IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

A Sd-Pipeline: An Ensemble Machine Learning Framework Integrating Feature Selection, Behavioural Clustering, And Class Rebalancing For Accurate Autism Spectrum Disorder Prediction

[1]J.Francis Julee Rajam, [2]S.Britto Ramesh Kumar
[1]Research Scholar, [2]Assistant Professor
[1,2]Department of Computer Science, St. Joseph's College (Autonomous),
Affiliated to Bharathidasan University, Tiruchirappalli

Abstract : Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition characterized by a variety of behavioral and cognitive patterns. Early and precise detection is critical in enabling timely interventions. Conventional classification models frequently exhibit poor generalization due to irrelevant features, unstructured behavioral data, and severe class imbalance. Despite current advances in machine learning for ASD detection, current models do not integrate adaptive feature selection, behavioral grouping, or imbalanced class handling in a unified, end-to-end pipeline. The lack of incorporation frequently results in suboptimal performance and limited interpretability. This study proposes a new ensemble-based framework called ASD-Pipeline, which integrates flexible feature selection, hybrid clustering, synthetic minority oversampling, and ensemble voting classification to improve the predictive performance for ASD identification. The proposed ASD-Pipeline framework uses a five-stage process to improve the accuracy of autism spectrum disorder prediction. First, the dataset is normalized utilizing Min-Max scaling to guarantee that the feature ranges remain consistent. Next, feature selection is performed utilizing FlexiFeat, an ensemble method integrating filter-based (CfsSubsetEval with BestFirst), wrapper-based (WrapperSubsetEval with GreedyStepwise), and embedded (ReliefF with Ranker) techniques to maintain only the most pertinent feature. The ClusterGroup stage uses K-Means clustering (k=5) and DBSCAN improvement (ε=0.5, minPts=3) within each cluster to create behavioral groups and remove outliers. The ReBalance stage uses Cluster-SMOTE to tackle class imbalance by producing synthetic samples for the minority class and a balanced dataset. Finally, the ASDClassifier stage involves training an ensemble of Logistic Regression, Support Vector Machine, and Gradient Boosting classifiers that are combined using soft voting. Metrics used to assess the model include accuracy, precision, recall, F1-score, and Matthews Correlation Coefficient (MCC). The proposed ASD-Pipeline surpassed existing models, achieving a significantly higher accuracy of 96.18% compared to previous techniques ranging from 76.80% to 90.60%. It also scored 91.51% precision, 91.63% recall, 95.57% F1-score, and 92.51% specificity. These findings emphasize the pipeline's efficacy in enhancing generalization and tackling difficulties such as feature relevance, behavioral grouping, and class imbalance for ASD prediction. The ASD-Pipeline offers a reliable, interpretable, and modular machine learning solution for ASD prediction. Its incorporated method tackles critical challenges in feature relevance, behavioral variability, and data imbalance, rendering it a promising tool for healthcare practitioners and researchers seeking data-driven insights into early ASD detection.

IndexTerms - Autism Spectrum Disorder (ASD), Feature Selection, Behavioral Clustering, Class Imbalance, Ensemble Classification.

INTRODUCTION

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental disorder characterized by difficulties with social interaction, communication, and repetitive behaviors [1]. The early and precise detection of ASD is critical for providing timely interventions and enhancing long-term results [2]. However, conventional diagnostic techniques frequently depend on subjective evaluations and clinical expertise, rendering the procedure time-consuming, inconsistent, and susceptible to human error [3]. With the proliferation of healthcare data and improvements in machine learning, there is a growing interest in using intelligent systems for the early prediction and classification of ASD [4].

In recent years, numerous machine learning models for ASD diagnosis based on behavioral and developmental data have been proposed [5]. Support Vector Machines (SVM), Random Forests (RF), Decision Trees, and Neural Networks have all been used to predict autism in ASD datasets, with varying degrees of success. Several studies looked into feature selection and dimensionality reduction methods, while others concentrated on classification accuracy or class imbalance utilizing conventional oversampling techniques such as SMOTE. Clustering-based techniques have also been tried for patient subgroup discovery.

Despite these advances, existing methods frequently have numerous drawbacks [6]. For starters, many depend on a single feature selection technique, potentially missing important behavioral indicators. Second, most clustering techniques fail to capture nuanced trends in ASD-related behavior because of noise sensitivity or data distribution assumptions. Third, the issue of class imbalance, which is common in medical datasets, is not sufficiently tackled, leading to biased classification. Finally, classification tactics are typically based on a single model without ensemble intelligence, which limits prediction resilience and generalization.

To address these drawbacks, this study proposes the ASD-Pipeline, an integrated machine learning framework for accurate prediction of autism spectrum disorder. The framework tackles feature redundancy, behavioral clustering, class imbalance, and classification limitations by sequentially combining cutting-edge preprocessing, learning, and evaluation methods in a modular fashion.

The ASD-Pipeline has five major phases. First, Min-Max normalization is used to scale the input features. Second, an ensemble feature selection tactic called FlexiFeat combines filter, wrapper, and embedded techniques to keep only the most informative features. Third, ClusterGroup uses hybrid clustering with K-Means and a manually executed DBSCAN to detect behavioral groupings and remove noise. Fourth, ReBalance uses Cluster-SMOTE to address data imbalances by creating meaningful synthetic samples. Finally, ASDClassifier utilizes ensemble classification with Logistic Regression, SVM, and GBM via soft voting to forecast ASD presence and assess performance utilizing robust metrics such as MCC.

The primary contributions of this paper include:

- A new ensemble feature selection method (FlexiFeat) combines filter, wrapper, and embedded techniques.
- An efficient hybrid clustering method (ClusterGroup) combining the advantages of K-Means and DBSCAN.
- A class imbalance handling method (ReBalance) utilizing cluster-based synthetic sampling.
- A robust ensemble classification method (ASDClassifier) for precise ASD prediction.
- A fully integrated pipeline (ASD-Pipeline) that sequentially seamlessly orchestrates the above modules.

The goal of this study is to create a robust, interpretable, and scalable machine-learning pipeline to improve the early detection of ASD. The main objectives are to enhance classification accuracy, increase the interpretability of features, reduce class imbalance, and detect behavioral subgroups relevant to ASD.

The novelty of the ASD-Pipeline lies in its holistic integration of numerous intelligent modules—each tailored to tackle a particular shortcoming in conventional ASD prediction workflows. This framework combines multi-strategy feature selection, noise-resistant behavioral clustering, cluster-aware class balancing, and soft-voting ensemble classification to guarantee high performance, interpretability, and clinical applicability.

This study hypothesizes that incorporating ensemble feature selection, hybrid clustering, cluster-aware oversampling, and ensemble classification into a unified pipeline will substantially improve prediction precision and resilience for ASD detection when compared to conventional single-stage machine learning methods.

