

Bank Failure Prediction and Financial Data Reconstruction using Novel Soft-Computing Approaches

W. L. Tung, C. Quek¹ and P.Y.K. Cheng²

¹ Intelligent Systems Laboratory, Nanyang Technological University,
School of Computer Engineering, Blk N4 #2A-32, Nanyang Avenue, Singapore 639798
ashcquek@ntu.edu.sg

² Mail Box 1C-90, Nanyang Business School, Nanyang Technological University,
Nanyang Avenue, Singapore 639798
aykcheng@ntu.edu.sg

Abstract. Bank failure prediction is important for the regulators (such as the central banks and the finance ministries) of the banking industries. The collapse and failure of a bank could have devastating consequences to the entire banking system and a widespread repercussion effect on other banks and financial institutions. Some of the negative impacts are the massive bail out cost for a failing bank and the negative sentiments and loss of confidence developed by investors and depositors. Very often, bank failures do not occur over night and are usually due to a prolonged period of *financial distress*. Hence, it is desirable to have an early warning system that identifies potential failing or high-risk banks through financial distress. Various traditional statistical models have been employed to study bank failures [1]-[4]. However, these models have not identified the symptoms of financial distress leading to eventual bank failure. This paper attempts to identify the financial distress (the symptoms) that leads to a bank failure using *financial covariates* derived from publicly available financial statements using a novel *neural fuzzy system* named the *Generic Self-organising Fuzzy Neural Network* (GenSoFNN) [5]. Subsequently, the performance of the *Cox proportional hazards model* [3][4] is benchmarked against that of the GenSoFNN in predicting bank failures based on a population of 3635 US banks observed over a 21 years period. In addition, it is believed that the event of financial distress does not develop out of the blue. The deterioration of the financial conditions of distressed banks can be observed over time. Thus, the performance of a bank can be tracked and studied from its annual financial statements over a period of time, which essentially is *time-series modeling*. However, it may not be possible to obtain all the financial statements or there may be missing information in the observed period of a bank. Hence, as part of the study, the *Pseudo-Outer-Product Fuzzy Neural Network* (POPFNN) [6] is used to reconstruct missing financial data that tracks the solvency (financial health) of banking institutions. The performances of both the GenSoFNN as a bank failure classification system and the POPFNN network as a tool to reconstruct missing financial data are encouraging.

Keywords: Bank failure prediction, financial distress, Cox model, neural fuzzy system, GenSoFNN, bank failure classification, time-series modeling, POPFNN, data reconstruction.

1. Introduction

Various traditional models have been employed to study bank failures. There are different concepts of *failure*—economic, business and official—and there are further distinctions within each of these concepts [7]. In this paper, *regulatory closure* is the defining event of failure, the reasons being that the event of regulatory closure is unambiguous and is more important and consistent than the straight-forward identification of *problem banks*, as such banks might come good in the future, given time or financial assistance or both. Besides, only the regulatory authorities can revoke or remove a bank's charter to operate under existing ownership. The more popular statistical methodologies used in the study of bank failures are *multivariate discriminant analysis* (MDA) [1], *logit analysis* [2] and *Cox's proportional hazards model* (the Cox model) [3][4]. Very often, bank failures do not occur over night and are usually due to a prolonged period of *financial distress*. As it is known, the collapse and failure of a bank could have devastating consequences to the entire banking system and a widespread repercussion effect on other banks and financial institutions. Some of the negative impacts are the massive bail out cost for a failing bank and the negative sentiments and loss of confidence developed by investors and depositors. Hence, it is desirable to have an early warning system that identifies potential failing or high-risk banks through financial distress. However, traditional models do not identify the symptoms of financial distress that eventually lead to bank failure.

On the other hand, problem modeling based on *soft computing* [8], which emulates a human style of reasoning when solving complex problems, can be employed to handle the modeling of both failed and survived banks. The objective of 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 called *neural fuzzy network*.

