Improving Bankruptcy Prediction Using Oversampling and Feature Selection Techniques

Duaa Alrasheed[1], Dongsheng Che[1]
[1]Department of Computer Science, East Stroudsburg University, East Stroudsburg, PA 18301, USA

Abstract - *There is a continuous interest in finding better methods to predict bankruptcy because many financial decisions can be made based on the result of such methods. The machine learning techniques are increasingly being developed to improve the prediction of bankruptcy. However, because bankruptcy events are relatively few, machine learning techniques accuracy have been unsatisfactory. In this study, we examine the effectiveness of using oversampling to improve the performance of learning algorithms in imbalanced data. We found that oversampling does improve the precision of the learning algorithms. In addition, the study identified the most important attributes that highly correlate to bankruptcy. Lastly, we tested six major ML algorithms for predicting bankruptcy which are Neural Networks, Decision Trees, Random Forests, Support Vector Machine, K-Nearest Neighbor and Logistic Regression. Random Forest, Decision Tree, and KNN were found to be the best techniques for such problem as they produced higher prediction accuracy.*

Keywords: Bankruptcy Prediction, Feature Selection, Machine Learning Techniques, Oversampling**.**

1 INTRODUCTION

Machine learning is increasingly being applied in many fields. In the financial field, machine learning has received a wide application, especially in the analysis of data for financial decision making. Bankruptcy is a legal situation in which a person, organization or government has failed to pay its debts [1]. In case a company, for instance, is declared bankrupt, the financial assets of the company are sold out to repay the debt, resulting in severe financial consequences to investors. Bankruptcy filings, as Wagenmans [2] observed, are mainly caused by financial decision makers, as well as the financial environments in which a company operates. The severe fiscal impact of bankruptcy evidently calls for bankruptcy prediction in order to prevent future occurrences. Because bankruptcy prediction influences decision making made by financial players, erroneous prediction can severely lead to financial crisis. For this reason, researchers are increasingly exploring different methods to improve the accuracy of bankruptcy prediction.

Most recently, machine learning techniques in bankruptcy prediction have attracted the attention of researchers. Traditionally, statistical techniques have been widely used; however, with the emergence of machine learning techniques, researchers are studying the effectiveness of machine learning techniques over the traditional methods. For instance, a study done by Barboza, Kimura, and Altman [3] examined the effectiveness of machine learning models, boosting, support vector machine and random forest in predicting bankruptcy. Similarly, Kostopoulos, Karlos, Kotsiantis, and Tampakas [4] tested active learning methods in predicting bankruptcy. Even though several studies have explored machine learning techniques in bankruptcy prediction, there is still little evidence on the most effective method, based on comparative analysis.

In this paper, we focus on two main things related to bankruptcy. First, we identify the most important features that are highly correlated with bankruptcy using a dataset of Polish companies. Second, we apply six different classification machine learning algorithms, which are Neural Networks, Decision trees, Random Forests, Support Vector Machine, K-Nearest Neighbor and Logistic Regression, to build models that predict bankruptcy 1, 2, 3, 4 and 5 years ahead. Those algorithms are selected because the features in the dataset are continuous. Our study indicates that the key features differ from one year to another while some features, such as x_{40}: (current assets - inventory) / short-term liabilities, are highly correlated to bankruptcy in all the years. We also examine the effect of balancing the data on the performance of the algorithms.

The remainder of this report is organized as follows. Section 2 provides information about previous and related work that has been done in this field. Section 3 explains our approach and method to address the problem. Section 4 discusses the results Finally, we will conclude with the findings and challenges that we faced in this study.

2 RELATED WORK

Bankruptcy prediction based on ML techniques has been examined in various research studies. Nagaraj and Sridhar [1] conducted a study that proposed bankruptcy prediction system for analyzing companies that have a high risk of bankruptcy. The study first measured the performance of 5 ML classification algorithms: neural network, logistic regression, support vector machines, random forest and Bayesian method. The study used a test dataset to evaluate the performance of the output of each model. Similarly, the researcher used a strategy called ten-fold cross-validation to measure test accuracy [1]. And finally, the ML algorithm that

had the highest accuracy was selected to develop the prediction system.

