A Nature Inspired Ying-Yang Approach for Intelligent Decision Support in Bank Solvency Analysis

M.N Nguyen, D.Shi and C.Quek
School of Computer Engineering, Nanyang Technological University, Singapore 639798
Email: asdmshi@ntu.edu.sg, ashcquek@ntuix.ntu.ac.sg

ABSTRACT

Since the collapse or failure of a bank could trigger an adverse financial repercussion and generate negative impacts, it is desirable to have an Early Warning System (EWS) that identifies potential bank failures or high-risk banks through the traits of financial distress. This research is aimed to construct a novel neural fuzzy Cerebellar Model Articulation Controller (FCMAC) as an alternative to analyze bank solvency, in which a nature inspiration motivated from the famous Chinese ancient Ying-Yang philosophy is introduced to find the optimal fuzzy sets, and Truth Value Restriction (TVR) inference scheme is employed to derive the truth-values of the rule weights. The proposed model functions as an early warning system and is able to identify the inherent traits of financial distress based on financial covariates (features) derived from publicly available financial statements. Our experiments are conducted on a benchmark dataset of a population of 3635 US banks observed over a 21 years period. Three sets of experiments are performed – bank failure classification based on the last available financial record and prediction using financial records one and two years prior to the last available financial statements. The performance of the proposed Ying-Yang FCMAC network as a bank failure classification and early warning system is very encouraging.
1. Introduction

The collapse and failure of a bank could have devastating consequences to the entire banking system and an adverse repercussion effect on other banks and financial institutions. Some of the negative impacts are the massive bailout cost for a failing bank and the negative sentiments and loss of confidence developed by investors and depositors. Hence, bank failure prediction is an important issue for regulators of the banking industries. The other benefit of the prediction of bankruptcies is for accounting firms. If an accounting firm audits a potentially troubled firm, and misses giving a warning signal then it might face costly lawsuits.

Very often, bank failures are due to financial distress. Various traditional models have been employed to study bank failures. In literature, there are two main approaches to bankruptcy prediction. The first approach, the structural approach, is based on modeling the underlying dynamics of interest rates and firm characteristics and deriving the default probability based on these dynamics. The second approach is the empirical or the statistical approach. Instead of modeling the relationship of the default with the characteristics of a firm, this relationship is learned from the data. The focus of this paper is on the empirical approach, particularly the use of fuzzy neural networks.

The more popular statistical methodologies used are Altman’s multivariate discriminant analysis (MDA) [1], Ohlson’s logistic regression approach (LR) [2] and Cox’ proportional hazards model [3, 4]. Altman’s MDA is based on applying the Bayes classification procedure, under the assumption that the two classes have Gaussian distributions with equal covariance matrices. The covariance matrix and the class means are estimated from the training set. Ohlson introduced the LR to bankruptcy prediction problem. It is essentially a linear model with a sigmoid function at the output and is thus similar to a single-neuron network. The model has a nice probabilistic interpretation as the output is between 0 and 1. Ohlson used a novel set of financial ratios as inputs. Both
the MDA and the LR model have been widely used in practice and in many academic studies. They have been standard benchmarks for the loan default prediction problems. However, these models have various deficiencies and they are not able to identify the traits of financial distress that led to bank failure and thus function as black boxes.

On the other hand fuzzy neural network models, which simulate the human style of reasoning and decision-making when solving complex problems, can overcome the deficiencies of the traditional statistical models and can be employed to handle the classification of both failed and survived (non-failing) banks. The objective of various soft computing approaches is to synthesize the human ability to tolerate and process uncertain, imprecise and incomplete information during the decision-making process. A popular approach is the integration of neural network and fuzzy system to create a hybrid structure known as a fuzzy neural network. Fuzzy neural networks [5, 6] such as those done by Tam and Kiang [7, 8], Salchenberger [9], Coats and Fant [10], Kerling and Poddig [11], Boritz and Kennedy [12] and Zhang [13] all compare neural network approaches to the MDA and LR techniques. The main advantage of a fuzzy neural network is its ability to model the characteristics of a given problem using a high-level linguistic model instead of low-level complex mathematical expressions. The linguistic model is essentially a fuzzy rule base consisting of a set of IF-THEN rules. The fuzzy rules are highly intuitive and easily comprehended by the human users. In addition, the hybrid structure of a fuzzy neural network is highly transparent as the fuzzy rules can be used to interpret the weights and linkages of the connectionist network. Moreover, a fuzzy neural network can self-adjust the parameters of the fuzzy rules using learning algorithms derived from the neural network paradigm.

