International Journal of Computer and Information System (IJCIS)
Peer Reviewed – International Journal
Vol : Vol. 06, Issue 01, January 2025
e-ISSN : 2745-9659
https://ijcis.net/index.php/ijcis/index

Hybrid Approach to Improving the Results of the SVM Classification Using Posterior Probability and Correlation

[1,3]Canggih Ajika Pamungkas, [2]Megat F. Zuhairi
[1]Malaysian Institute of Information Technology, Universiti Kuala Lumpur, Malaysia
[2]Malaysian Institute of Information Technology, Universiti Kuala Lumpur, Malaysia
[3]Politeknik Indonusa Surakarta, Indonesia
E-mail : [1]pamungkas.canggih@s.unikl.edu.my, [2]megatfarez@unikl.edu.my

Abstract— Class imbalance in datasets poses significant challenges to traditional machine learning models, such as Support Vector Machines (SVM), leading to poor performance in minority class classification. To address this issue, this study introduces a hybrid approach, Posterior Probability and Correlation-SVM (PC-SVM), which combines posterior probability estimation and correlation analysis. The purpose of this research is to enhance SVM's ability to classify imbalanced datasets by weighting attributes based on their correlation with the target class and leveraging posterior probabilities to refine decision boundaries. The methodology includes preprocessing datasets to ensure data quality, applying correlation analysis to calculate attribute weights, and using these weights to transform input features into posterior probability estimates. The transformed features serve as inputs to the SVM for classification. Experiments were conducted on two datasets: Yeast and Churn, which exhibit varying degrees of class imbalance. The results demonstrate that the PC-SVM model achieves 100% accuracy, precision, recall, and F1-scores across all classes, significantly outperforming the standard SVM. The approach effectively mitigates the bias toward majority classes by improving sensitivity to minority instances. This study highlights the robustness and reliability of the PC-SVM model in handling imbalanced data classification. In conclusion, integrating posterior probabilities with correlation-based attribute weighting significantly enhances the performance of SVMs on imbalanced datasets. Future research should focus on extending this approach to multiclass problems and optimizing its computational efficiency.

Keywords : *SVM, imbalanced datasets, posterior probability, correlation analysis, classification, machine learning*

I. INTRODUCTION

Supervised categorization entails employing a training dataset and statistical learning techniques to classify items into designated categories, followed by the use of this knowledge to categorize new data [1]. Supervised learning, a core component of machine learning, employs algorithms that discern patterns in data by leveraging known independent and dependent variables to forecast future outcomes, with supervised classification presuming cluster labels as parameters while tackling issues such as class distribution imbalance [2].

The importance of imbalanced data classification is growing in the domains of data mining and machine learning [3]. A dataset exhibits imbalance when one class substantially exceeds the other, with the minority class designated as the positive (+) class and the majority class as the negative (−) class in data classification [3]. The problem of class imbalance has received considerable attention in recent studies. [4]–[10]. Previous research efforts have aimed to tackle the problem of data imbalance by incorporating sampling, ensemble, and cost-sensitive techniques into classification systems. Sampling strategies are employed to convert the distribution of imbalanced data into a balanced distribution. [11]. The challenge of learning from imbalanced data sets is a considerable impediment in the domain of data mining. Conventional support vector machines often exhibit robust performance in addressing classification issues with imbalanced datasets; nevertheless, they regard all training samples uniformly during the learning phase. This may result in a bias in the ultimate decision boundary favoring the majority class, especially in the presence of outliers or noise [12]. Imbalanced data categorization transpires when one class possesses a greater number of instances than another, resulting in the majority class frequently eclipsing the minority class, which conventional classifiers often regard as noise. This bias towards the majority class has necessitated the creation of diverse methodologies to mitigate this problem [13].

The support vector machine (SVM) is an effective machine learning tool recognized for its speed, simplicity, reliability, and capacity to yield accurate categorization outcomes [14]. Support Vector Machine (SVM) constructs a model utilizing the available sample sizes of each class. The concept of SVM learning is based on the principles of structural risk minimization. The Support Vector Machine (SVM) can be utilized to reduce the limitations of generalization error, hence improving its efficacy when applied to data beyond the training set [15]. The objective of Support Vector Machine (SVM) is to determine the hyperplane that distinguishes two classes inside a vector space. [16]. The separating hyperplane is positioned between two parallel hyperplanes, with one positioning vectors of the first class above it and the other positioning vectors of the second class below it. The margin denotes the distance between these hyperplanes, and in scenarios permitting misclassifications for enhanced generalization, the margin is considered "soft." Meanwhile, SVM continues to be an exceptionally effective method for supervised classification. [1].

Conventional classification methods presume uniform probabilities for data across various classes; however, in practical situations, minority classes often possess fewer data points than majority classes. This discrepancy leads to a bias in traditional algorithms favoring majority classes, thereby diminishing the accuracy of minority class classification [2]. To tackle the issues of imbalanced data, numerous solutions have been suggested, classified into three categories: Data-level techniques that alter sample probabilities via oversampling or undersampling to equilibrate the dataset, algorithm-level techniques that modify classification systems with cost-sensitive strategies to impose greater penalties on the misclassification of minority samples, and fusion approaches that integrate various methods, such as sampling and cost-sensitive techniques, to address the imbalance problem [2].

II. RESEARCH METHODS

The research process has five unique phases: Data Sources, Data Preparation, Experimentation, Modeling, and Model Evaluation.

2.1 Data Sources

The research issue necessitates the use of data in order to provide a response. The research resources utilized in this study consist of publicly available data sets obtained from the UC Irvine (UCI) Machine Learning Repository and Kaggle.

Table 1 displays the roster of utilized data sources.

Table 1. List of data source

Public Dataset	References
UCI	[17] [18] [19] [20] [21] [22] [23] [24] [25] [26]
Kaggle	[27] [28] [29] [30] [31] [32] [33] [34] [35] [36][37]

Table 2. The churn dataset exhibits the highest imbalance ratio at 5.36. The maximum occurrences in the churn dataset are 3150. The dataset with the most features is churn, comprising 13 features, while the dataset with the fewest features is yeast, containing 8 features.

Table 2. Detail Dataset

Dataset	Public dataset	Instances	Features	Positive Instances (%)	Negative Instances (%)	IR
Churn	UCI	3150	13	16	84	5,36
Yeast	Kaggle	1484	8	28,9	71,1	2,46

2.2 Data Preparation

Data preparation is an essential technique in machine learning that can significantly improve model outcomes. [38] [39] [40]. Data pre-processing is a crucial and fundamental phase in the machine learning lifecycle. A major challenge in the healthcare sector is acquiring a complete and untainted dataset. The quality of data is crucial, since it can profoundly affect the model's learning ability and overall generalizability [41]. Efficient and precise algorithms can be attained through the utilization of excellent data preparation techniques and processes. This provides a robust foundation for data-driven decision-making and application development [42]. The data was subjected to preprocessing techniques such as encoding, imputation, transformation of skewed distributions, balancing, scaling, and selection of features [43] [44] [45]. Data preprocessing involves a set of techniques designed to enhance the quality of the original data, such as the removal of outliers and the imputation of missing values [46]. A crucial phase in the data analysis process is preprocessing, which involves converting raw data into a format that is interpretable by computers and machine learning algorithms. This critical phase profoundly impacts the precision and efficacy of machine learning models [47]. The data preparation process encompasses essential stages, such as missing value identification and data transformation.

