Bankruptcy prediction using Machine learning models

Mani Krishna Tippani[a,*], Surender Gandham[b], Srikanth Konkata[c]

[a] 19mbmb11, Business Analytics, School Of Management Studies, University Of Hyderabad
[b] 19mbmb13, Business Analytics, School Of Management Studies, University Of Hyderabad
[c] 19mbmb16, Business Analytics, School Of Management Studies, University Of Hyderabad

article info

Submitted to
Prof. Vijay Bhaskar Marisetty
Submission Date-21-12-2020

Keywords:
Bankruptcy prediction
Machine learning
Logistic Regression
Support vector machines
Stacking
Random forest

abstract

There has been intensive research regarding models for predicting bankruptcy and default events, for credit risk management. In this study, we test with 8 additional predictor variable using machine learning models (logistic Regression, k-NN, naïve bayes, support vector machines, Stacking using logistic regression and random forest) to predict bankruptcy and compare their performance with results of original Altman's Z-score variables, we include six complementary financial indicators. Here, the dataset of different companies across several industries and their different fianacial ratios from period of 2000-2017 are taken to predict the bankruptcy using BSE group as a basis forTargetvariable.

1. Introduction

Financial institutions, fund managers, lenders, governments, and financial market players seek to develop models to efficiently assess the likelihood of counterparty default. Although default events behave stochastically, capital market information can be used to develop bankruptcy prediction models. For example, Altman (1968), in a seminal paper, applies multivariate statistical techniques, primarily discriminant analysis, to classify solvent and insolvent companies using financial statement data.

The purpose of this paper is to predict bankruptcy using machine learning models. We also introduce 8 additional variables. The data set consists of more than 39907 rows and 60 columns. Target variables is defined as bankruptcy of 2 levels(1- BSE groups A,B, 0- BSE group Z). The purpose of the bankruptcy prediction is to assess the financial condition of a company and its future perspectives within the context of long-term operation on the market. Typically, enterprises are quantified by numerous indicators that describe their business condition that are further used to induce a mathematical model using past observations.

Contributions of this study are reflected in: (1) to take additional variables to the existing model to improve accuracy (2) apply machine learning techniques and (3) comparing the performance results.

The data is trained and test on models like k-NN, logistic regression, support vector machine kernels, naïve Bayes and stacking (a meta classifier) is applied on the top of these models to combine predictions to improve accuracy and control over fitting using a

random forest meta estimator explained in **section 2**.

2. Theoretical background

Machine learning in finance is the utilization a variety of techniques to intelligently handle large and complex volumes of information.

ML excels at handling large and complex volumes of data, something the finance industry has in excess of. Due to the high volume of historical financial data generated in the industry, ML has found many useful applications in finance. The technology has come to play an integral role in many phases of the financial ecosystem, from approving loans and carrying out credit scores, to managing assets and assessing risk.

Application of ML techniques is simple and effective in predicting the bankruptcy of a company and mines the parameters which are responsible. This section mainly reviews the mechanism of machine learning methods applied in this research.

2.1 Logistic Regression

It is one of the simplest classifier which uses sigmoid function to classify the things. Logistic Regression is used when the dependent variable (target) is categorical. The logit model is used to predict the probability of occuring an event. The sigmoid function refers if the probability is below 0.5 then it is considered as 0 and if the probability is more than 0.5 then it is considered as 1.There should be no high correlations (multicollinearity) among the predictors.

The logistic function is defined as:

$$\mathbf{logistic(\eta)=1/1+exp(-\eta)}$$

The step from linear regression to logistic regression is kind of straightforward. In the linear regression model, we have modelled the relationship between outcome and features with a linear equation:

$$\hat{y}^{(i)} = \beta_0+\beta_1 x_1^{(i)}+\ldots+\beta_p x_p^{(i)}$$

For classification, we prefer probabilities between 0 and 1, so we wrap the right side of the equation into the logistic function. This forces the output to assume only values between 0 and 1.

$$P(y^{(i)}=1)=1/1+exp(-(\beta_0+\beta_1 x_1^{(i)}+\ldots+\beta_p x_p^{(i)}))$$

2.2 K-Nearest Neighbour (K-NN)

K Nearest Neighbour is a lazy learning algorithm which uses the distance for classification which can be interpreted in the terms like proximity or closeness. The k value is the key for classifying the things in a predominant manner. KNN makes no assumptions about the functional form of the problem as it referred to as non-parametric machine learning algorithm. It uses Euclidean distance to calculate the proximity.

