Bankruptcy Prediction In Polish

A Thesis Presented to

Professor: Luong Van Thien

National Economics University

In Submission for Final Test of

Data Science in Economics and Bussiness

by

GROUP 11

November 28, 2024

Abstract

Bankruptcy prediction is essential in finance and accounting, helping stakeholders like creditors and investors assess financial risks. By identifying firms at risk of financial distress, these models enable corrective actions to prevent bankruptcy or reduce its impact. This study compares classification models, including Support Vector Machines (SVM), Neural Networks, Random Forests, and Logistic Regression, Decision Tree using data from Polish companies. Predictive techniques range from traditional statistical methods to advanced machine learning approaches, with recent models offering significant improvements in accuracy.

The methodology involves preprocessing data to handle missing values using techniques like k-Nearest Neighbors (k-NN), Expectation-Maximization (EM), and Multivariate Imputation by Chained Equations (MICE). To address data imbalance, the Synthetic Minority Oversampling Technique (SMOTE) is applied, ensuring better representation of bankrupt firms. Models are evaluated using K-Fold Cross Validation and performance metrics such as accuracy, precision, recall, and F1-score.

Results show that advanced models like SVM and Neural Networks outperform traditional methods, offering higher prediction accuracy. However, they require more computational resources and careful tuning. Future improvements could focus on feature engineering or ensemble techniques to enhance robustness. Incorporating macroeconomic factors may also provide deeper insights into financial distress and improve predictive power.

Acknowledge

We wish to express our deepest gratitude to our esteemed professor, Luong Van Thien, of the National Economics University, for his unwavering guidance and support throughout the course of this thesis. As a dedicated and inspiring educator in the field of machine learning, his mentorship has been invaluable to us, both academically and personally. Under his direction, we gained not only technical knowledge but also the confidence and skills needed to navigate the complexities of research. His insights, patience, and encouragement played a crucial role in shaping the direction of our work and ensuring its successful completion.

Although our journey was not without challenges, particularly when certain aspects of our reports didn't proceed as planned, every session we attended with Professor Thien provided us with fresh perspectives and renewed motivation. His thoughtful feedback and constructive criticism guided us to refine our approach and achieve a deeper understanding of the subject matter.

We are also immensely grateful to those who actively engaged with our progress reports, offering valuable comments and suggestions. Their input greatly enriched our research. Lastly, we extend our heartfelt thanks to everyone who has taken the time to read this thesis. Your interest and support mean the world to us, and we hope this work reflects the collective effort and dedication that made it possible.

Contents

1. Introduction

1.1 Overview

Bankruptcy prediction plays a crucial role in financial risk management by enabling stakeholders to proactively identify potential insolvency and take preventive measures. Accurately predicting bankruptcy can protect investors, lenders, and organizations from severe financial losses. This research focuses on leveraging machine learning techniques to develop robust models that classify entities as either bankrupt or non-bankrupt.

1.2 Research Objectives

Traditional approaches, such as Logistic Regression and Decision Trees, are compared with advanced methods, including Support Vector Machines (SVMs), Neural Network and Random Forest models. Additionally, this study addresses challenges in data quality such as missing data and data imbalance to enhance model reliability, evaluates the predictive performance of these advanced methods, and emphasizes how improved predictions can guide better decision-making, minimize risks, and contribute to the overall stability of financial systems. By incorporating diverse models and addressing key data challenges, this research aims to improve the efficiency of bankruptcy prediction systems and foster a more accurate and transparent framework for financial analysis and risk assessment.

2. Background

2.1 Significance of Bankruptcy Prediction

Bankruptcy prediction holds immense importance within financial systems, significantly influencing various stakeholders such as investors, creditors, regulators, and management teams. Early identification of potential bankruptcies helps mitigate financial risks and allows stakeholders to implement timely corrective measures. For investors, it provides crucial insights to protect their portfolios from high-risk investments. Creditors benefit from assessing a firm's creditworthiness, minimizing the risk of loan defaults. Regulators rely on such predictions to maintain economic stability and prevent cascading financial crises. Additionally, predicting bankruptcy enables companies' management to proactively address financial challenges, restructure debts, and explore turnaround strategies. Overall, accurate bankruptcy prediction fosters a more resilient financial ecosystem by reducing uncertainties and facilitating better decision-making for all involved parties.