2.3 Experiment

This research conducts a series of experiments to evaluate classification performance under various conditions, focusing on the performance of PC-SVM and SVM classifiers when applied to imbalanced datasets. The study aims to compare the effectiveness of these classifiers in handling the challenges posed by imbalanced data, analyzing their ability to produce accurate predictions and assess the impact of dataset imbalance on their performance. The experiments will explore how each classifier responds to skewed data distributions, providing insights into their robustness and efficiency in such contexts.

2.4 Modelling

The general architecture of a Support Vector Machine (SVM) involves three main layers: the input layer, the hidden layer, and the output layer. The input layer accepts the features of the data, denoted as $X_1, X_2, \ldots, X_n,$, which are then transformed using a kernel function, $K\ (X, X_i)$, to map the input data into a higher-dimensional space. This transformation enables the SVM to find a hyperplane that maximizes the margin between different classes. The hidden layer computes these transformations and combines them with a bias term, b, and the outputs of the kernel functions are summed to generate the result. Finally, the output layer provides the classification or regression outcome based on the computed values. The architecture emphasizes the importance of kernel functions in handling non-linear relationships, making SVM a versatile and powerful model for various types of data. **Figure 1** is the General architecture of the SVM.

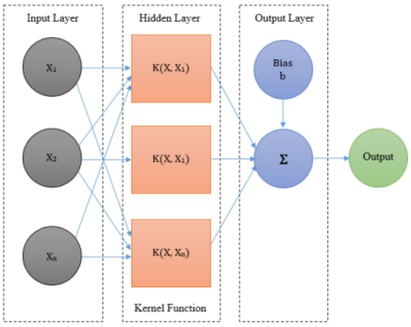

Figure 1: General architecture of the SVM

The proposed concept integrates the method known as PC-SVM, which stands for Posterior Probability and Correlation-Support Vector Machine. The core premise of the PC-SVM method is the integration of Posterior Probability and Correlation Techniques, which significantly enhances SVM performance on imbalanced datasets. In an imbalanced dataset, where one class predominates, posterior probability facilitates the computation of class probabilities based on feature likelihood, thus elucidating the probability of a sample belonging to either the minority or majority class. The probability distribution is optimized by multiplying the prior probability with the aggregate product of the R-squared value of feature i of class Y and the independent probabilities of all feature vectors X. The attribute weights in the proposed method are derived from the correlation coefficient between the attribute and the class. The correlation coefficient ranges from -1 to 1, indicating that the attribute weighting value may be negative. To avert the emergence of negative values, the R Square value is utilized for attribute weighting. Attribute weighting is a technique in which the R Square value of each attribute for the class is multiplied by the probability of each attribute to compute the conditional probability in the Naive Bayes Classifier utilizing the joint probability method.

Figure 2 depicts the overall structure of the suggested algorithm:

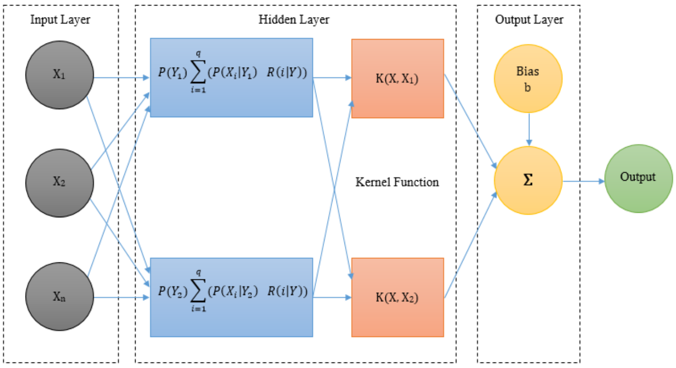

Figure 2: General architecture of the proposed algorithm

This attribute weighting can enhance accuracy by evaluating the strength of the attribute's correlation with the designated class. The posterior probabilities are utilized as input features for the SVM, which is proficient in identifying the ideal separating hyperplane between classes. By employing posterior probabilities to transform the original feature space into probability estimates, SVM can concentrate on maximizing the margin between classes utilizing these probabilities. This alleviates disparities by enhancing the sensitivity of decision limits to minority class instances, hence improving classification accuracy and diminishing bias towards the majority class.

The PC-SVM algorithm utilizes an attribute weighting method grounded in R Square. The R Square value is a statistic that quantifies the extent of influence a characteristic has on the class, taking into account its weight. The dataset has attributes $X_1, X_2, X_3, \ldots, X_n$, each associated with matching weights $R_1, R_2, R_3, \ldots, R_n$. The SVM approach use joint probability to determine conditional probability. attribute weighting is a technique that allocates a numerical number to each attribute to signify its relative significance. Therefore, the correct strategy in conditional probability is utilizing the principle of total addition instead of total multiplication. The posterior probability is utilized to determine the level of confidence in a classification.

The posterior probability alone offers probabilities without accounting for the strength of associations among features. Incorporating R Square enhances the model's informational depth, as each feature is evaluated not only by its likelihood distribution but also by its significance to the target class. This leads to a more profound probability framework, which is crucial for addressing imbalanced datasets, because the bulk of attributes may be more pertinent to the dominant class and less effective in identifying the minority class. By integrating R Square with the independent probabilities of characteristics, enhance the model's sensitivity to fluctuations and patterns associated with the minority class. This method enables the model to identify nuanced patterns crucial for recognizing the minority class in an imbalanced dataset, which frequently becomes obscured by the prevalence of the dominant class.

2.5 Model Evaluation

2.5.1 Confusion Matrix

In supervised learning classification problems, model performance evaluation frequently depends on the metrics obtained from the confusion matrix. This matrix illustrates the actual and forecasted values for the categories of the target attribute [48]. The confusion matrix is a widely utilized metric in addressing classification challenges. It is applicable to both binary and multiclass classification issues [49]. Table 3 presents an example of a confusion matrix for binary classification.

Table 3. Confusion Matrix for Binary Classification

		Predicted	
Actual		Negative	Positive
	Negative		
	Positive		

The models are assessed using Accuracy, Precision, Recall, and F1 Score derived from the confusion matrix. The confusion matrix is structured by Total True Positive (TTP), Total False Negative (TFN), Total True Negative (TTN), and Total False Positive (TFP) to delineate model performance, as articulated in equations (1) through (4) [50].

Accuracy is the ratio of correctly predicted labels to the total number of predicted labels, as expressed by equation (1).

$$Accuracy = \frac{1}{1 + \left(\frac{TFP+TFN}{TTP+TTN}\right)} \quad (1)$$

Precision is determined by dividing the count of true positives by the total of true positives and false positives. False positives arise when the model erroneously classifies negative instances as positive. In this instance, false positives imply individuals erroneously identified as terrorists by the model, while not being so. The definition of precision is provided in equation (2):

$$Precision = \frac{1}{1 + \frac{TFP}{TTP}} \quad (2)$$

The formula for recall in a classification algorithm is given by equation (3):

$$Recall = \frac{(\text{TTP})}{(\text{TTP} + \text{TFN})} \quad (3)$$

The F1 Score of a system is determined as the weighted harmonic mean of its precision and recall. The F1 Score is defined by equation (4).

$$F1\,Score = \frac{2\ x\ Recall\ x\ \ Precision}{Recall\ x\ \ Precision} \quad (4)$$

2.5.2 Scatter Plot

A scatter plot is a form of data visualization that represents values for usually two variables within a dataset. Every point on the graph signifies an observation inside the dataset. The location of a point on the horizontal (x) axis denotes the value of one variable, while the position on the vertical (y) axis indicates the value of the other variable. Scatter plots are employed to examine relationships, patterns, and correlations between two variables. Below are few essential elements:

a. Correlation: Scatter plots can illustrate positive correlation, negative correlation, or the absence of correlation between the variables.
    1) Positive correlation: An increase in one variable corresponds with an increase in the other variable.
    2) Negative correlation: An rise in one variable corresponds to a decrease in the other variable.
    3) No correlation: No discernible relationship exists between the variables.

b. Trend lines: Occasionally, a line of best fit, or trend line, is incorporated into the scatter plot to illustrate the overall trajectory of the data points. This can assist in discerning linear correlations among the variables.
c. groupings and outliers: Scatter plots help elucidate groupings of data points and pinpoint outliers that markedly diverge from the other observations.
d. Multidimensional scatter plots: Unlike conventional scatter plots that are two-dimensional, further dimensions can be illustrated through variations in color, size, or shape of the points.

