Comprehensive Evaluation of Bankruptcy Prediction in Taiwanese Firms Using Multiple Machine Learning Models

Put authors name here[1*], Put author name here[1], Put author name here[1], Put author name here[2], Put authors name here[2]

[1] Put author's affiliation here, complete with address, postal code, and country

Abstract. Bankruptcy prediction is a significant issue in finance because accurate predictions would enable stakeholders to act quickly to reduce their financial losses. This study developed an advanced bankruptcy prediction model using Support Vector Machines (SVM), Random Forest (RF), and Artificial Neural Network (ANN) algorithms based on datasets from the UCI machine learning repository. The core contribution of this research is the establishment of a hybrid model that effectively combines multiple machine learning (ML) algorithms with advanced data with the Synthetic minority oversampling technique Tomek (SMOTE Tomek) or SMOTE- Edited Nearest Neighbor (SMOTE-ENN) resampling data technique to improve bankruptcy prediction accuracy. Additionally, a wrapper-based feature selection (FS) utilizing Binary Particle Swarm Optimization (BPSO) was utilized to find an optimal feature subset and boost the model's predictive performance. After selecting the best features, these were used to train the three ML algorithms, and hyper-parameter optimization was implemented to boost model performance. From the results measured by evaluation metrics, the proposed model ANN with the combination of parameter tuning, feature selection algorithm, SMOTE-ENN, and optimal hyper-parameters demonstrates superior performance compared to traditional methods, achieving an F_1 Score of 98.5% and an accuracy of 98.6%. The results imply that the prediction performance of bankruptcy models may be considerably improved by combining many analytical methodologies. This approach not only improves the accuracy but also the reliability of financial risk assessments, providing valuable insights for investors, financial analysts, and policymakers. The success of the model opens avenues for further research into hybrid predictive models in various sectors of finance, potentially transforming risk assessment methodologies.

Keywords: Artificial neural networks; Bankruptcy prediction; Binary particle swarm; Optimization random forest; Synthetic minority oversampling technique (SMOTE).

1. Introduction

Bankruptcy prediction has emerged as a critical area of research due to its profound implications for stakeholders, including investors, creditors, and policymakers.

*Corresponding author's email

The evolution of bankruptcy prediction research spans several decades, with early models like Altman's Z-score (Prasetiyani, 2020) and Ohlson's logistic regression (Najib, 2020) using financial ratios to predict bankruptcy risk effectively. Over the years, the field has transitioned from these simple ratio-based models to more sophisticated multivariate methods such as logistic regression and discriminant analysis, which have been widely adopted for their enhanced predictive power (Huo, 2024). However, the emergence of data-rich environments and complex corporate structures has spurred interest in leveraging advanced ML models. These techniques, including neural networks, Decision Trees (DT), and Support Vector Machine (SVM) (Tobback et al., 2017), have demonstrated superior performance in capturing nonlinear relationships and complex interaction effects among predictors, which are often missed by traditional models.

ML is widely applied in commercial applications and research projects in many fields, e.g. disease detection (Tran et al., 2022; Dao et al., 2022; Le et al., 2020a), smart city (Minh, Mai and Minh, 2021), and forecasting (Ho et al., 2022, Samsudin et al., 2024). The application of ML techniques has grown in popularity recently and is progressively enhancing productivity by utilizing modern algorithms in the economy, industry, and agriculture (Tran et al., 2023a; Tran et al., 2023b; Lomakin et al., 2022; Ismat et al., 2022), etc. ML and predictive analytics employ different approaches to problem-solving, yet they share a connection. Predictive analytics can seamlessly integrate with other applications, while ML offers greater adaptability, flexibility, modernity, and more degrees of freedom in addressing problems. Predictive analytics, with its longer history, follows a more procedural approach. It utilizes models that make predictions based on historical data patterns and serves specific purposes such as predicting pricing trends, drug dosage, risk analysis, propensity modeling, diagnosis, and document categorization (Kelleher, Mac Namee and D'arcy, 2020). ML is an interdisciplinary field that is linked to mathematical fields (Shalev-Shwartz and BenDavid, 2014), providing a robust framework for addressing complex classification problems like bankruptcy prediction. The incorporation of machine learning into bankruptcy prediction not only enhances the adaptability and accuracy of prediction models but also allows the integration of a broader array of data types beyond conventional financial data. Despite these advancements, a research gap remains in the comparative analysis and effectiveness of these ML models when integrated with traditional data sources in predicting bankruptcy. Most existing studies have focused on optimizing individual models without a comprehensive examination of hybrid approaches that combine traditional financial metrics with modern machine learning techniques.

The primary objectives of this study are: (1) to address the challenge of class imbalance in bankruptcy prediction using advanced resampling techniques; (2) to enhance model performance through optimal feature selection and hyperparameter tuning; (3) to compare the effectiveness of different ML models in the context of bankruptcy prediction; and (4) to propose a robust, adaptable framework for improving predictive accuracy in financial scenarios.

Using the UCI Taiwanese Bankruptcy Prediction Data Set from 1999 to 2009, this research compares several ML models, such as Random Forest (RF), Support Vector Machine (SVM), and Artificial Neural Network (ANN), to reduce this gap. We employ advanced resampling methods such as SMOTE-Tomek and SMOTE-ENN and feature selection through Binary Particle Swam Optimization (BPSO) to improve the predictive accuracy of our models. This approach not only allows us to determine which model

offers the highest predictive performance or the lowest relative error but also contributes to the field by demonstrating the effectiveness of integrating ML techniques. Based on the evaluation metrics used, the enhanced ANN model, which incorporates parameter tuning, a feature selection algorithm, SMOTE-ENN, and finely tuned hyperparameters, shows remarkable superiority over conventional models. It has achieved an F_1 Score of 98.5% and an accuracy rate of 98.6%. These results underscore the significant advantages of integrating various analytical strategies to boost the efficiency of models for predicting bankruptcy.

The rest of the paper is organized as follows. The relevant literature is reviewed in Section 2. Section 3 outlines the methodology, including the conceptual design description, approach comparison, and selection. Section 4 summarizes and analyzes the empirical results and discusses the findings. Finally, Section 5 presents the conclusions and offers insights for further research.

2. Literature Review

2.1. Data Resampling

A skewed or biased distribution of examples across known classes is referred to as an imbalanced classification problem. The number of instances in a single class, known as the majority, is greater than the amount of data in the minority class. The imbalance ratio can vary dramatically, with one example in the minority class being compared to hundreds or thousands of examples in the majority class. The data obtained from the Taiwan Economic Journal (1999–2009) is used in this research. It contains 6819 instances and 96 attributes with two categories. Figure 1 demonstrates the massive imbalance with 6599 non-bankruptcy enterprises (96%) and 220 bankruptcy enterprises (4%). Generally, resampling methods can be categorized into three types. First, by eliminating instances from the majority class, undersampling techniques address class imbalance. The second type, oversampling methods, involves creating a larger dataset by replicating instances from the minority class. Finally, hybrid methods combine aspects of both undersampling and oversampling techniques.

2.1.1. Undersampling Methods

Under-sampling methods include Random Undersampling (RUS), which aims to achieve a balanced dataset by randomly eliminating examples from the majority class, allowing for adjustment of the final balancing ratio. Tomek Links (Tomek) identifies pairs of instances from opposite classes that are nearest neighbors to each other and removes only examples from the majority class. The Condensed Nearest Neighbor Rule (US-CNN) reduces the dataset for k-nearest neighbor classification by eliminating examples from the majority class that are distanced from the decision boundary. The Neighborhood Cleaning Rule (NCL) combines the CNN Rule and the Edited Nearest Neighbors (ENN) Rule to remove superfluous and noisy/ambiguous examples, respectively. Class Purity Maximization (CPM) discovers pairs of centers, one from the minority class and one from the majority class, and divides the remaining instances into subsets with high-class purity. This process is repeated until no further clusters can be formed. Undersampling Based on Clustering (SBC) divides samples into k clusters using a clustering algorithm and randomly selects majority samples based on the proportion of majority to minority samples within each cluster. NearMiss approaches refer to a group of undersampling strategies that select samples based on the disparity between majority and minority class examples.

2.1.2. Oversampling Methods

One such oversampling technique is the SMOTE, which generates synthetic examples by interpolating between existing minority instances. These synthetic examples are generated by considering the feature space rather than the data space, concentrating on positive instances that are close to each other (Chawla et al., 2002) (Verma et.al, 2019). ADASYN is another type of oversampling technique that creates new instances of the minority category depending on the distribution and the degree of learning difficulty. Based on the ideas of correcting class imbalance and shifting the classification decision border to more difficult samples, ADASYN optimizes the distribution of data.

2.1.3. Hybridizations of Undersampling and Oversampling

The objective of this hybridization technique is to find an optimal trade-off between removing the majority of examples and creating new minority examples. Over time, several hybrid approaches have been developed by combining SMOTE with subsequent cleaning techniques applied to the entire dataset such as Tomek or ENN. Techniques such as SMOUTE (SMOTE with k-means) and CSMOUTE (Synthetic Majority Undersampling Technique and SMOTE) have been introduced (Koziarski, 2021).