2.2 Machine Learning in Bankruptcy Prediction

Traditional methods like logistic regression, decision trees, and genetic algorithms have been extensively used in bankruptcy prediction. Logistic regression provides interpretable results but assumes linear relationships, limiting its effectiveness for complex, non-linear financial patterns. Decision trees, while intuitive, often suffer from overfitting and may not generalize well to unseen data, particularly when the dataset is imbalanced. Genetic algorithms optimize prediction models through iterative evolution but can be computationally intensive, challenging to tune effectively, and may struggle with noisy data. Naive Bayes, a probabilistic classifier, is computationally efficient and works well with smaller datasets but relies on the assumption of feature independence, which may not hold in financial data. In contrast, machine learning techniques, such as support vector machines, neural networks, and ensemble methods like Random Forests, address these limitations by leveraging advanced algorithms to capture intricate data patterns and interactions between variables. These models enhance predictive accuracy, handle large datasets efficiently, and provide more reliable outcomes across diverse scenarios. However, they require significant computational resources, careful hyperparameter tuning, and extensive data preprocessing efforts. Despite these challenges, machine learning methods have revolutionized bankruptcy prediction, offering robust tools for analyzing complex financial datasets and enabling deeper insights into financial distress.

3. Methodology

In the previous part, we emphasized the importance of bankruptcy prediction. Here, we detail the step-by-step methodology used to achieve benchmark results in this domain. We begin by introducing the Polish bankruptcy dataset, providing an overview of its structure, including features, instances, and data organization. Following this, we describe the data preprocessing phase, highlighting challenges such as missing values and class imbalance, and elaborate on the strategies implemented to address these issues. Subsequently, we present the classification models utilized in our study, discussing the training process applied to the dataset. Finally, we evaluate and compare the models' performance using metrics such as accuracy, AUC, and other metrics.

3.1 Data

Firstly, come to our collected data: Polish Bankruptcy Data

The dataset we have considered for addressing the bankruptcy prediction problem is the Polish bankruptcy data, hosted by the University of California Irvine (UCI) Machine Learning Repository—a huge repository of freely accessible datasets for research and learning purposes intended for the Machine Learning/Data Science community. The dataset is about bankruptcy prediction of Polish companies. The data was collected from Emerging Markets Information Service (EMIS), which is a database containing information on emerging markets around the world. The bankrupt companies were analyzed in the period 2000-2012, while the still operating companies were evaluated from 2007 to 2013

Data	Total Instances	Bankrupt Instances	Non-bankrupt instances	Number of features
1st year	7027	271	6756	64
2nd year	10173	400	9773	64
3rd year	10503	495	10008	64
4th year	9792	515	9227	64
5th year	5910	410	5500	64

Table 3.1: Polish Data Description

3.2 Data Quality

3.2.1 Missing Data

Firstly, we will check its correlation among the missing data This involves analyzing the relationship between the null values in different features of the dataset. By calculating the nullity correlation, we can identify whether the missing data in one feature is associated with missing data in another feature. A positive correlation suggests that the two features tend to have missing values simultaneously, while a negative correlation indicates that the presence of missing data in one feature is often paired with the absence of missing data in the other. Understanding these correlations is crucial for handling missing values effectively, as it can reveal patterns that may inform the imputation process or suggest dependencies between variables that need to be addressed before further analysis. This step is an important part of the data cleaning process, as it helps ensure that the missing data does not introduce bias or distort the results of any models we build.

Base on this, we can see that **Figure 1** presents a correlation heatmap for the 1st Year data, illustrating the degree of relationship between missing values across various features. The nullity correlation ranges from -1 to 1, where -1 indicates a perfect negative correlation (one feature has missing values while the other does not) and 1 represents a perfect positive correlation (both features have corresponding missing values). Features without any missing values are excluded to focus solely on those with nullity relationships. Values close to zero ($-0.05 < R < 0.05$), indicating negligible correlation, are omitted from the display for clarity. This visualization effectively highlights patterns in missing data, enabling better understanding of the dataset's structure.