III. RESULT AND ANALYSIS

3.1 Correlation

Correlations are extensively employed statistical methods that underpin several applications, including exploratory data analysis, structural modeling, and data engineering [51]. The objective of correlation analysis is to identify the link between the independent variable (X) and the dependent variable (Y), contingent upon specific data conditions. Equation (5) is employed to ascertain the correlation value, while equation (6) is utilized to derive R Square.

$$r = \frac{\sum(X_i - \bar{X})(Y - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y - \bar{Y})^2}} \tag{5}$$

$$R = r^2 \tag{6}$$

If X is a vector of unspecified classification, with features indexed by i = {1, … , q}, and R(i|Y) represents the weight attributes obtained from the R-squared value of each feature in class Y. The R value is calculated using equation (6). The coefficient of determination, R^2, for all features in vector X relative to the dependent variable Y is calculated using equation (7).

$$R(i|Y) = R \tag{7}$$

with

R	:	r Square
r	:	correlation coefficient value
$\bar{X}$	:	Mean of the attribute X_i
$\bar{Y}$	:	Mean of Y
$R(i\|Y$	:	r Square attribute i to class Y

In classification tasks, posterior probability primarily assesses the likelihood of class membership based on the observed data. This strategy is restrictive as it presumes that all features contribute equally and independently to the classification task, disregarding the strength of relationships between individual features and the target class.

The incorporation of R Square in the equation alters this dynamic by offering a metric for the extent to which each characteristic accounts for the variance in the target class. Incorporating R Square into the model provided further insights into the significance of each feature about the target class, surpassing just probability predictions. If a specific feature exhibits a high R Square value, it is significantly linked with the target class and should be prioritized in the classification decision.

4.1 Posterior Probability

Probability theory is a scientific discipline that employs statistical approaches to comprehend random events [52]. Possibility theory is founded on two fundamental principles. These Prior probability and posterior probability. The posterior probability represents the possibility of an event occurring, calculated after considering all available information or data [52]. To compute the posterior probability for the PC-SVM approach, it is essential to select the maximum value from the various prior probabilities, utilizing conditional probability. The equation (8) is employed to compute the posterior probability.

$$Posterior\ Probability = \max\left(P(Y) \sum_{i=1}^{q} P(X_i|Y) \right) \tag{8}$$

Posterior probability and correlation analysis are potent methodologies that can enhance one another in predictive modeling. This probabilistic output facilitates more nuanced decision-making in categorization problems. Conversely, correlation analysis assesses the magnitude and orientation of correlations among variables. Integrating these strategies can improve model performance by comprehending feature interdependencies and identifying the most pertinent variables. This method enhances classification accuracy and facilitates the interpretation of the underlying data structure, providing insights into the collective influence of features on outcomes. Equation (9) represents the Posterior Probability with Correlations.

$$Posterior\ Probability\ with\ \text{Correlations} = \max\left(P(Y) \sum_{i=1}^{q} (P(X_i|Y)\ R(i|Y)) \right) \tag{9}$$

4.2 Posterior Probability and Correlation-Support Vector Machine (PC-SVM)

The integration of posterior probability and SVM employs the product of prior probability and the cumulative sum of the products of R Square feature i of class Y with the independent probability of all attribute vectors X as input features for the SVM model. The SVM will utilize these features to distinguish the classes by the maximum margin.

4.2.1 Probability Posterior

The posterior probability for each class Y_1 and Y_2 based on the attributes $X = (x_1, x_2, \ldots, x_n)$ of the sample.

$$P(Y_1|X) = P(Y_1)\sum_{i=1}^{q}(P(X_i|Y_1)\quad R(i|Y)) \qquad (10)$$

$$P(Y_2|X) = P(Y_2)\sum_{i=1}^{q}(P(X_i|Y_2)\quad R(i|Y)) \qquad (11)$$

$P(Y_1)$ and $P(Y_2)$ represent the prior probabilities of classes Y_1 and Y_2, while $P(X_1|Y_1)$ and $P(X_2|Y_2)$ denote the conditional probabilities of attribute X inside each class. Following the acquisition of the probabilities $(Y_1|X)$ and $P(Y_2|X)$, these probabilities are utilized as novel input features for the SVM model.

4.2.2 SVM Formulation with Posterior Probability

SVM aims to find a hyperplane $f(z)$ that separates two classes Y_1 and Y_2 based on a new input attribute $z = (P(Y_1|X), P(Y_2|X))$. The decision function for SVM, given a feature vector z is mathematically expressed by equation (12):

$$f(z) = w^\top z + b \qquad (12)$$

subject to,

$$\hat{y} = sign\big(f(z)\big) = sign(w_1.z + b\ ) \qquad (13)$$

Where :

- w is the weight vector of the SVM
- $z = (P(Y_1|X), P(Y_2|X))$ is a feature vector consisting of Naive Bayes posterior probabilities
- b is the bias of the SVM.
- $f(z)$ determine which class the sample belongs to:
  - If $f(z) > 0$, then the sample is predicted as Y_1 (positive class).
  - If $f(z) \leq 0$, then the sample is predicted as Y_2 (negative class).

4.2.3 Combination Formula of SVM and Naive Bayes

The integration of Naive Bayes posterior probabilities into the SVM results in the following combination formula:

$$f(z) = w_1.P(Y_1|X) +\ w_2.P(Y_2|X) + b \qquad (14)$$

Input to SVM: $P(Y_1|X)$ and $P(Y_2|X)$ are the posterior probabilities.

4.2.4 Margin Optimization in SVM

The Support Vector Machine (SVM) optimizes the margin between classes Y_1 and Y_2 by addressing the subsequent optimization problem:

$$\min_{w,b} \tfrac{1}{2}\,\|w\|^2$$

with condition

$$Y_i(w^\top z_i + b) \geq 1\ \forall_i \qquad (16)$$

Where:

- Y_1 is the original class label of the i sample (1 for Y_1, -1 for Y_2).
- $z_i = [P(Y_1|X_i), P(Y_2|X_i)]$ is the feature vector from Naive Bayes for the i sample.