Another study was conducted by Pompe and Feelders [5] to determine whether machine learning algorithms outperform the statistical methods. The study, specifically, conducted a comparative study between the statistical approach, in this case linear discriminant analysis, and the artificial intelligence approach, in this case neural network and decision trees [5]. The data was collected from the financial reports of Belgian construction companies. Overfitting was minimized using stratified tenfold cross-validation. The study found that, contrary to many research findings, there was no significant difference between the two approaches.

A study was conducted to evaluate the effectiveness of the various approaches to active learning in predicting corporate bankruptcy [4]. The study used financial data from Greek companies. According to the findings, the active learning method was highly effective in predicting bankruptcy. Thus, the study concluded that using active learning methodologies improves the accuracy of learning, and subsequently prediction performance.

On the other hand, two studies have examined the most important measures that are associated with high bankruptcy risk. Campbell, Hilscher, and Szilagyi observed that financial and market-based forecast are major indicators of high risk of bankruptcy [6]. The study specifically noted stock, capital asset and return on investment as clear predictors of bankruptcy. Similarly, a study done by Situm [7] assessed firm size and age as possible predictors of high risk of bankruptcy. However, the research found that company size and age are no the main factors to consider when predicting bankruptcy.

3 METHODS

3.1 Data

The dataset was imported from UCI Machine Learning Repository [13]. The dataset consists of 64 calculated ratios which are obtained from the companies' financial annual report, including profit and loss statement and income statement. The target value is categoric with 1 means "bankrupt" and 0 for "non-bankrupt". The size of the files is different, as well as the percentage of the bankruptcy instances. The data was also collected for surviving companies. The size of the files is different, as well as the percentage of the bankruptcy instances. For Example, year 1 consists of 5910 instances while bankruptcy makes only 6.9% of the data.

One of the technique to deal with an imbalanced dataset is oversampling. This method works with minority class by duplicating the minority class (bankrupt instances) to be equal to the majority class (non- bankrupt instances).

3.2 Oversampling

Since the data is imbalanced where the bankrupt companies are way fewer than the surviving company, it is crucial to balance the data in order to improve the performance of the model in terms of predicting the minority class. Learning algorithms build their models by learning the dataset and finding patterns between the inputs, which are the features in our case, and the output. If less information about a certain class is provided to the algorithms, they will fail to predict that class. Therefore, it is important to have enough information about all classes in order to build a reliable model. Oversampling is among other techniques that can be used to overcome the issue of imbalanced data. Oversampling is favorable because it does not omit data that might be relevant to the target variable. The technique we used is to randomly duplicate the minority class to make a certain percentage of the data. Three different ratios of bankrupt to nonbankrupt were created to examine the impact of imbalance in the reliability of the models. The ratios are original data, 30/70 (bankruptcy class makes 30% of the data), and 50/50 (bankruptcy class makes 50% of the data).

3.3 Feature Selection

Feature selection is a crucial step to create an accurate predictive model. There are four types of features: predictive, interacting, redundant and irrelevant [8]. Predictive features provide useful information to predict the target. Interacting features are useful only when combined with other features but not by themselves. Redundant Features are features that have a strong correlation with other features. Irrelevant features are useless and don't provide any information to predict the target value. Thus, we try to identify those features in order to find the best subset that gives the best prediction results. Removing irrelevant and redundant features improve the prediction models by focusing only on the features that are correlated to the target value. This also leads to avoiding overfitting which makes the model limited to predict the testing set only but not instances that are new to the model. In this study, finding the most important features has an economic importance because companies can evaluate their performance by focusing on those features.

There are different methods to identify the key features. Each method has its pros and cons, but we observed that each method identifies different features to be the most important. In this study, we tested three techniques and compared them based on the results of the prediction models.

3.3.1 Mutual Information

Mutual information is a univariate method. Univariate methods are simple statistical techniques that examine each feature individually and determine the strength of the relationship of the feature with the target variable. There are a lot of different options for univariate selection, one of which is Mutual Information. Mutual information has been widely used in machine learning for the classification tasks to find the most important features [9]. Mutual information is a measure between two variables X and Y and one variable gives the amount of the information about another variable. Each variable is unrelated (independent features). MI between two random variables is a non-negative value,

which measures the dependency between the variables. It is equal to zero if and only if two random variables are independent, and higher values mean higher dependency. Mutual information is given by:

$$I(X;Y) = \int_X \int_Y p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dxdy \quad (1)$$

where "*I*" is the mutual information value. *x* and *y* are two variables. *p(x)* and *p(y)* are the probability distribution of *x* and *y*. *p(x,y)* is the joint probability density function of x and y. If x and y are completely uncorrelated, then *p(x, y)* would equal *p(x)p(y)*, and the integral would be zero. In this study, y is the bankruptcy while *x* is the feature.