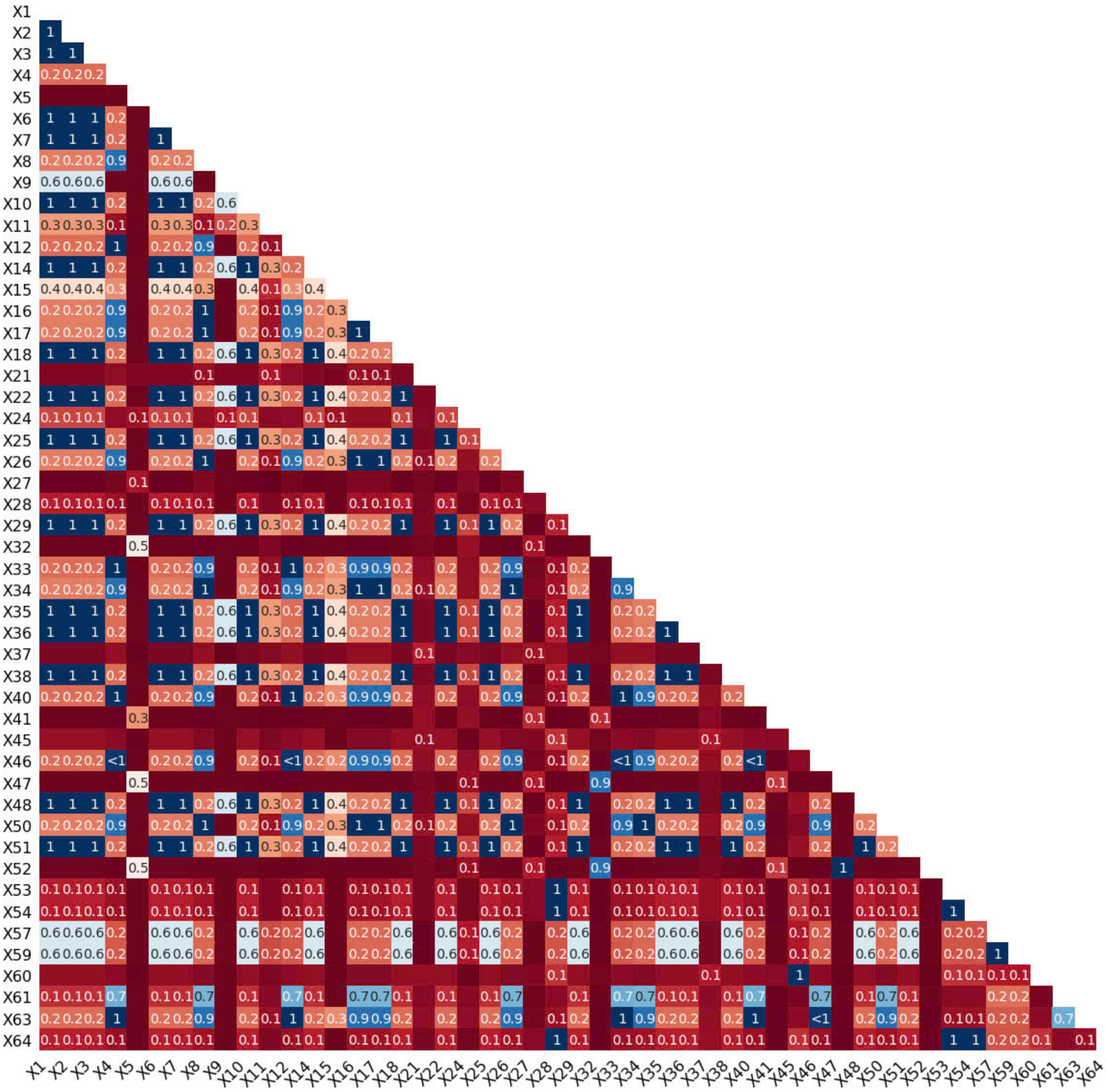

Figure 1: Features correlation for heatmap in 1st year dataset

Next, we can draw a table to see the problem of missing data is more clearly, and throughout we can find the way to resolve. The problem is as you can see the table below:

Data	Total Instance	% Data loss in rows
year 1	7027	% 54.54
year 2	10173	% 59.81
year 3	10503	% 53.48
year 4	9792	% 51.29
year 5	5910	% 48.71

Table 3.2: Assessing missing data for datasets

Column 3 presents the percentage of data that would be lost if all rows containing missing values were removed. Given that the data loss exceeds 50% in most cases, it is evident that simply discarding rows with missing values is not a viable option. Such an approach would result in a significant reduction in the dataset's representativeness, potentially compromising the quality and accuracy of any subsequent analysis. Therefore, alternative strategies for handling missing data must be considered to preserve the integrity of the dataset.

3.2.2 Data Imbalance

Having addressed the issue of missing data in the data quality assessment, we now turn our attention to the aspect of data imbalance. Table 3.1 below provides a summary of the class label distributions across each dataset. Column 2 presents the total number of instances, while Columns 3 and 4 detail the counts of instances labeled as ”Bankrupt” and ”Non-Bankrupt,” respectively. Upon examining the data, it is evident that the ”Bankrupt” class represents a minority compared to the ”Non-Bankrupt” class. This imbalance in the class distribution is a critical factor to consider, as it can lead to biased models that favor the majority class, potentially overlooking important patterns associated with the minority class. Addressing data imbalance is crucial for ensuring that the model can accurately detect instances of both classes, and various techniques, such as oversampling, undersampling, or using specialized algorithms, may be employed to mitigate the effects of this imbalance and improve model performance.

