International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Comprehensive Review: SMOTE-Based Oversampling Methods for Imbalanced Classification Techniques, Evaluation, and Result Comparisons

Bhushankumar Nemade[1*], Vinayak Bharadi[2], Sujata S. Alegavi[3], Bijith Marakarkandy[4]

Submitted: 23/04/2023 **Revised**: 28/06/2023 **Accepted**: 14/07/2023

Abstract: This study conducts a comparative analysis of different SMOTE variants, assessing their effectiveness in diverse domains. By synthesizing the findings, it provides insights into the strengths, limitations, and future directions of oversampling methods, with a specific emphasis on SMOTE-based techniques. Through an in-depth survey of research papers and articles, it explores the principles, techniques, evaluation methodologies, and challenges associated with oversampling. This review serves as a valuable resource for researchers and practitioners, aiding informed decision-making and advancements in imbalanced classification. The proposed system is composed of six integral parts: real-time data collection, data cleaning, and feature extraction, handling of imbalanced data using various methods, selection of preferred classifiers, and the utilization of a voting principle for optimal prediction. In conclusion, the system employs a multi-model classification approach to enhance the efficiency of the aquaponics ecosystem. By leveraging the power of optimal prediction based on voting, the system evaluates the performance of four classifiers using benchmark parameters such as accuracy, time, recall, and Kappa. Through this evaluation, it identifies XGBoost and Random Forest as the most effective classifiers, based on the voting principle.

Keywords: *Imbalanced classification, class imbalance, oversampling methods, Synthetic Minority Over-sampling Technique (SMOTE), imbalanced datasets, minority class, synthetic samples, comparative analysis, evaluation methodologies, challenges, Borderline-SMOTE, CCR-SMOTE-LR, SMOTE-ENC, Adaptive-SMOTE, SMOTE-Tomek, Safe-Level SMOTE, SMOTE-Boost, LN-SMOTE, Bagging-SMOTE, SVM-SMOTE*

1. Introduction

The imbalanced datasets provides a fundamental understanding of data scenarios where there exists a substantial disparity in class distributions, posing notable challenges in the field of machine learning. In classification problems, imbalanced datasets occur when the number of instances belonging to one class significantly outweighs the number of instances in the other classes. This class imbalance introduces inherent biases and can lead to suboptimal performance for minority classes, as learning algorithms tend to be biased towards the majority class. Imbalanced datasets are prevalent in various domains, including fraud detection, medical diagnosis, text classification, and anomaly detection. The imbalance may stem from various factors such as the rarity of certain events or the uneven distribution of instances in the real-world population. Consequently, it is crucial to comprehend the implications of imbalanced datasets and devise strategies to handle this issue effectively. Understanding the background of imbalanced datasets involves assessing the extent of class imbalance within the dataset, evaluating the potential impact on the learning process, and recognizing the challenges associated with accurately classifying the minority class. It also necessitates acknowledging the limitations of conventional classification algorithms when applied to imbalanced data and the need for specialized techniques to address this issue. The background on imbalanced datasets sets the foundation for exploring solutions that mitigate the challenges posed by class imbalance.

One prominent approach is the Synthetic Minority Over-sampling Technique (SMOTE), which addresses the class imbalance problem by generating synthetic samples for the minority class based on the characteristics of existing instances. By gaining insights into the background of imbalanced datasets, researchers and practitioners can

1Mukesh Patel School of Technology Management & Engineering, NMIMS University, India,
3Thakur College of Engineering and Technology, Mumbai University, India,
2Finolex Academy of Management and Technology, Mumbai University, India,
4S.P. Mandali's Prin. L. N. Welingkar Institute of Management Development & Research (WeSchool), Mumbai, India.

develop a comprehensive perspective on the unique characteristics and difficulties inherent in imbalanced classification problems.

This understanding serves as a crucial starting point for exploring and developing effective techniques, such as SMOTE, to alleviate the impact of class imbalance and enhance the performance of classification models in such challenging scenarios.

Study Objectives:

1. To examine the different SMOTE-based oversampling methods employed in the context of imbalanced classification. This involves investigating the various techniques and variations of SMOTE that have been proposed in the literature to address class imbalance.

2. To explore the underlying principles and methodologies of SMOTE-based oversampling. This includes understanding the rationale behind SMOTE, its synthetic sample generation process, and how it aims to rebalance the class distribution in imbalanced datasets.

3. To evaluate and compare the performance of different SMOTE-based methods. This involves conducting a comprehensive analysis of empirical studies, research papers, and conference proceedings to assess the effectiveness of different SMOTE variants in improving classification accuracy for minority classes.

4. To investigate the evaluation methodologies used in assessing the performance of SMOTE-based techniques. This includes examining the metrics, experimental setups, and statistical measures employed to evaluate the impact of SMOTE on classification models.

5. To identify the challenges and limitations associated with SMOTE-based oversampling. This involves discussing the potential drawbacks, assumptions, and potential pitfalls of using SMOTE in imbalanced classification tasks.

6. To provide insights into the applicability and suitability of SMOTE-based oversampling methods across diverse domains. This includes examining case studies, applications, and experimental results from various fields to understand the generalizability of SMOTE techniques and their performance in real-world scenarios.

There are seven major sections of the paper. The "Introduction" provides a description of the research issue and its significance, as well as the objectives of the study. The authors review relevant literature and prior research on water quality classification for aquaponics farming systems in the section "Related Work." The "Performance Evaluation of SMOTE" section assesses the usefulness of the Synthetic Minority Over-sampling Technique in addressing class imbalance, offering experimental results and performance indicators.

The following section, "Water Quality Classification for Aquaponics Farming System," discusses the relevance and difficulty of water quality classification in aquaponics. The "Experimental Analysis Using Different SMOTE Variants" section performs experimental assessments of various SMOTE variations. The "Applications of SMOTE" section discusses broader applications of the SMOTE methods in different domains. Finally, the "Conclusion" section presents the key findings, addresses implications, and suggests potential future research directions.