Fuzzy neural networks are universal data-mining tools [14, 15] and possess strong capability to derive the intrinsic relationships between the selected inputs and outputs. In addition, the generalization attribute of fuzzy systems enables them to interpolate the decision-making process to new (unseen) cases. This serves very well the objectives of a bank failure prediction system in the study of bank failures since fuzzy neural networks
can be employed to identify the inherent characteristics of failed banks. It allows one to interpret the traits of the financial distress that leads to a bank failure.

Typically, the fuzzification phase in a fuzzy neural network is fulfilled by clustering training data in each dimension independently. All the existing clustering algorithms can be divided into two groups: clustering with or without pre-specified number of clusters [16, 17]. However, all of these conventional methods apply “one way” clustering, that is, they consider only either forward path of the mapping of the input data into the clusters, or, the backward path of learning the clusters from the input data. Therefore, they cannot guarantee to obtain optimal cluster architecture for the input data.

Discrete incremental clustering (DIC) [16] proposed by Tung and Quek has the characteristics of noise tolerance and does not require prior knowledge of the number of clusters present in the training data set. It used the same initial information to create a new cluster for all dimensions of the input data. However, since the ranges of dimensions are different, DIC cannot obtain an optimal number of clusters because it is increased ratio to the range of dimensions.

Lee, Chen and Fu proposed a self-organizing Hierarchical CMAC neural network [17] which combines a self-organizing input space module and a binary hierarchical CMAC neural network. However, the self-organizing input space module based on Shannon’s entropy measure and the golden-section search method can not guarantee to obtain an optimal input space quantization. Moreover, the rigid structure of the binary hierarchical CMAC neural network is another disadvantage.

Fuzzification is actually equivalent to get the underling distribution of each dimension of a finite size of the training patterns, so that we can apply it to the unknown data. This research aims to find the optimal number of fuzzy sets and form clusters in the fuzzification phase to achieve higher generalization ability. In this paper, a nature inspiration of the Ying-Yang philosophy is applied in the fuzzification phase of the fuzzy CMAC. The Ying-Yang fuzzification (YYF) using Bayesian Ying-Yang learning (BYY)
[18, 19] is based on the harmony of two representations: the mapping of the input data $x$ into an inner representation cluster $y$, and the generation of the input data $x$ from an inner representation cluster $y$. This nature harmony principle makes the proposed fuzzifier be able to obtain optimal clusters from the input training data. In the inference phase of the Ying-Yang FCMAC, Truth-value restriction (TVR) inference scheme is used to derive the truth-values of the rule weights from the truth-value of the antecedents. YYF together with TVR provide the Ying-Yang FCMAC system with an optimal fuzzy rule set, and a consistent rule base, a strong theoretical foundation, more logical and intuitive to the human reasoning process.

This paper is organized as follows. A novel Ying-Yang fuzzification is proposed in the next section. Section 3 describes the structure of Ying-Yang FCMAC. Section 4 presents the experimental results of the Ying-Yang FCMAC network when applied to the classification of failed and survived (non-failing) banks. In this section, the details on the selection of the financial covariates are also provided, followed by our conclusions in Section 5.

2. A Ying-Yang inspired approach to Fuzzification

In this section, the fuzzifier using Bayesian Ying-Yang [18-19] is employed to automatically form the fuzzy set for each dimension. As shown Figure 2.1, the Ying-Yang fuzzification system considers two complement representations of the joint distribution of input pattern $x$ and fuzzy cluster $y$.

The first one is the forward/training model. It considers the first representation of the joint distribution $p(x,y)$:

$$p(x, y) = p(y | x)p(x),$$  \hspace{1cm} (2.1)

This mode is called a Yang/(visible) model which focuses on the mapping function of the input data $x$ into a cluster representation $y$ via a forward propagation distribution $p(y|x)$. In this process, the input data $x$ are visible and are considered as known, whereas the
clusters $y$ are invisible and are considered as unknown. By this model, $p(x)$ is estimated from the given input data and is then transferred into unknown fuzzy cluster $p(y) = \int p(y|x)p(x)dx$. This process is regarded as an unsupervised learning process.

$$p(x,y) = p(y|x)p(x)$$

The second one is the backward/running model. It considers the first representation of the joint distribution $p(x,y)$:

$$p(x,y) = p(x|y)p(y), \quad (2.2)$$

This mode is called Ying/(invisible) model which focuses on the generation function of the input data $x$ from a cluster representation $y$ via a backward propagation distribution $p(x|y)$. In this process, clusters $y$ are visible and are considered as known, whereas the input data $x$ are invisible and are considered as unknown. By this model, the input data are generated from the constructed fuzzy clusters by $p(x) = \int p(x|y)p(y)dy$. This process is regarded as a supervised learning process.