2.2.   Feature Selection

The FS method identifies a set of representative variables from a given dataset, where the reduced-dimensional training dataset holds greater discriminatory power in distinguishing between classes than the original dataset. Previous research has demonstrated that a model produced using the dimension-reduced dataset is likely to perform better than a model built with the original dataset (Huynh et al., 2022; Le et al., 2021a; Le, Pham and Dao, 2021b; Le, Van Tran and Dao, 2021c; Le et al., 2020b; Guyon and Elisseeff, 2003). The four primary processes in the feature selection process are usually subset creation, evaluation, stopping criteria, and result validation. First, various potential feature subsets are created using a specific search strategy. Each candidate subset is then assessed based on a particular criterion and compared to the current best subset. Finally, the optimal feature subset is validated using test data or prior knowledge.

Feature selection can be approached through three main methods: Wrapper methods, Filter methods, and Heuristic search algorithms. Filter methods primarily focus on the characteristics of the data and can be seen as a preprocessing step. These methods evaluate features independently, which can lead to the oversight of possible data redundancy. They are quick in terms of processing time. The features are assigned scores based on a suitable ranking criterion, and those features that fall below a specified threshold are eliminated. Examples of filter methods consist of the Fisher Score, Mutual Information, and Pearson Correlation Criteria (Chandrashekar and Sahin, 2014). Wrapper methods treat the task (e.g., regression or classification) with the performance of the selected feature subset determining the quality of the selection. These methods have the capability to handle redundancy but are often computationally expensive. Moreover, they run the risk of overfitting when the available data is insufficient. Another method for selecting features is to use heuristic search algorithms, including PSO and Genetic Algorithm (GA). The chosen subset aims to minimize the feature dimension while maximizing the model's performance.

In summary, filter methods concentrate on the data features independently, wrapper methods consider the overall performance of the selected feature subset, and heuristic search algorithms explore the search space to seek a feature subset maximizing model performance while minimizing feature dimension.

ML is one of the computer science fields that is expanding the fastest. It is well adapted to the complexities of dealing with disparate data sources, a variety of variables, and a huge amount of data, where machine learning thrives on increasing datasets (Osisanwo et al., 2017). In this study, Machine Learning mechanisms are deeply combined with data resampling methods, feature selection algorithm, and hyperparameter optimization to improve the prediction results. Our proposed methods are designed to distinguish between bankrupt companies and those that are not based on financial factors. The next section will review each of the mechanisms including SVM, RF, and ANN with specific objectives, mathematical modeling, and learning algorithms.

3.   Methodology

3.1. Conceptual Design Description

The conceptual design model describes steps to apply different algorithms to handle binary class data. The main steps with suitable methods have been listed in the below sections. The process of the proposed model is presented in Figure 1.

The data processing step extracts features and resamples data using SMOTE-Tomek or SMOTE-ENN then selects important features by applying feature selection methods (Binary PSO). The selected features are trained in the next step and optimized by Hyperparameter optimization. Then, three ML models such as SVM, ANN, and RF are implemented to predict outputs, and the model's performance outcomes are compared and validated with the baseline model where the collected dataset is directly trained by machine learning models without any preprocessing steps, which described in Figure 2. Since the complexity of data processing, the results of these combinations outperform the results of other methods (Chou, Hsieh and Qiu, 2017; Liang et al., 2016; Liang, Tsai and Wu, 2015) which are known as only using the simple structure of baseline algorithms.
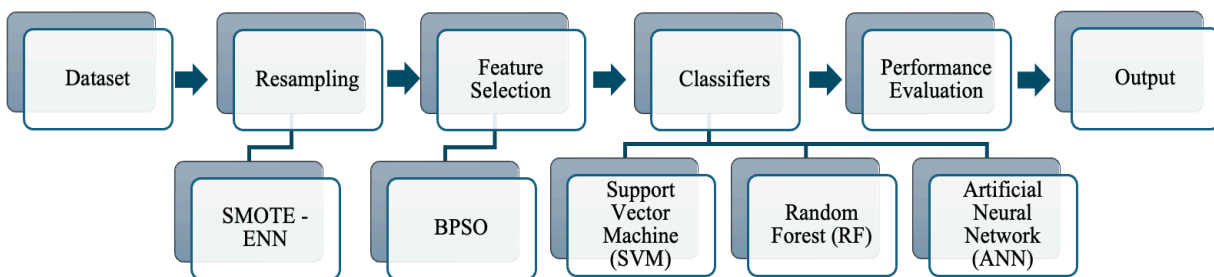

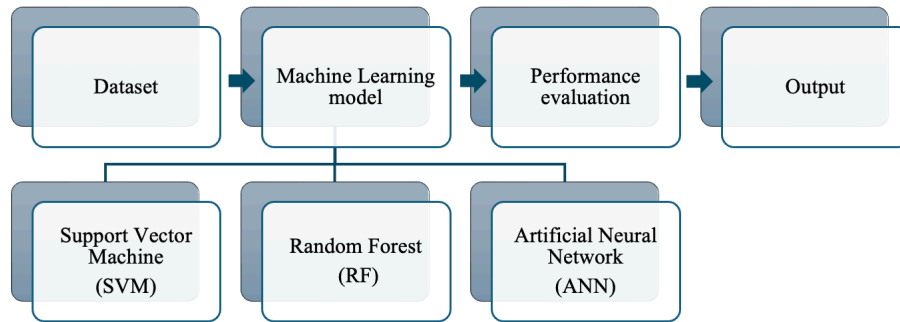**Figure 1** The process for developing the proposed model

Figure 2 The process for developing a baseline model

3.2. Approaches Comparison and Selection

3.2.1. Collection of Bankruptcy dataset

The bankruptcy dataset was retrieved from the UCI Taiwanese Bankruptcy Prediction Data Set. The business regulations of the Taiwan Stock Exchange were used to define company bankruptcy. As described above, the dataset comprised 6819 instances based on 96 attributes. The binary class dataset describes the instance as "Bankrupt" (6599 cases) or "non-bankrupt" (220 cases). This dataset has comprehensive financial attributes and a clear definition of bankruptcy. Additionally, the significant class imbalance in this dataset reflects real-world challenges in predicting rare events like bankruptcy, making it an ideal benchmark for testing advanced machine learning and resampling techniques.

3.2.2. Data Exploration and Deduplication

The dataset composed of a combination of 96 variables and 6819 observations. All of the features are numerical with integer numbers (3 features) or float numbers (93 features) and the data does not contain any missing values (Nan). However, features with duplicated values or have the same values as other features are removed, such as "Net Income Flag", "Current Liabilities/Liability", " Current Liabilities/Equity" and "Gross Profit to Sales". The dataset is unbalanced, as the non-bankruptcy group dominates when its population is roughly 32 times more than that of the remaining class.

3.2.3. Data-level Techniques and Data Splitting

In this study, two hybrid approaches are employed for the data sampling step, combining the SMOTE with Tomek Link and the SMOTE with ENN (Batista, Prati and Monard, 2004). To increase the representation of the smaller class, the K-Nearest Neighbor (KNN) algorithm is first applied with SMOTE. This approach avoids randomly duplicating minority data. Additionally, the Tomek link is utilized to identify and remove points that extend into the region of another class. This step is crucial in mitigating potential issues related to overlap or overfitting. Unlike simple under-sampling, it should be noted, that occurrences of both types can be eliminated. On the other hand, SMOTE-ENN employs a comparable principle to SMOTE-Tomek. Nevertheless, ENN identifies points for deletion using the KNN algorithm. Specifically, when the majority class of the selected point's K-nearest observations is misclassified, those points are eliminated from the dataset. SMOTE-ENN is anticipated to yield a more refined dataset compared to SMOTE-TOMEK due to the elimination of a larger number of instances. The SMOTE-ENN technique, while effective for addressing class imbalance, has limitations. It may remove informative instances near class boundaries, leading to potential loss of critical data, and is computationally intensive compared to simpler methods.

The training set and the testing test are the two distinct subsets of the data that are divided out before the pre-processing step. The testing set remains for evaluating the

method, while the training set is used as a training sample for the model. The ratio for splitting is 80% training and 20% testing.

3.2.4. Feature Extraction and Correlation Analysis

The number of features not only affects the accuracy but also affects the time to train a model. The reason is that the dataset has many features that will make the training and classification process take more time. In addition, it also causes the program to take up more memory and hard disk space. Consequently, for ML methods, it is essential to choose from a smaller subset of characteristics while maintaining the classification process's accuracy. This process is referred to as feature selection, and it is also referred to as feature reduction, attribute selection, or variable subset selection. Pearson's correlation coefficient (r) is a well-known method for measuring the statistical relationship or association between features to each other and between features and labels. Therefore, this study conducts correlation analysis using this test to obtain information about the association of every pair of variables in the dataset using Equation (1).

$$r = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum(x_i - \overline{x})^2 \sum(y_i - \overline{y})^2}} \tag{1}$$

Where, $\overline{x}$, $\overline{y}$ are mean values of two variables x and y.

After that, the Pearson correlation heatmap was plotted and indicated in Figure 3 (Feature names on the x and y axes have been scaled to preserve the length of the figure and ensure that each letter is easily visible). The Pearson correlation coefficient (r) varies in the continuous range from -1 to 1, two features have no linear correlation while r equals 0, and two features have an absolute linear relationship when r equals -1 or 1. If the coefficient is 1, it is considered to be a positive correlation. This indicates that while one variable increases, the other increases as well. If the coefficient is -1, it is considered to be a negative correlation. When training the model, if the correlation between the predictors is larger than 0.9 or less than -0.9, one of these variables can be eliminated as a predictor. The shortlist of the feature pairs with high correlation scores can be seen in Table 1.