Neural fuzzy (or neuro-fuzzy) networks [9][10] such as the *Generic Self-organising Fuzzy Neural Network* (GenSoFNN) [5], the *Pseudo-Outer-Product Fuzzy Neural Network* (POPFNN) [6], the *Adaptive Neuro-Fuzzy Inference System* (ANFIS) [11] and Falcon-ART [12] are the realizations of the functionality of fuzzy systems using neural techniques. The main advantage of a neural fuzzy 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 fuzzy rules. The IF-THEN fuzzy rules are highly intuitive and easily comprehended. In addition, the *black-box* nature of the integrated neural network is resolved as the intuitive IF-THEN fuzzy rules can be used to interpret the weights and linkages of the connectionist structure. Moreover, the embedded fuzzy system in a neural fuzzy network can self-adjust the parameters of the fuzzy rules using neural network learning algorithms to achieve the desired results.

Neural fuzzy networks are universal *data-mining* tools [9][10] and possessed strong capability to derive the intrinsic relationships between the selected inputs and outputs. In addition, the generalization attribute of the neural fuzzy systems enables them to interpolate the decision-making process to new cases. This serves very well the objectives of a bank failure prediction (classification) system in the study of bank

failures since neural fuzzy networks can be employed to identify the inherent characteristics of failed banks. It allows one to appreciate the symptoms of the financial distress that leads to a bank failure. In this paper, the performance of the Cox model in predicting bank failures is compared against that of the GenSoFNN network. The GenSoFNN network is used to analyze the solvency of banks given the *financial covariates* extracted from their last available annual financial statements. That is, the banks are classified as failed or survived banks based on the selected financial indicators (covariates) extracted from the last available financial statements. More details on the selection of financial covariates will be provided in Section 4.

In addition, it is believed that financial distress does not develop out of the blue. The deterioration of the financial condition of distressed banks can be observed over time. Thus, the performance of a bank may be tracked and studied from its annual financial statements over a period of time. However, it may not be possible to obtain all the financial statements or there may be missing information during the observed period of a bank. Hence, as part of the study, the POPFNN network is used to reconstruct missing financial data that tracks the solvency (financial health) of banking institutions.

This paper is organized as follows. Section 2 briefly describes the generic structure of the GenSoFNN network. Section 3 outlines the functionality of the POPFNN network while Section 4 introduces the Cox model widely used in bank failure prediction and analysis. In addition, the process of selecting the financial indicators (covariates) used in bank failure prediction simulation in the paper is also highlighted. Section 5 presents the experimental results of the GenSoFNN network when applied to the classification of failed and survived banks and the reconstruction of missing financial data using the POPFNN network. Section 6 concludes this paper.

2. The Generic Self-organising Fuzzy Neural Network

The GenSoFNN network [5] (Fig. 1) consists of five layers of nodes. Each input node IV_i , $i \in \{1, \dots, n1\}$, has a single input x_i . The vector $X = [x_1, \dots, x_i, \dots, x_{n1}]^T$ represents all the inputs to the GenSoFNN network. Each output node OV_m , where $m \in \{1, \dots, n5\}$, computes a single output denoted by y_m . The vector $Y = [y_1, \dots, y_m, \dots, y_{n5}]^T$ denotes the outputs of the GenSoFNN network with respect to the input stimulus X . In addition, the vector $D = [d_1, \dots, d_m, \dots, d_{n5}]^T$ represents the desired network outputs required during the *parameter-learning* phase of the training cycle. The training cycle of the GenSoFNN network consists of three phases: *Self-organizing*, *rule formulation* and *parameter learning*.

The trainable weights of the GenSoFNN network are found in layers 2 and 5 (enclosed in rectangular boxes in Fig. 1). Layer 2 links contain the parameters of the input fuzzy sets while layer 5 links contain the parameters of the output fuzzy sets. The weights of the remaining connections are unity. The trainable weights (parameters) are interpreted as the corners of the trapezoidal-shaped fuzzy sets computed by the integrated *Discrete Incremental Clustering* (DIC) [5] technique. They are denoted as l and r (left and right support points), and u and v (left and right kernel points). The

subscripts denote the pre-synaptic and post-synaptic nodes respectively. For clarity in subsequent discussions, the variables i, j, k, l and m are used to refer to arbitrary nodes in layers 1, 2, 3, 4 and 5 respectively. The output of a node is denoted as Z and the subscripts specify its origin.

