MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **70** (2013) 29-44

ISSN 0340 - 6253

Classification Ability of Self Organizing Maps in Comparison with Other Classification Methods

Mahdi Vasighi[1] and Mohsen Kompany-Zareh[*2]

[1] Department of Computer Science and Information Technology, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan 45137-66731, Iran.
[2] Department of Chemistry, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan 45137-66731, Iran.

(Received March 6, 2012)

Abstract

In this study, performance of different classification methods based on self organizing maps (SOMs), including counter propagation network (CPN), supervised kohonen networks (SKN) and XY-fused network (XYF), were compared to linear discriminant analysis (LDA), learning vector quantization (LVQ) and support vector machine (SVM). Performance of classification was statistically investigated using both simulated and real data sets according to percent of correct classified samples (non-error rate) in the test set. Effect of selection of calibration samples on model stability and performance was investigated. There were several adjustable parameters in each modeling techniques (except LDA) which were optimized for better comparison. Each simulated dataset regenerated 50 times and performance of classification was computed for each method to obtain a population of results. Obtained results showed that the distribution and structure of the samples in data space is an important factor influences on the relative performance of classification methods. CPN and SVM performed better in the cases with nonlinear discriminant boundary but for overlapped classes with normal distribution, performance of LDA was slightly better than other methods. In addition, CPN showed a comparable performance and stability in comparison with SVM which known as a powerful classification method.

1. Introduction

Classification is one of the major subdivisions of pattern recognition methods and refers to techniques in which a priori knowledge about the category membership of samples is used for building a classifier. The classification model is developed on a training set of samples with

known classes [1]. Subsequently, the model performance is evaluated using a validation set comparing predictions with known categories.

In recent years, many advanced classification techniques have been widely used within chemistry, biology, pharmaceutical and food sciences [2-4]. Comparison of different classification methods can be found in many chemometrics literatures [5-7,3]. The question of which classification approach is suitable for a specific study (dataset) is not easy to answer. Generally, depending on the method chosen and the nature of data, different results may be obtained. Therefore, it is important to find the method with highest predictability and know if the difference between available methods is statistically significant or not [5]. Accordingly, quantitative criteria like prediction rate (non-error rate) or misclassification percentage on a test set (or cross-validated set) are frequently used to compare performance of the built models [8]. Among the numerous classification techniques, artificial neural networks (ANNs) and machine learning algorithms gain more popularity in recent pattern recognition researches [9-10,3].

In this study, the performance of different supervised pattern recognition techniques, namely traditional linear discriminant analysis (LDA), three different types of supervised self-organizing maps (SOMs), learning vector quantization (LVQ) and support vector machine (SVM) was investigated.

1.1 Linear discriminant analysis (LDA):

Linear discriminant analysis is a linear classification method that focuses on finding optimal linear boundaries between classes [11]. LDA is a feature reduction method based on selecting direction, which achieves maximum separation among the different classes and classify unknown samples based on Euclidean distance [1].

1.2 Self organizing maps (SOMs):

Self-organizing maps (SOMs) or the Kohonen neural networks have a high popularity among the competitive neural network field [12.]. SOMs bring the advantages of both clustering and projection methods together. Basically, SOMs analyze the data in an unsupervised manner but there are also several supervised variants like counter propagation artificial neural network (CP-ANN), supervised kohonen networks (SKN) and XY-Fused

networks (XYF) [13]. The Kohonen network is based on a single layer of neurons which are arranged in two dimensions.

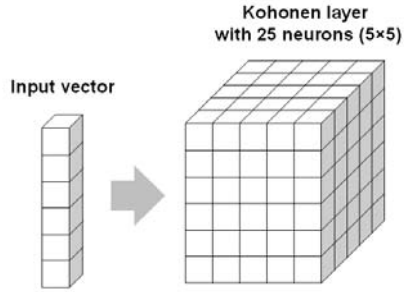

Figure 1. A self organizing map with 25 neurons. Each neuron is a vector (weight vector) with the same dimension as input vector.

