International Journal of Computer Theory and Engineering, Vol.3, No.1, February, 2011
1793-8201

Incorporate Cost Matrix into Learning Vector Quantization Modeling: a Comparative Study of Genetic Algorithm, Simulated Annealing and Particle Swarm Optimization

Ning Chen, Bernardete Ribeiro, Armando Vieira, Joao Duarte and Joao C. Neves

Abstract—**Cost-sensitive learning is an important topic in bankruptcy prediction concerning the unequal misclassification cost of different classes. Learning vector quantization (LVQ) is a powerful tool to solve bankruptcy prediction problem as a classification task. The heuristic algorithms are applied widely in conjunction with artificial intelligent methods for solving optimization problems. The hybridization of heuristic techniques with existing classification algorithms is well illustrated in the field of bankruptcy prediction. In this paper, three hybrid heuristic-based LVQ approaches which combine LVQ with genetic algorithm, simulated annealing and particle swarm optimization respectively, are proposed to minimize the total misclassified cost under the asymmetric cost preference. The idea behind the hybrid classifier is the adoption of heuristic algorithms for the determination of the connection weights of the LVQ network. Experiments on French private company data show the proposed approaches offer interesting and viable alternatives for predictive reinforcement in cost-sensitive context.**

Index Terms—**bankruptcy prediction, learning vector quantization, heuristic algorithm, asymmetric misclassification cost, cost-sensitive learning, expected misclassified cost.**

I. Introduction

As one of the most important research problems in finance, bankruptcy prediction gives a great potential to reduce the risk of a credit process and shorten the time of credit evaluation [1]. The main task is building effective prediction models to estimate the potential failures.

Typically, it can be solved as a classification problem to separate distressed companies from healthy ones based on the analysis of financial characteristics. A large variety of methods for bankruptcy prediction have been developed including statistical approaches, artificial intelligent approaches, hybrid and ensemble approaches [2]. Among them, learning vector quantization (LVQ) is a powerful tool to solve bankruptcy prediction problem. The capability of LVQ to predict the bankruptcy of French companies is illustrated in [3], [4]. In financial failure prediction of Turkey banks, LVQ is shown to outperform multi-layer perceptron (MLP), support vector machines and multivariate statistical methods [5].

Common classification systems mainly focus on building accurate prediction models to decrease the total number of misclassifications and do not consider the implications of misclassification in decision making. However, in many real applications such as credit risk analysis, medical diagnostics and fraud detection, the costs of different classes are unequal. In the field of bankruptcy prediction, classifying a bankrupt company as a healthy one is more costly than classifying a healthy company as a bankrupt one. There are many evidences that the cost information is important for risk management of financial institutions. In reality, the costs can be estimated from the loan loss experience or foreseeable profit, and integrated into the classification system as a kind of existing knowledge. Classification systems that can handle asymmetric cost preference are crucial in providing a more desirable result to meet the needs of particular bankruptcy prediction applications.

Cost-sensitive learning becomes a hot topic in diverse applications since the asymmetric costs become critical. Meantime, the class imbalance problem is pervasive which could be solved in a similar manner to the unequal cost setup [6]. Cost-sensitive classification algorithms that enable effective prediction, where the costs of misclassification can be very different, are crucial to creditors and auditors in credit risk analysis.

Cost-sensitive learning is studied in various classification methods. The standard sampling techniques are considerably applied as a preprocessing phase to any classification method with the intention to modify the distribution of original data by increasing the number of expensive class and (or) decreasing the number of inexpensive class. Threshold-moving which moves the output towards the expensive class acts as a postprocessing technique on most classifiers with real-valued output. In [7], the cost matrix is integrated with basic LVQ algorithm using standard sampling and threshold-moving techniques. The cost

Manuscript received January 9, 2010. This work was supported by project C2007-FCT/442/2006-GECAD/ISEP (Knowledge Based, Cognitive and Learning Systems.

Ning Chen is with GECAD, Instituto Superior de Engenharia do Porto, Instituto Politecnico do Porto (telphone: 351-22-8340500; fax: 351-22-8321159; email: ningchen74@gmail.com).

Bernardete Ribeiro is with CISUC, Department of Informatics Engineering, University of Coimbra, Portugal (email: bribeiro@dei.uc.pt).

Armando Vieira is with Instituto Superior de Engenharia do Porto, Instituto Politecnico do Porto (email: asv@isep.ipp.pt).

Joao Duarte is with GECAD,.Instituto Superior de Engenharia do Porto, Instituto Politecnico do Porto (email: jmmd@isep.ipp.pt).