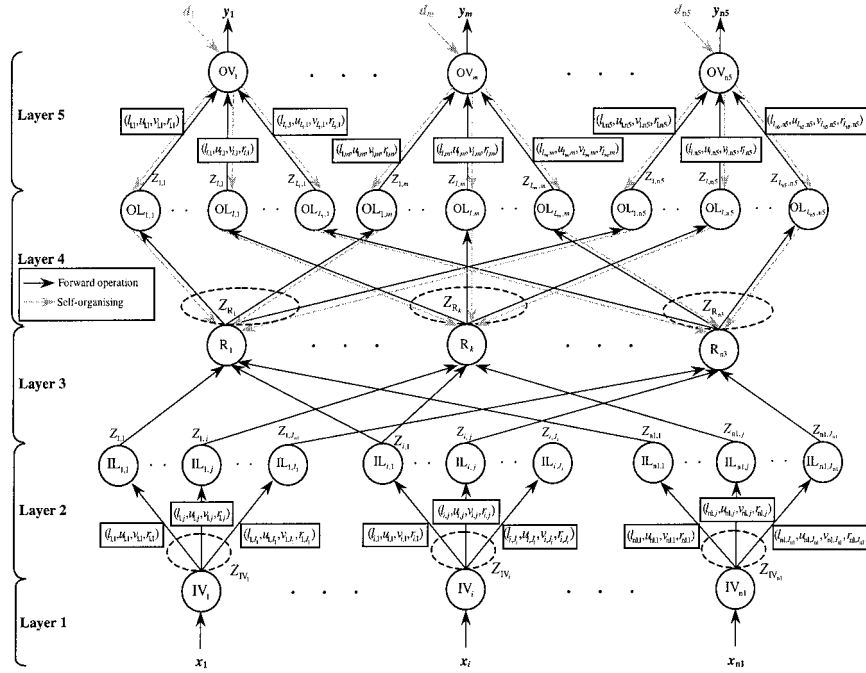


Fig. 1. Structure of the GenSoFNN network

Each input node IV_i may have different number of input terms J_i . Hence, the number of layer 2 nodes is $n2 = \sum_{i=1}^{n1} J_i$. Layer 3 consists of the rule nodes R_k , where $k = \{1, \dots, n3\}$. At layer 4, an output term node $OL_{l,m}$ may have more than one fuzzy rule attached to it. Each output node OV_m in layer 5 can have different number of output terms L_m . Hence, the number of layer 4 nodes is $n4 = \sum_{m=1}^{n5} L_m$. In Fig.1, the solid arrows denote the links that are used during the feed-forward normal operation of the GenSoFNN network. The dashed arrows denote the backward links used during the self-organizing phase of the training cycle of the GenSoFNN. The GenSoFNN network adopts the Mamdani's fuzzy rule model [10].

In this paper, the *Compositional Rule of Inference* (CRI) [13] is mapped to the GenSoFNN network to define the node functions. Please refer to [14] for more details on how this mapping is performed. As stated earlier, the training cycle of the GenSoFNN network consists of three phases: *Self-organizing*, *rule formulation* and *parameter learning*. These are performed sequentially with a single pass of the training data. The DIC technique is responsible for the self-organizing phase of the GenSoFNN network and automatically computes the input-output clusters from the nu-

merical training data. Connecting the appropriate input and output clusters during the rule formulation phase of the training cycle subsequently derives the fuzzy rules. Consequently, the parameters of the GenSoFNN network (the links of layer 5 and layer 2) are tuned during the parameter-learning phase using the *negative-descent*-based back-propagation learning algorithm [15]. The back-propagation learning equations for the GenSoFNN network are derived in [5].

3. The Pseudo-Outer-Product Fuzzy Neural Network

The POPFNN network [6] uses neural network techniques to implement a fuzzy rule-based system using the singleton fuzzifier and the CRI fuzzy inference model [13]. The resultant system has a structure as shown in Fig. 2.