3.3.2 Random Forest (Embedded Method)

Univariate techniques don't take into consideration the correlation between the features. MI, as described previously, measures the dependency between each feature and target value but not among the features. Therefore, univariate techniques might include redundant features and cannot identify interacting features which add valuable information if put together. As a result, there is a need to test another method that ranks features based on their performance in a model. Embedded techniques can solve this problem. Embedded methods incorporate knowledge about the specific structure of the class of functions used by a certain learning machine. Random Forest has feature selection embedded in the algorithm. The tree-based strategies used by random forests naturally ranks the features by how well they improve the purity of the node. Nodes with the greatest decrease in impurity happen at the start of the trees, while notes with the least decrease in impurity occur at the end of trees. As a result, a subset of the most key features can be created. In our study, we used "RandomForestClassifier" function in python which provides, along with the model, the significance of each feature as a percentage in the prediction model. We ranked the features based on that and selected the most significant five features, to be strategy "best-5" in our models.

3.3.3 Genetic Algorithm

A more computationally expensive but rather more effective method is the Genetic Algorithm. GA is a method to find the best solution for a problem, based on natural selection, from all feasible solutions. In Genetic Algorithms, a set of possible solutions in the feasible region (called individuals) are encoded to the optimization problem by a population of strings. The quality of a solution corresponding to an individual is measured using a fitness function. Generally, the evolution process begins with arbitrarily produced individuals based on their fitness. Individuals are selected from every generation and modified to form a new population, which is used in the next iteration. An algorithm stops only if the maximum number of generations has been produced or an acceptable fitness level has been achieved. It is not necessary that an acceptable fitness level will be achieved if GA is ended by the maximum number of generations.

The steps used in to identify the best subsets of features using GA are described as follows:

Step 1: select n number of sets of features randomly. each subset consists of 64 digits: 1 means attribute is included, and 0 means attribute is not included
Step 2: for each set, Random Forest learning algorithm is used to train the data and calculate the ROC which is our fitness function.
Step 3: choose the sets that will recombine for the next generation using crossover probability of 0.5 and mutation probability of 0.2. the crossover process recombines the selected sets to generate a new population.
Step 4: repeat the process for 20 generations
Step 5: List the set that results in the highest ROC

GA is performed for the four different ratios that we consider in our study. We also compared the results of GA for each year because the performance of the companies differs as they get closer to bankruptcy.

3.4 Learning Algorithms

We used Scikit-learn package in python to implement the six classification algorithms [10]. Python's Scikit-learn is a machine learning library supporting various classification, regression and clustering algorithms and is designed to operate with the NumPy (numerical) and SciPy (scientific) libraries. The algorithms are briefly described below.

3.4.1 Neural Network

Inspired by the information processing system concept of the brain, NN is composed of interconnected neurons working in to accomplish specific goal of solving a particular problem. NN consists of the input function, the training synapse, and the output function. NN is trained in a way that it can associate output with a particular input pattern [11]. In case an input pattern has no corresponding output pattern, the NN apply the training (learning) to produce output pattern. Thus, NN is effective for analyzing unstructured data.

3.4.2 Decision Trees

DT is a supervised learning algorithm used for classification and regression to help in coming up with a model that can be used in the prediction of the target variables value by learning simple decision-making rules derived from the input pattern. It is effective in multi-output problems. Several outputs can be inferred from a single input pattern. Decision trees are applied to classification problems, where the output is binary and regression problems, where the output is a continuous value.