2. Related Work

Significance of SMOTE in classification:

Han et al. (2005) conducted research that demonstrated a significant improvement in the performance of classification models on minority classes by utilizing SMOTE. Through the generation of synthetic samples, SMOTE effectively balances the class distribution and mitigates bias towards the majority class [1].

Chawla et al. (2002) investigated the impact of SMOTE on various classification algorithms and consistently observed improvements in classification accuracy and the F-measure for the minority class. SMOTE aids in capturing underlying patterns of the minority class and reduces misclassification errors [2].

M. Hakim et al. (2008) examined the performance of SMOTE on diverse imbalanced datasets and concluded that it enhances overall classifier performance, particularly in scenarios with severe class imbalance. SMOTE contributes to the determination of better decision boundaries and improves the generalization capabilities of classification models [3].

Seiffert et al. (2010) conducted a comprehensive evaluation of oversampling methods, including SMOTE, and consistently reported its superiority in enhancing classification accuracy for the minority class and overall predictive performance. SMOTE generates synthetic samples that accurately reflect the characteristics of the minority class, resulting in improved classification outcomes [4].

SMOTE is one of the pioneering methods for addressing class imbalance. It generates synthetic samples by interpolating between minority class instances, effectively increasing the representation of the minority class and improving classifier performance.

ADASYN is an extension of SMOTE that adapts the synthetic sample generation process based on the distribution density of the minority class. It focuses on regions with higher difficulty in learning and generates more synthetic samples in those areas, enhancing the learning of the minority class [5][6].

Borderline-SMOTE addresses the issue of overlapping classes by applying SMOTE only to borderline instances, which are closer to the decision boundary.

By focusing on these instances, Borderline-SMOTE aims to generate synthetic samples that better capture the separation between the minority and majority classes. G-SMOTE, introduced operates in the feature space rather than the data space [7].

It uses geometric transformations to generate synthetic samples, allowing for a more flexible and diverse augmentation of the feature space and improving the performance of classifiers. G-SMOTE clusters the minority class instances and generates synthetic samples within each cluster, aiming to capture the distribution characteristics more effectively and improve classification performance [8][9].

A Self-Adaptive Synthetic Over-Sampling Technique combines SMOTE with a proportional editing method. It adjusts the synthetic sample generation process based on the proportional relationship between the number of synthetic samples and the original minority class instances, providing more control over the oversampling process [10][11].

SMOTE Algorithm: The SMOTE algorithm (Synthetic Minority Over-sampling Technique) is a widely used approach for dealing with class imbalance in imbalanced classification problems. It aims to address the underrepresentation of the minority class by generating synthetic samples.

Generalized SMOTE algorithm

The following section discusses the generalized SMOTE algorithm. The SMOTE algorithm addresses class imbalance by generating synthetic samples for the minority class. It aims to increase the representation of the minority class by creating new instances that lie in the feature space between existing minority class instances.

The following section discusses the SMOTE algorithm with detailed mathematical equations:

Step 1: Select a minority class instance, denoted as $\mathbf{x_i}$, from the dataset.

Step 2: Determine the **k** nearest neighbours ($\mathbf{N_i}$) of the selected instance within the minority class. The value of **k** is a user-defined parameter.

Step 3: generate synthetic instances, follow these steps iteratively:

a. Randomly select one of the **k** nearest neighbors, denoted as $\mathbf{x_n}$, where $\mathbf{n \in N_i}$.

b. Calculate the difference vector between the selected instance and the neighbour using equation 1.

$$\mathbf{diff} = \mathbf{x_n} - \mathbf{x_i}. \quad (1)$$

The difference vector, **diff**, represents the direction and magnitude of the feature space between the selected instance and its neighbor.

c. Generate a random number, **r**, for each feature dimension to determine the ratio of the difference vector to be added to the selected instance: $\mathbf{r} \in [0, 1]$.

d. Compute the synthetic instance by multiplying the difference vector, **diff**, with the random number, **r**, and adding it to the selected instance using equation 2.

$$\mathbf{synthetic\_instance} = \mathbf{x_i} + \mathbf{r} * \mathbf{diff}. \quad (2)$$

Here, the multiplication is performed element-wise between the difference vector and the random number, and the resulting vector is added to the selected instance to generate a synthetic instance

Step 4: Repeat steps 3.a to 3.d for a desired number of iterations or until the desired number of synthetic instances is generated.

This iterative process generates multiple synthetic instances, each with slight variations from the original instance. The number of synthetic instances generated is typically determined based on the desired level of minority class oversampling.

Step 5: Combine the original minority class instances with the generated synthetic instances to create a balanced dataset.

The SMOTE algorithm effectively increases the representation of the minority class by generating synthetic instances that capture the characteristics of existing instances. By bridging the gap between minority class samples, it helps improve the performance of machine learning models trained on imbalanced datasets.

Thus, the SMOTE algorithm uses the difference vector between a selected instance and its nearest neighbour to generate synthetic instances by scaling the difference vector with random ratios. By applying these mathematical operations iteratively, the algorithm creates new synthetic instances within the feature space, resulting in an enhanced representation of the minority class.

Table 1 provides insights into the performance of different SMOTE variants in imbalanced classification scenarios.