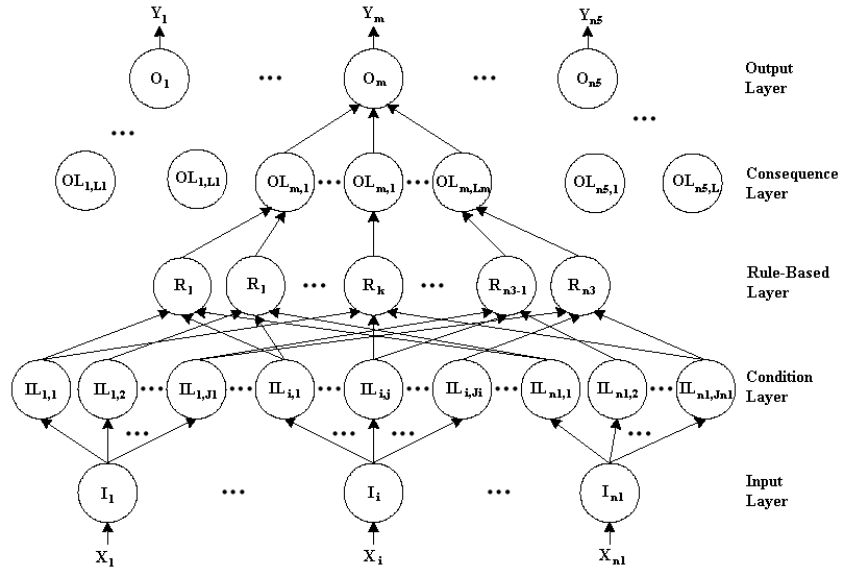


Fig. 2. Structure of the POPFNN network

The POPFNN network consists of 5 layers: Input layer, Condition layer, Rule-base layer, Consequent layer and Output layer. The inputs and outputs of POPFNN are represented by $X = [x_1, \dots, x_i, \dots, x_{n1}]^T$ and $T = [y_1, \dots, y_m, \dots, y_{n5}]^T$, where $n1$ and $n5$ denote the number of input and output variables respectively. The label Ji of Fig. 2 denotes the number of linguistic labels for the i th node in the input layer. As a result, there are a total of $\sum_{i=1}^{n1} Ji$ term nodes in the condition layer. The number of output term nodes for the m th output node in layer 5 is denoted by Lm . Hence, there are $\sum_{m=1}^{n5} Lm$ term nodes in the consequent layer. The number of fuzzy rules in POPFNN is denoted by the label $n3$. From Fig. 2, one can observed that the fuzzy rule base of the POPFNN network is a *consistent* one [10]. That is, each input/output term node represents one unique linguistic term (and its fuzzy set) and each input term can con-

tribute to the condition section of more than one fuzzy rule. Similarly, several fuzzy rules can have the same output term node as part of their consequent section.

4. Cox's proportional hazards model

The hazard function of Cox's proportional hazards model is given as

$$h(t|z) = \exp(\beta z)h_0(t) \quad (1)$$

Where z is the vector of variables for the banks and β is the corresponding vector of regression coefficients. The baseline hazard rate $h_0(t)$ is the assumed hazard rate for an *average* bank, with the values of the variables equal the population means. As all the variables are fixed at time 0, the ratio of the hazard rates of two banks with distinct values of z is a constant independent of time; that is, the hazard rates of the two banks are proportional:

$$\frac{h(t|z_1)}{h(t|z_2)} = \frac{\exp(\beta z_1)}{\exp(\beta z_2)} = \exp[\beta (z_1 - z_2)] \quad (2)$$

4.1 Financial Variables (Covariates)

The financial variables (covariates) used in the bank failure prediction and their expected impact on banking failures are defined in [7]. Apart from loan loss provisions for the period (PLAQLY) and adequacy of loan loss provisions with respect to problematic loans (ADQLLP), all the variables have been found significant in past studies [1]-[4]. Normality plots of these variables indicate that the variables are not normally distributed [7]. 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 variables CAPADE, OLAQLY, NINMAR, LIQUID, ADQLLP, GROWLA, ROE, NIEOIN, PLAQLY and PROBLO are incorporated into the Cox model and the subsequent GenSoFNN based model for bank failure classification.

5. Experimental results

This section presents the simulation results using the GenSoFNN [5] network as a bank failure classification system based on the ten selected financial covariates introduced in the last section. In addition, the results of reconstructing missing financial data using the POPFNN [6] network are also presented and discussed.

5.1 Experiment 1: Bank Failure Classification

In this paper, the US commercial banking data used for the simulation are extracted from the Call Reports downloaded from the web site of Federal Reserve Bank of Chicago, US [16]. The original data set has been preprocessed to filter out the last available financial statement for each of the banks during the observation period of 21 years from January 1980 to December 2000 inclusively.