3.4.3 Logistic Regression

LR is a supervised learning algorithm used for both classification and regression predictive problems by creating

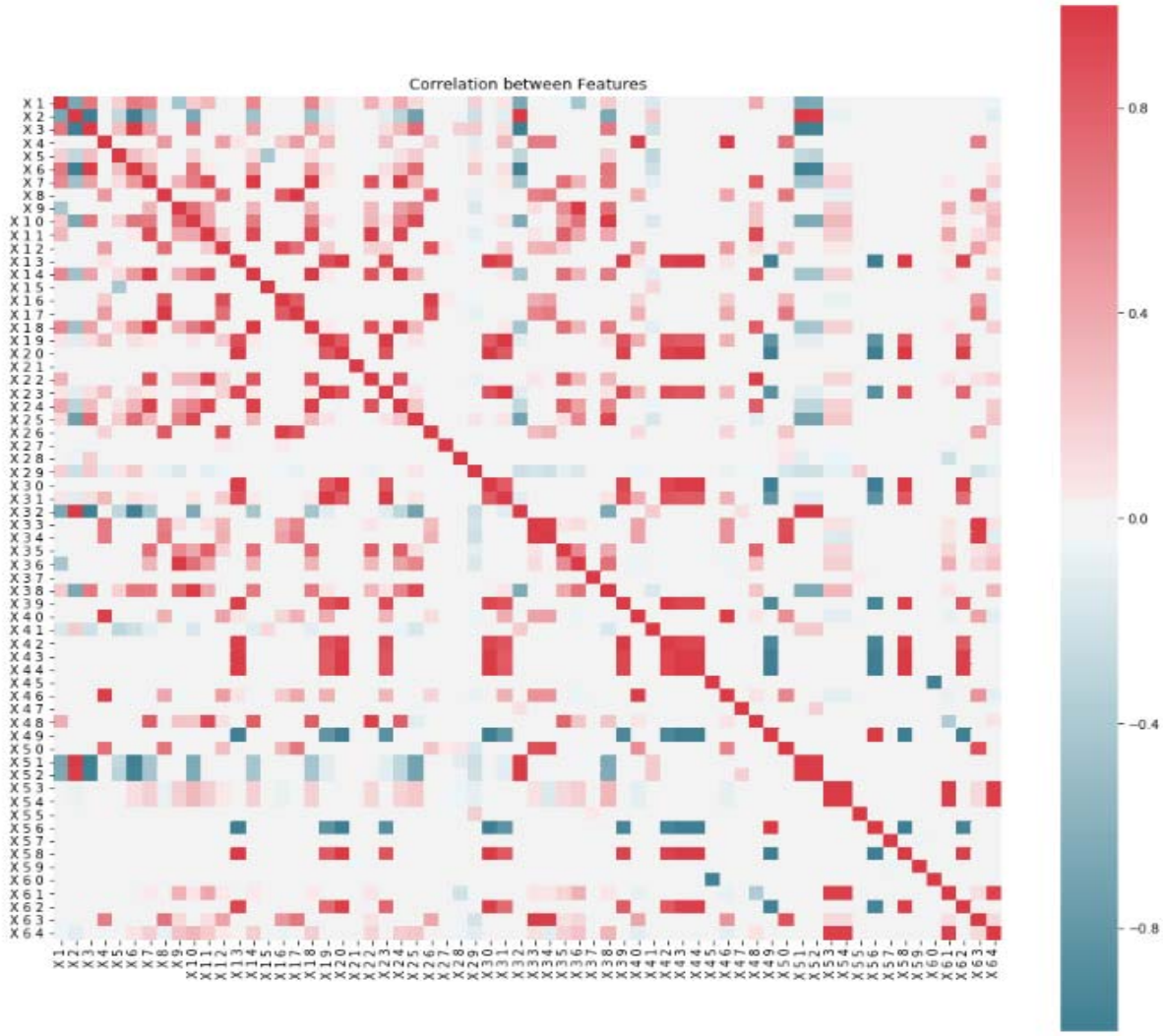

Fig. 1. Correlation matrix for year 1 explains the relationship between the features

an equation that makes a relationship between input *X* and output *Y*. Input values *(X)* are combined linearly using weights to predict an output value *(Y)*.

3.4.4 K-Nearest Neighbor

KNN is a supervised learning algorithm used for both classification and regression predictive problems. The performance of KNN in classification problem can be done by predicting the target feature with the majority vote from the set of *K* nearest neighbor where *K* is the closest training examples in the feature space [8].