According to the Bayesian theory, two complement representations of the joint distribution of Equation (2.1) and Equation (2.2) should be equal. However, the result of these two equations is not equal unless $y$ is the optimal solution. Under the Ying-Yang
harmony principle, the difference between the two Bayesian representations in (2.1) and (2.2) should be minimized. Thus, the trade-off between the forward/training model and the backward/running model is optimized. It means that the input data are well mapped into the clusters and at the same time the clusters also well cover the input data. Therefore, the entire BYY fuzzification system should be the least complexity. In other words, the proposed Ying-Yang FCMAC has the highest generalization ability when the harmony of two Ying and Yang models is achieved.

The Ying-Yang Fuzzification involves two phases, namely, parameter learning and cluster selection. Parameter learning does the task of determining all the unknown parameter \( \Theta^j \) for a specific value of cluster \( K^j \). Then, cluster selection is made to select the optimal number of clusters \( K^j^* \) from a collection of specific BYY systems with different value of cluster \( K^j \).

In the first phase, the Kullback-Leibler divergence [20] is used to evaluate the difference of the joint probability between Ying and Yang. The minimization of the Kullback-Leibler divergence will produce the optimal parameter \( \Theta^j^* \) at each value of cluster \( K^j \). The learning procedure can be implemented by the following iterative EM algorithm [21].

E-step:

\[
P(y^j_i | x^j_i) = \frac{\alpha^j_y G(x^j_i, m^j_i, \sigma^j_y)}{\sum_{j=1}^{K^j} \alpha^j_y G(x^j_i, m^j_i, \sigma^j_y)}
\]  

(2.3)

where \( x^j_i \) is the input data of the \( j \)th dimension, \( G(\cdot) \) is Gaussian membership function, \( \Theta^j = \{ \theta^j_y \equiv \alpha^j_y, m^j_y, \sigma^j_y \}_{y=1}^{K^j} \) is a set of finite mixture model parameter, \( \alpha^j_y \) is the prior probability, \( m^j_y \) and \( \sigma^j_y \) refer to the mean value and the width of the \( y \)th cluster of the \( j \)th dimension.

M-step:
\[ \alpha_i^{j\text{(new)}} = \frac{1}{N} \sum_{i=1}^{N} \alpha_i^{j} G(x_i^j, m_i^{j}, \sigma_j^i) = \frac{1}{N} \sum_{i=1}^{N} P(y_i^j \mid x_i^j) \]  
\[ m_i^{j\text{(new)}} = \frac{\sum_{i=1}^{N} P(y_i^j \mid x_i^j) x_i^j}{\sum_{i=1}^{N} P(y_i^j \mid x_i^j)} = \frac{1}{\alpha_i^{j} N} \sum_{i=1}^{N} P(y_i^j \mid x_i^j) x_i^j \]  
\[ \sigma_{i}^{j\text{(new)}} = \frac{1}{\alpha_i^{j} N} \sum_{i=1}^{N} P(y_i^j \mid x_i^j) (x_i^j - m_i^{j})^2 + (h_i^j)^2 \]

where \( h_i^j \) is the smoothing parameter [19].

In the second phase, the optimal number of fuzzy cluster \( K^j^* \) is determined by cluster selection [18] as follows:

\[ K^j = \arg \min_{K_j} J(K^j) \]  
\[ J(K^j) = \sum_{y=1}^{K^j} \alpha_y^{j*} \left[ \frac{1}{2} \ln \sigma_y^{j*} + \frac{1}{2} \frac{(h_i^j)^2}{\sigma_y^{j*}} - \ln \alpha_y^{j*} \right] \]

where \( \theta_i^{j*} \) and \( h_i^j \) are the results of the parameter learning in the first phase.

In practical, for each dimension \( j \), we start with \( K^j = 1 \), estimate the parameter \( \Theta^j \) by the EM algorithm based on the given training data, and compute \( J(K^j) \). Then, we proceed to \( K^j \rightarrow K^j+1 \), and compute \( J(K^j) \) again. After we gather a series of \( J(K^j) \), the optimal cluster number, \( K^j \), is selected from the one with minimal \( J(K^j) \).
3. Ying-Yang FCMAC Structure

The generalization concept behind CMAC neural networks where similar inputs are clustered in the effort to generate the output is the key point in our model. Hence a crucial point to note about bank failure prediction with CMAC networks is that the input data needs to be similar for desired similar outcomes.

Once the data is fit for input into the CMAC network, the values will then be clustered accordingly in the association layer of the CMAC network. Inline with the principle of a CMAC output, which is “the function with similar outputs for similar inputs”, the output for bank failure prediction therefore represents the most significant cluster of similarities that have been identified in the association layer.