3.3 Dealing with Data

3.3.1 Feature Enginnering

Step 1: Solving missing data by KNN Imputation

The k-nearest neighbors (k-NN) algorithm is a flexible, non-parametric method widely used for classification and regression. It works by identifying the k nearest training examples in the feature space to make predictions. Additionally, k-NN is effective for data imputation, where missing values (NaNs) are replaced with values from the closest row or column based on Euclidean distance. If the nearest-neighbor value is also missing, the algorithm moves to the next closest neighbor to ensure accuracy.

In this process, we employed the **KNNImputer** class from the **sklearn.impute** module, specifying 100 nearest neighbors for imputing missing data. This technique effectively fills in gaps in the dataset while maintaining its overall structure and consistency. By borrowing information from similar data points, k-NN imputation minimizes the impact of missing values and ensures that the dataset remains suitable for further analysis or modeling tasks.

Step 2: Standardize the data

Standardizing data is crucial for working with the Polish Bankruptcy Prediction Dataset, where feature ranges often vary significantly (e.g., financial ratios like liquidity and profitability). It ensures that models sensitive to feature magnitudes, such as Logistic Regression, k-Nearest Neighbors (k-NN), and Support Vector Machines (SVM), treat all features equally, improving fairness and accuracy. Standardization also enhances numerical stability and accelerates convergence in algorithms like Random Forest and Decision Trees, which is vital given the skewed distributions common in financial data.

Additionally, standardizing before imputing missing values ensures meaningful distance calculations, enabling accurate and reliable imputations using methods like k-NN Imputer. This preserves the dataset's integrity and ensures consistency. For datasets with class imbalance, standardization helps models differentiate between minority and majority classes by preventing dominant feature scales from overshadowing minority signals. When combined with oversampling techniques like SMOTE or class weighting, it supports better decision-making and robust model performance.

Here is the standardized value formula:

$$z = \frac{x - \mu}{\sigma}$$

where:

- z is the standardized value,
- x is the original feature value,
- μ is the mean of the feature, and
- σ is the standard deviation of the feature.

Step 3: Solving data imbalance by SMOTE

The Synthetic Minority Oversampling Technique (SMOTE) is an effective method for oversampling in imbalanced datasets. Suppose we have training data with s samples and f features, such as a dataset of birds with features like beak length, wingspan, and weight. To apply SMOTE, we select a sample from the minority class and identify its k nearest neighbors. Then, to generate a synthetic sample, we calculate the vector between the current sample and one of its neighbors, scale it by a random factor between 0 and 1, and add this to the original sample to create a new data point. SMOTE is implemented using the **imbalanced-learn** library.

3.3.2 Feature selection methods with SelectBest

Feature selection is the process of identifying and selecting the most relevant features from a dataset to improve the performance of machine learning models. One commonly used method is SelectKBest, which selects the k best features based on their relationship with the target variable, determined by statistical tests such as the chi-square test, mutual information, or ANOVA F-test. For example, with **SelectKBest**, you can choose the top 10 features that have the strongest connection to the target. The advantages of feature selection include improved model efficiency, as fewer features lead to faster training times, and reduced risk of overfitting by eliminating irrelevant features. It also enhances model interpretability, making it easier to understand which features are most important.

However, feature selection has some drawbacks, such as the risk of excluding potentially valuable features that may not show up strongly in statistical tests. Additionally, it may require domain knowledge to choose the right statistical test and the optimal number of features to retain.

3.3.3 Training Data and Validation

We divided the dataset into two parts: 80% for training the models and 20% for testing. The training data is used to teach the models by allowing them to learn patterns and relationships within the data. Once trained, the models are evaluated using the test data, which provides an unbiased measure of how well the model generalizes to new, unseen examples. This split ensures that the model's performance is not just based on memorization of the training data but reflects its ability to make accurate predictions on real-world, unseen data, providing a reliable estimate of its effectiveness.

4. Model Description

Model we will use including Support Vector Machines (SVM),Deep Neural Networks, Random Forests, Logistic Regression, and Decision Tree.

