# Overcoming Data Imbalance and Feature Bias in Autism Classification with Overall Local Accuracy

Thanh Nguyen Van Quoc [1], Hao Nguyen Thi Bich [2], Nhut Nguyen Minh[3], Thuan Nguyen Dinh[4]
*Faculty of Information Systems*
*University of Information Technology - Vietnam National University*
Ho Chi Minh City, Vietnam
{21521447@gm.uit.edu.vn, 21522049@gm.uit.edu.vn, nhutnm.17@grad.uit.edu.vn, thuannd@uit.edu.vn}

**Abstract**

This study addresses the challenges of data imbalance and feature bias in predicting autism spectrum disorder (ASD) using pediatric psychological records. We evaluate four classification models: Decision Tree, XGBoost, CatBoost, and Overall Local Accuracy (OLA) on real-world clinical data. Our findings highlight OLA's superior performance, achieved by mitigating feature dominance and effectively handling imbalanced datasets. By reducing reliance on specific data columns, OLA provides a balanced and robust approach to ASD classification, offering promising implications for early diagnosis in pediatric psychology.

## I. Introduction

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition marked by difficulties in social interaction, communication, and repetitive behaviors. Early and accurate diagnosis of ASD in children is vital for timely intervention, which can greatly improve long-term outcomes. However, applying clinical psychological records to predict ASD faces challenges, including data imbalance, feature bias, and limited integration of information technology (IT) with medical and psychological fields, which complicates obtaining reliable datasets. Effective data preprocessing-such as collecting, selecting, labeling, and refining data with guidance from psychology experts-is crucial to build a meaningful dataset for model training, ensuring it accurately reflects clinical realities.

Machine learning has shown promise in predicting Autism Spectrum Disorder (ASD), with models like Decision Trees, XGBoost, and CatBoost commonly used. However, these models often produce overly optimistic results that fail in real-world applications due to uneven training data and over-reliance on specific data points. For example, a single behavioral indicator may dominate the model's predictions, limiting its ability to handle diverse cases, especially in pediatric settings where data varies widely. This highlights the need for methods that address data imbalance and bias to ensure reliable and fair ASD diagnosis.

To address these challenges, this study introduces Overall Local Accuracy (OLA), a new approach that balances the influence of data points to reduce bias while effectively handling uneven datasets using SMOTE-IPF, a technique for data augmentation. Unlike traditional models, OLA automatically selects the most suitable classifier for each input sample based on its local context, with tailored code-level adjustments to prevent over-reliance on dominant features. This leads to fairer and more accurate ASD predictions. Our research compares OLA with Decision Trees, XGBoost, and CatBoost using real-world pediatric psychological records to evaluate their performance. The goal is to demonstrate that OLA's balanced approach improves diagnostic accuracy, offering a practical tool for early ASD detection in clinical settings.

## II. Related Works

Harshita Chandrappa, Swaathi BR, Prof. Swimpy Pahuja. [**?**] proposed a hybrid machine learning framework for predicting Autism Spectrum Disorder (ASD) across all age groups. The system incorporates multiple ML classifiers such as Decision Tree, Naive Bayes, Random Forest, SVM, and Logistic Regression into a web-based diagnostic application. Through preprocessing and classification on a structured ASD dataset, their approach demonstrated high accuracy in predicting ASD traits. Among the models evaluated, Support Vector Machine and Random Forest yielded the best results in terms of accuracy and robustness. The authors highlighted the importance of early screening and presented a chatbot-based interface to assist users in understanding ASD symptoms and diagnosis.

José A. Sáez, Julián Luengo, Jerzy Stefanowski, Francisco Herrera [**?**] presented a novel approach for text classification that combines SMOTE-based oversampling with an IDF (Inverse Document Frequency) weighting mechanism to address class imbalance in textual datasets. The authors utilized a Deep Neural Network (DNN) as the main classifier and applied their method to benchmark datasets such as 20 Newsgroups. Their technique, referred to as SMOTE-IDF, effectively balances minority class samples while preserving the importance of rare but informative terms. Experimental results showed that the SMOTE-IDF+DNN model outperformed traditional oversampling techniques and baseline classifiers in terms of accuracy, precision, recall, and F1-score. The study emphasizes the importance of hybrid sampling and feature weighting in improving classification performance on imbalanced textual data.

Hanen Karamti, Raed Alharthi, Amira Al Anizi, Reemah M. Alhebshi, Ala' Abdulmajid Eshmawi, Shtwai Alsubai, Muhammad Umer. [**?**] introduced an enhanced oversampling technique named SMOTE-KNN, which improves upon the original SMOTE algorithm by integrating a K-Nearest Neighbors (KNN)-based adaptive mechanism. The proposed method dynamically selects appropriate neighbor samples for synthetic instance generation, aiming to reduce noise and overlap between classes in imbalanced datasets. The authors validated SMOTE-KNN on multiple benchmark datasets and compared its performance with traditional SMOTE, Borderline-SMOTE, and ADASYN. Experimental results demonstrated that SMOTE-KNN significantly enhances classification metrics such as precision, recall, and F1-score across various classifiers. The study underlines the importance of local data structure in oversampling and offers a more robust strategy for handling imbalance in binary classification problems.

B. D. Hung, V. V. Thoa, and X. T. Dang. [**?**] introduced KSI, a novel hybrid method that combines clustering with SMOTE and iterative noise filtering (IPF) to enhance classification performance on imbalanced datasets. While traditional SMOTE and SMOTE-IPF approaches are commonly used for over-sampling minority classes, they often generate noisy or overlapping samples, reducing classification accuracy. KSI addresses this by using k-means clustering to identify local distributions where minority instances may dominate or be sparse. Synthetic samples are only generated in regions where local minority density is low, followed by IPF to remove noise. Experiments on multiple UCI datasets demonstrated that KSI outperforms baseline methods (SMOTE, IPF, SMOTE-IPF) in terms of G-mean and AUC, particularly on noisy and borderline samples.

Youngkyu Hong, Eunho Yang. [?] introduced a dual-branch learning framework that tackles bias in image classification tasks by separating biased and unbiased representations. Their method employs two key components: Bias-Contrastive Learning (BCL), which guides the biased branch to focus on spurious correlations, and Bias-Balanced Learning (BBL), which steers the unbiased branch to rely on robust features. By minimizing cross-entropy loss from the biased branch and maximizing agreement with ground truth labels in the unbiased branch, the framework effectively reduces model reliance on dataset-specific biases. Evaluations on benchmark datasets including Biased-MNIST and ImageNet-9 demonstrate that this method significantly outperforms conventional debiasing techniques, especially in scenarios where training labels are confounded by dominant visual cues.

Y. Zeng, J. Liu, H. Lam, and H. Namkoong. [?] investigated the use of Large Language Model (LLM) embeddings to enhance the robustness of tabular classifiers under distributional shifts. Their method integrates LLM-derived feature representations with original tabular features and applies group-wise robust optimization at test time to adapt to both covariate (X) and label (Y) shifts. The study shows that this approach improves the generalization performance of traditional classifiers such as XGBoost and logistic regression across diverse datasets. The results emphasize the potential of LLM embeddings as transferable features for tabular classification tasks under domain shift scenarios.

## III. Methodology

### *A. Dataset Description*

To comprehensively evaluate the proposed approach, two datasets were utilized:

*1) Public Dataset:* The public dataset was sourced from Kaggle [?], consisting of responses to 10 behavioral screening questions designed to detect ASD symptoms in children. The dataset is balanced, with an approximately equal distribution between ASD and non-ASD labels, each record includes binary responses to 10 questions, personal information and other relevant details. These 10 questions in the dataset are labeled to correspond with evaluation topic, which also support the labeling of the private dataset under the guidance and verification of psychological experts.