4.2.5 Final Decision

The ultimate prediction is ascertained by the decision function $f(z)$, wherein Naive Bayes and SVM operate in concert:

$$Posterior\ Probability = \max\left(P(Y)\sum_{i=1}^{q}(P(X_i|Y)\ R(i|Y)) \right) \qquad (17)$$

$Posterior\ Probability\ (Y|X)$ is calculated using equation (18):

$$Posterior\ Probability\ (Y|X) = \max\left(\frac{P(Y)\sum_{i=1}^{q}(P(X_i|1)\quad R(i|Y))}{P(X)}\right) \qquad (18)$$

where w is the weight vector, z is the feature vector, and b is the bias

Upon completion of training, the Naive Bayes component calculates the posterior probabilities $P(Y_1|X)$ and $P(Y_2|X)$ for a fresh sample X, which are subsequently input into the SVM. The Support Vector Machine (SVM) derives the ultimate prediction through the decision function:

$$\hat{y} = sign\big(f(z)\big) \tag{19}$$
$$= sign(w_1.P(Y_1)\sum_{i=1}^{q}(P(X_i|\,Y_1) \quad R(i|Y))$$
$$+\ w_2.P(Y_2)\sum_{i=1}^{q}(P(X_i|\,Y_2) \quad R(i|Y)) + b)$$

The ultimate SVM decision function is:

$$\hat{y} = sign\big(f(z)\big) \tag{20}$$
$$= sign(w_1.P(Y_1|X)$$
$$+\ w_2.P(Y_2|X) + b)$$

Where:

- $f(z)$ is the decision score: the sign of $f(z)$ determines the class assignment.
- $P(Y_1|X)$ and $P(Y_2|X)$ are posterior probabilities from Naive Bayes.
- w_1 and w_2 are the SVM weights.
- b is the bias term learned by SVM.

The class prediction is made by evaluating the sign of $f(z)$:

- If $f(z) > 0$, the prediction is Y_1.
- If $f(z) \leq 0$, the prediction is Y_2.

3.2 Data Preprocessing

3.2.1 Missing Value Detection

Table 4. Detection of Missing Values in Yeast Dataset

Attribute	Valid	Missing	Missing (%)
Mcg	1484	0	0
Gvh	1484	0	0
Alm	1484	0	0
Mit	1484	0	0
Erl	1484	0	0
Pox	1484	0	0
Vac	1484	0	0
Nuc	1484	0	0
Class	1484	0	0

Table 4 indicates that the analysis of missing values in the yeast dataset shows a complete dataset, with no missing values in any attributes. Each attribute, specifically Mcg, Gvh, Alm, Mit, Erl, Pox, Vac, Nuc, and Class, contains a total of 1484 valid entries, with no entries recorded as missing. This results in a 0% missing value rate for each attribute, signifying that the dataset is wholly complete. The comprehensiveness of the yeast dataset is a significant advantage, ensuring that the data is ready for further analysis and modeling without necessitating imputation or corrective measures. This reliable data quality enhances confidence in the analytical outcomes and predictive effectiveness of any models developed with this dataset.

Table 5. Detection of Missing Values in Churn Dataset

Attribute	Valid	Missing	Missing (%)
Call Failure	3150	0	0
Complains	3150	0	0
Subscription Length	3150	0	0
Charge Amount	3150	0	0
Seconds of Use	3150	0	0
Frequency of use	3150	0	0
Frequency of SMS	3150	0	0
Distinct Called Numbers	3150	0	0
Age Group	3150	0	0
Tariff Plan	3150	0	0
Status	3150	0	0
Age	3150	0	0
Customer Value	3150	0	0
Churn	3150	0	0

Table 5 indicates that the analysis of missing values in the churn dataset shows no missing data entries for any attributes, which is an admirable outcome. Every attribute, such as Call Failure, Complaints, Subscription Length, Charge Amount, Seconds of Use, Frequency of Use, Frequency of SMS, Distinct Called Numbers, Age Group, Tariff Plan, Status, Age, Customer Value, and Churn, comprises 3150 valid entries. This yields a 0% incidence of missing values across all attributes. The comprehensive dataset indicates a high degree of data integrity, since the lack of missing values obviates the necessity for data imputation or corrections. The completeness of data is essential for performing comprehensive analysis and developing predictive models, guaranteeing that the insights obtained are grounded in a solid foundation and improving the dependability of conclusions related to customer behavior and churn determinants.

3.3 Data Transformation

Table 6. Data Transformation of the Yeast Dataset

Attribute	Type data Before Transformation	Type Data After Transformation
Mcg	float64	Category
Gvh	float64	Category
Alm	float64	Category
Mit	float64	Category
Erl	float64	Category
Pox	float64	Category
Vac	float64	Category
Nuc	float64	Category
Class	float64	Category

Table 6 displays the outcomes of the Yeast Dataset Transformation. The conversion of the yeast dataset was essential for facilitating posterior probability calculations and correlation analysis, as it entailed changing several characteristics from float64 to Category type. Initially, qualities such as Mcg, Gvh, Alm, Mit, Erl, Pox, Vac, Nuc, and Class were denoted as continuous numerical values, potentially distorting their category essence in subsequent analysis. Transforming these features into categorical data types enables the analysis to effectively analyze and manage the interactions among various classes without supposing a linear relationship, which is typical with continuous variables. This change is crucial for computing posterior probabilities, as the models must regard these attributes as discrete categories instead of continuous scales. Furthermore, categorizing the data improves correlation analysis by ensuring the statistical methods employed are suitable for categorical data, facilitating the detection of significant connections among various yeast features. This data transformation enhances the dataset for efficient statistical modeling and guarantees precise outcomes in posterior probability and correlation calculations.

Table 7. Data Transformation of the Churn Dataset

Attribute	Type data Before Transformation	Type Data After Transformation
Call Failure	Int64	Category
Complains	Int64	Category
Subscription Length	Int64	Category
Charge Amount	Int64	Category
Seconds of Use	Int64	Category
Frequency of use	Int64	Category
Frequency of SMS	Int64	Category
Distinct Called Numbers	Int64	Category
Age Group	Int64	Category
Tariff Plan	Int64	Category
Status	Int64	Category
Age	Int64	Category
Customer Value	float64	Category
Churn	Int64	Category

Table 7 displays the outcomes of the churn dataset transformation. The conversion of the churn dataset was essential for further probability calculations and correlation analysis, requiring the transformation of several properties from their original integer (Int64) and floating-point (float64) types into categorical kinds. Initially, qualities such as Call Failure, Complaints, and Subscription Length were denoted as integer values, potentially suggesting a numerical relationship that does not adequately convey their category essence. Transforming these attributes into categorical types enhances the dataset's alignment with the intended analysis, enabling each feature to be regarded as a discrete category instead of a continuous variable. This is especially crucial for posterior probability estimation, as categorical data can yield more explicit insights regarding churn likelihood based on characteristics like Age Group or Tariff Plan, devoid of the false connotations of numerical scales. Moreover, the transformation improves correlation analysis by allowing the examination of correlations between categorical variables, so permitting a more significant interpretation of how various features interact and affect customer churn. This data transformation enhances the dataset for efficient statistical modeling, guaranteeing precise outcomes in posterior probability and correlation analyses about customer behavior.

3.4 Performance SVM With Imbalanced Dataset

The results of performing SVM with an imbalanced dataset are displayed in Table 8, Figure 3, Figure 4, Table 9, Figure 5, and Figure 6. The performance evaluation include metrics such as accuracy, precision, recall, and FI-score.