2.3 Naive Bayes

Naïve Bayes is collection of algorithms based on the BAYES theorem in probability and statistics. The classifiers assume that all the features are independent of each other and contribute to the classification of the things accordingly. Naïve Bayes is particularly probabilistic model.

Let's take a target function $f : X \rightarrow Y$, or equivalently $P(Y|X)$. We will assume Y is a Boolean-valued random variable, and X is a vector containing n Boolean attributes. In other words,

$X = X_1, X_2 \ldots, X_{ni}$, where Xi is the Boolean random variable denoting the ith attribute of X. Applying Bayes rule, we see that $P(Y = y_i |X)$ can be represented as 1.

$$\mathbf{P(Y = y_i |X = x_k) = P(X = x_k|Y = y_i)P(Y = y_i) / \sum_j P(X = x_k|Y = y_j)P(Y = y_j)}$$

where y_m denotes the m^{th} possible value for Y, x_k denotes the k^{th} possible vector value for X, and where the summation in the denominator is over all legal values of the random variable Y. One way to learn P(Y|X) is to

use the training data to estimate P(X|Y) and P(Y). We can then use these estimates, together with Bayes rule above, to determine $P(Y|X = x_k)$ for any new instance x_k.

2.4 Support Vector Machines (SVM)

Support Vector Machines are basically dichotomous classifiers. This SVM basically represents the datum as points in space. So that the different group of points are classified differently. There are different types of kernels available for SVM. Here, LINEAR and RADIAL kernels are used for classification of the things. In the SVM algorithm, each data item is plotted as a point in n-dimensional space (where n is number of features) with the value of each feature being the value of a particular coordinate. Then, classification is performance by finding the hyper-plane that differentiates the two classes very well. Support Vectors are simply the co-ordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper-plane/ line). SVM algorithm is implemented with kernel that transforms an input data space into the required form. SVM uses a technique called the kernel trick in which kernel takes a low dimensional input space and transforms it into a higher dimensional space. In simple words, kernel converts non-separable problems into separable problems by adding more dimensions to it. It makes SVM more powerful, flexible and accurate.

2.4.1 Linear SVM:

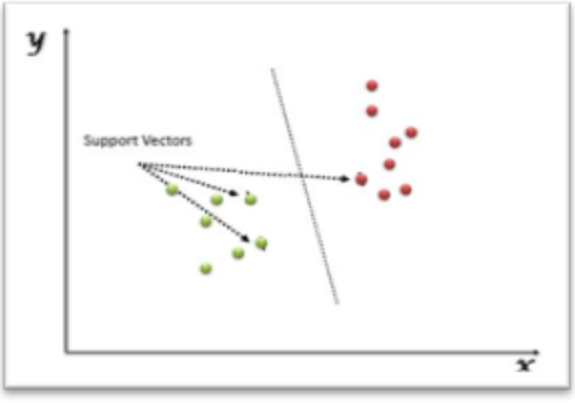

Figure 2 SVM Linear kernel Mechanism

Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM. It uses a linear kernel.

Here's the function that defines the linear kernel:

$$f(X) = w^T * X + b$$

In this equation, **w** is the weight vector that you want to minimize, **X** is the data that you're trying to classify, and **b** is the linear coefficient estimated from the training data.

2.4.2 Non-linear SVM (Radial Basis Function)

Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data

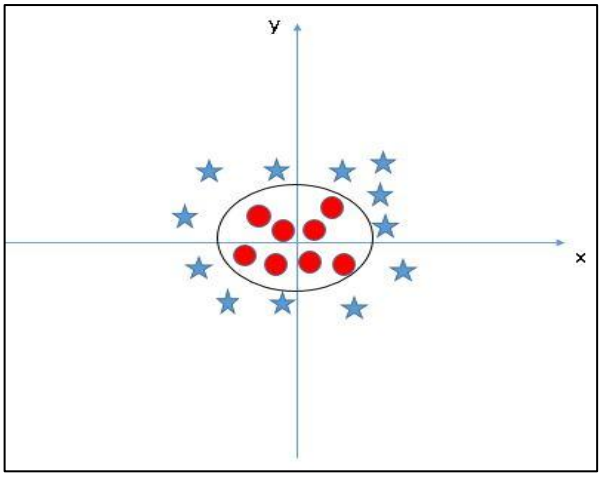

Figure 1 SVM RBF mechansim

and classifier used is called as Non-linear SVM classifier.

Here's the equation for an RBF kernel:

$$f(X1, X2) = \exp(-gamma * ||X1 - X2||^2)$$

In this equation, **gamma** specifies how much a single training point has on the other data points around it. **||X1 - X2||** is the dot product between your features.