- **Total samples:** 292 records
- **Class distribution:** 141/292 records ASD about 48,3%
- **Number and types of features:** 20 columns included raw columns and label columns

*2) Private Dataset:* The private dataset was collected from the medical records of the Psychology Clinic, featuring a more complex structure and class imbalance due to medical records in favor of disease data. It contains over 600 samples, with richer information such as personal information, acknowledging behavior from relatives of children and evaluating from a specialist after a direct examination at the clinic. A major challenge in this dataset is the excessive dominance of a single feature, which represents the standard assessment by doctors. This feature exhibits a bias tendency, affecting the final clinical outcomes and resulting in model decisions with high experimental accuracy but prone to misapplication in practical settings.

a, Raw data: This study utilizes a dataset comprising over **1,200 pediatric psychological records** collected from a child psychology clinic, covering various conditions such as Autism Spectrum Disorder (ASD), intellectual developmental disorder, language disorder, and attention deficit hyperactivity disorder. The initial step involved standardizing the data by converting PDF records into structured Excel tables. The records were then categorized by condition, resulting in the selection of over 800 cases related to ASD and non-clinical cases.

b, Filter data: Records with missing or substandard data were filtered out, yielding a refined dataset of **about 600 records**. Then, data labeling was performed to assign features known to influence ASD prediction, including eye contact [?], pointing gestures [?], response to name [?], joint attention [?], imitation [?], and symbolic play [?]... These behavioral indicators were selected based on their strong association with early autism markers and clinical diagnostic standards. To ensure privacy, personal information was anonymized through encoding before further preprocessing.

c, Dataset prepared:

- **Number of samples:** 594 records
- **Class distribution:** 550/594 records ASD-Tracking and ASD about 93%
- **Number and types of features:** 36 columns included raw columns and label columns

### *B. Data Preprocessing*

**a, Overview Data Preprocessing**

Bảng I: Data Preprocessing Steps for private dataset

| **Step** | **Description** |
|---|---|
| Anonymization | All personal identifiers were encoded or removed to ensure participant privacy before model training. |
| Redundant Column Removal | Eliminated non-informative columns such as administrative metadata to reduce noise and dimensionality. |
| Missing Value Imputation | Applied median imputation for numeric features and mode imputation for categorical features, guided by domain-specific psychology rules. |
| Feature Encoding | Encoded target variable (ASD diagnosis) as binary (0 = non-ASD, 1 = ASD). Behavioral features were encoded as binary (0/1) or ordinal values (0, 0.5, 1). |
| **SMOTE-IPF Balancing** | Used SMOTE to synthesize minority-class samples and IPF to eliminate noisy synthetic instances, improving class balance and data quality. |

**b, Method Imbalanced Data Handling**

The dataset exhibited notable class imbalance, particularly when considering ASD as the minority class in binary classification settings (ASD vs. non-ASD). To address this, two advanced resampling techniques were explored: **SMOTE–IPF** [**?**] and **SMOTE-KNN** [**?**].

**SMOTE-IPF** (Synthetic Minority Oversampling Technique with Iterative Partitioning Filter) was ultimately selected due to its superior performance in both class balancing and noise reduction. This method works by generating synthetic ASD samples via interpolation of feature vectors. It then applies a K-Nearest Neighbors (KNN) algorithm to evaluate the local class distribution of each sample. Samples are flagged as *positive* if the majority of their neighbors belong to the same class, and *negative* otherwise. Unsafe non-ASD samples flagged as negative were relabeled as ASD to improve class balance. The process was repeated iteratively to refine labeling and filter out noisy data points.

In contrast, **SMOTE-KNN** focuses on handling missing values before oversampling. It first imputes missing features using KNN, then applies SMOTE to synthetically augment the minority class, often combined with ensemble learning. While this method improves feature completeness, it lacks mechanisms to eliminate unsafe or borderline samples, which may introduce noise into the training process.

To further enhance the training data, an additional amplification step was applied to unsafe ASD samples (flagged as negative). Using KNN, the system selectively generated synthetic samples based on the local neighborhood class composition. This resulted in three datasets:

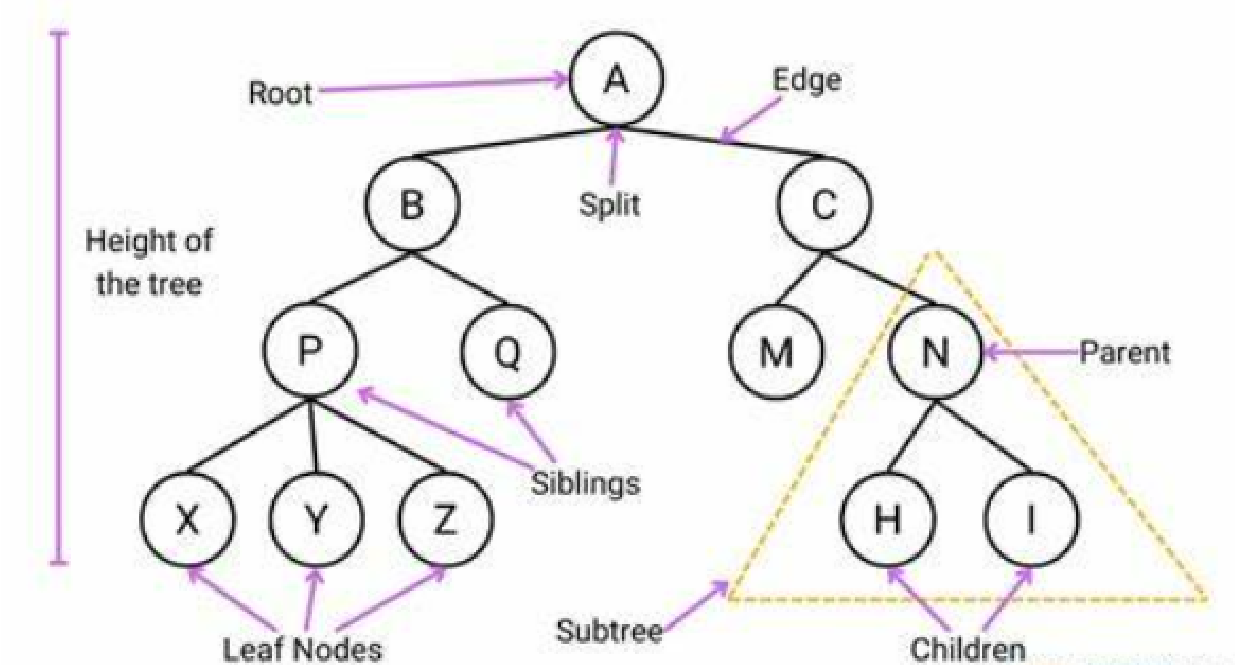
- **A0**: baseline dataset after SMOTE–IPF;
- **A1**: all unsafe ASD samples were amplified;
- **A2**: selective amplification based on local neighbor voting.

These datasets were saved in Excel format for subsequent model training. Empirical results showed that **SMOTE-IPF consistently produced better** balanced distributions with less overfitting, making it the preferred preprocessing strategy for ASD classification tasks.