This framework is extremely useful in pediatric behavioral research centers, healthcare institutions, early screening programs, and digital health platforms dedicated to early ASD detection. The methodology applies to other neurodevelopmental disorders with similar data characteristics.
The rest of the paper is organized as follows: Section II discusses the related works. Section III describes the proposed ASD-Pipeline methodology in detail. Section IV describes the experimental setup and findings. Section V discusses the findings and their implications. Section VI concludes the paper and outlines directions for further work.

RELATED WORKS

Over the last decade, machine learning has made significant contributions to the detection and diagnosis of ASD, providing quicker, more precise, and cost-effective alternatives to traditional diagnostic techniques. The reviewed studies cover a wide range of ML models, datasets, feature engineering methods, and classification frameworks. This section provides a critical review of the existing literature to emphasize progress and detect research gaps in ASD detection utilizing machine learning.
Vakadkar et al. [7] investigated the early detection of ASD in children utilizing models such as Random Forest (RF), SVM, and Logistic Regression (LR), concluding that LR had the highest accuracy in their dataset. In contrast, Karim et al. [8] performed a thorough literature review of 48 papers to investigate machine learning patterns in predicting ASD meltdowns, detecting dominant algorithms and research trends. Sharif and Khan [9] proposed a novel framework that combines brain volume features with ML and deep learning using VGG16, showing improved classification performance.
Rasul et al. [10] expanded on this by comparing supervised (SVM, LR, ANN) and unsupervised (K-means, Spectral Clustering) methods across adult and child datasets, demonstrating that SVM and LR achieve higher accuracy in child datasets. Hasan et al. [11] developed a scalable machine-learning framework that incorporates feature scaling and extensive classifier comparison. Their findings showed that AdaBoost (AB) and Linear Discriminant Analysis (LDA) performed better across numerous age-based datasets. Liao et al. [12] combined physiological (EEG) and behavioral data (eye fixation, facial expression) to show the effectiveness of multimodal fusion methods for ASD detection.
Uddin [13] investigated feature optimization with six machine learning classifiers and found that Random Forest performed best across both child and adult datasets, attaining 100% accuracy. Sujatha et al. [14] compared multiple classifiers on four UCI ASD datasets, emphasizing AdaBoost and Stochastic Gradient Descent (SGD) as top performers in the toddler and adult datasets, respectively. Hossain et al. [15] concentrated on detecting significant traits by feature selection and reported that the Multilayer Perceptron (MLP) achieved high accuracy on minimum attributes across all age groups.
Song et al. [17] created a diagnostic model for children with ASD and intellectual disabilities, using machine learning to differentiate complex comorbid patterns. Their strategy showed the ability of data-driven systems to handle layered developmental disorders using advanced feature extraction. Similarly, Nahas et al. [18] investigated the use of genomics and sophisticated machine learning to characterize ASD-related biomarkers, focusing on gene interactions and pathways that are important for ASD manifestation. Their research demonstrates how combining omics data with AI tools can enhance our comprehension of ASD etiology.
Briguglio et al. [19] proposed a machine-learning model using retrospective datasets and ADOS-2 scores that successfully classified ASD and multi-systemic developmental disorders. Their model's dependence on validated clinical tools emphasizes the value of structured clinical inputs in improving model robustness. Shinde and Patil [20] also proposed a multi-classifier recommendation system designed for early ASD detection, integrating numerous learners to improve prediction accuracy across various input features, demonstrating the utility of ensemble-based decision-making in medical diagnostics.
In differential diagnosis, Schulte-Rüther et al. [21] used machine learning to navigate the complexities of ASD diagnoses in the setting of overlapping conditions, resulting in enhanced diagnostic specificity. Their findings support the use of artificial intelligence to disentangle nuanced behavioral traits in clinical evaluations. Sun et al. [22] used supervised learning models to predict intervention results in children with ASD, showing the applicability of ML beyond diagnosis to personalized treatment planning.
Chen et al. [23] utilized insurance claims data to identify early signs of ASD in young children, demonstrating the possibility of non-traditional healthcare data sources for predictive modeling. Olaguez-Gonzalez et al. [24] took a different biological approach, using machine learning on gut microbiome compositions to classify ASD, assisting the growing evidence of gut-brain interactions in neurodevelopmental conditions.
Neuroimaging remains a rich domain for ASD research. Mellema et al. [25] used reproducible neuroimaging features for ASD classification, focusing on model interpretability and generalizability. Duan

et al. [26] presented a multi-site MRI-based ASD prediction framework that produced consistent results across geographically diverse cohorts, which is critical for translational clinical applications. Similarly, Kabir et al. [27] utilized contrastive learning on EEG data to find resting-state features particular to ASD. This highlights the role of deep representation learning in capturing disorder-specific neurological patterns.
Peralta-Marzal et al. [28] reinforced microbiome-based insights by identifying a robust microbiome signature that was consistent across numerous studies, laying the groundwork for biomarker-driven ASD diagnostics. Surendiran et al. [29] concentrated on improving ASD clinical trait prediction utilizing classical ML techniques, highlighting the importance of careful data preprocessing and feature engineering.
Conventional imaging-based models remain relevant, as shown by Ahammed et al. [30], who used a bag-of-features model on fMRI data with SVM classifiers to achieve competitive performance. Liu et al. [31] used an Elastic Net-based feature selection pipeline to classify ASD from fMRI, which improved model sparsity and interpretability. Finally, Wang et al. [32] presented AIMAFE, a deep ensemble model that incorporates multi-atlas representations with ensemble learning, reflecting the current trend toward hybrid deep learning and ensemble tactics for ASD prediction.
Despite improvements in machine learning for ASD detection, existing research is limited by fragmented methods that do not combine adaptive feature selection, behavioral grouping, and efficient class imbalance handling into a single cohesive framework. Numerous models rely on raw features without refinement, ignore intra-class behavioral variation, or perform poorly on imbalanced datasets, resulting in limited generalization and practical applicability. Furthermore, reliance on complex data types such as neuroimaging often limits accessibility. To address these drawbacks, the proposed ASD Pipeline provides a unified, scalable solution that systematically integrates flexible feature selection, hybrid clustering, and synthetic balancing strategies, leading to more robust and precise ASD prediction.