Each neuron in the map is included a weight vector. The number of elements in the weight vectors is equal to the number of variables in input vector. The elements of all weight vectors (neurons) should be randomly initialized before training the network. In the training phase, each input vector is presented to the network and the most similar neuron (winner neuron) to this input can be found due to the minimal Euclidean distance between input vector and weight vectors of the neurons. The weight vectors of winner neuron and its neighbors are updated by adding the difference between the actual input vector and the respective weight vector to the elements of the weight vectors. Thus, weight vector of winner neurons and its neighbors become more similar to the input vector. Degree of weight correction attenuated by leaning rate and neighbor distance. In this way, weight vector of the winner neuron will become more similar to the input and this similarity will decrease for more distant neurons. Weight correction process is repeated iteratively until all vectors in the training set are presented a sufficient number of times (epochs) to the network. Fig. 1 shows a schematic representation of a self organizing map.

In supervised variants of SOMs, there is also an output layer with same topology but different in weight vectors dimension.

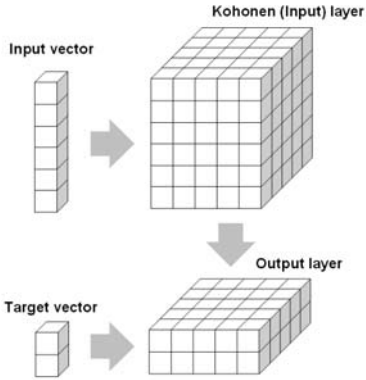

Figure 2. A counter propagation artificial neural network, CP-ANN.

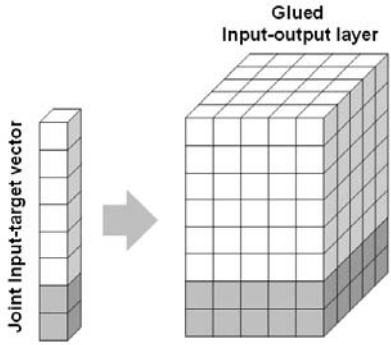

Figure 3. A supervised kohonen network (SKN).

In CP-ANN, the winner neuron in the input layer specifies the position of winner neuron at the output layer (Fig. 2).

For training a SKN, input and output layer should be glued together (Fig. 3) and training performs like a simple Kohonen map but the information in the output layer is used to specify the winner neurons during training phase.

In XY-Fused networks, a fused similarity is calculated from both input and output layer and used to find the winner position. Details of the methods can be found in cited literatures [13].

1.3 Learning vector quantization (LVQ):

Learning Vector Quantization (LVQ) is a classification method which can classify set of input vectors using a complex linear boundary. It is applicable when a simple linear boundary can not be defined for sets of input vectors. In other word, LVQ calculates multiple vectors (codebook vectors) to represent each class, which are located in the optimal position to describe the best boundary between that class and any neighboring classes in variable space. The only important thing is that the competitive layer must have enough neurons, and each class must be assigned enough neurons [2,14,15].

1.4 Support vector machines (SVMs):

Support vector machines (SVMs) are able to create a nonlinear boundary for discrimination between two classes. SVMs have been applied to a wide variety of classification problems because of good discrimination ability [16-18]. A small number of samples in training set which lie near the decision boundary (Support Vectors) are used to determine the classification (SVs) [19]. If the decision boundary between two classes in data space is not well defined by a linear function, then an appropriate kernel function can be used to transform the data into a higher dimensional feature space [20]. More details about the SVMs can be found in the literature [19].

This paper includes a statistical comparative study of six different classification methods. Performance and effect of calibration sample selection (stability) for considered methods were investigated using both simulated and real data sets according to non-error rate as a classification performance.