Joao C. Neves is with ISEG, School of Economics, Technical University of Lisbon, Portugal (email: jcneves@iseg.utl.pt).

information can be explicitly incorporated into the learning methodology on the algorithm level. Some efforts have been undertaken on specific classification methods including decision tree [8], regularized least square [9], boosting learning [10], mathematical programming [11], back-propagation neural network [12], logit and probit [13]. In [14], a cost-sensitive variant of LVQ is presented based on the modification of basic LVQ algorithm, which incorporates cost information into the model when performing the update of map neurons.

Recently, more and more efforts suggest hybrid approaches for predictive reinforcement. In a hybrid system, several approaches are exploited in the analysis and only one is employed for the final prediction [15]. The hybridization of heuristic techniques with existing classification algorithms is well illustrated for optimizing the prediction performance. At present, the most common used optimization techniques include genetic algorithm, simulated annealing, particle swarm optimization, tabu search and grid search [16]. Many evidences demonstrate the hybrid computational intelligence performs superior or in a competitive way to simple standard intelligent techniques due to the global search [17].

Since the objective of cost-sensitive classification can be described as minimizing the total misclassification cost [18], it is ideal to build the classifier with the aim to achieve the lowest value of the total misclassification cost. Heuristic algorithms are promising approaches to solve the optimization problem and construct a hybrid classifier. Previous studies have proposed many hybrid models of heuristic techniques and data mining techniques for the purpose of feature selection, topology optimization, network training and parameter determination. Despite the great potential of hybrid models based on heuristic techniques, few studies have hitherto been focused on the integration of LVQ with optimization approaches for cost-sensitive bankruptcy prediction and little information about the performance comparison is provided in the literature.

In this paper, three well-known global optimization approaches, namely, genetic algorithm, simulated annealing, and particle swarm optimization are applied to construct the heuristic-based LVQ models. The hybrid classification system is used to classify data when the misclassification costs of different classes are unequal and known. The rational of employing hybrid classifier is that LVQ has been shown to perform well in bankruptcy prediction applications, and heuristic algorithms are effective to enhance the capability of classifiers. The connection weights of network are coded as the input to heuristic evolution and optimized through the global search of the complex solution space. The expected misclassified cost is used as the objective function for evaluating the performance of the resultant LVQ model. The benefits of the hybrid classification approaches are demonstrated through comparative studies using real-world bankruptcy data.

The rest of this paper starts from the priori research in section II. Section III presents the framework of hybrid heuristic-based LVQ modeling. In section IV, the experimental design and empirical results are described. Lastly, the conclusions and expectations for future work are addressed in section V.

II. Research Backgrond

A. Learning Vector Quantization

LVQ is a neural network approach useful for complicated non-linear separation problems. The modeling technique is based on the neurons representing prototype vectors and the nearest neighbor approach for classifying data. The neurons are arranged on a regular low-dimensional grid and associated with the input vectors by connection weights called prototypes. LVQ starts from a trained and labeled map, then attempts to find the accurate boundaries to classify the instances in a supervised way. LVQ can be trained in two different ways. In the sequential way, the prototypes are updated for each training example, and in the batch way the prototypes are updated after a complete run of all training examples. In this paper, we use the batch LVQ version due to the advantages such as order insensitivity, fast convergence and elimination of learning rate influence [19].

In one round, an instance $x_i (1 \le i \le n)$ is input and the distance between x_i and prototypes is calculated, consequently the input is projected to the BMU.

$$BMU(x_i) = \arg\min_{1\le p\le m} d(x_i, m_p) \tag{1}$$

The projection of input x_i is defined by an indicative function h_{ip} whose value is 1 if m_p is the BMU of x_i, and 0 otherwise. The Voronoi set of a neuron comprises the instances which are projected to it.

$$h_{ip} = \begin{cases} 1 & if\, m_p = BMU(x_i) \\ 0 & otherwise \end{cases} \tag{2}$$

The class matching between the input instance and BMU is represented by a denotative function.

$$s_{ip} = \begin{cases} 1 & if\, label(m_p) = label(x_i) \\ -1 & otherwise \end{cases} \tag{3}$$

After all the inputs are processed, the neurons are assigned by the majority of class labels in Voronoi set for acquiring the labeled map. The indicative and denotative functions are then used in the prototype update in a batch round.

Let $m_p(t)$ be the prototype vector of the p^{th} unit at epoch t. The learning rule is formulated in (4). (If the denominator is 0 or negative for some m_p, no updating is done.)

$$m_p(t+1) = \sum_{i=1}^{n} h_{ip} s_{ip} x_i \Big/ \sum_{i=1}^{n} h_{ip} s_{ip} \tag{4}$$

This training process is repeated iteratively until the maximum number of iteration is reached or the amount of variation of prototypes between two consecutive iterations is less than a specified threshold.

