Neural Processing Letters (2021) 53:3561–3572
https://doi.org/10.1007/s11063-021-10564-0

Semi-Supervised Clustering for Financial Risk Analysis

Yihan Han[1] · **Tao Wang**[2]

Accepted: 12 June 2021 / Published online: 24 June 2021

Abstract
Many methods have been developed for financial risk analysis. In general, the conventional unsupervised approaches lack sufficient accuracy and semantics for the clustering, and the supervised approaches rely on large amount of training data for the classification. This paper explores the semi-supervised scheme for the financial data prediction, in which accurate predictions are expected with a small amount of labeled data. Due to lack of sufficient distinguishability in financial data, it is hard for the existing semi-supervised approaches to obtain satisfactory results. In order to improve the performance, we first convert the input labeled clues to the global prior probability, and propagate the'soft' prior probability to learn the posterior probability instead of directly propagating the'hard' labeled data. A label diffusion model is then constructed to adaptively fuse the information at feature space and label space, which makes the structures of data affinity and labeling more consistent. Experiments on two public real financial datasets validate the effectiveness of the proposed method.

Keywords Financial risk analysis · Data clustering · Semi-supervised learning · Affinity diffusion

1 Introduction

The outbreak of the (COVID-19) pandemic on the global scale led to the significant change in the world over the past year, destabilizing the global economy and stock markets. The massive economic hit from COVID-19 has dramatically increased financial risk and forced an increasing number of companies into bankruptcy. Financial risks, such as credit risk, operational risk, and business risk are generally uncertainties with any form of financing, which causes the difficulty of data analysis. Data analysis can help predict risk in advance, which is a key step for company decision-making [1, 2] in order to minimize the defaults. The evolving nature of the COVID-19 pandemic and the associated economic uncertainties

✉ Yihan Han
hyh5676@163.com

[1] Suzhou Institute of Trade and Commerce, Suzhou 215009, People's Republic of China

[2] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, People's Republic of China

require more efforts to support financial resilience. Therefore, the research on risk prediction is particularly important.

Many methods have been proposed for financial data analysis, which can be generally divided into two categories: unsupervised approaches and fully-supervised approaches. Typical unsupervised methods include the popular clustering algorithms, such as the k-means algorithm [3], expectation–maximization (EM) algorithm [4] and graph-partitioning algorithm [5]. In [6], Kohonen's self-organizing feature map is utilized to uncover automobile bodily injury claims fraud. In [7], a fuzzy clustering system is developed to detect anomalous behaviors in healthcare provider claims. In [8], unsupervised neural networks are utilized to identify fraud in mobile communications. In [9], hierarchical clustering method is developed to predict risks in insurance industry. These methods can automatically analyze the data without any prior information. However, they are generally limited to the accuracy of data analysis. Since no label prior is provided, they also cannot assign the clusters to the corresponding labels (lack of semantic understanding). Therefore, it's hard to evaluate the performance of these unsupervised clustering methods. As suggested in [10], a multiple criteria decision making strategy can better evaluate clustering algorithms in the domain of risk analysis. Typical fully-supervised methods include the machine learning-based methods [11, 12]. Compared with unsupervised methods, fully-supervised methods can generally achieve higher prediction accuracy. However, the high-quality performance of fully-supervised methods relies on large amount of training data. They are inapplicable when not enough labeled data is provided. Due to uncertainty in financial data, these fully-supervised approaches generally lack versatility. For example, a trained model for credit risk analysis cannot be applied for business risk analysis. They need to retrain the model with the new labeled data in business risk.

To address the above problems, in this paper, the semi-supervised scheme is explored for financial data analysis. Only a small amount of labeled data is needed in semi-supervised scheme. Then all the unlabeled data can be automatically clustered based on the labeled data. Compared with unsupervised methods, the label (normal or abnormal) of each data can be specifically determined in semi-supervised strategy since each label prior is provided. Furthermore, the provided label information can help to improve the clustering performance. Compared with fully-supervised methods, semi-supervised scheme has greater versatility and it can be directly applied to different data without any additional cost. Moreover, only a small amount of labeled data is needed to obtain a semantic classification. Comprehensively, semi-supervised scheme is more practical for financial data analysis.