2. Material and methods

2.1. Datasets and software

For evaluating the performance of the six classification methods for solving a two class problem, three synthetic datasets were created. First dataset was consisted of 400 objects in a two dimensional plane (a 400×2 data matrix). The range of data values in both dimensions were from 0.00 to 1.00.

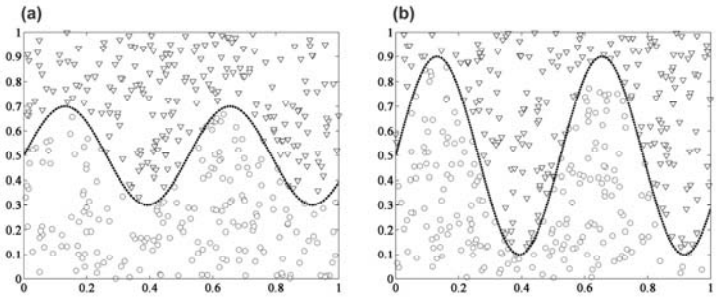

Figure 4. Synthetic datasets include 400 points uniformly distributed in two dimensions and two classes can be discriminated by a sinusoidal boundary. **(a)** Amplitude of sinusoidal boundary is equal to 0.40; **(b)** Amplitude of sinusoidal boundary is equal to 0.80.

Objects were uniformly distributed in a two dimensional space and a sinusoidal boundary was defined with a peak-to-peak difference value equal to 0.40 and define two classes as shown in Fig. 4a. Second 400×2 dataset was created in similar way but the sinusoidal boundary has higher -to-peak difference value of 0.80 in this case to make the problem more complex (Fig. 4b).

Third data set was consisted of 400 objects in a three-dimensional space (a 400×3 data matrix). First 200 objects were belong to class A and second half belong to class B. Coordination of objects were taken from a normal distribution with population mean [0 0 0] for class A and [1 1 1] for class B. Standard deviation for both populations was set to be 0.4 (Fig. 5).

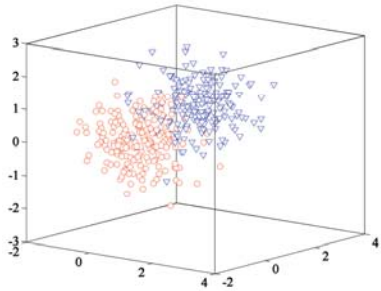

Figure 5. Synthetic dataset includes 400 points. First 200 points for each class and normally distributed in three-dimensional space).

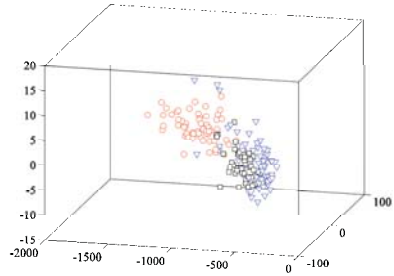

Figure 6. 3D score plot resulted from PCA analysis on wine dataset. Explained variance for PCs: PC_1: 64.6%, PC_2:13.7%, PC_3:4.7%

Experimental dataset, which is taken from the UCI machine learning repository, was the results of a chemical analysis of 178 wine samples grown in the same region in Italy but derived from different cultivars. The analysis determined the quantities of 13 constituents (variables) found in each of the three types of wines. Number of instances is 59, 71 and 48 for class 1, 2 and 3 respectively. Therefore, data matrix has dimensions 178×13. Fig. 6 shows the 3D score plot resulted from PCA analysis on this dataset.

3. Results and discussion

3.1. Model quality

For assessing quality of the built models, simulated datasets were split into training and test sets. Training and test sets for all synthetic datasets with equal number of samples (200 samples) were selected randomly. Almost an equal number of samples from each class were included in training set, and as a result in the test set. Samples of the experimental dataset were randomly divided into a training set containing 2/3 (120 samples) of all samples with the remaining 1/3 (58 samples) forming the test set. Non-error rate percent (NER%) which is related to number of correctly classified samples was used as an estimate of the classification ability of the built model for classifying test set samples.

