Reliability of Bank Note Classifier by Neural Networks

Sigeru Omatu

Computer Science and Intelligent Systems, Graduate School of Engineering
Osaka Prefecture University, Sakai, 599-8531,Osaka, Japan
Tel: +81-722-254-9278, Fax: +81-722-257-1788, E-mail: omatu@cs.osakafu-u.ac.jp

Abstract

This paper addresses the reliability of neuro-classifiers for bank note recognition. A local principal component analysis (PCA) method is applied to remove non-linear dependencies among variables and extract the main principal features of data. At first the data space is partitioned into regions by using a self-organizing map (SOM) model and then the PCA is performed in each region. A learning vector quantization (LVQ) network is employed as the main classifier of the system. By defining a new algorithm for rating the reliability and using a set of test data, we estimate the reliability of the system. The experimental results taken from 1,200 samples of US dollar bills show that the reliability is increased up to 100% when the number of regions as well as the number of codebook vectors in the LVQ classifier are taken properly.

Keywords:

Bank Note Classifier, Neural Networks, Reliability

Introduction

The recognition of bank note recently has been concerned effectively by using of neural networks, and it is shown that neuro-classifiers are robust for recognition of defective, taint, and worn out bank note. Takeda et al. [1] have used a random mask for preprocessing the data and then a multi-layer neural network as the classifier for recognition of bank note. Teranishi et al. [2] have applied a method based on acoustic cepstrum patterns for extracting the features of bill and then a competitive neural network as the classifier. Tanaka [3] has employed a probabilistic principal component analysis (PCA) for extracting the main characteristics of bill data.

Due to high risk of misclassification in such systems, the reliability of recognition becomes of a high importance. Basically the classifier must be fully robust for frayed or dirty bills of different models, and also has insensitivity to shift rotation and different directions of inserting bill. In fact, even if the classification rate is 100% over the data space, still it is necessary to make sure about the reliability of classification over all variety of real data.

We have already proposed in our pervious study [4] a method which uses the PCA algorithm for extracting the features of bill image data and utilizes a learning vector quantization (LVQ) network as the main classifier, where the reliability is evaluated thorough a new defined algorithm using Gaussian mixture densities for distribution of data. We have found out that in case of large variability in input data or any non-linear correlation between the data, some additional discrimination process is needed to keep the reliability high enough.

As the main limitation of PCA is its global linearity, that is, it only defines a linear projection of data and does not model non-linear relationship among variables, some developments of non-linear principal component analysis (NLPCA) have been presented to address this limitation [5]. However, both PCA and NLPCA algorithms try to model the entire data by the same global features. As an alternative, the complexity of the data can be modeled by using a mixture of the local linear PCA. The local PCA algorithm clusters the input data into regions and performs PCA on the data that falls within each region.

In this paper, we apply a local PCA method where a SOM model is used as for clustering the data into homogenous regions. Our approach is similar to Kerschen [6] in the sense of local PCA application but differs in the clustering method as he has used vector quantization (VQ) for clustering phase.

The current system is intended for classifying different kinds of bank note, however, we examined only US dollar bills. The experimental results show a growth in reliability by 0.2% after using feature extracted by the local PCA model comparing to the method based on the conventional PCA, and a growth by 2.3% comparing to classification without the PCA.

Bill Data Preprocessimg

The original image of bill money comes as a 10x170 array of data taken through three main advance sensors and two auxiliary ones. Each sensor uses two different waves lengths for generating two channels of data. At first by using a linear function we generate a new channel of data based on two channels of each sensor. Thus totally 15 channels are obtained among them we select 6 main channels which represent the main characteristic of data. A simple compression algorithm is used to reduce the size of data from 170 pixels in each channel to 30. Then a linear transformation is applied for normalization as follow:

$$z_i = \frac{x_i - \bar{x}}{S_x} G + C$$

where x_i is the pixel value in each channel, $\bar{x}$ is the mean value of pixels, S_x is standard deviation, and G=512 and C=128 are the coefficients of gain and offset, respectively whose values are taken experimentally. Thus, a matrix of 6x30 size is provided for using in feature extraction step.