4.1 Logistic Regression

Logistic regression is a linear model for classification, often referred to as logit regression, maximum-entropy classification (MaxEnt), or the log-linear classifier. This model models the probabilities of different outcomes of a single trial using a logistic function. The LogitClassifier for a given input x is calculated as follows:

$$\text{LogitClassifier}(x) = \min_{w,c} \left(\|w\|_1 + C \sum_{i=1}^{n} \log \left(\exp \left(-y_i \left(X_i^T w + c \right) \right) + 1 \right) \right)$$

Incorporating an L1 penalty (Lasso) into the model, the LogitClassifier value for data points x_i is adjusted as follows:

$$\text{LogitClassifier}(x) = \text{LogitClassifier}^*(x) - \lambda \|\theta\|_1$$

where

$$\|\theta\|_1 = \sum_{i=1}^{n} |\theta_i|$$

In this implementation, $\lambda = 1$ is used, and equal weights are assigned to all features, employing L1 regularization.

4.2 Decision Trees

Decision Trees (DTs) are non-parametric supervised learning algorithms used for both classification and regression tasks. In our classification task, the model predicts the target variable (e.g., whether a firm will go bankrupt) by learning simple decision rules derived from the dataset's features (such as financial distress variables of a firm, denoted as $x_1, x_2, \ldots, x_{64}$). The dataset is structured as records in the form:

$$(x, y) = (x_1, x_2, \ldots, x_{64}, y)$$

The decision tree model evaluates all features, assigning equal importance to each one when determining the best split.

4.3 Random Forests

A random forest is an ensemble method that builds multiple decision tree classifiers on different random subsets of the data, combining their predictions through averaging to enhance predictive accuracy and reduce overfitting. In this approach, each decision tree is constructed from a bootstrap sample, meaning a sample drawn with replacement from the training data. During the tree-building process, instead of considering all features for a split, only a random subset of features is evaluated to determine the best split. This randomness results in slightly higher bias for the forest compared to individual decision trees, but the variance is significantly reduced due to the averaging of predictions from multiple trees.

4.4 Support Vector Machine

Support Vector Machines (SVM) are powerful supervised learning algorithms used for classification tasks, especially in high-dimensional spaces. When applied to the Polish Bankruptcy Prediction Dataset, SVM can be particularly useful due to its ability to handle complex decision boundaries, making it effective for predicting whether a firm will go bankrupt based on its financial ratios.

SVM works by finding a hyperplane that best separates the data into two classes—in this case, bankrupt and non-bankrupt firms. The algorithm aims to maximize the margin between the classes, which leads to better generalization on unseen data. In the context of the Polish Bankruptcy Dataset, where financial data may have nonlinear relationships, SVM's kernel trick allows it to transform the data into a higher-dimensional space, making it easier to find a separating hyperplane.

4.5 Neutral Netwrorks

Neural Networks (NN) are a class of machine learning models capable of capturing complex, non-linear relationships between input features, making them suitable for the Polish Bankruptcy Prediction Dataset. In this context, NNs can learn intricate patterns within financial data, such as liquidity and profitability ratios, to predict the likelihood of bankruptcy. By using multiple layers of neurons, neural networks can automatically extract high-level features and model complex decision boundaries that traditional linear models may struggle with.

The advantages of using neural networks for bankruptcy prediction include their ability to handle large, high-dimensional datasets and detect subtle patterns in the data. They are particularly effective when the relationships between financial variables are non-linear and complex. Furthermore, neural networks are flexible and can be fine-tuned to optimize performance. With proper regularization, they can also avoid overfitting, making them a robust choice for accurately predicting bankruptcy in the Polish dataset.

5. Code Step

5.1 Exploratory Data Analysis

We first need to import the libraries needed for the project. Then, we do some visualization to visualize the data, therefore gain many insights on the current data characteristics. When checking for missing values, the dataset shocked us by showing vast amount of nulls in it, if we drop rows that contain at least one null, we could end up drop nearly half of the data we have. By visualizing, we also discover how severely imbalanced the data is, the number of observations that match our positive target feature is so small compared to those of negative target feature. The last thing to tell here is that since we are not familiar with the '.arff' file format, so we change the data files into '.csv' files, not something big but it is still in the process.

5.2 Data pre-processing

Now we start to transform our data to resolve many of its problems that could affect badly the prediction ability of our models. The problems are (as mentioned above): huge amount of missing values, data imbalance, not normalized data. Based on what we have learned, KNNImputer is used to impute the missing values, SMOTE (Synthetic-Minority-Oversampling-Technique) is used to deal with the severe imbalance by creating many more '1' observations. StandardScaler helps to standardize the dataset. After all, to select 10 appropriate features for building models and predicting, we use SelectBest, an algorithm that use ANOVA F-value between label/feature for classification tasks. During the process, the data has been split into train and test sets, the test size is 20% of the general data. We split the data first, then we imputed missing values and rescaled both X_train and X_test, then we use SMOTE on the train test, and finally selected 10 features from 64 independent features to build models and predict.

