Credit Risk Evaluation Using Support Vector Machine with Mixture of Kernel

Liwei Wei^{1,2}, Jianping Li¹, and Zhenyu Chen^{1,2}

¹ Institute of Policy & Management, Chinese Academy of Sciences, Beijing 100080, China

² Graduate University of Chinese Academy of Sciences, Beijing 100039, China
{lwwei, ljpp, zychen}@casipm.ac.cn

Abstract. Recent studies have revealed that emerging modern machine learning techniques are advantageous to statistical models for credit risk evaluation, such as SVM. In this study, we discuss the applications of the support vector machine with mixture of kernel to design a credit evaluation system, which can discriminate good creditors from bad ones. Differing from the standard SVM, the SVM-MK uses the 1-norm based object function and adopts the convex combinations of single feature basic kernels. Only a linear programming problem needs to be resolved and it greatly reduces the computational costs. More important, it is a transparent model and the optimal feature subset can be obtained automatically. A real life credit dataset from a US commercial bank is used to demonstrate the good performance of the SVM- MK.

Keywords: Credit risk evaluation SVM-MK Feature selection Classification model.

1 Introduction

Undoubtedly credit risk evaluation is an important field in the financial risk management. Extant evidence shows that in the past two decades bankruptcies and defaults have occurred at a higher rate than any other time. Thus, it's a crucial ability to accurately assess the existing risk and discriminate good applicants from bad ones for financial institutions, especially for any credit-granting institution, such as commercial banks and certain retailers. Due to this situation, many credit classification models have been developed to predict default accurately and some interesting results have been obtained.

These credit classification models apply a classification technique on similar data of previous customers to estimate the corresponding risk rate so that the customers can be classified as normal or default. Some researchers used the statistical models, such as Linear discriminate analysis [1], logistic analysis [2] and probit regression [3], in the credit risk evaluation. These models have been criticized for lack of classification precision because the covariance matrices of the good and bad credit classes are not likely to be equal.

Recently, with the emergence of Decision tree [4] and Neural network [5], artificial intelligent (AI) techniques are wildly applied to credit scoring tasks. They have

obtained promising results and outperformed the traditional statistics. But these methods often suffer local minima and over fitting problems.

Support vector machine (SVM) is first proposed by Vapnik [6]. Now it has been proved to be a powerful and promising data classification and function estimation tool. Reference [7], [8] and [9] applied SVM to credit analysis. They have obtained some valuable results. But SVM is sensitive to outliers and noises in the training sample and has limited interpretability due to its kernel theory. Another problem is that SVM has a high computational complexity because of the solving of large scale quadratic programming in parameter iterative learning procedure.

Recently, how to learn the kernel from data draws many researchers' attention. The reference [10] draws the conclusion that the optimal kernel can always be obtained as a convex combinations of many finitely basic kernels. And some formulations [11] [12] have been proposed to perform the optimization in manner of convex combinations of basic kernels.

Motivated by above questions and ideas, we propose a new method named support vector machines with mixture of kernel (SVM-MK) to evaluate the credit risk. In this method the kernel is a convex combination of many finitely basic kernels. Each basic kernel has a kernel coefficient and is provided with a single feature. The 1-norm is utilized in SVM-MK. As a result, its objective function turns into a linear programming parameter iterative learning procedure and greatly reduces the computational complexity. Furthermore, we can select the optimal feature subset automatically and get an interpretable model.

The rest of this paper is organized as follows: section 2 gives a brief outline of SVM-MK. To evaluate the performance of SVM-MK for the credit risk assessment, we use a real life credit card data from a major US commercial bank in this test in section 3. Finally, section 4 draws the conclusion and gives an outlook of possible future research areas.

2 Support Vector Machine with Mixture of Kernel

Considering a training data set $G = \{(\bar{x}_i, y_i)\}_{i=1}^n$, $\bar{x}_i \in R^m$ is the i^{th} input pattern and y_i is its corresponding observed result $y_i \in \{+1, -1\}$. In credit risk evaluation model, x_i denotes the attributes of applicants or creditors, y_i is the observed result of timely repayment. If the applicant defaults a loan, $y_i = 1$, else $y_i = -1$.

