Credit Analysis Using a Combination of Fuzzy Robust PCA and a Classification Algorithm

Onesfole Kurama, Pasi Luukka and Mikael Collan

Abstract Classification is a key part of credit analysis and bankruptcy prediction and new powerful classification methods coming from artificial intelligence are often applied. Most often classification methods require pre-processing of data. This paper presents a two-part classification process that combines a pre-processing step that uses fuzzy robust principal component analysis (FRPCA) and a classification step. Combinations of three FRPCA algorithms and two different classifiers, similarity classifier and fuzzy k-nearest neighbor classifier, are tested to find the combination that gives the most accurate mean classification result. Tests are run with a small Australian credit data set that can be considered "rough" and to require "robust" methods, due to the small number of observations. The created principal components are used as inputs in the classification methods. Results obtained indicate a mean classification accuracies of over 80 % for all combinations. It becomes clear that parameters of the used methods clearly affect the results and emphasis is put on finding suitable parameters.

Keywords Credit decision-making · Australian credit screening dataset · Fuzzy robust PCA · Similarity classifier · Fuzzy k-nearest neighbor

1 Introduction

Business and academic communities have paid a lot of attention to bankruptcy prediction and credit scoring within the fields of accounting and finance. Many methods that are commonly used in bankruptcy prediction stem from the broad

O. Kurama (✉) · P. Luukka
Laboratory of Applied Mathematics, Lappeenranta University of Technology, P.O Box 20, 53851 Lappeenranta, Finland
e-mail: pasi.luukka@lut.fi

P. Luukka · M. Collan
School of Business and Management, Lappeenranta University of Technology, P.O Box 20, 53851 Lappeenranta, Finland
e-mail: mikael.collan@lut.fi

J. Gil-Aluja et al. (eds.), *Scientific Methods for the Treatment of Uncertainty in Social Sciences*, Advances in Intelligent Systems and Computing 377,
DOI 10.1007/978-3-319-19704-3_2

discipline of artificial intelligence (AI). The problem is real, failed credit decision-making in financial institutions may run them into financial difficulty, while simultaneously it is not good business to not lend to good borrowers – the accuracy of the ability of financial institutions to *select good borrowers* from the group of potential borrowers is matter of competitive advantage.

The grand assumption in modern bankruptcy prediction and credit analysis, since the late 1960s, is that financial information, extracted from financial statements, such as financial ratios, contains large amounts of relevant information (Altman 1968, Beaver 1968). A variety of information about the status of client companies is available from financial and accounting data and in addition to this "hard" data, information may be based on normative judgments of experts. When it comes to individual borrowers the data may be in a different format, but the same principles for credit analysis hold. Results from these analyses can also be used as ''early warning signs" that can be acted on to prevent business failure by stakeholder intervention (such as, forcing mergers of distressed firms, liquidation, or reorganization). Another, perhaps a more commonly used, way of using the results is to use them in credit decision-making that is, in the credit approval process (Dimitras et al. 1996).

Bankruptcy prediction and credit analysis in general can be thought of as a classification problem, where by using multiple inputs, we try to classify potential lenders into two categories: ones with a high bankruptcy risk and the ones with no or low bankruptcy risk that is, to the group to grant a loan to and the group not to grant a loan to. As the information systems era has brought with it large data sets of relevant data for credit analysis, both with regards to companies and individuals as clients, the main focus of research has shifted to using modern data mining techniques and a variety of these methods has been applied to bankruptcy prediction, for overviews on different methods see, e.g., (Back et al. 1996; Dimitras et al. 1996; Huang et al. 2004; Lensberg et al. 2006; Kumar and Ravi 2007). It has been shown that data mining and AI techniques can outperform classical statistical approaches in bankruptcy prediction (Zhang et al. 1999; Min and Lee 2005; Shin et al. 2005; Kumar and Ravi 2007). Recently, models that incorporate more than just client specific that is, "environmental" information in the bankruptcy prediction process have started to emerge see, e.g., (Karas and Reznakova 2014; López Iturriaga and Pastor Sanz 2015). Bankruptcy prediction methods from fuzzy and artificial intelligence have recently grown in popularity see, e.g., (Zopounidis et al. 1999; Ahn et al. 2000; Mckee 2000; Quek et al. 2009; Terceno and Vigier 2011).

