Ensemble KNNs for Bankruptcy Prediction

Qi Yu[1], Amaury Lendasse[1],
and Eric Séverin[2]

1- Helsinki University of Technology - Information and Computer Science Department
Konemiehentie 2, Espoo - Finland

2- University of Lille 1 - Laboratoire Economie Management
59653 Villeneuve d'Ascq cedex - France

Abstract.
The business failure has been widely researched, trying to identify the various determinants that can affect the existence of firms. However, the variety of models as well as the variety of the theoretical frameworks, illustrates the lack of consensus on how to understand the phenomenon and the difficulties in formulating a general model interpretation. One hotspot nowadays is the prediction of the bankruptcy, the final stage of failure, which has been regarded as a classification problem. Thus, this paper presents a global methodology to classify bankrupt companies and healthy ones. The computational time of this method is extremely small since it use k nearest neighbors (KNN) to build several classifiers, each of the classifiers use different nearest neighbor on different subset of input variables and try to minimize the mean square error. Finally a linear combination of these classifiers is calculated to get even better performance. On the other hand, this method is robust because the ensemble of classifiers has smaller variance than each single classifier. The method is tested using a real world data, which comprises 41 financial variables measured on 500 companies (250 healthy and 250 bankrupt) from the year 2002 and 2003. The result confirms that the advantage of this method, which is that it is robust while it provides good performance and a comparatively simple model at extremely high learning speed.

1 Introduction

Bankruptcy prediction is the art of predicting bankruptcy and various measures of financial distress of public firms. Since the work of Beaver (1966) and Altman (1968) there has been considerable interest in using financial ratios for predicting the financial failures. Using univariate analysis Beaver concluded that 'Cash Earnings to Total Debt' was the best ratio for signalling bankruptcy, before long, people realized single predictor is far away from good enough.

Thus, in this paper, the dataset used has 41 input variables which are all important financial ratios like Total Sales/Total Assets, Total Debt/Shareholders' Funds, or Change in Equity Position, etc.

In the paper, we investigate the methodology: Ensemble KNNs to achieve these tasks in a very small computational time.

As we know, it is usual to have very long time for training a neural network even some other sophisticated models have been tested on bankruptcy prediction. Thus, K-Nearest Neighbors (KNN) which is a very simple but powerful tool comes into our mind. Ensemble modeling is also used on KNN techniques to get even better performance. The most significant characteristics of this method is that it tends to provide good classification results and a comparatively simple model at extremely high learning speed.

This method has several notable achievements:

- keeping good performance while being simpler than most learning algorithms,
- using randomly selected variables as the initialization to reduce the dimension of the input and computational time,
- the computational time of the method being extremely low. In our financial experiments, the computational time for one attempt is about 15 seconds (for a classification problem with 500 samples and 41 variables),
- Ensemble modeling is used both in ensemble the K different nearest neighbors and ensemble N KNN models.
- Leave-One-Out (LOO) technique is used in the training set, and PRESS statistic makes it easy to calculate.

All the steps are detailed in the Section 3. Before that, in order to get better results, Section 2 shows the whole process of the methodology. Section 4 shows the results on a toy example and on bankruptcy prediction task.

2 The methodology of Ensemble KNNs

In this section, a methodology called Ensemble KNNs is presented. Ensemble KNNs has several major steps like KNN, ensemble modeling and LOO. By using these techniques, the method successfully shows its accuracy and efficiency. The following part describes how the method works with those steps and the next section will detail the algorithms in each steps.

2.1 Description of the methodology

In order to get lowest mean square error for the classification task, we propose a global methodology in 4 steps.

- Initialization step: a subset of input variables are randomly selected for modeling. In the methodology, different subsets of input variables are random selected for each KNN model. This selection process takes no time since we use the random function and it reduces the dimension of input for KNN.
- KNN step: after input subset selected, number k is chosen to build k nearest-neighbors models. Then, ensemble k different nearest neighbors is done to get the best weights of each neighbors using NNSL algorithm. This is actually the first time ensemble idea is used in this paper. After that, LOO is used to calculate the output y_{loo}.

- Ensemble KNN step: for each subset of input variables, ensemble KNNs model is built in second step. Let us suppose we have N such models, meaning we have N such y_{LOO}. Therefore, in this step, we construct eventual ensemble modeling based on these N models. The weights are calculated again using NNSL algorithm by the N outputs. This is the second time in this paper we use the ensemble idea.