### *C. Classification Models*

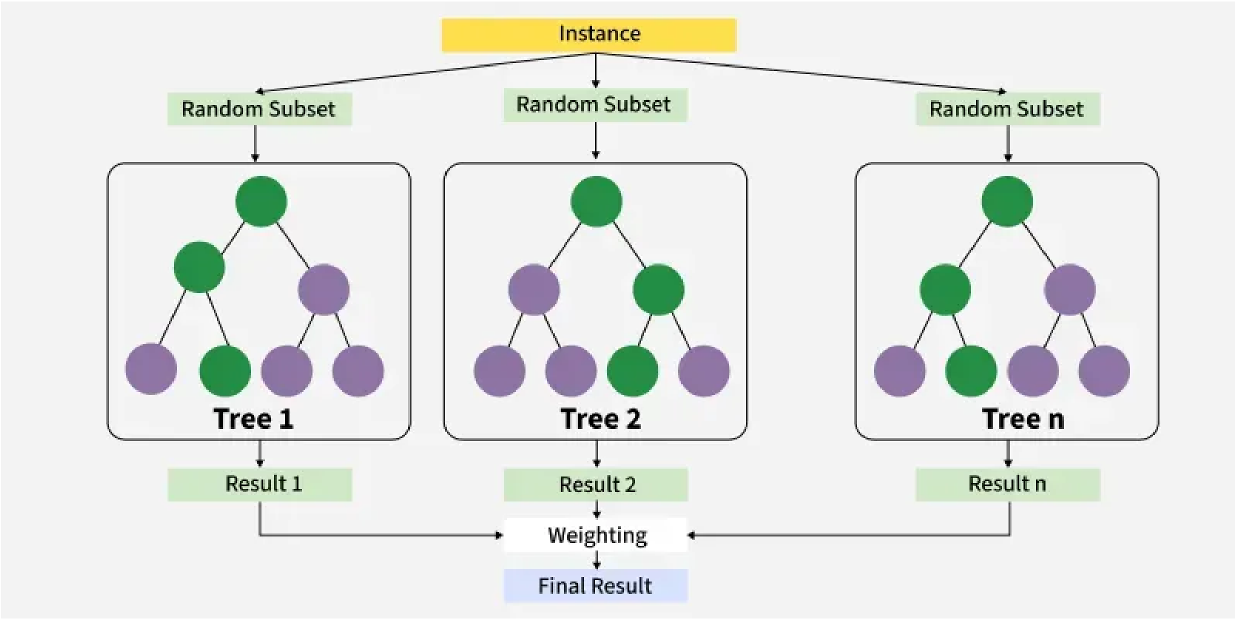
1) **Decision Tree**

A hierarchical structure of binary decisions based on feature thresholds to classify samples as ASD or non-ASD. Its simplicity and interpretability make it suitable for clinical applications. However, Decision Trees are prone to overfitting, especially on imbalanced datasets, and may overly rely on dominant features, such as specific behavioral indicators, leading to biased predictions. The model was implemented with default parameters, using Gini impurity as the splitting criterion, and evaluated on the preprocessed dataset.



Hình 1: Decision Tree structure

2) **XGBoost**
An ensemble learning method based on gradient boosting, combines multiple weak learners (decision trees) to improve predictive performance. It handles complex feature interactions effectively and includes regularization to mitigate overfitting. Despite its robustness, XGBoost can still exhibit bias toward dominant features in imbalanced datasets, limiting its generalizability in pediatric ASD classification.

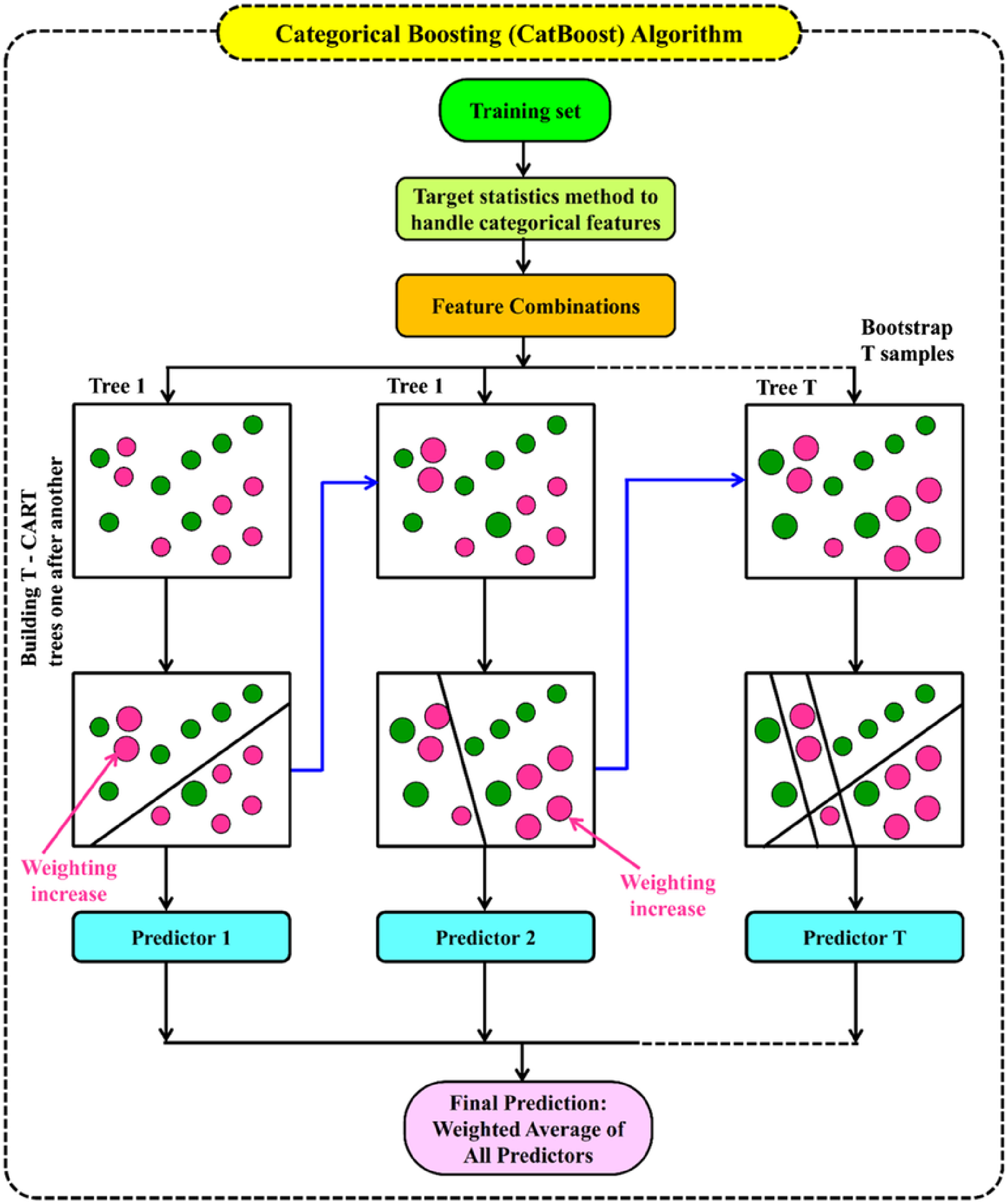


Hình 2: XGBoosting structure

3) **CatBoost**
An advanced Gradient Boosting Decision Trees (GBDT) algorithm, builds an ensemble of symmetric (oblivious) decision trees to enhance predictive performance and computational efficiency. Unlike traditional Decision Trees, which rely on a single, overfitting-prone model, CatBoost combines multiple weak learners with ordered boosting and regularization to

reduce overfitting, achieving superior generalization on complex datasets. It was configured with a depth of 6 and 100 iterations, leveraging its ability to process ordinal features (e.g., poor-moderate-good encodings) directly.

Compared to Decision Trees, CatBoost excels in handling categorical features through automated label encoding with Bayesian smoothing, eliminating the need for manual preprocessing. Relative to XGBoost, CatBoost offers improved categorical feature processing, optimized GPU performance via histogram-based computations, and automatic feature combinations for complex data patterns. Its structure, illustrates the ensemble of oblivious trees and the ordered boosting process, highlighting efficient memory usage and parallelization.


Categorical Boosting (CatBoost) Algorithm
Training set
Target statistics method to handle categorical features
Feature Combinations
Bootstrap T samples
Tree 1
Tree 1
Tree T
Building T - CART trees one after another
Weighting increase
Weighting increase
Predictor 1
Predictor 2
Predictor T
Final Prediction: Weighted Average of All Predictors

Hình 3: CatBoost Algorithm Structure

However, CatBoost may still struggle bias toward dominant features in imbalanced dataset, as applied in this study.

4) **OLA (Online Local Accuracy [?])**

a, Overview OLA model

A Dynamic Classifier Selection (DCS) technique from the DESlib library [?], dynamically selects the most competent classifier for each test sample based on its local accuracy within a region of competence. For each test instance $x$, OLA identifies the $k$-nearest neighbors in the training set, evaluates the classification accuracy of each base classifier on these neighbors, and selects the classifier with the highest local accuracy to predict $x$.

