STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 0–18.
Published online in International Academic Press (www.IAPress.org)

A Comprehensive Evaluation of Balancing Approaches for Genomic Data in Autism Classification

Mahmoud Lham *, Aicha Majda

Informatics and Applications Laboratory, Moulay Ismail University, Meknes, Morocco

Abstract Class imbalance is a major challenge in the transcriptomic classification of autism spectrum disorders (ASD). High dimensionality and small sample sizes frequently cause machine learning models to favor the majority class. This work presents a comprehensive evaluation of 30 data balancing approaches using several ASD-related transcriptomic datasets derived from the GEO database. Each method is evaluated through three classifiers: Random Forest, Support Vector Machine, and Logistic Regression. Performance is ranked primarily using the precision-recall area under the curve (PR-AUC), along with ROC–AUC, accuracy, precision, recall, and F1-score. Stability across different settings is also assessed. The performance of the baseline model is worst on the minority class, and also shows greater variability. These findings highlight the need to address class imbalance directly. SMOTE+ENN ranked as the best method in our tests, the average PR–AUC reaches 0.94 with Random Forest, 0.97 with SVM, and 0.99 with Logistic Regression, these results remain stable across datasets with moderate variability. SMOTE-Tomek and oversampling-based methods demonstrate competitive but slightly lower performance. Generative and optimization-based approaches vary more in their performance and tend to incur higher computational cost. Overall, the findings show that hybrid, noise-aware balancing methods work best. They also offer clear guidance for building reliable and reproducible genomic ASD classifiers, especially when working with highly imbalanced, high-dimensional data by achieving a favorable balance between performance, stability, and computational efficiency.

Keywords Autism Spectrum Disorder, Genomic Data, Class Imbalance, Data Balancing Approaches, Precision medicine, Machine Learning.

AMS 2010 subject classifications 68T05, 62P10

DOI: 10.19139/soic-2310-5070-4142

1. Introduction

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental condition characterized by chronic impairment in social communication and the presence of restricted or repetitive behaviors, which has become increasingly prevalent worldwide[1][2]. High prevalence rates are consistently reported in large-scale epidemiological studies, underscoring the need for trustworthy and scalable computational tools for early detection and diagnosis [3].

Genomic studies have demonstrated that ASD is highly heritable and genetic, with rare variants and polygenic risk contributing to a convergence of neurodevelopmental pathways. Advances in gene expression profiling technologies have facilitated the generation of high-dimensional transcriptomic datasets, offering new opportunities for machine learning in autism classification and biomarker discovery [4][5].

Public repositories, such as the Gene Expression Omnibus (GEO) of the National Center for Biotechnology Information (NCBI), have facilitated large-scale access to transcriptomic datasets in ASD and control groups [6]. However, these data are often extremely high-dimensional, with few samples and pronounced class imbalance. In

*Correspondence to: Mahmoud Lham (Email: m.lham@umi.ac.ma). Informatics and Applications Laboratory, Moulay Ismail University, Meknes, Morocco

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 202x International Academic Press

many transcriptomic studies of ASD, one class (usually either ASD or control) has a population size orders of magnitude larger than the other; machine learning models trained on such data therefore tend to be biased toward the majority class and lose the ability to detect minority-class samples [7].

Class imbalance is a well-known concern in biomedical machine learning, particularly in genomic classification tasks, where false negatives may have serious clinical implications [8]. Conventional performance metrics such as accuracy and ROC–AUC can be overly optimistic in imbalanced cases, masking poor precision–recall performance under the minority class [9]. Thus, the choice of data balancing methods and imbalance-aware evaluation metrics is important for obtaining reliable and interpretable predictive models, and to ensure stable performance across data splits and datasets.

To address class imbalance, a wide range of approaches have been proposed, comprising oversampling, undersampling, hybrid methods, ensemble-based techniques, generative models, representation learning strategies, and loss-level optimization methods. Although several of these methods have been previously applied in biomedical or genomic data, the literature studies cover a more restricted set of applications and typically consider a reduced number of methods and metrics for comparison, with performance assessment based only on a single dataset. Recent comparative studies of imbalance-handling techniques, such as the evaluation of resampling strategies across generic UCI/OpenML benchmarks [10] or multimodal ensemble learning for biomedical outcome prediction [11], typically restrict their scope to a limited number of datasets, a single classifier family, or non-genomic tabular data, and do not jointly address dataset-, classifier-, and metric-level robustness. Therefore, a systematic metric-conscious comparison for balancing approaches in genomic autism classification is still lacking, particularly with respect to robustness across datasets and computational efficiency considerations[12].

The contribution of this work lies in the design and execution of a systematic, multi-dataset, multi-classifier, and metric-aware benchmarking framework that clarifies which existing families of balancing strategies constitute the most reliable defaults for high-dimensional, small-sample genomic ASD classification. To this end, we compare 30 data balancing approaches for genomic ASD classification across multiple public datasets downloaded from the GEO repository, using three classifiers: Random Forest, Support Vector Machine, and Logistic Regression. Performance is ranked using PR–AUC, along with ROC–AUC, accuracy, precision, recall, and F1-score. In addition to overall performance, we examine how methods behave across metrics and datasets, as well as their stability and computational cost. The results provide guidance for selecting suitable balancing methods in imbalanced genomic classification tasks related to autism.

2. Background and Related Works

2.1. Class Imbalance in Biomedical and Genomic Data

Data imbalance is a well-known problem in supervised learning, especially in the healthcare field where disease frequency varies and data collection is expensive. Multiple studies have shown the impact of class imbalance on model evaluation and training leading to classifiers with high overall performance but poor performance for clinically relevant classes [8] [13]. Imbalance, feature redundancy and noise increases overfitting and variance of high-dimensional genomic data. Recent surveys reveal that there is no one-size-fits-all solution for class imbalance, and that the performance of various methods varies substantially depending on features of the dataset such as sample size, dimensionality, and class overlap [14].

2.2. Data-level balancing approaches

- **Oversampling Methods**

Oversampling techniques address class imbalance by increasing the minority class by either replicating existing minority-class samples or creating synthetic ones. One of these techniques is SMOTE, designed to generate artificial data points via interpolation from the minority-class nearest neighbors [15] [16]. Based on the idea behind SMOTE, many variants have been proposed to address its known limitations, especially by emphasizing difficult boundary regions, utilizing density information or highlighting informative samples during generation.

While oversampling techniques are attractive in fields like biomedicine and genomics, where preserving all original data is often crucial, they can be susceptible to issues arising from noisy data and high dimensionality. Several studies have indicated that the synthetic samples generated by these methods might inadvertently amplify existing noise or misleading patterns when applied to datasets with a very large number of features [17]. Consequently, it's often beneficial to employ more refined oversampling methods and incorporate additional strategies to effectively manage noise in real-world applications.

- **Undersampling Methods**

Undersampling refers to a group of techniques designed to balance the class distribution for a classification dataset that has a skewed class distribution. To handle class imbalance, the undersampling methods use neighborhood-based rules to exclude redundant or confusing samples. Although this could facilitate the learning process and lower computational complexity, it may also reject informative majority-class instances, which pose a threat to generalization; an effect that is not negligible in small-sample clinical data analyses.

- **Hybrid Methods**

These approaches handle such trade-offs by combining oversampling with post-resampling cleaning strategies that remove noisy or duplicate samples. A classic example is SMOTE combined with Edited Nearest Neighbors (SMOTE+ENN), which, after oversampling minority-class samples through synthetic generation techniques, performs neighbourhood cleaning to remove uncertain samples, thereby increasing both class separability and stability [7]. Hybrid approaches are typical for biomedical machine learning, as high-throughput data is often noisy, and the value of a feature often depends on other features.

2.3. Ensemble and Generative Methods

To address the issue of class imbalance, ensemble-based approaches incorporate resampling methods into ensemble learners. Methods like EasyEnsemble, Self-Paced Ensemble (SPE), which progressively includes majority-class samples of increasing classification difficulty to build a sequence of balanced sub-classifiers [18], and variants of boosting train multiple classifiers based on balanced datasets, leveraging diversity to enhance robustness and recognition of the minority class [19]. They have been extensively studied in the context of imbalanced learning due to their ability to integrate sampling and classification within a unified framework.

Generative models have been introduced more recently as an alternative to classical oversampling methods. Variational autoencoders (VAEs), generative adversarial networks (GANs), and diffusion models approach the problem differently, learning the data distribution and generating synthetic samples that are more consistent with global properties instead of simply interpolating locally. However, despite their conceptual advantages, single studies have shown that generative models are not trivial to train and can suffer from training instabilities, mode collapse, and lack of scalability -in the case of tabular data with minority examples- [16] [20] [21].