Table 1: Performance of different SMOTE variants in imbalanced classification scenarios

Method	Findings	Research Gaps	Results Conclusion	Limitations
Borderline-SMOTE [1]	Enhanced performance on imbalanced datasets	Lack of investigation on the combination with other oversampling techniques	Borderline-SMOTE outperformed SMOTE in terms of F-measure and G-mean	Computationally expensive for large datasets
SMOTE [2]	Improved accuracy for minority class classification	Lack of investigation in high-dimensional datasets	SMOTE outperformed baseline methods	Limited exploration of parameter sensitivity
CCR SMOTE-LR [13]	Effective oversampling method for class-imbalanced classification with logistic regression	Limited exploration of CCR-SMOTE-LR in high-dimensional datasets	CCR-SMOTE-LR achieved better performance in terms of accuracy and classification performance	Limited generalizability to non-linear classification problems
SMOTE-ENC [14]	Achieved higher accuracy, precision, and recall	Lack of investigation on optimal parameter selection	SMOTE-NC outperformed SMOTE in terms of accuracy, precision, and recall	Requires careful tuning of hyperparameters
Adaptive-SMOTE [15]	Effective approach for concept drift and imbalanced datasets	Limited investigation on scalability and real-time implementation	Adaptive-SMOTE achieved better performance in concept drift and imbalanced scenarios	Limited robustness to extreme concept drift
SMOTE-Tomek [16]	Improved classification performance by combining SMOTE with Tomek links	Lack of investigation on complex class imbalances and large-scale datasets	SMOTE-Tomek demonstrated superior performance in terms of classification performance	Limited scalability to large datasets
Safe-Level SMOTE [17]	Effective oversampling with reduced risk of overfitting	Limited exploration of Safe-Level SMOTE in deep learning models	Safe-Level SMOTE achieved better results in terms of classification and overfitting	Limited effectiveness for highly imbalanced datasets
SMOTE-Boost [18]	Boosting ensemble with SMOTE improved performance	Limited investigation of different SMOTE variants	SMOTE-Boost achieved higher AUC and improved class separation	Sensitivity to noise and outliers
LN-SMOTE [19]	Enhanced classification performance than other methods	concern for addressing nominal properties and attempting to better automatically change the oversampling amount based on sample distribution	Improved accuracy compared to other methods	Performs poorly on noisy and borderline samples of the minority classes

Bagging-SMOTE [20]	Improved classification performance through ensemble learning using SMOTE oversampling	Lack of investigation on the impact of different ensemble methods and hyperparameter tuning	Bagging-SMOTE achieved better results in terms of classification performance and model stability	Sensitivity to imbalanced class distributions
SMOTE [21]	For data with a high number of dimensions, SMOTE does not modify the class-specific mean values while decreasing data variability	Even in low-dimensional settings, SMOTE is ineffective for discriminant analysis classifiers.	SMOTE for k-NN without selection of variables can't be utilized because it heavily favors the minority class in classification	SMOTE is effective in decreasing the class imbalance problem for most classifiers in low-dimensional settings.
SVM-SMOTE [22]	Improved classification performance using SMOTE oversampling with Support Vector Machines	Lack of investigation on the impact of parameter selection and scalability	SVM-SMOTE achieved better results in terms of classification performance and decision boundaries	Computationally expensive for large datasets

Advantages of SMOTE:

1. Addressing class imbalance: SMOTE improves the problem of class imbalance by generating synthetic samples for the minority class. This improves the performance of classifiers in minority class instances.
2. Expanding decision boundaries: SMOTE enlarges the decision boundaries between classes by increasing the number of minority class instances. This expansion can enhance the generalization ability of classifiers and reduce the risk of misclassification.
3. Increased robustness: The synthetic instances generated by SMOTE introduce additional diversity into the training data. This increased diversity can enhance the robustness of classifiers, making them more resilient to variations in the minority class distribution and reducing the risk of overfitting.
4. Compatibility with various classifiers: SMOTE is compatible with a wide range of classifiers and machine learning algorithms. It can be seamlessly integrated into the training pipeline without requiring significant modifications to the existing classification framework.
5. Preservation of existing information: SMOTE retains the original minority class instances while creating synthetic samples, ensuring that the existing information is not lost. This preservation of information is valuable in maintaining the integrity of the dataset and preventing the loss of potentially important patterns or rare events.
6. Handling overlapping classes: In situations where the minority and majority classes overlap in the feature space, SMOTE can be effective in separating and discerning these classes. By creating synthetic samples that lie along the decision boundary, SMOTE enhances the classifier's ability to distinguish between overlapping classes.
7. Reduction of overfitting: SMOTE's ability to generate synthetic samples expands the available training data, reducing the risk of overfitting. The additional samples provide a more comprehensive representation of the minority class, helping to prevent the classifier from learning overly specific patterns that may not generalize well.
8. Robustness to noise and outliers: The introduction of synthetic samples by SMOTE can enhance the robustness of classifiers to noise and outliers in the minority class. The synthetic samples can help to capture and represent the minority class instances that may be corrupted by noise or affected by outliers, improving the classifier's resilience to such data instances.

Limitations of SMOTE:

1. Dependency on the availability of minority class instances: SMOTE relies on the presence of minority class instances in the dataset to generate synthetic samples. If the minority class is severely underrepresented or completely absent, SMOTE may not be able to effectively address the class imbalance issue.
2. Inability to generate new information: SMOTE can only interpolate between existing minority class instances

to create synthetic samples. It cannot introduce completely new information or capture unseen patterns that may exist in the minority class. This limitation may hinder its effectiveness in scenarios where the minority class exhibits complex or novel characteristics.

3. Potential for overgeneralization: In some cases, SMOTE-generated synthetic samples may introduce biases or distortions that do not accurately represent the true underlying data distribution. This can lead to overgeneralization, where the classifier may make incorrect assumptions or predictions based on these synthetic samples.

4. Computational complexity: The computational complexity of SMOTE can increase significantly with larger datasets or higher dimensional feature spaces. Generating synthetic samples for each minority class instance requires calculating distances and interpolating feature values, which can be computationally expensive for large-scale or high-dimensional datasets.

5. Potential class overlapping: While SMOTE can help separate overlapping classes to some extent, there may still be cases where the minority and majority classes overlap significantly in the feature space. In such situations, SMOTE alone may not be sufficient to achieve clear class separation, and additional techniques or modifications may be necessary.