From the filtered financial statements, ten variables (known as financial covariates) are extracted. These covariates are selected based on classical analytical study (Section 4) to determine their significance and expected impact on the financial health of the banking institutions. The inputs are denoted as *Input1* to *Input10*. 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 banks. The failed banks constituted approximately 17.7% of the data set while the survived banks made up the remaining 82.3%. Hence, there are a total of 3103 observed banks. 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 a training and test set and is randomly generated. The training sets of the five cross-validation groups are mutually exclusive. Two outputs (denoted as *Output1* and *Output2*) are used to differentiate between failed and survived banks. Failed banks are denoted with outputs “1 0” while survived banks are identified by outputs “0 1”. The GenSoFNN 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 model used for the simulation is shown as Fig. 3.



Fig. 3. The bank failure (solvency) analysis model

Since the training set of each cross-validation group is randomly generated, statistically one can assumed that survived banks would have a majority in the training set as compared to the failed banks. This scenario is labeled as “unbalanced” training. For each cross-validation group, the GenSoFNN network is trained using the training set and the generalization capability of the network is evaluated using the test set. The results for the five cross-validation groups are summarized in Table 1.

Two indicators are used to track the performance of the GenSoFNN network for the experiment. They are the *mean classification rate* (Mean c rate) and the *standard deviation* (Std Dev) of the classification rates across the five cross-validation groups. A higher “mean c rate” reflects better classification result and the “Std Dev” tracks

the consistency of the results across the five cross-validation groups. Hence, a small “Std Dev” indicates strong tolerance by the GenSoFNN network to data variations across the cross-validation groups and thus has good generalization capability.

Table 1. Classification results by the GenSoFNN network based on the “unbalanced” training scenario to differentiate between failed and survived banks given the ten financial covariates (FB = Failed Banks; SB = Survived Banks; U = Unclassified; c_rate = classification rate; FB c_rate = Failed Bank classification rate; SB c_rate = Survived Bank classification rate; Std Dev = Standard Deviation)

CV		FB	SB	U*	FB c_rate = 63.76%
	CV1	FB	278	158	0
SB		22	2022	0	Fuzzy rules = 57
CV2		FB	SB	U*	FB c_rate = 32.10%
	FB	141	298	0	SB c_rate = 99.71%
	SB	6	2038	0	Fuzzy rules = 59
CV3		FB	SB	U*	FB c_rate = 71.75%
	FB	315	124	0	SB c_rate = 99.41%
	SB	12	2032	0	Fuzzy rules = 33
CV4		FB	SB	U*	FB c_rate = 37.36%
	FB	164	275	0	SB c_rate = 99.17%
	SB	17	2027	0	Fuzzy rules = 102
CV5		FB	SB	U*	FB c_rate = 65.15%
	FB	286	153	0	SB c_rate = 98.53%
	SB	30	2014	0	Fuzzy rules = 132
Mean FB C_rate = 54.02%				Std Dev = 17.96%	
Mean SB C_rate = 99.15%				Std Dev = 0.45%	

* Note: Unclassification occurs when the GenSoFNN network computes the same output for both *Output1* and *Output2*. When this occurs, that particular bank cannot be classified based on the ten selected financial covariates.

Table 1 clearly shows that the failed bank classification rate (denoted as FB c_rate) of each of the cross-validation groups is poor and differs greatly across the different cross-validation groups. Hence the “mean FB c_rate” is low (54.02%) and the “Std Dev” of the failed bank classification rates is large (17.96%). On the other hand, the classification rates for the survived bank (SB c_rate) is high and consistently similar for all the five cross-validation groups. The “mean SB c_rate” is 99.15% and the “Std Dev” is just 0.45%. Thus, Table 1 indicated that the classification results are lop-sided and favors the survived banks. This could be due to the “unbalanced” training scenario that exists in the training sets of the different cross-validation groups. Since the survived banks have a majority in the training sets, it is thus inevitable that they have an *overwhelming effect* over the failed banks during the training cycle of the Gen-

SoFNN network. This causes the GenSoFNN network to be more sensitive to the characteristics of survived banks than that of the failed banks. Hence, the low classification rates for the failed banks but consistently high classification rates for the survived banks. Moreover, the number of fuzzy rules formulated by the GenSoFNN network to model the intrinsic relationships between the financial covariates and the eventual status of the banks (that is, either failed or survived banks) also varies greatly across the different cross-validation groups. The number of fuzzy rules ranges from 33 (CV 3) to 132 (CV5), a difference of nearly 100 fuzzy rules.

