INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING AND COMPUTER SYSTEMS
E-ISSN: 2180-0650
VOLUME 11, ISSUE 1, 2025, pp. 16 - 31
DOI: https://doi.org/10.15282/ijsecs.11.1.2025.2.0134

RESEARCH ARTICLE

Optimizing support vector machine for imbalanced datasets by combining posterior probability and correlation methods

Canggih Ajika Pamungkas[1, 2]*, Megat Farez Zuhairi[1]

[1]Malaysian Institute of Information Technology, Universiti Kuala Lumpur, 1016 Jalan Sultan Ismail, 50250 Kuala Lumpur, Malaysia
[2]Politeknik Indonusa Surakarta, Jalan KH. Samanhudi No 31, Surakarta, Indonesia

Abstract –The challenge of classifying imbalanced data persists in machine learning, particularly in critical applications such as medical diagnosis, fraud detection, and anomaly identification, where detecting the minority class is essential. Conventional classifiers, such as support vector machines (SVM), tend to favor the majority class, leading to reduced sensitivity in identifying minority instances. This study introduces posterior probability and correlation-support vector machine (PC-SVM), a novel approach that integrates posterior probability estimation with correlation analysis to enhance the performance of SVMs on imbalanced datasets. Unlike traditional SVM models, which struggle with class imbalance and require additional data-balancing techniques, PC-SVM dynamically adjusts classification thresholds using posterior probability values and correlation-weighted features, simplifying the classification process while improving its effectiveness. The effectiveness of PC-SVM was evaluated using multiple imbalanced datasets from KEEL, UCI, and Kaggle repositories. The results demonstrate that PC-SVM achieves 100% recall for the minority class, significantly outperforming the traditional SVM, which attained only 80% recall on average. This 20% improvement in recall underscores PC-SVM's ability to mitigate the imbalance issue without relying on oversampling or cost-sensitive adjustments. Furthermore, PC-SVM exhibits consistent performance across various evaluation metrics, including accuracy, precision, recall, and F1-score, ensuring robust classification results. By improving the detection of minority classes, PC-SVM offers a transformative solution for real-world applications that demand high sensitivity in identifying rare but crucial instances. Its ability to maintain classification integrity without additional balancing techniques positions it as a valuable model for industries such as healthcare, finance, and cybersecurity, where accurate minority class recognition is critical.

Article History
Received : 1 January 2025
Revised : 31 January 2025
Accepted : 28 April 2025
Published : 2 May 2025

Keywords
Supervised classification
Imbalanced dataset
Posterior
Probability
Correlation

1. Introduction

Supervised categorization uses a training dataset and statistical learning techniques to classify items into distinct classes and apply this knowledge to categorize new data [1]. Supervised learning, a fundamental aspect of machine learning, uses algorithms to identify patterns in data by leveraging known independent and dependent variables to predict future outcomes; supervised classification treats class labels as parameters while addressing challenges like class distribution disparity [2]. The importance of imbalanced data classification is growing in machine learning [3]. A dataset is considered imbalanced when one class significantly outnumbers the other, with the minority class referred to as the positive (+) class and the majority class as the negative (−) class in data categorization [3]. Class imbalance has garnered significant attention in recent studies [4]-[10]. Sampling techniques transform the distribution of imbalanced data into a balanced distribution [11]. Undoubtedly, learning from imbalanced data sets is a significant obstacle in data mining. While conventional support vector machines can typically perform well on classification problems with imbalanced data sets, they treat all training samples equally during learning. This can bias the final decision boundary toward the majority class, particularly when outliers or noise are present [12]. Imbalanced data categorization occurs when one class has more examples than the other, with the majority class often overshadowing the minority class, which conventional classifiers may treat as noise, leading to bias toward the majority class and prompting the development of various methods to address this issue [13].

The support vector machine (SVM) is an efficient machine learning tool known for its speed, simplicity, reliability, and ability to produce accurate classification results [14]. The SVM generates a model based on the available sample sizes of each class. The SVM learning formulation is derived from the ideas of structural risk minimization. SVM can be used to mitigate the generalization error, thereby improving the performance of data outside the training set [15]. The goal of SVM is to identify the hyperplane that separates two classes in a vector space [16]. The separating hyperplane lies between two parallel hyperplanes, with one placing vectors of the first class above it and the other placing vectors of the second class below it, where the margin is the distance between these hyperplanes; in cases where misclassifications are allowed for improved generalization, the margin is "soft," while SVM remains a highly efficient method for supervised classification [1].

Traditional classification approaches assume equal probabilities for data from different classes; however, in real-world scenarios, minority classes may have fewer data points than majority classes, causing traditional algorithms to bias toward majority classes and reducing the accuracy of minority-class classification [2]. To address the challenges of imbalanced data, various techniques have been proposed, categorized into three groups: data-level techniques that modify the sample probabilities through oversampling or undersampling to balance the dataset, algorithm-level techniques that adjust

*CORRESPONDING AUTHOR | Canggih Ajika Pamungkas | ✉ pamungkas.canggih@s.unikl.edu.my
© 2025 The Author(s). Published by Universiti Malaysia Pahang Al-Sultan Abdullah Press. This is an open access article under the CC BY-NC 4.0 license

classification systems with cost-sensitive approaches to penalize misclassifying minority samples more heavily, and fusion approaches that combine different tactics, such as sampling and cost-sensitive methods, to tackle the imbalance issue [2].

The proposed posterior probability and correlation-support vector machine (PC-SVM) improves upon traditional SVM-based methods by integrating posterior probability estimation and correlation-weighted feature analysis, addressing the longstanding issue of class imbalance in machine learning. Unlike standard SVM, which constructs decision boundaries without considering class distribution, PC-SVM dynamically adjusts classification thresholds using posterior probabilities to improve minority-class recognition. Furthermore, unlike cost-sensitive SVM or resampling techniques (e.g., SMOTE), which require manual parameter tuning or risk overfitting, PC-SVM automatically balances the class representation without modifying the dataset. Existing SVM solutions struggle with biased decision boundaries that favor the majority class, limiting their effectiveness in real-world applications such as fraud detection and medical diagnosis, where detecting the minority class is crucial. PC-SVM fills this gap by using correlation analysis to refine feature selection, ensuring that the most relevant features drive classification and improve recall, accuracy, and generalization across datasets.

This study introduces a novel classification method, PC-SVM, that integrates posterior probability and correlation techniques to improve SVM performance on imbalanced datasets. Section 1 discusses the background and motivation for addressing class imbalance in machine learning. Section 2 reviews related work and existing approaches to tackle this issue. Section 3 outlines the research methodology, including data sources, preprocessing, modeling, and evaluation techniques. Section 4 presents and analyzes the experimental results, demonstrating that PC-SVM outperforms traditional SVM methods. The final section summarizes the findings and highlights PC-SVM's contributions and potential for real-world applications.

2. Related Work

A key challenge in classification learning systems is the class distribution disparity, where one or more classes have a high frequency of instances, while others are underrepresented, causing conventional algorithms, including SVM classifiers, to perform well on the dominant class but exhibit bias and fail to incorporate data distribution information in addressing class imbalance [17]. Li et al. [18] developed a novel technique called ant colony optimization resampling (ACOR) to address class imbalance. ACOR consists of two stages: first, an oversampling technique is applied to balance the dataset, followed by the use of an ant colony optimization algorithm to select a suboptimal subset, enabling the creation of an optimal training set, which has shown superior performance compared to other oversampling methods, although challenges remain, such as quick descent into local optima, slow convergence, and low precision in convergence [19]. As mentioned before, the SVM classifier exhibits bias towards the dominant class because of the class imbalance. Furthermore, current SVM-based approaches to address class imbalance lack information about the data distribution.

Richhariya and Tanveer [17] proposed a Reduced Universum Twin Support Vector Machine for Class Imbalance Learning (RUTSVM-CIL), motivated by the concept of prior information about data distribution. This study combines universum learning with Support Vector Machine to address class imbalance, using oversampling and undersampling techniques alongside universum data points for prior information, and employs a compact rectangular kernel matrix to reduce computational time and storage, with the RUTSVM-CIL method demonstrating superior generalization performance and minimal computational cost on diverse datasets [17]. The sampling strategy modifies the dataset before learning, using oversampling to increase the minority class size by adding data and undersampling to decrease the majority class size by removing data, though undersampling can lose valuable information [20].