- Repetition Step: since the computational time of the above process is very small, take a 500×41 sized input data for example, it just takes 1 second, in practice, we always repeat the whole methodology several hundreds of times. And the final result is the mean of those. This step is not compulsory but optional.

The next block diagram summarizes the global methodology.

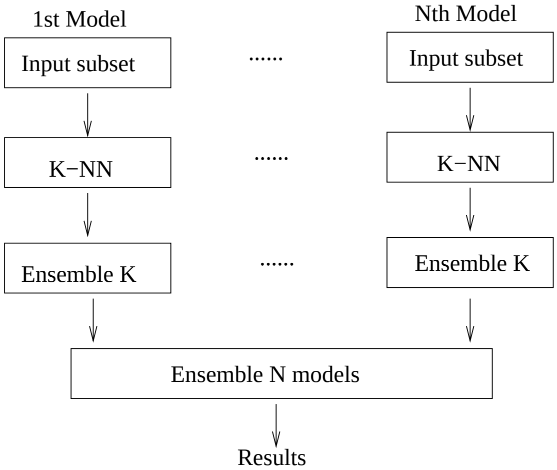

Figure 1: Block diagram of the global methodology

3 Algorithms of Each steps

3.1 K-Nearest Neighbors (KNN)

The method of k-Nearest Neighbors (KNN), an important approach to nonparametric classification, is quite easy and efficient. It has been used in many different applications and particularly in classification tasks. If we have infinitely many sample points, then the density estimates converges to the the actual density function. The classifier becomes the Bayesian classifier [12] if the largescale sample is provided. But in practice, given a small sample, the Bayesian classifier usually fails in the estimation of the Bayes error especially in a high-dimensional space, which is called the disaster of dimension. The methods of Parzen and k-NN are often used in the case of small sample. The following content is about the approach details of KNN.

There is a famous Chinese idiom which says the the Scotch cousins are not more helpful than the nearest neighbors. Because the things gather together by classes, so it is natural to assume that the interest belongs to the class of its nearest neighbors. The 'nearest neighbors' are determined if given the considered the vector and the distance measurement. Usually, we use Euclidean distance to measure the nearest k neighbors.

Given the training set $T = (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$, to determine the class of the input vector x. The most special case is the 1-NN method, which just searches the nearest neighbor:

$$j = \arg\min_{x} \| x - x_i \|$$

then, (x, y_j) is the solution.

> Definition: A training set $T = (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ is called $(k, d\%)$- stable if the error rate of k-NN method is $d\%$, where $d\%$ is the empirical error rate from independent experiments [3]. If the clusterings of data are quite distinct (the class distance is the crucial standard of classification), then the k must be small. The key idea is we prefer the least k in the case that $d\%$ is bigger the threshold value. The k-NN method gathers the nearest k neighbors and let them vote the class of most neighbors wins. Theoretically, the more neighbors we consider, the smaller error rate it takes place.

In a word, KNN is a simple and powerful tool which becomes an important step in our methodology.

3.2 Ensemble modeling

No guideline is always correct. No single method is always the best. This has lead to the idea of trying to combine models into an ensemble rather than selecting among them. The idea seems to work well as demonstrated by many practical classification applications [11].

The ensemble error can be calculated between $y_{Ensemble}$ and y, where, $y_{Ensemble} = \sum_{i=1}^{n} \omega_i \hat{y}_i$ is the weighted sum of the output of each i^{th} individual models, ω_i is the weighted assigned to the output of the i_{th} model; these weighted satisfy $\sum_i \omega_i = 1$; y is the real output of the data and $\hat{y}_i$ is ensemble target.

In this paper, we use the idea of ensemble modeling in two steps to obtain a more accurate output than the results of single predictors. Thus, $\hat{y}_i$ can be either y_{KNN} or Y_{LOO} which is also explained in next paragraph.

3.3 Nonnegative Least Squares (NNLS) Problem

NNLS problem can be as simple model as to minimize:

$$\| A\omega - b \|, \text{ subject to } \omega \geq 0$$

where A is a $m \times K$ matrix, $m \geq K$, b is the m element data vector, and ω is the K element solution vector.

To solve this problem, Charles L. Lawson and Richard J. Hanson stated 11-steps NNLS algorithm [9], as well as the finite convergence of the algorithm which was proved in their book.