This approach excels in handling complex, heterogeneous, or imbalanced datasets, such as pediatric psychological records, where global models may underperform in specific regions of the feature space. A key strength of OLA is its flexibility in combining diverse base classifiers within a pool.

In this study, OLA was applied to an ensemble comprising **DecisionTreeClassifier** (simple but prone to overfitting), **KNeighborsClassifier** (effective in local regions), **XGBClassifier** (robust boosting for hard samples), and **RandomForestClassifier** (resilient to overfitting with many features). By leveraging local accuracy, OLA dynamically identifies the most suitable classifier for each sample, enhancing fairness and accuracy in ASD classification.

Implemented with custom code to integrate with the SMOTE-IPF preprocessed dataset, OLA mitigates feature bias and improves robustness on imbalanced data. OLA was hypothesized to outperform the other methods due to its ability to adapt to local data patterns, making it a promising tool for accurate and fair ASD diagnosis in pediatric settings.

b, Reducing Feature Bias Impact with OLA

Bảng II: Enhanced OLA Implementation Pipeline

| Step | Objective | Actions and Outcome |
|---|---|---|
| **Step 1: Data Normalization and Dimensionality Reduction with PCA** | Reduce correlations between features and mitigate the impact of biased features like *TiepXucMat*. | • Apply StandardScaler to normalize data (mean=0, variance=1).<br>• Use PCA to retain 95% variance and minimize dependence on *TiepXucMat*.<br>• **Outcome:** Bias from dominant features is minimized while essential information is preserved. |
| **Step 2: Create a Diverse Classifier Pool** | Build a diverse set of classification models to enhance OLA's flexibility and accuracy. | • Model Group 1 (PCA-based): DecisionTreeClassifier, XGBClassifier.<br>• Model Group 2 (excluding *TiepXucMat*): RandomForestClassifier.<br>• Model Group 3 (raw data): KNeighborsClassifier.<br>• **Outcome:** Enables OLA to select the most appropriate model per sample. |
| **Step 3: Train OLA on PCA-based Models** | Use OLA to dynamically select the best classifier for each test sample based on local accuracy. | • Train OLA on PCA-based classifiers using kNN ($k = 7$).<br>• Compute local accuracy in each neighborhood.<br>• **Outcome:** Classifier is chosen based on local performance, reducing global feature bias. |
| **Step 4: Combine Prediction Probabilities from All Model Groups** | Enhance prediction accuracy and robustness by integrating outputs from diverse model types. | • Aggregate probabilities from all groups (PCA, reduced, raw).<br>• Average the results to improve ROC AUC.<br>• **Outcome:** Increases reliability by mitigating influence of any single biased model group. |

### D. *Evaluation Metrics and Validation*

This section defines the evaluation metrics and validation techniques used to assess the performance of the proposed Overall Local Accuracy (OLA) model, alongside Decision Tree, XGBoost, and CatBoost, on the preprocessed ASD dataset.

1) *Evaluation Metrics:*

*a) Accuracy:* Measures the proportion of correctly classified samples out of the total.

**Formula**:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

**Explanation**:

- $TP$: True Positives – correctly predicted ASD cases.
- $TN$: True Negatives – correctly predicted non-ASD cases.
- $FP$: False Positives – non-ASD cases predicted as ASD.
- $FN$: False Negatives – ASD cases predicted as non-ASD.

**Significance**: Accuracy gives a general performance measure but can be misleading on imbalanced datasets like ASD.

**Evaluation**: Values closer to 1 are better, but should be interpreted cautiously in imbalanced settings.

*b) Recall (Sensitivity):* Measures the proportion of actual ASD cases correctly identified.

**Formula**:

$$\text{Recall} = \frac{TP}{TP + FN}$$

**Significance**: Critical for medical applications, ensuring ASD cases are not missed.

**Evaluation**: High recall reduces the risk of undetected ASD diagnoses.

*c) Specificity:* Measures the proportion of non-ASD cases correctly identified.

**Formula**:

$$\text{Specificity} = \frac{TN}{TN + FP}$$

**Significance**: Important for reducing false alarms (false positives).

**Evaluation**: High specificity prevents unnecessary concern or intervention for non-ASD individuals.

*d) Precision:* Measures the proportion of predicted ASD cases that are correct.

**Formula**:

$$\text{Precision} = \frac{TP}{TP + FP}$$

**Significance**: Indicates reliability of positive (ASD) predictions.

**Evaluation**: High precision reduces false ASD labels.

*e) F1-score:* The harmonic mean of precision and recall.

**Formula**:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

**Significance**: Provides a balanced measure for imbalanced data, combining both false positives and false negatives.

*f) ROC AUC:* Area under the Receiver Operating Characteristic curve.

**Formula**:

- True Positive Rate (Recall):

$$\text{TPR} = \frac{TP}{TP + FN}$$

- False Positive Rate (FPR):

$$\text{FPR} = \frac{FP}{FP + TN}$$

**Significance**: Reflects the model's ability to distinguish between ASD and non-ASD classes across thresholds.

**Evaluation**: Ranges from 0.5 (random) to 1.0 (perfect).

*g) G-Mean:* Geometric mean of recall and specificity.

**Formula**:

$$\text{G-Mean} = \sqrt{\text{Recall} \cdot \text{Specificity}}$$

**Significance**: Balances performance across classes by combining recall and specificity, addressing the limitations of accuracy in imbalanced datasets like ASD, where minority class detection (e.g., ASD cases) is critical.

**Evaluation**: Ranges from 0 to 1; a value close to 1 indicates balanced and effective classification across both classes, while lower values suggest imbalance or poor performance, necessitating model refinement.

*2) Validation Techniques:*

*a) Single Run Evaluation:* The dataset was split into 80% training and 20% testing with stratified sampling to preserve class distribution.

**Process**: Provides a fast and interpretable performance snapshot.

*b) Cross-Validation:* A 5-fold stratified cross-validation was conducted.

**Process**: The dataset was divided into 5 folds with preserved class distribution. Each fold was used once as test while others served as training.

*c) Feature Importance Analysis:* Evaluates the influence of original features on the model's predictions, providing insights into feature relevance for the ASD classification task.

**Process**: The importance scores of each original feature were computed using the model's built-in feature importance method. These scores were extracted for all features, ranked in descending order based on their contribution, and summed to reflect their overall impact. The results were visualized to highlight the most influential features, aiding in understanding feature bias, such as from the *TiepXucMat* column.

## IV. Experiments

This section outlines the experimental design, implementation, and results to evaluate the performance of the proposed Overall Local Accuracy (OLA) model and baseline models on two distinct ASD datasets. The experiments aim to assess model accuracy on a balanced public dataset and address feature bias and class imbalance in a private dataset, culminating in an enhanced OLA approach.

### A. Experimental Setup

*a)* ***Datasets****:*

- **Public Dataset:** A publicly available Kaggle dataset get 10 columns features and binary target, data not imbalacing in target and features.
- **Private Dataset:** A clinical ASD screening dataset collected from pediatric psychological evaluations. It is characterized by class imbalance and feature dominance, especially from the feature *TiepXucMat*.

*b)* ***Data Preprocessing****:*

- Both datasets were split into 80% training and 20% testing subsets using stratified sampling to maintain class proportions.
- Redundant and irrelevant columns were removed based on low variance, high correlation, high missing rate, or clinical irrelevance. Visualization tools such as heatmaps and bar plots supported this analysis.
- A consistent subset of 10 representative features-mapped to clinical ASD screening questions (e.g., pointing, eye contact)-was selected to ensure fairness and alignment between models.

*c)* ***Models****:*

- **Baseline Models:** Decision Tree, XGBoost, CatBoost, and the initial OLA configuration.
- **Enhanced OLA:** After identifying bias in the private dataset, OLA was refined to include multiple classifier pools to enhance robustness and fairness.

### B. Experimental Workflow

The experimental pipeline consisted of three main steps:

*a)* ***Step 1: Initial Evaluation on Both Datasets****:* Evaluate the baseline performance of Decision Tree, XGBoost, CatBoost, and OLA.