The Cost-LVQ [14] resembles the basic batch LVQ except that the misclassification costs are utilized as weights guiding the prototype learning with the aim that more attention is paid to the class associated with higher cost. In the algorithm, the cost information is incorporated in the denotative function

s_{ip} , in which $C_{label(x_i)}$ is the misclassification cost associated with the class of observation x_i .

$$s_{ip} = \begin{cases} C_{label(x_i)} & if\, label(m_p) = label(x_i) \\ -1 & otherwise \end{cases} \quad (5)$$

By incorporating the costs into the learning rule, the examples of relatively higher cost impose more impact on the prototypes so that they are harder to be misclassified. In this study, the Cost-LVQ is used as a base line for performance comparison.

Although the Cost-LVQ performs superior over the Sole-LVQ without integrating cost in terms of total misclassification cost, it is not designed for achieving the optimal solution. It is promising to embed the heuristic search into LVQ modeling for global optimization. The idea behind the hybrid classifier is the adoption of heuristic algorithms for determining the connection weights. Unlike the learning process of LVQ, the heuristic-based approaches directly adjust the connection weights of LVQ and search for a training-completed model through an optimization process.

B. Genetic Algorithm

Genetic algorithm (GA) is an effective evolutionary tool in a wide variety of combinatorial optimization problems. It is used to search the solution space through the natural selection evolution process. The genetic algorithm starts from a population of solutions encoded in the chromosomes and attempts to create better offsprings for the next generation through some genetic operators including crossover, mutation, and selection. The fitness function is used to evaluate the merit of individuals in the population. The individuals having a higher fitness value are selected with a higher probability to the next generation. The crossover operates on two selected individuals by exchanging parts of them and results into two new solutions. Mutation changes some positions of an individual in order to search for a broader space and avoid the local convergence. The iterative process is repeated a number of times until the stop criteria are satisfied, such as the maximum number of iteration, the minimal fitness value, or the improvement of the best fitness value.

The potential of genetic algorithm is increasingly illustrated in conjunction with artificial intelligence techniques for obtaining globally optimal solutions. It is usually used to determine the parameters of predictors, such as the weights of neural networks and the kernel width of support vector machine (SVM). As indicated in [20], hybrid models can advance the single prediction model in conjunction with genetic algorithm. In [11], the hybrid global programming and genetic algorithm approach that incorporates the asymmetric costs is compared with three linear classification models. The results indicate the hybrid model provides the best performance in terms of total misclassification cost. GA is employed to optimize the connection weights of neural network and the thresholds of feature discretization through the evolutionary operators [21]. A novel fitness function of GA is applied to optimize the parameters and weights of back-propagation neural networks for classifying imbalanced data sets [22]. In [23], a GA-based neural network is proposed to incorporate asymmetric error costs. The results therein show that an equal or lower holdout sample misclassification cost is obtained when compared with the other statistical, mathematical, and machine learning misclassification cost-minimizing approaches. A hybrid technique that optimizes the weights of the features and training instances simultaneously by GA enhances the performance of case-based reasoning (CBR) [24]. Genetic algorithm is also used to improve the performance of SVM in both feature subset selection and parameter optimization [20]. Additionally, a genetic algorithm-based sampling is performed on the imbalanced data taken the area under ROC as the fitness function. The combined technique of GA and SVM shows the superiority over random sampling [25].

C. Particle Swarm Optimization

Particle swarm optimization (PSO) [26] is a population-based heuristic search method, inspired by the social behavior of organisms such as birds flocking and fish schooling to configure the global optimization mechanism. Different from genetic algorithm, PSO has no complicated operators such as crossover and mutation. In essence, the PSO search starts from an initial population of particles and updates the property of particles iteratively. Each particle is characterized by the position denoting the problem solution and velocity directing the learning grade in the multidimensional space. The primary searching stratagem of PSO stems from the constructive cooperation and information sharing among particles, in the sense that the evolution of candidate solutions is based on the orientation of their own best experience and other individuals' best experience. In each iteration, the fitness of particles is evaluated and the best locations of individual (local best) and swarm (global best) are remembered. The searching direction of particles is guided by their own local best and the global best of the swarm members in order to discover the optimal or near optimal solution.

The success application of PSO has been demonstrated in a variety of optimization tasks, such as function minimization, neural network design, feature selection and finite element updating. In [27], PSO is applied to feature selection and parameter setting of SVM and decision tree. The results on Taiwanese commercial bank data show the PSO-based methods are effective in selecting beneficial features and determining suitable parameters. In [28], the PSO-SVM model in which PSO is used to determine the parameters of SVM, shows higher accuracy than grey model and neural network in predicting the dissolved gases content in power transform oil. Likewise, in the fuzzy inference system PSO is applied to find the optimal parameters and generate the appropriate codebooks for image compression [29].