2.4. Loss-Level and Representation Learning Methods

Loss-level methods address this class imbalance by adjusting the learning criterion instead of the data distribution. Instead of oversampling or undersampling, these methods directly incorporate sampling ratios in the cost function to overcome class imbalance. As representative examples, Focal-Loss down-weights easy to classify samples and focuses learning on harder minority instances, while Class-Balanced Loss reweighs the classes according to their effective number of samples [22] [23]. Recent methods, such as Dynamic Curriculum Learning (DCL), dynamically control the importance of samples during training to sequentially focus on hard or undersampled instances. These methods are widely used in deep learning frameworks, especially for severely imbalanced visual and biomedical datasets.

Representation learning methods address the imbalance issue by learning a more compact and informative latent representation from the original feature space before classification. Autoencoders and similar architectures learn low-dimensional representations that can help clean up noise and redundancy in high-dimensional data, making the subsequent classification task more manageable [24]. Representation learning is integrated with oversampling

in some studies by using SMOTE in the learned latent space to generate more meaningful synthetic samples. Although these methods lead to better feature quality, their success is highly sensitive to representation stability, the availability of training data, and the utility of the learned latent space for resampling.

2.5. Evaluation Metrics and Comparative Limitations

Metric selection plays an important role in evaluating the approaches used to address class imbalance. Standard performance metrics are known to be unreliable in imbalanced settings, where ROC–AUC and PR–AUC offer complementary, more informative alternatives. These precision-recall-based metrics can provide useful information, especially when the correct identification of a target class is prioritized [9].

Nevertheless, many genomic classification investigations use a limited number of evaluation metrics or evaluate only some balanced algorithms on a single dataset. These practices prevent reproducibility and undermine meaningful comparisons between studies, so it is impossible to formulate general conclusions about the robustness of the imbalance-handling methods. On the same three GEO cohorts used in the present study, Garbulowski et al. applied an interpretable rule-based classifier without explicit class-balancing, reporting an average accuracy of 74% and an AUC of 0.80 across datasets [25]; this benchmark builds directly on that precedent by evaluating how 30 balancing strategies affect performance on these same cohorts.

3. Materials and Methods

3.1. Problem Formulation

Let $X \in \mathbb{R}^{n \times p}$ denote a transcriptomic dataset with n samples and p features ($p \gg n$), and let $y \in \{0,1\}^n$ represent the corresponding class labels. In ASD-related transcriptomic studies, Class imbalance is common in ASD-related transcriptomic studies, in which one class is under-represented compared with another. Under such conditions, standard learning algorithms tend to minimize overall error by favoring the majority class, leading to a loss in recognition of minority classes due.

The goal of data balancing is to reduce this bias, thereby altering the effective class distribution that the learning algorithm encounters during the training. However, the balancing approach in high-dimensional dataset and small samples size may give rise to side effects (e.g. higher variance, noise propagation or boundary distortion) thus requires a further exploration on how classifier performance is affected due to imbalance.

3.2. Overall pipeline

The overall experimental pipeline followed in this study is depicted in Figure 1.

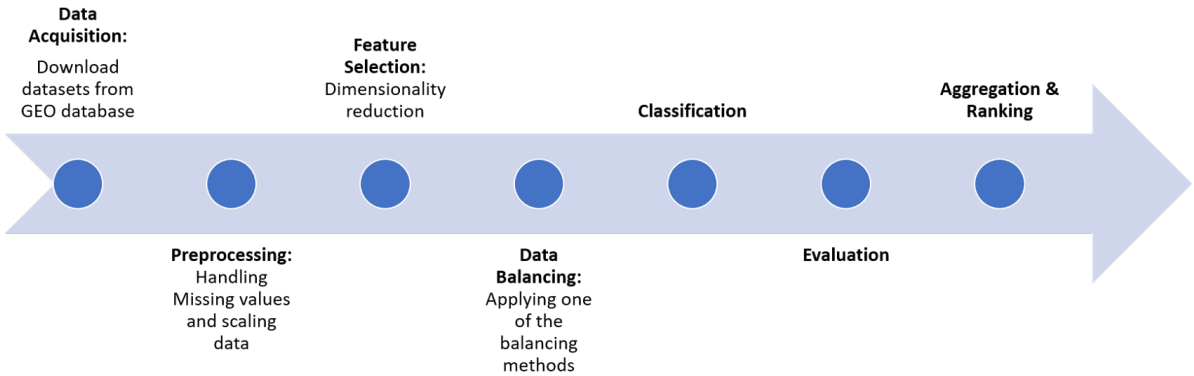


Figure 1. Experimental pipeline for genomic autism classification, from data acquisition to aggregation and ranking of balancing approaches.

3.3. Datasets

We carried out experiments using multiple genome-wide autism data from GEO (Gene Expression Omnibus) of NCBI [26]. All datasets are derived from microarray gene expression profiles, which were chosen to be comparable with regard to experimental and data preprocessing. For both datasets, samples have been labeled as Autism Spectrum Disorder (ASD) or control.

Table 1. Characteristics of the genomic microarray datasets used in this study.

Dataset	Platform	Number of Features	ASD	Control	Total	Imbalance ratio
GSE25507	GPL570	54613	82	64	146	1.28
GSE6575	GPL570	54613	35	21	56	1.67
GSE18123	GPL570	54613	66	33	99	2.00

All datasets, presented in Table 1, are produced by the same microarray platform (GPL570), so that there is some technical consistency between experiments. Both datasets feature extremely high feature dimensionality (54613 genes) and a small number of samples, making the learning problem particularly difficult due to the high dimensionality and class imbalance. The imbalance ratios vary across datasets, reflecting realistic cohort compositions commonly encountered in genomic autism studies.

3.4. Data Preprocessing

Preprocessing involved handling missing values and normalizing the data through feature scaling to prevent numerical instability and enable comparison between features.

1. The datasets were downloaded from the GEO repository by using the GEOParse Python library.
2. After downloading the data, we cleaned it up by handling missing values: samples for which expression values were missing across all probes were removed prior to feature selection.
3. Transposition of rows and columns to represent probes as columns and samples as rows.
4. Remove irrelevant columns that start with AFFX- prefix (control probes).
5. Adding DiseaseState column indicating the patient status (ASD or NONASD).
6. Saving each dataset separately in CSV format.

3.5. Feature selection

Given the high dimensionality of microarray data, feature selection was applied prior to balancing and classification through a staged filter pipeline for progressive dimensionality reduction while controlling computational cost:

- **Variance filtering:** the full probe set was first reduced to the top 2000 genes by expression variance, as the less expensive preliminary filter prior more expensive, more discriminative filters.
- **ANOVA F-test:** a univariate filter method that ranks genes by their discriminative power based on variance between classes [27], for further reduction to top 1000 genes.
- **Minimum Redundancy Maximum Relevance (mRMR):** a technique that chooses features maximizing relevance to the target class (ASD versus control) while minimizing redundancy among the selected features [28], applied last to reduce the set to the final top 100 genes used for balancing and classification.

This staged combination was chosen because ANOVA F-test is computationally inexpensive and removes clearly non-discriminative genes at scale, whereas mRMR is more costly but explicitly penalizes redundancy among the retained genes, a property that a purely univariate filter such as ANOVA cannot provide; applying mRMR directly to the full probe set would be computationally prohibitive, which motivates the staged design.

3.6. Evaluated balancing approaches

The 30 data balancing approaches were assessed, covering multiple methodological families as defined in table 2. All such balancing methods were performed on the training data only during k-fold cross-validation, while once again validation data was left intact to obtain a fair performance assessment.

Table 2. Taxonomy of balancing methods evaluated in this study.