3.2. Model optimization

There were several adjustable parameters for developing each of the six considered classification models. In order to properly compare the classification ability of all six approaches, they should be at most desirable condition at which all the adjustable parameters were optimized. NER% was the criterion for optimizing these parameters. For LDA there is no need to optimize the parameters of model, whereas for CPN, SKN, XYF, LVQ and SVM it was essential to optimize model parameters and assure the model performs at optimal condition.

Using simulated data sets, parameters of the models other than LDA were optimized based on estimation of NER% at different combination of model parameters. For the experimental dataset, splitting of the dataset in training and test sets, NER% values were estimated at different combination of parameters, as well. Arrangement of samples in the training and test set was randomly chosen. Optimized values for model parameters were chosen due to maximum NER% for the test set.

3.2.1. Counter propagation network (CPN)

In the CP-ANN model, there are several network parameters which should be optimized to gain better classification ability. Some parameters like network topology and learning rate were adjusted by experience before optimization. Network size (NS×NS) and number of epochs (EP) in training procedure can affect the model quality and should be optimized at

first. In this study, network sizes tested were 5,10,15,20 and 25, number of epochs (EP) were 25, 50, 75, 100 and 125. Network topology was set to be square shape and maximum and minimum learning rate were set to be 0.2 and 0.001 respectively.

CP-ANN models were built in all different combinations of EP and NS levels, and NER% for the test set were estimated. Randomly initializing the networks, models were built five times at all combinations of the two parameters and the combination with highest mean of NER% for the test set was chosen as the optimum combination. Relative standard deviation of the five times estimation of NER% at each condition was equal to zero. Fig. 7 shows the NER% at different combination of NS (from 5 to 25) and EP (from 25 to 100) for the test set (dataset 1).

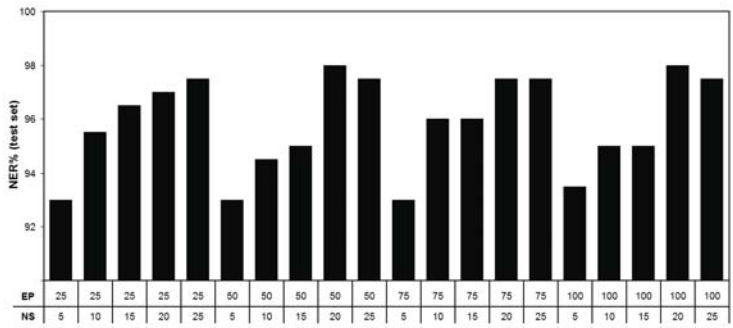

Figure 7. NER% for CP-ANN model at different combination of NS (from 5 to 25) and EP (from 25 to 100) for the test set (dataset 1).

There are two maximum NER% at (EP=50, NS=20) and (EP=100, NS=20). Combination with the less number of epochs (EP) is preferred because of saving time in data processing. Similar plots were obtained for other datasets but are not shown here.

In this study, optimum combination was chosen at NS=20 and EP=50 for the dataset 1 and 2. For dataset 3 optimum values for NS and EP were 15 and 100 respectively.

3.2.2. Supervised kohonen network (SKN)

For SKN, network size (NS) and scaling factor (SF) were optimized and the EP value was set to be 50 which was the optimal value for EP in CPN for the first and second dataset.

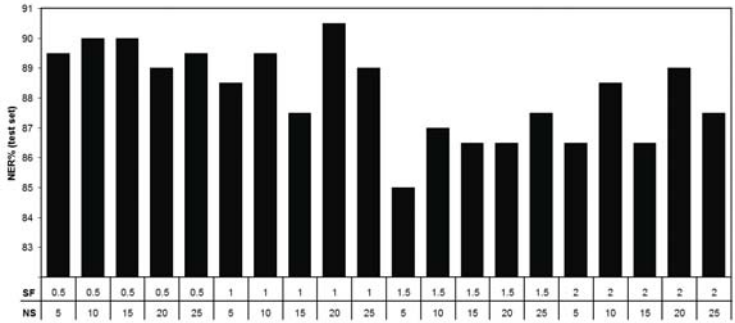

Figure 8. NER% for SKN model at different combination of SF (from 0.5 to 2) and NS (from 5 to 25) for the test set (dataset 3).