2.5 Stacking

Stacking is a meta classifier model which is a part of ensembles method used to build multiple models

(typically of differing types) and supervisor model that learns how to best combine the predictions of the primary models. It uses the strength of each base classifier by

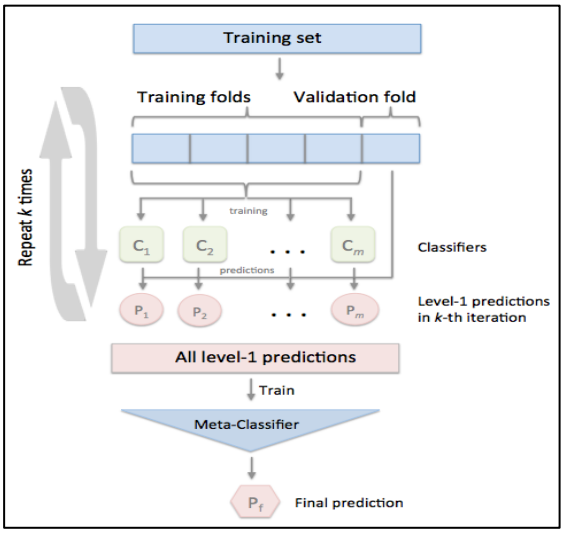

Figure 3 Stacking Meta Classifier MEchanism

using their output as the input for final classifier.

The base level models are trained based on a complete training set, then the meta-model is trained on the outputs of the base level model as features. The base level often consists of different learning algorithms and therefore stacking ensembles are often heterogeneous.

2.6 Random forest Classifier

Random forest classifier is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. Each model in the ensemble is used to generate m predictions run over a new sample of dataset to generate predictions and further averaged to give final prediction.

Random forest classifier creates a set of decision trees from a randomly selected subset of training set. It is a set of decision trees from a randomly selected subset of the training set and then it collects the votes from different trees to decide the final prediction. Every decision tree has high variance, but when we combine all of them together the resultant variance is low as each decision tree gets perfectly trained on that particular sample data and hence output doesn't depend on one decision tree but multiple decision trees.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of over fitting. Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random forest classifier: There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result. The predictions from each tree must have very low correlations.

Our RF algorithm follows:

1. Create random subsets of the parent set, composed of an arbitrary number of observations and different features.

2. Each subset from step 1 produces a decision tree, and all elements of the set have a label (correct or not).

3. For each element, the forest takes a large number of votes. The class with the most votes is chosen as the preferred classification of the element. These investigate applications of machine learning techniques in financial issues, mainly in bankruptcy prediction.

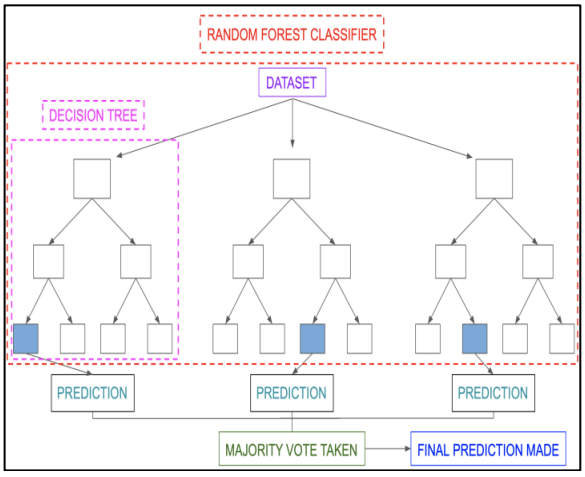

Figure 4 Random Forest Classifier Mechanism

3. Data and Method

Table 1 Predictor variables used for the study

Variable	Formula
X1 (Liquidity)	Net working capital / Total assets
X2 (Profitability)	Retained earnings / Total assets
X3 (Productivity)	Earnings before interest and taxes / Total assets
X4 (leverage)	Market value of share * number of shares/ Total debt
X5 (asset turnover)	Sales / Total assets
Operational Margin (OM)	Earnings before interest and taxes / Sales
Growth of Assets (GA)	$\text{Total assets}_t - \text{Total assets}_{t-1} / \text{Total assets}_{t-1}$
Growth in Sales (GS)	$\text{Sales}_t - \text{Sales}_{t-1} / \text{Sales}_{t-1}$
Growth in number of Employees (GE)	$\text{Number of employees}_t - \text{Number of employees}_{t-1} / \text{Number of employees}_{t-1}$
Change in Return on Equity (CROE)	$\text{ROE}_t - \text{ROE}_{t-1}$
change in price-to-book ratio (CPB)	$\text{Price-to-Book}_t - \text{Price-to-Book}_{t-1}$
PBDIT	net profit+interest+taxes+depreciation and amortization
PBDITA to Income	(Operating income + Amortization+Depreciation)/Income
Return on Assets (ROA)	Net income /Total Assets
Current Ratio	Current Assests / current liabilities
Debt to Service Coverage Ratio	Net operating income / Debt Service
Net cash Flows from Operations	Net Income + Depreciation, Depletion, & Amortization + Adjustments To Net Income + Changes In Accounts Receivables + Changes In Liabilities + Changes In Inventories + Changes In Other Operating Activities
Debt to Equity	Total liabilities/Shareholders Equity