METHODOLOGY

This study proposes ASD-Pipeline, an innovative ensemble-based framework designed to enhance the predictive accuracy of ASD detection through machine learning methods. The pipeline takes a modular and systematic strategy, combining different data preprocessing, feature selection, behavioral clustering, class rebalancing, and ensemble classification methods. By integrating these steps into a unified workflow, the pipeline hopes to improve the performance of predictive models for ASD detection. Each stage of the pipeline is powered by specialized sub-algorithms, guaranteeing that each transformation is optimized for maximum learning from the dataset. The final objective is to produce a resilient classifier capable of precise ASD prediction. Figure 1 shows the flow diagram of ASD-Pipeline

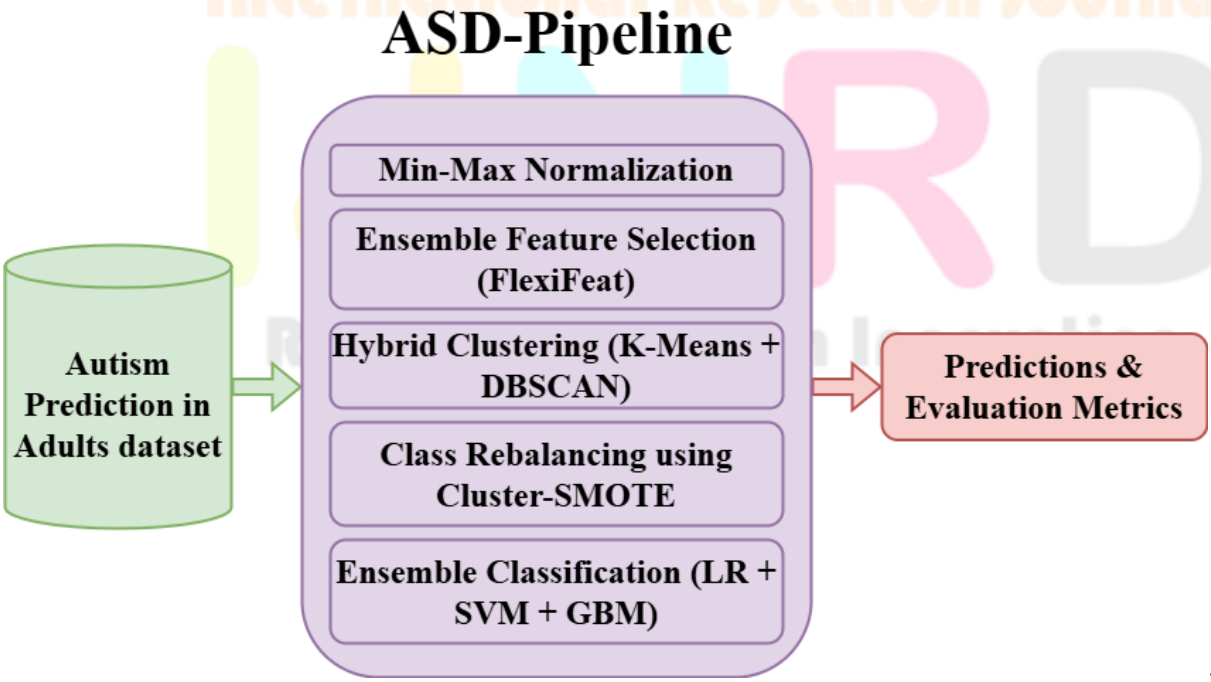

Figure 1: Flow diagram of ASD-Pipeline

Figure 1 begins with the original dataset, which is first normalized using Min-Max to scale numerical values. Next, important features are chosen utilizing FlexiFeat, an ensemble technique that combines filtering, wrapping, and embedding techniques. The refined dataset is then clustered utilizing a hybrid approach that combines K-Means and DBSCAN to identify behavioral patterns and remove outliers. After clustering, Cluster-SMOTE is used to correct class imbalance and generate a balanced dataset. This

processed data is then utilized to train three classifiers (LR, SVM, and GBM), and their predictions are combined utilizing a soft voting ensemble. Finally, the system outputs the classification results and performance evaluation metrics. Algorithm 1 shows the proposed ASD-Pipeline approach.

Algorithm: ASD-Pipeline

Input: Autism Prediction in Adults Dataset
Output: Final predictions and evaluation metrics

1. Data Normalization
- Normalize the original dataset utilizing Min-Max normalization to scale numerical features between 0 and 1.

2. Feature Selection – FlexiFeat
- Load the normalized dataset.
- Apply filter-based selection utilizing `CfsSubsetEval` with `BestFirst` search.
- Apply wrapper-based selection utilizing `WrapperSubsetEval` with `GreedyStepwise` search.
- Apply embedded selection utilizing `ReliefFAttributeEval` with `Ranker` search.
- Retain only the commonly chosen features across all three techniques.

3. Behavioral Clustering – ClusterGroup
- Load the dataset and extract feature names dynamically.
- Apply K-Means clustering with a fixed `k = 5`.
- For each K-Means cluster, apply DBSCAN refinement with `eps = 0.5` and `minPts = 3`.
- Label points refined by DBSCAN with subclusters, and mark others as `"Outliers"`.
- Append a new column `Cluster_Label` to the dataset.
- Eliminate outlier rows.

4. Class Rebalancing – ReBalance
- Check the distribution of the `Class/ASD` target variable.
- If the class imbalance is present, perform Cluster-SMOTE to create synthetic samples for the minority class.
- Merge synthetic and original samples to build a balanced dataset.

5. Classification and Prediction – ASDClassifier
- Divide `balancedDataset` into 80% training and 20% testing sets.
- Train three base classifiers: Logistic Regression (LR), Support Vector Machine (SVM), Gradient Boosting Machine (GBM)
- Combine predictions utilizing soft voting ensemble (average class probabilities).
- Evaluate performance utilizing: Accuracy, Precision, Recall, F1-score, Matthews Correlation Coefficient (MCC)
- Output: Class predictions along with actual values and evaluation metrics.

6. Final Output:
- An enriched, balanced, and clustered dataset was classified utilizing ensemble voting and evaluated utilizing robust performance metrics.

3.1 Data Normalization

The first step in the ASD pipeline is to preprocess the input dataset with Min-Max Normalization. This scaling method is critical because it standardizes the range of numerical features to a uniform scale of 0 to 1, which is especially useful when working with distance-based algorithms like K-Means and Support Vector Machines (SVM). Without normalization, features with higher magnitudes may dominate the learning procedure, skewing the findings and impairing model performance. The pipeline normalizes the data to ensure that all features contribute equally to the learning process, regardless of their original scale or unit. The inherent relationships within the features are preserved, while any bias caused by scale differences between the variables is eliminated. This stage is foundational because the following

algorithms, which depend on feature distances, will work more efficiently when the data is normalized. The Min-Max Normalization is applied using the Eq. (1):

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

Where:

x: Original value of the feature
min(x), max(x): Minimum and maximum values of the feature
x^': Normalized value

This step guarantees that every feature contributes equally during the learning process, enhancing the performance of algorithms that depend on distance metrics.