In the semi-supervised model [13–15], the label information can be propagated from labeled data to unlabeled data based on their pairwise relationships. The data manifold is represented as a weighted graph, where the vertices in the graph represent each data and the edge connecting two adjacent vertices is determined by the initial pairwise similarity values. After the diffusion, the geometry of the data manifold can be effectively captured. However, due to lack of sufficient distinguishability, the conventional semi-supervised approach [13] cannot obtain accurate risk prediction with limited labeled data, and also be sensitive to the number of labeled data. Furthermore, the pairwise similarity is not always consistent with the category information, which causes the label prior cannot be correctly propagated following the mismatched smoothing structure.

The contributions of this paper can be described as: *first*, instead of directly propagating the'hard' prior label information, we transform the'hard' prior information to the'soft' global probability first, and then the'soft' prior probability is propagated to learn the posterior probability, which helps to produce more accurate risk prediction and specific

semantic labeling without the demand of a large number of labeled data; *second*, the label prior is utilized to correct the pairwise relationship, trying to make the structures of data affinity and labeling more consistent, and an automatic fusion strategy is proposed to effectively combine the data affinity and the labeling information by an adaptive label diffusion framework.

2 Conventional Semi-supervised Model

A set of financial data can be denoted as $X = \{x_i\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^d$ represents the risk factors of each data, d is the number of attributes, and N represents the number of data. The purpose of data clustering is to assign each data $x_i \in X$ a risk discriminating label $f_i \in L$, where the label set L generally contains two label values, one is normal (no risk) and the other is abnormal (risky). In semi-supervised scheme, a small amount of data is labeled for each label first. The labeled data set with each label $l \in L$ is denoted as $X^l \subset X$. The label information is then propagated from the labeled data to unlabeled data following the structure of their pairwise similarities $W = [W_{ij}]_{N\times N}$, generally defined as a typical Gaussian function:

$$W_{ij} = \exp\left(-\nu \left\| x_i - x_j \right\|_2^2\right) \tag{1}$$

$$\nu = \frac{1}{2\mathrm{EP}\left(\left\| x_i - x_j \right\|_2^2\right)} \tag{2}$$

where i and j represent the data x_i and x_j, respectively. The automatic constant ν is utilized to control the strength of the weight and $\mathrm{EP}(\cdot)$ represents the expectation over all data pairs. It can be noticed that the weight W_{ij} is large (close to 1) if their attribute characteristics are similar, and vice versa.

As described in [13], the label learning process with respect to the label $l \in L$ can be formulated as minimizing:

$$E(\Pi_l) = \sum_{i,j=1}^{N} W_{ij}(\pi_{il} - \pi_{jl})^2 + \lambda \sum_{i=1}^{N} d_i(\pi_{il} - z_{il})^2 \tag{3}$$

where $\Pi_l = [\pi_{il}]_{N\times 1}$ represents the posterior probability of being learned with the label l. $d_i = \sum_{j=1}^{N} W_{ij}$ and $\lambda = (1-\alpha)/\alpha$ $(0 < \alpha < 1)$ is utilized to balance these two energy terms. z_{il} represents the'hard' prior label information, where z_{il} equals 1 if x_i is labeled with l, and otherwise equals 0.

The first energy term in Eq. (3) restricts that if the pairwise similarity W_{ij} is large, x_i and x_j should have similar posterior probabilities. The second energy term in Eq. (3) tries to keep the posterior probability be consistent with the'hard' prior condition.

After derivation optimization, it has:

$$\Pi_l = (1-\alpha)(I - \alpha P)^{-1} Z_l \tag{4}$$

where $P = D^{-1}W$ with $D = diag([d_1, ..., d_N])$, I is an identity matrix, and $Z_l = [z_{il}]_{N\times 1}$. Suggested by [13], the above probability learning process is also equivalent to the following label diffusion strategy:

$$\Pi_l^{(t+1)} = \alpha P \Pi_l^{(t)} + (1-\alpha) Z_l \tag{5}$$

where t represents the diffusion steps. $\Pi_l^{(t+1)}$ converges to the same solution with Eq. (4) when $t \to \infty$. The final labeling can be obtained as:

$$f = \arg\max_l \Pi_l \tag{6}$$

3 The Proposed Model

The above 'hard' label diffusion model is not suitable for data analysis since the limited label information cannot be correctly propagated following the inaccurate structure of data affinity. We estimate the 'soft' prior probability from the 'hard' labeled data first, which can also be regarded as a unary diffusion process from the local seeds to the global probabilities. The prior probability that x_i belongs to the label l can be estimated as:

$$\bar{\pi}_{il} = \exp\left(-\left\|x_i - c_l\right\|_2\right) \tag{7}$$

where c_l represents the clustering center produced by unsupervised clustering algorithms, such as the k-means algorithm [3], from the labeled data set X^l. The value is normalized under the constraint $\sum_{l\in L} \bar{\pi}_{il} = 1$. If x_i is close to the clustering center, its prior probability $\bar{\pi}_{il}$ is large, and vice versa.

In order to keep the labeling and data affinity consistent, we should try to merge these two kinds of information before the label diffusion. For easy combination, we represent them in the same dimensional space:

$$W^{(1)} = W \tag{8}$$

$$W^{(2)} = \sum_{l\in L} \bar{\Pi}_l \bar{\Pi}_l^{\mathrm{T}} \tag{9}$$

where $\bar{\Pi}_l = [\bar{\pi}_{il}]_{N\times 1}$. $W^{(1)}$ represents data similarity in the feature space and $W^{(2)} \in \mathbb{R}^{N\times N}$ is a similarity matrix in the label space.

Borrowing ideas from the binary affinity fusion model in image retrieval [16], the automatic fusion strategy for data analysis is described as:

$$E(\Pi_l, \beta) = \sum_{h=1}^{H} \beta_h \left[\sum_{i,j=1}^{N} W_{ij}^{(h)} \left(\pi_{il} - \pi_{jl}\right)^2 + \lambda \sum_{i=1}^{N} d_i^{(h)} \left(\pi_{il} - \bar{\pi}_{il}\right)^2 \right] + \frac{1}{2}\gamma \|\beta\|_2^2, s.t \sum_{h=1}^{H} \beta_h = 1 \tag{10}$$

where H is the number of fusion components ($H = 2$), $d_i^{(h)} = \sum_{j=1}^{\mathrm{N}} W_{ij}^{(h)}$, $\beta = [\beta_h]_{H\times 1}$ ($0 \le \beta_h \le 1$), and γ is an adjusting parameter to control the influence of the last energy term. The fusion coefficients β_1 and β_2 can be automatically learned. Compared with the affinity fusion with diffusion model [16], the proposed model focuses on automatically determining the fusion coefficient for the information at the feature space and the label space, respectively, by a unary label diffusion framework.

Equation (10) can be reformulated as the matrix form:

$$E(\Pi_l, \beta) = \sum_{h=1}^{H} \beta_h \left[\Pi_l^{\mathrm{T}} L^{(h)} \Pi_l + \lambda \left(\Pi_l - \bar{\Pi}_l \right)^{\mathrm{T}} D^{(h)} \left(\Pi_l - \bar{\Pi}_l \right) \right] + \frac{1}{2} \gamma \|\beta\|_2^2 \tag{11}$$

where $L^{(h)} = D^{(h)} - W^{(h)}$ is the Laplacian matrix with $D^{(h)} = diag([d_1^{(h)}, ..., d_N^{(h)}])$.

Two variables are contained in Eq. (11) and their values are updated iteratively. Differentiating $E(\Pi_l, \beta)$ with respect to Π_l first, it has:

$$\Pi_l = (1 - \alpha) \left(\sum_{h=1}^{H} \beta_h D^{(h)} - \alpha \sum_{h=1}^{H} \beta_h W^{(h)} \right)^{-1} \sum_{h=1}^{H} \beta_h D^{(h)} \bar{\Pi}_l \tag{12}$$

Substituting Lagrange term into Eq. (11) and differentiating $E(\Pi_l, \beta)$ with respect to β_h, it has:

$$\beta_h = \frac{1}{H} + \frac{\sum_{h=1}^{H} M_h}{H\gamma} - \frac{M_h}{\gamma} \tag{13}$$

where $M_h = \Pi_l^{\mathrm{T}} \mathrm{L}^{(h)} \Pi_l + \lambda (\Pi_l - \bar{\Pi}_l)^{\mathrm{T}} D^{(h)} (\Pi_l - \bar{\Pi}_l)$. Based on the constraint $0 \le \beta_h \le 1$, we can derive $\gamma \ge |M_1 - M_2|$. Consequently, the detailed steps of the proposed algorithm are described as:

1. Initializing the parameters: α and γ
2. Setting the initial $\beta_h = 1/H$ and $f^{old} = [-1]_{N \times 1}$
3. Estimating prior probability $\bar{\Pi}_l$ with Eq. (7)
4. Computing $W^{(1)}$ and $W^{(2)}$ with Eqs. (8–9)
5. Estimating posterior probability Π_l with Eq. (12)
6. Updating the value of β with Eq. (13)
7. Computing the new labeling f^{new} with Eq. (6)
8. Checking the termination condition: if f^{new} equals f^{old}, stop; otherwise $f^{old} = f^{new}$, go to 5

4 Experiments

To evaluate the performance of the proposed semi-supervised clustering algorithm for financial risk prediction, two public credit approval risk data sets: German [17] and Australian [18] credit card application data sets, and one public Chinese growth enterprise market (GEM) dataset, are selected in this paper. There are common uncertainties with different forms of financing in these three datasets and the potential financial risks lead to the necessary risk prediction in order to minimize the defaults in advance. Therefore, the above datasets are suitable for our experiments. The compared clustering approaches include the popular k-means (KM) algorithm [3], the expectation–maximization (EM) algorithm [4], the repeated-bisection (RB) algorithm [19], the graph-partitioning (GP) algorithm [5], the density-based (DB) algorithm [20], the conventional semi-supervised learning (SSL) algorithm [13] and the state-of-the-art tensor product graph-based (TPG) algorithm [21].

Table 1 Comparison of performance on German credit card application data set

Methods	Precision	Purity	TPR	TNR
KM	0.30	0.70	0.78	0.22
EM	0.25	0.70	0.58	0.26
RB	0.26	0.70	0.62	0.24
GP	0.61	0.69	0.29	0.58
DB	0.30	0.70	**0.79**	0.23
SSL	0.48	0.71	0.48	0.78
TPG	0.55	0.72	0.52	0.23
Ours	**0.62**	**0.76**	0.56	**0.85**

It is hard to judge the performance of the algorithm with a single evaluation index. In this paper, four quantitative indexes: Precision, Purity [19], True Positive Rate (TPR) and True Negative Rate (TNR) are utilized to evaluate the compared methods. Precision represents the percentage of a cluster that contains positive objects, where in risk analysis, a positive class normally refers to bankrupt, fraudulent or erroneous activities. Purity is a simple measure of the number of correctly assigned objects in clustering. TPR measures in all positive instances how many instances are predicted to be positive category (correct prediction rate for positive instances), and TNR measures in all negative instances how many instances are predicted to be negative category (correct prediction rate for negative instances). Negative class is normal activities in risk analysis. More detailed definition of the above indexes can refer to this paper [10]. For the four evaluation indexes, a larger value represents a better clustering result.

Two controlling parameters α and γ are involved in the proposed algorithm and we set them to 0.3 and 10,000, respectively. For the semi-supervised algorithms SSL and the proposed method, 10% data is randomly selected as the labeled samples each time. We repeat the experiment 20 times and select the average performance as the final result.

The German credit card application data set was provided by UCI machine learning databases [17], which contain 1000 instances with 24 dimensional features and 1 label variable. The features correspond to the status of existing checking account, duration, credit history, purpose of credit application, credit amount, education level, employment status, personal status, other debtors, present residence, property type, age, job, and so on. The label variable describes whether an instance is accepted or declined, in which 70% instances are accepted and 30% instances are declined. Table 1 lists the Precision, Purity, TPR and TNR values of all compared methods in this data set, where the results of KM, EM, RB, GP and DB are reported in [10]. It can be seen that KM, EM, RB and DB obtain low precision values (below 0.3). Though GP obtains a high precision value 0.61, the TPR and TNR values are low. The proposed method obtains the highest precision, purity and TNR values among all the compared methods. Furthermore, compared with semi-supervised methods SSL and TPG, the proposed method obtains better performance in precision, purity, TPR and TNR, which validates the effectiveness of the proposed model in this data set.