The network sizes tested were 5,10,15,20 and 25, and Scaling factor (SF) was set to be 0.5, 1, 1.5 and 2. SKN model was build five times at different combinations of these parameters and optimum combination was obtained at NS=15 and SF=1 for dataset 1, NS=20 and SF=2 for dataset 2 and NS=20 and SF=1 for dataset 3. NER% at different combination of parameters (dataset 3) is shown in Fig. 8**.** Optimum values for the parameters were calculated using similar approach (related plots are not shown here).

3.2.3. X-Y fused network (XYF)

Optimization of parameters was performed similar to previous sections and maximum NER% for the test set was obtained at NS=15 and EP=50 for dataset 1, NS=15 and EP=100 for dataset 2 and NS=5 and EP=75 for dataset 3.

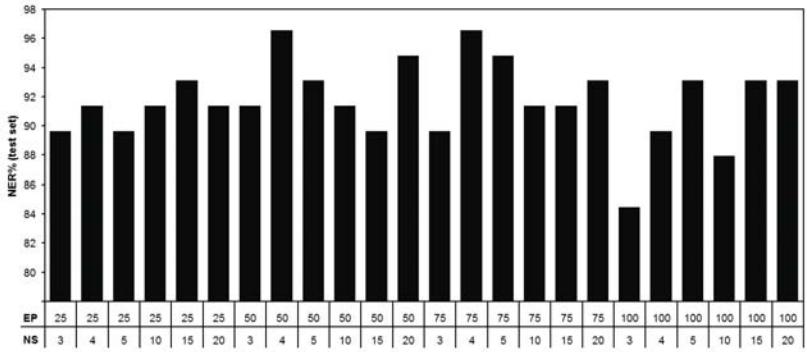

Figure 9. NER% for XY-fused model at different combination of EP (from 25 to 100) and NS (from 3 to 20) for the test set (dataset 3).

As shown in Fig. 9**,** maximum NER% was obtained in two combinations: (EP=50, NS=4) and (EP=75, NS=4), and the one with less number of epochs (EP=50, NS=4) was chosen as the optimal combination.

3.2.4. Learning Vector Quantization (LVQ)

Determination of optimal number of codebooks for LVQ model is necessary. Randomly initializing the LVQ model, it was built 10 times using training set for each dataset at different number of codebooks. Fig. 10 shows mean of NER% values at different number of codebooks for the experimental dataset. Maximum NER% of the test set for datasets 1, 2, 3 and experimental dataset were obtained using 20, 45, 8 and 20 codebooks respectively. Learning rate and number of epochs in training process was fixed at 0.02 and 50 respectively.

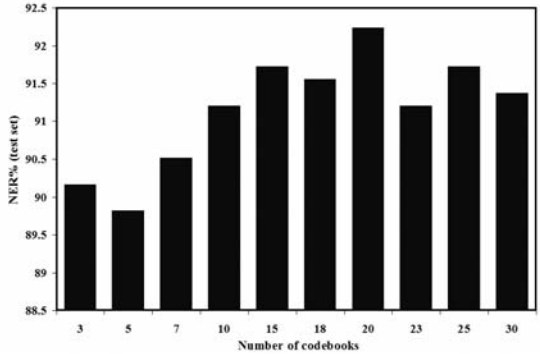

Figure 10. Mean of NER% values for LVQ model at different number of codebooks for the experimental dataset.