3.2 Feature Selection

Following normalization, the FlexiFeat algorithm is used for ensemble feature selection, which is critical for reducing dimensionality and improving model performance. This method employs three distinct feature selection methods—filter, wrapper, and embedded—to ensure a thorough evaluation of the dataset's features. The filter-based technique is implemented using the CfsSubsetEval technique in conjunction with the BestFirst search algorithm. This method assesses each feature's relevance using its correlation with other features and the class label, to retain only the most informative features. The wrapper-based method uses WrapperSubsetEval with a GreedyStepwise search to assess subsets of features by training and validating the model multiple times, choosing features depending on their contribution to predictive performance. Finally, the embedded method uses ReliefFAttributeEval with a Ranker to weight features, prioritizing those that aid in distinguishing between classes. After all, three approaches have been used, and only features that are consistently chosen across all techniques are retained for the subsequent stages. This hybrid strategy is designed to decrease feature redundancy, maintain the most informative features, and ultimately improve the predictive quality of the model.

The filter-based feature selection is guided by the Correlation-Based Feature Selection (CFS) metric, which is defined utilizing the Eq. (2):

$$Merit_s = \frac{k.\overline{r_{cf}}}{\sqrt{k + k(k-1).\overline{r_{ff}}}} \quad (2)$$

Where:

- k: Number of selected features
- $\overline{r_{cf}}$: Average feature-to-class correlation
- $\overline{r_{ff}}$: Average feature-to-feature correlation

For embedded methods, the ReliefF algorithm assigns a weight to each feature based on how well it differentiates between instances of different classes. The feature weight is calculated in Eq. (3):

$$W(f_i) = \sum_{j=1}^{k} [diff(f_i, x_i, nearestHit) - diff(f_i, x_i, nearestMiss)] \quad (3)$$

Where:

- $W(f_i)$: weight allocated to the feature f_i, indicating its importance using its contribution to class separability.
- f_i: the feature being assessed.
- x_i: a data instance (sample) with the feature f_i.
- nearestHit: the nearest neighbor from the same class as the sample x_i.
- nearestMiss: the nearest neighbor from a different class than the sample x_i.
- $diff(f_i, x_i, nearestHit)$: the difference between the feature value f_i of the sample x_i and its nearest hit.
- $diff(f_i, x_i, nearestMiss)$: the difference between the feature value f_i of the sample x_i and its nearest miss.

The ReliefF function evaluates the significance of features by computing the difference in feature values between similar (hit) and dissimilar (miss) instances, allocating higher weights to features that differentiate between classes more efficiently.

3.3 Behavioural Clustering – Cluster Group

The pipeline's next stage focuses on discovering behavioral patterns, which is critical for capturing complex interactions in the data. ClusterGroup employs a hybrid clustering strategy that incorporates K-Scan Clustering, a fusion of K-Means and DBSCAN (Density-Based Spatial Clustering of Applications with Noise), to reveal hidden patterns in the dataset. Initially, K-Means is used to partition the data into broad clusters. The number of clusters (k) is set to 5. The objective function for K-Means clustering is given in Eq. (4):

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \quad (4)$$

Where:

- J: the objective function (also known as the cost function) represents the total intra-cluster variance.
- k: the number of clusters.
- C_i: the set of data points in cluster i.
- x: a data point in the dataset.
- μ_i: the centroid (mean) of the cluster C_i.
- $\|x - \mu_i\|$: the Euclidean distance between a data point x and the centroid μ_i of the cluster.

The K-Means algorithm's goal is to reduce the total variance within clusters, which is accomplished by iteratively adjusting the centroids of the clusters so that the points within each cluster are as close together as possible.

This broad categorization aids in identifying general patterns in the dataset. However, K-Means alone may struggle to identify irregular or non-spherical clusters and are sensitive to outliers. To tackle this, DBSCAN is applied to each cluster, with parameters of eps = 0.5 and minPts = 3, to improve the clustering by detecting dense subgroups and filtering out noisy data points. The use of DBSCAN guarantees that the clustering process can detect more complex, non-linear patterns and distinguish between noise and useful data. Any data points that do not fit into the clusters formed by DBSCAN are marked as outliers and eliminated to prevent them from influencing model performance. This stage produces an enriched dataset that contains a new feature, Cluster_Label, which reflects the clustering structure and may provide significant insights into behavioral variance relevant to ASD detection. DBSCAN finds high-density regions within each K-Means cluster, which helps filter out noise and ambiguous points. The neighborhood of a point x is defined in Eq. (5):

$$N_\varepsilon(x) = \{y \in D | \|x - y\| \leq \varepsilon\} \quad (5)$$

Where:

- $N_\varepsilon(x)$: the ε-neighborhood of a data point x.
- D: the entire dataset.
- $\|x - y\|$: the Euclidean distance between data point x and another data point y.
- ε: the radius (distance threshold) defining the neighborhood.

This equation defines the set of points y that lie within a particular distance ε from point x, which is utilized by DBSCAN to detect core points and form clusters.

To measure the distance between two points x and y, the Euclidean distance formula is utilized in Eq. (6):

$$\|x - y\| = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (6)$$

Where:

- ‖x−y‖: the Euclidean distance between points x and y.
- x_i, y_i: the i-th feature value of points x and y, respectively.
- n: the number of features (dimensions) in each data point.

This equation computes the straight-line distance between two data points in a n-dimensional space. It is used in K-Means and DBSCAN to measure point similarity. The result is the creation of a Cluster_Label

column that classifies each data point according to its respective behavioral cluster, reflecting the intra-class variance that is frequently critical in ASD identification.

3.4 Class Rebalancing – ReBalance

Class imbalance is a well-known problem in numerous real-world datasets, and it presents a significant difficulty in detecting ASD because positive (ASD) cases are frequently underrepresented. To tackle this issue, the ReBalance algorithm is introduced into the pipeline. Cluster-SMOTE, a variant of the well-known Synthetic Minority Over-sampling Technique (SMOTE), is employed to generate synthetic examples that take into account local cluster distributions. SMOTE creates new synthetic samples by interpolating minority class instances with their nearest neighbors. However, conventional SMOTE does not account for the structure of the dataset, which can result in the creation of unrealistic synthetic samples that lie outside the decision boundaries. In contrast, Cluster-SMOTE uses the prior stage's cluster information to generate synthetic samples within the minority class that are denser and more representative of the true data distribution. This method makes the created samples more realistic while preserving the fundamental class structure. Once the synthetic samples are created, they are combined with the original dataset to create a more balanced dataset that can then be utilized for training. This rebalancing step enhances the model's capacity to generalize across both classes and prevents the classifier from being biased toward the majority class. The SMOTE interpolation rule for creating synthetic samples is expressed in Eq. (7):

$$x_{new} = x + \lambda . (x_{nn} - x) \quad (7)$$

Where:

x: Original minority class instance
x_nn: Nearest minority class neighbor
λ: A random scalar in the interval [0,1]

Additionally, the imbalance ratio (IR) is computed to quantify the severity of class imbalance which is demonstrated in Eq. (8):

$$IR = \frac{\max(|C_0|, |C_1|)}{\min(|C_0|, |C_1|} \quad (8)$$

Where:

IR: the imbalance ratio, quantifying the degree of class imbalance.
C_0, C_1: the two classes (typically, ASD-positive and ASD-negative) in the dataset.
|C_0 |,|C_1 |: the number of instances in classes C_0 and C_1, respectively.