Obviously, the classification results between failed and survived banks are unacceptable in financial sense as the cost of misclassification for failed and survived banks are very different. A survived bank misclassified as failed (denoted as Type II error) suffers opportunity loss in profits when depositors and customers shun it while a failed bank misclassified as survived (defined as Type I error) may result in escalating bail out costs and a loss in confidence of the regulatory capabilities of the various control bodies. Hence, the desired result of a bank failure classification system is to have consistently high classification rates for both failed and survived banks.

Table 2. Classification results by the GenSoFNN network based on the “balanced” training scenario to differentiate between failed and survived banks given the ten financial covariates (c_rate = classification rate; FB c_rate = Failed Bank classification rate; SB c_rate = Survived Bank classification rate; Std Dev = Standard Deviation)

CV		FB	SB	U*	FB c_rate = 89.68% SB c_rate = 96.09% Fuzzy rules = 70
	CV1	FB	391	45	
	SB	80	1964	0	
CV2		FB	SB	U*	FB c_rate = 60.82% SB c_rate = 99.32% Fuzzy rules = 46
	CV2	FB	267	172	
	SB	14	2030	0	
CV3		FB	SB	U*	FB c_rate = 87.93% SB c_rate = 94.47% Fuzzy rules = 71
	CV3	FB	386	53	
	SB	113	1931	0	
CV4		FB	SB	U*	FB c_rate = 77.45% SB c_rate = 95.60% Fuzzy rules = 65
	CV4	FB	340	99	
	SB	90	1954	0	
CV5		FB	SB	U*	FB c_rate = 84.97% SB c_rate = 94.13% Fuzzy rules = 58
	CV5	FB	373	66	
	SB	120	1924	0	
Mean FB C_rate = 80.17%				Std Dev = 11.78%	
Mean SB C_rate = 96.02%				Std Dev = 1.98%	

* Note: Unclassification occurs when the GenSoFNN network computes the same output for both *Output1* and *Output2*.

To demonstrate that the poor classification results for the failed banks in the previous simulation are due to the “unbalanced” training scenario, the experiment is repeated with “balanced” training sets. That is, the test sets of the various cross-validation groups are unchanged and only the training sets are modified to reflect a “balanced” training scenario. This is achieved by discarding survived banks from the training sets until the proportion of failed to survived banks is almost unity. The bank failure classification experiment using the GenSoFNN network is repeated using these modified training sets and the results are tabulated as Table 2.

From Table 2, one can easily observed that the failed bank classification rates (FB c_rate) for the various cross-validation groups have improved tremendously. The “mean FB c_rate” is now 80.17% and the “Std Dev” is 11.78% as compared to that of Table 1 (54.02% and 17.96% respectively). Meanwhile, the classification rates for the survived banks, as tracked by the variables “mean SB c_rate” and “Std Dev”, are still comparable to the results presented in Table 1.

Hence, Table 2 justified the use of the “balanced” training scenario in the training sets of the cross-validation groups for the GenSoFNN based bank failure classification system. There is a marked improvement in the classification results of the failed banks but such an improvement is not achieved at the expense of the classification results of the survived banks. Thus, the overwhelming effect of the survived banks on the failed banks in the data set is greatly reduced. This also addresses the issue of minimizing misclassification cost for both failed and survived banks since the classification errors have been greatly reduced.

Table 3. Comparison of classification performance between the Cox model and GenSoFNN to differentiate between failed and survived banks (Type I error is defined as the percentage of failed banks misclassified as survived, and Type II error is defined as the percentage of survived banks misclassified as failed. The range of relative misclassification cost considered is 1:1 (equal misclassification costs as the bench mark) and with an increment of 5 to 30:1 where the cost of misclassifying a failed bank is 30 times higher than that of misclassifying a survived bank. The optimal cut-off point, which is used to distinguish failed banks from survived banks, is chosen such that the total probability of misclassification is minimized)

