses the MLPClassifier from scikit-learn, with hyperparameters like hidden_layer_sizes, activation, and alpha (L2 regularization) tuned using grid search to optimize model performance.

V. Evaluation

The evaluation of machine learning models was conducted using a combination of hyperparameter tuning and cross-validation techniques to ensure optimal performance. GridSearchCV was employed to systematically explore different hyperparameter combinations for each model, identifying the best-performing configuration based on sensitivity, a crucial metric for ASD screening. To further validate the model’s robustness, Stratified K-Fold cross-validation with five folds was used, ensuring balanced class representation across splits. Performance metrics such as AUC, sensitivity, specificity, and accuracy were calculated to provide a comprehensive assessment of each model’s effectiveness in distinguishing ASD cases from non-ASD cases while addressing class imbalance challenges.

Sensitivity is the most important metric for disease-related datasets, especially for ASD screening, because it measures the model's ability to correctly identify individuals with the condition. In medical diagnostics, missing a true positive (i.e., failing to detect ASD in someone who has it) can have severe consequences, leading to delayed intervention and missed opportunities for early support. Given that ASD diagnosis can significantly impact an individual's access to specialized care and resources, maximizing sensitivity ensures that fewer cases go undetected, making the screening process more effective and reliable.

Feature Selection	Accuracy (+/-)	AUC (+/-)	Sensitivity (+/-)	Specificity (+/-)	CV Runtime (sec)
All	0.9705 (+/- 0.02)	0.9629 (+/- 0.03)	0.9444 (+/- 0.07)	0.9814 (+/- 0.03)	1.139
Mutual Information (MI)	0.9392 (+/- 0.03)	0.9230 (+/- 0.05)	0.8833 (+/- 0.12)	0.9627 (+/- 0.05)	0.455
Recursive Feature Elimination (RFE)	0.9721 (+/- 0.03)	0.9657 (+/- 0.04)	0.9500 (+/- 0.08)	0.9814 (+/- 0.04)	0.400
CMIM-inspired RF-based Selection	0.9705 (+/- 0.02)	0.9629 (+/- 0.03)	0.9444 (+/- 0.07)	0.9814 (+/- 0.03)	0.502

Table 2. XGBoost evaluation

Feature Selection	Accuracy (+/-)	AUC (+/-)	Sensitivity (+/-)	Specificity (+/-)	CV Runtime (sec)
All	0.9573 (+/- 0.02)	0.9681 (+/- 0.02)	0.9944 (+/- 0.02)	0.9417 (+/- 0.02)	0.472
Mutual Information (MI)	0.9343 (+/- 0.04)	0.9356 (+/- 0.06)	0.9389 (+/- 0.12)	0.9324 (+/- 0.05)	0.172
Recursive Feature Elimination (RFE)	0.9721 (+/- 0.03)	0.9802 (+/- 0.02)	1.0000 (+/- 0.00)	0.9604 (+/- 0.05)	0.186
CMIM-inspired RF-based Selection	0.9622 (+/- 0.02)	0.9700 (+/- 0.02)	0.9889 (+/- 0.03)	0.9510 (+/- 0.02)	0.307

Table 3. SVM- RBF Kerner evaluation

Feature Selection	Accuracy (+/-)	AUC (+/-)	Sensitivity (+/-)	Specificity (+/-)	CV Runtime (sec)
All	0.9819 (+/- 0.01)	0.9856 (+/- 0.01)	0.9944 (+/- 0.02)	0.9767 (+/- 0.02)	0.088
Mutual Information (MI)	0.9392 (+/- 0.02)	0.9472 (+/- 0.02)	0.9667 (+/- 0.04)	0.9278 (+/- 0.03)	0.065
Recursive Feature Elimination (RFE)	0.9918 (+/- 0.03)	0.9942 (+/- 0.02)	1.0000 (+/- 0.00)	0.9884 (+/- 0.04)	0.087
CMIM-inspired RF-based Selection	0.9836 (+/- 0.01)	0.9883 (+/- 0.01)	1.0000 (+/- 0.00)	0.9767 (+/- 0.02)	0.083

