Analysis of Credit Risk Prediction Using ARSkNN

Ashish Kumar[1](✉), Roheet Bhatnagar[1], and Sumit Srivastava[2]

[1] Department of Computer Science and Engineering,
Manipal University Jaipur, Jaipur, Rajasthan, India
aishshub@gmail.com, roheet.bhatnagar@jaipur.manipal.edu
[2] Department of Information Technology,
Manipal University Jaipur, Jaipur, Rajasthan, India
sumit.srivastava@jaipur.manipal.edu

Abstract. Credit risk is characterized as the risk that borrowers will neglect to pay its advance commitments and loan obligations. It is very hard to predict the outcomes (risky borrower) manually as the evaluation of large features set is quite time consuming. That's why, we need some good predictor as classifier. The traditional k-NN is one pre-established classifier used in various domains along with credit risk predictions. The newly conceptualized ARSkNN is another such classification which reduces the runtime in predicting the outcomes and improves overall accuracy percentage of the predicted classes over Traditional k-NN. The method adopt the similarity measure which is based on the Mass estimation rather than distance estimation for predicting the K- nearest neighbor. The results were compared using WEKA 3.7.10 as tool and found significant improvement vis-á-vis the evaluation parameters by the ARSkNN method.

Keywords: Classification · Nearest neighbors · ARSkNN · Credit risk

1 Introduction

Credit will happen as a result of plentiful reasons: house loans or bank mortgages, automobile purchase, credit card purchases, and so on. Basel Committee on Banking Supervision defined credit risk as the potential of a counterparty or bank debtor will be unsuccessful to pay its debts in accord with pre-established terms [1]. This credit risk analysis is vital to monetary establishments which offer loans to businesses and people. In recent years, Indian banks have seen a massive increase in their credit card customers. According to the latest available established data from the Reserve Bank of India (RBI), the growing rate of credit cards in India is nearly 24%. Also till March 2016, more than twenty four million credit cards are issued to their customers by all banks in India. This increases the credit card loan risk of being defaulted. Credit provider banks regularly collect immense volume of data about borrowers to fathom risk levels of credit borrowers. This data

A. E. Hassanien et al. (Eds.): AMLTA 2018, AISC 723, pp. 644–652, 2018.
https://doi.org/10.1007/978-3-319-74690-6_63

has additionally been utilized with analytical predictive methods to evaluate or to determine risky and unsafe clients associated in credits and loans.

The possibility that a credit card aspirant will default must be estimated from information about the aspirant provided at the time of the application, and the estimate will assist as the basis for accepting or rejecting his application. While classifying, monetary background and subjective aspects of credit borrowers are assessed. Among these, monetary ratios perform an vital role for risk level estimation [2]. The implementation of Basel Committee's principle turns out to be a daily decision based on a binary classification problem distinguishing good payers from bad payers [3]. The first researches on credit scoring were done by [4,5] who applied linear and quadratic discriminant analysis respectively to categorize credit applications as "good" or "bad" ones. Since precise classification is of advantage both to the creditor and to the aspirant, many statistical methods, including multivariate discriminant analysis [6], logistic regression [7], and nearest neighbor [8], have been used to develop models of risk prediction. With the evolution of artificial intelligence and machine learning, artificial neural networks [3,9,10] and classification trees [11], were also employed to forecast credit risk. According to [12], "Despite the intense study of credit scoring, there is no consensus on the most appropriate classification technique to use."

This paper tackles the following question: How to predict good and bad borrowers more accurately so that banks can reduce credit risk? At first the authors explore the traditional nearest neighbor techniques to predict the defaulter and then further establishes ARSkNN a new nearest neighbor classification technique, for analyzing the credit risk.

The paper is divided into different sections and in the following sections, authors review the pre-established kNN classifier models in the domain of credit risk. Section 2 provides an introduction to credit risk. Nearest Neighbor classifier and ARSkNN are described in Sects. 3 and 4 respectively. Evaluation Parameters used to evaluate these classifiers are discussed in Sect. 5. Description of the dataset used is mentioned in Sect. 6. Results and Discussions forms Sect. 7. Conclusion and Future Work is given in Sect. 8.

2 Credit Risk - An Introduction

Banks should target on three kinds of risk: Credit, Operational and Market [13]. Among these, credit risk is one of the biggest risk faced by most the banks. Credit Risk can be defined as a loss in sense of money, a bank would undergo, if a bank's debtor is unsuccessful to fulfill his commitments viz; partially or fully pay interest money on borrowed loan, partially or fully refund the amount borrowed in line with the concurred terms and conditions [14].