Category	Method	Balancing Level	Role
Baseline	No Balancing	None	Reference point for evaluating imbalance impact
Oversampling	Random Oversampling [29]	Data-level	Increases minority density, reduces bias
	SMOTE [15]	Data-level	Synthetic interpolation of minority samples
	SVM-SMOTE [30]	Data-level	Boundary-focused generation using SVM
	MWMOTE [31]	Data-level	Weighted generation of informative samples
	Safe-Level-SMOTE [32]	Data-level	Noise-aware synthetic generation
	Borderline-SMOTE [33]	Data-level	Focuses on decision boundary regions
	KMeans-SMOTE [34]	Data-level	Density-aware synthetic sample generation
	ADASYN [35]	Data-level	Adaptive generation in complex regions
Undersampling	NearMiss [36]	Data-level	Removes majority samples near minority
	Tomek Links [37]	Data-level	Removal of overlapping samples
	Cluster Centroids [38]	Data-level	Compresses majority class representation
	ENN [39]	Data-level	Cleans overlapping and noisy samples
	NCL [40]	Data-level	Neighborhood-based sample cleaning
	OSS (One-Sided Selection) [41]	Data-level	Selective removal of majority samples
Hybrid	SMOTE + ENN [29]	Data-level	Balances density while removing noise
	SMOTE + Tomek Links [42]	Data-level	Boundary cleaning after oversampling
Ensemble-Based	EasyEnsemble [19]	Algorithm-level	Variance reduction via ensemble learning
	SMOTEBoost [43]	Algorithm-level	Boosting with oversampling
	RUSBoost (Random Under-Sampling with AdaBoost) [44]	Algorithm-level	Boosting with undersampling
	SPE (Self-Paced Ensemble) [18]	Algorithm-level	Progressive sample inclusion
Generative Models	VAE [45]	Data-level	Latent-space sample generation
	GAN (Vanilla) [46]	Data-level	Adversarial synthetic data generation
	WGAN (Wasserstein GAN) [47]	Data-level	Stable adversarial training
	CTGAN (Conditional Table GAN) [48]	Data-level	Conditional tabular data generation
	TabDDPM [49]	Data-level	Diffusion-based tabular generation
Optimization-Level	Class-Balanced Loss [23]	Loss-level	Adjusts loss weighting by class frequency
	Focal Loss[22]	Loss-level	Focuses learning on hard samples
	DCL [50]	Loss-level	Curriculum-based dynamic reweighting
Representation Learning +Oversampling	Autoencoder + SMOTE [24]	Feature-level	Learns latent feature representations prior to oversampling

3.7. Classification Models

To evaluate the strength of balancing approaches for diverse learning paradigms, each balancing approach was evaluated using three classifiers:

- **Random Forest Classifier:** an ensemble method consisting of a collection of decision trees where the model makes predictions using an average (for regression) or majority vote (for classification) over its member decision trees. Being capable of coping with high-dimensional data and resistant to noise, it is suitable for transcriptomic based classification, and was accordingly selected as the primary classifier for ranking balancing methods.
- **SVM-RBF (Support Vector Machine with a radial basis function kernel):** is considered as a typical margin-based non-linear classifier. SVMs are known for their sensitivity to class overlap and feature scaling, both of which make SVMs especially informative for examining the consequences of data balancing on decision boundary geometry.
- **Logistic Regression:** offers interpretable decision boundaries and acts as a useful benchmark for evaluating whether performance improvements can rather be attributed to balancing data than estimating more complex models; together with SVM, it serves to corroborate that the ranking obtained with Random Forest is not an artifact of the classifier choice.

To guarantee a fair comparison, default hyperparameters of all classifiers were deliberately kept constant between datasets and balancing methods, so that observed performance differences could be attributed to the balancing strategy rather than to per-method tuning. Internal class weighting was not used within the classifiers, since external balancing effects could be determined separately.

3.8. Evaluation metrics

For each balancing method, evaluation metrics were calculated separately for every classifier, dataset and cross-validation fold. We use these measures based-on the confusion matrix (Table 3):

Table 3. Confusion matrix

	Actual Positive	Actual Negative
Predicted Positive	True Positive (TP)	False Positive (FP)
Predicted Negative	False Negative (FN)	True Negative (TN)

- **Accuracy (ACC):** measures the overall proportion of correctly classified instances among all samples.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

- **Recall (Sensitivity):** measures the proportion of actual positive cases that are correctly identified by the model.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

- **Precision**: measures the proportion of predicted positives that are actually positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

- **F1-score**: Precision and recall are combined in a harmonic manner to trade off between a false positives and false negatives.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

- **ROC–AUC**: region under Receiver Operating Characteristic curve, threshold independent separability measure.

$$\text{ROC_AUC} = \int_0^1 \text{TPR}(\text{FPR})\, d\text{FPR} \tag{5}$$

where,

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \& \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

- **PR–AUC**: More informative than ROC Area under the Curve when there is an imbalanced dataset[9].

$$\text{PR_AUC} = \int_0^1 \text{Precision}(\text{Recall})\, d\text{Recall} \tag{6}$$

Since the classification problem was imbalanced and the detection of ASD-related samples is more important, PR–AUC was considered as the ranking metric by default, with other metrics included for completeness, and results are reported as mean $\pm$ standard deviation to reflect variability across evaluation settings.

3.9. Aggregation and Ranking

For each balancing technique, inferred performance was calculated separately on each dataset and k-fold cross-validation. The performance across different cross-validation folds and datasets was averaged to obtain stable estimates.

For comparison purposes, balancing approaches were ordered by the mean of PR–AUC across datasets. This aggregation approach favors generalization and consistency over the best possible performance on any single dataset.

Unless otherwise noted, the reported rankings correspond to those obtained using the Random Forest classifier (see Section 3.7 for the classifier-choice rationale), with SVM and Logistic Regression results reported separately for comparison.

3.10. Implementation details

We implemented all experiments using Python 3.13 and the following libraries:

- scikit-learn (classification, metrics, cross-validation) [51],
- imbalanced-learn (oversampling/undersampling strategies) [52],
- Random Forest Classifier (RandomForestClassifier) fixing random_state.
- Support Vector Classifier (SVC) with RBF as kernel using (C = 1.0, gamma = 'scale'), enabling probabilistic outputs (probability = True) to allow computation ROC-AUC and PR-AUC, and fixing random_state to ensure reproducibility.
- Logistic Regression (LogisticRegression) with L2 regularization the liblinear solver, fixing random_state to guarantee reproducibility.
- SDV (synthetic data generation, CTGAN),
- Custom implementations for TabDDPM and DCL.

Feature values were standardized prior to classification and to all balancing methods using scikit-learn's StandardScaler. Table 4 reports the key hyperparameters used for each of the 30 balancing methods.

Table 4. Key hyperparameters used for each balancing method (fixed across datasets, classifiers, and folds).

Method	Key hyperparameters
Random Oversampling	default (`RandomOverSampler`)
SMOTE	$k = 3$
SVM-SMOTE	$k = 3$, $m = 5$
MWMOTE	default (`smote_variants`)
Safe-Level-SMOTE	default (`smote_variants`)
Borderline-SMOTE	kind = borderline-1, $k = 3$
KMeans-SMOTE	cluster_balance_threshold = 0.01, $k = 3$
ADASYN	$n = 3$
NearMiss	version = 1
Tomek Links	default
Cluster Centroids	default
ENN	default (`EditedNearestNeighbours`)
NCL	default (`NeighbourhoodCleaningRule`)
One-Sided Selection (OSS)	default
SMOTE + ENN	default (`SMOTEENN`)
SMOTE + Tomek Links	default (`SMOTETomek`)
EasyEnsemble	$n_{\text{estimators}} = 10$
SMOTEBoost	SMOTE (default) + AdaBoost (decision-tree stump, $n_{\text{estimators}} = 50$)
RUSBoost	random undersampling + AdaBoost (decision-tree stump, $n_{\text{estimators}} = 50$)
SPE (Self-Paced Ensemble)	random undersampling + Bagging (tree depth = 3, $n_{\text{estimators}} = 10$)
VAE	latent dim = 32, epochs = 50, lr = $1\text{e}{-3}$
GAN (Vanilla)	batch size = 16, lr = $2\text{e}{-4}$ (generator and discriminator)
WGAN	noise dim = 32, epochs = 300, batch size = 32, lr = $1\text{e}{-4}$, critic iters = 5, $\lambda_{\text{gp}} = 10$
CTGAN	epochs = 50, batch size = 16, pac = 1
TabDDPM	$T = 200$ diffusion steps, epochs = 500, batch size = 64, lr = $1\text{e}{-3}$, hidden = (256, 256)
Class-Balanced Loss	hidden dim = 64, lr = $1\text{e}{-3}$, epochs = 30, $\beta = 0.9999$
Focal Loss	lr = $1\text{e}{-3}$, $\alpha = 1$, $\gamma = 2$
DCL	hidden dim = 64, lr = $1\text{e}{-3}$, epochs = 30, $\gamma = 2.0$, $\beta = 0.9999$
Autoencoder + SMOTE	encoding dim = 50, epochs = 10, lr = $1\text{e}{-3}$, then SMOTE ($k = 3$) in latent space

All experiments are run on a personal computer environment running **Windows 11 Home (64-bit, build 26100)**. The hardware configuration included an **HP Laptop 15s-fq2xxx** equipped with an **11th Gen Intel® Core™ i7-1165G7 CPU @ 2.80GHz (8 logical cores)** and **8 GB of RAM**. All PyTorch-based models (VAE, GAN, WGAN, TabDDPM, Class-Balanced Loss, Focal Loss, DCL, and the autoencoder) were configured to use a CUDA GPU when available and to fall back to CPU otherwise; no GPU was available in the reported experiments, so all runs were executed on CPU. SMOTE-family neighbor counts (Table 4) were reduced from their typical default ($k = 5$) to $k = 3$ (or $n = 3$ for ADASYN) to remain valid given the smallest minority class across datasets (21 samples in GSE6575).