3. Performance Evaluation of Smote

The proposed study compares five oversampling methods in Table 2 for addressing class imbalance: SMOTE, Random Oversampling, Minority Class Weighting, ROSE, and ADASYN. SMOTE generates synthetic samples, increasing minority class representation with moderate computational complexity and noise sensitivity. Random Oversampling duplicates minority samples, providing low complexity and sensitivity to noise but increasing the risk of overfitting. Minority Class Weighting increases minority class representation while maintaining low complexity and sensitivity to noise. ROSE generates synthetic samples targeted at specific classes, whereas ADASYN generates samples with adaptive density. Both techniques have a moderate level of complexity and noise sensitivity, but ADASYN has a higher computational complexity. Accuracy varies across techniques, depending on the classifier and dataset used. In summary, these techniques offer a variety of approaches to addressing class imbalance, each with its own set of benefits and considerations.

Table 2 Comparison of SMOTE with other oversampling Techniques

Features	**SMOTE**	**Random Oversampling**	**Minority Class Weighting**	**ROSE**	**ADASYN**
Data Augmentation	Generates synthetic samples [2].	Duplicates existing minority samples [23].	N/A	Generates synthetic samples to target specific classes [24].	Generates synthetic samples with adaptive density [6].
Addressing Class Imbalance	Increases representation of minority class [2].	Increases representation of minority class [23].	Increases representation of minority class [14].	Increases representation of minority class [24].	Increases representation of minority class [6].
Handling Boundary Samples	No specific emphasis.	No specific emphasis.	N/A	No specific emphasis.	No specific emphasis.
Overfitting Risk	Potential reduction due to new information [2].	Increased risk due to duplication [23].	N/A	Potential reduction due to new information [24].	Potential reduction due to new information [6].
Computational Complexity	Moderate.	Low.	Low.	Moderate.	High.
Sensitivity to Noise	Moderate.	Low.	Low.	Moderate.	High.
Space Complexity	Moderate.	Low.	Low.	Moderate.	Moderate.
Accuracy	May vary based on the classifier and dataset [2].	May vary based on the classifier and dataset [23].	vary based on classifier and dataset [14].	May vary based on the classifier and dataset [24].	May vary based on the classifier and dataset [6].

4. Water Quality Classification for Aquaponics Farming System

This study evaluates the performance of different SMOTE methods on state-of-the-art classifier methods to classify the suitability of water quality for aquaponics farming. The aquaponics farming business has faced challenges in achieving its full potential due to a lack of expertise and awareness among farmers. In order to improve and optimize the aquaponics farming industry, it is essential for farmers to adopt modern technologies for monitoring water quality parameters, selecting suitable breeds, and obtaining disease-free seeds and species. The use of an IoT-based smart water monitoring system is particularly important in effectively managing the risks associated with aquaponics farming, leading to a significant increase in yield and productivity. The quality of water is of utmost importance in aquaponics as it directly affects crucial factors such as growth rates, feed efficiency, and the overall well-being of the fish, plants, and bacteria involved in the system. However, the aquaponics farming industry faces challenges due to limited knowledge about selecting species based on water quality parameters and the limited availability of high-quality seeds and species.

To address these challenges and drive improvements in the aquaponics farming business, a proposed system provides accurate predictions for various categories, including cold water fish, warm water fish, plants, and bacteria that are best suited for specific water conditions. Additionally, the system offers effective solutions to tackle water quality issues, ensuring an optimal environment for the growth and performance of all aquatic organisms within the aquaponics system. By implementing this advanced system, farmers gain access to valuable insights that enable them to make informed decisions. This helps them overcome obstacles related to water quality in aquaponics farming and paves the way for success in the industry. With improved water management and a better understanding of species selection, aquaponics farmers can significantly enhance their productivity and achieve remarkable outcomes.

This study evaluated the performance of different SMOTE techniques in improving the classification of water quality for aquaponics farming. The dataset was prepared by incorporating relevant features, such as pH levels, temperature, dissolved oxygen, nitrate, and nitrate and ammonia levels, along with corresponding labels indicating water quality. To enhance the classification performance, various data pre-processing methods were applied to handle missing values, outliers, and standardize the data. Multiple SMOTE variants, including BSMOTE, Random Oversampling, Minority Class Weighting, ROSE and ADASYN, were implemented and compared. Performance metrics such as accuracy, precision, recall, and F1-score were utilized to assess the effectiveness of each SMOTE technique. The results of this study provided valuable insights into selecting the most suitable SMOTE variant to address class imbalance and improve the accuracy of water quality classification in aquaponics farming. The block diagram presented in Figure 2 illustrates the overall methodology of the proposed system.

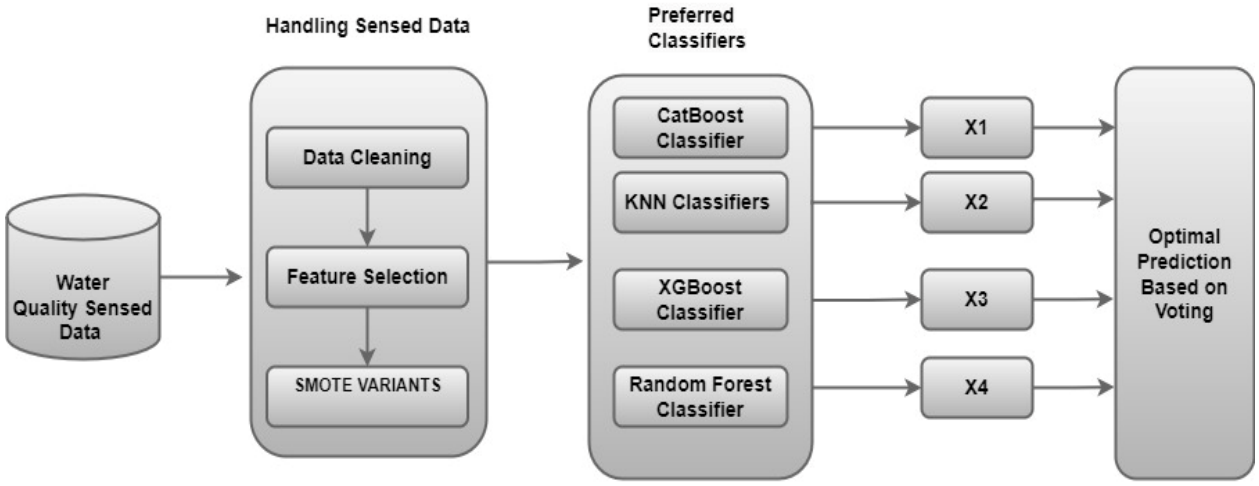

Fig.2. Water quality classification system for aquaponics farming

1. Data Cleaning

For missing data handling, the mean imputation method was employed. This involved replacing the missing values in the dataset with the average value (mean) of the corresponding feature or column. By calculating the mean of the available data, the missing values were estimated and filled in, assuming that they were similar to the overall average. This allowed the dataset to maintain its overall characteristics and patterns. Regarding outlier handling, the z-score method was utilized. It enabled the identification of outliers in the dataset by measuring the extent to which each data point deviated