The columns of "Debt ratio %", "ROA(B) before interest and depreciation after tax", "ROA(C) before interest and depreciation before interest", "Operating Profit Rate", "Working capital Turnover Rate", "Persistent EPS in the Last Four Seasons", "Borrowing dependency", "Net Income to Total Assets", "Per Share Net profit before tax (Yuan ¥)", "Current Liability to Equity", "After-tax net Interest Rate", "Continuous interest rate (after tax) ", "After-tax Net Profit Growth Rate", "Operating profit/Paid-in capital" , "Net Value Per Share (B)", "Net Value Per Share (C)" and "Realized Sales Gross Margin" are dropped to avoid redundancy so that there are 75 variables remaining in the dataset.

Table 1 The list of the feature pairs with high correlation

| Feature 1	Feature 2	Correlation coefficient (r)
Net worth/Assets	Debt ratio \%	-1
Net Income to Total Assets	ROA(B) before interest and depreciation after tax	0.912
Continuous interest rate (after tax)	Operating Profit Rate	0.916
Pre-tax net Interest Rate	Operating Profit Rate	0.916
ROA(C) before interest and depreciation before interest	ROA(A) before interest and \% after tax	0.940
Cash Flow to Sales	Working Capital Turnover Rate	0.948
Per Share Net profit before tax	Persistent EPS in the Last Four Seasons	0.956
ROA(B) before interest and depreciation after tax	ROA(A) before interest and \% after tax	0.956
Liability to Equity	Borrowing dependency	0.956
Net profit before tax/Paid-in capital	Persistent EPS in the Last Four Seasons	0.959
Net Income to Total Assets	ROA(A) before interest and \% after tax	0.962
Net profit before tax/Paid-in capital	Per Share Net profit before tax	0.963
Liability to Equity	Current Liability to Equity	0.964
Continuous interest rate (after tax)	After-tax net Interest Rate	0.984
Pre-tax net Interest Rate	After-tax net Interest Rate	0.987
ROA(C) before interest and depreciation before interest	ROA(B) before interest and depreciation after tax	0.987
Continuous interest rate (after tax)	Pre-tax net Interest Rate	0.994
Regular Net Profit Growth Rate	After-tax Net Profit Growth Rate	0.996
Operating Profit Per Share	Operating profit/ Paid-in capital	0.999
Net Value Per Share (B)	Net Value Per Share (C)	0.999
Net Value Per Share (B)	Net Value Per Share (A)	0.999
Realized Sales Gross Margin	Operating Gross Margin	0.999
Net Value Per Share (A)	Net Value Per Share (C)	0.999
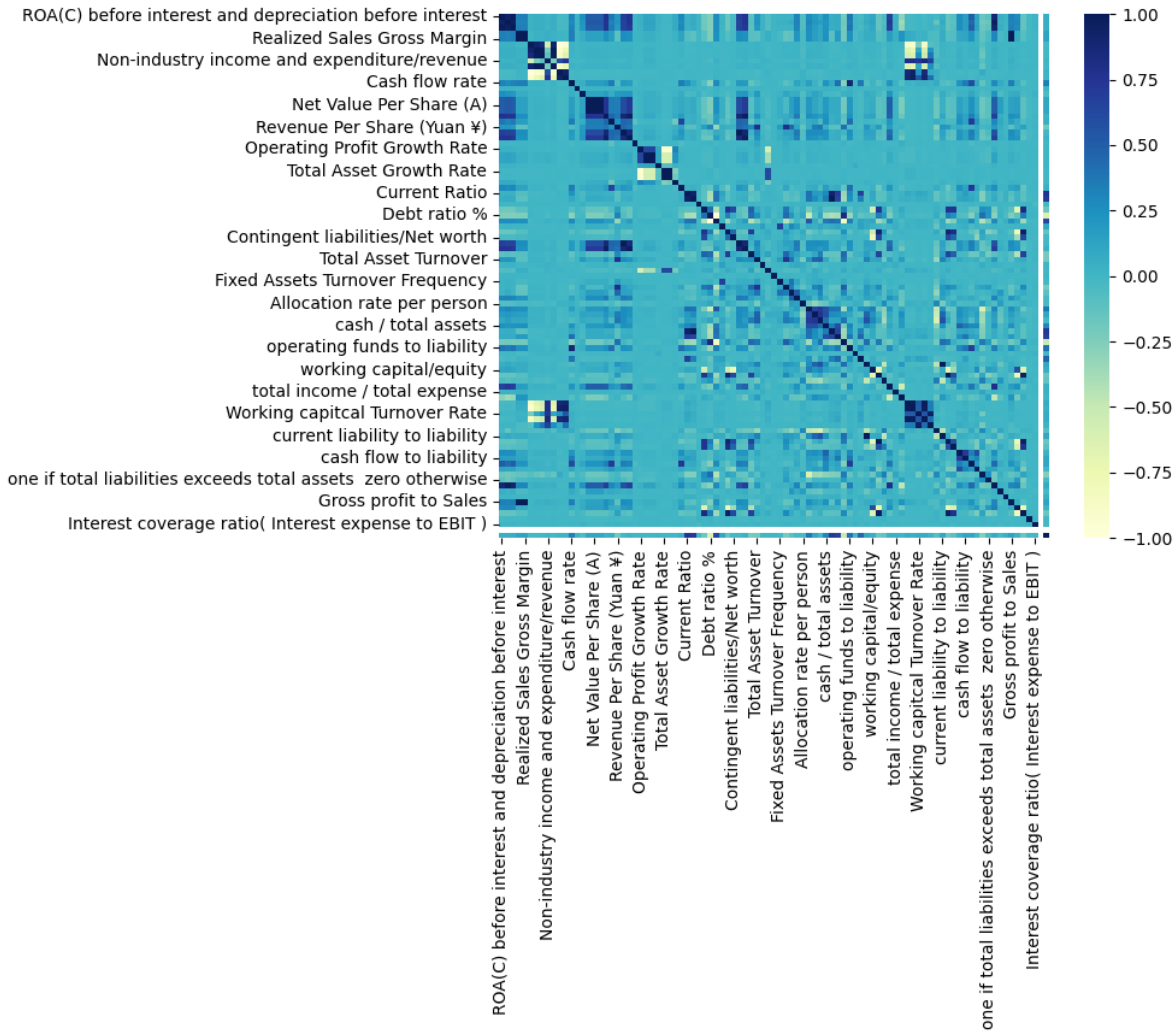
Figure 3 Pearson correlation heatmap for original features

3.2.5. Data Standardization

The technique of standardization is widely employed in numerous ML models such as SVM, and ANN, for normalization. The process of converting data into a unified format to enable user processing and analysis is known as data standardization. The importance of data standardization stems from various factors. Each data column exhibits a distinct range of values, some including negatives and other positives; there is a mix of integers and decimals as well. Consequently, without converting the data into a standardized format, it becomes arduous to compare and analyze them effectively. Upon implementing the Data Standardization step, a concise summary of the converted dataset is presented in Figure 4. To achieve dataset normalization using standardization, Equation (2) is applied to transform each x value within the dataset into its corresponding x_{stand} value.

$$x_{stand} = \frac{x - \overline{x}}{\sigma} \tag{2}$$

where x is the original feature vector.

	Flag	ROA(C) before interest and depreciation before interest	ROA(A) before interest and % after tax	ROA(B) before interest and depreciation after tax	Operating Gross Margin	Realized Sales Gross Margin	Operating Profit Rate	Pre-tax net Interest Rate	After-tax net Interest Rate	Non-industry income and expenditure/revenue	...
0	1	0.370594	0.424389	0.405750	0.601457	0.601457	0.998969	0.796887	0.808809	0.302646	...
1	1	0.464291	0.538214	0.516730	0.610235	0.610235	0.998946	0.797380	0.809301	0.303556	...
2	1	0.426071	0.499019	0.472295	0.601450	0.601364	0.998857	0.796403	0.808388	0.302035	...
3	1	0.399844	0.451265	0.457733	0.583541	0.583541	0.998700	0.796967	0.808966	0.303350	...
4	1	0.465022	0.538432	0.522298	0.598783	0.598783	0.998973	0.797366	0.809304	0.303475	...
...											
6814	0	0.493687	0.539468	0.543230	0.604455	0.604462	0.998992	0.797409	0.809331	0.303510	...
6815	0	0.475162	0.538269	0.524172	0.598308	0.598308	0.998992	0.797414	0.809327	0.303520	...
6816	0	0.472725	0.533744	0.520638	0.610444	0.610213	0.998984	0.797401	0.809317	0.303512	...
6817	0	0.506264	0.559911	0.554045	0.607850	0.607850	0.999074	0.797500	0.809399	0.303498	...
6818	0	0.493053	0.570105	0.549548	0.627409	0.627409	0.998080	0.801987	0.813800	0.313415	...

Figure 4 Dataset after the Data Standardization step

3.2.6. Feature Selection

The PSO algorithm is a popular metaheuristic technique that draws inspiration from the collective behaviors observed in certain species in nature. PSO has demonstrated that this method converges to the optimal solution. So, one of the elements that form the key idea of PSO is social interaction among the population's candidates. In this search algorithm, a fitness function can be utilized. Since the wrapper technique wraps itself around the induction algorithm, it obtains better predictive accuracy estimates than filter models (Tang, Alelyani and Liu, 2014). In this research, Binary PSO is selected as the heuristic search algorithm.