D. Simulated Annealing

In addition to GA and PSO, another well-known heuristic method is simulated annealing (SA). SA is a probabilistic meta-heuristic for global optimization problems. The concept of SA comes from the annealing in metallurgy, which involves a heating to a high energy level and a gradual cooling until the solid state is reached. The slow cooling produces configurations with low energy, whereas the fast cooling produces poor configurations. The search of SA finds the optimal solution by simulating the natural annealing procedure. The search starts from a random initial solution and generates a new solution from the neighborhood. The solution is evaluated by the objective function and compared with the current solution. If the new solution is better than the current one, it is accepted immediately and retained as the current solution for the next step. Otherwise, it is accepted with a probability determined by Metropolis criteria. The temperature is gradually decreased during the process so that the worse solutions become less likely accepted. The search continues from the new solution until the temperature reaches the predefined minimum. The rational of SA is that by not only accepting the better solution but occasionally accepting the worse solution, the search is able to avoid the local minimum.

SA is shown to perform well for optimizing complex problems. In a SA-based approach, the optimal parameter setting of back-propagation network is derived by simulated annealing optimization, as well as the beneficial subset of features [30]. Results on several UCI datasets demonstrate the proposed approach outperforms other approaches on both binary and multi classification. In [31], the appropriateness of SA and GA as global search algorithms is investigated in optimizing the neural network. A SA-based approach is developed to facilitate the optimization in process planning and scheduling [32].

III. Hybrid Heuristic-Based LVQ Approaches

A. Objective Function

In the case of a binary classification, the cost information and classification result can be represented in a two-dimensional matrix as shown in Table I. The off-diagonal values denote the two kinds of errors and their costs, in which B_g is the number of observations classified as good when they are actually bad, conversely, G_b is the number of observations classified as bad when they are actually good, C_b is the misclassification cost of predicting a bad instance as a good one, and C_g is the cost of predicting a good instance as a bad one. For simplicity, we set C_g as 1, and C_b a number no less than 1 in this study.

Table I Confusion matrix and cost.

Actual class	Predicted class		
	bad	good	total
bad	$B_b\,(0)$	$B_g\ (C_b)$	B
good	$G_b\ (C_g)$	$G_g\,(0)$	G
total	b	g	T

Expected misclassified cost (EMC) is a criterion used in credit risk analysis [18]. As described in (6), it combines type I error rate with type II error rate and takes into consideration the prior possibility and the corresponding cost. In the formulation, P_b is the prior probability of class 'bad', and P_g is the prior probability of class 'good'. Type I error rate is the conditional probability of 'bad' class when the actual class is 'good', i.e., $P(b\,|\,g) = G_b / G$. Type II error rate is the conditional probability of 'good' class when the actual class is 'bad', i.e., $P(g\,|\,b) = B_g / B$. In the case of equal cost setup, the EMC is same to overall error rate, which is the percent of instances classified incorrectly, i.e., $B_g + G_b / T$.

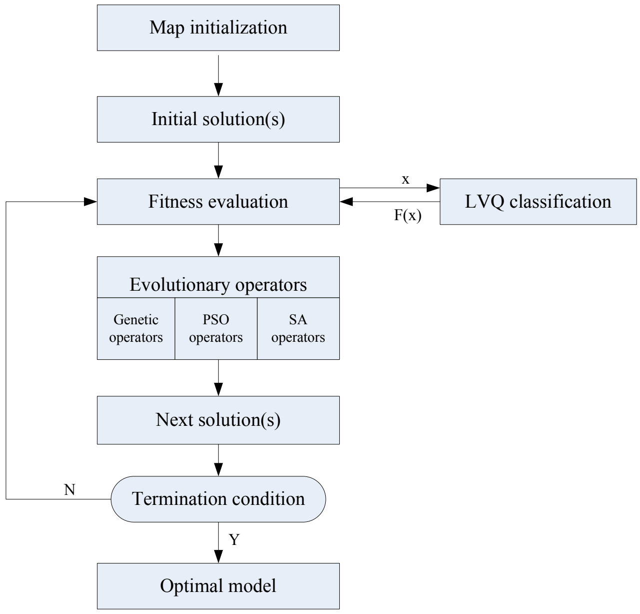

Fig. 1. Framework of heuristic-based LVQ approaches.

$$EMC = C_b \cdot P_b \cdot P(g\,|\,b) + C_g \cdot P_g \cdot P(b\,|\,g) \qquad (6)$$

In the heuristic-based approaches, EMC is used as the objective function to evaluate the predictive performance of solutions. The aim is to search for the optimal prototypes of LVQ model which produces the minimal expected misclassified cost under a specific cost setup.