Table 8. SVM Classification Report on Imbalanced Yeast Dataset

	Precision	Recall	f1-score	support
-1	0.74	0.96	0.84	1055
1	0.67	0.19	0.29	429
accuracy			0.74	1484
macro avg	0.71	0.57	0.57	1484
weighted avg	0.72	0.74	0.68	1484

Table 8 indicates that the SVM model's accuracy on the unbalanced yeast dataset is 73.79%, which is suboptimal owing to the class imbalance. The majority class (-1) has 1055 instances, whereas the minority class (1) contains merely 429, resulting in a bias in the model towards the majority class. The model has commendable performance for the majority class, achieving a precision of 0.74, a high recall of 0.96, and a robust F1-score of 0.84. Nonetheless, the performance for the minority class is significantly inferior, with a precision of 0.67 and a notably low recall of 0.19, culminating in a subpar F1-score of 0.29.

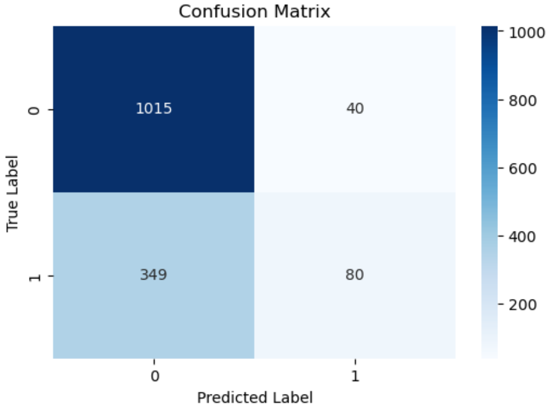

Figure 3. SVM Confusion Matrix on Imbalanced Yeast Dataset

The confusion matrix in Figure 3 reveals numerous False Negatives for the minority class, signifying the model's difficulty in recognizing these situations.

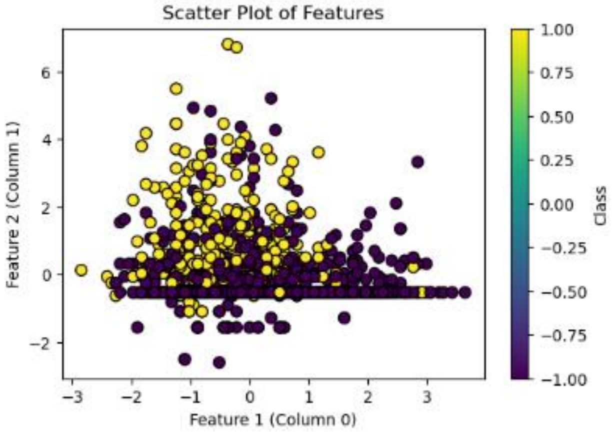

Figure 4: SVM Scatter Plot on Imbalanced Yeast Dataset

The scatter plot in Figure 4 demonstrating evident separation for the majority class while exhibiting less significant separation for the minority class, resulting in misclassifications. The macro average indicates a precision of 0.71 and a recall of 0.57, illustrating the imbalance, but the weighted average shows a modest improvement although still underscores the challenges associated with the minority class.The imbalance yields marginally improved scores; however, the low recall for class 1 persists as a significant concern.

Table 9. SVM Classification Report on Imbalanced Churn Dataset

	precision	Recall	f1-score	support
-1	0.84	1.00	0.91	2654
1	0.00	0.00	0.00	495
accuracy			0.84	3149
macro avg	0.42	0.50	0.46	3149
weighted avg	0.71	0.84	0.77	3149

Table 9 illustrates that the performance of the SVM model on the unbalanced churn dataset underscores the problem of class imbalance, achieving an overall accuracy of 84.28%. Nonetheless, this accuracy is deceptive owing to the significant disparity between the majority class (-1, non-churn) including 2654 instances and the minority class (1, churn) consisting of 495 instances. The model exhibits strong performance for the majority class, attaining a precision of 0.84, a recall of 100%, and an F1-score of 0.91, indicating it accurately detects almost all non-churners. Nevertheless, the confusion matrix in Figure 5 indicates that the model erroneously categorizes all churn events as non-churn, yielding a recall of 0% for the minority class.

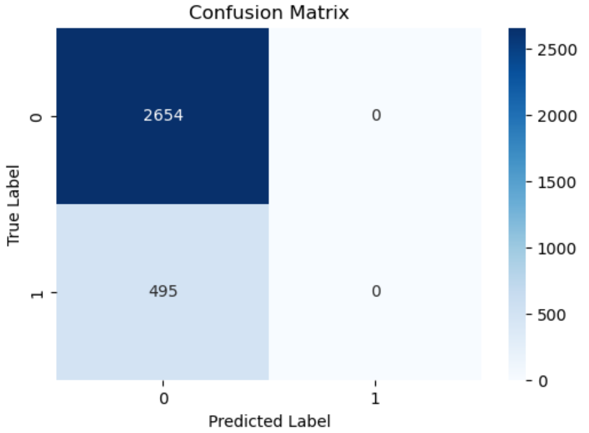

Figure 5: SVM Confusion Matrix on Imbalanced Churn Dataset

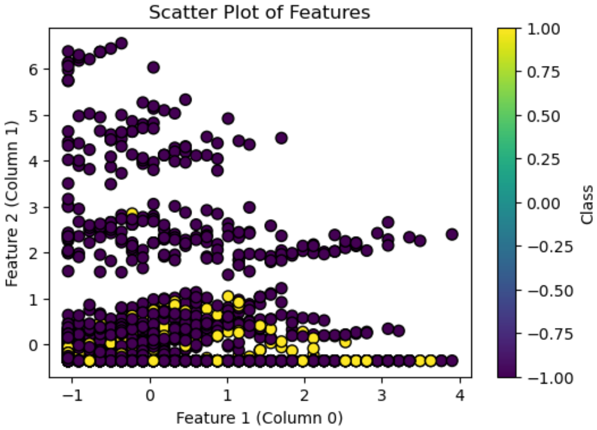

Figure 6: SVM Scatter Plot on Imbalanced Churn Dataset

The scatter figure underscores this, Figure 6 indicating that the decision boundary favors the majority class, resulting in numerous churn events being misclassified. The model is unable to predict churners, yielding a recall and F1-score of 0 for class 1. The macro average scores (accuracy 0.42, recall 0.50, F1-score 0.46) underscore the model's inadequate generalization across both classes, however the weighted average F1-score of 0.77 indicates robust performance on the majority class, concealing its deficiencies with the minority class.

3.5 Performance PC-SVM With Imbalanced Dataset

The results of performance PC-SVM with an imbalanced dataset are presented Table 10, Figure 7, Figure 8, Table 11, Figure 9, and Figure 10. The performance metrics assessed include accuracy, precision, recall, and F1-score.

Table 10. PC-SVM Classification Report on Imbalanced Yeast Dataset

	precision	Recall	f1-score	support
-1	1.00	1.00	1.00	1423
1	1.00	1.00	1.00	61
accuracy			1.00	1484
macro avg	1.00	1.00	1.00	1484
weighted avg	1.00	1.00	1.00	1484

Table 10 illustrates that the PC-SVM model exhibits outstanding performance on the unbalanced yeast dataset, with a flawless accuracy of 100%. The algorithm accurately predicted all cases without errors, showcasing its efficacy and dependability in classifying the yeast data. The confusion matrix validates this robust performance, since all cases from both classes (-1 and 1) are accurately categorized, with no errors in categorization. The matrix in Figure 7 exclusively displays True Positives and True Negatives, indicating the absence of False Positives and False Negatives. This signifies that the PC-SVM model can effectively differentiate between the two classes, despite the class imbalance (61 instances of class 1 compared to 1423 instances of class -1).