Credit Risks are premeditated based on the debtors' complete capability to pay back. To evaluate credit risk on a customer loan, creditors investigate the five C's namely: the candidate's **credit history**, his refund **capability**, his **capital**, the **credit's conditions** and related **collateral**.

Credit Risk is frequently characterized by three factors: loss risk, default risk, and exposure risk. Default and credit risk are generally synonymous. Credit

Risk management is a technique involving following steps – recognition of possible risks, the assessment of these risks, the relevant treatment, and at last the employment of risk models [15].

Assessment of Credit Risk is an important activity to avoid immense amount of losses for any financial institution. The financial institution follows a robust framework to successfully diminish and anticipate credit risks [16]. Thus in a nutshell, the comprehensive objective of credit risk assessment is to equate the features of a going to be debtor with other previous debtors, whose loans they have already deposited back.

3 Nearest Neighbor Classifier

The classification methods can be broadly classified into parametric and nonparametric problems. In fact, parametric methods are based upon the assumptions of normally distributed population and estimate the parameters of the distributions to solve the problem [17]. However, according to Berry and Linoff [18] nonparametric methods make no assumptions about the specific distributions involved, and are therefore distribution-free. The k-nearest neighbor classifier serves as an illustration of a non-parametric statistical approach.

The cornerstones of k-nearest neighbor classification [19] are Nearest Neighbor (NN) classifier and the k-NN rule proposed by Fix and Hodges in 1951. It is also acknowledged by names such as instance based classification, memory based classification, case based classification and much more. There are three key building blocks of a k-NN classifier: a set of class labeled instances; a dissimilarity (e.g. Euclidean Distance) or similarity metric to work out distance or similarity among two instances; and the value of k i.e., the number of nearest neighbors to be considered.

According to Berry and Linoff [18] "the choice of k also affects the performance of the k-NN algorithm. This can be determined experimentally. Starting with $k = 1$, we use a test case to estimate the error rate of the classifier. This process is repeated each time by incrementing k to allow for one more neighbors. The K-value that gives the minimum error rate may be selected. In general, larger the number of training samples is, the larger the value of k will be." Various metrics have been suggested to enhance the k-NN classifiers for example, Mahalanobis distance [20], adaptive distance [21] and local metric [22]. Moreover K-NN classifier requires an equal number of good and bad sample cases for better performance [8].

4 ARSkNN - A Novel Mass Based Classifier

ARSkNN, which is conceptualized by the same authors [23], is an efficient k-nearest neighbor classifier exploits Massim, a mass-based similarity measure in spite of utilizing any distance-based similarity measures.

ARSkNN has got two stages: 1. Modeling Stage and 2. Class Assignment Stage. In modeling (preprocessing) stage, a Similarity Forest (sForest) with t number of Similarity Trees (sTrees), is built from D dataset which has $(x_1, c_1), (x_2, c_2), ..., (x_n, c_n)$ without consideration of c_i. After this in class assignment stage, ARSkNN is used to find the k-nearest neighbors (instances) in D with respect to a query instance.

For a query instance y, $Massim^h(x, y)$ is estimated for all instances x in dataset D. For this estimation, x and y are parsed through t similarity trees (sTrees) of the similarity forest (sForest) After this in class assignment stage, ARSkNN is used to find the k-nearest neighbors (instances) in D with respect to a query instance.

Algorithm 1. ARSkNN

```
Input: $y$ — Query instance, $D$ — Dataset which has
        $\{(x_1, c_1), (x_2, c_2), ..., (x_n, c_n)\}$, $k$ — number of nearest neighbors
Output: $c_y$ — Class of query instance y
Let $A \leftarrow \{\}$;
for each x in D do
    $Massim \leftarrow Massim(x_i, y, F, e)$ ;
    $A \leftarrow A \cup \{x_i, c_i, Massim\}$;
end
Sort in ascending order, the pairs in $A$ using the third components;
$c_y \leftarrow$ the most frequent class in [Select the first $k$ instances from $A$];
return $c_y$
```

5 Performance Evaluation

The works of [24,25] reveals that error rates were often used as the measurement of classification accuracy of models. However, most records in the data set of credit card customers are non-risky (87.88%); therefore, the error rate is insensitive to classification accuracy of models.

It also varies from application to application in which classification technique is used. For the binary classification problem, some researchers have been used accuracy percentage for comparing the performance of different models than the error rate [26,27].