3.2.5. Support Vector Machine (SVM)

SVM model with radial basis kernel function has two parameters, kernel width (Kr) and penalty parameter (Pn), need to be optimized. In this study, Kr was tested in the range 0.01-0.5 using an increment equal to 0.01 and Pn tested at values 0.0, 0.1, 0.5, 1.0 and 2.0. Optimized values for Kr and Pn were obtained at Kr=0.05 and Pn=0 for dataset 1 and 2 and Kr=0.2 and Pn=0.5 for dataset 3. For the experimental dataset, best result was obtain at Kr=0.01 and Pn=1.

3.3. Model Results

To investigate the difference between the six classification methods, each simulated dataset was generated 50 times and six methods were applied each time and finally we have a population of resulting NER% of the test set for each method. The applied methods were used when parameters were in optimum conditions, for each considered dataset. These populations can help for better comparison of classification ability and model stability. For better representation, these populations are shown in box and whisker plot. The box has lines at the lower quartile, median, and upper quartile values. Whiskers extend from each end of the box to the adjacent values in the data. Position of median and the width of the whiskers are indicating model quality and stability respectively. Fig.11 shows box and whisker plot for population of the results for dataset 1.

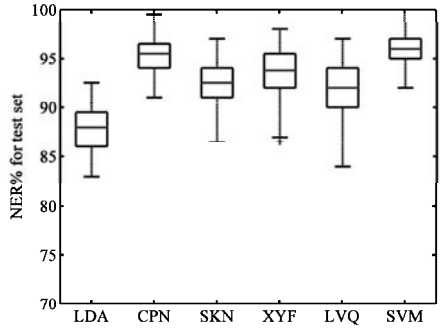

Figure 11. Box and whisker plot for population of the results for dataset 1**.**

According to this figure, modeling performance for all methods was good except for LDA. It is evident that the two classes can not be completely discriminated using a straight line and the points near the boundary could fall in wrong classes. Other methods performed better than LDA due to their ability to handle nonlinear discrimination. Among the methods, CP-ANN and SVM show better model quality in accord with position of median and whiskers.

For dataset 2 which is more complicated problem, difference between qualities of models is more distinct (Fig.12).

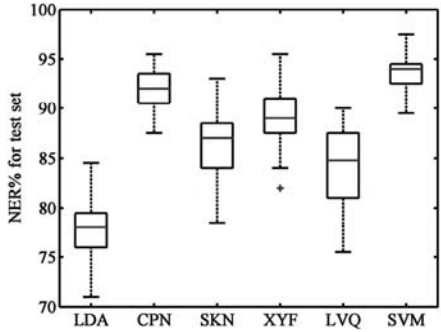

Figure 12. Box and whisker plot for population of the results for dataset 2.

As it was expected, model quality of LDA is lowest as before. CP-ANN and SVM performed better in model quality and stability. XYF was slightly better classification ability than SKN and LVQ but its stability is worse in compare with CP-ANN and SVM. This could be revealing that CP-ANN, as a supervised self organizing map network, has good capability to handle such classification problem as well as SVM.

Moreover, CP-ANN has the capability of mapping input objects in a two dimensional map of neurons which let to capture structure of data and easy inspection of the patterns. Importance of this advantage will be more distinct in the cases which dimension of data is more than two.

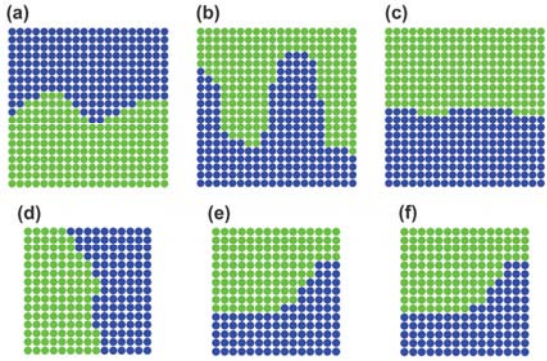

Figure 13. Class maps in (a) CP-ANN (20×20) for dataset 1, (b) CP-ANN (20×20) for dataset 2, (c) SKN (20×20) for dataset 2, (d) SKN (15×15) for dataset 1, (e) XYF (15×15) for dataset 1 and (f) XYF (15×15) for dataset 2.