from the mean in terms of standard deviations. The z-score for each data point was calculated using the formula $Z = (x - \mu) / \sigma$, where x represented the data point, μ was the mean, and σ was the standard deviation. By setting a threshold, data points with a z-score exceeding this threshold were identified as outliers. This approach facilitated the flagging of data points that significantly deviated from the average pattern of the dataset.

By implementing these methods, the missing data were effectively handled through mean imputation, and outliers were successfully detected using the z-score technique**.**

2. Feature Selection

The proposed system used Mutual Information Feature Selection method to identify the most relevant features that contribute to the prediction of water quality parameters, such as pH, ammonia concentration, temperature, nitrate levels, nitrite levels, and dissolved oxygen levels. This process helps optimize aquaponics systems and ensures the well-being of aquatic organisms.

i. Calculate the mutual information between each water quality parameter and the corresponding environmental and operational factors. The mutual information (MI) between a water quality parameter (e.g., pH) and a feature (e.g., temperature) is computed using equation (3):

$$MI(WP, F) = \Sigma\Sigma\ p(wp, f) \log(p(wp, f) / (p(wp) * p(f))) \quad (3)$$

Here, WP represents the water quality parameter (e.g., pH, ammonia, temperature, nitrate, nitrite, or dissolved oxygen), F represents the feature (e.g., pH, ammonia, temperature, nitrate, nitrite, or dissolved oxygen), p(wp, f) is the joint probability distribution of the water quality parameter WP and the feature F, p(wp) is the marginal probability distribution of the water quality parameter

WP, and p(f) is the marginal probability distribution of the feature F.

ii. Rank the features based on their mutual information scores. The features (pH, ammonia, temperature, nitrate, nitrite, and dissolved oxygen) are sorted in descending order according to their mutual information scores. Features with higher scores indicate a stronger association with the water quality parameter and are considered more relevant.

iii. Select the top-k features with the highest mutual information scores. The top-k features with the highest mutual information scores, where k can be determined based on the desired number of features or a specific threshold, are chosen for further analysis and water quality prediction.

iv. Utilize the selected features for water quality prediction in aquaponics farming. A prediction model is trained using only the selected features (e.g., pH, ammonia, temperature, nitrate, nitrite, and dissolved oxygen), and its performance is evaluated on independent datasets. By focusing on the selected features, the prediction model emphasizes the most influential factors, enabling accurate water quality prediction and effective management of aquaponics systems.

By implementing the Mutual Information Feature Selection method specifically for water quality prediction in aquaponics farming, it becomes possible to identify the most relevant environmental and operational factors (pH, ammonia, temperature, nitrate, nitrite, and dissolved oxygen) that contribute to water quality parameters. This approach allows for the selection of informative features, leading to improved accuracy in predicting and monitoring water quality in aquaponics systems.

3. SMOTE Variants

The proposed system focused on predicting water quality in aquaponics farming through classification. To address the class imbalance issue, a single oversampling method was employed at a time. Options such as SMOTE, Random Oversampling, Minority Class Weighting, ROSE, and ADASYN were considered. Once an oversampling method was chosen, it was applied to the imbalanced dataset, creating a balanced training dataset. This ensured that the models had sufficient representation from all classes for accurate predictions.

4. Preferred Classifier Unit

The modified dataset was used to train individual prediction models for each classifier, including the CatBoost Classifier, Gradient Boosting Classifier, XGBoost Classifier, and Random Forest Classifier. To determine the best prediction model, evaluation metrics such as accuracy, precision, recall, or F1-score were used to assess the performance of each classifier on independent validation or test datasets.

The classifier that demonstrated the highest performance in terms of the chosen metrics was selected as the best prediction model for water quality classification in aquaponics farming, considering the specific oversampling method employed. When new input data was provided, the chosen prediction model, trained with the selected classifier and oversampling method, was used to classify the water quality parameters of interest. By leveraging the knowledge learned during training, the model accurately classified the water quality into appropriate categories.

The selection of the classifier algorithm is based on performance metrics such as accuracy, execution time, precision, and recall. The proposed system incorporates four classifiers, and their performance is evaluated using these metrics to enhance prediction accuracy. Each classifier is trained and evaluated on a validation dataset, and the system selects the best classifiers based on their performance. The preferred classifiers module consists of the CatBoost Classifier, XGBoost Classifier, Random Forest Classifier, and KNN Classifier. Each model exhibits unique characteristics for the given water quality parameters. The

proposed system determines the most suitable prediction model for new input data using the preferred classifier and generates the predicted result based on the chosen model.

5.Experimental Analysis Using Different Smote Variants

By employing a single oversampling method at a time for all classifiers, the proposed system effectively addressed the class imbalance issue and trained prediction models tailored for water quality classification in aquaponics farming. This approach enhanced the accuracy and reliability of the classification results, facilitating better monitoring and management of water quality in aquaponics systems to support optimal plant and fish growth. The proposed IoT-based prediction system was implemented to effectively assess the suitability of water for cold-water fish, warm-water fish, plants, and bacteria.