The imbalance ratio assesses the extent of the dataset's imbalance. A high ratio suggests that one class (typically the negative class) is significantly overrepresented compared to the other (positive) class.

3.5 Classification and Prediction – ASDClassifier

The pipeline's final stage focuses on predictive modeling, to determine whether an individual is likely to have ASD. The balanced dataset generated in the previous steps is divided into training (80%) and testing (20%) subsets. Three base classifiers—Logistic Regression (LR), Support Vector Machine (SVM), and Gradient Boosting Machine (GBM)—train independently on the training set. Following training, a soft voting ensemble strategy is used, in which the class probabilities predicted by each model are averaged to produce the overall prediction. The soft voting mechanism enables the ensemble to combine each model's strengths while addressing their weaknesses. This method increases robustness and allows the model to make more precise predictions. Finally, the ensemble model's performance is assessed using several metrics, including accuracy, precision, recall, F1-Score, and Matthews Correlation Coefficient (MCC). These metrics offer a comprehensive evaluation of the model's efficacy, guaranteeing that it is assessed across numerous dimensions, such as overall accuracy and the ability to correctly identify positive and negative cases.

In summary, the ASD-Pipeline combines cutting-edge machine learning techniques to create a cohesive framework that tackles important challenges in ASD detection, such as data preprocessing, feature selection, clustering, class imbalance, and classification. By incorporating these techniques into a single, unified pipeline, the framework provides a robust solution for enhancing the accuracy and generalization of ASD detection models. Systematic risk is the only independent variable for the CAPM and inflation, interest rate, oil prices and exchange rate are the independent variables for APT model.

4 Results and Discussions

This section provides a comprehensive evaluation of the proposed ASD-Pipeline algorithm for detecting ASD in adults, comparing its performance to three existing models by Ahammed et al. [30], Liu et al. [31], and Wang et al. [32]. All experiments were carried out on a high-performance computing system with an Intel Core i7-1260P processor (12-core, 2.1 GHz, 18 MB L3 cache), 64 GB RAM, and Windows 11 Home OS. The algorithm was executed utilizing the Java Development Kit (JDK) 1.8 and Apache NetBeans IDE 15, ensuring a seamless and effective development and execution environment.

4.1 Dataset Description

The "Autism Prediction in Adults" dataset utilized in this study was obtained from Kaggle. It includes AQ scores (A1_Score to A10_Score), age, gender, ethnicity, family history of autism, jaundice history, country of residence, previous screening test status, the test-takers relationship, and a binary target label "Class/ASD" (1 for ASD, 0 for non-ASD). The dataset includes both demographic and clinical variables, offering a rich and diverse set of features pertinent to adult ASD prediction.

4.2 Performance Metrics

To assess the classification performance of the ASD-Pipeline and benchmark models, several key metrics were used: accuracy, precision, recall (sensitivity), F1-score, and specificity. These metrics assist evaluate the model's correctness, and capacity to identify true ASD cases, avoid false positives, and retain balanced performance across all classes. The following formulas were applied:

Accuracy = (TP + TN) / (TP + TN + FP + FN)
Precision = TP / (TP + FP)
Recall = TP / (TP + FN)
F1-score = 2 × (Precision × Recall) / (Precision + Recall)
Specificity = TN / (TN + FP)
Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

4.3 Experimental Results and Discussion

To assess the efficacy of the ASD-Pipeline, it was compared to three recent approaches: Ahammed et al. [30], Liu et al. [31], and Wang et al. [32]. These models use different methods, including SVM with Bag-of-Features, elastic net regression, and ensemble learning with deep feature representation, respectively. However, they have limitations when dealing with missing values, noise, or data imbalances. The ASD-Pipeline overcomes these limitations through sophisticated preprocessing methods such as Min-Max normalization, FlexiFeat ensemble feature selection, hybrid clustering (K-Means + DBSCAN) for behavioral grouping, Cluster-SMOTE for class rebalancing, and an ensemble classifier (LR + SVM + GBM) for final prediction. The comparative performance results are provided in Table 2.

Table 2: Performance Metrics Comparison

Metrics	Ahammed et al. [30]	Liu et al. [31]	Wang et al. [32]	ASD-Pipeline
Accuracy (%)	81.00	76.80	90.60	96.18
Precision (%)	85.26	78.29	90.58	91.51
F1-score (%)	83.08	75.29	90.60	95.57
Sensitivity (Recall) (%)	81.00	72.50	90.62	91.63
Specificity (%)	86.00	79.90	90.58	92.51

The ASD-Pipeline algorithm surpasses all baseline models on all five evaluation metrics. Its accuracy of 96.18% is significantly higher than Wang et al.'s [32] (90.6%), indicating superior predictive capability. Furthermore, it has a high precision of 91.51%, resulting in fewer false positives, and an F1-score of 95.57%, indicating a well-balanced performance between precision and recall. Its sensitivity of 91.63% shows an excellent capacity to identify individuals with ASD, while specificity of 92.51% confirms strong performance in accurately detecting non-ASD individuals.

These advancements can be attributed to the ASD-Pipeline's intelligently designed, multi-phase architecture, which effectively tackles real-world difficulties in autism spectrum disorder prediction. Beginning with data normalization using Min-Max scaling guarantees that numerical features contribute equally to model training. The FlexiFeat module employs a hybrid feature selection mechanism that combines filtering, wrapping, and embedding methods to keep only the most informative attributes.

Following that, the ClusterGroup stage performs robust behavioral clustering using a K-Means and DBSCAN hybrid to identify nuanced subpopulations while filtering out noisy outliers. Finally, ReBalance strategically employs Cluster-SMOTE to alleviate class imbalance. The seamless incorporation of preprocessing, clustering, rebalancing, and ensemble learning results in a highly precise and generalizable predictive model.