Data Compression

CA is one of the most popular methods for preprocessing, compression and feature extraction of data and it is discussed in most documents on multivariate analysis. The most common derivation of the PCA is in terms of a standardized linear projection which maximizes the variance in the projected space. As explained in the introduction, the PCA only removes linear correlation among the data and is only sensitive to second order statistics, i.e., it is assumed that the distribution of data is Gaussian. In case of non-linear relations among the variables, we need to consider higher order statistics to eliminate the dependencies which are not removed by the PCA. Here, we apply a local PCA model where the data is clustered into regions by using a SOM model at first and then the PCA is performed on the data of each region. The procedure is explained in the following.

A Pre-Clustering

SOM is shown to have desirable properties compared to classical clustering methods. It provides a natural measure for the distance of a vector from a cluster which is adaptive from the local statistics of the data [7]. The SOM forms a map corresponding to the data distribution so that regions of the map can be interpreted as clusters in the data space. The main key point is defining a set of codebook vectors $\mathbf{m_i}$, i=1,2,..,m which represent units of the map. Then as for a given input $\mathbf{x}$, it is mapped to a unit associated to $\mathbf{m_c}$ such that:

$$\| \mathbf{x} - \mathbf{m_c} \| = \min_{i} \{\| \mathbf{x} - \mathbf{m_i} \|\}$$

$\mathbf{m_i}$ codebook vectors are updated through a training process iteratively as:

$$\mathbf{m_i}(t+1) = \mathbf{m_i}(t) + h_{ci}(t)\, [\mathbf{x}(t) - \mathbf{m_i}(t)]$$

where t indicates the iteration and h_{ci} is a neighborhood function taken as:

$$h_{ci} = \alpha\,(t).\ exp(-d_{ci}^2 / 2\, r^2(t)\,).$$

Here, α and r are learning rate and neighborhood radius, respectively both decrease monotonically as a linear function of time. d_{ci} is the distance between m_c and m_i . We consider a 6x5 map size for clustering the preprocessed data and mapping the data of 24 classes onto 30 partitions. The initial radius for neighborhood is taken 10 and initial learning rate is taken 0.2.

B PCA Modeling

If $\mathbf{x_i}$ is supposed to be an n-dimensional vector of a data set with i=1,…,N, then the goal of the PCA is to find r dimensional axes $\mathbf{p_i}$ onto which the retained variance under projection is maximal. These axes are given by the eigenvectors associated with r largest eigenvalues of the covariance matrix of data as:

$$\Sigma\, \Phi = \Lambda\, \Phi$$

where $\Phi = [\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_r}]$ is the eigenvectors matrix, Λ is the eigenvalue matrix as diag$\{\lambda_1, \lambda_2, ..., \lambda_\rho\}$ with $\lambda_1 > \lambda_2 > .... > \lambda_\rho$, and Σ is the covariance matrix which is defined as:

$$\Sigma = E\,[(\mathbf{x}-\mu)(\mathbf{x}-\mu)^T]$$

where $\mu = E[\mathbf{x}]$ is the mean vector of data.

Then the transformed data vector $\mathbf{y_i}$ is determined as:

$$\mathbf{y_i} = f(\mathbf{x_i}) = \Phi^T\,(\mathbf{x_i} - \mu)$$

which is a reduced r-dimensional representation of data vector $\mathbf{x_i}$.