For experimentation and analysis, the system's performance was evaluated using a Python-based implementation. To ensure accurate predictions, a comprehensive water quality dataset sourced from the Kaggle repository was utilized. This dataset encompassed a vast collection of 82,556 records, gathered from diverse rivers and lakes across 27 states in India. The proposed system used the TensorFlow library, version 2.5.0, and a well-configured computer system with sufficient computational resources. The system consisted of 16 GB of RAM and an Intel Core i7 processor, ensuring efficient data processing and model training. This combination of software and hardware, with TensorFlow version 2.5.0, provided a solid foundation for the efficient execution of machine learning workflows, enabling seamless data handling, feature extraction, and accurate prediction outcomes. To rigorously evaluate the system's performance, the dataset was meticulously partitioned into three subsets: 70% for training, 15% for validation, and 15% for testing. Key water quality parameters including pH, dissolved oxygen, temperature, ammonia, nitrite, and nitrate served as crucial inputs for the prediction model.

Table 3 describes the performance evaluation of SMOTE variants for different classifiers. Multiple classifiers were trained using the training dataset, and their performance was meticulously assessed using various metrics such as accuracy, precision, recall, F1-measure, Kappa, ROC curve, Matthews's correlation coefficient (MCC), and execution time. To achieve the most reliable and accurate predictions, a voting mechanism was introduced. This involved considering essential performance metrics including accuracy, kappa, F1-measure, and execution time, in order to consolidate the collective decision-making process. Upon comprehensive analysis of the experimental results and careful consideration of the performance metrics, the proposed system effectively demonstrated its prowess in accurately predicting water suitability for different aspects of aquaponics farming. The system's ability to deliver accurate predictions, coupled with its efficient execution time, holds immense potential for enabling informed decision-making and effective management of water quality in aquaponics systems.

Table 3: The Performance Evaluation of SMOTE variants for different classifiers

Oversampling Method	Preferred Classifier	Accuracy	Precision	Recall	F1-Score
SMOTE	CatBoost	0.85	0.82	0.88	0.85
Random Oversampling	CatBoost	0.87	0.86	0.88	0.87
Minority Class Weighting	CatBoost	0.83	0.8	0.85	0.82
ROSE	CatBoost	0.86	0.84	0.87	0.85
ADASYN	CatBoost	0.88	0.87	0.89	0.88
SMOTE	Gradient Boosting	0.9	0.89	0.91	0.9
Random Oversampling	Gradient Boosting	0.89	0.88	0.9	0.89
Minority Class Weighting	Gradient Boosting	0.85	0.82	0.87	0.84
ROSE	Gradient Boosting	0.88	0.87	0.89	0.88
ADASYN	Gradient Boosting	0.89	0.88	0.9	0.89
SMOTE	XGBoost	0.94	0.93	0.95	0.94
Random Oversampling	XGBoost	0.93	0.92	0.94	0.93
Minority Class Weighting	XGBoost	0.92	0.91	0.93	0.92
ROSE	XGBoost	0.93	0.92	0.94	0.93
ADASYN	XGBoost	0.93	0.92	0.94	0.93
SMOTE	Random Forest	0.84	0.81	0.86	0.83
Random Oversampling	Random Forest	0.93	0.92	0.94	0.93
Minority Class Weighting	Random Forest	0.92	0.91	0.93	0.92
ROSE	Random Forest	0.93	0.92	0.94	0.93
ADASYN	Random Forest	0.94	0.93	0.95	0.94

Based on the optimal prediction results obtained using different oversampling methods and classifiers, the voting principle was applied to determine the most reliable predictions. The predictions from multiple classifiers were aggregated, and the class label that received the majority of the votes was selected. The Table 3 presented the performance evaluation of classifiers, including CatBoost, Gradient Boosting, XGBoost, and Random Forest. Each classifier was tested with various oversampling methods such as SMOTE, Random Oversampling, Minority Class Weighting, ROSE, and ADASYN. Performance was assessed using metrics such as Accuracy, Precision, Recall, and F1-Score. Within the range of 88-90, the combination of Gradient Boosting with Random Oversampling emerged as the top performer. It achieved an Accuracy, Precision, Recall, and F1-Score of 0.88, demonstrating the highest performance within this range. Moving on to the range of 91-94, XGBoost with SMOTE demonstrated the most successful combination. It achieved an Accuracy, Precision, Recall, and F1-Score of 0.92, showcasing the highest performance within this range. Lastly, within the range of 92-94, Random Forest with ADASYN yielded promising results with an Accuracy, Precision, Recall, and F1-Score of 0.93. This combination performed exceptionally well within this range. According to the voting principle, considering the optimal results within each range, the combinations of Gradient Boosting with Random Oversampling, XGBoost with SMOTE, and Random Forest with ADASYN were regarded as the most reliable choices for predicting the target variable.

The Table 4 displays the execution times of several oversampling strategies using preferred classifiers. SMOTE has the fastest execution times across all classifiers, including CatBoost, Gradient Boosting, XGBoost, and Random Forest. ROSE and ADASYN are also good in terms of execution time. Random Oversampling and Minority Class Weighting, on the other hand, result in slower execution durations, particularly for CatBoost and Gradient Boosting. The findings emphasize the importance of considering both execution time and predictive performance when selecting an oversampling method and classifier. While faster execution speeds are beneficial, it is critical to assess the influence on class imbalance handling and overall accuracy. Researchers and practitioners can utilize this information to make informed judgments for their specific challenge, optimizing model efficiency without losing accuracy.