In the PSO approach, each particle, representing a potential solution, exists as a point within a multi-dimensional search area. These particles possess personal memories, which retain the most successful solutions they've encountered individually and as a collective swarm. The movement of each solution through this search domain is governed by a velocity that evolves dynamically, shaped by its past successes and the experiences of other particles in the swarm.

Initially, the swarm's particles are dispersed throughout the search space. The location of each particle is denoted by a vector, which corresponds to the dimensionality D of the search space. The velocity of the search ($v_i = v_{i1} + v_{i2} + v_{i3} + \cdots + v_{iD}$) increases as each particle with coordinates ($x_i = x_{i1} + x_{i2} + x_{i3} + \cdots + x_{iD}$) travels in the search space to locate the best solution. As they move, particles modify their positions and speeds, drawing on their own experiences and those of their neighbors. Every particle has a memory that records its most successful location, known as p_{best}. The entire swarm's optimum experience is referred to as the global best, symbolized by g_{best}. Every time a particle iterates, its location, and velocity are formatted using Equations 3 and 4 (Chawla et al., 2002):

$$x_{id}(t + 1) = x_{id}(t) + v_{id}(t + 1) \tag{3}$$

$$v_{id}(t+1) = w \times v_{id}(t) + c_1 \times r_1 \times \left(p_{id} - x_{id}(t)\right) + c_2 \times r_2 \times \left(p_{gd} - x_{id}(t)\right) \tag{4}$$

where t is iteration, $d \in D$ is a dimension in the search space, w is inertia weight, c_1 and c_2 are cognitive and social acceleration coefficients, r_1 and r_2 are randomly and uniformly distributed numbers in the range of [0, 1]. p_{id} (is also p_{best}) represents the local best in dimension d^{th}, while g_{id} indicates the global best in dimension d^{th}, which is also referred to as g_{best}.

The search algorithm comes to a halt when a predetermined stopping criterion is satisfied. This may differ in certain scenarios, such as feature selection and other optimization problems (Mafarja et al., 2018), which involve discrete search spaces. To tackle this challenge, a discrete binary variant of PSO known as BPSO, which is tailored for solving optimization problems in discrete domains. In BPSO, the update rule for the velocity remains unchanged from the original PSO. the key difference is that the variables x_{id}, p_{id} and p_{gd} can only take binary values of 0 or 1. As a result, the velocity indicates the probability that a particle in the position vector will have a value of 1. In BPSO, the position of the particle is dictated in Equation (5), utilizing the probability $T(V_t)$ received from Equation (6).

$$x(t+1) = 1, rand < S\left(v(t+1)\right) or = 0, otherwise \tag{5}$$

$$S\left(v(t+1)\right) = \frac{1}{1 + e^{-v(t)}} \tag{6}$$

where $S(v(t+1))$ is the Sigmoid function and rand is a random number in the range [0,1].

The wrapper algorithm is guided by a fitness function, which evaluates the quality of the optimizer's solutions. Both the selected features and the classification performance should be taken into account by this function. The best solution should reflect a subset of features that yields lower features and higher classification performance value. The fitness function inspired by the work of Vieira et al. (2013) is used in Equation (7):

$$Fitness function = \alpha(1 - P) + (1 - \alpha)\left(1 - \frac{N_f}{N_t}\right) \tag{7}$$

where $\alpha \in [0,1]$ indicates the trade-off between the classification performance's error rate, and P is the classifier performance measure. N_f is the size of the tested feature subset and N_t is the total number of available input variables. It is important to note that both terms in the objective function are normalized. The value $\alpha = 0.99$ is also adopted.

3.2.7. Hyper-parameter Optimization

The process of tuning hyperparameters (HPs) plays a crucial role in optimizing the performance of the ML model. Hyperparameters (HPs) are crucial because they define the model's structure before training and can either configure the ML model or determine the algorithm used for performance optimization (like the kernel type in SVM). Manual tuning of hyperparameters has traditionally been used but is often ineffective due to the large

number of parameters, complex models, and time-consuming evaluations (Yang and Shami, 2020). To overcome these challenges, Hyperparameter Optimization (HPO) has been developed.

HPO aims to automate and improve the hyperparameter tuning process. The Random Search (RS) method is chosen as the main HPO technique. For each classifier, RS will be executed for 20 iterations on the training set, utilizing the hyperparameter search space outlined in Table 2 for each model. By employing HPO, the expectation is to achieve an optimal machine learning model.

In this study, the rationale for selecting specific hyperparameters for each model was carefully considered to ensure optimal performance. For the SVM model, the kernel options (linear, polynomial, sigmoid, and RBF) were selected to address both linear and non-linear relationships within the data. The C parameter, which balances bias and variance, was varied between 0 and 50 to find an optimal trade-off, with higher values reducing bias at the expense of increased variance. In the case of Random Forest, the maximum tree depth max_{depth} was explored in the range of 5 to 50 to control tree complexity, while $min_{samples\ leaf}$ (2–11) and $min_{samples\ split}$ (2–11) were adjusted to ensure stable trees. The number of estimators ($n_{estimator}$ was varied between 10 and 100 to balance model stability and efficiency, and the Gini criterion was preferred for its computational efficiency, with entropy serving as an alternative. For the ANN, a variety of activation functions (softmax, ReLU, tanh, sigmoid, and linear) were evaluated to introduce non-linearity into the model. The Adam optimizer was chosen to facilitate faster convergence, and the number of neurons (ranging from 5 to 50) was fine-tuned to mitigate the risk of overfitting. These choices collectively reflect a systematic approach to hyperparameter optimization, tailored to the unique requirements of each model.

Table 2 Hyper-parameters Search Space

Classifier	Hyper-parameters Search Space
SVM	kernel: linear, poly, sigmoid, rbf
	C: float numbers in the range (0,50)
RF	max_{depth}: integer numbers in range (5,50)
	$min_{samples\ leaf}$: integer numbers in range (2,11)
	$min_{samples\ split}$: integer numbers in range (2,11)
	$n_{estimator}$: integer numbers in range (10,100)
	$criterion$: gini, entropy
ANN	Activation: softmax, relu, tanh, sigmoid, linear
	Optimizer: SGD, Adam, Adamax
	Neurons: 5, 10, 15,20, 25, 30, 35,40, 45, 50

3.2.8. Implementing Machine Learning Model

This paper implements SVM, RF, and ANN as ML models to validate and compare the results. These algorithms were chosen for their complementary strengths: SVM excels in handling high-dimensional data and binary classification, RF is robust in managing diverse datasets and ranking feature importance, and ANN is highly effective in capturing non-linear relationships and complex patterns. Their selection ensures a comprehensive evaluation of predictive performance in bankruptcy detection.

The purpose of the SVM model is to train a model that can classify data points within a given feature space. The fundamental idea surrounding SVM is the concept of a margin, which is essentially an area surrounding a hyperplane for classifying different classes of information. SVM identifies the best hyperplane that can best separate these classes with the maximum distance referred to as the margin to the closest points that are called the support vectors. In cases where data is non-separable, as in potential decision areas, potential functions using kernels like Linear Polynomials or Gaussians (radial basis functions) are used so that the problem becomes separable in a higher dimension. (Sekeroglu, Hasan and Abdullah, 2020). The separation hyperplane can be mathematically defined using Equation 8.

$$r = (w\char94 Tx + b)/\|w\| \tag{8}$$

where w is parameters, b is the distance from an example x_i. There are two main hyper-parameters of SVM that are used to tune the model in the proposed approach: C is the regularization parameter and the kernel function type to be used in SVM.

RF consists of multiple Decision Trees, each built from a randomly chosen subset of data samples. The process of building these trees involves selecting features at random during their development. For making predictions, RF aggregates the outcomes from all the trees in the ensemble, with the final prediction being the one that receives the majority of votes. Notably, each tree within the RF is allowed to grow to its maximum depth, as there is no pruning involved in this methodology (Mafarja et al., 2018).

For RF, there are five hyper-parameters used for HPO:

- $n_{estimator}$: total number of trees.

- max_{depth}: the maximum depth allowed for the tree to grow.

- $min_{samples\ leaf}$: the minimum number of samples allowed to be in a leaf node.

- $min_{samples\ split}$: the minimum number of samples needed to split an internal node.

- $criterion$: the function assesses the quality of each tree's split

ANN refers to a special learning model that simulates how synapses work in the human brain. ANN consists of three primary parts: Input and output layers have only one layer each, whereas hidden layers might have one or more levels depending on the issue. ANN aims to describe how the nervous system functions by using neural connections.

3.2.9. Determining the performance of models

The performances of classifiers are assessed in terms of several measures such as Confusion Matrix, Accuracy, Precision, Recall, and F_1 measure. These metrics give the overall evaluation of the classifier as various sides of the model's prediction are examined. Since the classification is made between Bankruptcy and Non-Bankruptcy, i.e., a two-class classification problem, the ordinary Confusion Matrix is used to measure our classification model. There are four possible classification outcomes:

- True-positive (TP): points of bankruptcy are exactly classified.

- False-positive (FP): points are classified into the bankruptcy class, but in fact, they belong to non- bankruptcy class.

- True-negative (TN): The points of the non-bankruptcy class are perfectly defined the non-bankruptcy class is outlined quite clearly.