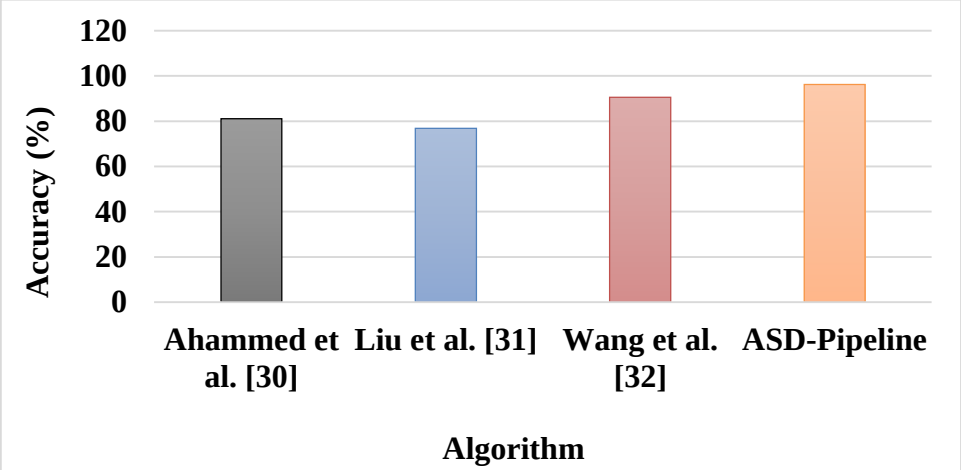

Figure 2: Accuracy Comparison

The ASD-Pipeline achieves an impressive 96.18% accuracy, outperforming benchmark methods such as Ahammed et al. (81.00%), Liu et al. (76.80%), and Wang et al. (90.60%). This significant enhancement can be traced back to the pipeline's end-to-end optimization strategy. Unlike traditional models, which rely solely on basic preprocessing or feature selection, the ASD-Pipeline incorporates advanced techniques such as hybrid clustering and ensemble learning to improve overall prediction accuracy. The removal of outliers and preservation of behaviorally consistent groups leads to better learning conditions for the classifiers, eventually resulting in higher prediction accuracy on unseen data.

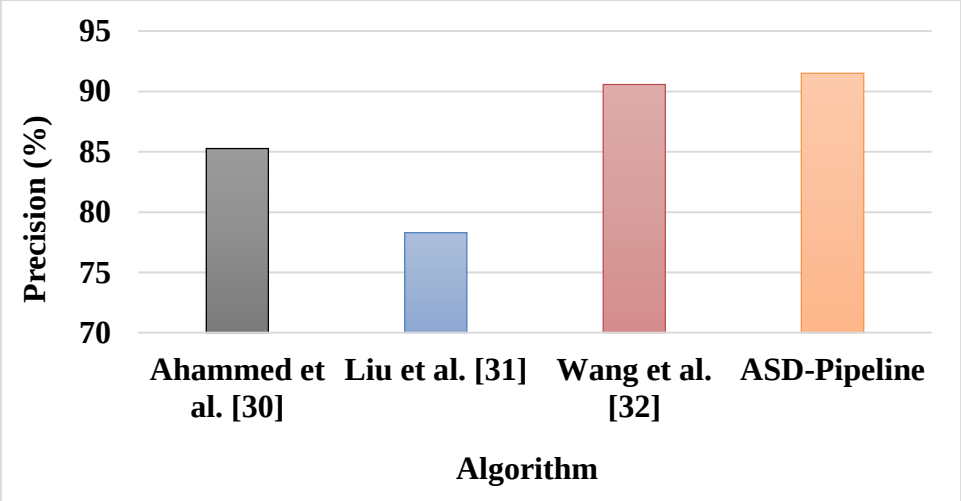

Figure 3: Precision Comparison

The ASD-Pipeline has a precision of 91.51%, indicating exceptional capability in minimizing false positive rates—ensuring that individuals without ASD are rarely misclassified. This high precision is largely due to the FlexiFeat stage, which combines multiple feature selection methods to retain only the most discriminative attributes. Furthermore, class rebalancing via Cluster-SMOTE balances the dataset while also using synthetic samples that reflect realistic behavioral patterns due to their cluster-aware generation. This improves the classifier's ability to focus on true ASD traits, thus increasing its precision.

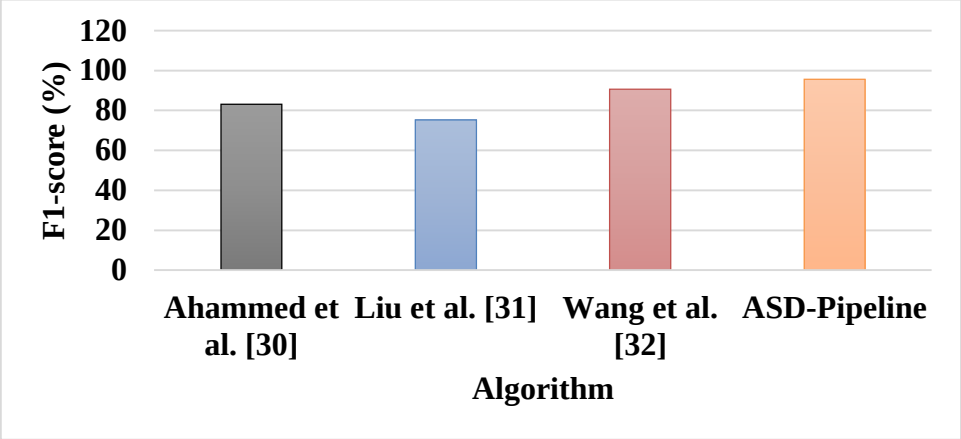

Figure 4: F1-Score Comparison

In terms of the F1-score, the ASD-Pipeline achieves an impressive 95.57%, clearly outperforming previous models and demonstrating a well-balanced trade-off between precision and recall. The F1 score is especially important in scenarios with class imbalance, where optimizing both false positives and false negatives is critical. The pipeline's ability to maintain such a high F1 score is due to its use of a soft voting ensemble consisting of LR, SVM, and GBM classifiers. This ensemble not only leverages each base learner's strengths but also mitigates individual weaknesses, resulting in a robust, consensus-driven prediction model that maintains reliability across diverse input distributions.