Financial data of companies across several industries with different financial ratios from the period of 2000-2017 are considered to determine the risk of bankruptcy for a company. The data contains 39907 rows and 60 columns. Further dataset has been reduced to 2000 rows and 19 variables. BSE group has been taken as the base for defining target variable (bankruptcy). The values Z are classified as bankrupt (1) and the groups A and B as non-bankrupt (0).

This dataset has a total of 2000 observations and 21 columns, out of which are 19 predictors and bottom 7

are extra variable's which have been added to test using ML methods to improve accuracy.

All variables were included in the models at their original values. No transformation, such as normalization, was conducted. The use of originally available data without any transformation was also followed. Total of 7 techniques were applied:

- K-NN and K-NN with Hypertuned parameters
- Logistic Regression
- Naïve Bayes
- SVM linear
- SVM radial Basis Function
- Stacking using Logistic Regression
- Stacking Using Random Forest

We implemented the models using the R statistical software packages. In this study, we used caTools, caret, randomForest, ROSE, caretEnsembles, MASS, ROCR. The Dataset is split into 70:30 ration for training and testing. For the machine learning models, the logistic regression, naïve bayes, SVM Linear and RBF kernels, k-NN were used as base learners for prediction.

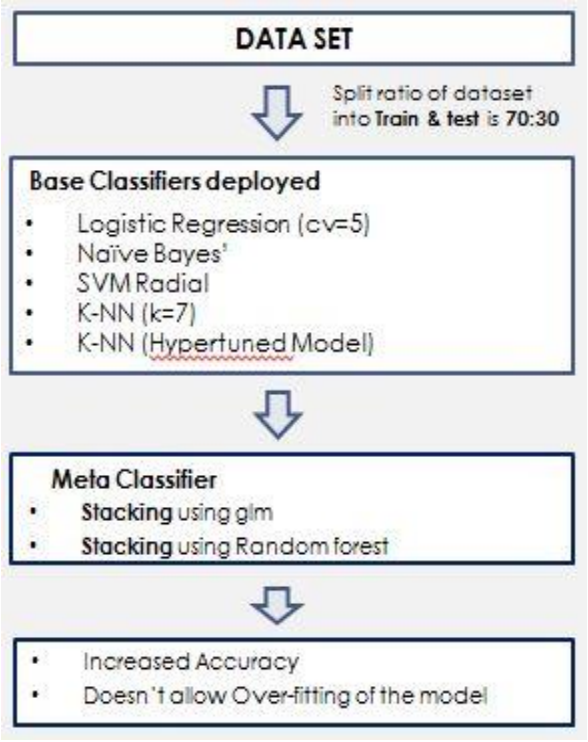

Figure 5 Data Analysis Methodology

Then using ensemble methods, the predictions (outputs) of base classifiers are combined to feed as inputs for stacking. Stacking uses these output predictions as inputs for testing on the sample data. Stacking with random forest and logistic regression algorithms were applied to form a combined model for prediction to improve accuracy and control over-fitting of the data.

4. RESULTS

Classifier	Training Accuracy	Training kappa	Prediction Accuracy	Prediction kappa	Sensitivity	Specificity
Logistic Regression	0.8163628	0.616077	0.8144	0.6124	0.7958	0.8489
SVM Linear	0.7931108	0.5566724	0.7908	0.5526	0.7473	0.9083
SVM Radial	0.8155859	0.6214083	0.8493	0.6882	0.8493	0.8647
KNN (k=7)	0.7616147	0.5075571	0.7435	0.4688	0.7523	0.7292
KNN (Hypertuned Model)	0.7549223	0.4944547	0.8031	0.5977	0.8242	0.7745
Naïve Bayes	0.8253684	0.6342016	0.8133	0.6097	0.7935	0.8506

Table 2 Prediction results of Base classifiers after addition of Extra Variables

SVM Radial algorithm shown a very good accuracy of 84.93% compared to all other base algorithms deployed.The cohen kappa statistic is also high for this SVM Radial model which shows a good inter rater reliability of the predictors with the target variable. Sensitivity or true positive rate which is the major model evaluation for the bankruptcy is also high with SVM radial Model.