In this paper we use a two-step data classification procedure for credit analysis, where in the first step, the data undergoes a fuzzy robust principal component analysis (FRPCA) in line with (Yang and Wang 1999) and by using (testing) three different FRPCA algorithms. The resulting principal components are then used in the second step as inputs in classification. In the classification step we use two different classification methods, the fuzzy K-nearest neighbor classifier (FKNN) by (Keller et al. 1985) and the similarity classifier presented in (Luukka et al. 2001, Luukka and Leppälampi 2006). A blueprint of the process is visible in Fig. 1.

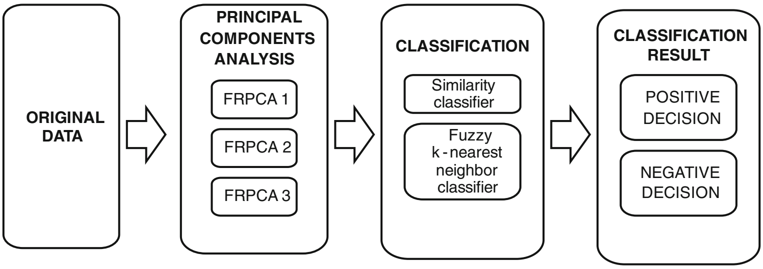

Fig. 1 Blueprint of the two-step classification process with the tested methods indicated. The known correct values are compared to the resulting classification

Combinations of the three first-step FRPCA methods and the two second-step classification methods are all tested for parameter values to find combinations that result in good classification accuracy. A two-step system with these characteristics is, to the best of our knowledge, a new approach.

The next section presents the details of the two-step classification procedure by first going through the three different fuzzy robust principal component analysis algorithms and how they are used, the two classification algorithms are then presented in detail. This is followed by a walk-through of classification results with the procedure, obtained with test data. The paper closes with discussion and conclusions.

2 Classification Procedure

In this section we first briefly review the applied methods, then summarize how they are used together, and then the "combinations" are compared. Methods include three fuzzy robust principal component analysis methods (Yang and Wang 1999) in forming robust linear combinations from the data and the fuzzy k-nearest neighbor (Keller et al. 1985) and the similarity classifier (Luukka and Leppälampi 2006) as classification methods. The task is to find the best combination through selecting the most suitable FRPCA method with the proper principal components, and parameter values and to run with the two classifiers.

2.1 Fuzzy Robust Principal Component Analysis (FRPCA)

In standard PCA, principal components are often influenced by outliers that can in a bad case distort the weights applied in forming these linear combinations. This distortion can affect the accuracy of the results, or even lead to misleading results. The underlying idea in fuzzy robust PCA methods is that these outliers are removed

in formation of the weights, and in this way we can produce more robust principal components. The identification and possible removal of outliers is one of the main topics in robustness theory. FRPCA algorithms are designed to detect outliers so that they are removed before weight computations are made, this makes them often yield better results (accuracy) than conventional PCA algorithms. In this work we use FRPCA algorithms proposed by Yang and Wang (Yang and Wang 1999). Next FRPCA is represented shortly in an algorithmic form, allowing also a simplified way to present the different algorithm versions that we apply.
FRPCA1 algorithm:

Step 1: Initially set the iteration count to $t = 1$, iteration bound T, learning coefficient $\alpha_0 \in (0, 1]$, soft threshold λ to a small positive value, and randomly initialize the weight ϕ.
Step 2: While t is less than T, perform the next steps 3–9.
Step 3: Compute $\alpha_t = \alpha_0\ (1 - t/T)$, set $i = 1$ and $\sigma = 0$.
Step 4: While i is less, than n perform steps 5–8.
Step 5: Compute $y = \phi^T\ x_i$, $v = y\phi$, and $u = \phi^T\ v$
Step 6: Update the weight with: $\varnothing^{new} = \varnothing^{old} + \boldsymbol{\alpha}_t\boldsymbol{\beta}(x_i)[\mathbf{y}(x_i - \mathbf{v}) + (\mathbf{y} - \mathbf{u})x_i]$
Step 7: Update the temporary count, $\sigma = \sigma + \theta_1(x_i)$
Step 8: Add 1 to i.
Step 9: Compute $\lambda = \sigma/n$ and add 1 to t.