3.4.5 Support Vector Machine

SVM is a supervised learning algorithm used for both classification and regression to help in coming up with a model that can be used in the prediction of the target variables value [8]. SVM plots each data item as a point based on the number of features. The value of each feature can be the value of a particular coordinate. Then the performance of the classification can be done by using the decision boundary that differentiates between the two classes.

3.4.6 Random Forest

Random Forest improves prediction of decision trees model by ensuring less or reduced correlation among the independent decision tree model [12]. The algorithm randomly selects a sample of independent decision trees using the following equation:

$$N = \text{sqrt (number of input variables)} \quad (2)$$

where *N* is the number of randomly selected patterns.

3.5 Evaluation Metrics

The performance of the ML algorithms was assessed using Accuracy ROC AUC, F1-score, Precision, and Recall. First, the following terms are defined:

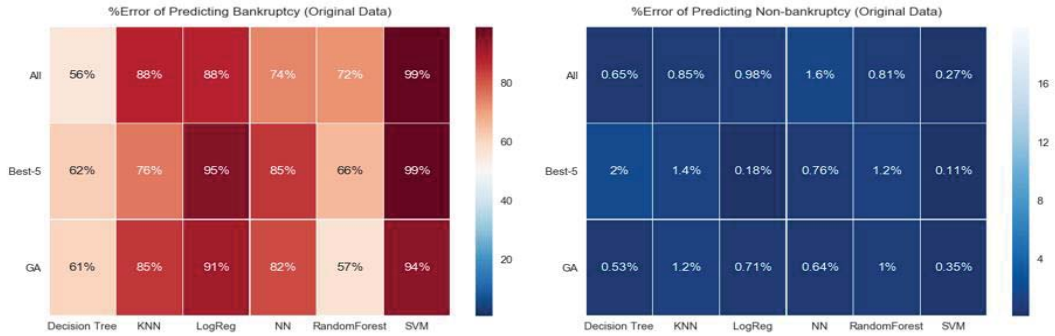

Fig. 2. The percentage of error in predicting bankruptcy and non-bankruptcy using year 1's original data

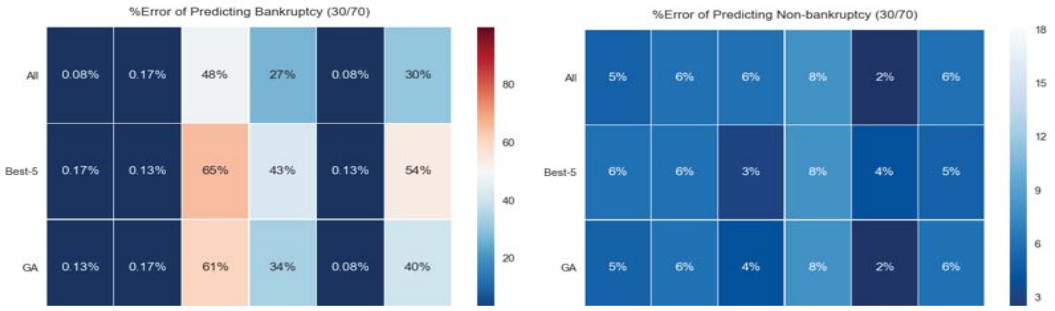

Fig. 3. The percentage of error in predicting bankruptcy and non-bankruptcy using year 1's 30/70 data

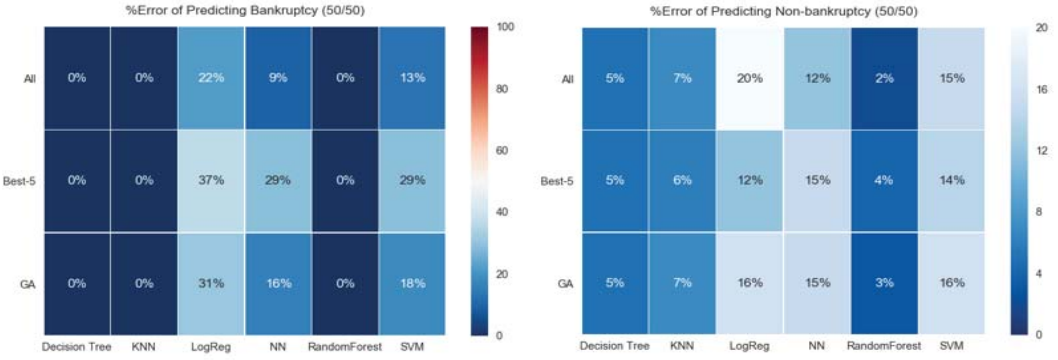

Fig. 4. The percentage of error in predicting bankruptcy and non-bankruptcy using year 1's 50/50 data