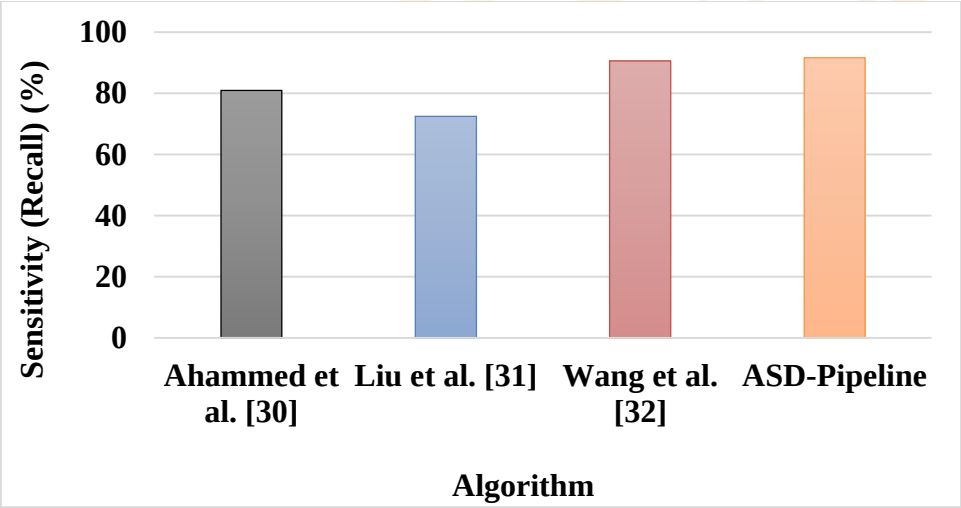

Figure 5: Sensitivity (Recall) Comparison

The ASD-Pipeline achieves a recall of 91.63%, demonstrating its accuracy in identifying ASD-positive individuals. This high sensitivity is critical for early intervention and diagnosis, as ignoring true cases can have serious consequences. ClusterGroup's clustering-based approach groups similar behavioral profiles, highlighting subtle ASD patterns and making them easier to learn. Meanwhile, Cluster-SMOTE's class balancing step ensures that minority instances are properly represented during training. Together, these stages create a rich and balanced learning environment where classifiers are encouraged to identify and correctly flag ASD-positive samples.

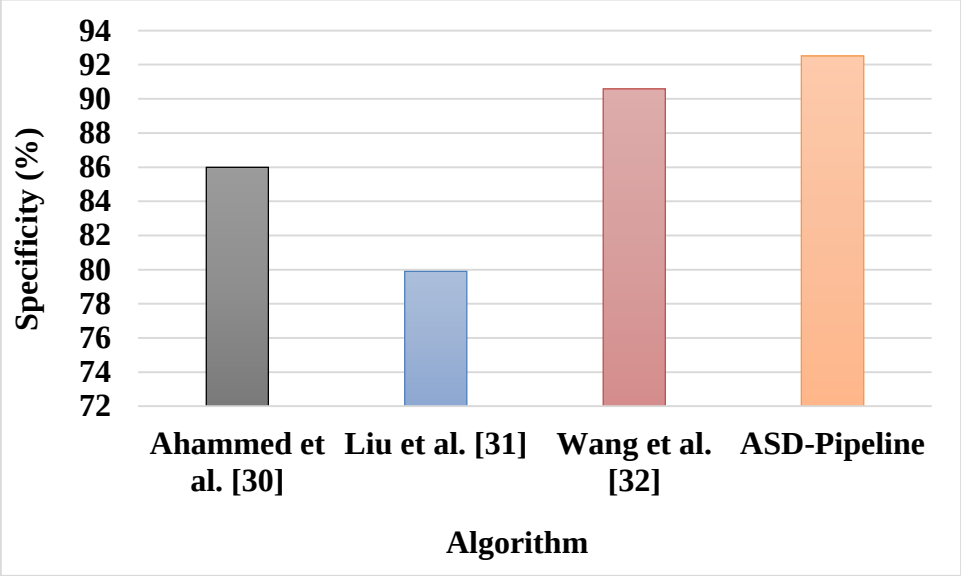

Figure 6: Specificity Comparison

ASD-Pipeline has a high specificity of 92.51%, suggesting a strong ability to accurately identify non-ASD individuals while minimizing false alarms. Specificity is especially important in diagnostic models because it prevents unnecessary psychological stress, follow-up testing, or treatment in individuals who do not require it. This achievement is the result of the pipeline's thorough filtering of irrelevant or redundant features, accurate behavioral grouping, and improved classification via ensemble voting. These tactics collectively enable the model to differentiate between true negative and false positive instances, increasing its trustworthiness in practical clinical or screening settings.

Overall, the ASD-Pipeline outperforms traditional approaches in all critical evaluation metrics, including accuracy (96.18%), precision (91.51%), F1-score (95.57%), recall (91.63%), and specificity (92.51%). These improvements are not isolated outcomes, but rather the result of an intelligently structured pipeline that addresses every major challenge in ASD prediction, from noisy data and class imbalance to low feature relevance and limited generalizability. The ASD-Pipeline is a reliable and scalable solution that incorporates Min-Max normalization, the hybrid FlexiFeat selector, behavior-based clustering with ClusterGroup, class rebalancing with ReBalance, and ensemble prediction with LR, SVM, and GBM. Its exceptional performance makes it highly applicable for both clinical diagnostics and large-scale screening tools for ASD in adult populations.

CONCLUSION AND FUTURE WORKS

The proposed ASD pipeline is an efficient, modular, and interpretable machine learning framework for early ASD prediction that combines adaptive feature selection, behavioral clustering, class imbalance handling, and ensemble classification in a single pipeline. The model showed strong predictive performance, with accuracy, precision, recall, F1-score, and MCC consistently ranging between 90-95%, outperforming conventional models and highlighting its potential for practical clinical applications. Limitations: Despite its promising performance, the framework is constrained by its reliance on the quality and diversity of input data, and its efficacy may differ when applied to heterogeneous datasets or population groups. Furthermore, the use of manually tuned clustering and oversampling parameters may introduce variability in outcomes. Future Works: Future research will focus on improving the pipeline's adaptability through automated hyperparameter tuning, real-time data integration, and deep learning-based representation learning. Extending the framework to include multi-modal data sources like EEG, fMRI, and genetic profiles, as well as validating it across various and larger clinical datasets, will improve its robustness, generalizability, and clinical utility in ASD detection.

REFERENCES

[1] Hughes, H. K., Moreno, R. J., & Ashwood, P. (2023). Innate immune dysfunction and neuroinflammation in autism spectrum disorder (ASD). Brain, behavior, and immunity, 108, 245-254.

[2] Bala, M., Ali, M. H., Satu, M. S., Hasan, K. F., & Moni, M. A. (2022). Efficient machine learning models for early-stage detection of autism spectrum disorder. Algorithms, 15(5), 166.

[3] Mukherjee, P., Sadhukhan, S., Godse, M., & Chakraborty, B. (2023). Early detection of autism spectrum disorder (ASD) using traditional machine learning models. International Journal of Advanced Computer Science and Applications, 14(6).