The optimal separating hyper-plane is found by solving the following regularized optimization problem [6]:

$$\min J(\bar{\omega}, \xi) = \frac{1}{2} \|\omega\|^2 + c \sum_{i=1}^n \xi_i \quad (1)$$

$$s.t. \begin{cases} y_i (\bar{\omega}^T \phi(\bar{x}_i) + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \end{cases} \quad i = 1, \dots, n \quad (2)$$

where c is a constant denoting a trade-off between the margin and the sum of total errors. $\phi(\bar{x})$ is a nonlinear function that maps the input space into a higher dimensional feature space. The margin between the two parts is $2/\|\omega\|$.

The quadratic optimization problem can be solved by transforming (1) and (2) into the saddle point of the Lagrange dual function:

$$\max \left\{ \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j k(\bar{x}_i, \bar{x}_j) \right\} \quad (3)$$

$$s.t. \quad \begin{cases} \sum_{i=1}^n y_i \alpha_i = 0 \\ 0 \leq \alpha_i \leq c, i = 1, \dots, n \end{cases} \quad (4)$$

where $k(\bar{x}_i, \bar{x}_j) = \phi(\bar{x}_i) \bullet \phi(\bar{x}_j)$ is called the kernel function, α_i are the Lagrange multipliers.

In practice, a simple and efficient method is that the kernel function being illustrated as the convex of combinations of the basic kernel:

$$k(\bar{x}_i, \bar{x}_j) = \sum_{d=1}^n \beta_d k(x_{i,d}, x_{j,d}) \quad (5)$$

where $x_{i,d}$ denotes the d^{th} component of the input vector $\bar{x}_i$.

Substituting Equation (5) into Equation (3), and multiplying Equation (3) and (4) by β_d , suppose $\gamma_{i,d} = \alpha_i \cdot \beta_d$, then the Lagrange dual problem change into:

$$\max \left\{ \sum_{i,d} \gamma_{i,d} - \frac{1}{2} \sum_{i,j,d=1}^{n,m} \gamma_{i,d} \gamma_{j,d} y_i y_j k(x_{i,d}, x_{j,d}) \right\} \quad (6)$$

$$s.t. \quad \begin{cases} \sum_{i,d} y_i \gamma_{i,d} = 0 \\ 0 \leq \sum_{d=1}^m \gamma_{i,d} \leq c, i = 1, \dots, n \\ \gamma_{i,d} \geq 0, d = 1, \dots, m \end{cases} \quad (7)$$

The new coefficient $\gamma_{i,d}$ replaces the Lagrange coefficient α_i . The number of coefficient that needs to be optimized is increased from n to $n \times m$. It increases the computational cost especially when the number of the attributes in the dataset is large. The linear programming implementation of SVM is a promising approach to reduce the computational cost of SVM and attracts some scholars' attention. Based on above idea, a 1-norm based linear programming is proposed:

$$\min J(\bar{\gamma}, \bar{\xi}) = \sum_{i,d} \gamma_{i,d} + \lambda \sum_{i=1}^n \xi_i \quad (8)$$

$$s.t. \quad \begin{cases} y_i \left(\sum_{j,d} \gamma_{j,d} y_j k(x_{i,d}, x_{j,d}) + b \right) \geq 1 - \xi_i \\ \xi_i \geq 0, i = 1, \dots, n \\ \gamma_{i,d} \geq 0, d = 1, \dots, m \end{cases} \quad (9)$$

In equation (8), the regularized parameter λ controls the sparse of the coefficient $\gamma_{i,d}$.