- False-negative (FN): when classifying the points, some of them are classified into the non-bankruptcy class, however, they belong to the bankruptcy class.

Accuracy is the ratio of correct prediction to the total records, which shown in Equation (9).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{9}$$

The precision is the ability of the classifier to not categorize as positive a sample that is negative. The worst value is 0 and the best value is 1 and Equation (10) is as follows:

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

The recall can be defined as the ability of a classifier to find all of the positive samples. Its values range from zero to 1 (optimum), the recall is as Equation (11):

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

The F_1 score is calculated by taking the precision and recall harmonic mean, the Equation (12) of the F_1 score depends completely upon precision and recall and is as follows:

$$F1score = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \tag{12}$$

Its minimal value (of 0) is obtained whenever precision or recall is 0 (even if the other one is 1) and its maximal value (of 1) is obtained when both of them are 1.

4.   Results and Discussions

4.1. Baseline Results

In the baseline model, Oversampling techniques, FS method, and HPO are not performed. All 75 features will be classified by three classifier algorithms with the hyper-parameters: SVM with C =1; RF with max_{depth}: None, $min_{samples\ leaf}$: 1, $min_{samples\ split}$: 2, $n_{estimator}$: 100, $criterion$: gini; and ANN with Activation = linear, Optimizer = Adam, Neurons: 5

Four popular types of kernels in SVM are applied to find out which kernel is the most appropriate for the dataset. All parameters are set at the default values especially C equal to 1.

RF is a supervised ML algorithm based on the ensemble learning concept in which multiple Random Forest classifiers are combined together, so it has both the simplicity of a Decision Tree and the flexibility to solve complex problems. All parameters are set at the default values as above table.

In the ANN baseline model, the model will have three layers, namely one input layer, one hidden layer, and one output layer. The input layer uses 5 nodes and requires relative activation. Then the output layer uses one number of nodes and the sigmoid activation function. Since the dataset is partitioned randomly, the program runs 5 times to get an objective observation and accuracy conclusion. The results with average F1 scores and accuracies are shown in Table 3.

Table 3 Evaluation metrics of baseline models.

Model	F_1 Score	Accuracy	Precision	Recall
SVM - RBF Kernel	0.0177	0.968	0.02	0.0046
SVM - Linear Kernel	0.1137	0.968	0.4388	0.0636
SVM - Poly Kernel	0.1878	0.967	0.428	0.1044
SVM - Sigmoid Kernel	0.2074	0.962	0.3348	0.1592
RF	0.1942	0.969	0.412	0.1416
ANN	0.216	0.964	0.364	0.155

4.2. Results of Model Improvement

4.2.1. The advanced approach using balancing techniques (SMOTE-Tomek and SMOTE-ENN)

SMOTE-Tomek and SMOTE-ENN are applied to make the dataset balanced. Figure 5 shows that two classes which are bankruptcy and non-bankruptcy have the same ratio as well as the same number of instances after using balancing techniques and Table 4 shows the average F_1 scores and accuracy of models after balancing data implementation.
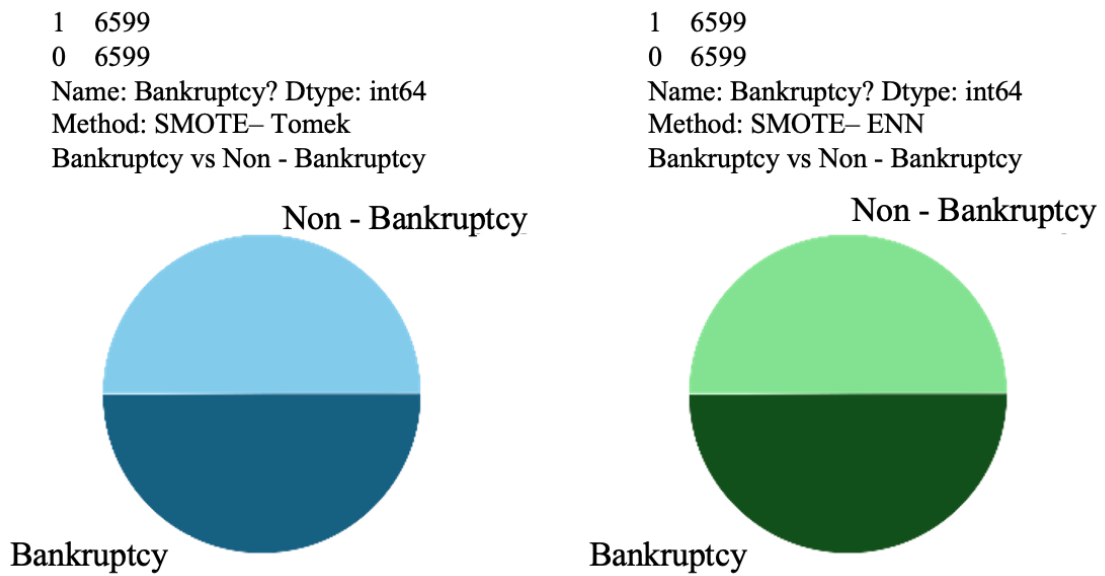
1   6599   1   6599
0   6599   0   6599
Name: Bankruptcy? Dtype: int64   Name: Bankruptcy? Dtype: int64
Method: SMOTE– Tomek   Method: SMOTE– ENN
Bankruptcy vs Non - Bankruptcy   Bankruptcy vs Non - Bankruptcy

Figure 5 The ratio of Bankruptcy to non-bankruptcy after using Data balancing (SMOTE-Tomek and SMOTE-ENN).

Table 4 Average F1 score and Accuracy of models after using SMOTE-Tomek and SMOTE-ENN balancing data techniques.

Model	F1 Score		Accuracy	
	SMOTE-Tomek	SMOTE-ENN	SMOTE-Tomek	SMOTE-ENN
SVM - RBF Kernel	0.934	0.952	0.931	0.948
SVM - Linear Kernel	0.9	0.921	0.899	0.919
SVM - Poly Kernel	0.835	0.872	0.85	0.883
SVM - Sigmoid Kernel	0.85	0.886	0.85	0.885
RF	0.978	0.981	0.978	0.98
ANN	0.962	0.968	0.961	0.97

4.2.2. Feature selection using Binary PSO

The feature selection process using Binary PSO (BPSO) was configured with specific parameter values to ensure efficient optimization. The process runs for a maximum of 1000 iterations. The cognitive coefficient (c_1) and the social coefficient (c_2:) were both set to 0.7, reflecting a balanced influence between individual and social components in the particle updates. The inertia weight (w) was set at 0.5 to maintain a balance between exploration and exploitation during the search process. Each particle considers 75 neighbors (k) in its decision-making, and the total number of particles in the swarm was set to 75, ensuring sufficient diversity in the search space.

Figure 6 shows the convergence curve (cost history) of the fitness function during the iterations. The FS-BPSO algorithm achieves an optimal fitness value of $best_{cost}$ = 0.1218). Notably, FS-BPSO selects 34 features from the original set of 75 features, resulting in a reduction of feature dimension by 55%. This method uses a particle swarm optimization (PSO) framework adapted to binary search spaces, enabling it to effectively evaluate feature

subsets by balancing feature relevance and redundancy. The specific features selected are listed in Table 5. These features include critical financial ratios (e.g., debt ratio, equity to liability), profitability indicators (e.g., net income to stockholder's equity, operating profit per share), liquidity and cash flow metrics (e.g., quick assets/current liability, cash flow to total assets), and operational efficiency metrics (e.g., total asset turnover, working capital to equity). Such features capture essential dimensions of financial health and operational sustainability, enhancing model performance. BPSO optimizes feature selection by minimizing dimensionality while maximizing classification accuracy, which ensures the inclusion of features with high predictive power. The method inherently prioritizes features that align with established financial theories, such as Altman's Z-score, which emphasizes the importance of financial ratios in assessing bankruptcy risk, as well as profitability, liquidity, and efficiency metrics that are widely recognized as key indicators of financial stability. This holistic feature set enhances the efficiency, and generalizability of the model, demonstrating the effectiveness of BPSO in identifying the most relevant and informative predictors for bankruptcy prediction.