B. Framework Description

The hybrid heuristic-based LVQ framework is outlined in Fig. 1. LVQ is used as the predictive model and heuristic algorithms are employed to explore good solutions, i.e., the prototypes of network, in the solution space as a global optimization model. Firstly, an initial solution in the case of SA or population of solutions in the case of GA and PSO is generated containing the coding of prototypes.

Then the individual(s) are imported into LVQ model and the fitness value of solutions is calculated in terms of the EMC criterion using the training data. The evolutionary operators, namely, selection, crossover and mutation in the case of GA, local best and global best operators in the case of PSO, Metropolis operator in the case of SA, are performed to generate new solution(s) for the next iteration. The optimal solution is output when the stop criterion is satisfied. Afterwards, the optimal solution is used to predict the bankruptcy on the test data.

The parameter settings of the three heuristic approaches are illustrated in Table II. We use *genetic algorithm toolbox in Matlab* for the implementation of GA. The connection weights are coded in the real-valued chromosomes. The parameters of the algorithm are set to the recommended values.

The version of PSO used in this study comes from [33]. The model is trained for 1000 iterations, including a search of 20 particles in the swarm. Other parameters are set to the recommended values. The general simulated annealing algorithm [34] implemented by Joachim Vandekerckhove is embedded in the SA-LVQ classifier. Unlike the population-to-population algorithms, SA is a point-to-point search algorithm and is greatly affected by the starting point [31]. For each cost configuration and training data, 10 replications are trained starting with random initial seeds. The resultant model achieving the best solution is used for validation.

IV. Empirical Analysis

The proposed hybrid approaches are implemented based on the well-known *somtoolbox* [35] in Matlab.

A. Data Description

The Diane data set contains financial statements of private-owned French companies of small or medium size. As described in Table III, each company is characterized by a set of 29 financial ratios. The companies are divided into two categories, one is 'bad' which declares bankruptcy or submit reconstruction plan to the French court in 2007, the other is 'good' indicating the healthy situation. The problem of interest is to predict the failure of a company based on the past financial ratios over a given period. We consider one year preceding the default in the paper. A balanced sample is selected for experiment in which 600 examples are 'good' and the rest is 'bad'.

The ratios are preprocessed by logarithmized operation followed by normalization. We use the linear normalization which transforms the data into unity range.

B. Experimental Results

The experiments are performed in the following steps:

1) We perform 10-fold cross validation on the data set. The entire data set is divided randomly into ten folds, in which 9 folds are used for model training, and the remaining is used for model test.
2) For each trial, the proposed approaches are applied to the training data set.
3) For validation, the test data is fed to the resultant map and the class is predicted.
4) After the experiment is repeated 10 times, the classification result is evaluated in terms of type I error rate, type II error rate, overall error rate and EMC.

We made comparisons at varying cost ratio (C_b / C_g) between the performance of heuristic-based LVQ approaches and two competing approaches, one is the Sole-LVQ that does not incorporate cost matrix, the other is Cost-LVQ that incorporates cost matrix into the standard updating rule of LVQ. As observed in Fig. 2, the proposed hybrid classifiers indeed lead to performance improvement in comparison with both Sole-LVQ and Cost-LVQ, in the sense that they yield a lower type II error which is more expensive than type I error and hence decrease the EMC value.

The variation of costs does not affect the prediction of Sole-LVQ, hence the resultant EMC increases linearly with the cost ratio and produces the worst performance. As indicated in [14], the Cost-LVQ has a tradeoff between type I error and type II error at a small (or moderate) cost ratio, however, the improvement is not significant when the cost ratio is higher than 10. Through the objective function which directly minimizes the misclassification cost and global search which finds a near optimal solution, the heuristic-based LVQ approaches produce better result than Cost-LVQ in almost all cases. The gains of heuristic search are much more evident as the cost ratio increases. These observations lead us to conclude that the hybridization of heuristic algorithms with existing LVQ modeling helps to improve the predictive performance under the asymmetric costs.

For a comparative study, GA-LVQ is able to systematically achieve superior solutions for optimizing the network than SA-LVQ and PSO-LVQ, leading to lower EMC and overall error simultaneously in Fig. 3. A t-test is conducted to show the difference of EMC between pairwise alternatives. Table IV illustrates the observed difference of each comparison at the level of 5%. If a test of significance gives a p-value lower than the 0.05-level, the null hypothesis is rejected. As can be seen in this table, the GA-LVQ finds superior solutions over PSO-LVQ and SA-LVQ in almost all cost configurations, in which five and three comparisons are significantly different respectively. Meantime, the comparisons between PSO-LVQ and SA-LVQ are not significantly different at six cases out of the total seven.