That is how the data pre-processing step was done. We have used what are common to us when it comes to these kinds of data quality issues.

5.3 Building Models

To predict the bankruptcy of Polish companies in the dataset, we built in total 5 models. They are respectively: Logistic Regression, Decision Tree, Random Forest, Neural Network and Support Vector Machine. These models are popular and for the most reason are what we have studied in the Machine Learning 1 course. All 5 models is implemented on each data files so that we can compare how effective they are on each of them. This is because the data files are different in the number of years in the data collecting period. The parameters of models are not from the calculation, they are just some ordinary and very basic settings of parameters in the models.

5.4 Metrics used for evaluating

AUC (Area Under The Curve) score and classification report were used to evaluate our models' performances. Between them, we focus more on the AUC score as our data is heavily imbalanced. AUC score measure the overall performance of the binary classification model. As both TPR(True Positive Rate) and FPR(False Positve Rate) range between 0 to 1, the area will always lie between 0 and 1, and a greater value of AUC denotes better model performance. Our main goal is to maximize this area in order to have the highest TPR and lowest FPR at the given threshold. The AUC measures the probability that the model will assign a randomly chosen positive instance a higher predicted probability compared to a randomly chosen negative instance.

6. Results

6.1 Performance Overview

This report evaluates the performance of several machine learning models applied to a heavily imbalanced dataset. The models were assessed using two metrics: Area Under the Curve (AUC) and Accuracy. Given the dataset's imbalance, AUC is emphasized as it provides a threshold-independent measure of model discrimination. Accuracy, though informative, may be less reliable due to its sensitivity to class distribution.

Neural Networks

Neural Networks showed significant variation in performance. Neural Network year 5 achieved an AUC of 0.89 and accuracy of 0.72, indicating strong discriminatory ability and good overall classification.Neural Network year 4 displayed a competitive AUC of 0.75 with moderate accuracy (0.62), while Neural Network year 3 had the highest accuracy (0.71) among the lower AUC models. Neural Networks year 1 and year 2 performed poorly, with AUC values of 0.69 and 0.66 and accuracies of 0.43 and 0.62, respectively.

Support Vector Machine (SVM)

Support Vector Machines demonstrated strong performance, particularly in later iterations. SVM year 5 achieved an AUC of 0.88 and accuracy of 0.78, closely followed by SVM year 4 with an AUC of 0.75 and accuracy of 0.72. On the other hand SVM year 2 and SVM year 3 delivered moderate results with AUC values between 0.65 and 0.69 and accuracies of 0.57 and 0.55, respectively. SVM year 1 performed comparably to Neural Network 1 (AUC 0.68, accuracy 0.43).

Logistic Regression

Model	Year	AUC	Accuracy
Neural Network	1	0.69	0.43
Neural Network	2	0.66	0.62
Neural Network	3	0.57	0.71
Neural Network	4	0.75	0.62
Neural Network	5	0.89	0.72
Support Vector Machine	1	0.68	0.43
Support Vector Machine	2	0.65	0.57
Support Vector Machine	3	0.69	0.55
Support Vector Machine	4	0.75	0.72
Support Vector Machine	5	0.88	0.78
Logistic Regression	1	0.63	0.6
Logistic Regression	2	0.5	0.51
Logistic Regression	3	0.76	0.5
Logistic Regression	4	0.75	0.7
Logistic Regression	5	0.88	0.78
Random Forest	1	0.7	0.92
Random Forest	2	0.43	0.94
Random Forest	3	0.65	0.85
Random Forest	4	0.73	0.9
Random Forest	5	0.8	0.86
Decision Tree	1	0.57	0.53
Decision Tree	2	0.56	0.67
Decision Tree	3	0.58	0.61
Decision Tree	4	0.57	0.86
Decision Tree	5	0.66	0.77

Table 6.1: Results of models on each year data file

Logistic Regression models exhibited a wide range of performance. Logistic Regression year 5 matched SVM year 5, achieving an AUC of 0.88 and accuracy of 0.78. Logistic Regression year 3 demonstrated high AUC (0.76) but had a low accuracy of 0.5, indicating a misalignment in threshold optimization. Logistic Regression year 2 performed poorly, with an AUC of 0.5, equivalent to random guessing.