4. Results

4.1. Overall Performance Trends

The table 5 presents a concise overview of classification performance across balancing approach categories using Random Forest Classifier.

Table 5. Family-wise classification performance averaged across all evaluated methods, datasets, and cross-validation folds using Random Forest Classifier.

Category	Accuracy	F1-score	ROC–AUC	PR–AUC	Precision	Recall
Hybrid	**0.87**	**0.88**	**0.87**	**0.92**	**0.88**	**0.89**
Oversampling	0.82	0.82	0.82	0.88	0.83	0.84
Generative Models	0.81	0.75	0.79	0.85	0.84	0.72
Undersampling	0.75	0.71	0.74	0.81	0.77	0.71
Optimization-Level	0.78	0.73	0.80	0.79	0.71	0.78
Ensemble-Based	0.71	0.64	0.70	0.75	0.66	0.67
Representation Learning + Oversampling	0.76	0.67	0.79	0.74	0.70	0.68
Baseline (No Balancing)	0.75	0.60	0.71	0.74	0.68	0.57

As shown in Table 5, all balancing strategy categories achieve higher performance than the baseline category with a PR–AUC of 0.74 and a recall of 0.57, demonstrating a strong bias toward the majority class. Hybrid approaches achieve the best overall performance across all metrics, with PR–AUC = 0.92. Both oversampling and generative model-based methods also present competitive results with PR–AUC values of 0.88 and 0.85, respectively. Undersampling, optimization-level, and ensemble-based strategies show more moderate improvements over the baseline. Representation learning combined with oversampling improves several metrics compared with the baseline, but still performs below hybrid and oversampling methods.

4.2. Method-Level Performance Analysis

The top 10 balancing methods ranked by PR–AUC using the Random Forest classifier are reported in Table 6. Results are expressed as mean ± standard deviation aggregated across cross-validation folds and datasets.

Table 6. Top-10 balancing approaches ranked by PR–AUC using Random Forest Classifier (mean ± standard deviation across datasets and cross-validation folds).

Rank	Method	Accuracy	F1-score	ROC–AUC	PR–AUC	Precision	Recall
1	**SMOTE+ENN**	**0.90±0.12**	**0.91±0.11**	**0.90±0.12**	**0.94±0.07**	**0.91±0.12**	**0.94±0.14**
2	KMeansSMOTE	0.83±0.10	0.84±0.09	0.83±0.10	0.89±0.07	0.85±0.14	0.85±0.12
3	Borderline-SMOTE	0.85±0.11	0.85±0.13	0.85±0.12	0.89±0.08	0.85±0.13	0.87±0.17
4	SMOTE+Tomek Links	0.84±0.11	0.84±0.10	0.84±0.12	0.89±0.08	0.85±0.14	0.84±0.13
5	Random Oversampling	0.83±0.09	0.84±0.09	0.83±0.09	0.89±0.07	0.84±0.13	0.85±0.14
6	ADASYN	0.83±0.11	0.83±0.10	0.83±0.11	0.89±0.08	0.84±0.14	0.86±0.15
7	VAE	0.82±0.11	0.80±0.12	0.82±0.11	0.88±0.09	0.88±0.15	0.78±0.17
8	ENN	0.85±0.17	0.82±0.21	0.83±0.17	0.88±0.16	0.89±0.21	0.79±0.25
9	MWMOTE	0.81±0.11	0.81±0.12	0.82±0.11	0.88±0.08	0.84±0.14	0.82±0.16
10	GAN (Vanilla)	0.81±0.11	0.80±0.12	0.81±0.11	0.88±0.09	0.86±0.14	0.77±0.17

SMOTE+ENN achieves the best overall performance, with a PR–AUC of 0.94 ± 0.07. Followed by KMeansSMOTE and Borderline-SMOTE, with slightly lower scores but comparable stability. Overall, most balancing methods show strong performance with moderate variability (std typically between 0.07 and 0.14).

Table 7. Top-10 balancing approaches ranked by PR–AUC using Random Forest Classifier (mean ± standard deviation across datasets).

Rank	Method	Accuracy	F1-score	ROC–AUC	PR–AUC	Precision	Recall
1	**SMOTE+ENN**	**0.90±0.02**	**0.91±0.02**	**0.90±0.02**	**0.94±0.02**	**0.91±0.04**	**0.94±0.04**
2	KMeansSMOTE	0.83±0.04	0.84±0.03	0.83±0.04	0.89±0.03	0.85±0.04	0.85±0.04
3	Borderline-SMOTE	0.85±0.05	0.85±0.06	0.85±0.05	0.89±0.05	0.85±0.06	0.87±0.08
4	SMOTE+Tomek Links	0.84±0.06	0.84±0.06	0.84±0.06	0.89±0.04	0.85±0.05	0.84±0.06
5	Random Oversampling	0.83±0.06	0.84±0.07	0.83±0.06	0.89±0.04	0.84±0.04	0.85±0.11
6	ADASYN	0.83±0.05	0.83±0.06	0.83±0.05	0.89±0.04	0.84±0.06	0.86±0.07
7	VAE	0.82±0.06	0.80±0.06	0.82±0.06	0.88±0.06	0.88±0.09	0.78±0.06
8	ENN	0.85±0.04	0.82±0.08	0.83±0.04	0.88±0.06	0.89±0.05	0.79±0.12
9	MWMOTE	0.81±0.06	0.81±0.06	0.82±0.06	0.88±0.04	0.84±0.06	0.82±0.07
10	GAN (Vanilla)	0.81±0.06	0.80±0.06	0.81±0.06	0.88±0.05	0.86±0.07	0.77±0.07

Note that Table 6 reports variability pooled across folds and datasets, whereas Table 7 reports variability across datasets only, after averaging out fold-level noise; the same distinction applies to Tables 11 and 12. The results, presented in Table 7, show that the variability across datasets remains low for most methods (std $\approx$ 0.02–0.06), indicating good generalization.

For comparison, the baseline model shows significantly lower performance and much higher variability. Its PR-AUC reaches only 0.74, with a global standard deviation of 0.19, and similarly high variability is observed for F1-score (0.60 $\pm$ 0.28) and recall (0.57 $\pm$ 0.30). This indicates unstable behavior and a strong sensitivity to data partitioning. But accross datasets only, the baseline appears relatively stable with PR-AUC = 0.74 $\pm$ 0.04. This highlights that inter-dataset variability alone may underestimate the true variability of the model.

Overall, balancing methods not only improve predictive performance but also substantially reduce variability compared to the baseline. Hybrid approaches, particularly SMOTE+ENN, provide the best trade-off between performance and stability, while methods such as MWMOTE and GAN remain more sensitive to dataset characteristics.

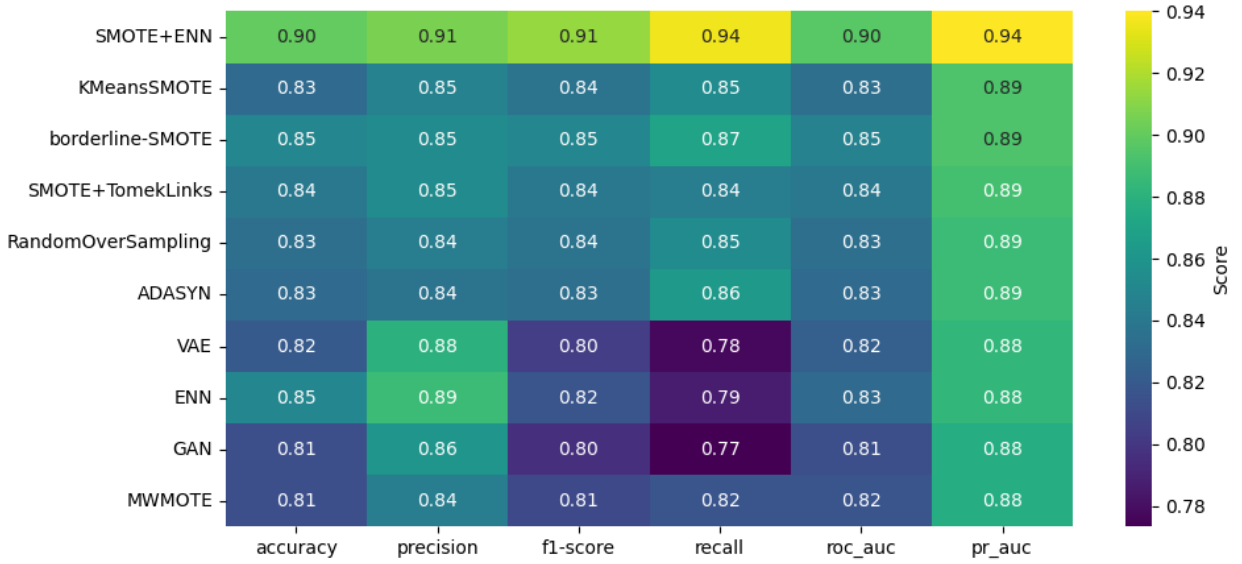


Figure 2. Heatmap of average performance metrics for the top-10 balancing approaches ranked by PR–AUC across the three transcriptomic datasets using Random Forest Classifier.