- True Negative (TN): the case was negative and the predicted is negative
- True Positive (TP): the case was positive and the predicted is positive
- False Negative (FN): the case was positive but the predicted is negative
- False Positive (FP): the case was negative but the predicted is positive

Thus,

$$\text{Accuracy} = TP+TN / (TP + TN + FP + FN) \quad (3)$$

$$\text{Precision} = TP / (TP + FP) \quad (4)$$

$$\text{Recall} = TP / (TP + FN) \quad (5)$$

$$\text{F1-Score} = 2 * (\text{Recall}*\text{Precision}) / (\text{Recall} + \text{Precision}) \quad (6)$$

Recall measure performance for all the positive part of the dataset while precision measure performance based on positive predictions. Evaluating the performance of the ML algorithms is based on the perfect ROC curve. The closeness to the perfect ROC AUC shows high performance.

4 Results

4.1 Feature Selection

The three methods produced different subsets of features to be the most important. Because MI is a statistical method that studies the relationship between each feature and the output independently without taking into consideration the correlation among the features, MI showed that almost all the 64 features are significant.

However, correlation analysis, Figure 1, shows that many features have the same level of correlation with the target value and the other features. In Figure 1, the red color indicates a negative correlation while blue means positive. For example, features 42, 43 and 44 have the same correlation behavior with other features which indicates that

those three features provide the same information and that keeping only one would improve the model. Despite that, MI gives those three features the same amount of significance. In our study, we decided to test the models by including all the features to examine the effectiveness of using univariate methods to select features. We call this "All" in our model.

On the other hand, Random Forest identified only a few features to be the most significant. We selected to train the algorithms with including only the highest 5 features, we call it "Best-5", which are:

X_{27}: profit on sales / total assets
X_{35}: profit on sales / sales
X_{39}: total liabilities / ((profit on operating activities + depreciation) * (12/365))
X_{41}: profit on operating activities / total assets
X_{22}: profit on operating activities / financial expenses

This result makes more sense because those five features have different relationships with the other features as can be seen in Figure 1. Also, from a financial point of view, those features highlight the main Key Performance Indicators, which are profit, sales, cost and liabilities, which can provide sufficient information about the performance of a company. The result in the next section supports this as "Best-5" gives a better result than "All".

The final strategy is GA (also referred to as "best-mean"). GA is dependent on the algorithm that is used to train the data and on the metric that is used as a fitness function. We treat each year separately without considering the performance of a specific company throughout the 5 years. Therefore, GA identifies different sets of features to be the most significant in each year: 16 features for year 1, 8 for year 2, 10 for year 3, 15 for year 4, and 13 features for year 5. However, there are features that appear to be common. It is noticed that the best 5 features using Random Forest are among the best features identified by GA. This is expected since both methods used Random Forest to train the data. Other algorithms also prove to give a good result using the features selected with GA.

4.2 Oversampling

4.2.1 Original Data

We first applied the algorithms to the original data to examine the performance of the algorithms before balancing the data. This gives a better understanding of how difficult it is to build prediction models for imbalanced data. Figure 2 shows the percentage of error in predicting both classes, bankrupt and non-bankrupt, for year 1. When predicting the minority class (bankrupt), the results from all the algorithms are unsatisfactory as the percentage of error varies from 56% to 99%. This is expected since the number of the companies that went bankrupt is very low. The algorithms cannot find the patterns that lead to bankruptcy. Thus, oversampling seems to be a necessary step in this case.

4.2.2 30/70

Increasing the number of bankruptcy incidents improves the performance of the models significantly especially Random Forest, KNN, and Decision Tree. As shown in Figure 3, the error of predicting bankruptcy considerably reduce to very low levels. However, the error rate increases for the non-bankrupt class, but the increase is not significant. Overall, increasing the ratio of the minority class to 30% was found to be very effective.