FRPCA2 algorithm:

The procedures above are followed, except for steps 6 and 7, where we change the way we update the weight and the count:

Step 6: Update the weight with: $\varnothing^{new} = \varnothing^{old} + \alpha_t\beta(x_i)\left(x_i\mathrm{y} - \frac{\varnothing}{\varnothing^T\varnothing}y^2\right)$
Step 7: Update the temporary count: $\sigma = \sigma + \theta_2(x_i)$

FRPCA3 algorithm:

Follow the same procedure as in FRPCA1, except for steps 6 and 7:

Step 6: Update the weight with: $\varnothing^{new} = \varnothing^{old} + \boldsymbol{\alpha}_t\boldsymbol{\beta}(x_i)(x_i\mathbf{y} - \varnothing\boldsymbol{y}^2)$
Step 7: Update the temporary count: $\boldsymbol{\sigma} = \boldsymbol{\sigma} + \boldsymbol{\theta}(x_i)$

2.2 Fuzzy k-Nearest Neighbor Classifier

The k-nearest neighbor algorithm is used to assign patterns of unknown classification to the class of the majority of its k-nearest neighbors of known classification (Keller et al. 1985). The classical model assumes that each sample belongs to a specific (single) class, the fuzzy extension however allows a sample to belong to more than one class simultaneously. Membership values are assigned to the samples in relation to their contribution to the classification process. Starting from the closest neighbor, the algorithm runs until all the clusters in the neighborhood are covered. Initialization determines to which local minima the algorithm will terminate. Thus the algorithm will always search for the labeled patterns in the nearest class (Keller et al. 1985). To compute membership values for different samples, the expression below is useful:

$$\mu_i(x) = \frac{\sum_{\mathrm{j}=1}^{\mathrm{K}} \mu_{i,j}\left(1/\left\|x - x_j\right\|^{2/(n-1)}\right)}{\sum_{\mathrm{j}=1}^{\mathrm{K}}\left(1/\left\|x - x_j\right\|^{2/(n-1)}\right)} \tag{1}$$

where $\mu_{i,j}$ is the membership in the i^{th} class of the j^{th} sample of the labeled pattern and n_1 is the scaling factor for fuzzy weights.

2.3 Similarity Classifier

Similarity classifier was introduced in (Luukka and Leppälampi 2006). Main components in the classifier are the computation of ideal vectors for each class and computing similarity between samples and the ideal vectors. Ideal vectors, v_i for class i are given as:

$$v_i = (v_{i1}, v_{i2}, \ldots v_{in}) \tag{2}$$

In current computations we compute ideal vectors by using the generalized mean, by fixing a parameter m, and the ideal vector is then computed as:

$$v_i(d) = \left(\frac{1}{\#X_i} \sum_{\mathbf{x}\in X_i} x_d^m\right)^{1/m}, \quad \forall d = 1, 2, \ldots, n \tag{3}$$

Generalized Łukasiewicz similarity (Luukka and Leppälampi 2006) have been found to be useful in comparisons between samples and ideal vectors. A weight parameter w_d, which varies feature by feature and depends on feature importance, is also applied and makes the resulting similarity:

$$S\langle x, y\rangle = \frac{1}{n}\sum_{d=1}^{n} w_d\left(1 - \left|x_d^p - \mathbf{v}_d^p\right|\right)^{m/p} \tag{4}$$

In experiments with real data it is important to calibrate the parameters m and p, see the Eq. (4), since they affect the classification accuracy.