In general, the top-10 ranking shown in Table 7, Table 6 and Figure 2 is focused on hybrid and oversampling-based approaches, featuring a small portion of generative and undersampling ones. All top-performing methods significantly outperformed the baseline, demonstrating the necessity of explicit tackling imbalance for genomic autism classification.

4.3. Statistical Significance of Performance Differences

To assess whether the observed differences in PR–AUC between balancing methods reflect genuine effects rather than random cross-validation variability, pairwise Welch's t-tests (which do not assume equal variances across methods) were performed on the per-fold PR–AUC scores obtained with the Random Forest classifier, separately for each of the three datasets. Table 8 reports p-values comparing the baseline (no balancing) against each top-10 balancing method.

Table 8. Welch's t-test p-values for baseline-vs-method PR–AUC comparisons using Random Forest (10-fold cross-validation). Values below 0.05 (marked *) indicate a statistically significant difference at the 5% level.

Baseline vs.	GSE6575	GSE18123	GSE25507
SMOTE+ENN	0.084	0.013*	0.0003*
KMeansSMOTE	0.123	0.024*	0.009*
Borderline-SMOTE	0.102	0.019*	0.024*
SMOTE+Tomek Links	0.195	0.015*	0.017*
Random Oversampling	0.141	0.020*	0.025*
ADASYN	0.123	0.030*	0.022*
VAE	0.106	0.021*	0.069
ENN	0.499	0.159	0.0004*
MWMOTE	0.181	0.026*	0.044*
GAN (Vanilla)	0.173	0.026*	0.066

On GSE18123 and GSE25507, most top-10 methods significantly outperform the baseline ($p < 0.05$), consistent with the ranking reported in Section 4. On GSE6575, the smallest dataset (56 samples, 21 in the minority class), none of the comparisons reach significance at the 5% level, although every point estimate favors balancing over the baseline; this is consistent with reduced statistical power on a very small cohort rather than an absence of effect, and we report it as such rather than overstating significance. We additionally compared SMOTE+ENN against plain oversampling (Random Oversampling, SMOTE, Borderline-SMOTE) to test whether the neighborhood-cleaning step adds a genuine benefit beyond oversampling alone: this additional benefit is significant on GSE25507 ($p < 0.01$) but not on GSE6575 or GSE18123 ($p > 0.4$), indicating that this specific advantage, while directionally consistent, is not uniformly significant across all three datasets.

4.4. Metric Sensitivity and Ranking Behavior

To evaluate the stability of the method ranking in different evaluation metrics and under class imbalance, several metrics are inspected. We use PR–AUC as the main ranking metric since it aggregates precision and recall. The ROC–AUC is provided because it is independent of a threshold and can be viewed as class separability, while precision and recall specifically describe false positive and false negative properties, respectively.

Comparing rankings based on PR–AUC, ROC–AUC, accuracy, and recall reveals distinct rankings of the methods. Although SMOTE+ENN is consistently ranked among the best-performing methods, other methods vary in their rankings depending on the evaluation criterion. Overall, ranking methods, including hybrids and oversampling, yield stable ranks, whereas generative, optimization-level, and ensemble-based approaches are sensitive to the choice of metric. Overall, the stability of top-performing approaches across different metrics is reported in Table 9.

Table 9. Stability of top-10 performing balancing approaches across different evaluation metrics. **"Yes"** indicates that the method appears in the top-10 ranking under the corresponding metric using Random Forest Classifier.

Method	Top-10(PR–AUC)	Top-10 (ROC–AUC)	Top-10 (Accuracy)	Top-10 (Recall)
SMOTE+ENN	Yes	Yes	Yes	Yes
Borderline-SMOTE	Yes	Yes	Yes	Yes
KMeansSMOTE	Yes	Yes	Yes	Yes
SMOTE+Tomek Links	Yes	Yes	Yes	Yes
Random Oversampling	Yes	Yes	Yes	Yes
ADASYN	Yes	Yes	Yes	Yes
VAE	Yes	Yes	Yes	No
ENN	Yes	Yes	No	No
MWMOTE	Yes	Yes	No	No
GAN (Vanilla)	Yes	No	No	No

Recall-based ranking discriminates methods with high recall in the minority class, while accuracy and ROC–AUC rank methods differently even if their recall is lower compared to others. These findings suggest that the ranking of methods depends on biometric choice and further substantiates the position of PR–AUC as a balanced and discriminative performance comparison metric for imbalanced genomic classification. Examining Table 6, most top-10 methods keep precision and recall close to one another (e.g., SMOTE+ENN: precision 0.91 vs. recall 0.94; KMeansSMOTE: 0.85 vs. 0.85), suggesting a relatively balanced trade-off; the main exceptions are VAE and GAN, which achieve comparatively high precision (0.88 and 0.86) but low recall (0.78 and.077), thus being more conservative when identifying ASD cases. In a clinical screening context, this recall gap is potentially more consequential than the corresponding precision gain, since missed ASD cases (false negatives) delay intervention, which is a further argument for preferring hybrid, higher-recall methods such as SMOTE+ENN over generative approaches in this application.

4.5. Cross-Dataset Consistency

To explore whether the top-10 performing balancing approaches are stable across multiple genomic datasets, the overlap of methods in the top-10 PR–AUC ranking for GSE18123, GSE6575, and GSE25507 is used to evaluate consistency.

Table 10. Presence of top-10 performing balancing approaches in the top-10 PR–AUC ranking across the three datasets (GSE18123, GSE6575, and GSE25507) using Random Forest Classifier. **"Yes"** indicates that the method appears in the top-10 for the corresponding dataset.

Method	Top-10(GSE18123)	Top-10(GSE6575)	Top-10(GSE25507)
SMOTE+ENN	Yes	Yes	Yes
Borderline-SMOTE	Yes	Yes	Yes
KMeansSMOTE	Yes	Yes	Yes
Random Oversampling	Yes	Yes	Yes
ADASYN	Yes	Yes	Yes
SMOTE+Tomek Links	Yes	Yes	No
VAE	Yes	No	Yes
ENN	No	Yes	Yes
MWMOTE	Yes	No	No
GAN (Vanilla)	No	Yes	No

In general, SMOTE+ENN shows the best cross-dataset consistency and ranks among the top-10 approaches across all three datasets. A few oversampling schemes, such as Borderline-SMOTE, KMeansSMOTE, Random Oversampling and ADASYN always appear in the best-performing methods over datasets, showing their excellent generalization to diverse benchmark problems. The stability of top-performing methods is shown in Table 10, which lists, for each method, how many datasets were present in its dataset-specific top-10 rankings.

4.6. Classifier Robustness Analysis

To assess the stability of the estimated rankings of balancing approaches using other classifiers, we performed further experiments with Support Vector Machine (SVM) and Logistic Regression. Tables 11 and 12 report the top 10 balancing methods ranked by PR–AUC using SVM and Logistic Regression, respectively. Results are expressed as mean ± standard deviation aggregated across cross-validation folds and datasets.

Table 11. Top-10 balancing approaches ranked by PR–AUC using SVM (mean ± standard deviation across datasets and cross-validation folds).