**Process:** All models were trained on the 10-question feature subset. Evaluation metrics on both datasets.

*b)* ***Step 2: Analysis of features Bias on Private Dataset****:* Assess whether model performance was heavily influenced by a single dominant feature.

**Process:** All four models assigned disproportionately high importance to the feature *TiepXucMat*, making it the most influential in predictions. Despite model diversity, their outputs converged due to reliance on this feature.

*c)* ***Step 3: Enhanced OLA with Feature Bias Mitigation****:* Mitigate the influence of *TiepXucMat* and improve generalization in OLA.

**Process:** The OLA classifier pool was enhanced with:

- PCA-based models (to decorrelate features),
- Models excluding *TiepXucMat* entirely (to enforce fairness),
- KNN-based models on raw and reduced data.

This strategy helped reduce model dependency on a single dominant feature and increased fairness in detecting ASD across varied behavioral patterns.

### C. Experimental Results

This subsection presents the results of the three experimental steps, including performance metrics for all models on both datasets, feature importance analysis, and the impact of the enhanced OLA model. The results are reported using single-run evaluations, 5-fold cross-validation, and visualizations.

*1) Initial Evaluation on Public and Private Datasets:*

*a) Public Dataset (Single-Run):*

Bảng III: Single-run performance of baseline models on the public dataset

| **Model** | **Accuracy** | **Recall** | **Specificity** | **Precision** | **F1-score** | **ROC AUC** | **G-Mean** |
|---|---|---|---|---|---|---|---|
| Decision Tree | 0.766 | 0.579 | 0.893 | 0.800 | 0.667 | 0.758 | 0.719 |
| XGBoost | 0.851 | 0.842 | 0.857 | 0.800 | 0.821 | 0.968 | 0.850 |
| CatBoost | 0.872 | 0.789 | 0.929 | 0.889 | 0.833 | 0.856 | 0.856 |
| Initial OLA | 0.830 | 0.737 | 0.893 | 0.842 | 0.824 | 0.938 | 0.811 |

*b) Public Dataset (Cross-Validation):*

Bảng IV: 5-fold cross-validation performance of baseline models on the public dataset

| **Model** | **Fold** | **Accuracy** | **Recall** | **Specificity** | **Precision** | **F1-score** | **ROC AUC** | **G-Mean** |
|---|---|---|---|---|---|---|---|---|
| Decision Tree | 1 | 0.787 | 0.895 | 0.714 | 0.680 | 0.773 | 0.914 | 0.799 |
| | 2 | 0.872 | 0.842 | 0.893 | 0.842 | 0.842 | 0.958 | 0.867 |
| | 3 | 0.745 | 0.789 | 0.714 | 0.842 | 0.714 | 0.850 | 0.751 |
| | 4 | 0.870 | 0.889 | 0.857 | 0.800 | 0.842 | 0.940 | 0.873 |
| | 5 | 0.804 | 0.684 | 0.893 | 0.842 | 0.743 | 0.854 | 0.780 |
| | **Mean** | **0.816** | **0.820** | **0.813** | **0.757** | **0.783** | **0.893** | **0.814** |
| XGBoost | 1 | 0.957 | 1.000 | 0.929 | 0.957 | 0.950 | 1.000 | 0.964 |
| | 2 | 0.957 | 0.947 | 0.964 | 0.947 | 0.947 | 0.998 | 0.956 |
| | 3 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 4 | 0.913 | 1.000 | 0.857 | 0.818 | 0.900 | 0.978 | 0.926 |
| | 5 | 0.978 | 0.947 | 1.000 | 1.000 | 0.973 | 0.998 | 0.973 |
| | **Mean** | **0.961** | **0.979** | **0.950** | **0.944** | **0.954** | **0.995** | **0.964** |
| CatBoost | 1 | 0.915 | 1.000 | 0.857 | 0.826 | 0.905 | 1.000 | 0.926 |
| | 2 | 0.979 | 1.000 | 0.964 | 0.950 | 0.974 | 0.992 | 0.982 |
| | 3 | 0.957 | 0.895 | 1.000 | 1.000 | 0.944 | 0.989 | 0.946 |
| | 4 | 0.935 | 1.000 | 0.893 | 0.857 | 0.923 | 0.976 | 0.945 |
| | 5 | 0.957 | 0.947 | 0.963 | 0.947 | 0.947 | 0.990 | 0.955 |
| | **Mean** | **0.949** | **0.968** | **0.935** | **0.916** | **0.939** | **0.989** | **0.951** |
| Initial OLA | 1 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 2 | 0.979 | 1.000 | 0.964 | 0.959 | 0.979 | 1.000 | 0.982 |
| | 3 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 4 | 0.978 | 1.000 | 0.964 | 0.947 | 0.973 | 1.000 | 0.982 |
| | 5 | 0.978 | 0.947 | 1.000 | 1.000 | 0.973 | 1.000 | 0.973 |
| | **Mean** | **0.987** | **0.989** | **0.986** | **0.979** | **0.985** | **1.000** | **0.987** |

*c) Private Dataset (Single-Run):*

Bảng V: Single-run performance of baseline models on the private dataset.

| **Model** | **Accuracy** | **Recall** | **Specificity** | **Precision** | **F1-score** | **ROC AUC** | **G-Mean** |
|---|---|---|---|---|---|---|---|
| Decision Tree | 0.965 | 0.949 | 0.987 | 0.989 | 0.969 | 0.986 | 0.968 |
| XGBoost | 0.973 | 0.959 | 0.987 | 0.990 | 0.974 | 0.997 | 0.973 |
| CatBoost | 0.983 | 0.980 | 0.987 | 0.990 | 0.985 | 0.997 | 0.983 |
| Initial OLA | 0.977 | 0.969 | 0.987 | 0.990 | 0.979 | 0.992 | 0.978 |

*d) Private Dataset (Cross-Validation):*

Bảng VI: 5-fold cross-validation performance of baseline models on the private dataset.

| Model | Fold | Accuracy | Recall | Specificity | Precision | F1-score | ROC AUC | G-Mean |
|---|---|---|---|---|---|---|---|---|
| Decision Tree | 1 | 0.870 | 0.920 | 0.820 | 0.836 | 0.876 | 0.903 | 0.869 |
| | 2 | 0.850 | 0.760 | 0.940 | 0.927 | 0.835 | 0.867 | 0.845 |
| | 3 | 0.840 | 0.840 | 0.840 | 0.840 | 0.840 | 0.833 | 0.840 |
| | 4 | 0.810 | 0.880 | 0.740 | 0.772 | 0.822 | 0.912 | 0.807 |
| | 5 | 0.790 | 0.740 | 0.840 | 0.822 | 0.779 | 0.799 | 0.788 |
| | **Mean** | **0.832** | **0.828** | **0.836** | **0.839** | **0.830** | **0.863** | **0.830** |
| XGBoost | 1 | 0.910 | 0.920 | 0.900 | 0.902 | 0.911 | 0.956 | 0.910 |
| | 2 | 0.950 | 0.940 | 0.960 | 0.959 | 0.949 | 0.988 | 0.950 |
| | 3 | 0.900 | 0.900 | 0.900 | 0.900 | 0.900 | 0.959 | 0.900 |
| | 4 | 0.900 | 0.960 | 0.840 | 0.857 | 0.906 | 0.958 | 0.898 |
| | 5 | 0.840 | 0.800 | 0.880 | 0.870 | 0.833 | 0.945 | 0.839 |
| | **Mean** | **0.900** | **0.904** | **0.896** | **0.898** | **0.900** | **0.961** | **0.899** |
| CatBoost | 1 | 0.890 | 0.900 | 0.880 | 0.882 | 0.891 | 0.963 | 0.890 |
| | 2 | 0.960 | 0.920 | 1.000 | 1.000 | 0.958 | 0.995 | 0.959 |
| | 3 | 0.910 | 0.920 | 0.920 | 0.918 | 0.909 | 0.980 | 0.919 |
| | 4 | 0.860 | 0.940 | 0.780 | 0.810 | 0.870 | 0.967 | 0.856 |
| | 5 | 0.850 | 0.820 | 0.880 | 0.872 | 0.845 | 0.949 | 0.849 |
| | **Mean** | **0.894** | **0.896** | **0.892** | **0.896** | **0.895** | **0.971** | **0.893** |
| Initial OLA | 1 | 0.970 | 0.940 | 1.000 | 1.000 | 0.969 | 1.000 | 0.970 |
| | 2 | 0.990 | 0.980 | 1.000 | 1.000 | 0.990 | 1.000 | 0.990 |
| | 3 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| | 4 | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 1.000 | 0.980 |
| | 5 | 0.960 | 0.920 | 1.000 | 1.000 | 0.958 | 1.000 | 0.959 |
| | **Mean** | **0.980** | **0.964** | **0.996** | **0.996** | **0.979** | **1.000** | **0.980** |

*2) Evaluate Metrics Enhanced OLA Performance:*

Bảng VII: Single-run performance of models on the private dataset with enhanced OLA (PCA + OLA with TiepXucMat control).