Classification algorithms often struggle with imbalanced datasets, making effective classification challenging, particularly in detecting loose particles in sealed electronic components, which is addressed using the Synthetic Minority Over-sampling Technique, a standard oversampling method [21], The LR-SMOTE algorithm, designed to create new samples close to the center of the dataset and avoid generating outliers or altering the distribution, was tested on publicly available UCI datasets and custom data, showing superior performance over SMOTE in terms of G-means, F-measure, and AUC, although SMOTE's insensitivity to majority class distribution can lead to the creation of redundant minority samples, worsening issues for borderline and noisy instances [19]. Tao et al. [12] presented a novel approach called Affinity and Probability-based Fuzzy Support Vector Machine (ACF SVM). The proposed ACF SVM approach uses an SVDD model trained on majority-class samples in kernel space to detect anomalies and edge samples, applies the kernel k-nearest neighbor technique to reduce noise impact, and improves classification by prioritizing minority-class importance while achieving superior performance on imbalanced UCI datasets. Class imbalance in real-world datasets biases models toward the dominant class, reducing minority-class performance and potentially leading to unreliable outcomes in critical applications, such as illness detection, prompting researchers to address this issue through hybrid techniques [22]. The study used simulated annealing for undersampling and applied support vector machines, decision trees, k-nearest neighbors, and discriminant analysis for classification, validating the approach on 51 real-world datasets and demonstrating superior effectiveness in reducing misclassification compared to previous methods.

Conventional SVM models are widely used across many domains. However, research indicates that they lack a coherent geometric definition, which might potentially undermine their theoretical performance, particularly in high-dimensional scenarios [23]. Qi et al. [23] explored the use of a combined penalty and introduced an elastic net support vector machine (ENSVM), which penalizes slack variables rather than the hyperplane's normal vectors, showing that ENSVM outperforms standard SVM and Doubly Regularized SVM (DrSVM) in terms of logical specification, stability, and high-dimensional characteristics, while the integration of fused weights leads to the adaptive weighted ENSVM (AWENSVM), which

enhances adaptability and robustness in handling imbalanced data and shows superior performance compared to other common SVM models. Most classification techniques assume an even sample distribution across classes; however, this leads to biased performance toward the dominant class. The proposed Enhanced Automatic Twin Support Vector Machine (EATWSVM) method addresses this issue by integrating a kernel representation into the optimization of twin support vector machines (TWSVM) using a Gaussian similarity based on Mahalanobis distance, enhancing data distinguishability with a centered kernel alignment strategy, and determining regularization parameters based on the imbalance ratio and dataset overlap, with experimental results showing superior performance and training efficiency compared to state-of-the-art methods [13].

Imbalanced data can produce unsatisfactory classification models because they often lead to frequent misclassification of minority cases and hinder optimal performance. Shamsudin et al. [24] presented an enhanced approach called SVM-GA for handling imbalanced data. This method optimizes the SVM algorithm using a genetic algorithm (GA) in combination with a synthetic minority oversampling strategy. The experimental results show that the proposed method improves performance by 97% compared to the baseline and optimized models, outperforming SVM with Grid and Randomized search, especially for datasets with rare instances; however, the study is limited by small sample sizes (under 5000) and fewer features, suggesting that the model's performance may need further testing with more complex datasets to assess its efficacy in more intricate settings. Imbalanced data categorization is crucial in machine learning because it focuses on detecting abnormalities, which are often of interest but occur less frequently than regular instances in real-world systems. Developing classifiers on imbalanced data can be challenging because there is no definitive criterion for determining how imbalanced a dataset is [25]. To address this problem, this article proposes enhancing the current Support Vector Machine technique. A key aspect of classification involves estimating the likelihood of a new point x belonging to a specific class, where most classifiers generate a score linked to posterior probability through an implicit relationship, with models like SVM providing an approximate posterior probability that represents the likelihood of a class being assigned to a dataset, which can be refined for greater accuracy. A major challenge in data mining is handling imbalanced data in classification tasks, where some classes (majority classes) have a large amount of data, and others (minority classes) have only a few instances, leading to biased performance in traditional classifiers that focus on data distribution rather than error rates, often neglecting the minority classes in the classification outcome, a problem common in many real-world applications [26].

3. Materials and Methods

The research methodology consists of five distinct phases: data sources, data preparation, experiment, modeling, and model evaluation.

3.1 Data Sources

The research issue requires data to provide a response. The research resources used in this study consist of publicly available datasets obtained from the Knowledge Extraction based on Evolutionary Learning (KEEL) Database, the UC Irvine (UCI) Machine Learning Repository, and Kaggle. Table 1 shows the details of the datasets used in this study. The churn dataset has the highest imbalance ratio (IR = 5.36), and the heart dataset has the lowest (IR = 2.11). The heart dataset has the fewest instances (299), whereas the churn dataset has the most (3150). Churn has the most features (13), whereas yeast has the fewest (8).

Table 1. Detail dataset

Dataset	Features	Instances	Positive Instances (%)	Negative Instances (%)	IR	Public dataset
Yeast	8	1484	28,9	71,1	2,46	Kaggle
Heart	12	299	68	32	2,11	Keel
Churn	13	3150	16	84	5,36	UCI

3.2 Data Preparation

Data preprocessing is a crucial undertaking in machine learning that can greatly improve a model's results [27]-[29]. Data preprocessing is a pivotal stage in the machine learning life cycle. Obtaining a comprehensive and uncontaminated dataset is a major obstacle in the healthcare industry. Data quality is critical because it can significantly affect the model's learning capacity and final generalizability [30]. Efficient and accurate algorithms can be achieved by employing effective data preparation methods and procedures. This provides a strong basis for data-driven decision-making and application development [31]. Data preprocessing techniques, including data encoding, data imputation, transformation of skewed data, data balancing, feature scaling, and feature selection, were used [32]-[34]. Data preprocessing encompasses methods aimed at improving the integrity of the original data, including eliminating outliers and filling in missing values [35]. Preprocessing is an essential stage in data analysis, as it transforms raw data into a format that computers and machine learning algorithms can understand. This crucial stage significantly influences the accuracy and effectiveness of machine learning models [36]. The data preparation process includes key stages such as missing-value detection and data transformation.

3.3 Experiment

This research includes a series of experiments to evaluate the classification performance under various conditions: (1) assessing the performance of SVM with an imbalanced dataset to examine its bias towards the majority class, (2) evaluating SVM with a balanced dataset to observe improvements after addressing class imbalance, and (3) testing the PC-SVM model with an imbalanced dataset to determine its effectiveness in mitigating class imbalance while maintaining classification accuracy.

3.4 Modelling

Model development refers to the process of creating and refining models in the context of data analysis and machine learning. A model is a mathematical or statistical representation of a system or process that aims to uncover patterns or relationships within data.

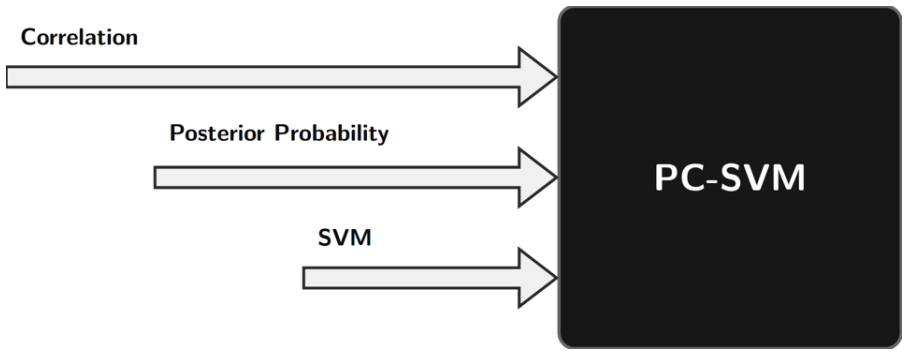


Figure 1. Proposed algorithm

The proposed idea shown in Figure 1 combines the algorithm referred to as PC-SVM, which stands for posterior probability and correlation-support vector machine. The fundamental principle behind the PC-SVM method combines posterior probability and correlation techniques, which is effective for improving SVM performance on imbalanced datasets. In an imbalanced data set, when one class is larger than the other, posterior probability provides the advantage of calculating class probabilities based on feature likelihood, thereby providing insight into the probability of a sample belonging to the minority or majority class. The probability distribution is optimized by multiplying the prior probability with the total sum of the multiplication of the R-squared of feature i of class Y with the independent probability of all feature vectors X. The attribute weights in the proposed method are obtained from the coefficient value correlation between the attribute and the class. The correlation coefficient ranges from -1 to 1; therefore, the attribute weight may be negative. To prevent negative values, attribute weighting uses the R-squared value. Attribute weighting is a method where the R-square value of each attribute for the class is multiplied by the probability of each attribute in calculating the conditional probability of the Naive Bayes classifier using the join probability method. Figure 2 illustrates the general architecture of the proposed algorithm.