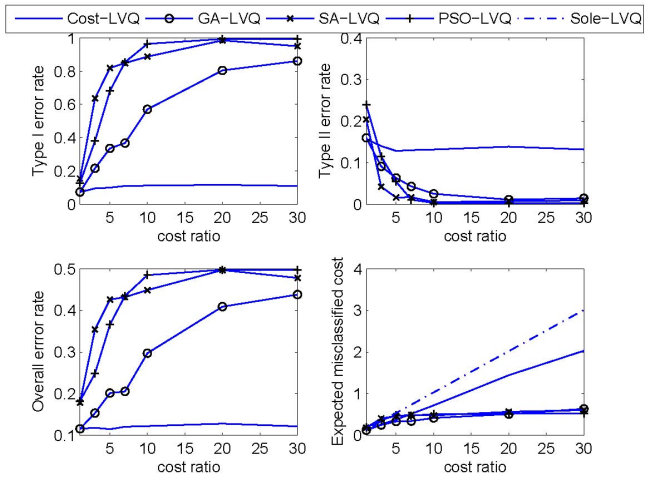

Fig. 2 Classification results on Diane data.

TABLE IV PAIRWISE COMPARISON AMONG THREE APPROACHES AT 5% LEVEL (TEST STATISTIC VALUES ARE GIVEN AND SIGNIFICANCE ARE MARKED BY *).

Cost ratio	GA-LVQ vs. PSO-LVQ	GA-LVQ vs. SA-LVQ	PSO-LVQ vs. SA-LVQ
1	-3.54*	-5.42*	0.75
3	-4.52*	-1.75	2.8*
5	-5.46*	-4.59*	-0.37
7	-3.93*	-2.36*	1.4
10	-2.49*	-2.04	-0.27
20	-0.15	-1.31	-1.31
30	0.34	-1.42	-1.57

We use receiver operating characteristic (ROC) to characterize the performance of the binary classifier. The performance is demonstrated in Fig. 3, giving the evidence that GA-LVQ outperforms SA-LVQ and PSO-LVQ in making tradeoff between two kinds of errors.

V. Conclusions and Future Remarks

The problem of predicting bankruptcy is extensively studied in finance analysis and machine learning to evaluate the risk associated in credit decisions. Due to the presence of unequal misclassification costs, the cost-sensitive classification that can handle asymmetric costs is of crucial interest to credit decisions. This paper describes how the asymmetric costs of two kinds of errors are integrated into a hybrid classification system for the prediction of bankruptcies. Three heuristic methods are used to search the solution space for an optimal or near optimal configuration of the LVQ model to minimize the expected misclassified cost. Experimental tests using real world bankruptcy data demonstrate that the heuristic search greatly increases the capability of LVQ to avoid the local minimum and hence improves the predictive performance in asymmetric cost setup. Comparatively, genetic algorithm appears to be able to obtain superior solutions to particle swarm optimization and simulated annealing for optimizing the neural network.

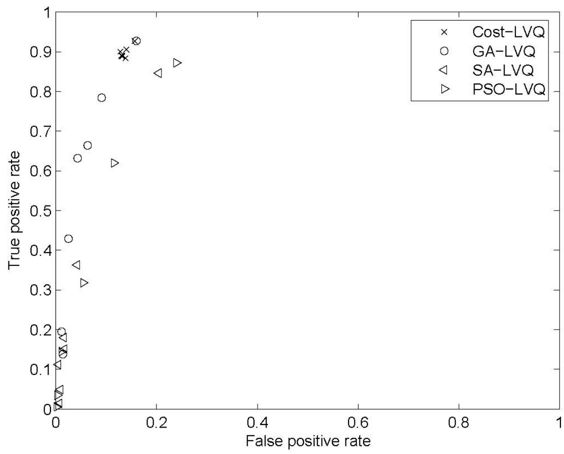

Fig. 3 ROC of competing approaches.

More studies are expected in the future. First, comparative analysis will be undertaken between the proposed hybrid approaches and state-of-the-art cost-sensitive classification systems. Second, this study employs all ratios as the input, whereas the selection of important ratios or weight specification of features can be optimized simultaneously in the mechanism to enhance the predictive performance. Third, the specification of a proper cost setup is nontrivial in practical applications and deserves further discussion through comprehensive study. Fourth, the objective function used in this study is one of the possible alternatives for misclassification cost minimization [11], [23]. Other formulations will be tested for the same objective in future. Nevertheless, the hybrid approaches provide the flexibility in integrating the cost matrix into the objective function. Several extensions could be implemented by modifying the objective function, e.g., considering both benefit and cost of classification. Finally, the heuristic methods can be combined with other data mining models and applied to other real world problems.