Classical CMAC has the advantages of local generalisation, output superposition, easy hardware implementation, incremental learning and faster learning because of the limited number of computations per cycle. However, CMAC suffers from inherent disadvantages like its inefficiency in terms of data storage, since the memory size required grows exponentially with respect to the number of input variables and its weak performance in classifying inputs that are similar and highly overlapping.

The motivation of advancing classical CMAC to FCMAC is to increase the learning capability for the model. The introduction of fuzzy membership in respective fields has the effect of smoothing the network output and increasing the approximation ability in function approximation. The FCMAC structure also reduces the memory requirement by a great deal as compared to the original CMAC.

The main difference between the FCMAC and the original CMAC is that the association layer in the FCMAC is the rule layer and each associative cell represents a fuzzy rule that links input cluster to the output cluster, so that the input data is first fuzzificated into fuzzy clusters before fed into the system. As shown in Figure 3.1, the Ying-Yang FCMAC neural network can be viewed as a 5-layer hierarchical structure as follows:
(1) **Sensor layer S.** This is the layer where the input X is obtained from the raw data (or using sensors for hardware realization).

(2) **Fuzzification sensor F.** In this layer, the novel Ying-Yang fuzzification is conducted on the input training data set to obtain fuzzy labels. Each neuron (sensor) in this layer represents a particular cluster and a group of sensors is associated with input variables. In contrary to the binary outputs of the traditional CMAC, the outputs of these sensors are real numbers from 0 to 1, which are corresponded to their membership values.

(3) **Association Layer A.** This layer is the rule layer and each association cell represents a fuzzy rule. In the case of lack of memory, this layer is considered as a conceptual/logical memory space. The AND operation is carried out to ensure that any cell is active only when **all the inputs** to it are fired. The weight of fuzzy rules is derived by truth value restriction scheme in this research.

(4) **Post association Layer P.** To address the problem of a large memory size required in Layer A, it can be mapped to a physical memory space P. This is done by either linear mapping or hashing [22]. The logical OR operation makes any cell in this layer fired if **any of its connected inputs** is activated.

(5) **Output Layer O.** This layer is fully connected to layer P. The defuzzification center of area (COA) method [23] is used to compute the output of the structure. The output of Ying-Yang FCMAC is derived by the following equation:

\[
y_o = \frac{\sum_{p=1}^{P} \omega_p \times w_p}{\sum_{p=1}^{P} \omega_p} \quad \text{for } p=1,2, \ldots, P
\]

(3.1)

where \( \omega_p \) refers to the total matching degree of the antecedent, and \( w_p \) is the weight of the \( p \)th fuzzy inference rule. The detailed fuzzy inference scheme will be described in Section 3.2.
This research is mainly focused on the fuzzification phase and the fuzzy inference scheme. We proposed a fuzzifier using BYY learning to specify the number of fuzzy sets and form clusters in the fuzzification phase. Truth-value restriction (TVR) inference scheme is then used to derive the truth-values of the rule weights.

### 3.1 Fuzzification

First of all, for each dimension, the proposed Ying-Yang fuzzification in the next section is conducted on the input training data set to obtain fuzzy clusters. Each fuzzy cluster is represented by a neuron in the Fuzzified Sensor Layer. Each combination of the fuzzy cluster in this layer becomes a cell/neuron in the next layer, Association Layer.

As illustrated in Figure 3.2, two dimensions, D₁ and D₂, are considered and represented by their Gaussian fuzzy subsets. There are 4 fuzzy labels per dimension. Hence, resulting in 16 fuzzy cells. If any value lies between the dimensions (as shown in the figure, X₁’
and $X_2$) it will be allocated to the four different points; namely cell$_1$, cell$_2$, cell$_3$, and cell$_4$, depending on the Euclidean distance between them. In other words, the closer the data point is to an intersection point the higher will be the weight for that particular intersection point with respect to the given data point. This is how fuzzification is achieved.

![Diagram of activated cells by input data X in the sensor layer.](image)

**Figure 3.2 Illustration activated cells by input data X in the sensor layer.**

### 3.2 Inference scheme

Truth-value restriction uses implication rules to derive the truth-values of the consequents from the truth-value of the antecedents. In the TVR methodology, the degree to which the actual given value of $A'$ of a variable $x$ agrees with the antecedent value $A$ in the proposition "$IF x is A then y is B$" is represented as a fuzzy subset of truth space. This fuzzy subset of truth space, known as truth-value restriction, is used in a fuzzy deduction process to determine the corresponding restriction on the truth-value of the proposition "$y is B$". The latter truth value restriction is then ‘inverted’, which means that a fuzzy
proposition “y is B’” in the Y universe of discourse is found such that its agreement with “y is B” is equal to the truth value restriction derived by the fuzzy inference process.