The dual of this linear programming is:

$$\begin{aligned} & \max \sum_{i=1}^n u_i \quad (10) \\ s.t. \quad & \begin{cases} \sum_{i=1}^n u_i y_i y_j k(x_{i,d}, x_{j,d}) \leq 1, j = 1, \dots, n, d = 1, \dots, m \\ \sum_{i=1}^n u_i y_i = 0 \\ 0 \leq u_i \leq \lambda, i = 1, \dots, n \end{cases} \quad (11) \end{aligned}$$

The choice of kernel function includes the linear kernel, polynomial kernel or RBF kernel. Thus, the SVM-MK classifier can be represented as:

$$f(\bar{x}) = \text{sign} \left(\sum_{j,d} \gamma_{j,d} y_j k(x_{i,d}, x_{j,d}) + b \right) \quad (12)$$

It can be found that above linear programming formulation and its dual description is equivalent to that of the approach called “mixture of kernel” [12]. So the new coefficient $\gamma_{i,d}$ is called the mixture coefficient. Thus this approach is named “support vector machine with mixture of kernel” (SVM-MK). Comparing with the standard SVM that obtains the solution of α_i by solving a quadratic programming problem, the SVM-MK can obtain the value of mixture coefficient $\gamma_{i,d}$ by solving a linear programming. So the SVM-MK model greatly reduces the computational complexity. It is more important that the sparse coefficients $\gamma_{i,d}$ give us more choices to extract the satisfied features in the whole space spanned by all the attributes.

3 Experiment Analysis

In this section, a real-world credit dataset is used to test the performance of SVM-MK. The dataset is from a major US commercial bank. It includes detailed information of 5000 applicants, and two classes are defined: good and bad creditor. Each record consists of 65 variables, such as payment history, transaction, opened account etc. This dataset includes 5000 applicants, in which the number of bad accounts is 815 and the others are good accounts. Thus the dataset is greatly imbalance. So we pre-process the data by means of sampling method and making use of 5-fold cross-validation to guarantee valid results. In addition, three evaluation criteria measure the efficiency of classification:

$$\text{Type I error} = \frac{\text{number of observed good but classified as bad}}{\text{number of observed good}} \quad (13)$$

$$\text{Type II error} = \frac{\text{number of observed bad but classified as good}}{\text{number of observed bad}} \quad (14)$$

$$\text{Total error} = \frac{\text{number of false classification}}{\text{the number of evaluation sample}} \quad (15)$$

3.1 Experiment Result

Our implementation was carried out on the Matlab6.5. Firstly, the data is normalized. In this method the Gaussian kernel is used, and the kernel parameter needs to be chosen. Thus the method has two parameters to be prepared set: the kernel parameter σ^2 and the regularized parameter λ . The Type I error (e1), Type II error (e2), Total error (e), number of selected features and the best pairs of (σ^2, λ) for each fold using SVM-MK approach are shown in table 1. For this method, its average Type I error is 24.25%, average Type II error is 17.24%, average Total error is 23.22% and average number of selected features is 18.

Table 1. Experimental results for each fold using SVM-MK

Fold #	e1 (%)	e2 (%)	e (%)	Optimized σ^2	Optimized λ	selected features
1	20.2	15.7	18.9	4	5	19
2	27.75	15.39	26.3	5.45	3	13
3	24.54	14.29	23.1	5	1	8
4	20.27	13.68	19.5	3	5	25
5	28.49	27.14	28.3	1.2	8	26
Average	24.25	17.24	23.22			18

Then, we take fold #2 as an example. The experimental results of SVM-MK using various parameters are illustrated in table 2. The performance is evaluated using five measures: number of the selected features (NSF), selected specific features, Type I error (e1), Type II error (e2) and Total error (e).

From table 2, we can draw the following conclusions:

- (1) The values of parameters to get the best prediction results are $\lambda = 3$ and $\sigma^2 = 5.45$, the number of selected features is 13, the bad ones' error is 15.39% and the total error is 26.3%, they are the lowest errors compared to the other situations. And almost eight of ten default creditors can be discriminated from the good ones using this model at the expense of denying a small number of the non-default creditors.
- (2) With the increasing of λ , the number of selected features is gradually increasing. A bigger parameter λ makes the value of coefficient matrix become decreasing so that the constraints of the dual linear programming can be satisfied. As a

Table 2. Classification error and selected features of various parameters in SVM-MK (fold #2)