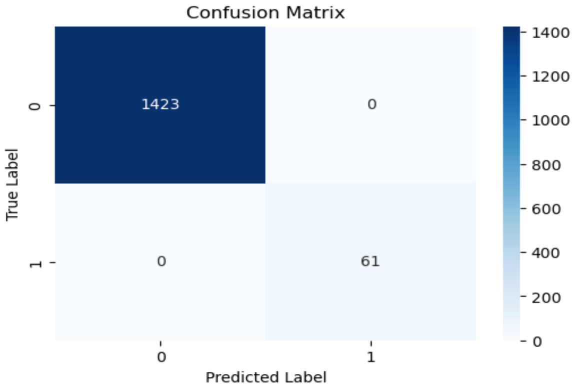

Figure 7: PC-SVM Confusion Matrix on Imbalanced Yeast Dataset

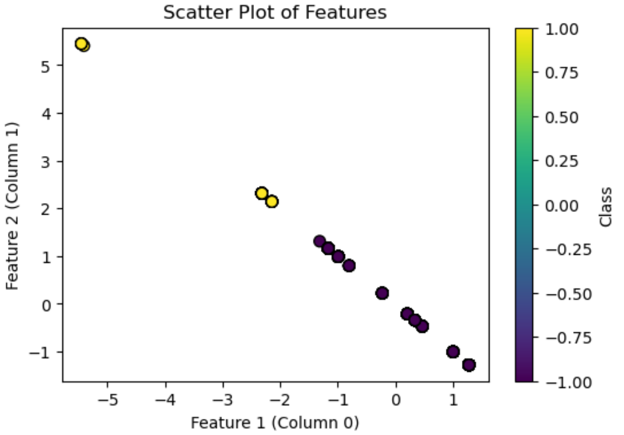

Figure 8: PC-SVM Scatter Plot on Imbalanced Yeast Dataset

Figure 8 a scatter plot of the results would probably exhibit a distinct demarcation between the two classes, with no overlap. examples of class -1 would create a dense cluster, whilst examples of class 1 would be distinctly identifiable, hence enhancing the model's capacity to address class imbalance and accurately categorize both categories. The classification report indicates flawless results for both classes, with precision, recall, and F1-score all at 1.00. The model exhibits complete accuracy in its predictions, devoid of false positives or false negatives, and maintains an ideal equilibrium between precision and recall. The support values indicate 1423 occurrences for class -1 and 61 for class 1. Notwithstanding the imbalance, the PC-SVM approach effectively manages the gap, attaining flawless categorization for both classes. The macro and weighted average scores are also 1.00, further emphasizing the model's uniform performance throughout the dataset. This exceptional performance illustrates the efficacy of PC-SVM in managing imbalanced datasets for classification tasks.

Table 11. PC-SVM Classification Report on Imbalanced Churn Dataset

	precision	Recall	f1-score	support
-1	1.00	1.00	1.00	1867
1	1.00	1.00	1.00	1283
accuracy			1.00	3150
macro avg	1.00	1.00	1.00	3150
weighted avg	1.00	1.00	1.00	3150

Table 11 illustrates that the PC-SVM model exhibits outstanding performance on the imbalanced churn dataset, with flawless accuracy of 100%. The model precisely classified each case in the dataset, differentiating between the two classes without errors, despite the class imbalance.

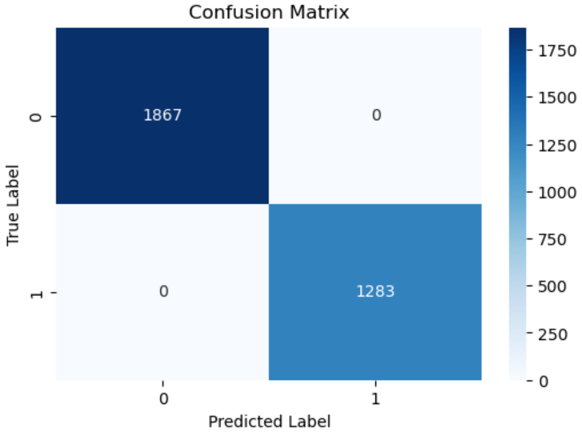

Figure 9: PC-SVM Confusion Matrix on Imbalanced Churn Dataset

The confusion matrix in Figure 9 validates this, indicating that all cases for both groups (-1 for non-churned customers and 1 for churned customers) were accurately identified. The absence of false positives and false negatives illustrates the model's robust performance and capacity to manage class imbalance.

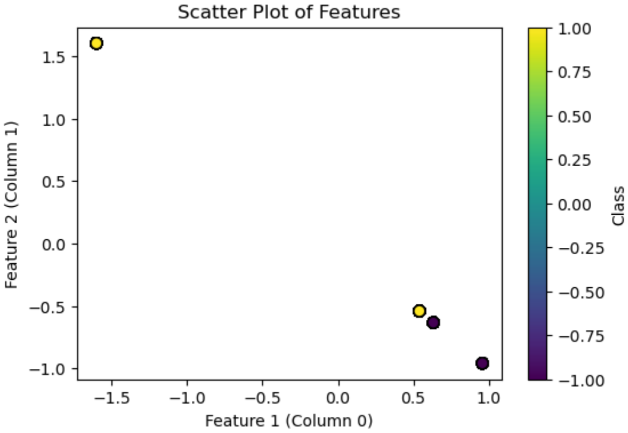

Figure 10: PC-SVM Scatter Plot on Imbalanced Yeast Dataset

Figure 10 a scatter plot would likely exhibit a clear delineation between the two classes, with distinct clusters for churned and non-churned clients. The absence of overlap would visually validate that the model correctly recognized all churned consumers without misclassifying any non-churned individuals, hence reinforcing the model's efficacy. The classification report verifies flawless outcomes, with precision, recall, and F1-scores all at 1.00 for each class. This signifies that the model accurately identified all true positives while evading both false positives and false negatives, achieving an impeccable equilibrium between precision and recall. The support data indicate 1867 instances of class -1 and 1283 instances of class 1, reflecting a little imbalance. Nonetheless, the PC-SVM model accurately categorized all occurrences, demonstrating its proficiency in managing imbalanced datasets. The macro and weighted averages are both 1.00, indicating uniform performance across the two classes. This remarkable performance underscores the PC-SVM's capability as an effective instrument for categorizing imbalanced datasets, including customer churn prediction.

VI. CONCLUSION

This study proposes a hybrid approach to improving Support Vector Machine (SVM) classification performance on imbalanced datasets by integrating posterior probability and correlation analysis. Imbalanced data often hampers the accuracy of traditional classifiers, as minority classes are underrepresented and frequently misclassified. The introduced Posterior Probability and Correlation-SVM (PC-SVM) method enhances minority class detection by combining posterior probabilities, which measure class likelihood, with attribute correlation coefficients to weigh feature importance. The study demonstrates the effectiveness of the PC-SVM model on the Yeast and Churn datasets, achieving significantly improved accuracy, precision, recall, and F1-scores for minority classes. This approach highlights the potential of fusion techniques in addressing the challenges posed by imbalanced datasets, providing a robust framework for enhancing classification performance.

The hybrid PC-SVM model integrating posterior probability and correlation techniques demonstrates exceptional performance in addressing class imbalance challenges. By incorporating attribute weighting through correlation analysis and transforming input features with posterior probabilities, the method effectively enhances the sensitivity and accuracy of SVM models for minority classes. Experimental results on the Yeast and Churn datasets highlight the model's ability to achieve balanced classification metrics across all classes, resolving the limitations of traditional SVMs. This study underscores the importance of tailored techniques in machine learning for tackling dataset imbalances, paving the way for more accurate and fair predictive modeling.