The Australian credit card application data set was provided by a large bank and concerns consumer credit card applications [18], which contains 690 instances with 14 dimensional features and 1 label variable. To protect confidentiality of the data, attribute names and values have been changes to meaningless symbols. Attribute types include continuous, nominal with small number of values, and nominal with larger numbers of values [17].

Table 2 Comparison of performance on Australian credit card application data set

Methods	Precision	Purity	TPR	TNR
KM	0.78	0.85	0.92	0.79
EM	0.70	0.73	0.67	0.77
RB	**0.92**	0.81	0.73	**0.92**
GP	0.57	0.66	**0.95**	0.43
DB	0.81	0.82	0.78	0.85
SSL	0.82	0.58	0.27	0.82
TPG	0.83	0.85	0.84	0.69
Ours	0.84	**0.87**	0.88	0.87

Table 3 Comparison of performance on Chinese GEM dataset

Methods	2018			2017			2016		
	Precision	TPR	TNR	Precision	TPR	TNR	Precision	TPR	TNR
SSL	0.85	0.47	**0.98**	0.80	0.41	**0.95**	0.80	0.31	**0.97**
TPG	0.87	0.69	0.93	0.81	**0.74**	0.82	0.82	0.66	0.88
Ours	**0.89**	**0.71**	0.95	**0.84**	**0.74**	0.87	**0.83**	**0.71**	0.87

The label variable describes whether an instance is accepted or declined, in which 55.5% instances are accepted and 44.5% instances are declined. Table 2 lists the Precision, Purity, TPR and TNR values of all compared methods in this data set, where the results of KM, EM, RB, GP and DB are reported in [10]. It can be seen that RB obtains the highest precision and TNR values 0.92 and 0.92. By comparison, the proposed method obtains slightly lower precision and TNR values 0.88 and 0.91 than RB. However, the proposed method produces much higher purity and TPR values than RB. Compared with semi-supervised methods SSL and TPG, the proposed method obtains higher values in precision, purity, TPR and TNR, which validates the effectiveness of the proposed model in this data set. By comprehensive comparison with all the methods, our method obtains the best performance in Australian credit card application data set.

To specially verify the effectiveness among the semi-supervised approaches, we further chosen the Chinese GEM dataset provided by the Wind database[1] to conduct the comparison, from which we selected 360 companies from 2016 to 2018 with 24 dimensional features. The features correspond to the status of existing return on equity, return on total assets, net profit margin, gross profit margin, earnings per share, current ratio, quick ratio, equity ratio, receivables turnover ratio, current assets turnover, total assets turnover, working capital turnover rate, sales to cash ratio, operation safety rate, intangible assets ratio, and so on. Meeting one of the following conditions: 1) net assets are negative, 2) the net profit is negative and the net interest rate of the previous year is less than 10%, 3) the opinion category of audit report is qualified opinion or unable to express opinion, then an instance was identified as at risk. As a result, 75% instances are accepted and 25% instances are declined in this dataset. Table 3 lists the Precision, TPR and TNR values of

[1] https://www.wind.com.cn.

Table 4 Comparison results with and w/o fusion in German credit card application data set

	Precision	Purity	TPR	TNR
Without fusion	0.58	0.75	0.61	0.80
With fusion	0.62	0.76	0.56	0.85

Table 5 Comparison results with and w/o fusion in Australian credit card application data set

	Precision	Purity	TPR	TNR
Without fusion	0.83	0.86	0.88	0.85
With fusion	0.84	0.87	0.88	0.87

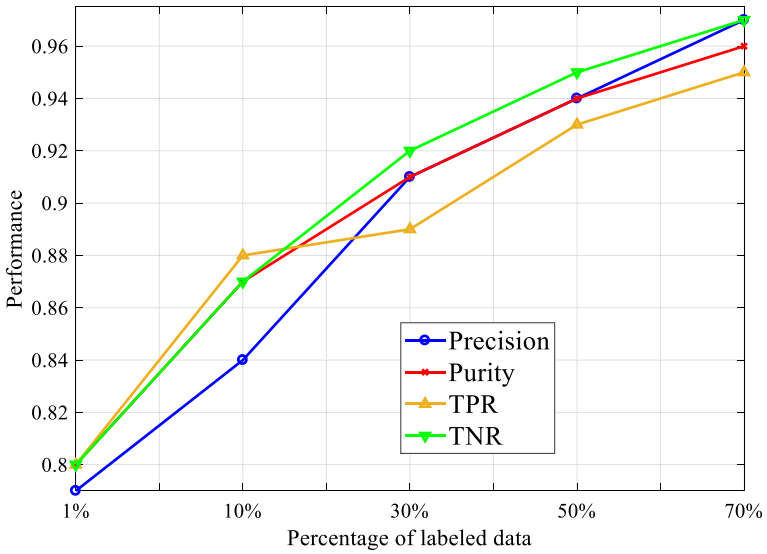

Fig. 1 Precision, Purity, TPR and TNR of the proposed method with different percentage of labeled data in Australian credit card application data set