We have also demonstrated the average runtime of both classification techniques in seconds to justify that ARSkNN is very much faster than the traditional k-NN classifier, which uses a distance based similarity measure.

As every time with Java code, to get judiciously precise measurements, we are needed to exercise the significant code a few times before considering any measurement, so that the JIT has essentially compiled the code. To fulfil this purpose, we run each experiment 10 times and all the experiments was done with 10-fold cross-validation technique to evaluate both the classification techniques. In k-fold cross-validation, the original sample is arbitrarily subdivided into k equal size subsamples. Then of the k subsamples, a particular subsample is reserved as the validation data for analysis the model, and the left over k-1 subsamples are used as training data. The cross-validation method is then reiterated k times (folds), with each one of the k subsamples used precisely one time as the validation data. The k outcomes from the folds can then be averaged to calculate a single estimation. The benefit of this method is that all are used for both validation and training, and each interpretations is used for validation precisely one time.

5.1 Accuracy Percentage

In binary classification, accuracy is statistical measure which tells about how well a binary classifier correctly classifies the instances. According to ISO 5725-1, (Reference BS ISO 5725-1) the overall term "accuracy" is used to define the nearness of a quantity to the true value. It is the number of correct predictions made divided by the total number of predictions made, multiplied by 100 to turn it into a percentage i.e.

$$AccuracyPercentage = ((TP + TN)/All) * 100 \tag{1}$$

where, TP = True Positive; TN = True Negative and All = total number of instances.

5.2 Average Runtime

The analysis of algorithms is the assurance of the quantity of resources (such as storage and time) required to execute them. To evaluate the time complexity of any algorithm, calculation of runtime of that algorithm has to be done. In weka, the average runtime of classification technique can be calculated using the Experimenter module. One can use UserCPU_Time_training (in seconds) and UserCPU_Time_testing (in seconds) fields to output the average time for the classifiers in the experiment.

6 Description of the Dataset

The data had been collected from a bank in Taiwan which provides credit cards to its customers. The default of credit card clients data set is made of 30000 instances and 24 attributes along with class attribute, which is a binary variable (Yes = 1, No = 0). Among these 30000 instances, 6636 instances are the card owners with default payments.

In the very first study done on this dataset has shown comparative study of six data mining techniques (k-nearest neighbor, logistic regression, discriminant analysis, naive bayes classifier, artificial neural networks, and classification trees).

The description of 23 explanatory attributes are as follows:

ATT1: Credit Amount (in NT dollars).

ATT2: Sex (1 = male; 2 = female).

ATT3: Education (1 = grad. school; 2 = university; 3 = high school; 4 = others).

ATT4: Marital status (1 = married; 2 = single; 3 = others).

ATT5: Age (in years).

ATT6 - ATT11: History of past 6 payment. The measuring scale for the repayment status is: -1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

ATT12 - ATT17: Amount of bill statement (in NT dollars).

ATT18 - ATT23: Amount of previous payment (in NT dollars).

7 Evaluation

This section presents the experimental results obtained by evaluating the performance of the ARSkNN classification technique against the traditional kNN classification technique. The whole classification experiment has been carried out by using a machine with an Intel Core i7 processor with 2.4 GHz speed and 8 GB RAM. The experiments were done using experimenter module of Weka 3.7.10 for both the classifiers. Traditional kNN classifier is already implemented in Weka with the name of IBK. In this experiment, IBK is used with LinearNNSearch as a nearest neighbor search algorithm with Euclidean distance as similarity measure. ARSkNN is implemented using Java Development Kit 1.8.0 with Netbeans 8.0.2 as the preferred IDE. The jar file titled ARSkNN.jar has been combined as a runtime module with Weka 3.7.10 platform.

Table 1 shows the obtained results in term of average accuracy percentage for IBK and ARSkNN (with 10, 50, and 100 sTrees) over 10-fold cross validation for different values of k which are 1, 3, 5 and 10. For the value of k as 1, ARSkNN has 8.22% gain in average accuracy in comparision to IBK, which is a huge gain in the classification domain. The same variation in results have been seen with the values of k as 3, 5 and 10.

The overall conclusion that can be drawn from Table 1 is, ARSkNN gives better average classification accuracy for every value of k. Figure 1 shows the same in the graphical form.