4.2.3 50/50

In this strategy, we duplicated the bankrupt instances to be equal to the non-bankrupt and assessed the performance of the algorithms. The percentage of error as shown in Figure 4 indicates some sort of overfitting as some models achieved 0% of error.

On the other hand, Figure 5 illustrates that Random Forest, KNN, and Decision Tree have very high ROC AUC values. This is a sign that those models would be so great predicting bankruptcy if the tested data was the same as the data used to train the algorithms. LogReg seems to be an ineffective algorithm in this study despite raising the minority class instances to 50%. NN showed an improvement but it still gave ROC AUC of below 90%.

In terms of selecting the best features, Best-5 seems to be a reasonable choice to avoid overfitting. Including all the features resulted in a higher ROC AUC, leading to overfitting. GA provided results that are very close to All. This also suggests that using embedded techniques are sufficient to identify the key features.

5 Conclusion

In this study, we tested six classification algorithms to predict bankruptcy events. When evaluating the performance of the algorithms, accuracy seems to be misleading because the models always do good in predicting the majority class which, eventually, leads to a high accuracy regardless of the models' ability to predict the minority class. Thus, we focused on the other metrics which give a better idea of the models' performance.

The original data has the problem of imbalance where the bankrupt instances are much fewer than the non-bankrupt instances. As a result, it was hard for the algorithms to predict bankruptcy while they did very well predict the majority class except for Random Forest which seems to be a better option if we choose to run the models without balancing the data. It is also noticed that the performance of the models is better when predicting bankruptcy one year ahead. This indicates that predicting bankruptcy 2 years or more ahead is very difficult because the market changes rapidly while companies' performance might vary from one year to another.

From 30-70 and 50-50 results, we saw improvements in the performance of the algorithms as % error decreases and ROC-AUC increase. Best-5 gives the best results compared to other strategies. Thus, we concluded that the those five are the most important features. Logistic Regression performed