A correlation heatmap for the 34 features that were selected is shown in Figure 7. The remaining features exhibit correlations that are either below 0.86 or above -0.86, indicating a weak to virtually no correlation among them.
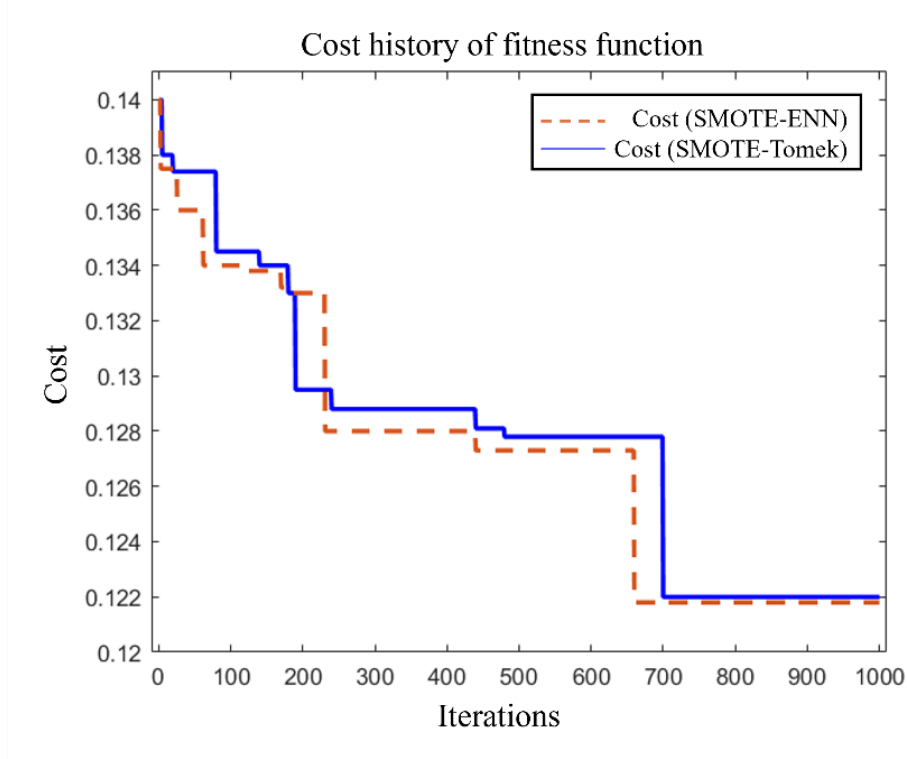

Figure 6 Convergence Curve (Cost History) of Fitness Function of balanced dataset (generated by SMOTE-Tomek and SMOTE-ENN.

Table 5 The list of selected features.

Selected features
X11: Operating Expense Rate: Operating Expenses/Net Sales
X12: Research and development expense rate: (Research and Development Expenses)/Net Sales
X14: Interest-bearing debt interest rate: Interest-bearing Debt/Equity
X15: Tax rate (A): Effective Tax Rate
X20: Cash Flow Per Share
X21: Revenue Per Share (Yuan Â¥): Sales Per Share
X22: Operating Profit Per Share (Yuan Â¥): Operating Income Per Share
X24: Realized Sales Gross Profit Growth Rate
X27: Regular Net Profit Growth Rate: Continuing Operating Income after Tax Growth
X29: Total Asset Growth Rate: Total Asset Growth
X30: Net Value Growth Rate: Total Equity Growth
X36: Total debt/Total net worth: Total Liability/Equity Ratio
X37: Debt ratio (\%) = Liability/Total assets
X38: Net worth/Assets: Equity/Total Assets
X39: Long-term fund suitability ratio (A): (Long-term Liability + Equity)/Fixed Assets
X41: Contingent liabilities/Net worth: Contingent Liability/Equity
X43: Net profit before tax or paid-in capital: Pretax income or capital
X45: Total Asset Turnover
X53: Allocation rate per person: Fixed Assets Per Employee
X58: Quick Assets/Current Liability
X60: Current Liability to Assets
X65: Working Capital/Equity
X69: Total income/Total expense
X72: Quick Asset Turnover Rate: Quick Assets to Sales
X74: Cash Turnover Rate: Cash to Sales
X76: Fixed Assets to Assets
X77: Current Liability to Liability
X80: Cash Flow to Total Assets
X85: Liability-Assets Flag: 1 if Total Liability exceeds Total Assets, 0 otherwise.
X90: Net Income to Stockholder's Equity
X91: Liability to Equity
X92: Degree of Financial Leverage (DFL)
X93: Interest Coverage Ratio (Interest expense to EBIT)
X95: Equity to Liability

4.2.3. Hyper-parameter Optimization

The hyper-parameters in Table 6 were obtained through the Random Search HPO method. Random search utilized accuracy as the scoring metric to determine the optimal hyper-parameters.

Table 6 Optimal Hyper-parameters Results with Random Search.

Classifier	Optimal Hyper-parameters found by Random Search	Accuracy of train set from Random Search	
		SMOTE-Tomek	SMTOE-ENN
SVM	kernel: rbf C = 40	0.951	0.963
RF	max_{depth}: 41 $min_{samples\ leaf}$: 2 $min_{samples\ split}$: 4 $n_{estimator}$: 90 $criterion$: entropy	0.974	0.978
ANN	Activation = tanh Optimizer = Adam Neurons: 30	0.976	0.985
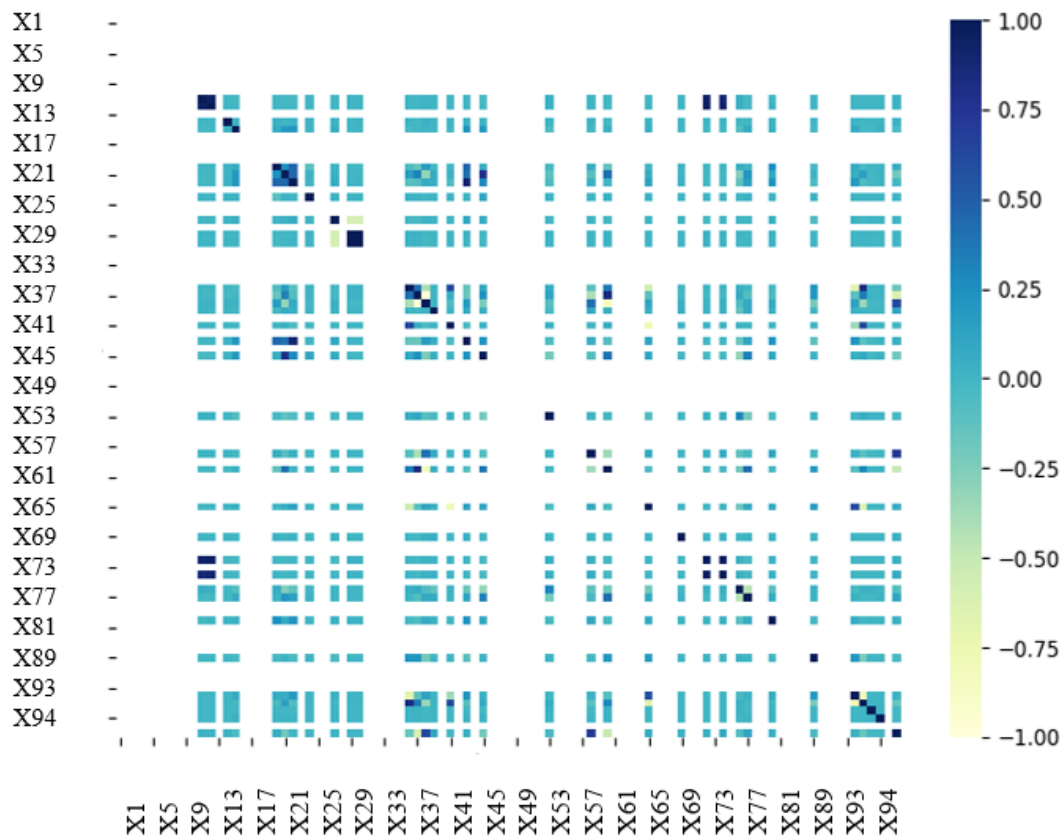
Figure 7 Pearson correlation heatmap.

4.3. Proposed Model Results

To perform the classification of two proposed models with SMOTE-Tomek and SMOTE-ENN on the test set, the classifiers are configured with the hyperparameters. The initial results of the proposed model, summarized in Table 7, are distinct from the baseline results. These initial results demonstrate significant improvements compared to both the baseline results (shown in Table 8) and the results from key references, as depicted in Table 10.

Among the classifiers, ANN with SMOTE-ENN achieved the highest accuracy of 98.5%. Their recall and F1 scores are also higher than those of the SVM and RF classifiers. The confusion matrices for the SVM, RF, and ANN classifiers for the two proposed models are presented in Figure 8 and Figure 9. Notably, ANN with SMOTE-ENN accurately predicted 1285 out of the total 1320 'non-bankruptcy' cases, while Random Forest with SMOTE-ENN correctly predicted 1271 out of 1320 'non-bankruptcy' cases. However, SVM's performance in predicting 'non-bankruptcy' instances was relatively weaker, with only 1254 cases correctly predicted and 23 instances falsely predicted. Furthermore, RF of the SMOTE-Tomek method accurately predicted 1266 out of the total 1320 'non-bankruptcy' cases, while the numbers of SVM and ANN in similar methods are respectively 1209 and 1278.

Table 7 Proposed Model Result Summary.