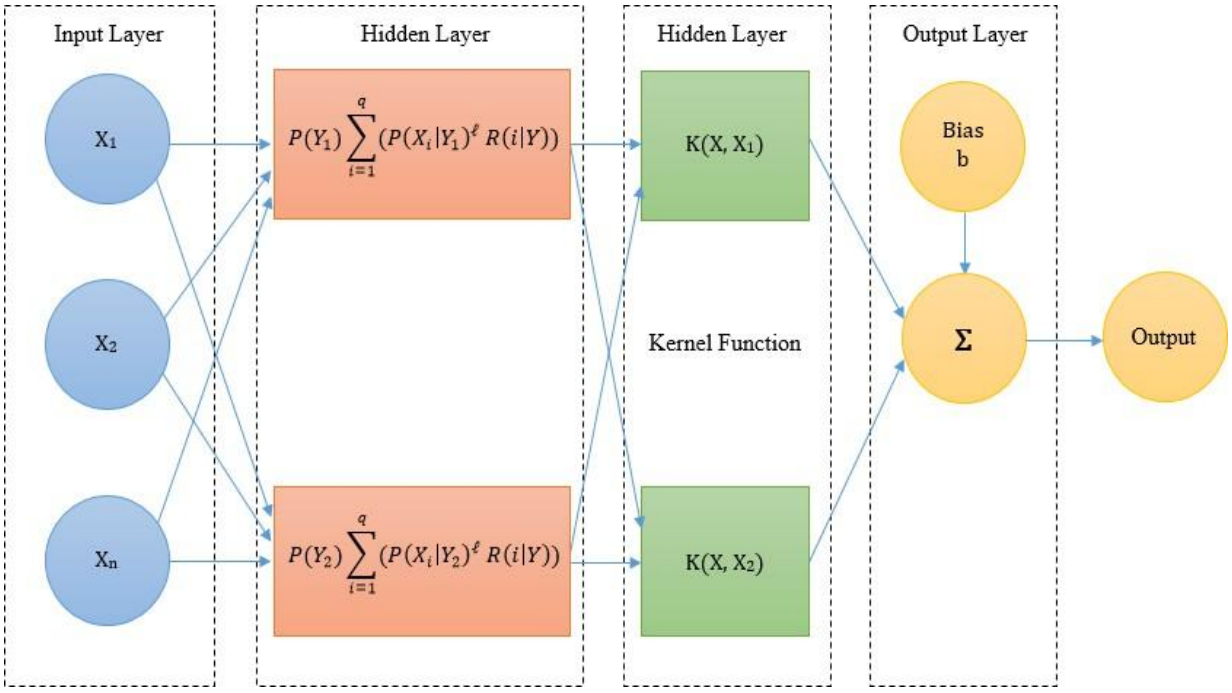


Figure 2. General architecture of the proposed algorithm

This attribute weighting can increase accuracy by considering how strongly an attribute relates to the specified class. To maintain consistency so that the R-square value always influences the classification process, the Laplacian method is used, which prevents a probability of 0. The Laplacian method is needed because if a probability is 0, the attribute's R-squared value has no effect. The Laplacian method forces the frequency of data occurrences in the dataset to be greater than 0 by adding 1 to each conditional probability calculation. The basic concept of attribute weighting is that each attribute has a different influence and priority for the class. To optimize the classification results, the research also used the Laplacian method to overcome the zero-probability problem. These posterior probabilities were used as input features for the SVM, which excels at finding the optimal separating hyperplane between classes. By using posterior probabilities to convert the original feature space into probability estimates, the SVM can focus on maximizing the margin between classes using these probabilities. This helps mitigate imbalance by making decision boundaries more sensitive to minority-class examples, increasing classification accuracy, and reducing bias toward the majority class. The PC-SVM algorithm uses an attribute-weighting technique based on R-squared. Figure 3 shows the R-squared weighting in the PC-SVM technique.

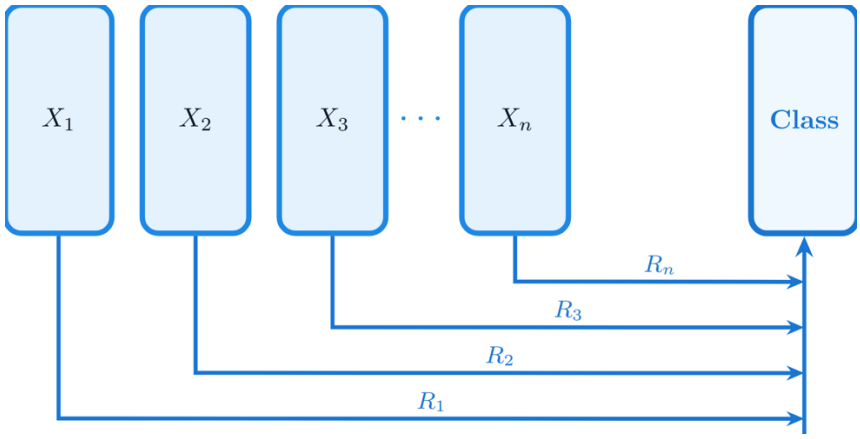


Figure 3. R-squared weighting

The SVM method utilizes joint probability to calculate conditional probability. Characteristic weighting assigns a numerical value to each characteristic to indicate its relative importance. Thus, an accurate approach to conditional probability uses the method of total addition rather than complete multiplication. Posterior probability is used to assess confidence in a classification. Posterior probability alone provides probabilities without considering the strength of the relationships between features. By incorporating R-squared, the model becomes richer in information, as each feature is measured by its likelihood distribution and its relevance to the target class. This creates a deeper probability structure, which is important in imbalanced datasets, where most features may be more relevant to the majority class and less helpful for detecting the minority class. By combining R-square with the independent probabilities of features, we effectively make the model more sensitive to variations and patterns present in features closely related to the minority class. In an imbalanced dataset, this approach helps the model capture subtle patterns important for detecting the minority class, which is often lost amid the dominance of the majority class.

3.5 Model Evaluation

Model evaluation in this study involved the confusion matrix, ROC curve, and scatter plot. The confusion matrix, applicable to both binary and multiclass problems, presents actual versus predicted classifications and supports performance metrics such as accuracy, precision, recall, and F1 score, based on TTP, TFN, TTN, and TFP [37]-[38]. The ROC curve assesses classifier performance across thresholds, with AUC indicating the model's ability to distinguish between classes, particularly useful in imbalanced data scenarios. Additionally, scatter plots visualize relationships and correlations between two variables, aiding in the interpretation of data patterns.

4. Result and Discussion

4.1 Correlation

The confusion matrix and correlations are widely used statistical tools that form the foundation of several applications, including exploratory data analysis, structural modeling, and data engineering [39]. Eq. (1) is used to determine the correlation value, and Eq. (2) is used to obtain the R-squared.

$$r = \frac{\sum (X_i - \bar{X})(Y - \bar{Y})}{\sqrt{\sum (X_i - \bar{X})^2 \sum (Y - \bar{Y})^2}} \tag{1}$$

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

Let X be a vector with an unknown class label, consisting of features indexed by i = {1, …, q}, where R(i|Y) denotes the attribute weights computed based on the coefficient of determination (R-squared) for each feature with respect to class Y.

The R is determined through the application of Eq. (2). The coefficient of determination R-squared for all features within vector X in relation to the dependent variable Y is calculated using Eq. (3):

$$R(i \mid Y) = R \tag{3}$$

with

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

When posterior probability is used in classification tasks, it generally only evaluates the probability of class membership based on the observed data. However, this approach can be limiting because it assumes that all features contribute equally and independently to the classification task, ignoring the strength of the relationships between individual features and the target class. Using R Square in the equation changes this dynamic by measuring how much variance in the target class each feature explains. By incorporating R Square into the model, we gained additional information about how relevant each feature is to the target class, going beyond simple probability estimations. For example, if a particular feature has a high R Square value, it is strongly correlated with the target class and should be given more weight in the classification decision.

4.2 Posterior Probability

Probability theory is a scientific discipline that uses statistical methods to understand random occurrences [40]. Possibility theory is based on two basic principles. These are prior probability and posterior probability. Posterior probability is the likelihood of an event occurring, computed after accounting for all available information or data [40]. To calculate the posterior probability for the PC-SVM method, it is necessary to choose the highest value among the numerous prior probabilities using conditional probability. Eq. (4) was used to calculate the posterior probability.

$$Posterior\ Probability = \max\left(P(Y) \sum_{i=1}^{q} P(X_i|Y) \right) \tag{4}$$

Posterior probability and correlation analyses are powerful techniques that can complement each other in predictive modeling. This probabilistic output allows for more nuanced decisions in classification tasks. In contrast, correlation analysis evaluates the strength and direction of the relationships between variables. Combining these methods can enhance model performance by understanding feature dependencies and selecting the most relevant variables. Eq. (5) is the posterior probability with correlations.

$$Posterior\ Probability\ with\ \text{Correlations} = \max\left(P(Y) \sum_{i=1}^{q} \left(P(X_i|Y)\ R(i|Y) \right) \right) \tag{5}$$

The Laplacian method estimates the conditional probabilities by addressing zero-frequency occurrences. This is achieved by adding one to the frequency count of each occurrence of X_i in the dataset. As outlined in Eq. (5), applying the Laplacian adjustment yields a revised posterior probability, as shown in Eq. (6).

$$Posterior\ Probability\ with\ Correlations\ and\ Laplace\ Smoothing = \max\left(P(Y) \sum_{i=1}^{q} \left(P(X_i|Y)^{\ell}\ R(i|Y) \right) \right) \tag{6}$$

Based on Eq. (6), by combining posterior probability with correlation and Laplacian, it can create a classification model that is more robust and can handle the shortcomings that arise due to the assumption of independence and lack of data in the minority class. This combination provides several advantages:

i) Overcoming zero probability in data that do not appear in training, especially in minority classes.
ii) Assigns more weight to features correlated with the target class, which is especially important for imbalanced datasets.
iii) The posterior probability is more representative in terms of feature probability and its relevance to the target, thereby increasing the model's ability to detect minority classes.