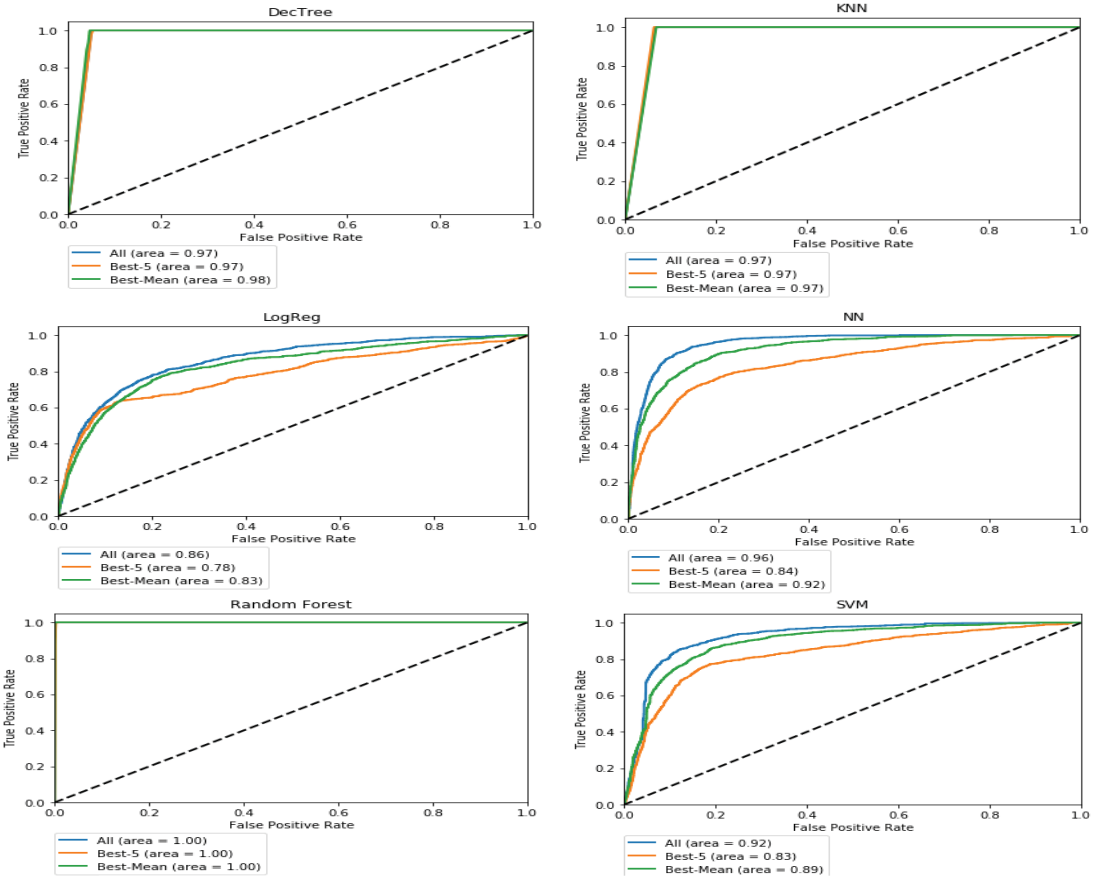

Fig. 5. ROC curves for year 1 (50/50)

very poorly for all strategies and in the five years, while Random Forest, Decision Tree, and KNN seem to be the most appropriate algorithms for such a problem.

Overall, including all the attributes leads to overfitting in all the ratios. GA proves to be a very effective method to select the key features and to increase the reliability of the models. Best-5 strategy seems to be a suitable alternative especially if we consider that GA is very time consuming and very expensive in computational terms.

Oversampling proves to be a good balancing method, but it is important to understand the drawbacks of using it, such as overfitting.

The main challenge in this study is imbalanced data. This is a very common problem in ML and requires the attention of researchers. Thus, moving forward, future research studies on the most effective method of dealing with imbalanced data are necessary to reduce the complexity of this kind of study.

6 References

[1] K. Nagaraj and A. Sridhar, "A predictive system for detection of bankruptcy using machine learning techniques," International Journal of Data Mining & Knowledge Management Process, pp. 1-11, 2015.

[2] F. Wagenmans, "Machine learning in bankruptcy prediction," Faculty of Science Theses: Utrecht University Repository, 2017.

[3] F. Barboza, H. Kimura and E. Altman, "Machine learning models and bankruptcy prediction," Expert Systems with Applications: An International Journal, vol. 83, no. C, pp. 405-417, 2017.

[4] G. Kostopoulos, S. Karlos, S. Kotsiantis and V. Tampakas, "Evaluating active learning methods for bankruptcy prediction," International Conference on Brain Function Assessment in Learning, vol. 10512, pp. 57-66, 2017.

[5] P. Pompe, A. Feelders and A. Feelders, "Using machine learning, neural networks, and statistics to predict bankruptcy," Journal of Microcomputers in civil engineering, vol. 12, no. 12, pp. 267-276, 1997.

[6] J. Y. Campbell, J. D. Hilscher and J. Szilagyi, "Predicting financial distress and the performance of distressed stocks," Journal of Investment Management, vol. 9, no. 2, pp. 14-34, 2011.

[7] M. Situm, "The relevance of employee-related ratios for early detection of corporate crisis," Economic and Business Review, vol. 16, no. 3, pp. 279-314, 2014.

[8] J. D. Kelleher, B. M. Namee and A. D'Arcy, Fundamentals of machine learning for predictive data analytics: Algorithms, worked examples, and case studies, Massachusetts: MIT Press, 2015.

[9] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," IEEE Transactions on Neural Networks, vol. 5, no. 4, pp. 537-550, 1994.

[10] "scikit-learn: Machine Learning in Python", 2017. [Online]. Available:http://scikit-learn,org/stable/

[11] C. Stergiou and D. Siganos, "Neural networks," 2017. [Online]. Available:https://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html.

[12] V. Jha, "Random forest – Supervised classification machine learning algorithm," 15 June 2017. [Online]. Available: https://www.techleer.com/articles/107-random-forest-supervised-classification-machine-learning-algorithm/. [Accessed 17 June 2018].

[13] Z. Tomczak, and J. Tomczak, "Ensemble Boosted Trees with Synthetic Features Generation in Application to Bankruptcy Prediction. Expert Systems with Applications" 2016. [Online]. Available:https://archive.ics.uci.edu/ml/datasets/Polish+companies+bankruptcy+data