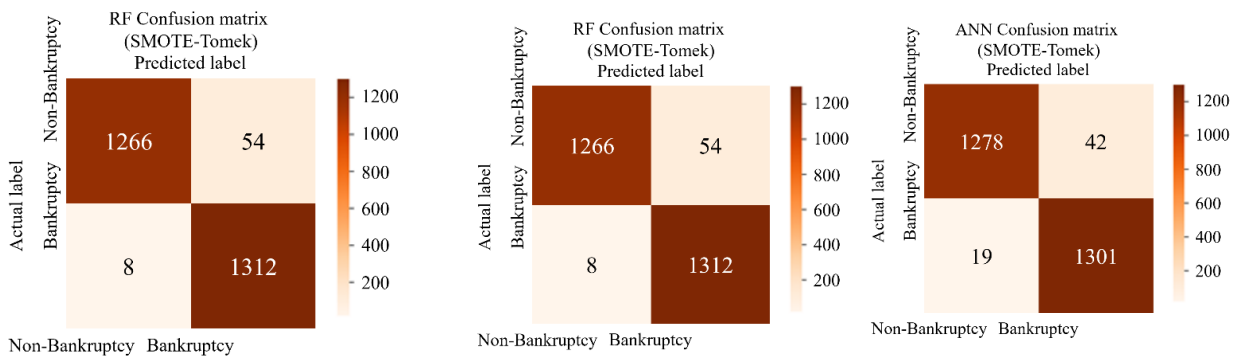|	Model	F1 Score	Accuracy	Precision	Recall
SMOTE -Tomek	SVM (RBF Kernel)	0.952	0.95	0.921	0.984
	RF	0.977	0.974	0.96	0.994
	ANN	0.977	**0.977**	0.969	0.986
SMOTE -ENN	SVM (RBF Kernel)	0.963	0.962	0.943	0.985
	RF	0.978	0.978	0.965	0.993
	ANN	0.985	**0.986**	0.979	0.992

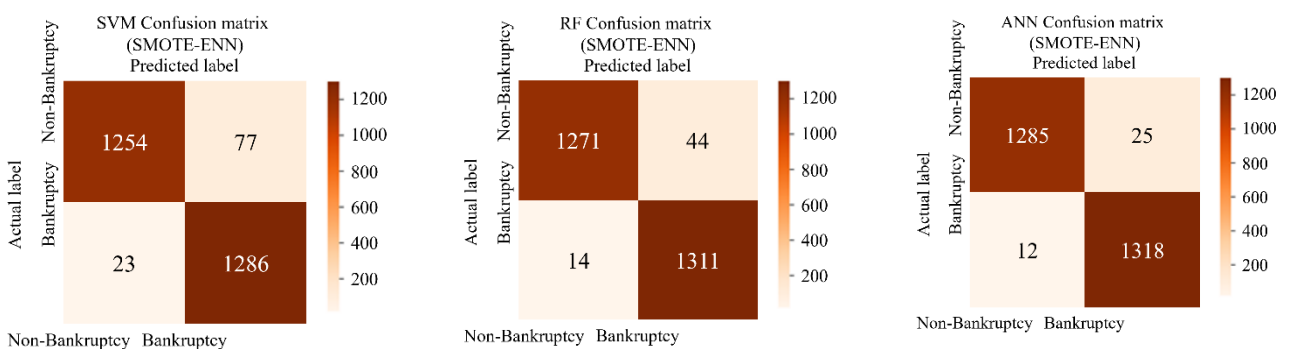**Figure 8** The confusion Matrices of SVM – RF – ANN model (SMOTE-Tomek).

Figure 9 The confusion Matrices of SVM – RF – ANN model (SMOTE-ENN).

4.4. Results Comparison

4.4.1. Compare With Baseline Results

Table 8 Comparison between the Suggested Models and Baseline Results.

	Model	F_1 Score	Accuracy	Precision	Recall
SMOTE-Tomek	SVM (RBF Kernel)	+0.934	-0.018	+0.901	+0.979
	RF	+0.783	+0.005	+0.548	+0.852
	ANN	+0.761	+0.013	+0.605	+0.831
SMOTE-ENN	SVM (RBF Kernel)	+0.945	-0.006	+0.923	+0.98
	RF	+0.784	+0.009	+0.553	+0.851
	ANN	+0.769	+0.022	+0.615	+0.837

The proposed model outperforms the baseline model, achieving more promising results while utilizing fewer features as shown in Table 9. The performance improvement is particularly notable in the SVM classifier. Other classifiers like RF and ANN also exhibit enhancements across all evaluation metrics for all of these two proposed methods. The primary challenge addressed in this study is the imbalance between the two classes in the dataset. To overcome this, the study employs resampling techniques, specifically SMOTE-Tomek and SMOTE-ENN. The results demonstrate the significant improvement achieved through the advanced approach. The application of these two methods has a crucial role in enhancing performance.

In particular, the RBF kernel SVM model, with parameter tuning (C=40), a balanced dataset generated by SMOTE-Tomek and SMOTE-ENN, and FS performed by Binary PSO, achieves remarkable performance. It attains an accuracy score of 0.95 and an F_1 score of 0.952, the RF model and ANN model achieve slightly higher scores with accuracy and F_1 score of 0.977 for SMOTE-Tomek while these numbers of SMOTE-ENN respectively 0.962, 0.978, 0.986 for Accuracy of SVM (RBF Kernel), RF and ANN model.

4.4.2. Compare With Key Reference

Compared to the three aforementioned references that were conducted on the same Taiwan bankruptcy dataset, the highlight of the proposed model is the application of a binary wrapper-based feature selection, which is the Binary PSO. Moreover, in the data pre-processing step, Pearson's correlation is conducted to eliminate high-correlation pairs of features, which has not been previously considered by Chou et al. (2017), Liang et al. (2016) and Liang et al. (2015). All three reference papers used a method of stratified sampling and multilayer perceptron to collect the same number of bankrupt and non-bankrupt cases. This study also tries model improvement approaches including hyper-parameters tuning and thresholds optimization. With all these innovation aspects, the proposed model trains the dataset with a small number of selected features, which is less computationally expensive and may avoid overfitting compared to previous works on the same problem as discussed in Table 10.

The results clearly show the superiority of the proposed hybrid model with SMOTE-ENN and ANN, which achieves an accuracy of 98.6%—significantly higher than all

compared models in the literature. For instance, the fuzzy clustering with backpropagation neural network (BPNN) used by Chou et al. (2017), reached an accuracy of 95.25%, while the AWOA-DL model by Elhoseny et al. (2022) attained 95.77%. These models, though effective, did not employ advanced feature selection or hybrid resampling techniques.

The highest average score achieved by the proposed SMOTE-ENN ANN model, as presented in Table 9, highlights its superior performance compared to previously studied models. This result reflects the robustness and reliability of the proposed model in accurately predicting bankruptcy. Achieving the highest score indicates that the combination of SMOTE-ENN for resampling, Binary PSO for feature selection, and ANN for classification contributes to a better balance between precision and recall, minimizing false positives and false negatives. This is particularly significant in the context of financial risk assessment, where accurate prediction is crucial for minimizing economic losses and guiding better decision-making for investors, analysts, and policymakers.

This performance highlights the practicality and reliability of the proposed model, setting a new standard for bankruptcy prediction. Investors can rely on the model to evaluate financial health and make better decisions

Table 9 The results of reference papers in the same dataset (Taiwan bankruptcy dataset).

Reference	Model	Average score
Liang et al. (2015)	The baseline RBF SVM	82.73%
	Wrapper methods GA + RBF SVM	81.91%
Liang et al. (2016)	The filter-based methods of Stepwise discriminant analysis (SDA) + SVM	81.50%
Chou et al. (2017)	Fuzzy clustering, BPNN	95.25%
Chen and Shen (2020)	LASSO-CART	89.74%
Brenes, Johannssen and Chukhrova (2022)	12 different MLPs	95.47% (highest) 81.69% (average)
Elhoseny et al. (2022)	AWOA-DL	95.77%
	Smote- ENN ANN (Proposed model)	**98.6%**

5.   Conclusions

This study proposes a machine learning approach for bankruptcy detection using the Taiwan Economic Journal data from 1999–2009, employing SVM, RF, and ANN for classification, along with feature selection, over- and undersampling methods, and parameter optimization. Initial baseline models were run on raw data, followed by Pearson's correlation analysis to eliminate highly correlated features and data standardization for consistency. The data imbalance was addressed using SMOTE-Tomek and SMOTE-ENN resampling techniques, and Binary PSO selected 34 features. Randomized Search fine-tuned hyperparameters, with the resampling technique significantly improving model performance. The ANN model, incorporating parameter tuning, Binary PSO, and SMOTE-ENN, achieved the highest results with a 98.5% F1 Score and 98.6% accuracy. This framework can be used as a scorecard for business credit risk evaluation, premium assessment in insurance, and to inform better credit and investment decisions. It supports

real-time compliance and strategic planning, enhancing predictive accuracy and reliability. The study addresses class imbalance and feature redundancy, demonstrating the efficiency and robustness of the proposed model, which can be adapted for different industries. The SMOTE-ENN ANN model offers valuable insights for investors, financial analysts, and policymakers by improving financial risk assessments and supporting data-driven decisions. Future research will expand parameter search ranges and explore other ML algorithms to further improve performance and applicability across various sectors.