the semi-supervised approaches SSL, TPG and the proposed method on the data from 2016 to 2018. Though SSL obtains the highest TNR values, its TPR values are very low, which implies many risky instances are wrongly classified into the risk-free category. It's obvious the proposed method produces the superior performance with the highest Precision and TPR values.

The similarity matrices $W^{(1)}$ (at the feature space) and $W^{(2)}$ (at the label space) are automatically merged by a label diffusion framework in this paper. To test the effectiveness of the proposed fusion strategy, Tables 4, 5 list the comparison results with and w/o fusion in German credit card application data set and Australian credit card application data set, respectively. From the quantitative comparisons in these two data sets, we can find that the proposed method with fusion produces higher Precision, Purity and TNR values than the approach without fusion.

The number of labeled data can affect the performance of the semi-supervised algorithms. Figure 1 shows the performance of the proposed algorithm with different

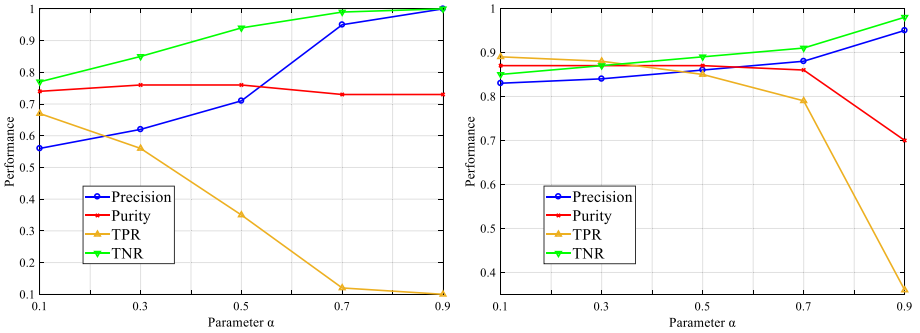

Fig. 2 Precision, Purity, TPR and TNR of the proposed method with different values of parameter α in German (left) and Australian (right) credit card application data sets

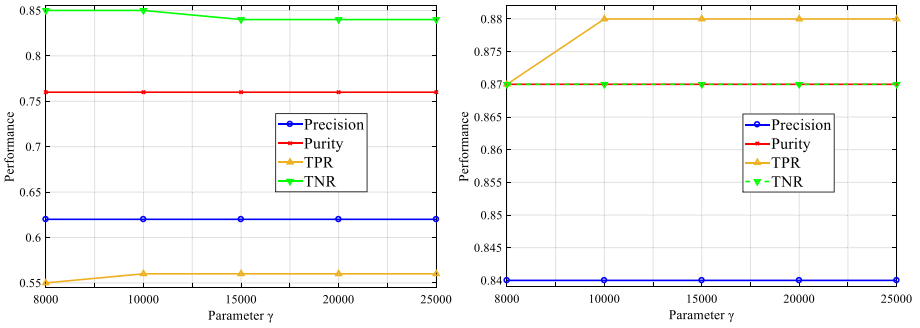

Fig. 3 Precision, Purity, TPR and TNR of the proposed method with different values of parameter γ in German (left) and Australian (right) credit card application data sets

percentage of labeled data in Australian credit card application data set. It can be seen that the values of Precision, Purity, TPR and TNR become higher along with the increase of the percentage of the labeled data. It can be also noticed that the values of Precision, Purity, TPR and TNR are around 0.8 with only 1% labeled data, which is still better than the most compared methods.