Recalling the ensemble modeling problems which mentioned in the above section, the ensemble weights (ω_i) can be calculated using the NNLS algorithm (Charles L. Lawson and Richard J. Hanson). As described in section 2, ensemble modeling is used twice in the method. For the first time happened in KNN step, the problem becomes: $\| y_{KNN}\omega - y_{real} \|$ subject to all $\omega \geq 0$. As to the second time in Step ensemble, the problem turns to $\| y_{LOO}\omega' - y_{real} \|$ subject to all $\omega' \geq 0$.

It should be also noted that NNSL algorithm can helps to avoid overfiting of the model [9].

3.4 Leave-One-Out (LOO)

LOO is an special case of cross-validation. In this paper, LOO method is used in the training set to get even better prediction. One problem with the LOO error is that it can get very time consuming if the dataset tends to have a high number of samples. Fortunately, the PRESS (or PREdiction Sum of Squares) statistics provide a direct and exact formula for the calculation of the LOO error for linear models. See [4, 5] for details on this formula and implementations:

$$\epsilon^{\text{PRESS}} = \frac{y_i - \hat{y}_i\omega}{1 - \hat{y}_i\mathbf{P}\hat{y}_i^T}, \quad (1)$$

where $\mathbf{P}$ is defined as $\mathbf{P} = (\hat{Y}^T\hat{Y})^{-1}$ and $\hat{Y}$ is the estimated output matrix, and ω is the ensemble weight for each model.

4 Experiments

4.1 Sine in one dimension

In this experiments, a set of 1000 training points are generated (and represented in Fig. 2), the output is a sum of two sines. This single dimension example is used to test the method without the final repetition step . The model built is also shown in this figure using red dots. This model approximates the dataset accurately, using 20 nearest neighbors; and it reaches a LOO error close to the noise introduced in the dataset which is 0.0625. The computational time for the whole method is one second (using Matlab© implementation).

4.2 Bankruptcy Prediction task

In this experiment, we use the data related to 500 French companies in 2002 and 520 companies in 2003. 41 input variables on these samples are finally used without any missing value, the input variables are financial indicators that are measured during the year and the output variable is the classification result which 1 represents healthy and 2 means bankruptcy.

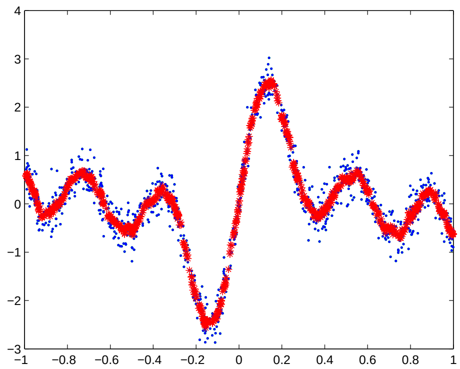

Figure 2: Sine Toy example

Table 1 shows the real meaning in financial field about all the input variables we have used.

The data are splited half in training set and half in testing set, for both year 2002 and year 2003. The results are shown in the Tables 2 and 4.2.

For the results in tables, we chose to use 30 nearest neighbors, and build 400 KNN models with different subsets of input variables. After we get the ensemble result from the 400 models, 200 times of the whole process are repeated. The mean of these repetition results are listed in the tables. The training error is in fact the LOO error and testing set is normalized using the mean and standard deviation of the training variables. The computation time of one repetition is about 15 seconds.

From these errors, we can see this methodology not only build the model in a simple and fast way, but also prove the accuracy of the algorithm. It should also be noted that on the experiments of this financial data, the method also helps to analyze financial ratios using the randomly selection step. The subset corresponding to the lowest error may be more important than others.

5 Conclusions

In this paper, we proposed a methodology Ensemble KNNs based on K-Nearest Neighbors which gives better performance than existing OP-KNN [10] or any other algorithms for the financial modeling we have tested. The technique of ensemble modeling used in two steps in the methodology because we believe an appropriate combination must be better than any one of single predictors.

We test our methodology on about 500 French industrial firms for year 2002 and 2003. Our results highlight that the financial situation in 2002 is easier to predict than 2003, even though both results for these years are satisfied. On the other hand, the method remains high quality performance while being extremely fast.