λ	1	3	5	10	15	20
NSF	3	13	20	32	41	41
Selected specific features	54	3,8	3,8,9,10,11	3, 4, 8,9,10,12	3,4,5,6,8,10,12	3,4,5,6,8,10,12,14,15
	55	10,11	14,20,24,31	14,15,16,17,20	14,15,16,17,19,20	16,17,19,20,23,24,25
	61	14,20,31	38,40,42,47	23,24,25,28,31	23,24,25,28,31,32,33	28,31,32,33,34,37,38
		38,47,52	51,52,53,54	32,34,38,39,40	34,37,38,39,40,41,42	39,40,41,42,43,45
		53,55,61	55,60,61	42,45,47,51,52	43,45,47,48,50,51,52	47,48,50,51,52,53
				53,54,55,58	53,54,55,58,59,61,63	54,55,58,59,61,63
e1	100	92	14. 52	22. 55	35. 4	0.11
e2	0	3. 37	48.2	41. 45	21. 55	98. 29
$\sigma^2=2$ e	87.3	63. 5	19. 2	26.2	32. 45	11. 6
5	e1	17.8	38.2	22	4. 2	14.04
	e2	32.7	8.6	29	83.89	29. 9
	e	18.1	34.1	23	17	15. 9
5.45	e1	0	27.75	21	11.7	11.7
	e2	100	15.39	32.4	100	100
	e	13.9	26.3	24.5	11.7	11.7
10	e1	0	98.88	19. 4	2. 3	88
	e2	100	0	35. 3	93.16	3. 1
	e	13.9	86	21. 6	11.8	77. 8
11.4	e1	0	99.88	26. 62	1. 1	92
	e2	100	0	16. 24	94. 87	7. 53
	e	13.9	87.3	25. 4	11. 9	79. 94
15	e1	0	100	32.1	1. 15	94. 36
	e2	100	0	10. 8	92.83	6. 3
	e	13.9	87.3	29.1	13.9	81. 6

result, the sparse of $\mathcal{Y}_{i,d}$ becomes bad in the primal LP problem. That is to say, a bigger parameter λ results in selecting a good many of features. But the parameter σ^2 has no effect on the feature selection. When parameter λ is equal to 3, only 13 attributes is selected and the best classification results are obtained. It is shown that a reasonable feature extraction can improve the performance of the learning algorithm greatly. These selected specific attributes also help the loaner draw a conclusion as to the nature of credit risk existing in the information of the creditors easily. This implies λ plays a key role in the feature selection. So we must pay more attention to select the appropriate values for parameter λ .

- (3) When the value of parameter σ^2 matches the certain values of parameter λ , we can get promising classification results. In general, there is a trade off between Type I and II error in which lower Type II error usually comes at the expense of higher Type I error.

3.2 Comparison of Results of Different Credit Risk Evaluation Models

The credit dataset that we used has imbalanced class distribution. Thus, there is non-uniform misclassifying cost at the same time. The cost of misclassifying a sample in the bad class is much higher than that in the good class. So it is quite important that the prior probabilities and the misclassification costs be taken into account in order to

obtain a credit evaluation model with the minimum expected misclassification [13]. When there are only two different populations, the cost function in computing the expected misclassification is considered as follows:

$$cost = c_{21} \times \pi_1 \times p(2|1) + c_{12} \times \pi_2 \times p(1|2) \quad (16)$$

where c_{21} and c_{12} are the corresponding misclassification costs of Type I and Type II error, π_1 and π_2 are prior probabilities of good and bad credit applicants, $p(2|1)$ and $p(1|2)$ measure the probability of making Type I error and Type II error. In this study, $p(2|1)$ and $p(1|2)$ are respectively equal to Type I and Type II error. The misclassification ratio associated with Type I and Type II error are respectively 1 and 5 [13]. In order to further evaluate the effectiveness of the proposed SVM-MK credit evaluation model, the classification results are compared with some other methods using the same dataset, such as multiple criteria linear programming (MCLP), multiple criteria non-linear programming (MCNP), decision trees and neural network. The results of the four models quoted from the reference [14]. Table 3 summarizes the Type I, Type II and Total error of the five models and the corresponding expected misclassification costs (EMC).