This approach is effective when class imbalance makes classification challenging because it increases the model's sensitivity to underrepresented classes.

4.3 Posterior Probability and Correlation-Support Vector Machine

In the combination of posterior probability and SVM, it uses the product of the prior probability and the total 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 learn to use these features to separate the classes by the largest margin.

4.3.1 Posterior probability and correlation-support vector machine

Naive Bayes calculates the posterior probability of each class Y_1 and Y_2 based on the attribute $X = (x_1, x_2, \ldots, x_n)$ of the sample in Eqs. (7-8);

$$P(Y_1 \mid X) = P(Y_1) \sum_{i=1}^{q} \left(P(X_i|Y_1)^{\ell} \, R(i|Y) \right) \tag{7}$$

$$P(Y_2 \mid X) = P(Y_2) \sum_{i=1}^{q} \left(P(X_i|Y_2)^{\ell} \, R(i|Y) \right) \tag{8}$$

where $P(Y_1)$ and $P(Y_2)$ are the prior probabilities of classes Y_1 and Y_2, and $P(X_1|Y_1)$, $P(X_2|Y_2)$ are the conditional probabilities of attribute X in each class. After getting the probabilities $(Y_1|X)$ and $P(Y_2|X)$, these probabilities are used as new input attributes for the SVM model.

4.3.2 SVM Formulation with Naive Bayes Features

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 Eq. (9):

$$f(z) = w^{\intercal} z + b \tag{9}$$

subject to Eq. (10):,

$$\hat{y} = sign(f(z)) = sign(w_1 . z + b) \tag{10}$$

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).

4.3.3 Combination formula of SVM and Naive Bayes

By combining the Naive Bayes posterior probabilities into the SVM, the combination formula becomes Eq. (11):

$$f(z) = w_1 . P(Y_1|X) + w_2 . P(Y_2|X) + b \tag{11}$$

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

4.3.4 Margin optimization in SVM

SVM maximizes the margin between classes Y_1 and Y_2 by solving the following optimization problem Eq. (12):

$$\min_{w, b} \frac{1}{2} \|w\|^2 \tag{12}$$

with condition Eq. (13):

$$Y_i(w^{\intercal} z_i + b) \geq 1 \; \forall_i \tag{13}$$

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.

4.3.5 Final Decision

The final prediction is determined by the decision function $f(z)$ in Eq. (14), where Naive Bayes and SVM work synergistically:

$$Posterior\ Probability = \max \left(P(Y) \sum_{i=1}^{q} \left(P(X_i|Y) \, R(i|Y) \right) \right) \tag{14}$$

In mathematical terms, $Posterior\ Probability\ (Y|X)$ is computed using Eq. (15):

$$Posterior\ Probability\ (Y|X) = \max \left(\frac{P(Y) \sum_{i=1}^{q} \left(P(X_i|Y)^{\ell} \; R(i|Y) \right)}{P(X)} \right) \tag{15}$$

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

After training, for a new sample X, the Naive Bayes component computes the posterior probabilities $P(Y_1|X)$ and $P(Y_2|X)$, which are then fed into the SVM. The SVM makes the final prediction using the decision function in Eqs. (16-17):

$$\hat{y} = sign\big(f(z)\big) = sign(w_1.P(Y_1)\sum_{i=1}^{q}(P(X_i|\,Y_1)^{\ell}\,R(i|Y)) + \ w_2.P(Y_2)\sum_{i=1}^{q}(P(X_i|\,Y_2)^{\ell}\,R(i|Y)) + b) \tag{16}$$

The final SVM decision function is given by Eq. (17).

$$\hat{y} = sign(f(z)) = sign\ (w1.P(Y1|X) + \ w2.P(Y2|X) + b) \tag{17}$$

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.

4.4 Data Preprocessing

4.4.1 Missing value detection

The analysis of missing values in the yeast dataset indicates a complete dataset, as evidenced by the absence of missing values across all attributes. Each attribute, including Mcg, Gvh, Alm, Mit, Erl, Pox, Vac, Nuc, and Class, contains 1484 valid entries, with no entries recorded as missing. This results in a 0% missing value percentage for each attribute, demonstrating that the dataset is fully populated. The completeness of the yeast dataset is a significant advantage, ensuring that the data are ready for further analysis and modeling without imputation or other corrective measures. This reliable data quality enhances confidence in the analytical outcomes and predictive performance of any models developed using this dataset. The examination of missing values in the heart dataset indicates no missing entries for any attribute, which is a noteworthy discovery. Each attribute has 299 valid entries. This results in a 0% incidence of missing values across the dataset. The absence of missing values across all attributes indicates that the dataset is complete and ready for analysis without imputation or correction. High-quality data are essential to ensure robust and reliable analyses and predictive modeling, improving the likelihood of precise insights into the factors affecting heart-related outcomes.

The examination of missing values in the churn dataset reveals no missing data entries for any attribute, which is a commendable result. Each attribute contains 3150 valid entries. This results in a 0% missing-value rate across all attributes. The complete dataset indicates high data integrity, as the absence of missing values eliminates the need for imputation or corrections. This completeness is crucial for thorough analysis and predictive modeling, ensuring that insights are based on a robust foundation and improving the reliability of conclusions about customer behavior and churn factors.

4.4.2 Data transformation

Transforming the yeast dataset was crucial for preparing it for posterior probability calculations and correlation analysis, as it involved converting various attributes from float64 to the category type. Initially, the attributes were represented as continuous numerical values, which could misrepresent their categorical nature in subsequent analyses. By transforming these attributes into categorical data types, the analysis can interpret and handle the relationships between classes without assuming a linear relationship, which is common with continuous variables. This conversion is particularly important for calculating posterior probabilities, where models must treat these attributes as distinct categories rather than continuous scales. Additionally, transforming the data into categories enhances the correlation analysis by ensuring that the statistical methods applied are appropriate for categorical data, allowing the identification of meaningful relationships between different yeast characteristics. Overall, this data transformation optimizes the dataset for effective statistical modeling and ensures accurate results for both posterior probability and correlation computations.

Transforming the heart dataset was essential to prepare it for posterior probability calculations and correlation analysis, involving converting various attributes from their original numerical types to Category. Initially, attributes like Age, Platelets, and serum creatinine were represented as float64, while others, such as Anaemia, creatinine phosphokinase, and DEATH_EVENT, were int64. Converting these variables to categorical data types better reflects their inherent characteristics and allows appropriate statistical analyses. This is particularly important for posterior probability estimation, as treating variables like Age and ejection fraction as categories prevents misinterpretation of their values as continuous data, which could lead to erroneous conclusions. Additionally, transforming all relevant attributes to Category improves the validity of correlation analyses by allowing examination of relationships between discrete classes rather than assuming a linear relationship between continuous variables. Ultimately, this transformation aligns the dataset with the requirements of the analytical methods to be applied, facilitating more accurate insights into the relationships among various heart-related factors and their impact on the outcome of interest.

Transforming the churn dataset was vital for posterior probability calculations and correlation analysis, converting various attributes from their original integer (Int64) and floating-point (float64) types into category types. Initially, attributes such as call failure, complaints, and subscription length were represented as integers, which could imply a numerical relationship that does not reflect their categorical nature. By transforming these attributes into categorical types, the dataset was aligned more closely with the intended analysis, allowing each attribute to be treated as a distinct category rather than a continuous variable. This is particularly important for posterior probability estimation, as categorical data can

provide clearer insights into the likelihood of churn based on various factors, such as Age Group or Tariff Plan, without the misleading implications of numerical scales. Furthermore, the transformation improves the correlation analysis by enabling the exploration of the relationships between categorical variables, allowing a more meaningful interpretation of how different attributes interact and influence customer churn. Overall, this data transformation optimized the dataset for effective statistical modeling, ensuring accurate results in posterior probability and correlation computations related to customer behavior.