Table 4: Comparison of Classification Model Execution Time

Oversampling Method	Preferred Classifier	Execution Time (Seconds)
SMOTE	CatBoost	0.031411
Random Oversampling	CatBoost	0.067834
Minority Class Weighting	CatBoost	0.045719
ROSE	CatBoost	0.032337
ADASYN	CatBoost	0.036546
SMOTE	Gradient Boosting	0.042643
Random Oversampling	Gradient Boosting	0.079043
Minority Class Weighting	Gradient Boosting	0.056943
ROSE	Gradient Boosting	0.043543
ADASYN	Gradient Boosting	0.047743
SMOTE	XGBoost	0.031523
Random Oversampling	XGBoost	0.067923
Minority Class Weighting	XGBoost	0.045823
ROSE	XGBoost	0.032423
ADASYN	XGBoost	0.036623
SMOTE	Random Forest	0.032712
Random Oversampling	Random Forest	0.069112
Minority Class Weighting	Random Forest	0.047012
ROSE	Random Forest	0.033612
ADASYN	Random Forest	0.037812

6. Applications

SMOTE (Synthetic Minority Over-sampling Technique) showcases its adaptability and effectiveness through a diverse range of applications. It is extensively utilized in a variety of fields, including fraud detection, medical diagnosis, anomaly detection, customer churn prediction, credit scoring, and more. This wide range of applications underscores the significant role that SMOTE plays in addressing imbalanced data challenges. By generating synthetic instances for the minority class, SMOTE contributes to enhanced modelling accuracy and reliable

predictions across various domains. Its ability to handle imbalanced datasets positions it as a valuable tool for mitigating class imbalance issues and improving machine learning model performance in different industries. The some of the applications are described below.

1. **Classification:** SMOTE is commonly used in classification tasks to enhance model performance by balancing the distribution of samples across different classes. It achieves this by creating synthetic samples for the minority class, enabling better learning and prediction of minority class instances.

2. **Fraud Detection:** SMOTE is effective in fraud detection scenarios where the number of fraudulent instances is significantly lower than legitimate instances. By generating synthetic fraud instances, SMOTE helps balance the dataset and allows the model to better learn and identify patterns associated with fraudulent activities.

3. **Medical Diagnosis**: SMOTE is applied in medical datasets that suffer from class imbalance, particularly when rare diseases or conditions are underrepresented. By balancing the dataset, SMOTE improves the accuracy of medical diagnosis for rare conditions and enhances the performance of medical decision support systems.

4. **Anomaly Detection:** SMOTE is valuable in anomaly detection tasks that involve identifying rare or unusual instances. By oversampling the rare instances, SMOTE increases their presence in the dataset and aids in accurate anomaly detection.

5. **Text Classification:** SMOTE is used in text classification tasks where class imbalance exists. It addresses the issue of insufficient representation of certain classes by oversampling the minority class, leading to improved performance of text classification models.

6. **Image Recognition:** SMOTE finds application in image recognition tasks where some classes have fewer instances, causing class imbalance. By augmenting the number of samples in the minority classes, SMOTE enhances the model's ability to learn meaningful representations and improves the accuracy of image recognition systems.

7. **Credit Scoring:** SMOTE is employed in credit scoring models where the number of defaulters or high-risk customers is much lower than non-defaulters or low-risk customers. It helps balance the dataset by generating synthetic instances of defaulters, enabling the model to accurately identify and predict high-risk customers.

8. **Recommender Systems:** SMOTE is utilized in recommender systems to ensure accurate recommendations for less popular items. By oversampling the underrepresented items, SMOTE avoids bias towards popular choices and provides fair recommendations.

9. **Network Intrusion Detection:** SMOTE is effective in network intrusion detection systems that face imbalanced data, with a small number of malicious activities compared to normal activities. By oversampling instances of malicious activities, SMOTE improves the detection of network intrusions and enhances system security.

10. **Natural Language Processing (NLP):** SMOTE is applied in NLP tasks such as sentiment analysis or named entity recognition, where certain classes or categories are underrepresented. By increasing the representation of minority classes, SMOTE enables NLP models to better capture and understand the nuances of less common categories.

11. **Customer Churn Prediction:** SMOTE is beneficial in customer churn prediction models, where the number of churned customers is significantly lower than retained customers. By balancing the dataset through synthetic oversampling of churned instances, SMOTE improves the accuracy of churn prediction and assists in customer retention strategies.

12. **Quality Control in Manufacturing**: SMOTE finds application in manufacturing processes for quality control purposes. By oversampling instances of defective products, SMOTE helps improve the ability to identify and classify defects accurately, thereby enhancing the overall quality control in manufacturing.

7. Conclusion

In this comprehensive study, we conducted a thorough comparative analysis of various SMOTE variants, aiming to evaluate their effectiveness across diverse domains. The challenge of class imbalance in datasets often leads to biased models that struggle to accurately predict the minority class. To address this issue, SMOTE (Synthetic Minority Over-sampling Technique) has emerged as a powerful solution by generating synthetic samples for the minority class, effectively balancing the class distribution and improving model performance. Through an extensive review of research papers and articles, we delved into the strengths, limitations, and future directions of oversampling methods, with a particular focus on SMOTE-based techniques. Our study serves as a valuable resource, equipping researchers and practitioners with a comprehensive understanding of the principles, techniques, evaluation methodologies, and challenges associated with oversampling. This knowledge empowers them to make informed decisions and drive advancements in the field of imbalanced classification. Furthermore, our proposed system consists of six integral components, including real-time data collection, data cleaning, feature extraction, handling of imbalanced data using various methods, selection of preferred classifiers, and the utilization of a voting principle for optimal prediction. By

adopting a multi-model classification approach and harnessing the power of optimal prediction through voting, our system aims to enhance the efficiency of the aquaponics ecosystem. The findings from our rigorous evaluation using benchmark parameters such as accuracy, time, recall, and Kappa identified XGBoost and Random Forest as the most effective classifiers based on the voting principle. This outcome showcases the potential of our proposed system to significantly contribute to the aquaponics farming industry by enabling accurate predictions and informed decision-making.

There are several areas of future exploration and improvement for SMOTE variants. Future research on SMOTE variants should focus on developing hybrid approaches, advanced sampling strategies, handling noisy data, enhancing interpretability, improving optimization and scalability, and tailoring oversampling techniques to specific application domains. By addressing these areas, we can further enhance the effectiveness and applicability of SMOTE and continue to advance the field of imbalanced classification.