The use of TVR as the inference scheme is in contrast to alternative techniques such as the TSK-rule base model [24], Gaussian function [25] or use of the compositional rule of inference [26] scheme. TVR offers a consistent rule base, a strong theoretical foundation and is more logical and intuitive to the human reasoning process as compared to other schemes.

The TVR based Generalized Modus Ponens (GMP) inference process is defined as shown in Table 3.1. The TVR inference scheme uses the inverse truth function modification (ITFM) process to compute a function $\tau_{\tilde{A}}$ that would transform the fuzzy proposition $p$ to $p'$. In other words, $\tau_{\tilde{A}}$ would modify the fuzzy set $A$ to $\tilde{A}$. Thus, $\tau_{\tilde{A}}$ denotes the truth-value of the proposition $p$ given observed proposition $p'$. The GMP reasoning rule is then subsequently used to infer the corresponding conclusion from the fuzzy system by means of the TVR inference scheme. The truth-value $\tau_{\tilde{A}}$ of the antecedent propagates through the fuzzy deduction process of the TVR inference scheme to determine the corresponding truth-value $\tau_B$ of the consequent “Output is B”. Where $\tau_{\text{implication}}$ is the truth-value of the fuzzy rule induced by the implication function adopted, while $a$ and $b$ are the truth-value of the propositions $p$ and $q$ respectively, and “$\circ$” is the inference operator.

<table>
<thead>
<tr>
<th>Fuzzy rule</th>
<th>If Antecedent Then Consequent</th>
<th>$(p \rightarrow q)$ is $\tau_{\text{implication}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observed Input</td>
<td>Input is $\tilde{A}$ fuzzy proposition $p'$</td>
<td>$p$ is $\tau_{\tilde{A}} = \begin{cases} \text{sup} {\mu_A(\text{Input})}, &amp; \mu_A^{-1}(a) \neq \phi \ 0, &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>Conclusion</td>
<td>Output is $\tilde{B}$ inferred fuzzy proposition $q'$</td>
<td>$q$ is $\tau_{\tilde{B}} = \tau_{\tilde{A}}(a) \circ \tau_{\text{implication}}(I(a,b))$</td>
</tr>
</tbody>
</table>

Table 3.1. TVR-implemented system
In the proposed model, the computed truth-values of the antecedents can be effectively propagated in the hybrid structure of a neural fuzzy system. The truth-value of the proposition in the antecedent is calculated and allowed to propagate through the network. It is this value that is used to compute the proposition in the consequent. This treatment makes the TVR a viable inference scheme for implementation in a neural fuzzy system. Given the input-output data \((x^o, y^o) = (x_1^o, x_2^o, \ldots, x_n^o, y^o)\), the simplified fuzzy inference rules are shown by the following implications:

\[
\text{Rule}^p = \text{If } x_1 \text{ is } A_1^{k_1}, x_2 \text{ is } A_2^{k_2}, \ldots, x_n \text{ is } A_n^{k_n} \quad \text{Then } y \text{ is } w^p
\]  

(3.2)

where \(p=1, 2, \ldots, P\), \(x_j\) is the \(j\)th input variable, \(y\) is an output variable, \(P\) is the number of fuzzy rules and \(w^p\) is the weight of the \(p\)th fuzzy inference rule. While \(A_j^{k_l}\) is the \(k\)th fuzzy cluster of the \(j\)th dimension, \(k_l=1, 2, \ldots, K_j\). The total number of clusters \(K_j\) of each dimension is obtained by the Ying-Yang fuzzification. Given the input data \(x^o\), the total matching degree \(\omega_p\) of the antecedent of rule \(p\)th using TVR inference [27]:

\[
\omega_p = \prod_{j=1}^{n} \mu_{A_j^{k_j}}(x_j^o), \quad \text{for } j=1, 2, \ldots, n
\]  

(3.3)

where \(\mu_{A_j^{k_j}}(x_j^o)\) is a membership value, and \(\omega_p\) is the total membership value of the antecedent part.

4. Experimental Results and Analysis

The financial variables (covariates) used in the bank failure prediction application are extracted from the Call Reports, which are downloaded from the website of Federal Reserve Bank, Chicago [28]. The expected impacts of the variables on bank failures are explained. Apart from loan loss provisions for the period (PLAQLY), all the variables have been found significant in past studies [3, 4, 29, 30]. Normality plots of these
variables indicates that the variables are not normally distributed. The statistical significance of the variables is investigated by best score selection, stepwise selection and purposeful selection. Based on the findings of these selection procedures and an analysis of the correlations between the variables, only the nine variables listed in Appendix A are incorporated into the Cox’s model and the subsequent Ying-Yang FCMAC model for bank failure classification and prediction.