3 Classification Results with an Australian Credit Screening Dataset

This section presents the results obtained by using the proposed classification process, by using the "Australian credit screening dataset" as a basis. The data set is available for download at the UCL machine learning data repository (URL: http://archive.ics.uci.edu/ml/). Key properties of the data are summarized in Table 1.

Table 1 Summary of properties of Australian credit screening data set

Number of variables:	14(continuous: 6, categorical:8)
Number of classes:	2
Number of samples:	690
Class distribution:	Bad credit: 307, Good credit 383

Emphasis is put on the performance of the classifiers, as well as, on the pre-processing procedures utilized. Parameters used have also been "optimized" in the sense that they have been pre-tested to obtain acceptable ranges for each parameter.

3.1 Results from the Australian Credit Scoring Data with FRPCA and the Similarity Classifier

In this experiment, ideal vectors are computed for each class across all features and compared with the rest of the samples by calculating the similarities between them. Generalized Łukasiewicz based similarity, with generalized mean is used for computation of the similarities. Matlab software has been utilized in designing and implementing the algorithms. Since not all the principal components necessarily hold important information related to classification process, we study how many principal components need to be considered, in order to obtain the best accuracy. This is straight-forward, because the first principal components should hold the most information, and in an ideal scenario the last component should mostly consist of noise. This way we also get a ranking order for principal components. Different FRPCA-algorithms (FRPCA1, FRPCA2, and FRPCA3) are used in forming these principal components. Random division of the data to training and testing sets is done several times (30 times in this case) – that is we run 30 runs with different sets selected from the data. Mean classification accuracies are computed. In these experiments we set the fuzziness variable set to n = 2.

The best mean accuracy of 82.92 % is reached with FRPCA3, when all principal components (14) are used. Since the classification accuracy improves until when all principal components are used, it seems that important information is "still left" even in the last principal component (PC). The other two methods also perform well: FRPCA2 gives a mean classification accuracy of 81.84 % (with 13 PC), and

FRPCA1 results in 81.73 % (also with 13 PC). Figure 2 shows how the number of principal components affects the mean classification accuracies with the similarity classifier.

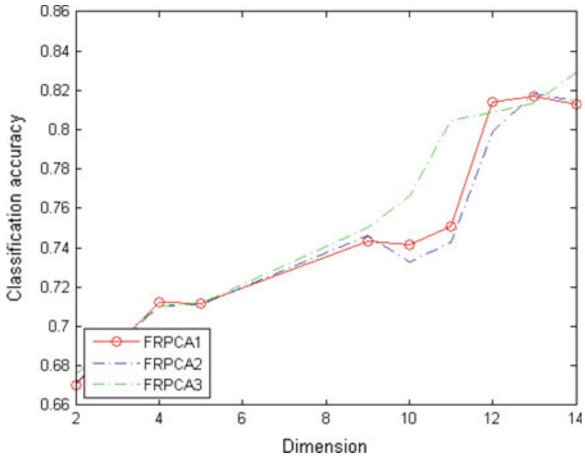

Fig. 2 Mean classification accuracies with respect to the number of principal components used for FRPCA1, FRPCA2, and FRPCA3

In a follow-up experiment we examine the effect of using different fuzziness variable *n* values for FRPCA algorithms. The best choice of the fuzziness parameter value for FRPCA1 was found to be around $n = 1.8$ and corresponding to a classification accuracy of 83.22 %. With the fuzziness parameter value around $n = 1.2$ FRPCA2 has a classification accuracy of 82.09 %, and FRPCA3 has an accuracy of 82.97 %. The experiment shows that the highest mean classification accuracy can be reached with FRPCA1. In addition to the parameter values of the FRPCA algorithms also the parameters in the similarity classifier have an effect on the mean classification accuracy of the process, (parameters m and p) were introduced shortly in Eq. (4).

A way to locate the best similarity classifier parameter values is to compute mean classification accuracies for a grid of suitable parameter values. By using figures, where we plot the mean classification accuracy with regards to different parameter values, we can by visual inspection find the best "area" for these parameters.