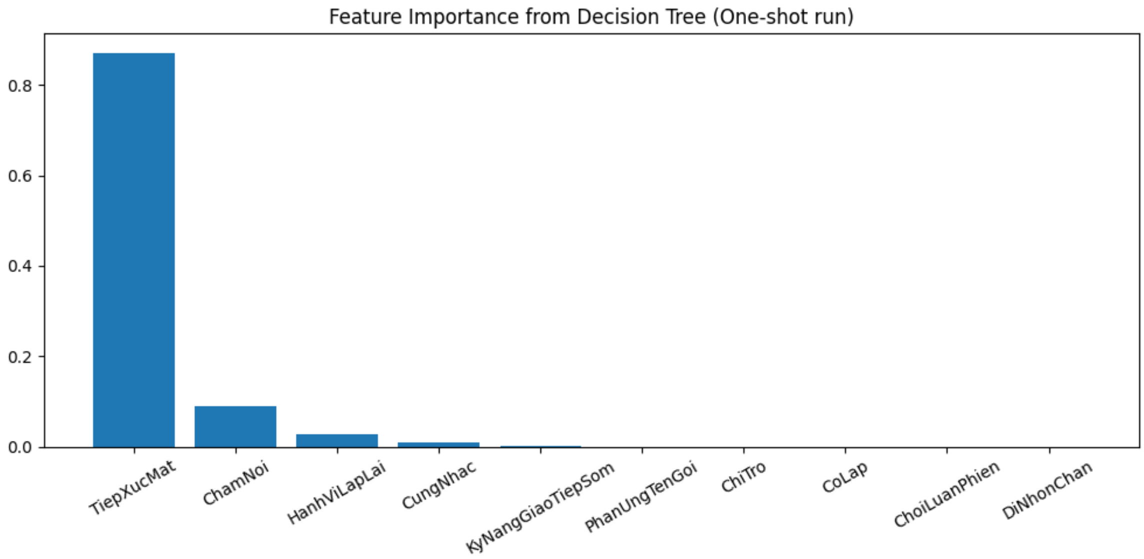
| Model | Accuracy | Recall | Specificity | Precision | F1-score | ROC AUC | G-Mean |
|---|---|---|---|---|---|---|---|
| Decision Tree | 0.965 | 0.949 | 0.987 | 0.989 | 0.969 | 0.986 | 0.968 |
| XGBoost | 0.973 | 0.959 | 0.987 | 0.990 | 0.974 | 0.997 | 0.973 |
| CatBoost | 0.983 | 0.980 | 0.987 | 0.990 | 0.985 | 0.997 | 0.983 |
| Initial OLA | 0.977 | 0.969 | 0.987 | 0.990 | 0.979 | 0.992 | 0.978 |
| Enhanced OLA | 0.977 | 0.969 | 0.987 | 0.990 | 0.979 | 0.983 | 0.978 |

Bảng VIII: 5-fold cross-validation performance of enhanced OLA (PCA + OLA with TiepXucMat control) on the private dataset.

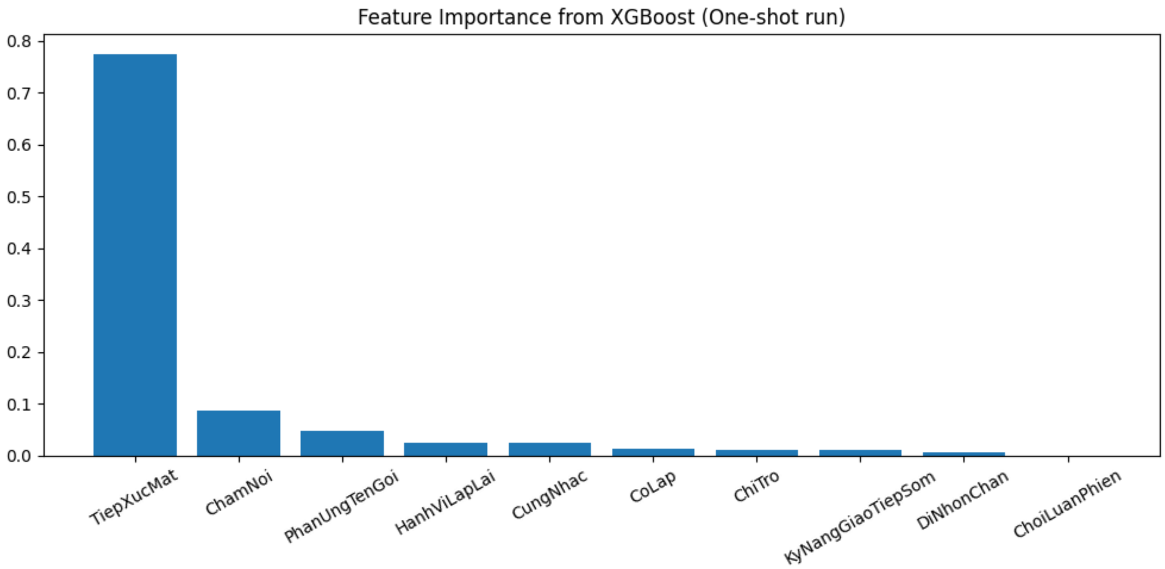
| Model | Fold | Accuracy | Recall | Specificity | Precision | F1-score | ROC AUC | G-Mean |
|---|---|---|---|---|---|---|---|---|
| Enhanced OLA | 1 | 0.971 | 0.969 | 0.973 | 0.979 | 0.974 | 0.994 | 0.971 |
| | 2 | 0.977 | 0.980 | 0.973 | 0.980 | 0.980 | 0.999 | 0.976 |
| | 3 | 0.977 | 0.969 | 0.987 | 0.990 | 0.979 | 0.992 | 0.978 |
| | 4 | 0.983 | 0.980 | 0.987 | 0.990 | 0.985 | 0.999 | 0.983 |
| | 5 | 0.965 | 0.938 | 1.000 | 1.000 | 0.968 | 0.985 | 0.969 |
| | **Mean** | **0.975** | **0.967** | **0.984** | **0.988** | **0.977** | **0.994** | **0.976** |

*3) Feature Importance Analysis on Private Dataset:* To evaluate the feature bias in the private dataset, we conducted a feature importance analysis before and after applying the Enhanced OLA model. The analysis focuses on identifying the extent of bias caused by the dominant feature *TiepXucMat* (contact face), which was previously identified as a column causing data bias.