Table 1. Average accuracy (in percentage)

	k = 1	k = 3	k = 5	k = 10
IBK	72.97	77.69	79.33	80.76
ARSkNN with 10 sTrees	78.70	80.69	80.88	80.90
ARSkNN with 50 sTrees	80.98	81.38	81.33	81.14
ARSkNN with 100 sTrees	**81.19**	**81.44**	**81.39**	**81.25**

Table 2 shows the obtained results in term of average runtime in seconds for IBK and ARSkNN with 10, 50, and 100 sTrees over 10-fold cross validation for different values of k which are 1, 3, 5 and 10. For the value of k as 1, even ARSkNN with 100 trees has taken 6.81 seconds lesser than IBK. As we increase the value of k, this difference becomes even more prominent.

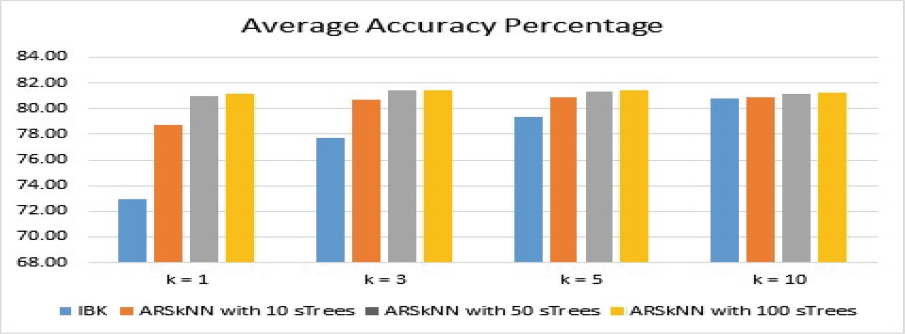

Fig. 1. Average Accuracy (in percentage) of classifiers.

Table 2. Average runtime (in seconds)

	k = 1	k = 3	k = 5	k = 10
IBK	7.80	9.71	10.41	10.30
ARSkNN with 10 sTrees	0.08	0.08	0.08	0.08
ARSkNN with 50 sTrees	0.46	0.52	0.56	0.72
ARSkNN with 100 sTrees	0.99	1.07	1.04	1.06

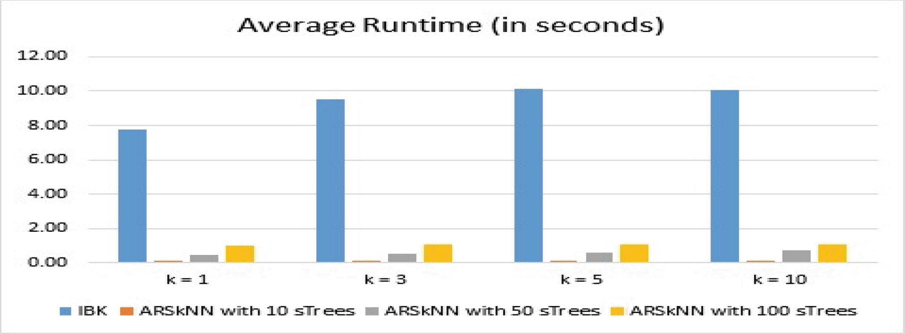

Fig. 2. Average Runtime (in seconds) of classifiers.

The overall conclusion that can be drawn from Table 2 is, ARSkNN gives significantly better average runtime for every value of k and it has also been shown in graphical form in Fig. 2.

8 Discussion

The research work assessed that ARSkNN is expressively improved than the IBK (with Euclidean distance) upon two parameters, i.e. accuracy percentage and average runtime. The accuracy percentage of IBK significantly depends upon the similarity measure used as learning metric.

ARSkNN is taking very less average runtime than the IBK because it computes the similarity measures during the modelling stage of sForest however IBK computes the similarity between each training instance and the testing instance which is an overhead for IBK. Also due to this computation overhead, IBK needs all the training instances in the memory, whereas ARSkNN does not needs any training instance in memory, which affects overall accuracy in the computed results at the end of the simulation.

After finding the k-nearest neighbors, both classifiers has to use the voting technique to decide the class of testing instance. The core difference between these classifiers can be seen in terms of the similarity measures via distance calculation in traditional kNN (IBK) and the mass based estimation in ARSkNN.

9 Conclusion

In this paper, we established ARSkNN classifier which uses similarity measure based upon mass estimation technique and demonstrate its effectiveness for the credit card risk analysis. The method was compared with traditional kNN technique on the credit data set, which shown the significant results in terms of the average accuracy percentage and average runtime. The modelling method was the major concerns as communicated in the paper. For the ARSkNN, the similarity Forest developed during the training phase is adopted to identify the similarity on the testing data set, which further acts as a class identifier during the voting phase in k-Nearest Neighbor measure. However, in case of traditional kNN the complete training set is used to calculate the distance metric during the testing phase for identifying k-Nearest Neighbor, which further voted for the class identification.