This section presents the simulation results using the Ying-Yang FCMAC network as a bank failure classification and early warning system (EWS) based on the nine selected financial covariates introduced at the beginning of this section. Three different sets of experiments are performed: (1) Bank failure classification using the financial covariates extracted from the last available financial statements and bank failure predictions using the same set of financial covariates but extracted from (2) one year, and (3) two years prior to the last available financial statements. The observation period of the survived (non-failing) banks consists of 21 years from January 1980 to December 2000 inclusively. For consistency, the data for the failed and survived banks have the same balance sheet dates.

4.1 Classification using last available financial statements

The original data set has been preprocessed to filter out the last available financial statement for each of the banks during the observation period. For the failed banks, it would be the records prior to failure while the records for the surviving banks are those of year 2000 (last year of the observation period). For the filtered financial statements, nine variables (known as financial covariates) are extracted. These covariates (highlighted in Section 4 and Appendix A) are selected based on classical analytical study to determine their significance and expected impact on the financial health of the banking institutions. The interim data set consists of 702 failed banks (with failure dates spreading across the entire observation period) and 2933 banks that survived the observation period, leading to
a total of 3635 observed banks. However, banks whose record has missing fields are removed leading to the final data set of 548 failed and 2555 survived (non-failing) banks. Hence, there are a total of 3103 observed banks. The failed banks constituted approximately 17.7% of the data set while the survived banks made up the remaining 82.3%.

The data set is split into one training and one test set. The training set consists of 20% of the data set while the test set contains the remaining 80%. There are five cross-validation groups. The five cross-validation groups are denoted as CV1, CV2, CV3, CV4 and CV5 respectively. Each cross-validation group consists of training and test sets that are randomly generated. The data set is initially split into two categories: failed and survived (non-failing) banks. For each cross-validation group, 20% of both two categories are randomly selected to form the training set. Hence, the number of survived banks is much greater than that of failed banks. This is termed as an “unbalanced” training scenario. The data in the training set is shuffled to randomize the presentation order. The training sets of the five cross-validation groups are mutually exclusive. One output is used to differentiate between failed and survived banks. Failed banks are denoted with output “0” while survived (non-failing) banks are identified by output “1”. The Ying-Yang FCMAC network is subsequently used to model the inherent relationships between the financial covariates and their impact on the financial solvency of the respective banks. The Ying-Yang FCMAC network is trained using the data instances in the training set and the modeling capability of the trained network is subsequently evaluated using the test set. The simulation is repeated for all the five cross-validation groups. The classification threshold (to discern between failed and survived (non-failing) banks based on the nine input financial covariates) is varied to obtain the receiver-operating-characteristic (ROC) curves depicted in Figure 4.1 Type I error is defined as the error of classifying a failed bank as a survived (non-failing) one whereas Type II error is the classification of a non-failing bank as a failed bank. Figure 4.1 illustrates the error rates for both Type I and Type II errors expressed in percentage. The EER line denotes the case of equal error rates, where both the Type I and Type II errors are the same during the classification test.
As one can easily observe from Figure 4.1 (a), Type I error rate is greater than Type II error rate for the “unbalanced” training scenario given the optimal settings of the classification threshold. For bank failure classification, the relative cost of wrongly classifying a failed bank as survived bank is much higher than that of classifying a non-failing bank as a failed bank. Thus, it is highly desirable to minimize Type I error instead of Type II error. A plausible reason for the higher Type I error rather than Type II error rate can be attributed to the overwhelming effect of survived (non-failing) banks over failed banks in the training sets of the five cross-validation groups. Hence, the Ying-Yang FCMAC network is trained to be more sensitive to the traits of survived (non-failing) banks than failed banks. Thus, the higher classification rates for survived banks over failed banks.

The simulation is subsequently repeated with a “balanced” training scenario. The training sets of the five cross-validation are modified by randomly pruning away redundant survived banks until the number of survived and failed banks is equal. The test sets of the cross-validation groups remain the same as before. The ROC curve of the newly trained Ying-Yang FCMAC network with the “balanced” training scenario is depicted in Figure 4.1 (b).