Besides, the choice of the parameter like K or N will influence the results in some

Table 1: The meaning of input variables

index	Variable	Meaning
1	RE1	Profit before Tax /Shareholders' Funds
2	RE2	Net Income/Shareholders' Funds
3	RE3	EBITDA/Total Assets
4	RE4	EBITDA/Permanent Assets
5	RE5	EBIT/Total Assets
6	RE6	Net Income/Total Assets
7	EF1	Value Added/Total Sales
8	EF2	Total Sales/Shareholders' Funds
9	EF3	EBIT/Total Sales
10	EF4	Total Sales/Total Assets
11	EF5	Gross Trading Profit/Total Sales
12	EF7	Operating Cash Flow/Total Assets
13	EF8	Operating Cash Flow/Total Sales
14	PR1	Financial Expenses/Total Sales
15	PR2	Labor Expenses/Total Sales
16	SF1	Shareholder's Funds/Total Assets
17	SF2	Total Debt/Shareholders' Funds
18	SF7	Total Debt/Total Assets
19	SF8	Net Operating Working Capital/Total Assets
20	SF10	Long Term Debt/Total Assets
21	SF11	Long Term Debt/Shareholders' Funds
22	LI1	(Cash + Marketable Securities)/Total Assets
23	LI2	Cash/Total Assets
24	LI3	(Cash + Marketable Securities)/Total Sales
25	LI4	Quick Ratio
26	LI5	Cash/Current Liabilities
27	LI6	Current Assets/Current Liabilities
28	LI7	Quick Assets/Total Assets
29	LI8	Current Liabilities/Total Assets
30	LI9	Quick Assets/Total Assets
31	LI10	EBITDA/Total Sales
32	LI11	Financial Debt/Cash Flow
33	LI12	Cash/Total Debt
34	RO1	Cash/Total Sales
35	RO2	Inventory/Total Sales
36	RO3	Net Operating Working Capital/Total Sales
37	RO4	Accounts Receivable/Total Sales
38	RO5	Accounts Payable/Total Sales
39	RO6	Current Assets/Total Sales
40	AP1	Change in Equity Position
41	AP2	Change in Other Debts

Table 2: Result for data 2002

k=30, N=400	Repetition=200
training error	0.94
testing error	0.92

Table 3: Result for data 2003

k=30, N=400	Repetition=200
training error	0.82
testing error	0.83

extent. That highlights another advantage of this method: the performance are not sensitive with the variance of the parameters.

References

[1] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani (2004), Least angle regression, *Annals of Statistics*, **vol. 32** p. 407-499.

[2] A. J. Jones (2004), New Tools in Non-linear Modeling and Prediction, *Computational Management Science*, p.109–149

[3] Yu Jiangsheng (2002), Method of k-Nearest Neighbors, Institute of Computational Linguistics Peking University, China

[4] R.H. Myers, (1990), Classical and Modern Regression with Applications, Duxbury, Pacific Grove, CA, USA

[5] G. Bontempi, M. Birattari, H. Bersini (1998), Recursive lazy learning for modeling and control, *European Conference on Machine learning*, p. 292-303.

[6] W.T. Miller, F.H. Glanz, L.G. Kraft (1990), Cmas: An associative neural network alternative to backpropagation, *Proceeding of the IEEE*, **vol. 70**97-102 p. 1561-1567.

[7] C.R. Rao, S.K. Mitra (1972), Generalized Inverse of Matrices and its Applications, *John Wiley & Sons Inc*,

[8] T. Similä, J. Tikka (2005), Multiresponse sparse regression with application to multidimensional scaling, *Proceedings of the 15th International Conference on Artificial Neural Networks*, **Part II** p. 97-102.

[9] Charles L. Lawson and Richard J. Hanson (1974), Solving Least Squares Problems, Prentice Hall, Englewood Cliffs NJ.

[10] Q. Yu, Antti Sorjamaa, Yoan Miche, E. Séverin, A. Lendasse (2008), Optimal Pruned K-Nearest Neighbors: OP-KNN – Application to Financial Modeling, *Computational Methods for Modeling and Learning in Social and Human Science 2008*, Créteil, France.

[11] S. V. Barai and Y. Reich (1999), Ensemble modelling or selecting the best model: Many could be better than one, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Cambridge University Press, P.377-386.

[12] E. Ocelikova, D. Klimesova (2005), Bayes Classifier in Multidimensional Data Classification, *15th Int. Conference Process Control*, June, 2005, Slovakia.

[13] A. Lendasse, V. Wertz, M. Verleysen (2003), Model Selection with Cross-Validations and Bootstraps - Application to Time Series Prediction with RBFN Models, *Joint International Conference on Artificial Neural Networks*, Istanbul, Turkey.

[14] M. Verleysen (2001), Learning high-dimensional data, *NATO Advanced Research Workshop on Limitations and Future Trends in Neural Computing*, Italy, p. 22-24.

[15] B. Efron and R. Tibshirani (1993), New Tools in Non-linear Modeling and Prediction, *Chapman and Hall*, London.