There are potential extensions of the current work. First, one can compare ARSkNN with kNN using different similarity metrics rather than Euclidean distance. Second, the application of ARSkNN should also be judged in various different domains.

References

1. Safakli, O.V.: Credit risk assessment for the banking sector of northern cyprus. Banks Syst. **2**(1), 21 (2007)
2. Bekiroglu, B., Takci, H., Ekinci, U.C.: Bank credit risk analysis with Bayesian network decision tool. IJAEST Int. J. Adv. Eng. Sci. Technol. **1**(9), 273–279
3. Karaa, A., Krichene, A.: Credit-risk assessment using support vectors machine and multilayer neural network models: a comparative study case of a tunisian bank. Account. Manag. Inf. Syst. **11**(4), 587 (2012)
4. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics **7**(2), 179–188 (1936)
5. Durand, D., et al.: Risk Elements in Consumer Instalment Financing. NBER Books (1941)

6. Altman, E.I.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. J. Finance **23**(4), 589–609 (1968)
7. Steenackers, A., Goovaerts, M.J.: A credit scoring model for personal loans. Insur. Math. Econ. **8**(1), 31–34 (1989)
8. Hand, D.J., Henley, W.E.: Statistical classification methods in consumer credit scoring: a review. J. Roy. Stat. Soc. Ser. A (Stat. Soc.) **160**(3), 523–541 (1997)
9. Desai, V.S., Crook, J.N., Overstreet, G.A.: A comparison of neural networks and linear scoring models in the credit union environment. Eur. J. Oper. Res. **95**(1), 24–37 (1996)
10. Matoussi, H., Abdelmoula, A., et al.: Using a neural network-based methodology for credit-risk evaluation of a Tunisian bank. Middle East. Finance Econ. **4**, 117–140 (2009)
11. Davis, R.H., Edelman, D., Gammerman, A.: Machine-learning algorithms for credit-card applications. IMA J. Manag. Math. **4**(1), 43–51 (1992)
12. Miguéis, V.L., Benoit, D.F., Van den Poel, D.: Enhanced decision support in credit scoring using Bayesian binary quantile regression. J. Oper. Res. Soc. **64**(9), 1374–1383 (2013)
13. Foust, D., Pressman, A.: Credit Scores: Not-so-Magic Numbers. Business Week 7 (2008)
14. Went, P., Apostolik, R., Donohue, C.: Foundations of banking risk: an overview of banking, banking risks, and risk-based banking regulation. Wiley, Hoboken (2009)
15. Van Gestel, T., Baesens, B.: Credit Risk Management: basic concepts: financial risk components, rating analysis, models, economic and regulatory capital. Oxford University Press, UK (2009)
16. Guo, Y., WU, C.: Research on credit risk assessment in commercial bank based on information integration. In: Proceedings of 2009 International Conference on Management Science and Engineering (2009)
17. Zhang, D., Huang, H., Chen, Q., Jiang, Y.: A comparison study of credit scoring models. In: Third International Conference on Natural Computation, ICNC 2007, vol. 1, pp. 15–18. IEEE (2007)
18. Berry, M.J., Linoff, G.: Data Mining Techniques: For Marketing, Sales, and Customer Support. Wiley, Hoboken (1997)
19. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)
20. Weinberger, K.Q., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res. **10**, 207–244 (2009)
21. Wang, J., Neskovic, P., Cooper, L.N.: Improving nearest neighbor rule with a simple adaptive distance measure. Pattern Recogn. Lett. **28**(2), 207–213 (2007)
22. Noh, Y.K., Zhang, B.T., Lee, D.D.: Generative local metric learning for nearest neighbor classification. IEEE Trans. Pattern Anal. Mach. Intell. **401**, 106–118 (2017)
23. Kumar, A., Bhatnagar, R., Srivastava, S.: ARSkNN-A k-NN classifier using mass based similarity measure. Procedia Comput. Sci. **46**, 457–462 (2015)
24. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
25. Nelson, B.J., Runger, G.C., Si, J.: An error rate comparison of classification methods with continuous explanatory variables. IIE Trans. **35**(6), 557–566 (2003)
26. Jiawei, H., Kamber, M.: Data mining: concepts and techniques. Morgan Kaufmann, San Francisco (2001)
27. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, Burlington (2016)