![Figure 4.1 Error Curves using last available financial records (a) unbalanced scenario (b) balanced scenario](image)
The bank failure classification results of the Ying-Yang FCMAC network trained with the “balanced” training sets displayed a smaller Type I error rate than Type II error rate as evidenced by Figure 4.1 (b). Hence, Figure 4.1 demonstrated that the Ying-Yang FCMAC network based bank failure classification and early warning system (prediction) should be trained with “balanced” training sets consisting of equal number of failed and survived (non-failing) banks. This reflects a consideration in the training of the Ying-Yang FCMAC network and most fuzzy neural models in general. That is, the ability to discern closely similar classes partly depends on the ratio of these classes in the training set. This corresponds closely to the human cognitive process, as one tends to relate to often-encountered situations.

4.2 Classification using financial statements one year prior to the last record

To implement the Ying-Yang FCMAC bank failure classification model as an early warning system, financial statements obtained one year prior to the last available records are used to train the network. Thus, the trained Ying-Yang FCMAC network can be used as a prediction (early warning) system by forecasting the financial health of the banking institutions one year in advance using the financial covariates extracted from the current financial statements. The experimental setup and generation of the training and test sets of the various cross-validation groups are similar to that of Section 4.1.

Figure 4.2 presents the prediction performances of the Ying-Yang FCMAC network for the forecasting of the financial health of the banks using financial covariates extracted from statements one year prior to the last record. Again, Figure 3 has clearly demonstrated that Type I error can be minimized in the presence of Type II error by using “balanced” training sets. Comparing Figure 4.1 and Figure 4.2, one can conclude that the classification rates of the Ying-Yang FCMAC network deteriorates from using the financial covariates extracted from one year prior to the last record. This is expected since noise and uncertainty sets in when the prediction period becomes longer.
4.3 Classification using financial statements two years prior to the last record

To evaluate the prediction accuracy and robustness of the Ying-Yang FCMAC network as an early warning system for prediction of bank failures, the experiment is repeated using financial statements two years prior to the last record. For failed banks, the records are obtained two years prior to the failure year and for the survived (non-failing) banks, the records are obtained from the year 1998. Figure 4.3 illustrates the ROC curves obtained.
Similar to what is observed in the previous results, the “balanced” training sets minimized Type I error in place of Type II error. In addition, comparing the classification results computed using the last financial statements (Figure 4.1 and Figure 4.2) and one year prior to that (Figure 4.3), it is noticed that the classification accuracy further deteriorates as the prediction period is two years in advance. Analyzing only the classification and prediction results based on “balanced” training sets, one can observe that the average classification rate for the failed banks using the last financial statements is about 95% (Figure 4.2) and it deteriorates to around 85% with statements obtained one year prior to the last record (Figure 4.3(b)) and subsequently to about 75% with financial statements two years prior to the last available record (Figure 4.3(b)). The bank failure and prediction results derived using the Ying-Yang FCMAC network are summarized in Figure 4.4.
A more detailed analysis of the bank failure classification and prediction results of the Ying-Yang FCMAC network can be performed by evaluating the equal error rates (EER) of the ROC curves. The EER values of the ROC curves and the classification rates for the various cross-validation groups (CV1-5) based on the six different sets of simulations performed (Figure 4.1-4.3) are extracted and tabulated as Table 4.1. The EER values together with the classification rates give an indication of the accuracy of the simulation results.

<table>
<thead>
<tr>
<th>Last record</th>
<th>1 year prior</th>
<th>2 years prior</th>
</tr>
</thead>
<tbody>
<tr>
<td>EER</td>
<td>Unclassified</td>
<td>EER</td>
</tr>
<tr>
<td>CV1</td>
<td>7.55%</td>
<td>1.84%</td>
</tr>
<tr>
<td>CV2</td>
<td>6.74%</td>
<td>0.99%</td>
</tr>
<tr>
<td>CV3</td>
<td>5.34%</td>
<td>3.03%</td>
</tr>
<tr>
<td>CV4</td>
<td>9.42%</td>
<td>0.78%</td>
</tr>
<tr>
<td>CV5</td>
<td>8.54%</td>
<td>1.45%</td>
</tr>
<tr>
<td>Mean value</td>
<td>7.52%</td>
<td>1.62%</td>
</tr>
<tr>
<td>c rate</td>
<td>90.86%</td>
<td>86.91%</td>
</tr>
</tbody>
</table>

*Table 4.1* Mean Classification rates using various available financial records
Thus, Figure 4.4 and Table 4.1 reinforced the observation between the failed bank classification rate and the prediction period. That is, the longer the prediction period, the less accurate is the classification and prediction result. In addition, the “balanced” training scenarios are demonstrated to be more effective in the modeling and prediction of banking failures, as shown by the mean EER values of Table 4.1.