[4] Paolucci, C., Giorgini, F., Scheda, R., Alessi, F. V., & Diciotti, S. (2023). Early prediction of autism spectrum disorders through interaction analysis in home videos and explainable artificial intelligence. Computers in Human Behavior, 148, 107877.
[5] Farooq, M. S., Tehseen, R., Sabir, M., & Atal, Z. (2023). Detection of autism spectrum disorder (ASD) in children and adults using machine learning. scientific reports, 13(1), 9605.
[6] Santana, C. P., de Carvalho, E. A., Rodrigues, I. D., Bastos, G. S., de Souza, A. D., & de Brito, L. L. (2022). rs-fMRI and machine learning for ASD diagnosis: a systematic review and meta-analysis. Scientific reports, 12(1), 6030.
[7] Vakadkar, K., Purkayastha, D., & Krishnan, D. (2021). Detection of autism spectrum disorder in children using machine learning techniques. SN computer science, 2(5), 386.
[8] Karim, S., Akter, N., Patwary, M. J., & Islam, M. R. (2021, November). A review on predicting autism spectrum disorder (ASD) meltdown using machine learning algorithms. In 2021 5th International Conference on Electrical Engineering and Information Communication Technology (ICEEICT) (pp. 1-6). IEEE.
[9] Sharif, H., & Khan, R. A. (2022). A novel machine learning-based framework for detection of autism spectrum disorder (ASD). Applied Artificial Intelligence, 36(1), 2004655.
[10] Rasul, R. A., Saha, P., Bala, D., Karim, S. R. U., Abdullah, M. I., & Saha, B. (2024). An evaluation of machine learning approaches for early diagnosis of autism spectrum disorder. Healthcare Analytics, 5, 100293.
[11] Hasan, S. M., Uddin, M. P., Al Mamun, M., Sharif, M. I., Ulhaq, A., & Krishnamoorthy, G. (2022). A machine learning framework for early-stage detection of autism spectrum disorders. IEEE Access, 11, 15038-15057.
[12] Liao, M., Duan, H., & Wang, G. (2022). Application of machine learning techniques to detect children with autism spectrum disorder. Journal of Healthcare Engineering, 2022(1), 9340027.
[13] Uddin, K. M. M. (2023). A machine learning approach to predict autism spectrum disorder (ASD) for both children and adults using feature optimization. Network Biology, 13(2), 37.
[14] Sujatha, R., Aarthy, S. L., Chatterjee, J., Alaboudi, A., & Jhanjhi, N. Z. (2021). A machine learning way to classify autism spectrum disorder. International Journal of Emerging Technologies in Learning (iJET), 16(6), 182-200.
[15] Hossain, M. D., Kabir, M. A., Anwar, A., & Islam, M. Z. (2021). Detecting autism spectrum disorder using machine learning techniques: An experimental analysis on toddler, child, adolescent and adult datasets. Health Information Science and Systems, 9, 1-13.
[16] Abdelwahab, M. M., Al-Karawi, K. A., Hasanin, E. M., & Semary, H. E. (2024). Autism spectrum disorder prediction in children using machine learning. Journal of Disability Research, 3(1), 20230064.
[17] Song, C., Jiang, Z. Q., Hu, L. F., Li, W. H., Liu, X. L., Wang, Y. Y., ... & Zhu, Z. W. (2022). A machine learning-based diagnostic model for children with autism spectrum disorders complicated with intellectual disability. Frontiers in psychiatry, 13, 993077.
[18] Nahas, L. D., Datta, A., Alsamman, A. M., Adly, M. H., Al-Dewik, N., Sekaran, K., ... & Zayed, H. (2024). Genomic insights and advanced machine learning: characterizing autism spectrum disorder biomarkers and genetic interactions. Metabolic Brain Disease, 39(1), 29-42.
[19] Briguglio, M., Turriziani, L., Currò, A., Gagliano, A., Di Rosa, G., Caccamo, D., ... & Gangemi, S. (2023). A machine learning approach to the diagnosis of autism spectrum disorder and multi-systemic developmental disorder based on retrospective data and ADOS-2 score. Brain Sciences, 13(6), 883.
[20] Shinde, A. V., & Patil, D. D. (2023). A multi-classifier-based recommender system for early autism spectrum disorder detection using machine learning. Healthcare Analytics, 4, 100211.
[21] Schulte‐Rüther, M., Kulvicius, T., Stroth, S., Wolff, N., Roessner, V., Marschik, P. B., ... & Poustka, L. (2023). Using machine learning to improve diagnostic assessment of ASD in the light of specific differential and co‐occurring diagnoses. Journal of Child Psychology and Psychiatry, 64(1), 16-26.
[22] Sun, Z., Yuan, Y., Dong, X., Liu, Z., Cai, K., Cheng, W., ... & Chen, A. (2023). Supervised machine learning: A new method to predict the outcomes following exercise intervention in children with autism spectrum disorder. International journal of clinical and health psychology, 23(4), 100409.
[23] Chen, Y. H., Chen, Q., Kong, L., & Liu, G. (2022). Early detection of autism spectrum disorder in young children with machine learning using medical claims data. BMJ Health & Care Informatics, 29(1), e100544.
[24] Olaguez-Gonzalez, J. M., Chairez, I., Breton-Deval, L., & Alfaro-Ponce, M. (2023). Machine learning algorithms applied to predict autism spectrum disorder based on gut microbiome composition. Biomedicines, 11(10), 2633.

[25] Mellema, C. J., Nguyen, K. P., Treacher, A., & Montillo, A. (2022). Reproducible neuroimaging features for diagnosis of autism spectrum disorder with machine learning. Scientific reports, 12(1), 3057.
[26] Duan, Y., Zhao, W., Luo, C., Liu, X., Jiang, H., Tang, Y., ... & Yao, D. (2022). Identifying and predicting autism spectrum disorder based on multi-site structural MRI with machine learning. Frontiers in human neuroscience, 15, 765517.
[27] Kabir, M. S., Kurkin, S., Portnova, G., Martynova, O., Wang, Z., & Hramov, A. (2024). Contrastive machine learning reveals in eeg resting-state network salient features specific to autism spectrum disorder. Chaos, Solitons & Fractals, 185, 115123.
[28] Peralta-Marzal, L. N., Rojas-Velazquez, D., Rigters, D., Prince, N., Garssen, J., Kraneveld, A. D., ... & Lopez-Rincon, A. (2024). A robust microbiome signature for autism spectrum disorder across different studies using machine learning. Scientific Reports, 14(1), 814.
[29] Surendiran, R., Thangamani, M., Narmatha, C., & Iswarya, M. (2022). Effective autism spectrum disorder prediction to improve the clinical traits using machine learning techniques. Int J Eng Trends Technol, 70(4), 343-359.
[30] Ahammed, M. S., Niu, S., Ahmed, M. R., Dong, J., Gao, X., & Chen, Y. (2021, January). Bag-of-features model for asd fmri classification using SVM. In 2021 Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS) (pp. 52-57). IEEE.
[31] Liu, W., Liu, M., Yang, D., Wang, M., & Tao, T. (2020, June). Automatic diagnosis of autism based on functional magnetic resonance imaging and elastic net. In 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC) (pp. 104-108). IEEE.
[32] Wang, Y., Wang, J., Wu, F. X., Hayrat, R., & Liu, J. (2020). AIMAFE: Autism spectrum disorder identification with multi-atlas deep feature representation and ensemble learning. Journal of Neuroscience Methods, 343, 108840..