Example of this is given in Fig. 3, where we have plotted the variance, mean, minimum, and maximum classification accuracies with regards to parameters m and p, for FRPCA1 with 13 PC used, and with the fuzziness parameter set to n = 1.8. By visual inspection one can see that the best value of p should be chosen from within the interval [0.5, 3], with p = 2 seeming to be the best choice. The value of m should be taken from the interval [0.3, 6] with m = 5 looking to be the best. The approximately best parameter pair $(p, m) = (2, 5)$ is chosen.

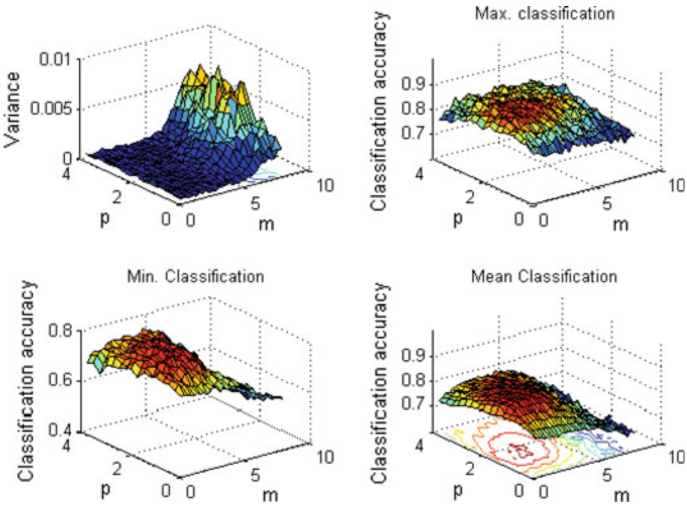

Fig. 3 Plots showing the location of best values of the parameters p and m with FRPCA1, with the fuzziness parameter fixed at $n = 1.8$ and 13PC used

3.2 *Results from the Australian Credit Scoring Data with FRPCA and Fuzzy k-Nearest Neighbor Classifier*

The experiment is set up as the first one, the number of principal components used for the best mean classification result and a suitable number of nearest neighbors are investigated. We also look into the fuzziness parameter n value for the FRPCA algorithms and the scaling factor value for fuzzy weights n_1 used in the FKNN classifier, see Eq. (3). Table 2 summarizes the results for the mean classification accuracy with different numbers of principal components used and the number of nearest neighbors, when using FRPCA2. The same analysis was made for all three FRPCA algorithms, but using FPRCA 2 with 14 PC and k = 13 nearest neighbors resulted in the best mean classification result of 84.55 %. These results were obtained with the fuzziness variable n fixed at 2.

Table 2 Mean classification results using FRPCA2 with FKNN classifier for principal components PC and the number of nearest neighbors K. Fuzziness variables are fixed to $n = n_1 = 2$. The best mean classification accuracy is given in **boldface**

K PC	8	9	10	11	12	13	14
14	0.8366	0.8380	0.8402	0.8414	0.8421	0.8417	0.8427
13	0.8392	0.8390	0.8419	0.8437	0.8437	0.8446	**0.8455**
12	0.8390	0.8410	0.8413	0.8425	0.8420	0.8418	0.8433
11	0.8344	0.8351	0.8363	0.8381	0.8395	0.8410	0.8422
10	0.8104	0.8101	0.8120	0.8138	0.8153	0.8167	0.8173

It can generally be seen that the mean classification accuracy improves with increase in both the number of PC used and the number of nearest neighbors. Next, the proper values for the fuzziness parameter *n,* for FRPCA algorithms is studied. The best classification result, 85.00 %, is obtained by using FRPCA2 with a fuzziness parameter value $n = 2.5$. The fuzziness parameter value for FKNN was fixed to $n_1 = 2$. The best classification accuracy for FRPCA1 is 84.87 % and was obtained, with $n = 2.2$, FRPCA3 reaches a classification accuracy of 84.82 %, when $n = 3.5$. For finding suitable FKNN classifier fuzziness parameter n_1 values we fixed FRPCA variable n values to those that return the best classification results. The mean classification accuracy obtained for different n_1 parameter values is shown in Table 3.