Table 4. Logistic Regression evaluation

Feature Selection	Accuracy (+/-)	AUC (+/-)	Sensitivity (+/-)	Specificity (+/-)	CV Runtime (sec)
All	0.9491 (+/- 0.04)	0.9268 (+/- 0.06)	0.8722 (+/- 0.12)	0.9813 (+/- 0.01)	1.204
Mutual Information (MI)	0.9409 (+/- 0.06)	0.9242 (+/- 0.06)	0.8833 (+/- 0.08)	0.9651 (+/- 0.07)	1.360
Recursive Feature Elimination (RFE)	0.9672 (+/- 0.04)	0.9606 (+/- 0.04)	0.9444 (+/- 0.04)	0.9767 (+/- 0.06)	1.109
CMIM-inspired RF-based Selection	0.9491 (+/- 0.02)	0.9268 (+/- 0.03)	0.8722 (+/- 0.06)	0.9813 (+/- 0.01)	1.181

Table 5. Random Forest evaluation

Feature Selection	Accuracy (+/-)	AUC (+/-)	Sensitivity (+/-)	Specificity (+/-)	CV Runtime (sec)
All	0.9556 (+/- 0.02)	0.9540 (+/- 0.04)	0.9500 (+/- 0.08)	0.9580 (+/- 0.01)	5.334
Mutual Information (MI)	0.9655 (+/- 0.03)	0.9594 (+/- 0.03)	0.9444 (+/- 0.04)	0.9744 (+/- 0.03)	4.285
Recursive Feature Elimination (RFE)	0.9918 (+/- 0.01)	0.9926 (+/- 0.02)	0.9944 (+/- 0.02)	0.9907 (+/- 0.02)	4.213
CMIM-inspired RF-based Selection	0.9573 (+/- 0.02)	0.9520 (+/- 0.03)	0.9389 (+/- 0.06)	0.9650 (+/- 0.03)	5.558

Table 6.. MLP evaluation

A. Results

For each algorithm, I ran the model with three different feature selection techniques: Mutual Information (MI), Recursive Feature Elimination (RFE), and CMIM-inspired RF-based selection. The models were evaluated based on their cross-validation (CV) scores, and both the performance metrics (accuracy, AUC, sensitivity, and specificity) and their respective +/- variations were recorded for each feature selection method. These evaluations provide a comprehensive understanding of how each feature selection technique affects model performance.

The performance of the machine learning models varied based on the feature selection method used, with different methods influencing sensitivity, AUC, and accuracy in distinct ways.

In Table 2, the XGBoost model showed that the Recursive Feature Elimination (RFE) technique yielded the best sensitivity score of 0.95, demonstrating the model's high capability in identifying true positive cases. The AUC of 0.97 also supported the model’s strong discriminatory power. In contrast, using all features or the CMIM-inspired RF-based selection method resulted in similar performance, implying that reducing the feature set didn’t drastically affect the model's accuracy or sensitivity. Accuracy remained high across all methods (ranging from 0.939 to 0.973), with minor variations in the +/- scores. The combination of RFE's optimal performance with sensitivity and specificity made it the most efficient feature selection method.

For the SVM (RBF Kernel) in Table 3, RFE once again delivered the highest sensitivity, achieving a perfect score of 1.0, which demonstrates the model’s precision in correctly identifying positive instances. The AUC was also impressive across all feature selection techniques, with RFE achieving the highest AUC of 0.98. Accuracy ranged from 0.9573 to 0.9721, with minor fluctuations depending on the feature selection. This shows that the SVM model, when combined with RFE, provides the most consistent performance in terms of maximizing sensitivity.

Table 4 highlights the strong performance of Logistic Regression, with accuracy consistently above 0.98. RFE emerged as the top performer, providing a sensitivity of 1.0 along with the highest AUC score of 0.9942. This suggests that Logistic Regression, particularly with RFE, is highly effective at identifying ASD cases. The accuracy and AUC metrics were stable across feature selection methods, underscoring RFE as a solid approach for improving model performance.

In Table 5, Random Forest displayed relatively lower sensitivity (ranging from 0.8722 to 0.9444) compared to the other models, although it still showed strong specificity. While RFE did enhance the sensitivity score, Random Forest's overall performance was less competitive in terms of identifying true positive cases compared to models like XGBoost and SVM. However, its accuracy remained fairly stable across feature selection techniques, and the model still performed adequately in terms of specificity.

Table 6 (MLP) exhibited high sensitivity scores, with RFE again achieving the best performance at 0.9944. While AUC and accuracy scores were slightly lower compared to other models, RFE's ability to optimize the MLP model demonstrated its potential in achieving strong classification performance. The longer CV runtime indicates that MLP is computationally heavier, but its sensitivity with RFE showed its potential for classification tasks.