But here considering the q local regions which have been already provided by SOM, we apply a sort of functions f_i (.) with i=1,…,q instead of a single encoding function. As we consider the local regions small enough according to number of different classes, we expect an adequate representation of data within each region by using this local PCA algorithm. The procedure is taken place as follow:

For each cluster of data corresponding to each region of the SOM with the mean vector value of μ_j, the covariance matrix is estimated as:

$$\Sigma_j = \frac{1}{N_j} \sum_{\mathbf{x} \in S_j} (\mathbf{x} - \boldsymbol{\mu}_\mathbf{j})(\mathbf{x} - \boldsymbol{\mu}_\mathbf{j})^T$$

where N_j is the number of vectors lied in the cluster S_j. Then by determining the eigenvectors ($\mathbf{p_{j1}}, \ldots, \mathbf{p_{jr}}$) of each matrix $\mathbf{\Sigma_j}$, the function f_j and thereby the transformed vector $\mathbf{y_i}$ can be obtained for each region as follow:

$$\mathbf{y_i} = f_j(\mathbf{x_i}) = [\mathbf{p_{j1}}, \ldots, \mathbf{p_{jr}}]^T\,(\mathbf{x_i} - \mu_j) \qquad \mathbf{x_i} \in S_j$$

As explained in the following, in this paper the data dimension n is 180 and the r dimension is taken as 30. The number of regions q is taken as 30 according to SOM partitions as described in the following A.

Accordingly through application of the PCA over all partitions a new 30 dimensional data set is produced which contains the main features of 180 dimensional data.

Classification

Kohonen's LVQ is a supervised learning algorithm associated with the competitive network [7] which basically consists of an input layer and an output layer, and an array of weight vectors $\mathbf{w_i}$ where w_{ij} denotes a connection weight between the jth node in the input layer and ith node in the output layer (Figure 1). Given a training data set X, each labeled with a class identifier, and a set M of codebooks vectors, the LVQ network adaptively modifies these codebooks so that they represent the class probability distribution in the training data set. This modification of codebooks consists of applying a "punishment" when a codebook is near a sample of a different class and "reward" when it is near a sample of its own class.

Since the LVQ network is beneficial in classification of data with large number of inputs and explanation of the misclassification, it is applied as the main classifier of present system. As we consider 6 kinds of US bills including 1, 5, 10, 20, 50, and 100 dollars and for each bill there exists 4 direction of inserting (Figure 2), totally 24 (i.e. 6x4) output categories are considered for the classifier. The system is trained by taking trial number of codebook vectors for each class looking for the best classification rate and maximum reliability. A total number of 120 codebook vectors (averagely 5 vectors per class) is experimentally found to be the best. The number of iterations for each training epoch is taken 10,000 while a linear function as $\alpha(t) = \alpha(0)\,(1.0 - t/T)$ is supposed for learning where T is the number of iterations.. Therefore, the LVQ classifier has a number of 30 neurons (the number of extracted features) in the input layer and a number of 120 neurons in the output.

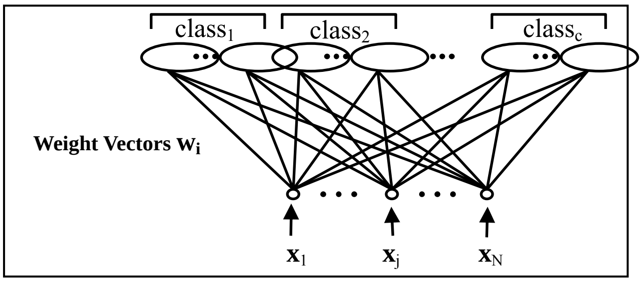

Figure 1 - A Schema of LVQ Network Structure

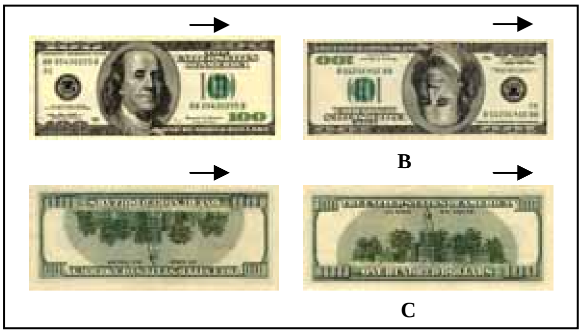

Figure 2 - Four Different Directions for Inserting a Bill

Reliability Evaluation

We propose a simple but effective algorithm for evaluating the classification reliability. After the LVQ classifier is trained and the codebook vectors are determined, the test data set is used to estimate the parameters of probability density function (pdf) supposing a Gaussian distribution around each codebook vector as:

$$p_i(\xi) = \sigma_i^{-1}(2\pi)^{-d/2}\exp\left(-(\xi-\mu_i)^2/2\sigma_i^{2}\right)$$

where ξ is the distance between data vector and codebook vector (i.e. $\| \mathbf{x_i} - \mathbf{m}_i \|$) , μ_i and σ_i are the mean and variance in pdf of codebook vector i respectively , and d is the dimensionality of the feature vectors (here, 30). Assuming the Gaussian probability density function, the interval [$\mu_i - 4.5\sigma_i$, $\mu_i + 4.5\sigma_i$] can be considered as an area that covers almost 100% of probabilities (100- 5.122 E^{-5}). For a given class of codebook vectors if the densities have no overlap within this interval with densities of other classes, the reliability for this class is supposed to be 100%, but in case that this interval is overlapped with other classes, the reliability can be calculated as:

$$RM = \sum_{i}^{L} \alpha_i \int_{\mu_i-4.5\sigma_i}^{\theta} p_i(\xi)\, d\xi$$

where L is the number of codebooks within each class, α_i is a normalizing coefficient, ($\Sigma\alpha_\iota=1$, $\alpha_\iota> 0$), and θ is the cross point of each pdf with the interval boundary of nearest density from another class (Figure 3).

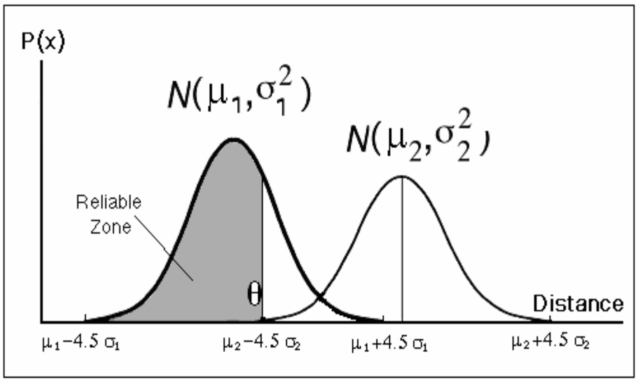

Figure 3 - The Overlap between Codebook Densities of Two Near Classes

Thus, the total reliability rate of the system finally can be determined by averaging these class reliability values.

Experimental Results and Discussion

A set of 2,400 sample data from 6 kinds of US dollar bills including 1, 5, 10, 20, 50, and 100 dollar and four directions for each bill (i.e. 100 samples for each direction), is used for learning the LVQ classifier. Also a number of 1,200 samples containing both normal data and slightly shifted data (to right and left) is taken for evaluating the system (that is 50 samples for each direction). The bills we used were of various levels of fatigue and made in different years. However, globally they can be considered as normal bills not frayed ones. The order of inputting data is quite random either in learning and testing phase. As for learning the LVQ classifier we have tried different number of codebook vectors from 3 to 10 per class looking for the best results.

On the other hand, concerning to local PCA application, we have tried different number of regions from 24 to 48 as the output of SOM to study its influence on the classification and reliability. The result of classification rate and reliability is shown in Table 1. The reliability is evaluated through the algorithm we have defined in the following. As it can be seen in Table 1 by increasing the number of codebook vectors in classifier as well as the number of regions in local PCA process, the reliability can be increased significantly. We have found that by taking a number of 120 codebook vectors in the LVQ classifier and 30 regions for the local PCA the reliability of system can be extended up to 100%.

Figure 4 indicates the relation between number of regions and reliability of system, clearly. As can be observed increasing the number of regions makes a significant increment in reliability value firstly but after some extend, the reliability is not influenced anymore by larger number of regions and if the number of regions increases so much, it makes an inverse affect on the reliability.