4.5 Performance SVM with Imbalanced Dataset

The results of performing SVM with an imbalanced dataset are presented in Figures 4-6. The preprocessing technique used to mitigate class imbalance is SMOTE. The performance measurements utilized comprise accuracy, precision, recall, and the F1-score. Figure 4 shows that the SVM model performs moderately well on the balanced yeast dataset, with an overall accuracy of 67.58%, meaning that it correctly predicts more than two-thirds of the time. However, this accuracy does not fully reflect performance across both classes, which have equal numbers of instances (1055 each for classes -1 and 1). The confusion matrix shows that for the majority class (-1), the model has high recall (82%) but lower precision (0.64), indicating that it misclassifies some class 1 instances as class -1. This results in an F1-score of 0.72. For the minority class (1), the model's precision is higher at 0.74, but recall drops to 54%, showing that it misses many class 1 instances and resulting in an F1-score of 0.62. The scatter plot shows that class -1 instances are classified more accurately, while class 1 instances are more spread out and misclassified. The macro-average scores (precision 0.69, recall 0.68, F1-score 0.67) suggest that the model is fairly balanced across classes, with slightly better performance for the majority class. The weighted average F1-score of 0.67 confirms that while the model is consistent, it still shows some bias toward the majority class despite the balanced dataset.

```
Accuracy: 0.6758293838862559
Classification Report:
              precision    recall  f1-score   support

          -1       0.64      0.82      0.72      1055
           1       0.74      0.54      0.62      1055

    accuracy                           0.68      2110
   macro avg       0.69      0.68      0.67      2110
weighted avg       0.69      0.68      0.67      2110
```

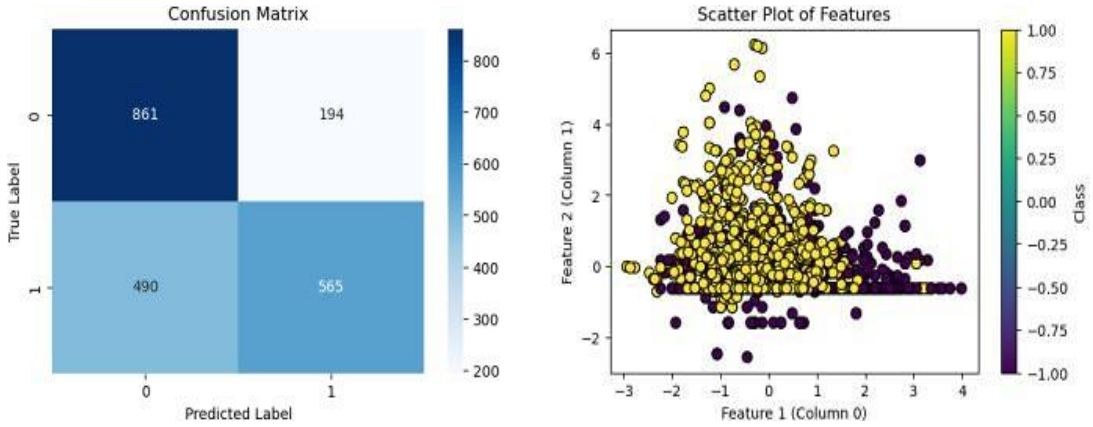


Figure 4. Performance of SVM on balanced datasets utilizing the yeast dataset

Figure 5 shows the classification results on the balanced heart dataset using SVM, where SMOTE helped address class imbalance and improve the model's ability to distinguish between the two classes. While the model's accuracy is 68.97%, indicating moderate effectiveness, there is still room for improvement. The confusion matrix shows that for class -1, the model has a precision of 0.72 but a lower recall of 0.62, meaning that it misses some true class -1 instances. For class 1, the recall is higher at 0.76, meaning that it correctly identifies more class 1 instances, but the precision is lower at 0.67, indicating a higher rate of false positives. The scatter plot shows that class 1 instances are more spread out, leading to overlap with class -1 and making it harder for the model to separate the classes perfectly. The F1-scores are 0.67 for class -1 and 0.71 for class 1, showing slightly better performance for class 1, especially in recall, which matters in medical contexts. The macro averages for precision, recall, and F1-score were all 0.69, indicating a balanced performance between the classes. The weighted averages confirmed the model's consistent performance, making it reliable for this balanced dataset.

Model Evaluation Metrics

Accuracy: 0.6897 (68.97%)

Classification Report:

Class	Precision	Recall	F1-Score	Support
-1	0.72	0.62	0.67	203
1	0.67	0.76	0.71	203
accuracy			**0.69**	**406**
macro avg	**0.69**	**0.69**	**0.69**	**406**
weighted av	**0.69**	**0.69**	**0.69**	**406**

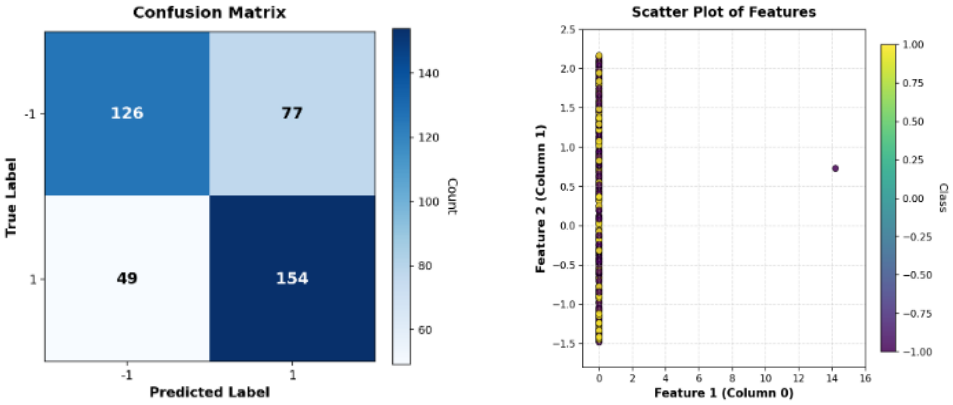


Figure 5. Performance of SVM on balanced datasets utilizing the heart dataset

Accuracy: 0.5410700828937453

Classification Report:

	precision	recall	f1-score	support
-1	0.79	0.11	0.20	2654
1	0.52	0.97	0.68	2654
accuracy			0.54	5308
macro av	0.66	0.54	0.44	5308
weighted	0.66	0.54	0.44	5308

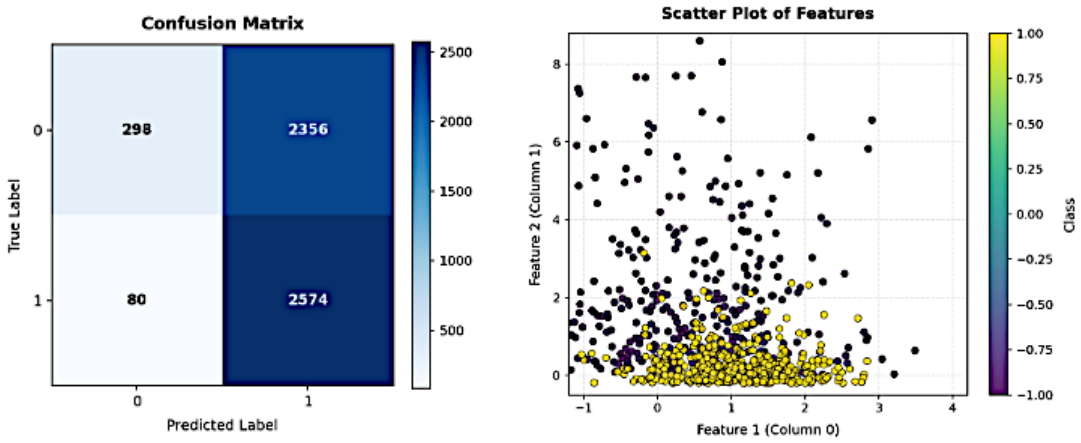


Figure 6. Performance of SVM on balanced datasets utilizing the churn dataset

Figure 6 shows the classification results on the balanced churn dataset using SVM, highlighting the mixed performance with notable differences between the two classes. The overall accuracy of 54.11% suggests that the model performs slightly better than random guessing. Despite the balanced dataset, the model struggles to classify churn and non-churn instances effectively. Looking at the confusion matrix, for class -1 (non-churn), the model has a low recall of 11%, meaning it fails to identify most non-churn instances. Although its precision is higher at 0.79, indicating that when it

predicts non-churn, it is often correct, the low recall results in a poor F1-score of 0.20, showing poor performance in identifying non-churn cases. For class 1 (churn), the recall is much higher at 97%, showing that the model is good at detecting churn cases. However, its precision is lower at 0.52, meaning many predicted churn instances are incorrect, which leads to a higher rate of False Positives. This reduces precision but improves the balance between precision and recall, reflected in the F1-score of 0.68. The scatter plot suggests a significant overlap between classes, particularly for class -1, which likely causes a low recall for non-churn instances. The decision boundary may be set to favor detecting churn but misclassifies non-churn instances. The macro average scores show limited ability to generalize, with a precision of 0.66 and recall of 0.54, leading to an F1-score of 0.44, indicating that the model is biased towards churn detection. The weighted averages also show an F1-score of 0.44, confirming that the model needs improvement to balance its predictions better between churn and non-churn cases.