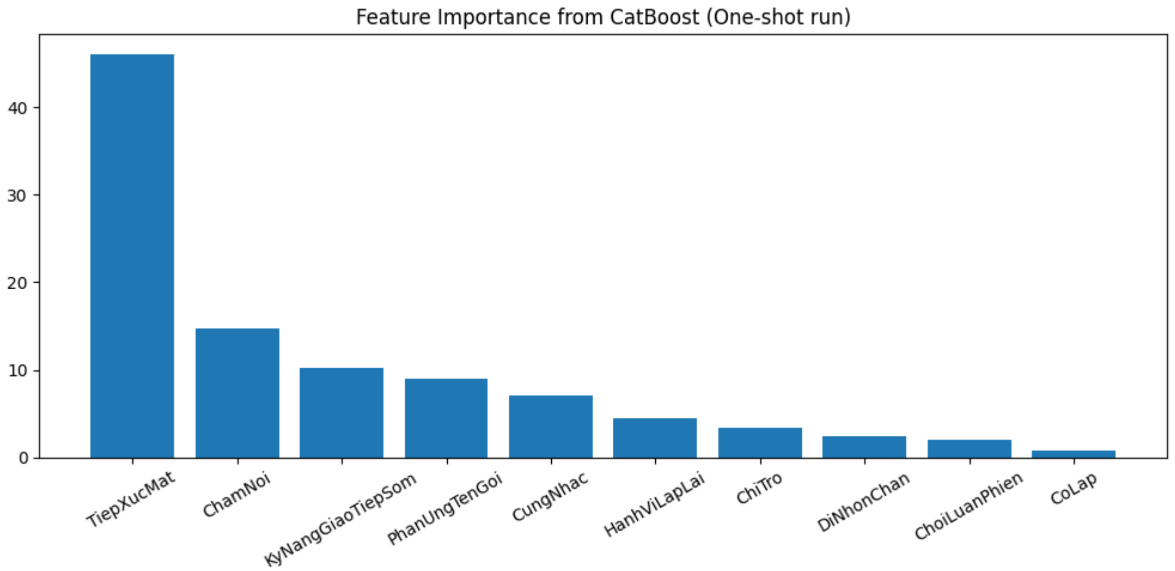
*a) Initial Feature Influence (Before Enhanced OLA):* Feature importance was analyzed using CatBoost, XGBoost, Decision Tree, and Initial OLA on the private dataset to quantify the influence of each feature.



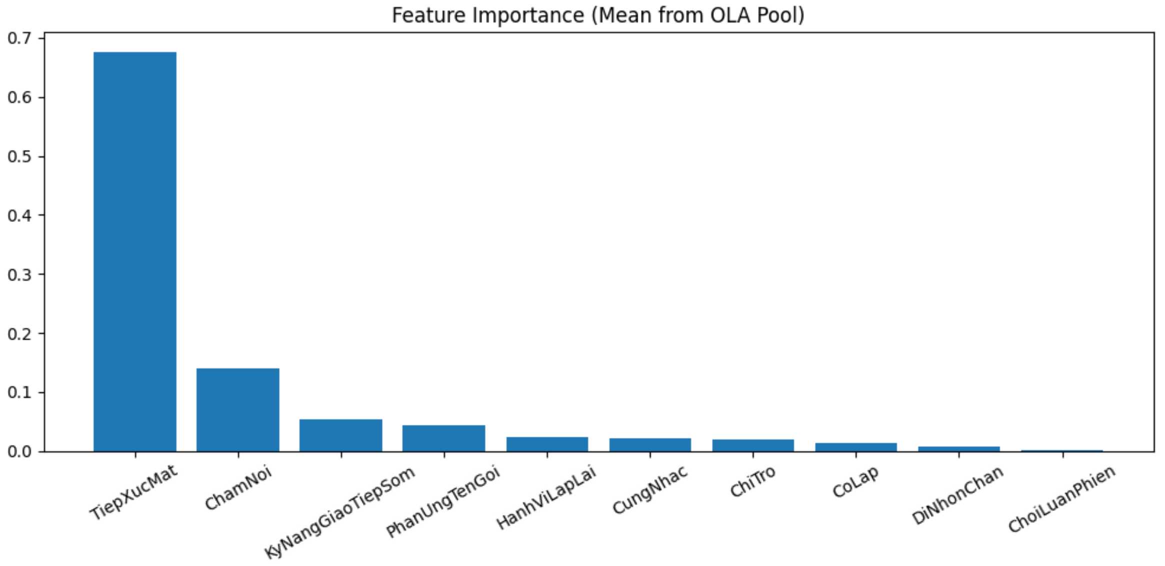
Hình 4: Decision Tree: *TiepXucMat* dominates with high contribution, causing generalization issues.



Hình 5: XGBoost: *TiepXucMat* dominates with high contribution, causing generalization issues.



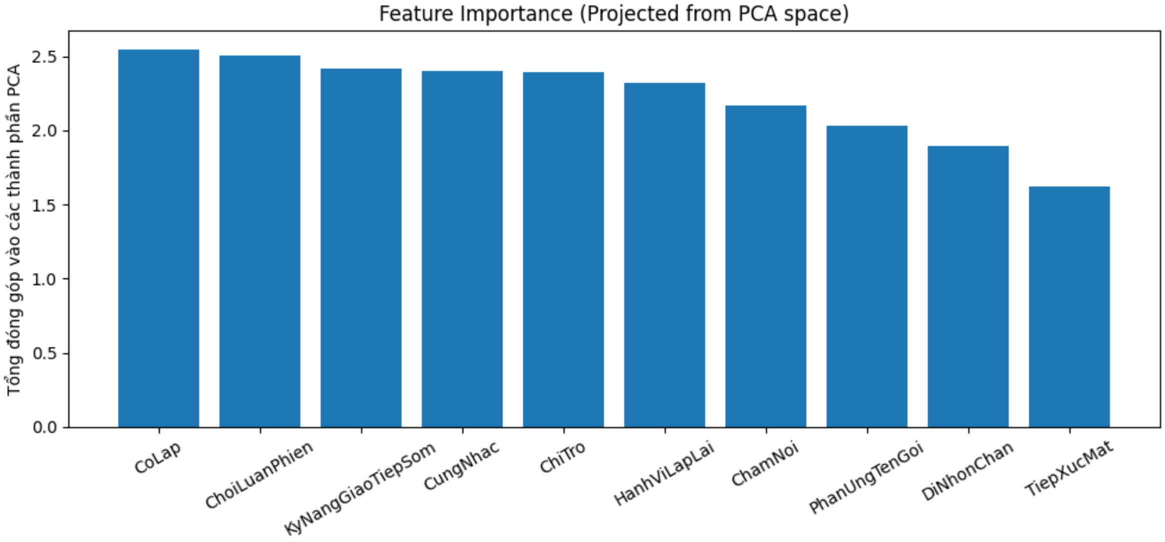
Hình 6: CatBoost: Balanced distribution, yet *TiepXucMat* still leads.



Hình 7: Initial OLA: Similar to CatBoost, more focus on local accuracy.

**Observation:** Across all models, *TiepXucMat* consistently emerged as the top feature, confirming its potential to introduce bias. This bias particularly reduced Recall for ASD cases, as models over-relied on this feature, limiting their reliability on the private dataset.

*b) Updated Feature Influence (After Enhanced OLA):* After applying Enhanced OLA with PCA and *TiepXucMat* control, we re-evaluated the feature importance to assess the impact of bias mitigation. Figure **??** visualizes the updated feature importance for Enhanced OLA.



Hình 8: Feature importance plot for Enhanced OLA after bias mitigation on the private dataset

**Observation:** The reduced dominance of *TiepXucMat* after applying Enhanced OLA indicates successful bias mitigation, aligning with the improved Recall and G-Mean observed in Section **??**.

## V. Discussion

This section analyzes the performance patterns across the evaluated models, discusses the improvements of OLA's local selection mechanism over global bias, compares our results with related work, and highlights limitations, strengths, weaknesses, and the practical applicability of OLA on the private dataset.

### *A. Public Dataset*

**CatBoost achieved the highest performance** (Accuracy 0.872) in the single-run scenario on the public dataset (Table III), significantly outperforming Decision Tree (Accuracy 0.766), XGBoost (Accuracy 0.851), and Initial OLA (Accuracy 0.830). However, **Initial OLA emerged as the top performer** (Accuracy 0.987, ROC AUC 1.000, G-Mean 0.987) in the 5-fold cross-validation (Table IV), surpassing CatBoost (Accuracy 0.957, ROC AUC 0.980, G-Mean 0.951), XGBoost (Accuracy 0.978, ROC AUC 0.995, G-Mean 0.973), and Decision Tree (Accuracy 0.816, ROC AUC 0.893, G-Mean 0.814). Notably, no feature bias occurred in the public dataset due to its standardized and balanced nature, enabling an accurate evaluation of the models' true performance and robustness across both single-run and cross-validation settings.