But the Specificity is higher for SVM Linear as shown in **Table 2.**As you see highest sensitivity and highest specificity is different for different base classifier.

Therefore these kind of best predictions are combined to make a hybrid model using Stacking. Stacking with glm and random forest are performed. The results show that there is an increase in accuracy and kappa after applying Stacking as shown in **Table 2**.

Classifier	Prediction Accuracy	Prediction kappa
SVM Radial	**0.8493**	**0.6882**
Stacking using GLM	**0.86509**	**0.72153**
Stacking using Random forest	**0.85666**	**0.75517**

Table 3 Stacking Performance results comparison with SVM Radial

4.1 Comparison of performance results with Altman Variables

Classifier	Training Accuracy	Training kappa	Prediction Accuracy	Prediction kappa	Sensitivity	Specificity
Logistic Regression	0.7294624	0.4308659	0.7424	0.4556	0.7264	0.7782
SVM Linear	0.6994039	0.348899	0.6985	0.3438	0.6705	0.8032
SVM Radial	0.7871071	0.5679355	0.8065	0.604	0.824	0.7823
KNN (k=7)	0.6971586	0.3839079	0.6828	0.3515	0.7193	0.633
KNN (Hypertuned Model)	0.7031103	0.3990414	0.7312	0.4552	0.7757	0.6774
Naïve Bayes	0.7361816	0.4437002	0.7402	0.4487	0.7204	0.7871

Table 4 Prediction results of Base classifier before addition of Extra Variables

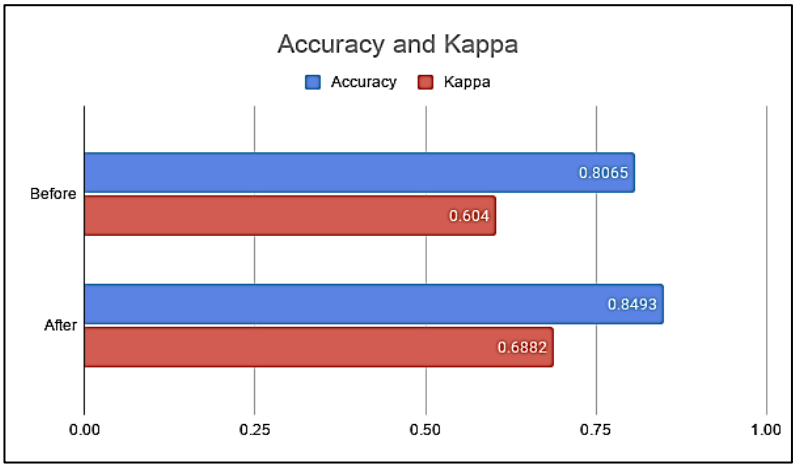

Figure 6 comparing performance results after adding extra variables and altman varibales

Accuracy and Kappa statistic are increased after adding extra variables to Altman variables as shown in **Figure 6**.

The incremental percentage of the accuracy is 4.28% after adding the extra variables. The kappa has also an increase upto 0.0842. The addition of variables has shown a significant effect on predicting the bankruptcy.

5. Conclusion

The addition of variables has shown a significant effect on predicting the bankruptcy. Adequate Training of the model with ample data makes the model to be more accurate. Applying Different Classifiers to train the model allows us in scalability in choosing the optimized model with good evaluation metrics. The best model obtained before stacking is SVM Radial. Ensemble methods also play a key role in training the model at a time. Stacking classifiers are one much technique which improves accuracy through combined best individual predictions to build a model for final prediction.

5. References

1. *Altman. (1968). FINANCIAL RATIOS, DISCRIMINANT ANALYSIS AND THE PREDICTION OF CORPORATE BANKRUPTCY. jstor, 22.*
2. *Dr M M Sulphey, N. S. (2013). THE ANALYTICAL IMPLICATION OF ALTMAN'S Z SCORE ANALYSIS OF BSE LISTED SMALL CAP COMPANIES. Global Institiute for Research Education, 11.*
3. *Edward Altman, H. K. (2017). Machine learning models and bankruptcy prediction. elsevier, 13.*
4. *Jakub Horak, J. V. (2020). Support Vector Machine Methods and Artificial Neural Networks Used for the Development of Bankruptcy Prediction Models and their Comparison. Journal of Risk and Financial Management , 15.*
5. *Obradović, D. B. (2018). Insolvency prediction model of the company. tandfonline, 20.*