Random Forests

Random Forest models excelled in accuracy but showed moderate AUC values. Random Forest year 5 achieved an AUC of 0.8 and accuracy of 0.86, closely followed by Random Forest year 4 with an AUC of 0.73 and accuracy of 0.9. Random Forest year 2 delivered the highest accuracy (0.94) but had the lowest AUC (0.43), indicating potential overfitting to the majority class.

Decision Tree

Decision Trees were the least consistent among the evaluated models. Decision Tree year 5 achieved an AUC of 0.66 and accuracy of 0.77, offering the best balance among Decision Tree models. Decision Tree year 4 exhibited a high accuracy (0.86) but a poor AUC (0.57), reflecting overfitting tendencies. Decision Trees year 1 to 3 performed poorly in both AUC (range: 0.56–0.58) and accuracy (range: 0.53–0.67).

6.2 Our comment

The heavily imbalanced dataset significantly influenced model performance, often inflating accuracy for models biased toward the majority class. AUC proved to be a more reliable measure of discriminatory ability, revealing performance nuances that accuracy alone could not capture. Random Forest and SVM models emerged as the most robust, consistently balancing AUC and accuracy, while Logistic Regression showed strong potential when optimally tuned. Neural Networks exhibited variability, with only Neural Network year 5 achieving high performance, while Decision Trees generally struggled to handle the imbalance effectively. The divergence between AUC and accuracy across models underscores the importance of using multiple evaluation metrics, particularly in the context of imbalanced datasets.

6.3 Conclusion and Discussion

6.3.1 Main Findings

This study evaluated the performance of multiple machine learning models on a heavily imbalanced dataset, focusing on AUC and accuracy as key metrics. The results revealed that Random Forest and Support Vector Machine (SVM) models consistently outperformed other approaches, achieving the best balance between AUC and accuracy. Logistic Regression demonstrated potential when hyperparameters were optimized, with Logistic Regression year 5 achieving results comparable to the best-performing models. Neural Networks showed significant variability, with only Neural Network year 5 achieving strong results, while Decision Trees struggled overall, with occasional strong accuracy but weak AUC.

Random Forest models were particularly notable for achieving high accuracy but displayed moderate AUC scores, indicating potential overfitting to the majority class. Conversely, some models, such as Logistic Regression year 3, had high AUC but suboptimal accuracy, suggesting a failure to optimize thresholds effectively.

6.3.2 Discussion

The imbalance in the dataset posed significant challenges for classification, favoring metrics like AUC that are less affected by class distribution. Accuracy, while intuitive, was often misleading, as it could be inflated by the models' tendency to favor the majority class. This is evident in models such as Random Forest 2 and Decision Tree 4, where high accuracy coincided with low AUC.

The strong performance of Random Forest and SVM highlights their robustness in handling complex patterns even in imbalanced datasets. Random Forest's ensemble nature allowed it to capture nuanced relationships, while SVM's kernel methods contributed to effective decision boundaries. Neural Networks, while promising in theory, require careful hyperparameter tuning and architecture design, as evidenced by the variability in results. Logistic Regression, though simple, demonstrated competitive performance when tuned effectively. Decision Trees struggled due to their propensity to overfit the majority class, leading to lower discriminatory power.

6.3.3 Limitations

Imbalanced Dataset: The class imbalance heavily influenced results, challenging models to perform well on minority classes. While AUC offered some robustness, additional metrics such as precision, recall, and F1-score would provide a more holistic evaluation.

Hyperparameter Tuning: Some models may not have been optimally tuned, potentially limiting their performance. More systematic tuning could yield better results, particularly for Neural Networks and Logistic Regression.

Dataset-Specific Observations: The findings may not generalize to datasets with different characteristics or imbalance ratios, as the models' performance heavily depends on the data distribution.

Evaluation Metrics: While AUC and accuracy were insightful, a lack of precision-recall or cost-sensitive metrics means the study might underestimate models' true potential in minority class prediction.

6.3.4 Conclusion

The study underscores the importance of selecting appropriate models and evaluation metrics when dealing with imbalanced datasets. Random Forest and SVM emerged as the most reliable models, offering a good balance between discrimination and accuracy. Logistic Regression and Neural Networks showed potential but require careful configuration to achieve competitive results, while Decision Trees were the least effective. To improve performance on imbalanced datasets, future work should consider implementing class-balancing techniques, expanding the range of evaluation metrics, and optimizing model hyperparameters. These steps would enhance both the interpretability and robustness of machine learning solutions in similar contexts.