References

[1] H. A. Abdou, "Genetic programming for credit scoring: the case of Egyptian public sector banks," *Expert Systems with Applications*, 2009, doi:10.1016/j.eswa.2009.01.076.

[2] P. R. Kumar and V. Ravi, "Bankruptcy prediction in banks and firms via statistical and intelligent techniques–a review," *European Journal of Operational Research*, 180(1), 2007, pp. 1–28.

[3] J. C. Neves and A. Vieira, "Improving bankruptcy prediction with hidden layer learning vector quantization," *European Accounting Review*, 15(2), 2006, pp. 253–271.

[4] N. Chen and A. Vieira, "Bankruptcy prediction based on independent component analysis," Proc. 1st International Conference on Agents and Artificial Intelligence (ICAART09), Porto, Portugal, 2009, pp. 150–155.

[5] M. A. Boyacioglu, Y. Kara, and O. K. Baykan, "Predicting bank financial failures using neural networks, support vector machines and multivariate statistical methods: a comparative analysis in the sample of savings deposit insurance fund (SDIF) transferred banks in Turkey," *Expert Systems with Applications*, 36, 2009, pp. 3355–3366.

[6] M. A. Maloof, "Learning when data sets are imbalanced and when costs are unequal and unknown," Proc. 20th International Conference on Machine Learning (ICML), 2003, Washing D.C., USA., pp. 154–160.

[7] N. Chen, A. Vieira, and J. Duarte, "Cost-sensitive LVQ for bankruptcy prediction: an empirical study," Proc. Li, W., Zhou, J. (eds.), 2nd IEEE International Conference on Computer Science and Information Technology, Beijing, 2009, pp. 115-119.

[8] C. X. Ling, Q. Yang, J. Wang, and S. Zhang, "Decision trees with minimal costs," Proc. Brodley, C.E. (ed.), ACM International Conference Proceeding Series, vol. 69, 21st International Conference on Machine Learning (ICML), 2004, Banff, Canada.

[9] N. H. Vo and Y. Won, "Classification of unbalanced medical data with weighted regularized least squares," Frontiers in the Convergence of Bioscience and Information Technologies, 2007, pp. 347–352.

[10] Y. M Sun, M. S. Kamela, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, 40, 2007, pp. 3358–3378.

[11] S. Nanda and P. Pendharkar, "Linear models for minimizing misclassification costs in bankruptcy prediction," *Int J Intell Syst Account Finance Manage*, 10, 2001, pp. 155–168.

[12] K.Y. Tam and M.Y. Kiang, "Managerial applications of neural networks: the case of bank failure predictions," *Manage Sci*, 38, 1992, pp. 926–946.

[13] H. C. Koh, "The sensitivity of optimal cutoff points to misclassification costs of type I and type II errors in the going-concern prediction context," *Journal of Business Finance and Accounting*, 17, 1992, pp. 187–197.

[14] N. Chen, A. S. Vieira, J. Duarte, B. Ribeiro, and J. C. Neves, "Cost-sensitive learning vector quantization for financial distress prediction," Proc. Lopes L.S. et al. (eds.), EPIA 2009, LNAI 5816, 2009, pp. 374–385.

[15] A. Verikas, Z. Kalsyte, M. Bacauskiene, and A. Gelzinis, "Hybrid and ensemble-based soft computing techniques in bankruptcy prediction: a survey," *Soft Computing*, 2009, doi:10.1007/s00500-009-0490-5.

[16] D. T. Pham and D. Karaboga, *Intelligent Optimisation Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks*, Springer-Verlag New York, Inc., 1998.

[17] A. Tsakonas and G. Dounias, "Hybrid computational intelligence schemes in complex domains: an extended review," *Lecture Notes in Computer Science*, Vol. 2308, 2002, pp. 747–765.

[18] T. Lee and I. Chen, "A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines," *Expert Systems with Applications*, 28(4), 2005, pp. 743–752.

[19] T. Kohonen, *Self-Organizing Maps*, Springer Verlag (3rd edition), 2001.

[20] S. H. Min, J. Lee, and I. Han, "Hybrid genetic algorithms and support vector machines for bankruptcy prediction," *Expert Systems with Applications*, 31, 2006, pp. 652–660.

[21] K. Kim and I. Han, "Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index," *Expert Systems with Applications*, 19, 2000, pp. 125–132.

[22] K. C. Huang, Y. H. Kuo, and I. C. Yeh, "A novel fitness function in genetic algorithm to optimize neural networks for imbalanced data sets," Proc. 8th International Conference on Intelligent Systems Design and Applications, 2008, pp. 647–650.