Rank	Method	Accuracy	F1-score	ROC–AUC	PR–AUC	Precision	Recall
1	**SMOTE+ENN**	**0.95±0.08**	**0.95±0.08**	**0.95±0.07**	**0.97±0.04**	**0.98±0.06**	**0.93±0.12**
2	ENN	0.90±0.11	0.89±0.14	0.90±0.12	0.92±0.09	0.91±0.13	0.89±0.19
3	borderline-SMOTE	0.88±0.12	0.89±0.12	0.88±0.12	0.91±0.09	0.87±0.14	0.92±0.11
4	KMeansSMOTE	0.85±0.12	0.85±0.12	0.86±0.11	0.91±0.08	0.89±0.13	0.83±0.15
5	SMOTE	0.86±0.11	0.87±0.11	0.86±0.11	0.90±0.08	0.87±0.13	0.88±0.13
6	SVM-SMOTE	0.86±0.10	0.86±0.10	0.86±0.10	0.90±0.07	0.86±0.13	0.89±0.13
7	SMOTE+TomekLinks	0.85±0.11	0.86±0.11	0.86±0.10	0.90±0.08	0.87±0.14	0.86±0.13
8	ADASYN	0.86±0.13	0.86±0.12	0.86±0.12	0.90±0.09	0.85±0.14	0.88±0.16
9	MWMOTE	0.86±0.12	0.86±0.12	0.86±0.12	0.90±0.09	0.86±0.14	0.89±0.14
10	NCL	0.85±0.12	0.83±0.14	0.84±0.13	0.90±0.10	0.88±0.16	0.82±0.19

The SVM results highlight the importance of boundary smoothness and effective noise handling in margin-based classification. Among the evaluated methods, SMOTE+ENN achieves the highest PR–AUC. Its performance also remains relatively stable across cross-validation folds, with moderate variability (std $\approx$ 0.04–0.12). Therefore, Combining synthetic samples with neighborhood cleaning appears to improve class separation. Borderline-SMOTE and KMeans-SMOTE also perform well, although they do not reach the same overall performance. These results suggest that SVM performance depends strongly on class overlap and the structure of the decision boundary.

Table 12. Top-10 balancing approaches ranked by PR–AUC using Logistic Regression (mean ± standard deviation across datasets and cross-validation folds).

Rank	Method	Accuracy	F1-score	ROC–AUC	PR–AUC	Precision	Recall
1	**SMOTE+ENN**	**0.98±0.04**	**0.98±0.03**	**0.98±0.05**	**0.99±0.03**	**0.98±0.05**	**0.99±0.05**
2	ENN	0.92±0.11	0.90±0.13	0.92±0.11	0.95±0.08	0.96±0.11	0.88±0.18
3	borderline-SMOTE	0.89±0.11	0.90±0.10	0.89±0.11	0.92±0.08	0.86±0.13	0.95±0.08
4	NCL	0.89±0.10	0.88±0.13	0.88±0.11	0.92±0.09	0.91±0.14	0.87±0.18
5	ADASYN	0.88±0.10	0.89±0.09	0.88±0.10	0.91±0.07	0.86±0.12	0.95±0.10
6	SMOTE+TomekLinks	0.88±0.10	0.88±0.09	0.88±0.10	0.91±0.07	0.86±0.13	0.92±0.11
7	SMOTE	0.87±0.10	0.88±0.09	0.88±0.11	0.91±0.08	0.86±0.13	0.91±0.11
8	GAN	0.87±0.10	0.87±0.10	0.87±0.11	0.91±0.08	0.87±0.13	0.89±0.11
9	KMeansSMOTE	0.86±0.11	0.86±0.11	0.86±0.11	0.91±0.07	0.87±0.13	0.88±0.15
10	SVM-SMOTE	0.87±0.10	0.88±0.09	0.87±0.10	0.91±0.07	0.86±0.12	0.91±0.12

As shown in Table 12, the best-performing techniques for Logistic Regression are those that improve linear separability and reduce noise. SMOTE+ENN emerges as the best method, which achieves the highest PR–AUC values with low variability (std $\approx$ 0.03–0.05). Undersampling techniques, particularly ENN and NCL, also remain competitive with the oversampling methods.

To quantify the top-10 ranking is preserved across classifiers (raised as a specific point by one referee), we computed Spearman's rank correlation [53] and Kendall's τ [54] between the top-10 PR–AUC rankings of Tables 6, 11, and 12, restricted to methods common to each pair of classifiers. Random Forest and SVM rankings show moderate positive agreement (Spearman's $\rho = 0.61$, Kendall's $\tau = 0.52$, $n = 7$ common methods), SVM and Logistic Regression rankings show weaker agreement (Spearman's $\rho = 0.47$, Kendall's $\tau = 0.28$, $n = 9$), and Random Forest and Logistic Regression show the weakest agreement (Spearman's $\rho = 0.21$, Kendall's $\tau = 0.14$, $n = 7$). This indicates that while the identity of the best method (SMOTE+ENN) is highly stable across classifiers.

These results demonstrate that the efficiency of balancing methods is largely, though not perfectly, independent of the classifier. Especially, hybrid approaches such as SMOTE+ENN consistently provide high performance and stability across different classifiers.

4.7. Computational Cost Analysis

Execution time was measured for the full pipeline on each dataset and is reported as mean $\pm$ standard deviation across datasets.

Table 13. Average execution time of balancing methods across classifiers, expressed in seconds (mean ± standard deviation across datasets).

Method	RF Time	SVM Time	LR Time
SPE	0.39 ± 0.21	0.45 ± 0.10	0.42 ± 0.09
FOCAL-LOSS	0.40 ± 0.21	0.52 ± 0.28	0.48 ± 0.21
Class-Balanced	0.43 ± 0.01	0.50 ± 0.12	0.61 ± 0.21
DCL	0.47 ± 0.03	0.65 ± 0.14	0.68 ± 0.20
NCL	1.16 ± 0.27	0.20 ± 0.06	0.12 ± 0.02
ENN	1.24 ± 0.38	0.19 ± 0.05	0.11 ± 0.02
RUSBoost	1.27 ± 0.78	1.56 ± 0.42	1.20 ± 0.31
OSS	1.31 ± 0.57	0.19 ± 0.07	0.12 ± 0.03
SMOTEBoost	1.36 ± 0.47	1.97 ± 0.33	1.44 ± 0.30
ClusterCentroids	1.36 ± 0.62	0.34 ± 0.26	0.13 ± 0.01
RandomOverSampling	1.49 ± 0.45	0.30 ± 0.13	0.11 ± 0.01
borderline-SMOTE	1.50 ± 0.58	0.22 ± 0.06	0.11 ± 0.01
SMOTE+ENN	1.52 ± 0.70	0.18 ± 0.05	0.12 ± 0.02
baseline	1.57 ± 0.36	0.41 ± 0.06	0.25 ± 0.05
ADASYN	1.57 ± 0.81	0.24 ± 0.09	0.13 ± 0.04
SMOTE	1.61 ± 0.67	0.30 ± 0.06	0.21 ± 0.03
TomekLinks	1.65 ± 0.85	0.22 ± 0.06	0.11 ± 0.01
Safe-Level-SMOTE	1.74 ± 0.93	0.21 ± 0.07	0.11 ± 0.02
MWMOTE	1.82 ± 0.98	0.35 ± 0.13	0.17 ± 0.04
NearMiss	1.83 ± 1.46	0.29 ± 0.17	0.14 ± 0.02
GAN	1.92 ± 0.63	0.61 ± 0.17	0.55 ± 0.19
KMeansSMOTE	2.02 ± 1.38	0.45 ± 0.17	0.24 ± 0.03
SVM-SMOTE	2.02 ± 1.39	0.23 ± 0.03	0.13 ± 0.01
SMOTE+TomekLinks	2.23 ± 1.49	0.23 ± 0.08	0.10 ± 0.01
VAE	2.31 ± 1.50	1.00 ± 0.81	0.39 ± 0.12
Representation Learning + Oversampling	3.51 ± 1.47	1.96 ± 1.10	1.65 ± 0.76
TabDDPM	5.32 ± 3.87	2.68 ± 1.14	3.00 ± 0.13
WGAN	7.91 ± 2.14	9.64 ± 5.86	6.57 ± 0.33
EasyEnsemble	13.56 ± 5.84	11.51 ± 0.94	11.20 ± 1.80
CTGAN	41.27 ± 26.23	42.74 ± 26.21	32.88 ± 15.28

Based on the experimental details described in subsection 3.10, as illustrated in Table 13, the simplest methods, such as SPE, Focal Loss, Class-Balanced, and DCL, are the fastest, with execution times consistently below on second. Traditional resampling methods require slightly more time, with some variation depending on the classifier.