Fig.13(a,b) shows class maps in CP-ANN (20×20) for dataset 1 and 2, which can easily give visual information about the patterns of objects in dataset. Class map for SKN (Fig.13(c,d)) and XYF (Fig.13(e,f)) was affected by both data structure and class membership of the points (samples). It is due to training algorithm and we can't exclude useful information about data structure.

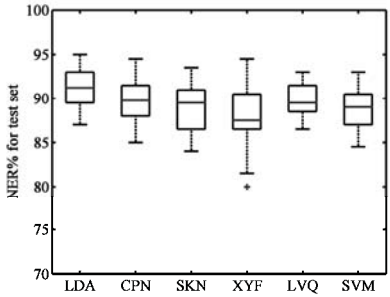

Figure 14. Box and whisker plot for population of the results for dataset 3

Dataset 3 was generated in a different manner and population of results for the methods is shown in Fig.14. In this case LDA was performed slightly better than other methods. In nonlinear boundary methods, the position of the boundary is directly depend to the pattern of samples in the training set and especially those present near the boundary. Choosing different samples in training set have smaller effect on model quality of LDA as this method is based on sample distribution over the entire class and the defined linear boundary is calculated according to within and between classes variance. In this way, LDA is less sensitive to the pattern of boundary samples.

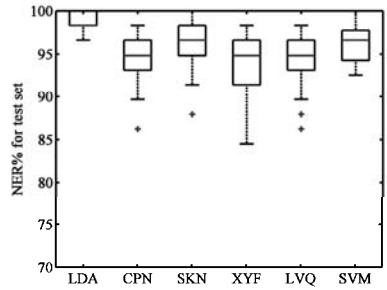

Figure 15. Box and whisker plot for population of the results for experimental dataset.

The results for experimental dataset are shown in Fig.15. It can be seen that LDA performed significantly well (obtaining up to 96% test set NER% in and outstanding model stability) and also SKN and SVM perform better than other methods (obtaining higher than 90% test set NER% with acceptable model stability). High performance of LDA to classify test set samples in experimental dataset could be related to similarity of this data to the third simulated data set that includes normally distributed points. Using the simple boundary methods like LDA helps the model to capture overall structure of data points and not to be over-trained. Complex boundary methods will precisely learn to classify all or most of training sample correctly, but will fail to predict the test samples. We can not recommend a method, but applying different visualization method to explore data and calculations of stability for the build models can guide us to use optimal method for any dataset.

5. Conclusion

In this study, classification performances of several supervised self organizing maps, including counter propagation network (CPN), supervised kohonen networks (SKN) and XY-fused network (XYF) were compared to linear discriminant analysis (LDA), learning vector quantization (LVQ) and support vector machine (SVM). We see that the classification performances for CPN, SKN and XYF were comparable with powerful classification methods like SVM and LVQ. Also, it is clear that the stability and performance of CPN models was good as SVM for the examined cases. In addition, CPN can capture and visually show the shape of boundary between classes. An interesting point was that for types of data structures, the simple linear discriminant analysis performs better than nonlinear, highly flexible classification techniques such as SVM and SOMs. In this way, performance of these methods should be checked anytime we deal with a new set of data to make a proper choice of classification method.

References

[1] M. Sharaf, D. Illman, B. Kowalski, *Chemometrics*, Wiley, New York, 1986, p. 228.

[2] Y. Roggo, L. Duponchel, J. P. Huvenne, Comparison of supervised pattern recognition methods with McNemar's statistical test: Application to qualitative analysis of sugar beet by near-infrared spectroscopy, *Anal. Chim. Acta* **477** (2003) 187–200.