There are two controlling parameters α and γ involved in the proposed model. Parameter α is utilized to control the extent of label diffusion in Eq. (12). Figure 2 shows the performance curves with different values of α in German (left) and Australian (right) credit card application data sets. A too large value of α will lead to an over-smooth result that apart from the labeled data, the rest positive instances are easily misclassified as negative category. Therefore, from the curves, we can find that the values of Precision and TNR become higher and the values of Purity and TPR become lower when α increases. Parameter γ is utilized to control the fusion process in Eq. (13). As described before, the value of γ should be larger than $|M_1 - M_2|$ in each iteration in order to satisfy the constraint $0 \leq \beta_h \leq 1$. Therefore, we should assign a large but not too large value to parameter γ since a too large γ will impose an average fusion constraint. Figure 3 shows the performance curves when γ varies from 8000 to 25,000 in German (left) and Australian (right) credit card application data sets. It can be seen that in this interval values, the performance is not sensitive to the change of γ. In this paper, the value of γ can be loosely set to 10,000.

Table 6 Algorithm complexity and average running times on the three datasets for SSL, TPG and the proposed method

Methods	SSL	TPG	Ours
Complexity	$\mathcal{O}(N^2)$	$\mathcal{O}(N^{2.4})$	$\mathcal{O}(N^2)$
Runtime (s)	0.1	0.4	0.9

5 Algorithm Complexity Analysis

Table 6 lists the algorithm complexity and the average running times of the semi-supervised approaches SSL, TPG and the proposed method on an Intel Core i7-7700 K CPU with 16 GB memory running at 4.20 GHz in MATLAB R2017a. The algorithm complexity of SSL and the proposed method are both $\mathcal{O}(N^2)$, which mainly focuses on the inversion operation of a similarity matrix. In the algorithm implementation, the multiplication of the inversion matrix by a single vector can be efficiently solved by the MATLAB division operator '\'. The algorithm complexity of TPG is $\mathcal{O}(N^{2.4})$ using the Coppersmith-Winograd algorithm, which mainly focuses on the iterative matrix product operation for a higher-order tensor product graph optimization. Limited by the iterative optimization for Π and β, the average running time of the proposed method is 0.9 s which is slightly higher than SSL and TPG.

6 Conclusion

In this paper, a semi-supervised clustering algorithm is proposed for financial risk analysis. In order to improve the performance of the conventional semi-supervised model, we first estimate the label prior probability from the labeled data, and this can be regarded as a diffusion process from the local'hard' labels to the global'soft' probabilities. Then a label diffusion model is designed to propagate the prior probabilities from labeled data to unlabeled data. Furthermore, to make the structures of data affinity and labeling more consistent, the similarity matrices in the feature space and label space are adaptively merged based on the label diffusion framework. The energy function can be effectively solved by an iterative optimization strategy. Experimental results on three public datasets demonstrate that the proposed method can obtain better performance than the compared methods.

Appendix

The derivation process for Eq. (12)

Differentiating $E(\Pi_l, \beta)$ with respect to Π_l, it has:

$$\frac{\partial E(\Pi_l, \beta)}{\Pi_l} = \sum_{h=1}^{H} \beta_h [L^{(h)} \Pi_l + \lambda D^{(h)} (\Pi_l - \bar{\Pi}_l)] \tag{14}$$

By setting Eq. (14) to zero, we can obtain:

$$\sum_{h=1}^{H} \beta_h [\frac{D^{(h)} - W^{(h)}}{\lambda} + D^{(h)}]\Pi_l = \sum_{h=1}^{H} \beta_h D^{(h)} \bar{\Pi}_l \tag{15}$$

Since $\lambda = (1-\alpha)/\alpha$ (defined in Eq. (3)), we can derive:

$$\frac{1}{1-\alpha} \sum_{h=1}^{H} \beta_h (D^{(h)} - \alpha W^{(h)})\Pi_l = \sum_{h=1}^{H} \beta_h D^{(h)} \bar{\Pi}_l \tag{16}$$

Then we can obtain:

$$\Pi_l = (1-\alpha)\left(\sum_{h=1}^{H} \beta_h D^{(h)} - \alpha \sum_{h=1}^{H} \beta_h W^{(h)}\right)^{-1} \sum_{h=1}^{H} \beta_h D^{(h)} \bar{\Pi}_l \tag{17}$$