Hybrid approaches, especially SMOTE+ENN, introduce additional overhead but remain computationally efficient while providing strong predictive performance.

Advanced and generative approaches such as VAE, TabDDPM, WGAN, CTGAN are significantly more time-consuming. CTGAN shows the highest execution time and considerable variability across all classifiers.

Overall, these results suggest a well-defined trade-off between computational cost and model complexity, with hybrid approaches offering a practical compromise between efficiency and performance. Although this benchmark focuses on high-dimensional, small-sample datasets (56–146 samples), the relative cost ranking observed here is expected to persist qualitatively at larger scale, since it is driven primarily by algorithmic complexity rather than sample size alone: simple resampling and loss-level methods scale close to linearly with the number of samples and features, whereas generative approaches (VAE, GAN, TabDDPM, CTGAN) are dominated by network training cost, which depends more on model capacity and number of training epochs than on dataset size. Consequently, on larger cohorts the absolute cost gap between simple and generative methods would likely widen rather than narrow, though this expectation remains to be empirically confirmed on larger datasets.

5. Discussion

This work provides a comprehensive comparison of balancing approaches for ASD classification using genomic data under realistic constraints: high dimensionality, limited sample sizes, and class imbalance. The results consistently demonstrate that balancing is imperative in this context: the baseline (absence of balancing) consistently produces weaker minority-class performance, and exhibits higher variability across cross-validation folds, and enhancements in performance predominantly result from strategies that modify the training distribution to improve minority separability without increasing noise while also improving the stability of the models.

A key observation is the dominance of hybrid approaches, particularly SMOTE+ENN, which is the top-performing when ranked by PR-AUC using Random Forest, SVM, and Logistic Regression. This cross-model consistency indicates that the gains are not an artifact of a particular classifier but instead reflect robust progress in the effective geometry of the training data — that is, minority oversampling combined with removal of the ambiguous or crossing instances. The strong presence of other cleaning- and boundary-based approaches in the top rankings, even without direct interpolation, suggests that for high-dimensional gene expression profiles, controlling overlap and noise in local neighborhoods is critical, since noise can otherwise reinforce spurious patterns; importantly, SMOTE+ENN achieves this while maintaining moderate computational cost, making it effective in both performance and efficiency terms.

In comparison, naive generative models and optimization-based techniques demonstrate more variable behavior in both predictive performance and computational cost, particularly among the costlier generative architectures (CTGAN, WGAN, TabDDPM; see Table 13); by contrast, the two generative methods that reach the top-10 by PR–AUC, VAE and GAN, show PR–AUC variability comparable to the other top-10 methods (Table 6). While generative approaches can, in principle, capture global structure, their benefits are not as clear in small-sample transcriptomic spaces, where distribution learning can be unstable and generated samples are unlikely to enhance class separability as much as neighborhood-cleaning hybrids. Similarly, loss-level methods are conceptually attractive, but their gains are highly dependent on the learning setup; in our experiments, the most consistent gains come from approaches that explicitly alter sample composition and increase purity near class boundaries, whereas more complex generative approaches (e.g., CTGAN) introduce substantial computational overhead and high variability in execution time.

Further metrics-based assessments serve to prioritize the methodological decision that PR–AUC be chosen as the metric for ranking. Because PR–AUC reflects the precision–recall trade-off directly, it is more sensitive to minority-class performance than accuracy and can reveal differences that ROC-AUC may hide under imbalance. The patterns of relative ranking across metrics that are observed are then to be expected: some methods produce high

separability (high ROC–AUC) but appear suboptimal based on precision or recall, as expected given PR–AUC's greater sensitivity to minority-class detection. From an applied perspective, this is particularly important because ASD screening and genomic decision support generally require rigorous control of false positives as well as false negatives, making precision and recall metrics more useful than overall accuracy.

The consistency across datasets reinforces the practical implication of this benchmark: despite variations in cohort composition and imbalance ratio across transcriptomic datasets, a small set of strategies consistently achieves top performance, with SMOTE+ENN ranking as the most stable and exhibiting low inter-dataset variability. This shows that practitioners have the option to prioritise hybrid and boundary-inclined methods as default solutions on imbalanced genomic classification tasks, regarding more complex generative or optimization-level methods as secondary choices that necessitate additional tuning and validation.

Finally, several limitations should be acknowledged. The benchmark focuses on the three microarray datasets (GPL570) and evaluates a fixed experimental design; different platforms, omics modalities, feature-selection regimes, or deep learning pipelines may shift the relative performance of some strategy families. Using a single platform (GPL570) reduces purely technical (platform-driven) variability and is a deliberate design choice, but it also means that dataset-to-dataset differences observed here reflect cohort- and study-level variation rather than cross-platform batch effects; we did not apply explicit batch-correction methods (e.g., ComBat) since the three datasets were analyzed independently rather than pooled, and no independent, external hold-out cohort was used beyond the cross-validation folds within each dataset. Explicit batch correction and external validation on independent cohorts, ideally including RNA-seq data, are natural extensions of this benchmark. In addition, balancing is evaluated in the context of supervised ASD/control classification and does not directly address the biological interpretability of selected genes, nor whether specific balancing methods are differentially well-suited to particular ASD-related gene expression patterns. A controlled simulation varying dataset characteristics such as imbalance ratio, sample size, and noise level could further delineate the conditions favoring each method family, but lies beyond the empirical, dataset-driven scope adopted here. Classifier hyperparameters were also kept at default values throughout, deliberately, to isolate the effect of the balancing strategy, rather than jointly tuned with each balancing method. However, based on the joint multi-dataset evaluation, metric-aware ranking, and cross-classifier robustness check, we suggest that these results may be exploited as a useful reference for choosing balancing approaches in ASD classification and other high-dimensional biomedical tasks. Explicit batch correction and joint tuning of classifier and balancing-method hyperparameters remain natural extensions of this benchmark.

6. Conclusion

This work provides a comprehensive evaluation of 30 data-balancing approaches for autism classification, assessing their effectiveness, stability, and computational efficiency across three transcriptomic datasets. The results show that explicit handling of class imbalance is necessary for consistent, reliable classification in high-dimensional genomic settings.

Hybrid balancing techniques, specially SMOTE+ENN, consistently yield the strongest predictive power under imbalance-aware metrics such as PR–AUC and Recall, and do so robustly across Random Forest, SVM, and Logistic Regression. Oversampling-based approaches also perform strongly and stably, while undersampling, optimization-level, and ensemble-based methods yield more modest or inconsistent gains; generative and representation-learning methods are the least consistent, being sensitive to dataset properties and training stability, and carry higher computational cost.

Method ranking depends substantially on the evaluation metric, with PR–AUC proving the most informative for minority-class detection. Cross-dataset and cross-classifier analyses show that only a subset of balancing strategies generalizes reliably across cohorts and learning paradigms, emphasizing the need for robustness characterization in machine learning studies in biomedicine and the importance of evaluating both performance and variability.

In summary, this work offers practical guidelines for the selection of efficient and robust balancing approaches in genomic autism classification. The findings validate that hybrid and boundary-aware oversampling methods represent reliable default choices for imbalanced transcriptomic datasets. This framework may be expanded in

future studies to other omics types and to larger, more heterogeneous cohorts such as RNA-seq datasets, and could incorporate explainable AI methods for better biological interpretability and clinical applicability, as well as more detailed analyses of scalability and computational efficiency in large-scale settings.

Code and Data Availability

The Python implementation used in this study is available from the corresponding author upon reasonable request. The three transcriptomic datasets (GSE25507, GSE6575, GSE18123) are publicly available from the NCBI Gene Expression Omnibus.