4.6 Performance Evaluation of the PC-SVM Model on Imbalanced Datasets

The results of the PC-SVM performance on an imbalanced dataset are presented in Figures 7–9. The performance metrics assessed included accuracy, precision, recall, and F1-score. Figure 7 shows that the PC-SVM model performs exceptionally well on the imbalanced yeast dataset, achieving 100% accuracy. This means that the model correctly predicted all instances without any errors, demonstrating its effectiveness and reliability in classifying the yeast data. The confusion matrix confirms this strong performance, as all instances from both classes (-1 and 1) were correctly classified, with no misclassifications. This matrix only shows true positives and true negatives, meaning there are no false positives or false negatives. This indicates that the PC-SVM model can accurately distinguish between the two classes, even with a class imbalance (61 instances of class 1 vs. 1423 of class -1). A scatter plot of the results would likely show a clear separation between the two classes, with no overlap. Class -1 instances would form a dense cluster, while class 1 instances would be clearly distinguishable, reinforcing the model's ability to handle class imbalance and classify both classes accurately. The classification report shows perfect scores for both classes, with precision, recall, and F1-score all at 1.00. This means that the model is 100% accurate in its predictions, with no false positives or false negatives, and an optimal balance between precision and recall. The support values show 1423 instances for class -1 and 61 for class 1. Despite the imbalance, the PC-SVM method handles the disparity well, achieving perfect classification for both classes. The macro and weighted average scores are also 1.00, further highlighting the model's consistent performance across the dataset. This outstanding performance demonstrates PC-SVM's power in handling imbalanced datasets in classification tasks.

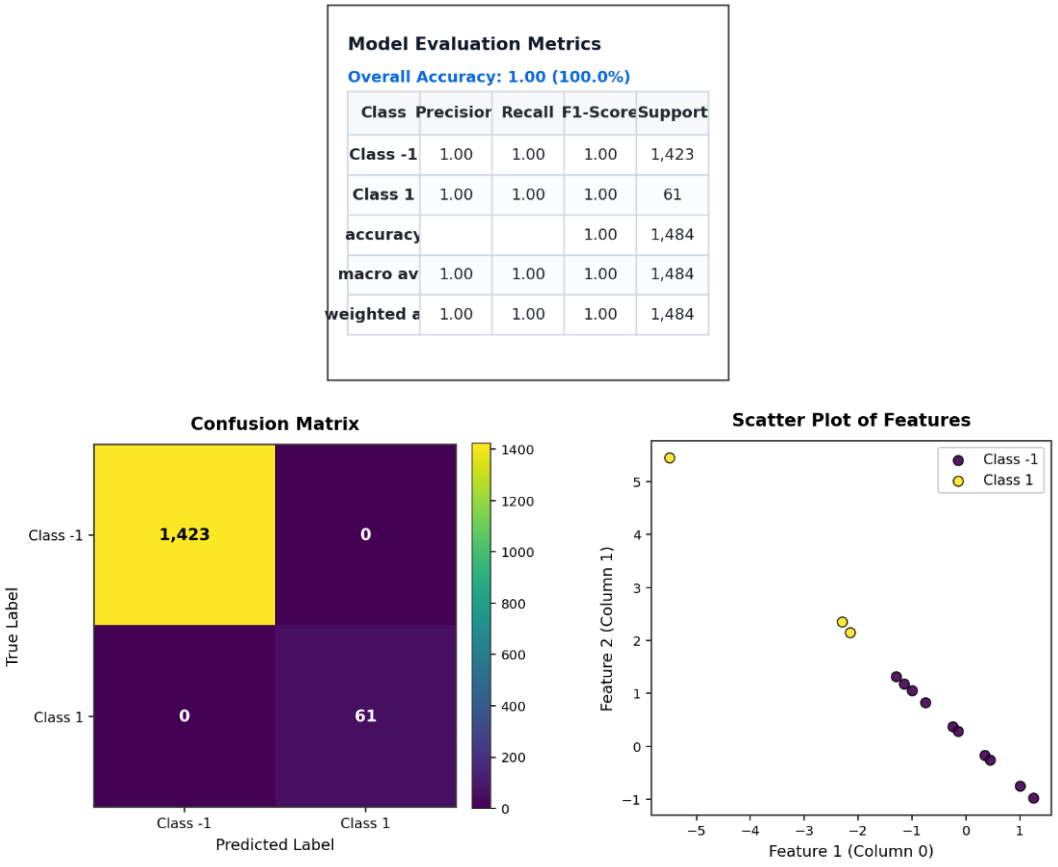


Figure 7. Performance of PC-SVM on imbalanced datasets utilizing the yeast dataset

Figure 8 shows that the PC-SVM model performs exceptionally well on the imbalanced heart dataset, achieving 100% accuracy. This means that the model correctly classified all instances without any errors, highlighting its ability to accurately distinguish between the two classes. The confusion matrix further confirms this, showing that all instances for

both classes (-1 and 1) were classified correctly, with no false positives or false negatives. This suggests that the PC-SVM model is highly effective, even with the class imbalance in the dataset. A scatter plot of the results would likely show clear separation between the two classes, with no overlap. It would clearly identify all instances of class -1 (no heart disease) and class 1 (with heart disease), reinforcing the model's perfect classification and its ability to handle the class imbalance effectively. The classification report shows perfect results for both classes, with precision, recall, and F1-score all at 1.00. This means that the model's predictions are flawless, with no false positives or false negatives, and a perfect balance between precision and recall. The support values show 27 instances of class -1 and 272 of class 1. Despite the imbalance, the PC-SVM model classifies all instances perfectly. Both the macro and weighted average metrics are also 1.00, highlighting the model's consistent performance across both classes. This performance demonstrates PC-SVM's power in classifying imbalanced datasets, particularly in important fields such as healthcare.

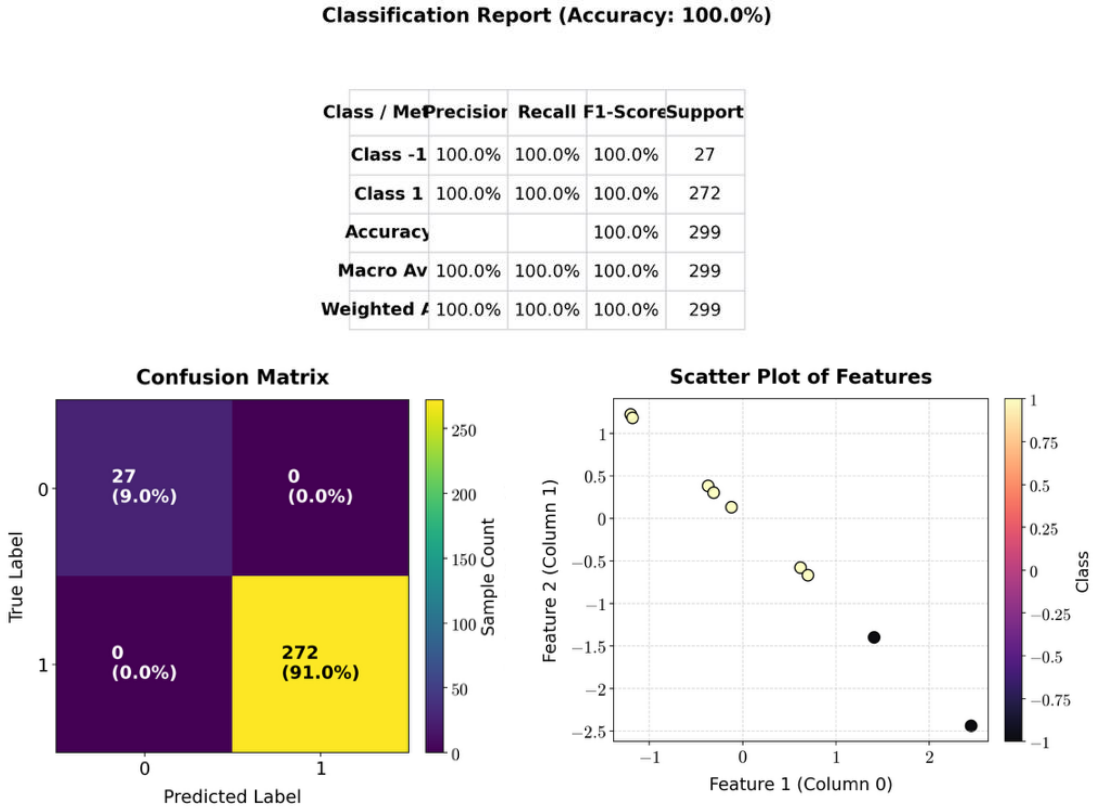


Figure 8. Performance of PC-SVM on imbalanced datasets utilizing the heart dataset

Figure 9 shows that the PC-SVM model performs exceptionally well on the imbalanced churn dataset, achieving 100% accuracy. This means that the model accurately classified every instance in the dataset, distinguishing between the two classes without any errors, even with the class imbalance. The confusion matrix confirms this, showing that all instances for both classes (-1 for non-churned customers and 1 for churned customers) were correctly classified. There were no false positives or false negatives, demonstrating the model's strong performance and ability to handle the class imbalance. A scatter plot would likely show clear separation between the two classes, with distinct groups for churned and non-churned customers. The lack of overlap would visually confirm that the model accurately identified all churned customers and did not misclassify any non-churned ones, further supporting the model's effectiveness. The classification report confirms perfect results, with precision, recall, and F1-scores all at 1.00 for both classes. This indicates that the model correctly predicted all true positives and avoided false positives and false negatives, maintaining a perfect balance between precision and recall. The support values show 1867 instances of class -1 and 1283 of class 1, indicating a slight imbalance. Despite this, the PC-SVM model classified all instances correctly, proving its ability to handle imbalanced datasets effectively. Both the macro and weighted averages are 1.00, showing consistent performance across both classes. This impressive performance highlights the PC-SVM's potential as a powerful tool for classifying imbalanced datasets, such as customer churn prediction.