Table 3. Errors and the expected misclassification costs of the five models

Model	e1 (%)	e2 (%)	e (%)	EMC
MCLP	24.49	59.39	30.18	0.51736
MCNP	49.03	17.18	43.84	0.52717
Decision Tree	47.91	17.3	42.92	0.51769
Neural Network	32.76	21.6	30.94	0.40284
SVM-MK	24.25	17.24	23.22	0.30445

The priors of good and bad applicants are set to as 0.9 and 0.1 using the ratio of good and bad credit customers in the empirical dataset.

From table 3, we can conclude that the SVM-MK model has better credit scoring capability in term of the overall error, the Type I error about the good class, the Type II error about the bad class and the expected misclassification cost criterion in comparison with former four models. Consequently, the proposed SVM-MK model can provide efficient alternatives in conducting credit evaluating tasks.

4 Conclusions

This paper presents a novel SVM-MK credit risk evaluation model. By using the 1-norm and a convex combination of basic kernels, the object function which is a quadratic programming problem in the standard SVM becomes a linear programming parameter iterative learning problem so that greatly reducing the computational costs. In practice, it is not difficult to adjust kernel parameter and regularized parameter to obtain a satisfied classification result. Through the practical data experiment, we have obtained good classification results and meanwhile demonstrated that SVM-MK model is of good performance in credit scoring system. And we get only a few

valuable attributes that can interpret a correlation between the credit and the customers' information. So the extractive features can help the loaner make correct decisions. Thus the SVM-MK is a transparent model, and it provides efficient alternatives in conducting credit scoring tasks. Future studies will aim at finding the law existing in the parameters' setting. Generalizing the rules by the features that have been selected is another further work.

Acknowledgements

This research has been partially supported by a grant from National Natural Science Foundation of China (#70531040), and 973 Project (#2004CB720103), Ministry of Science and Technology, China.

References:

1. G. Lee, T. K. Sung, N. Chang: Dynamics of modeling in data mining: Interpretive approach to bankruptcy prediction. *Journal of Management Information Systems*, 16(1999), 63-85
2. J. C. Wiginton: A note on the comparison of logit and discriminate models of consumer credit behavior. *Journal of Financial Quantitative Analysis* 15(1980), 757-770
3. B.J. Grablowsky, W. K. Talley: Probit and discriminant functions for classifying credit applicants: A comparison. *Journal of Economic Business*. Vol.33(1981), 254-261
4. T. Lee, C. Chiu, Y. Chou, C. Lu: Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. *Computational Statistics and Data Analysis*, Vol.50, 2006(4), 1113-1130
5. Yueh-Min Huang, Chun-Min Hung, Hewijin Christine Jiau: Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications* 7(2006), 720-747
6. V. Vapnik: *The nature of statistic learning theory*. Springer, New York(1995)
7. T. Van Gestel, B. Baesens, J. Garcia, and P. Van Dijcke: A support vector machine approach to credit scoring. *Bank en Financiewezen* 2 (2003), 73-82
8. Y. Wang, S. Wang, K. Lai, A new fuzzy support vector machine to evaluate credit risk. *IEEE Transactions on Fuzzy Systems*, Vol.13, 2005(6), 820-831
9. Wun-Hwa Chen, Jin-Ying Shih: A study of Taiwan's issuer credit rating systems using support vector machines. *Expert Systems with Applications* 30(2006), 427-435
10. A. Ch. Micchelli, M. Pontil: Learning the kernel function via regularization. *Journal of Machine Learning Research*, 6(2005), 1099-1125
11. G. R.G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, M.I. Jordan: Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5(2004), 27-72
12. F.R. Bach, G. R.G. Lanckriet, M.I. Jordan: Multiple kernel learning, conic duality and the SMO algorithm. *Twenty First International Conference on Machine Learning*, (2004), 41-48
13. D. West: Neural network credit scoring models. *Computers and Operations Research*, 27(2000), 1131-1152
14. J. He, Y. Shi, W. X. Xu: Classifications of credit cardholder behavior by using multiple criteria non-linear programming. In Y. Shi, W. Xu, Z. Chen (Eds.) *CASDMKM 2004*, LNAI 3327, Springer-Verlag Berlin Heidelberg, (2004), 154-163