The derivation process for $\gamma \geq |M_1 - M_2|$

Based on the condition $0 \leq \beta_h \leq 1$, from Eq. (13) we can derive:

$$0 \leq \frac{1}{H} + \frac{\sum_{h=1}^{H} M_h}{H\gamma} - \frac{M_h}{\gamma} \leq 1 \tag{18}$$

$$\gamma \geq \max_{h \in H} \left\{ \sum_{h=1}^{H} M_h - HM_h, HM_h - \sum_{h=1}^{H} M_h \right\} \tag{19}$$

Since the merge of two similarity matrices ($H = 2$) is explored in this paper, it has:

$$\gamma \geq |M_1 - M_2| \tag{20}$$

Funding This work was supported in part by Science and Technology Development Project of Suzhou under Grant SR201742, in part by the Jiangsu Planned Projects for Postdoctoral Research Funds, and in part by the Natural Science Foundation of Jiangsu Province, China, under Grant BK20180458.

References

1. Tay FEH, Cao LJ (2002) ε-Descending support vector machines for financial time series forecasting. Neural Process Lett 15(2):179–195
2. Arratia A, Belanche LA, Fábregues L (2019) An evaluation of equity premium prediction using multiple kernel learning with financial features. Neural Process Lett 52:117–134
3. MacQueen JB (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, pp. 281–297
4. Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J Roy Stat Soc Ser B Meth 39(1):1–38
5. Abou-Rjeili A, Karypis G (2006) Multilevel algorithms for partitioning power-law graphs. In: IEEE International Parallel and Distributed Processing Symposium (IPDPS)

6. Brockett P, Xia X, Derrig R (1998) Using Kohonen's self organizing feature map to uncover automobile bodily injury claims fraud. J Risk Insur 65(2):245–274
7. Cox E (1995) A fuzzy system for detecting anomalous behaviors in healthcare provider claims. In: Goonatilake S, Treleaven P (eds) Intelligent systems for finance and business. Wiley, New York, pp 111–134
8. Moreau Y, Lerouge E, Verrelst H, Vandewalle J, Stormann C, Burge P (1999) BRUTUS: a hybrid system for fraud detection in mobile communications. In: Proceedings of the European Symposium Artificial Neural Networks, pp. 447–454
9. Yeo AC, Smith KA, Willis RJ, Brooks M (2001) Clustering technique for risk classification and prediction of claim costs in the automobile insurance industry. Intell Syst Acc Finance Manage 10(1):39–50
10. Kou G, Peng Y, Wang G (2014) Evaluation of clustering algorithms for financial risk analysis using MCDM methods. Inf Sci 275:1–12
11. Nachev A, Hill S, Barry C, Stoyanov B (2010) Fuzzy, distributed, instance counting, and default artmap neural networks for financial diagnosis. Int J Inform Technol Decis Making 9(6):959–978
12. Ngoc MT, Park DC (2018) Centroid neural network with pairwise constraints for semi-supervised learning. Neural Process Lett 48(3):1721–1747
13. Zhou D, Bousquet O, Lal TN, Weston J, Scholkopf B (2004) Learning with local and global consistency. Adv Neural Inf Process Syst 16(4):321–328
14. Wang Y, Meng Y, Fu Z et al (2017) Towards safe semi-supervised classification: adjusted cluster assumption via clustering. Neural Process Lett 46(3):1031–1042
15. Ma X, Gao L, Yong X, Lidong Fu (2010) Semi-supervised clustering algorithm for community structure detection in complex networks. Physica A 389:187–197
16. Bai S, Zhou Z, Wang J, Bai X, Latecki LJ, Tian Q (2017) Ensemble diffusion for retrieval. In: Proceedings of the IEEE International conference on computer vision pp. 774–783
17. Frank A, Asuncion A (2010) UCI Machine Learning Repository, Irvine, CA: University of California, School of Information and Computer Science <http://archive.ics.uci.edu/ml>
18. Quinlan JR (1993) C45: Programs for machine learning. Morgan Kaufmann, San Franciso
19. Zhao Y, Karypis G (2001) Criterion functions for document clustering: experiments and analysis, Technical Report TR 01-40, Department of Computer Science, University of Minnesota
20. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco
21. Bai S, Bai X, Tian Q, Latecki LJ (2017) Regularized diffusion process for visual retrieval. Proc AAAI Conf Artif Intell 31:3967–3973