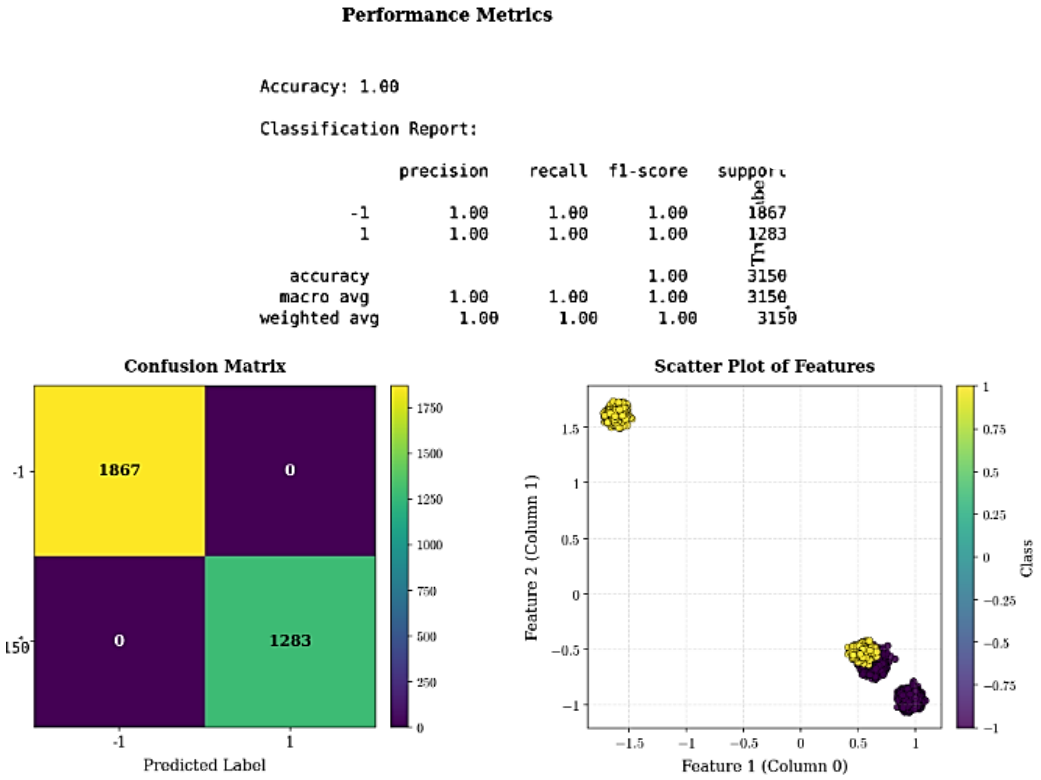


Figure 9. Performance of PC-SVM on imbalanced datasets utilizing the churn dataset

Table 2 provides a concise overview of the performance findings obtained from experiments conducted on SVM with an imbalanced dataset, SVM with a balanced dataset, and PC-SVM with an imbalanced dataset.

Table 2. Summary of experimental performance

No	Dataset	SVM Imbalanced Dataset				SVM Balanced Dataset (SMOTE)				PC-SVM Imbalanced Dataset			
		Accuracy	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score	Accuracy	Precision	Recall	F1 Score
1.	Yeast	0.74	0.72	0.74	0.68	0.68	0.69	0.68	0.67	1.00	1.00	1.00	1.00
2.	Heart	0.83	0.83	0.83	0.82	0.69	0.69	0.69	0.69	1.00	1.00	1.00	1.00
3.	Churn	0.84	0.71	0.84	0.77	0.54	0.69	0.54	0.44	1.00	1.00	1.00	1.00
	Mean	0.80	0.75	0.80	0.76	0.64	0.69	0.64	0.60	1.00	1.00	1.00	1.00

Table 2 compares three machine learning models: SVM (with and without balancing) and PC-SVM, on three datasets (Yeast, Heart, and Churn). The models were evaluated based on accuracy, precision, recall, and F1 score to determine how well they handle imbalanced data. The results show that PC-SVM performs the best. The performance of each model is discussed as follows. SVM on imbalanced data performs reasonably well, with an accuracy between 0.74 and 0.84 across the datasets. On the Yeast dataset, SVM achieves an accuracy of 0.74, with precision and recall of 0.72 and 0.74, respectively. Although these results are consistent, SVM struggles to balance precision and recall, with an F1 score of 0.68. On the Heart and Churn datasets, SVM performs better, with F1 scores of 0.82 and 0.77, respectively, but it still struggles with recall and misses some relevant instances. Overall, SVM handles imbalanced data decently, but its performance can improve, especially on more complex datasets, such as Churn. When the dataset is balanced using SMOTE, SVM's performance drops significantly, especially on the Churn dataset. Its accuracy falls to 0.54, and the F1 score drops to 0.44, with lower precision and recall. SMOTE, intended to improve learning from the minority class, seems to introduce noise and reduce model precision, particularly on the Churn dataset. This suggests that, although balancing techniques can help, they may not always improve performance and can even make things worse, especially for complex models like SVM.

PC-SVM was the best-performing model, achieving perfect scores (accuracy, precision, recall, and F1 score) across all datasets. It correctly classified all instances with no false positives or false negatives. This performance shows that PC-SVM is particularly good at handling imbalanced datasets, likely because of its ability to better differentiate between classes, especially the minority class, something that SVM struggles with. In summary, applying SMOTE to balance the dataset did not improve SVM's performance and even worsened the results on the Churn dataset. Conversely, PC-SVM

performed strongly across all datasets, establishing itself as a reliable classifier for imbalanced data. These findings indicate that for datasets such as Yeast, Heart, and Churn, PC-SVM is the most effective model, whereas traditional SVM requires further optimization. Moreover, balancing techniques such as SMOTE do not guarantee performance enhancement and can, in some cases, negatively impact the model.

5. Conclusions

This study aimed to improve the Support Vector Machine performance on imbalanced datasets by integrating posterior probability and correlation techniques, resulting in the development of the PC-SVM model. The research objectives were achieved by comprehensively evaluating the proposed model across various imbalanced datasets. The findings indicate that the PC-SVM model yields notable improvements in classification metrics, including accuracy, precision, recall, and F1-score. In all evaluated cases, PC-SVM consistently outperformed conventional models by effectively reducing the bias toward majority classes. These results suggest that combining posterior probability and correlation techniques substantially strengthens the SVM's ability to handle imbalanced data distributions. In this study, we propose a novel classification approach named PC-SVM, designed to overcome the shortcomings of conventional SVMs when dealing with imbalanced datasets. The main contributions of this study are summarized as follows:

i) Fusing posterior probability and correlation techniques into the SVM framework yields a novel algorithm that effectively addresses dataset imbalance, making it a practical solution for real-world problems.
ii) The PC-SVM model consistently outperformed the traditional SVM models on imbalanced datasets.
iii) The proposed method was rigorously tested on several publicly available datasets to validate its effectiveness and generalizability across different domains.
iv) Integrating correlation coefficients into posterior probabilities for classification improves the understanding of the relationships between features and their impact on class separation, a novel adaptation within the SVM framework.
v) PC-SVM can be recommended for industries where detecting rare events is crucial. In healthcare, it can help identify rare diseases more accurately, thereby reducing misdiagnoses. In cybersecurity, it can enhance intrusion detection and identify rare but serious threats. Additionally, in manufacturing and aviation, PC-SVM can predict equipment failures, thereby preventing costly breakdowns. These applications show the practical utility of PC-SVM in real-world scenarios.

Although the research yielded promising results, several limitations were identified that opened up opportunities for further investigation:

i) Although multiple datasets were used, the research was limited to publicly available data with specific imbalanced characteristics.
ii) The fusion of posterior probability and correlation techniques increased the computational complexity of the PC-SVM model. Future work should focus on optimizing the algorithm for faster computation, particularly for large-scale datasets.
iii) The PC-SVM performance relied on hyperparameter tuning, which was performed manually in this study. Future research could incorporate automated hyperparameter optimization techniques to further enhance model performance.
iv) While this study focused on posterior probability and correlation techniques, future studies could explore integrating other statistical methods or ensemble learning strategies to further improve classification performance on imbalanced datasets.

In conclusion, this study presents an effective solution for improving SVM performance on imbalanced datasets through PC-SVM, setting a foundation for future research in this area.

Acknowledgements

The authors thank the Malaysian Institute of Information Technology, Universiti Kuala Lumpur for providing the infrastructure needed to carry out this research.

Funding

This study was not supported by any grants from funding bodies in the public, private, or not-for-profit sectors.

Authors Contribution

Canggih Ajika Pamungkas: Conceptualisation; Methodology; Investigation; Writing - original draft
Megat F. Zuhairi: Validation; Writing- review & revise

Conflict of Interest

The authors declare no conflict of interest.