REFERENCES

[1] J. Alcaraz, M. Labbé, and M. Landete, “Support Vector Machine with feature selection: A multiobjective approach,” *Expert Syst. Appl.*, vol. 204, no. May, p. 117485, 2022, doi:

10.1016/j.eswa.2022.117485.

[2] J. Liu, "Fuzzy support vector machine for imbalanced data with borderline noise," *Fuzzy Sets Syst.*, vol. 1, pp. 1–10, 2020, doi: 10.1016/j.fss.2020.07.018.

[3] C. Wu, N. Wang, and Y. Wang, "Increasing Minority Recall Support Vector Machine Model for Imbalanced Data Classification," *Discret. Dyn. Nat. Soc.*, vol. 2021, 2021, doi: 10.1155/2021/6647557.

[4] H. Liu, Z. Liu, W. Jia, D. Zhang, and J. Tan, "A Novel Imbalanced Data Classification Method Based on Weakly Supervised Learning for Fault Diagnosis," *IEEE Trans. Ind. Informatics*, vol. 18, no. 3, pp. 1583–1593, 2022, doi: 10.1109/TII.2021.3084132.

[5] S. Shaikh, S. M. Daudpota, A. S. Imran, and Z. Kastrati, "Towards improved classification accuracy on highly imbalanced text dataset using deep neural language models," *Appl. Sci.*, vol. 11, no. 2, pp. 1–20, 2021, doi: 10.3390/app11020869.

[6] R. A. Hamad, M. Kimura, and J. Lundström, "Efficacy of Imbalanced Data Handling Methods on Deep Learning for Smart Homes Environments," *SN Comput. Sci.*, vol. 1, no. 4, pp. 1–10, 2020, doi: 10.1007/s42979-020-00211-1.

[7] V. Rupapara, F. Rustam, H. F. Shahzad, A. Mehmood, I. Ashraf, and G. S. Choi, "Impact of SMOTE on Imbalanced Text Features for Toxic Comments Classification Using RVVC Model," *IEEE Access*, vol. 9, pp. 78621–78634, 2021, doi: 10.1109/ACCESS.2021.3083638.

[8] H. Qin, H. Zhou, and J. Cao, "Imbalanced learning algorithm based intelligent abnormal electricity consumption detection," *Neurocomputing*, vol. 402, no. xxxx, pp. 112–123, 2020, doi: 10.1016/j.neucom.2020.03.085.

[9] S. S. Mullick, S. Datta, S. G. Dhekane, and S. Das, "Appropriateness of performance indices for imbalanced data classification: An analysis," *Pattern Recognit.*, vol. 102, p. 107197, 2020, doi: 10.1016/j.patcog.2020.107197.

[10] H. Shamsudin, U. K. Yusof, A. Jayalakshmi, and M. N. Akmal Khalid, "Combining oversampling and undersampling techniques for imbalanced classification: A comparative study using credit card fraudulent transaction dataset," *IEEE Int. Conf. Control Autom. ICCA*, vol. 2020-Octob, pp. 803–808, 2020, doi: 10.1109/ICCA51439.2020.9264517.

[11] K. H. Kim and S. Y. Sohn, "Hybrid neural network with cost-sensitive support vector machine for class-imbalanced multimodal data," *Neural Networks*, vol. 130, pp. 176–184, 2020, doi: 10.1016/j.neunet.2020.06.026.

[12] X. Tao *et al.*, "Affinity and class probability-based fuzzy support vector machine for imbalanced data sets," *Neural Networks*, vol. 122, pp. 289–307, 2020, doi: 10.1016/j.neunet.2019.10.016.

[13] C. Jimenez-Castaño, A. Alvarez-Meza, and A. Orozco-Gutierrez, "Enhanced automatic twin support vector machine for imbalanced data classification," *Pattern Recognit.*, vol. 107, 2020, doi: 10.1016/j.patcog.2020.107442.

[14] R. Abo Zidan and G. Karraz, "Gaussian Pyramid for Nonlinear Support Vector Machine," *Appl. Comput. Intell. Soft Comput.*, vol. 2022, 2022, doi: 10.1155/2022/5255346.

[15] Y. S. Solanki *et al.*, "A hybrid supervised machine learning classifier system for breast cancer prognosis using feature selection and data imbalance handling approaches," *Electron.*, vol. 10, no. 6, pp. 1–16, 2021, doi: 10.3390/electronics10060699.

[16] R. Kumar R *et al.*, "Investigation of nano composite heat exchanger annular pipeline flow using CFD analysis for crude oil and water characteristics," *Case Stud. Therm. Eng.*, vol. 49, p. 104908, 2023, doi: 10.1016/j.csite.2023.103297.

[17] X. W. Liang, A. P. Jiang, T. Li, Y. Y. Xue, and G. T. Wang, "LR-SMOTE — An improved unbalanced data set oversampling based on K-means and SVM," *Knowledge-Based Syst.*, vol. 196, 2020, doi: 10.1016/j.knosys.2020.105845.

[18] C. Wang, C. Deng, Z. Yu, D. Hui, X. Gong, and R. Luo, "Adaptive ensemble of classifiers with regularization for imbalanced data classification," *Inf. Fusion*, vol. 69, no. September 2020, pp. 81–102, 2021, doi: 10.1016/j.inffus.2020.10.017.

[19] G. A. Pradipta, R. Wardoyo, A. Musdholifah, and I. N. H. Sanjaya, "Improving classifiaction performance of fetal umbilical cord using combination of SMOTE method and multiclassifier voting in imbalanced data and small dataset," *Int. J. Intell. Eng. Syst.*, vol. 13, no. 5, pp. 441–454, 2020, doi: 10.22266/ijies2020.1031.39.

[20] F. Feng, K. C. Li, J. Shen, Q. Zhou, and X. Yang, "Using Cost-Sensitive Learning and Feature Selection Algorithms to Improve the Performance of Imbalanced Classification," *IEEE Access*, vol. 8, pp. 69979–69996, 2020, doi: 10.1109/ACCESS.2020.2987364.

[21] J. B. Wang, C. A. Zou, and G. H. Fu, "AWSMOTE: An SVM-Based Adaptive Weighted SMOTE for Class-Imbalance Learning," *Sci. Program.*, vol. 2021, 2021, doi:

10.1155/2021/9947621.

[22] S. Sreejith, H. Khanna Nehemiah, and A. Kannan, "Clinical data classification using an enhanced SMOTE and chaotic evolutionary feature selection," *Comput. Biol. Med.*, vol. 126, no. September, p. 103991, 2020, doi: 10.1016/j.compbiomed.2020.103991.

[23] S. Ketu and P. K. Mishra, "Empirical Analysis of Machine Learning Algorithms on Imbalance Electrocardiogram Based Arrhythmia Dataset for Heart Disease Detection," *Arab. J. Sci. Eng.*, vol. 47, no. 2, pp. 1447–1469, 2022, doi: 10.1007/s13369-021-05972-2.

[24] H. I. Hussein and S. A. Anwar, "Imbalanced Data Classification Using Support Vector Machine Based on Simulated Annealing for Enhancing Penalty Parameter," *Period. Eng. Nat. Sci.*, vol. 9, no. 2, pp. 1030–1037, 2021, doi: 10.21533/pen.v9i2.2031.