B. Overfitting

To ensure robust performance estimation, Stratified K-Fold cross-validation was employed. Hyperparameter tuning was conducted using GridSearchCV to find the optimal parameters for XGBoost, SVM, Logistic Regression, Random Forest, and MLP. Regularization techniques, including L2 for Logistic Regression, Alpha for MLP, and Class Weights for SVM and Random Forest, were applied to mitigate overfitting. Feature selection was performed using Recursive Feature Elimination (RFE) and Mutual Information to eliminate noise and irrelevant features. Additionally, model complexity was controlled by limiting depth, estimators, and splits in tree-based models like XGBoost and Random Forest. Finally, ensemble learning methods were utilized to combine models, enhancing generalization and overall performance.

VI. Conclusion

Recursive Feature Elimination (RFE) consistently provided the highest sensitivity across multiple machine learning algorithms, including XGBoost, SVM, Logistic

Regression, and MLP. Among all the models, SVM with RFE emerged as the most effective in maximizing sensitivity while maintaining high AUC and accuracy, making it the best-performing model for ASD classification in this evaluation. In conclusion, machine learning has proven to be a valuable tool in enhancing the detection and classification of Autism Spectrum Disorder (ASD), offering a more accessible and efficient approach compared to traditional diagnostic methods. By leveraging algorithms like SVM, CNN, and MLP, along with advanced feature selection techniques such as Recursive Feature Elimination (RFE), this study demonstrates the potential to identify critical behavioral patterns indicative of ASD. The combination of data cleaning, feature engineering, and strategic model evaluation ensures robust and reliable predictions, particularly by focusing on maximizing sensitivity to avoid false negatives. Among the models tested, SVM with RFE stood out as the top performer, providing the highest sensitivity, accuracy, and AUC, making it the ideal choice for early ASD screening. These findings emphasize the importance of optimizing machine learning workflows to improve diagnostic accuracy and accessibility, ultimately paving the way for more effective, scalable, and timely interventions for individuals with ASD.

VII. Future work

To further enhance model performance, hyperparameter tuning can be optimized using methods like randomized search or Bayesian optimization, which allow for a more efficient exploration of the hyperparameter space. Additionally, experimenting with advanced feature selection techniques, such as L1 regularization or mutual information-based approaches, can help improve the model's ability to focus on the most relevant features. Furthermore, implementing ensemble learning methods, like combining Random Forest and XGBoost through stacking or boosting, can reduce bias and variance, leading to more robust and accurate predictions. These strategies aim to refine the model's ability to detect Autism Spectrum Disorder (ASD) with higher precision and reliability.

References

[1] S. Raj and S. Masood, "Analysis and detection of autism spectrum disorder using machine learning techniques," *Procedia Computer Science*, vol. 167, pp. 994-1004, 2020.

[2] F. Thabtah, F. Kamalov, and K. Rajab, "A new computational intelligence approach to detect autistic features for autism screening," *International Journal of Medical Informatics*, vol. 117, pp. 112-124, 2018.

[3] R. K. Reghunathan, P. N. P. Venkidusamy, R. G. Kurup, B. George, and N. Thomas, "Machine learning-based classification of autism spectrum disorder across age groups," *Engineering Proceedings*, vol. 62, no. 1, p. 12, 2024.

[4] M. S. Farooq, R. Tehseen, M. Sabir, and Z. Atal, "Detection of autism spectrum disorder (ASD) in children and adults using machine learning," *Scientific Reports*, vol. 13, no. 1, p. 9605, 2023.

[5] A. R. Subhash and A. K. U. Motagi, "An efficient autism spectrum disorder classification in different age groups using machine learning models," *International Journal of Online & Biomedical Engineering*, vol. 20, no. 9, 2024.

[6] K. Chowdhury and M. A. Iraj, "Predicting autism spectrum disorder using machine learning classifiers," in *2020 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, IEEE, 2020, pp. 324-327.

[7] Tabtah, F. (2017). Autism Spectrum Disorder Screening: Machine Learning Adaptation and DSM-5 Fulfillment. Proceedings of the 1st International Conference on Medical and Health Informatics 2017, pp.1-6. Taichung City, Taiwan, ACM.