References

[1] J. Alcaraz, M. Labbé, and M. Landete, "Support vector machine with feature selection: A multiobjective approach," *Expert Systems with Applications*, vol. 204, Art. no. 117485, 2022, https://doi.org/10.1016/j.eswa.2022.117485

[2] J. Liu, "Fuzzy support vector machine for imbalanced data with borderline noise," *Fuzzy Sets and Systems*, vol. 413, pp. 64–73, 2021, https://doi.org/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," *Discrete Dynamics in Nature and Society*, vol. 2021, Art. no. 6647557, 2021, https://doi.org/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 Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 1583–1593, 2022, https://doi.org/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," *Applied Sciences*, vol. 11, no. 2, Art. no. 869, 2021, https://doi.org/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 Computer Science*, vol. 1, no. 4, Art. no. 205, 2020, https://doi.org/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, https://doi.org/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, pp. 112–123, 2020, https://doi.org/10.1016/j.neucom.2020.03.085
[9] S. S. Mullick, S. Datta, S. S. Dhekane, and S. Das, "Appropriateness of performance indices for imbalanced data classification: An analysis," *Pattern Recognition*, vol. 102, Art. no. 107197, 2020, https://doi.org/10.1016/j.patcog.2020.107197
[10] H. Shamsudin, U. K. Yusof, A. Jayalakshmi, and M. N. A. Khalid, "Combining oversampling and undersampling techniques for imbalanced classification: A comparative study using credit card fraudulent transaction dataset," in *Proceedings of the 2020 16th IEEE International Conference on Control & Automation (ICCA)*, Singapore, 2020, pp. 803–808, https://doi.org/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, https://doi.org/10.1016/j.neunet.2020.06.026
[12] X. Tao, Q. Li, C. Ren, *et al.*, "Affinity and class probability-based fuzzy support vector machine for imbalanced data sets," *Neural Networks*, vol. 122, pp. 289–307, 2020, https://doi.org/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 Recognition*, vol. 107, Art. no. 107442, 2020, https://doi.org/10.1016/j.patcog.2020.107442
[14] R. Abo Zidan and G. Karraz, "Gaussian pyramid for nonlinear support vector machine," *Applied Computational Intelligence and Soft Computing*, vol. 2022, Art. no. 5255346, 2022, https://doi.org/10.1155/2022/5255346
[15] Y. S. Solanki, P. Chakrabarti, M. Jasinski, *et al.*, "A hybrid supervised machine learning classifier system for breast cancer prognosis using feature selection and data imbalance handling approaches," *Electronics*, vol. 10, no. 6, Art. no. 699, 2021, https://doi.org/10.3390/electronics10060699
[16] R. Ramesh Kumar, K. Karthik, P. V. Elumalai, *et al.*, "Investigation of nano composite heat exchanger annular pipeline flow using CFD analysis for crude oil and water characteristics," *Case Studies in Thermal Engineering*, vol. 49, Art. no. 103297, 2023, https://doi.org/10.1016/j.csite.2023.103297
[17] B. Richhariya and M. Tanveer, "A reduced universum twin support vector machine for class imbalance learning," *Pattern Recognition*, vol. 102, Art. no. 107150, 2020, https://doi.org/10.1016/j.patcog.2019.107150
[18] M. Li, A. Xiong, L. Wang, S. Deng, and J. Ye, "ACO resampling: Enhancing the performance of oversampling methods for class imbalance classification," *Knowledge-Based Systems*, vol. 196, Art. no. 105818, 2020, https://doi.org/10.1016/j.knosys.2020.105818
[19] A. S. Hussein, T. Li, C. W. Yohannese, and K. Bashir, "A-SMOTE: A new preprocessing approach for highly imbalanced datasets by improving SMOTE," *International Journal of Computational Intelligence Systems*, vol. 12, no. 2, pp. 1412–1422, 2019, https://doi.org/10.2991/ijcis.d.191114.002
[20] P. Gnip, L. Vokorokos, and P. Drotár, "Selective oversampling approach for strongly imbalanced data," *PeerJ Computer Science*, vol. 7, pp. 1–22, 2021, https://doi.org/10.7717/peerj-cs.604
[21] 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 Systems*, vol. 196, Art. no. 105845, 2020, https://doi.org/10.1016/j.knosys.2020.105845
[22] A. S. Desuky and S. Hussain, "An improved hybrid approach for handling class imbalance problem," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3853–3864, 2021, https://doi.org/10.1007/s13369-021-05347-7
[23] K. Qi, H. Yang, Q. Hu, and D. Yang, "A new adaptive weighted imbalanced data classifier via improved support vector machines with high-dimension nature," *Knowledge-Based Systems*, vol. 185, Art. no. 104933, 2019, https://doi.org/10.1016/j.knosys.2019.104933
[24] H. Shamsudin, U. K. Yusof, H. Yang, and I. S. Isa, "An optimized support vector machine with genetic algorithm for imbalanced data classification," *Jurnal Teknologi*, vol. 85, no. 4, pp. 67–74, 2023, https://doi.org/10.11113/jurnalteknologi.v85.19695

[25] Y.-S. Park and J.-S. Lee, "A learning objective controllable sphere-based method for balanced and imbalanced data classification," *IEEE Access*, vol. 9, pp. 158010–158026, 2021, https://doi.org/10.1109/ACCESS.2021.3130272
[26] H. Patel, D. S. Rajput, G. T. Reddy, *et al.*, "A review on classification of imbalanced data for wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 16, no. 4, 2020, https://doi.org/10.1177/1550147720916404
[27] S. Strasser and M. Klettke, "Transparent data preprocessing for machine learning," in *Proceedings of the Workshop on Human-In-the-Loop Data Analytics (HILDA '24)*, Santiago, Chile, 2024, https://doi.org/10.1145/3665939.3665960
[28] J. Nalic and A. Svraka, "Importance of data pre-processing in credit scoring models based on data mining approaches," in *Proceedings of the 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, Opatija, Croatia, 2018, pp. 1046–1051, https://doi.org/10.23919/MIPRO.2018.8400191
[29] H. F. Tayeb, M. Karabatak, and C. Varol, "Time series database preprocessing for data mining using Python," in *Proceedings of the 2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, Beirut, Lebanon, 2020, pp. 20–23, https://doi.org/10.1109/ISDFS49300.2020.9116260
[30] S. Albahra, T. Gorbett, S. Robertson, *et al.*, "Artificial intelligence and machine learning overview in pathology & laboratory medicine: A general review of data preprocessing and basic supervised concepts," *Seminars in Diagnostic Pathology*, vol. 40, no. 2, pp. 71–87, 2023, https://doi.org/10.1053/j.semdp.2023.02.002
[31] Z. Liu, "Research on data preprocessing method for artificial intelligence algorithm based on user online behavior," *Journal of Computing and Electronic Information Management*, vol. 12, no. 3, pp. 74–78, 2024, https://doi.org/10.54097/qf6fv8j1
[32] A. Q. Md, S. Kulkarni, C. J. Joshua, *et al.*, "Enhanced preprocessing approach using ensemble machine learning algorithms for detecting liver disease," *Biomedicines*, vol. 11, no. 2, Art. no. 581, 2023, https://doi.org/10.3390/biomedicines11020581
[33] K. Maharana, B. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, 2022, https://doi.org/10.1016/j.gltp.2022.04.020
[34] T. H. Duong and T. A. Nguyen-Thi, "A review: Preprocessing techniques and data augmentation for sentiment analysis," *Computational Social Networks*, vol. 8, no. 1, pp. 1–16, 2021, https://doi.org/10.1186/s40649-020-00080-x
[35] 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," *Frontiers in Energy Research*, vol. 9, pp. 1–17, 2021, https://doi.org/10.3389/fenrg.2021.652801
[36] V. Chernykh, A. Stepnov, and B. O. Lukyanova, "Data preprocessing for machine learning in seismology," *CEUR Workshop Proceedings*, vol. 2930, pp. 119–123, 2021. [Online]. Available: http://ceur-ws.org/Vol-2930/
[37] A. J. Mohammed, "Improving classification performance for a novel imbalanced medical dataset using SMOTE method," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 3, pp. 3161–3172, 2020, https://doi.org/10.30534/ijatcse/2020/104932020
[38] A. Kulkarni, D. Chong, and F. A. Batarseh, "Foundations of data imbalance and solutions for a data democracy," in *Data Democracy: At the Frontier of Medical Informatics, Healthcare, and Cybersecurity*, F. A. Batarseh and R. Niu, Eds. London, U.K.: Academic Press, 2020, pp. 83–106, https://doi.org/10.1016/B978-0-12-818366-3.00005-8
[39] D. Makowski, M. S. Ben-Shachar, I. Patil, and D. Lüdecke, "Methods and algorithms for correlation analysis in R," *Journal of Open Source Software*, vol. 5, no. 51, Art. no. 2306, 2020, https://doi.org/10.21105/joss.02306
[40] M. S. Vural and M. Telceken, "Modification of posterior probability variable with frequency factor according to Bayes theorem," *Journal of Intelligent Systems with Applications*, vol. 5, no. 1, pp. 19–26, 2022, https://doi.org/10.54856/jiswa.202205195