References

[1] Fijorek, Kamil, and Michal Grotowski. "Bankruptcy prediction: some results from a large sample of Polish companies." International Business Research 5.9 (2012): 70.

[2] Zahiri, Parisa. Bankruptcy Prediction by Deep Learning and Machine Learning Methods. Diss. Concordia University, 2022.

[3] PASTERNAK-MALICKA, Monika; OSTROWSKA-DANKIEWICZ, Anna; DANKIEWICZ, Robert. Bankruptcy-an assessment of the phenomenon in the small and medium-sized enterprise sector-case of Poland. Polish journal of management studies, 2021, 24.1: 250-267.

[4] JOB, Natalia; STABRYŁA-TATKO, Kacper. DISCRIMINANT MODELS IN PREDICTING BANKRUPTCY OF POLISH COMPANIES.

[5] MUĆKO, Przemysław; ADAMCZYK, Adam. Does the bankrupt cheat? Impact of accounting manipulations on the effectiveness of a bankruptcy prediction. PloS one, 2023, 18.1: e0280384.

[6] KITOWSKI, Jerzy; KOWAL-PAWUL, Anna; LICHOTA, Wojciech. Identifying symptoms of bankruptcy risk based on bankruptcy prediction models—A case study of Poland. Sustainability, 2022, 14.3: 1416.

[7] DANKIEWICZ, Robert; SZYMAŃSKA, Anna. BANKRUPTCY IN POLISH CONDITIONS-AN ANALYSIS OF THE SCALE OF THE PHENOMENON OVER TIME. Journal of Security Sustainability Issues, 2020, 10.2.

[8] PILCH, Bartłomiej. An analysis of the effectiveness of bankruptcy prediction models–an industry approach. Folia Oeconomica Stetinensia, 2021, 21.2: 76-96.

[9] PTAK-CHMIELEWSKA, Aneta. Bankruptcy prediction of small-and medium-sized enterprises in Poland based on the LDA and SVM methods. Statistics in Transition. New Series, 2021, 22.1: 179-195.

[10] SEN, Prasenjit, et al. Evaluating Machine Learning and Deep Learning Analytics for Predicting Bankruptcy of Companies. In: International Conference on Mechatronics and Intelligent Robotics. Singapore: Springer Nature Singapore, 2023. p. 407-419.

[11] LIASHENKO, Olena; KRAVETS, Tetyana; KOSTOVETSKYI, Yevhenii. Machine learning and data balancing methods for bankruptcy prediction. Ekonomika, 2023, 102.2: 28-46.

[12] NGUYEN, Hoang Hiep; VIVIANI, Jean-Laurent; BEN JABEUR, Sami. Bankruptcy prediction using machine learning and Shapley additive explanations. Review of Quantitative Finance and Accounting, 2023, 1-42.

[13] HAMDI, Manel; MESTIRI, Sami; ARBI, Adnène. Artificial Intelligence Techniques for Bankruptcy Prediction of Tunisian Companies: An Application of Machine Learning and Deep Learning-Based Models. Journal of Risk and Financial Management, 2024, 17.4: 132.

[14] SUN, Caiwei. Company Bankruptcy Prediction with Machine Learning Techniques. In: Advances in Artificial Intelligence, Big Data and Algorithms. IOS Press, 2023. p. 425-437.

[15] ZHAO, Jinxian; OUENNICHE, Jamal; DE SMEDT, Johannes. A complex network analysis approach to bankruptcy prediction using company relational information-based drivers. Knowledge-Based Systems, 2024, 300: 112234.

[16] MÁTÉ, Domicián; RAZA, Hassan; AHMAD, Ishtiaq. Comparative Analysis of Machine Learning Models for Bankruptcy Prediction in the Context of Pakistani Companies. Risks, 2023, 11.10: 176.

[17] DA SILVA MATTOS, Eduardo; SHASHA, Dennis. Bankruptcy prediction with low-quality financial information. Expert Systems with Applications, 2024, 237: 121418.

[18] MUSLIM, Much Aziz, et al. An ensemble stacking algorithm to improve model accuracy in bankruptcy prediction. Journal of Data Science and Intelligent Systems, 2024, 2.2: 79-86.

[19] NOH, Seol-Hyun. Comparing the performance of corporate bankruptcy prediction models based on imbalanced financial data. Sustainability, 2023, 15.6: 4794.

[20] HORVATHOVA, Jarmila; MOKRISOVA, Martina. Overview of business bankruptcy models. Economic and Social Development: Book of Proceedings, 2023, 257-273.