Table 3 Mean classification accuracies for several FKNN parameter n_1 values. Values of n fixed for FRPCA1 $n = 2.2$, for FRPCA2 $n = 2.5$, and for FRPCA3 $n = 3.5$. The best result in **boldface**

n_1	FRPCA1	FRPCA2	FRPCA3
2.0	0.8481	0.8485	0.8491
2.8	0.8515	0.8552	0.8565
3.0	0.8527	0.8535	0.8566
4.0	0.8516	0.8531	0.8579
4.5	0.8521	**0.8593**	0.8574
5.0	0.8546	0.8544	0.8562

The best mean classification accuracy of 85.93 % is achieved with FRPCA2, when $n_1 = 4.5$ and $n = 2.5$. FRPCA3 reaches the accuracy of 85.79 %, when $n_1 = 4.0$ and $n = 3.5$ and FRPCA1 85.46 %, when $n_1 = 5.0$ and $n = 2.2$. The mean classification accuracy did not improve with $n_1 > 5.0$.

3.3 Short Discussion About the Results

We ran tests on the proposed credit analysis process by using three FRPCA algorithms with two classification algorithms and tuned the parameters of these methods for better results. Experiments with the similarity classifier show that combining it with FRPCA1 gives the best mean classification accuracy of 83.22 %, when $n = 1.8$, and p & m are chosen around 2 and 5 respectively. Experiments with the FRPCA2 paired with FKNN classifier produced the best results, with a mean classification accuracy of 85.93 % reached. Table 4 summarizes the results.

Table 4 Summary of the results obtained with suitable choices for fuzziness parameters n (FRPCA) and n_1 (FKNN), parameters p (Łukasiewicz structure) and m (generalized mean), number of principal components D, and the number of nearest neighbors K, for the two classifiers with all FRPCA methods used

Method	Similarity classifier		FKNN classifier	
FRPCA1	p = 2,	m = 5	n_1 = 2.2	K = 14
n	1.8		2.2	
D	13		13	
Accuracy	83.22 %		85.46 %	
FRPCA2	p = 1.9,	m = 3.5	n_1 = 4.5	K = 14
n	1.2		2.5	
D	13		13	
Accuracy	82.09 %		**85.93 %**	
FRPCA3	p = 2.1,	m = 2.5	n_1 = 4.0	K = 14
n	1.2		3.5	
D	14		14	
Accuracy	82.97 %		85.79 %	

Classification accuracies of above 80 % are achieved with all FRPCA and classifier combinations, when parameters are tuned. The choice of FRPCA-parameters is significant for the results, as is the number of principal components used in the classifiers. Furthermore, tuning of the classifier parameter values also affects the mean accuracy of the classification.

4 Summary and Conclusions

In this article we have tackled the problem of classifying potential borrowers with a two-step process that first uses a fuzzy robust principal components analysis and then a classifier that classifies the borrowers based on the principal components. Three FRPCA algorithms were tested for the first step and two classifier algorithms for the second step. All combinations of steps one and two were tested for mean classification accuracy, by using an Australian credit screening data set. The results show that with all combinations, when parameter values are set correctly, an above 80 % classification accuracy can be reached, in fact, the best accuracy obtained was 85.93 %. It is to be noted that the used data set is not very large and contains only 690 observations, making the problem a “robust” one. This is why we feel that the fit of the robust FRPCA method is a valid choice for the first-step of the process. What we can take home from the results of this paper is that in a two-step setup like the one presented, tuning of parameters influences the results and should be emphasized when designing and using systems.