REFERENCES

1. O. Ousley and T. Cermak, *Autism Spectrum Disorder: Defining Dimensions and Subgroups*, Curr. Dev. Disord. Rep., vol. 1, no. 1, pp. 20–28, 2014.
2. C. Lord et al., *Autism spectrum disorder*, Nat. Rev. Dis. Primers, vol. 6, no. 1, p. 5, 2020.
3. K. A. Shaw et al., *Prevalence and Early Identification of Autism Spectrum Disorder*, MMWR Surveill. Summ., vol. 74, no. 2, pp. 1–22, 2025.
4. S. Qiu et al., *Genetics of autism spectrum disorder: an umbrella review*, Translational Psychiatry, vol. 12, p. 249, 2022.
5. A. Gogate et al., *The genetic landscape of autism spectrum disorder in an ancestrally diverse cohort*, npj Genomic Medicine, vol. 9, no. 1, p. 62, 2024.
6. R. Edgar, M. Domrachev, and A. E. Lash, *Gene Expression Omnibus: NCBI gene expression and hybridization array data repository*, Nucleic Acids Research, vol. 30, no. 1, pp. 207–210, 2002.
7. A. Fernandez et al., *SMOTE for Learning from Imbalanced Data: Progress and Challenges*, JAIR, vol. 61, pp. 863–905, 2018.
8. G. Haixiang et al., *Learning from class-imbalanced data: Review of methods and applications*, Expert Systems with Applications, vol. 73, pp. 220–239, 2017.
9. T. Saito and M. Rehmsmeier, *The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets*, PLoS ONE, vol. 10, no. 3, e0118432, 2015.
10. T. Watthaisong, K. Sunat, and N. Muangkote, *Comparative Evaluation of Imbalanced Data Management Techniques for Solving Classification Problems on Imbalanced Datasets*, Stat., Optim. Inf. Comput., vol. 12, pp. 547–570, 2024.
11. A. Maigari, Z. Zainol, and C. Xinying, *Multi-modal Stacked Ensemble Model for Breast Cancer Prognosis Prediction*, Stat., Optim. Inf. Comput., vol. 13, no. 3, pp. 1013–1034, 2024.
12. Carvalho, M., Pinho, A.J. & Brás, S., *Resampling approaches to handle class imbalance: a review from a data perspectives*, Journal of Big Data, vol. 12, no. 71, 2025.
13. W. Chen et al., *A survey on imbalanced learning: latest research, applications and future directions*, Artif. Intell. Rev., vol. 57, no. 6, p. 137, 2024.
14. M. Salmi et al., *Handling imbalanced medical datasets: review of a decade of research*, Artif. Intell. Rev., vol. 57, no. 10, p. 273, 2024.
15. N. V. Chawla et al., *SMOTE: Synthetic Minority Over-sampling Technique*, JAIR, vol. 16, pp. 321–357, 2002.
16. A. G. Udu, M. T. Salman, M. K. Ghalati, A. Lecchini-Visintini, D. R. Siddle and H. Dong, *Emerging SMOTE and GAN Variants for Data Augmentation in Imbalance Machine Learning Tasks: A Review*, IEEE Access, vol. 13, pp. 113838-113853, 2025
17. R. Blagus and L. Lusa, *SMOTE for high-dimensional class-imbalanced data*, BMC Bioinformatics, vol. 14, p. 106, 2013.
18. Z. Liu et al., *Self-paced Ensemble for Highly Imbalanced Massive Data Classification*, in Proc. ICDE, pp. 841–852, 2020.
19. X.-Y. Liu, J. Wu, and Z.-H. Zhou, *Exploratory Undersampling for Class-Imbalance Learning*, IEEE Trans. Syst., Man, Cybern. B, vol. 39, no. 2, pp. 539–550, 2009.
20. L. Xu et al., *Modeling Tabular Data using Conditional GAN*, arXiv:1907.00503, 2019.
21. A. Kotelnikov et al., *TabDDPM: Modelling Tabular Data with Diffusion Models*, arXiv:2209.15421, 2022.
22. T.-Y. Lin et al., *Focal Loss for Dense Object Detection*, IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 2, pp. 318–327, 2020.
23. Y. Cui et al., *Class-Balanced Loss Based on Effective Number of Samples*, in Proc. CVPR, 2019, pp. 9260–9269, 2019.
24. Y. Bengio, A. Courville, and P. Vincent, *Representation Learning: A Review and New Perspectives*, IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 8, pp. 1798–1828, 2013.
25. M. Garbulowski, K. Smolinska, K. Diamanti, G. Pan, K. Maqbool, L. Feuk, and J. Komorowski, *Interpretable Machine Learning Reveals Dissimilarities Between Subtypes of Autism Spectrum Disorder*, Front. Genet., vol. 12, p. 618277, 2021.
26. T. Barrett et al., *NCBI GEO: archive for functional genomics data sets—update*, Nucleic Acids Research, vol. 41, no. D1, pp. D991–D995, 2012.
27. Y. Saeys, I. Inza, and P. Larrañaga, *A review of feature selection techniques in bioinformatics*, Bioinformatics, vol. 23, no. 19, pp. 2507–2517, 2007.
28. C. Ding and H. Peng, *Minimum redundancy feature selection from microarray gene expression data*, Journal of Bioinformatics and Computational Biology, vol. 3, no. 2, pp. 185–205, 2005.
29. G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, *A study of the behavior of several methods for balancing machine learning training data*, SIGKDD Explor. Newsl., vol. 6, no. 1, pp. 20–29, 2004.

30. H. M. Nguyen, E. W. Cooper, and K. Kamei, *Borderline over-sampling for imbalanced data classification*, International Journal of Knowledge Engineering and Soft Data Paradigms, vol. 3, no. 1, pp. 4–21, 2009.
31. S. Barua, M. M. Islam, X. Yao, and K. Murase, *MWMOTE–majority weighted minority oversampling technique for imbalanced data set learning*, IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 2, pp. 405–425, 2014.
32. C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, *Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem*, in Proc. Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD), pp. 475–482, 2009.
33. H. Han, W.-Y. Wang, and B.-H. Mao, *Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning*, in Advances in Intelligent Computing, pp. 878–887, 2005.
34. G. Douzas, F. Bacao, and F. Last, *Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE*, Information Sciences, vol. 465, pp. 1–20, 2018.
35. H. He, Y. Bai, E. A. Garcia, and S. Li, *ADASYN: Adaptive synthetic sampling approach for imbalanced learning*, in Proc. IEEE Int. Joint Conf. on Neural Networks (IJCNN), 2008.
36. I. Mani and I. Zhang, *kNN approach to unbalanced data distributions: a case study involving information extraction*, in Proc. Workshop on Learning from Imbalanced Datasets, 2003.
37. I. Tomek, *Two modifications of CNN*, IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-6, no. 11, pp. 769–772, 1976.
38. S.-J. Yen and Y.-S. Lee, *Cluster-based under-sampling approaches for imbalanced data distributions*, Expert Systems with Applications, vol. 36, no. 3, pp. 5718–5727, 2009.
39. D. L. Wilson, *Asymptotic Properties of Nearest Neighbor Rules Using Edited Data*, IEEE Trans. Syst., Man, Cybern., vol. SMC-2, no. 3, pp. 408–421, 1972.
40. J. Laurikkala, *Improving Identification of Difficult Small Classes by Balancing Class Distribution*, in Artificial Intelligence in Medicine, 2001, pp. 63–66.
41. M. Kubat, *Addressing the Curse of Imbalanced Training Sets: One-Sided Selection*, in Proc. 14th International Conference on Machine Learning (ICML), 2000.
42. G. E. de A. P. A. Batista, A. L. C. Bazzan, and M. C. Monard, *Balancing training data for automated annotation of keywords: a case study*, Revista Tecnologia da Informação, vol. 3, no. 2, pp. 15–20, 2003.
43. N. V. Chawla et al., *SMOTEBoost: Improving Prediction of the Minority Class in Boosting*, in PKDD, pp. 107–119, 2003.
44. C. Seiffert et al., *RUSBoost: A Hybrid Approach to Alleviating Class Imbalance*, IEEE Trans. Syst., Man, Cybern. A, vol. 40, no. 1, pp. 185–197, 2010.
45. D. P. Kingma and M. Welling, *Auto-Encoding Variational Bayes*, arXiv:1312.6114, 2013.
46. I. J. Goodfellow et al., *Generative Adversarial Networks*, arXiv:1406.2661, 2014.
47. M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein GAN*, in Proc. International Conference on Machine Learning (ICML), 2017.
48. L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, *Modeling Tabular Data using Conditional GAN*, in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2019.
49. A. Kotelnikov et al., *TabDDPM: Modelling Tabular Data with Diffusion Models*, in Proc. International Conference on Learning Representations (ICLR), 2023.
50. Y. Wang, W. Gan, J. Yang, W. Wu, and J. Yan, *Dynamic Curriculum Learning for Imbalanced Data Classification*, in Proc. ICCV, pp. 5016–5025, 2019.
51. F. Pedregosa et al., *Scikit-learn: Machine learning in Python*, Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
52. G. Lemaître, F. Nogueira, and C. K. Aridas, *Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning*, Journal of Machine Learning Research, vol. 18, no. 17, pp. 1–5, 2017.
53. C. Spearman, *The Proof and Measurement of Association between Two Things*, Amer. J. Psychol., vol. 15, no. 1, pp. 72–101, 1904.
54. M. G. Kendall, *A New Measure of Rank Correlation*, Biometrika, vol. 30, no. 1/2, pp. 81–93, 1938.