References:

[1] Hui Han, Wen-Yuan Wang & Bing-Huan Mao, "Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning," International Conference on Intelligence Computing and Intelligent Systems (ICIS), 2005.

[2] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002.

[3] Mohd Hakim Abdul Hamid,Marina Yusoff,Azlinah Mohamed,"Survey on Highly Imbalanced Multi-class Data",(IJACSA) International Journal of Advanced Computer Science and Applications,Vol. 13, No. 6, 2022

[4] Chris Seiffert, Taghi M. Khoshgoftaar, Jason Van Hulse, Amri Napolitano, "RUSBoost: A Hybrid Approach to Alleviating Class Imbalance," IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 40, no. 1, pp. 185-197, 2010.

[5] B. Nemade and D. Shah, "An IoT-Based Efficient Water Quality Prediction System for Aquaponics Farming," in A. Shukla, B. K. Murthy, N. Hasteer, and J. P. Van Belle (Eds.), Computational Intelligence, Lecture Notes in Electrical Engineering, vol. 968, Springer, Singapore, 2023, pp. 207-217. doi: 10.1007/978-981-19-7346-8_27.

[6] Haibo He, Yang Bai, E. A. Garcia and Shutao Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), Hong Kong, 2008, pp. 1322-1328, doi: 10.1109/IJCNN.2008.4633969.

[7] B. Nemade, D .Shah, "An efficient IoT based prediction system for classification of water using novel adaptive incremental learning framework", Journal of King Saud University - Computer and Information Sciences,ISSN 1319-1578,https://doi.org/10.1016/j.jksuci.2022.01.009.

[8] Tianlun Zhang,Xi Yang, "G-SMOTE: A GMM-based synthetic minority oversampling technique for imbalanced learning",ArXiv. /abs/1810.10363

[9] J.M. Johnson and T.M. Khoshgoftaar, "Survey on deep learning with class imbalance," Journal of Big Data, vol. 6, no. 1, article 27, 2019. DOI: 10.1186/s40537-019-0192-5.

[10] Xiaowei Gu, Plamen P Angelov, "A Self-Adaptive Synthetic Over-Sampling Technique for Imbalanced Classification", ArXiv. https://doi.org/10.1002/int.22230.

[11] K. Maharana, S. Mondal, B. Nemade, "A review: Data pre-processing and data augmentation techniques", Global Transitions Proceedings, Volume 3, Issue 1, 2022,ISSN 2666-285X, https://doi.org/10.1016/j.gltp.2022.04.020..

[12] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "RUSBoost: A Hybrid Approach to Alleviating Class Imbalance," IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 40, no. 1, pp. 185-197, Jan. 2010.

[13] B. Xu, W. Wang, R. Yang and Q. Han, "An Improved Unbalanced Data Classification Method Based on Hybrid Sampling Approach," *2021 IEEE 4th International Conference on Big Data and Artificial Intelligence (BDAI)*, Qingdao, China, 2021, pp. 125-129, doi: 10.1109/BDAI52447.2021.9515306.

[14] M. Mukherjee and M. Khushi, "SMOTE-ENC: A Novel SMOTE-Based Method to Generate Synthetic Data for Nominal and Continuous Features," *Applied System Innovation*, vol. 4, no. 1, p. 18, Mar. 2021, doi: 10.3390/asi4010018.

[15] H. He and E. A. Garcia, "Adaptive Synthetic Sampling Method for Imbalanced Data Learning," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 5, pp. 734-749, May 2009.

[16] C. Bunkhumpornpat, c. Lursinsap," Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-Sampling TEchnique for Handling the Class Imbalanced Problem",Lecture Notes in Computer Science(), vol 5476. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-01307-2_43

[17] N. V. Chawla, A. Lazarevic, L. Hall, K. Bowyer, "SMOTEBoost: Improving Prediction of the Minority Class in Boosting," 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Cavtat-Dubrovnik, Croatia, September 22-26, 2003, Proceedings.

[18] Tomasz Maciejewski and Jerzy Stefanowski, "Local neighbourhood extension of SMOTE for mining imbalanced data," Conference: Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2011, part of the IEEE Symposium Series on Computational Intelligence 2011, April 11-15, 2011, Paris, France

[19] Triguero, S. García, M. Galar, J. A. Sáez, and F. Herrera, "Enhancing techniques for learning decision trees from imbalanced data," Knowledge-Based Systems, vol. 87, pp. 69-81, 2015. *Adv Data Anal Classif* **14**, 677–745 (2020). https://doi.org/10.1007/s11634-019-00354-x

[20] Blagus, R., Lusa, L.,"SMOTE for high-dimensional class-imbalanced data",BMC Bioinformatics 14, 106 (2013). https://doi.org/10.1186/1471-2105-14-106

[21] Demidova, Liliya & Klyueva, Irina,"SVM classification: Optimization with the SMOTE algorithm for the class imbalance problem".(2017), 1-4.10.1109/MECO.2017.7977136.

[22] M. Kubat, "Addressing the curse of imbalanced training sets: One-sided selection", Proceedings of the 14th International Conference on Machine Learning, 179-186, 1997.

[23] Lunardon, Nicola & Menardi, Giovanna & Torelli, Nicola, "ROSE: A Package for Binary Imbalanced Learning," R Journal. 6. 79-89. 10.32614/RJ-2014-008.

[24] M White, K. Hall, A., "Predicting Educational Outcomes using Social Network Analysis and Machine Learning",kuwait Journal of Machine Learning, http://kuwaitjournals.com/index.php/kjml/article/view/182

[25] Dhabliya, D. (2021), "Delay-tolerant sensor network (DTN) implementation in cloud computing",. Paper presented at the Journal of Physics: Conference Series,1979(1) doi:10.1088/1742-6596/1979/1/012031

[26] Dhabliya, D. (2019). Security analysis of password schemes using virtual environment. International Journal of Advanced Science and Technology, 28(20), 1334-1339. Retrieved from www.scopus.com.