References

Ahn, B., Cho, S., Kim, C.: The integrated methodology of rough set theory and artificial neural network for business failure prediction. Expert Syst. Appl. **18**, 65–74 (2000)

Altman, E.I.: Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. J. Finance **23**, 589–609 (1968)

Back, B., Laitinen, T., Sere, K.: Neural networks and genetic algorithms for bankruptcy predictions. Expert Syst. Appl. **11**, 407–413 (1996)

Beaver, W.H.: Market prices, financial ratios, and the prediction of failure. J. Account. Res. **6**, 179–192 (1968)

Dimitras, A.I., Zanakis, S.H., Zopounidis, C.: A survey of business failures with an emphasis on prediction methods and industrial applications. Eur. J. Oper. Res. **90**, 487–513 (1996)

Huang, Z., Chen, H., Hsu, C.J., Chen, W.H., Wu, S.: Credit rating analysis with support vector machines and neural networks: a market comparative study. Decis. Support Syst. **37**, 543–558 (2004)

Karas, M., Reznakova, M.: To what degree is the accuracy of a bankruptcy prediction model affected by the environment? the case of the Baltic States and the Czech Republic. In: Gimžauskienė, E. (ed.) 19th International Scientific Conference Economics and Management 2014 ICEM-2014, vol. 156, pp. 564–568. Elsevier, Riga, Latvia (2014)

Keller, J., Gray, M., Givens, J.R.: A fuzzy K nearest neighbor algorithm. IEEE Trans. Syst. Man Cybern. **15**, 580–585 (1985)

Kumar, P., Ravi, V.: Bankruptcy prediction in banks and firms via statistical and intelligent techniques—a review. Eur. J. Oper. Res. **180**, 1–28 (2007)

Lensberg, T., Ellifsen, A., McKee, T.E.: Bankruptcy theory development and classification via genetic programming. Eur. J. Oper. Res. **169**, 677–697 (2006)

López Iturriaga, F.J., Pastor Sanz, I.: Bankruptcy visualization and prediction using neural networks: a study of U.S. commercial banks. Expert Syst. Appl. **42**, 2857–2869 (2015)

Luukka, P., Leppälampi, T.: Similarity classifier with generalized mean applied to medical data. Comput. Biol. Med. **36**, 1026–1040 (2006)

Luukka, P., Saastamoinen, K., Könönen, V.: A classifier based on the maximal fuzzy similarity in the generalized Lukasiewicz-structure. In: FUZZ-IEEE (ed.), pp. 195–198. IEEE, Melbourne, Australia (2001)

Mckee, T.: Developing a bankruptcy prediction model via rough sets theory. Int. J. Intell. Syst. Account. Finance Manage. **16**, 159–173 (2000)

Min, J.H., Lee, Y.C.: Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. Expert Syst. Appl. **28**, 603–614 (2005)

Quek, C., Zhou, D., Lee, C.: A novel fuzzy neural approach to data reconstruction and failure prediction. Intell. Syst. Account. Finance. Manage. **16**, 165–187 (2009)

Shin, K.S., Lee, T.S., Kim, H.J.: An application of support vector machines in bankruptcy prediction model. Expert Syst. Appl. **28**, 127–135 (2005)

Terceno, A., Vigier, H.: Economic-financial forecasting model of businesses using fuzzy relations. Econ. Comput. Econ. Cybern. Stud. Res. **2011**, 185–203 (2011)

Yang, T.N., Wang, S.D.: Robust algorithms for principal component analysis. Pattern Recogn. Lett. **20**, 927–933 (1999)

Zhang, G., Hu, M.Y., Patuwo, B.E., Indro, D.C.: Artificial neural networks in bankruptcy prediction: general framework and cross-validation analysis. Eur. J. Oper. Res. **116**, 16–32 (1999)

Zopounidis, C., Slowinski, R., Doumpus, M., Dimitras, A., Susmaga, R.: Business failure prediction using rough sets: a comparison with multivariate analysis techniques. Fuzzy Econ. Rev. **4**, 3–33 (1999)

Springer

http://www.springer.com/978-3-319-19703-6

Scientific Methods for the Treatment of Uncertainty in Social Sciences
Gil-Aluja, J.; Terceño-Gómez, A.; Ferrer-Comalat, J.C.; Merigó-Lindahl, J.M.; Linares-Mustaros, S. (Eds.)
2015, XVI, 444 p. 96 illus., Softcover
ISBN: 978-3-319-19703-6