Abbreviations

Abbreviation	Definition
SVM	Support Vector Machines
RF	Random Forest
ANN	Artificial Neural Network
SMOTE	Synthetic minority oversampling Technique
ENN	Edited Nearest Neighbor
BPSO	Binary Particle Swarm Optimization
ML	Machine Learning
FS	Feature Selection

Acknowledgment

References

Ali J, Khan R, Ahmad N, Maqsood I (2012) Random forests and decision trees. Int J Comput Sci Issues 9(5):272–278

Prasetiyani, E., & Sofyan, M. (2020). Bankruptcy Analysis Using Altman Z-Score Model and Springate Model In Retail Trading Company Listed In Indonesia Stock Exchange. Ilomata International Journal of Tax and Accounting, 1(3), 139-144. https://doi.org/10.52728/ijtc.v1i3.98

Najib, A. S., & Cahyaningdyah, D. (2020). Analysis of The Bankruptcy of Companies with Altman Model and Ohlson Model. Management Analysis Journal, 9(3), Article 3. https://doi.org/10.15294/maj.v9i3.37797

Huo, Y., Chan, L. H., & Miller, D. (2024). Bankruptcy Prediction for Restaurant Firms: A Comparative Analysis of Multiple Discriminant Analysis and Logistic Regression. Journal of Risk and Financial Management, 17(9), 399. https://doi.org/10.3390/jrfm17090399

Taiwanese Bankruptcy Prediction [Dataset]. (2020). UCI Machine Learning Repository. https://doi.org/10.24432/C5004D

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explor. Newsl., 6(1), 20–29. https://doi.org/10.1145/1007730.1007735

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explor. Newsl., 6(1), 20–29. https://doi.org/10.1145/1007730.1007735

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. Computers & Electrical Engineering, 40(1), 16–28.
https://doi.org/10.1016/j.compeleceng.2013.11.024

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

Chen, S., & Shen, Z.-D. (2020). Financial Distress Prediction Using Hybrid Machine Learning Techniques. Asian Journal of Economics, Business and Accounting, 1–12. https://doi.org/10.9734/ajeba/2020/v16i230231

Chou, C.-H., Hsieh, S.-C., & Qiu, C.-J. (2017). Hybrid genetic algorithm and fuzzy clustering for bankruptcy prediction. Applied Soft Computing, 56, 298–316. https://doi.org/10.1016/j.asoc.2017.03.014

Dao, S. V. T., Yu, Z., Tran, L. V., Phan, P. N. K., Huynh, T. T. M., & Le, T. M. (2022). An Analysis of Vocal Features for Parkinson's Disease Classification Using Evolutionary Algorithms. Diagnostics, 12(8), Article 8. https://doi.org/10.3390/diagnostics12081980

Elhoseny, M., Metawa, N., Sztano, G., & El-Hasnony, I. M. (2022). Deep Learning-Based Model for Financial Distress Prediction. Annals of Operations Research, 1–23. https://doi.org/10.1007/s10479-022-04766-5

Guyon, I. & Elisseeff, A. 2003. An introduction to variable and feature selection. Journal of machine learning research, Volume 3, pp. 1157-1182.

Ho, T. T. M., Tran, L. V., Tran, H. M. & Dao, S. V. T. 2022. Machine Learning in Demand Forecasting. International Research Journal of Advanced Engineering and Science.

Ahmad, F. A., Liu, J., Hashim, F., & Samsudin, K. (2024). Short-Term Load Forecasting Utilizing a Combination Model: A Brief Review. International Journal of Technology, 15(1), 121-129.

Huynh, T. T. M., Le, T. M., That, L. T., Tran, L. V., & Dao, S. V. T. (2022). A Two-Stage Feature Selection Approach for Fruit Recognition Using Camera Images With Various Machine Learning Classifiers. IEEE Access, 10, 132260–132270. IEEE Access. https://doi.org/10.1109/ACCESS.2022.3227712

Kelleher, J. D., Mac Namee, B. & D'arcy, A. 2020. Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies, MIT press.

Ismat, M. N., Chin Poo Lee, Heidi Yeen-Ju Tan, Tse Kian Neo, Yong Xuan Tan, Nazi Mahendru, Zahra. (n.d.). Enhancing Students' Online Learning Experiences with Artificial Intelligence (AI): The MERLIN Project. IJTech - International Journal of Technology. Retrieved December 23, 2024, from https://ijtech.eng.ui.ac.id/article/view/5843

Koziarski, M. (2021). CSMOUTE: Combined Synthetic Oversampling and Undersampling Technique for Imbalanced Data Classification (arXiv:2004.03409). arXiv. https://doi.org/10.48550/arXiv.2004.03409

Le, M. T., Thanh Vo, M., Mai, L., & Dao, S. V. T. (2020). Predicting heart failure using deep neural network. 2020 International Conference on Advanced Technologies for Communications (ATC), 221–225.
https://doi.org/10.1109/ATC50776.2020.9255445

Le, M. T., Vo, M. T., Pham, N. T., & Dao, S. V. T. (2021). Predicting heart failure using a wrapper-based feature selection. Indonesian Journal of Electrical Engineering and

Computer Science, 21(3), Article 3. https://doi.org/10.11591/ijeecs.v21.i3.pp1530-1539

Tuan, L., Pham, T., & Dao, S. (2021). A Novel Wrapper-Based Feature Selection for Heart Failure Prediction Using an Adaptive Particle Swarm Grey Wolf Optimization (pp. 315–336). https://doi.org/10.1007/978-3-030-70111-6_15

Le, T. M., Tran, L. V., & Dao, S. V. T. (2021). A Feature Selection Approach for Fall Detection Using Various Machine Learning Classifiers. IEEE Access, 9, 115895–115908. IEEE Access. https://doi.org/10.1109/ACCESS.2021.3105581

Le, T. M., Vo, T. M., Pham, T. N., & Dao, S. V. T. (2021). A Novel Wrapper–Based Feature Selection for Early Diabetes Prediction Enhanced With a Metaheuristic. IEEE Access, 9, 7869–7884. IEEE Access. https://doi.org/10.1109/ACCESS.2020.3047942

Liang, D., Lu, C.-C., Tsai, C.-F. & Shih, G.-A. 2016. Financial ratios and corporate governance indicators in bankruptcy prediction: A comprehensive study. European journal of operational research, Volume 252, pp. 561-572.

Liang, D., Tsai, C.-F., & Wu, H.-T. (2015). The effect of feature selection on financial distress prediction. Knowledge-Based Systems, 73, 289–297. https://doi.org/10.1016/j.knosys.2014.10.010

Mafarja, M., Jarrar, R., Ahmad, S., & Abusnaina, A. A. (2018). Feature selection using binary particle swarm optimization with time varying inertia weight strategies. Proceedings of the 2nd International Conference on Future Networks and Distributed Systems, 1–9. https://doi.org/10.1145/3231053.3231071

Minh, H. T., Mai, L., & Minh, T. V. (2021). Performance Evaluation of Deep Learning Models on Embedded Platform for Edge AI-Based Real time Traffic Tracking and Detecting Applications. 2021 15th International Conference on Advanced Computing and Applications (ACOMP), 128–135. https://doi.org/10.1109/ACOMP53746.2021.00024

J, O. F. Y., Akinsola J. E. T. ,. Awodele O. ,. Hinmikaiye J. O. ,. Olakanmi O. ,. Akinjobi. (n.d.). Supervised Machine Learning Algorithms: Classification and Comparison. Seventh Sense Research Group. Retrieved March 18, 2024, from https://dev.ijcttjournal.org//archives/ijctt-v48p126

Sekeroglu, B., Hasan, S. S., & Abdullah, S. M. (2020). Comparison of Machine Learning Algorithms for Classification Problems. In K. Arai & S. Kapoor (Eds.), Advances in Computer Vision (pp. 491–499). Springer International Publishing. https://doi.org/10.1007/978-3-030-17798-0_39

Shalev-Shwartz, S., & Ben-David, S. (2014). Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press. https://doi.org/10.1017/CBO9781107298019

Tang, J., Alelyani, S., & Liu, H. (2014). Feature selection for classification: A review. In Data Classification (pp. 37–64). CRC Press. https://doi.org/10.1201/b17320

Tobback, E., Bellotti, T., Moeyersoms, J., Stankova, M., & Martens, D. (2017). Bankruptcy prediction for SMEs using relational data. Decision Support Systems, 102, 69–81. https://doi.org/10.1016/j.dss.2017.07.004

Tran, H. M., Pham, K. T., Vo, T. M., Le, T.-H., Huynh, T. T. M., & Dao, S. V. T. (2023). A New Approach for Estimation of Physical Properties of Irregular Shape Fruit. IEEE Access, 11, 46550–46560. IEEE Access. https://doi.org/10.1109/ACCESS.2023.3273777

Tran, H. M., Pham, K. T., Vo, T. M., Tonthat, L., Huynh, T. T. M., & Dao, S. V. T. (2023). Physical Characteristics Estimation for Irregularly Shaped Fruit Using Two Cameras. 2023

IEEE   Statistical   Signal   Processing   Workshop   (SSP),   165–169. https://doi.org/10.1109/SSP53291.2023.10207992

Tran, L. V., Tran, H. M., Le, T. M., Huynh, T. T. M., Tran, H. T., & Dao, S. V. T. (2022). Application of Machine Learning in Epileptic Seizure Detection. Diagnostics, 12(11), Article 11. https://doi.org/10.3390/diagnostics12112879

Lomakin, N. L., Maxim Maramygin, Alexander Kataev, Sergey Kraschenko, Olga Yurova ,. Ivan. (n.d.). Cognitive Model of Financial Stability of the Domestic Economy Based on Artificial Intelligence in Conditions of Uncertainty and Risk. IJTech - International Journal   of   Technology.   Retrieved   December   23,   2024,   from https://ijtech.eng.ui.ac.id/article/view/6185

Verma, D. S. S., Upasna. Distinct Multiple Learner-Based Ensemble SMOTEBagging (ML-ESB) Method for Classification of Binary Class Imbalance Problems. IJTech - International Journal of Technology. Retrieved December 23, 2024, from https://ijtech.eng.ui.ac.id/article/view/1743

Vieira, S. M., Mendonça, L. F., Farinha, G. J., & Sousa, J. M. C. (2013). Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients. Applied   Soft   Computing,   13(8),   3494–3504. https://doi.org/10.1016/j.asoc.2013.03.021

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory   and   practice.   Neurocomputing,   415,   295–316. https://doi.org/10.1016/j.neucom.2020.07.061