[3] I. Kurt, M. Ture, A. T. Kurum, Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease, *Expert. Sys. Appl*. **34** (2008) 366–374.

[4] J. S. Torrecilla, E. Rojo, M. Oliet, J.C. Dominguez, F. Rodriguez, Self-organizing maps and learning vector quantization networks as tools to identify vegetable oils, *Agric. Food Chem*. **57** (2009) 2763–2769.

[5] S. J. Dixon, R.G. Brereton, Comparison of performance of five common classifiers represented as boundary methods: Euclidean distance to centroids, linear discriminant analysis, quadratic discriminant analysis, learning vector quantization and support vector machines, as dependent on data structure, *Chemom. Intl. Lab. Sys*. **95** (2009) 1–17.

[6] A. Astel, S. Tsakovski, P. Barbieri, V. Simeonov, Comparison of self-organizing maps classification approach with cluster and principal components analysis for large environmental data sets, *Water Res.* **41** (2007) 4566–4578.

[7] Z. Chalabi, N. Berrached, N. Kharchouche, Y. Ghellemallah, M. Mansour, H. Mouhadjer, Classification of the medical images by the Kohonen network SOM and LVQ, *J. Appl. Sci*. **8** (2008) 1149–1158.

[8] B. Alsberg, R. Goodacre, J. Rowland, D. Kell, Classification of pyrolysis mass spectra by fuzzy multivariate rule induction-comparison with regression, K-nearest neighbour, neural and decision-tree methods, *Anal. Chim. Acta* **348** (1997) 389–407.

[9] B.K.Bhattacharyya, P. Das, S.Datta, A comparative study for modeling of hot-rolled steel plate classification using a statistical approach and neural-net system, *Mater. Manufact. Proc*. **21** (2006) 747–755.

[10] C. Budayan, I. Dikmen, M.T. Birgonul, Comparing the performance of traditional cluster analysis, self-organizing maps and fuzzy C-means method for strategic grouping, *Expert. Syst. Appl*. **36** (2009) 11772–11781.

[11] D. Massart, B. Vandeginste, S. Deming, Y. Michotte, L. Kaufman, *Chemometrics: A Textbook*, Elsevier, New York, 1988.

[12] T. Kohonen, *Self-Organizing Maps*, Springer, New York, 2001.

[13] W. Melssen, R. Wehrens, L. Buydens, Supervised Kohonen networks for classification problems, *Chemom. Intl. Lab. Sys*. **83** (2006) 99–113.

[14] G. R. Lloyd, R. G. Brereton, R. Faria, J. C. Duncan, Learning vector quantization for multiclass classification: Application to characterization of plastics, *J. Chem. Inf. Model*. **47** (2007) 1553–1563.

[15] L. A. Fraser, D. A. Mulholland, D. D. Fraser, Classification of limonoids and protolimonoids using neural networks, *Phytochem. Anal*. **8** (1997) 301–311.

[16] M. P. S. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, D. Haussler, Knowledge-based analysis of microarray gene expression data by using support vector machines, *PNAS* **97** (2000) 262–267.

[17] S. Tong, D. Koller, Support vector machine active learning with applications to text classification, *J. Mach. Learn. Res*. **2** (2002) 45–66.

[18] S. Zomer, R.G. Brereton, J.F. Carter, C. Eckers, Support vector machines for the discrimination of analytical chemical data: application to the determination of tablet production by pyrolysis-gas chromatography-mass spectrometry, *Analyst* **129** (2004) 175–181.

[19] S. Zomer, M. D. N. Sánchez, R. G. Brereton, J. L. P. Pavón, Active learning support vector machines for optimal sample selection in classification, *J. Chemom*. **18** (2004) 294–305.

[20] S. Amari, S. Wu, Improving support vector machine classifiers by modifying kernel functions, *Neural Netw.* **12** (1999) 783–789.