### *B. Private Dataset*

**CatBoost achieved the highest performance (Accuracy 0.983) with single run** on the private dataset (Table V), outperforming Decision Tree (Accuracy 0.965), XGBoost (Accuracy 0.973), and Initial OLA (Accuracy 0.977). However, **Initial OLA emerged as the top performer (Accuracy 0.980) with cross-validation**, surpassing CatBoost (Accuracy 0.894), XGBoost (Accuracy 0.910), and Decision Tree (Accuracy 0.832). This suggests that while CatBoost excels in single-run settings due to its robust handling of feature complexity, it adapts less effectively to cross-validation compared to Initial OLA. The latter's superior flexibility in dynamically selecting optimal models highlights its robustness and potential for handling imbalanced data across varied conditions. However, despite the promising results, the significant skew in the dataset toward the *TiepXucMat* column suggests that the high performance may not accurately reflect the models' true quality or practical applicability. Further improvements to the OLA model, which is currently performing well, are needed to address this limitation.

### *C. Enhanced OLA Performance*

**Enhanced OLA maintained its top performance** (Accuracy 0.985) in the 5-fold cross-validation (Table VIII), outperforming CatBoost (Accuracy 0.894), XGBoost (Accuracy 0.900), and Decision Tree (Accuracy 0.832), confirming its consistent superiority in cross-validation settings. Moreover, **Enhanced OLA effectively eliminated the *TiepXucMat* bias**, as shown by feature importance visualizations, where all features were balanced post-PCA. This adjustment ensures that no single feature dominates, thereby improving the reliability of screening results for practical deployment on imbalanced datasets.

## VI. Conclusion

This study introduces the Overall Local Accuracy (OLA) framework to improving feature bias in autism spectrum disorder (ASD) classification with tackle challenges of data imbalance using SMOTE-IPF . Models struggled with imbalanced data, even advanced models like CatBoost still faced limitations due to a “column causing data bias” that reduced reliability, especially since real-world data often carries hidden risks of such biases. The enhanced OLA, with added techniques to control this bias, proved to be a stronger solution, improving detection of ASD cases and ensuring stable performance.

The key contribution is a new approach that balances local accuracy to overcome global biases, making it more reliable for ASD diagnosis. This work lays the foundation for better, fairer diagnostic tools, especially for early screening where missing cases is a concern. However, challenges like the need for customization and limited dataset size remain. Future efforts should test OLA on larger datasets and include multi-modal data, like images and videos, to further enhance its effectiveness.

## References

[1] Harshita Chandrappa, Swaathi BR, Prof. Swimpy Pahuja. “Prediction of Autism Spectrum Disorder based on Machine Learning Approach,” *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 11, no. V, pp. 3521–3526, May 2023. doi: 10.22214/ijraset.2023.52417. Available: https://doi.org/10.22214/ijraset.2023.52417

[2] José A. Sáez, Julián Luengo, Jerzy Stefanowski, Francisco Herrera. “A Novel Technique for Text Classification using SMOTE-IDF based Oversampling and Deep Neural Network,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 10, 2014. doi: 10.14569/IJACSA.2021.051.

[3] Hanen Karamti, Raed Alharthi, Amira Al Anizi, Reemah M. Alhebshi, Ala’ Abdulmajid Eshmawi, Shtwai Alsubai, Muhammad Umer. “SMOTE-KNN: An Improved Algorithm Based on SMOTE and K-Nearest Neighbors for Imbalanced Data,” *IEEE Access*, vol. 9, pp. 95690–95701, Cancers 2023. doi: 10.1109/ACCESS.2023.5174412.

[4] B. D. Hung, V. V. Thoa, and X. T. Dang. “KSI – A Combined Clustering and Resampling Method with Noise Filtering Algorithm for Imbalanced Data Classification,” *Tạp chí Công nghệ Thông tin và Truyền thông*, vol. 1, no. 1, pp. xx–xx, 2019.

[5] Youngkyu Hong, Eunho Yang. “Unbiased Classification through Bias-Contrastive and Bias-Balanced Learning,” in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, pp. 20153–20165, 2021.

[6] Zuyan Liu, Yuhao Dong, Jiahui Wang, Ziwei Liu, Winston Hu, Jiwen Lu, Yongming Rao. “Ola: Pushing the Frontiers of Omni-Modal Language Model with Progressive Modality Alignment,” arXiv preprint arXiv:2502.04328, 2025. Available: https://arxiv.org/abs/2502.04328

[7] Y. Zeng, J. Liu, H. Lam, and H. Namkoong. “LLM Embeddings Improve Test-time Adaptation to Tabular Y|X-Shifts,”arXiv preprint arXiv:2410.07395, 2024, vailable: https://arxiv.org/abs/2410.07395

[8] J. M. Moriuchi, A. Klin, and W. Jones, “Mechanisms of Diminished Attention to Eyes in Autism,” *American Journal of Psychiatry*, vol. 173, no. 8, pp. 825–833, 2016. doi:10.1176/appi.ajp.2016.15081034.

[9] S. Ramos-Cabo, V. Vulchanov, and M. Vulchanova, “Different Ways of Making a Point: A Study of Gestural Communication in Typical and Atypical Early Development,” *Autism Research*, vol. 14, no. 5, pp. 984–996, 2021. doi:10.1002/aur.2474.

[10] M. Miller, S. Iosif, A. Hill, D. Young, S. Schwichtenberg, and S. Ozonoff, “Response to Name in Infants Developing Autism Spectrum Disorder: A Prospective Study,” *The Journal of Pediatrics*, vol. 165, no. 2, pp. 332–337, 2014. doi:10.1016/j.jpeds.2014.04.017.

[11] M. Montagut-Asunción, L. Llorente-Comí, L. P. Núñez-Nogueira, S. Sabaté-Masferrer, and L. Giné-Garriga, “Joint Attention and Its Relationship with Autism Risk Markers at 18 Months of Age,” *Children*, vol. 9, no. 4, 2022. doi:10.3390/children9040480.

[12] B. Ingersoll, “The Social Role of Imitation in Autism: Implications for the Treatment of Imitation Deficits,” *Infants & Young Children*, vol. 21, no. 2, pp. 107–119, 2008. doi:10.1097/01.IYC.0000314482.24087.14.

[13] Y. G. Lam and S. S. Yeung, “Symbolic Play in Children with Autism,” in *Comprehensive Guide to Autism*, Springer, 2014, pp. 491–508. doi:10.1007/978-1-4614-4788-7_26.

[14] S. Jabbar, M. Imran, A. B. Sargano, R. Khan, and H. Abbas, “Minority Class Oversampling Technique for Imbalanced Data Classification,” *Neural Computing and Applications*, vol. 32, pp. 3965–3980, 2020.

[15] J. Huang, L. Zhao, and D. Liu, “An Improved SMOTE Approach with Missing Value Imputation and Feature Enhancement for Clinical Data,” *IEEE Access*, vol. 9, pp. 44264–44273, 2021.

[16] Overall Local Accuracy: https://deslib.readthedocs.io/en/latest/modules/dcs/ola.html

[17] R. M. O. Cruz, R. Sabourin, and G. D. C. Cavalcanti, “DESlib: A dynamic ensemble selection library in Python,” 2018. [Online]. Available: https://github.com/scikit-learn-contrib/DESlib. Accessed: Jun. 3, 2025.

[18] F. F. Tabtah, “Autism Spectrum Disorder Screening” in *Proceedings of the 1st International Conference on Medical and Health Informatics*, Taichung City, Taiwan, ACM, 2017, pp. 1–6. Available: http://fadifayez.com/wp-content/uploads/2017/11/Autism-Spectrum-Disorder-Screening-Machine-Learning-Adaptation-and-DSM-5-Fulfillment.pdf