Table 1 –The Classification Results and Reliability Test Data (The Number of PCA Components in all Cases Is 30)

No. of Codebooks	No. of Regions	Recog. Rate (%)	Reli. Rate (%)
60	1 (Standard PCA)	100	82.1
	12	100	90.1
	24	100	90.4
	30	100	94.6
	48	100	94.7
80	1	100	96.6
	12	100	97.5
	24	100	98.3
	30	100	98.4
	48	100	98.4
120	1	100	99.8
	12	100	99.98
	24	100	99.99
	30	100	100
	48	100	100
200	1	100	99.99
	12	100	99.99
	24	100	100
	30	100	100
	48	100	100

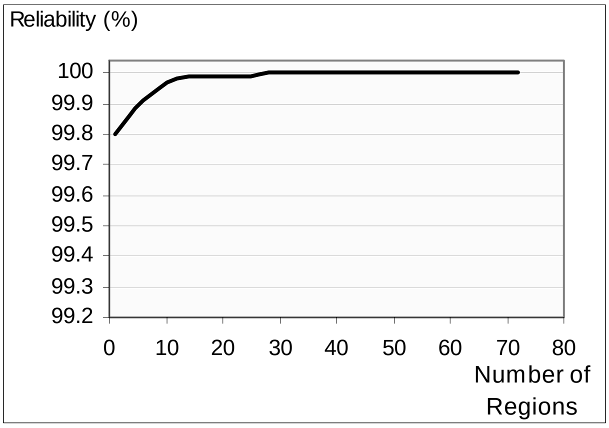

Figure 4 - The Relation between the System Reliability and Number of Regions in Local PCA

As we have used a test data set which contains shifting data, i.e., the data of bills with slightly shift to right or left at the inserting time, the high classification rate of system shows that the system is robust enough on shifting and sliding.

Conclusions

In this paper we have presented a local PCA approach for feature extraction of data in classification of bank note. The aim is to model the complexity of data and correlation between variables by using a simple linear model. The method first exploits a SOM model to cluster the data space into disjoints regions. Then a standard PCA model is applied in each region. The experimental results taken from 1,200 US dollar bills show that by taking a proper number of regions and also an optimized number of codebook vectors for LVQ classifier, the reliability of system can be increased up to 100%. Comparing to the conventional PCA method which was our pervious approach, the present method shows a significant growth in reliability rate. However, we have applied only US dollars for training and testing the system, it can be easily generalized for other kinds of bank note and considered as a multi-currency classifier with wide variety of data.

References

[1] Takeda F and Omatu S (1993), Bank Note Recognition System Using Neural Network with Random Masks, Proceeding of the World Congress on Neural Networks, Portland, USA, Vol. 1, pp. 241-244

[2] Teranishi M, Omatu S and Kosaka T (2000), Neuro-classifier of Currency Fatigue Level Based on Acoustic Cepstrum Patterns, Journal of Advanced Computational Intelligence, Vol. 4, No. 1, pp. 18-23

[3] Tanaka M (1999), Modeling of Mixtures of Principal Component Analysis Model with Genetic Algorithm, Proc. 31st Int'l Symposium on Stochastic Systems Theory and Its Applications, Yokohama, Japan, pp. 157-162

[4] Ahmadi A and Omatu S (2003), A High Reliability Method for Classification of Paper Currency Based on Neural Networks, Proceeding of The Eighth International Symposium on Artificial Life and Robotics (AROB 8th '03), Oita, Japan, Vol. 2, pp. 601-604

[5] Kramer MA (1991), Non-linear Principal Components analysis Using Auto-associative Neural Networks," A.I.Ch.E Journal, Vol. 37, pp. 233-243

[6] Kerschen G. and Golinval JC (2002), Non-linear Generalization of Principal Components Analysis: From A Global to Local Approach, Journal of Sound and Vibration , Vol. 254, No. 5, pp. 867-876

[7] Haykin S (1999), Neural Networsk, New Jersey: Prentice Hall