[23] P. Pendharkar and S. Nanda, "A misclassification cost-minimizing evolutionary–neural classification approach," *Naval Research Logistics*, 53(5), 2006, pp. 432–447.
[24] H. Aha and K. Kim, "Bankruptcy prediction modeling with hybrid case-based reasoning and genetic algorithms approach," *Applied Soft Computing*, 9, 2009, pp. 599-607.
[25] S. X. Zou et al., "SVM learning for imbalanced data by GA sampling for protein domain prediction," Proc. 9th International Conference for Young Computer Scientists, 2008, pp. 982–987.
[26] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*, Morgan Kaufmann, 2001.
[27] S. W. Lin, Y. R. Shiue, S. C. Chen, and H. M. Cheng, "Applying enhanced data mining approaches in predicting bank performance: a case of Taiwanese commercial banks," *Expert Systems with Applications*, 36, 2009, pp. 11543–11551.
[28] S. W. Fei, M. J. Wang, Y. Miao, J. Tu, and C. L. Liu, "Particle swarm optimization-based support vector machine for forecasting dissolved gases content in power transformer oil," *Energy Conversion and Management*, 50, 2009, pp. 1604–1609.
[29] H. M. Feng, C. Y. Chen, and F. Ye, "Evolutionary fuzzy particle swarm optimization vector quantization learning scheme in image compression," *Expert Systems with Applications*, 32, 2007, pp. 213–222.
[30] S. W. Lin, T. Y. Tseng, S. Y. Chou, and S. C. Chen, "A simulated-annealing-based approach for simultaneous parameter optimization and feature selection of back-propagation networks," *Expert Systems with Applications*, 34, 2008, pp. 1491–1499.
[31] R. S. Sexton, R. E. Dorsey, and J. D. Johnson, "Optimization of neural networks: a comparative analysis of the genetic algorithm and simulated annealing," *European Journal of Operational Research*, 114, 1999, pp. 589–601.
[32] W. D. Li and C. A. McMahon, "A simulated annealing-based optimization approach for integrated process planning and scheduling," *International Journal of Computer Integrated Manufacturing*, 20(1), 2007, pp. 80–95.
[33] PSO TOOLBOX, http://psotoolbox.sourceforge.net.
[34] J. Vandekerckhove, General Simulated Annealing Algorithm, http://www.mathworks.com/matlabcentral/fileexchange/10548
[35] Laboratory of Computer and Information Sciences & Neural Networks Research Center, Helsinki University of Technology, SOM Toolbox 2.0, http://www.cis.hut.fi/somtoolbox

TABLE II PARAMETER SETTING OF HEURISTIC SEARCH.

Approach	Parameter and Value
GA	100 %Maximum iterations
	20 %Population size
	[0,1] %Range of the individuals in the initial population
	20 %Maximum running seconds if no improvement in the objective function
	30 %Maximum running seconds
	50 %Maximum consecutive generations if no improvement in the objective function
	stochastic uniform %Selection function
	Gaussian %Mutation function
	scattered %Crossover function
	0.8 %Crossover fraction
	0.2 %Migration fraction
PSO	1000 %Maximum iterations
	20 %Swarm size
	2 %Social acceleration
	1 %Neighborhood acceleration
	0.95 %Value of velocity weight at the beginning of iterations
	0.4 %Value of velocity weight at the end of iterations
	0.7 %Fraction of maximum iterations
	100 %Maximum velocity step
SA	1 %Initial temperature
	10^{-8} %Temperature at which to stop
	-Inf %Value at which to stop immediately
	0.8 %Temperature reduction factor
	1000 %Maximum number of consecutive rejections
	300 %Maximum number of tries within one temperature
	20 %Maximum number of successes within one temperature

TABLE III FINANCIAL RATIOS OF DIANE COMPANIES.

Variable	Description
x1	Number of Employees Last available year
x2	Capital Employed / Fixed Assets
x3	Financial Debt / Capital Employed
x4	Depreciation of Tangible Assets
x5	Working Capital / Current Assets
x6	Current ratio
x7	Liquidity Ratio
x8	Stock Turnover days
x9	Collection Period days

x10	Credit Period days
x11	Turnover per Employee in EUR
x12	Interest / Turnover
x13	Debt Period days
x14	Financial Debt / Equity
x15	Financial Debt / Cashflow
x16	Cashflow / Turnover
x17	Working Capital / Turnover days
x18	Net Current Assets/Turnover days
x19	Working Capital Needs / Turnover
x20	Added Value per Employee in EUR
x21	Total Assets Turnover
x22	Operating Profit Margin
x23	Net Profit Margin
x24	Added Value Margin
x25	Part of Employees
x26	Return on Capital Employed
x27	Return on Total Assets
x28	EBIT Margin
x29	EBITDA Margin
x30	Class (‘bad’, ‘good’)