5. Conclusion

Many statistical models such as the Cox’s model have been applied to the bank solvency analysis. However, these classical models have not attempted to identify the possible traits of financial distress that eventually leads to bank failure. It is difficult to explicitly specify what constitutes a financial distress and the intrinsic relationship between financial distress and a failed bank. This paper attempts to apply a novel fuzzy system named Ying-Yang FCMAC to bank solvency analysis. The advantage of Ying-Yang FCMAC is accrued from its fuzzification technique using Bayesian Ying-Yang learning, which obtains Gaussian clusters from a raw training data as fuzzy rules. Another advantage of Ying-Yang FCMAC is the truth-value restriction inference scheme, which provides FCMAC an intuitive fuzzy logic-reasoning framework. The trained Ying-Yang FCMAC operates as a bank failure classification and prediction system and the formulated fuzzy rule base shed lights on the inherent contributions of the selected financial covariates to bank failure. Experiments have demonstrated that the Ying-Yang FCMAC network is very encouraging in classifying failed and survived banks using a set of US banking data.

Currently, efforts are being invested to further improve the classification rates and reduce the error rates of the Ying-Yang FCMAC classification and prediction system. The focus of the research is on enhancing the bank failure prediction capability of the Ying-Yang FCMAC network for it to serve as an early warning system (EWS). Furthermore, research to develop a combined strategy for the reconstruction of missing financial data (which is often encountered in the study of bank failure) and prediction model for bank
failure analysis is actively underway. In addition, banking analysts may examine hypothetical scenarios by modifying the fuzzy quantifiers to the prediction system. Development of advanced hybrid fuzzy neural architectures for the modeling of complex, dynamic and non-linear system is also underway.

**Appendix A**

<table>
<thead>
<tr>
<th>CAMEL category</th>
<th>Covariates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capital adequacy</td>
<td>CAPADE</td>
</tr>
<tr>
<td></td>
<td>Average total equity capital (3210)/average total assets (2170) (higher is the ratio, greater is the capacity to absorb losses, smaller is the probability of failure)</td>
</tr>
<tr>
<td>Asset (loan) quality</td>
<td>OLAQLY</td>
</tr>
<tr>
<td></td>
<td>Average (accumulated) loan loss allowance (3123)/average total loans and leases, gross (1400) (smaller is the ratio, better is the loan quality, smaller is the probability of failure)</td>
</tr>
<tr>
<td></td>
<td>PROBLO</td>
</tr>
<tr>
<td></td>
<td>Average (accumulated) loans 90 + days late (1407)/average total loans and leases, gross (1400) (higher is the ratio, poorer is the loan quality, higher is the probability of failure)</td>
</tr>
<tr>
<td></td>
<td>PLAQLY</td>
</tr>
<tr>
<td></td>
<td>(Annual) loan loss provisions (4230)/average total loans and leases, gross (1400) (higher is the ratio, poorer is the loan quality expected to be, higher is the probability of failure)</td>
</tr>
<tr>
<td>Management</td>
<td>NIEOIN</td>
</tr>
<tr>
<td></td>
<td>Non-interest expense (4093)/operating income (4000) (higher is the ratio, less operationally efficient and profitable is the bank, higher is the probability of failure)</td>
</tr>
<tr>
<td>Earnings</td>
<td>NINMAR</td>
</tr>
<tr>
<td></td>
<td>Total interest income (4107) - interest expense (4073)/average total assets (2170) (higher is the net interest margin, more profitable is the bank, smaller is the probability of failure)</td>
</tr>
<tr>
<td></td>
<td>ROE</td>
</tr>
<tr>
<td></td>
<td>Net income (after tax) (4340) + applicable income taxes (4302)/average total equity capital (3210) (higher is return on equity before tax, smaller is the probability of failure)</td>
</tr>
<tr>
<td>Liquidity</td>
<td>LIQUID</td>
</tr>
<tr>
<td></td>
<td>Average cash (0010) + average federal funds sold (1350)/average total deposits (2200) + average fed funds purchased (2800) + average banks’ liability on acceptances (2920) + average other liabilities (2930)</td>
</tr>
<tr>
<td></td>
<td>(higher liquidity indicates inefficient utilization of resources; it can also reflect an expectation of unfavourable events (runs on deposits for example). Overall, higher liquidity suggests a higher probability of failure)</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>GROWLA</td>
</tr>
</tbody>
</table>
|                         | Total loans and leases, gross (1400), - total loans and leases, gross (1400) /total loans and leases, gross (1400) (with appropriate credit
control and adequate loan loss provisions, a bank with higher loan growth rate would have better profitability and smaller probability of failure.

Table A1. Definition of covariates (Numbers in brackets are the identification of the data elements from the Call Reports)

REFERENCES