[25] S. Rezvani and X. Wang, "Class imbalance learning using fuzzy ART and intuitionistic fuzzy twin support vector machines," *Inf. Sci. (Ny).*, vol. 578, pp. 659–682, 2021, doi: 10.1016/j.ins.2021.07.010.

[26] B. B. Hazarika and D. Gupta, "Density-weighted support vector machines for binary class imbalance learning," *Neural Comput. Appl.*, vol. 33, no. 9, pp. 4243–4261, 2021, doi: 10.1007/s00521-020-05240-8.

[27] Y. Ünal and M. N. Dudak, "Classification of Covid-19 Dataset with Some Machine Learning Methods," *J. Amasya Univ. Inst. Sci. Technol.*, vol. 1, no. 1, pp. 36–44, 2020.

[28] M. C. Mihaescu and P. S. Popescu, "Review on publicly available datasets for educational data mining," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 11, no. 3, pp. 1–16, 2021, doi: 10.1002/widm.1403.

[29] S. Yadav and G. P. Bhole, "Handling Imbalanced Dataset Classification in Machine Learning," *2020 IEEE Pune Sect. Int. Conf. PuneCon 2020*, pp. 38–43, 2020, doi: 10.1109/PuneCon50868.2020.9362471.

[30] S. Rahman, M. Hasan, and A. K. Sarkar, "Prediction of Brain Stroke using Machine Learning Algorithms and Deep Neural Network Techniques," *Eur. J. Electr. Eng. Comput. Sci.*, vol. 7, no. 1, pp. 23–30, 2023, doi: 10.24018/ejece.2023.7.1.483.

[31] N. Matondang and N. Surantha, "Effects of oversampling SMOTE in the classification of hypertensive dataset," *Adv. Sci. Technol. Eng. Syst.*, vol. 5, no. 4, pp. 432–437, 2020, doi: 10.25046/AJ050451.

[32] N. Anđelić, S. Baressi Šegota, and Z. Car, "Improvement of Malicious Software Detection Accuracy through Genetic Programming Symbolic Classifier with Application of Dataset Oversampling Techniques," *Computers*, vol. 12, no. 12, 2023, doi: 10.3390/computers12120242.

[33] D. Liu, R. Sun, and H. Ren, "Efficient Fraud Detection Classification: Class Imbalanceand Attribute Correlations," *Thr Front. Soc. Sci. Technol.*, vol. 2, no. 11, pp. 96–103, 2020, doi: 10.25236/FSST.2020.021115.

[34] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," *2020 11th Int. Conf. Inf. Commun. Syst. ICICS 2020*, pp. 243–248, 2020, doi: 10.1109/ICICS49469.2020.239556.

[35] S. Ilyas, S. Zia, U. M. Butt, S. Letchmunan, and Z. un Nisa, "Predicting the future transaction from large and imbalanced banking dataset," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 1, pp. 273–286, 2020, doi: 10.14569/ijacsa.2020.0110134.

[36] V. Khattri and S. K. Nayak, "Performance Improvement of Classification Model with Imbalanced Dataset," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 13, pp. 402–408, 2021, [Online]. Available: https://coloradotech.idm.oclc.org/login?url=https://www.proquest.com/scholarly-journals/performance-improvement-classification-model-with/docview/2623929968/se-2%0Ahttps://media.proquest.com/media/hms/PFT/1/N218M?_a=ChgyMDIzMDMwNTAwMTI1MjExMjo0NjIwNjMSBzE

[37] Asniar, N. U. Maulidevi, and K. Surendro, "SMOTE-LOF for noise identification in imbalanced data classification," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3413–3423, 2022, doi: 10.1016/j.jksuci.2021.01.014.

[38] S. Strasser and M. Klettke, "Transparent Data Preprocessing for Machine Learning," *HILDA 2024 - Work. Human-In-the-Loop Data Anal. Co-located with SIGMOD 2024*, 2024, doi: 10.1145/3665939.3665960.

[39] J. Nalic and A. Svraka, "Importance of data pre-processing in credit scoring models based on data mining approaches," *2018 41st Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2018 - Proc.*, pp. 1046–1051, 2022, doi: 10.23919/MIPRO.2018.8400191.

[40] H. F. Tayeb, M. Karabatak, and C. Varol, "Time Series Database Preprocessing for Data Mining Using Python," *8th Int. Symp. Digit. Forensics Secur. ISDFS 2020*, pp. 20–23, 2020, doi:

10.1109/ISDFS49300.2020.9116260.

[41] S. Albahra *et al.*, "Artificial intelligence and machine learning overview in pathology & laboratory medicine: A general review of data preprocessing and basic supervised concepts," *Semin. Diagn. Pathol.*, vol. 40, no. 2, pp. 71–87, 2023, doi: 10.1053/j.semdp.2023.02.002.

[42] Z. Liu, "Research on data preprocessing method for artificial intelligence algorithm based on user online behavior," *J. Comput. Electron. Inf. Manag.*, vol. 12, no. 3, pp. 74–78, 2024, doi: 10.54097/qf6fv8j1.

[43] A. Q. Md, S. Kulkarni, C. J. Joshua, T. Vaichole, S. Mohan, and C. Iwendi, "Enhanced Preprocessing Approach Using Ensemble Machine Learning Algorithms for Detecting Liver Disease," *Biomedicines*, vol. 11, no. 2, 2023, doi: 10.3390/biomedicines11020581.

[44] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Glob. Transitions Proc.*, vol. 3, no. 1, pp. 91–99, 2022, doi: 10.1016/j.gltp.2022.04.020.

[45] H. T. Duong and T. A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," *Comput. Soc. Networks*, vol. 8, no. 1, pp. 1–16, 2021, doi: 10.1186/s40649-020-00080-x.

[46] C. Fan, M. Chen, X. Wang, J. Wang, and B. Huang, "A Review on Data Preprocessing Techniques Toward Efficient and Reliable Knowledge Discovery From Building Operational Data," *Front. Energy Res.*, vol. 9, no. March, pp. 1–17, 2021, doi: 10.3389/fenrg.2021.652801.

[47] V. Chernykh, A. Stepnov, and B. O. Lukyanova, "Data preprocessing for machine learning in seismology," *CEUR Workshop Proc.*, vol. 2930, pp. 119–123, 2021.

[48] A. J. Mohammed, "Improving Classification Performance for a Novel Imbalanced Medical Dataset using SMOTE Method," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 3, pp. 3161–3172, 2020, doi: 10.30534/ijatcse/2020/104932020.

[49] A. Kulkarni, D. Chong, and F. A. Batarseh, *Foundations of data imbalance and solutions for a data democracy*. Elsevier Inc., 2020. doi: 10.1016/B978-0-12-818366-3.00005-8.

[50] A. Arafa, N. El-Fishawy, M. Badawy, and M. Radad, "RN-SMOTE: Reduced Noise SMOTE based on DBSCAN for enhancing imbalanced data classification," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5059–5074, 2022, doi: 10.1016/j.jksuci.2022.06.005.

[51] D. Makowski, M. Ben-Shachar, I. Patil, and D. Lüdecke, "Methods and Algorithms for Correlation Analysis in R," *J. Open Source Softw.*, vol. 5, no. 51, p. 2306, 2020, doi: 10.21105/joss.02306.

[52] M. S. Vural and M. Telceken, "Modification of posterior probability variable with frequency factor according to Bayes Theorem," *J. Intell. Syst. with Appl.*, vol. 5, no. 1, pp. 19–26, 2022, doi: 10.54856/jiswa.202205195.