Relative misclassification cost: Failed vs. survived banks ¹	Optimal survived probability to classify failed and survived banks ²	Type I error (%)		Type II error (%)	
		Cox	GenSoFNN ³	Cox	GenSoFNN ⁴
1:1	0.73316	54.0	19.8	3.1	4.0
5:1	0.76949	49.5	19.8	4.0	4.0
10:1	0.81490	44.0	19.8	5.5	4.0
15:1	0.86031	36.5	19.8	8.8	4.0
20:1	0.90571	29.0	19.8	18.1	4.0
25:1	0.95112	18.5	19.8	42.5	4.0
30:1	0.99653	6.0	19.8	83.6	4.0

* Note: 1–The Cox model has the ability to assign different weights to the costs of misclassification for failed and survived banks. 2–The Cox model is essentially a probability-based model 3–The GenSoFNN network assumed equal misclassification costs for both failed and survived banks and 4–The GenSoFNN network does not consider the survival rates of the banks.

In addition, fewer numbers of fuzzy rules are generated by the GenSoFNN network to model the inherent characteristics of the input variables to the output status of the banks. The number of derived fuzzy rules becomes more consistent across the different cross-validation groups. The GenSoFNN network formulates an average of 62 fuzzy rules. This compares favorably against the average number of 77 rules generated by the GenSoFNN network using the “unbalanced” training scenario. Subsequently, the bank failure classification performance of the GenSoFNN network using the “balanced” training scenario is benchmarked against that of the traditional Cox model [3][4] using Table 3.

From Table 3, one can observe that the GenSoFNN based bank failure classification system consistently outperformed the traditional Cox model. Moreover, when the Cox model adjusts for the relative cost of misclassification (from 1:1 to 30:1), the misclassification error shifts from Type I error to Type II error. Hence, the issue of misclassification is still unresolved and the cost of misclassification is not minimized. For the Cox model, the worst Type I error and Type II error are 54.0% and 83.6% respectively while the respective worst errors for the GenSoFNN network is only 19.8% and 4.0%. In addition, the fuzzy rules formulated by the GenSoFNN network can be used to define the event of financial distress that is believed to eventually lead to bank failure. Such knowledge cannot be derived from the Cox model.

5.2 Experiment 2: Reconstruction of financial data

In the previous experiment, it is assumed that failed banks exhibit certain common characteristics that enable them to be differentiated from the survived banks. Hence, the GenSoFNN network is employed to automatically determine such traits. These characteristics are collectively addressed as financial distress.

However, it is believed that the event of financial distress does not develop out of the blue (although ample evidence would be reflected in the last available financial statements of banks that failed). The deterioration of the financial condition of distressed banks can be observed over time. Thus, the performance of a bank can be tracked and studied from its annual financial statements over a period of time, which essentially is time-series modeling. However, it may not be possible to obtain all the financial statements or there may be missing information during the observed period of a target bank. Hence, as part of the study, the CRI-based POPFNN [6] is used to reconstruct missing financial data that tracks the solvency (financial health) of banking institutions.

Due to constraint on paper length, only the result of one bank is presented here. The target bank has a designated FDIC identification number of 14531. FDIC is the official system used to assign unique identification numbers to the banks in the US. The target bank is henceforth referred to as Bank 14531. The preprocessed financial records of Bank 14531 consist of the ten selected financial covariates introduced in Section 4. Since the bank survived the observation period, there are 21 such records (one for each year in the observation period). These records indicated that some of the covariates have missing financial data during the period of observation. To demonstrate the feasibility of using the POPFNN network to reconstruct missing financial

data, the study is conducted only with covariates with full entries (no missing data) throughout the observation period of 21 years. Three such financial covariates for Bank 14531 are NIEOIN, NINMAR and ROE and are separately examined.

For the covariate NIEOIN, the financial data for a period of 18 years is extracted. This gives rise to 18 training instances for the POPFNN network. Due to the small number of training instances, the data is interpolated to 36 training instances. That is, half-yearly data is computed from two consecutive full year data points. With the data set of 36 instances, the middle section consisting of four half-yearly data points is filtered out as test set. This test set simulates the “missing” financial data. The 16 points before this filtered section formed the training set for the forward model while the 16 points after this section formed the training set for the backward model. Since the data points are in chronological order, therefore reconstructing the “missing” four points in the test set using the first 16 data points as training instances constitutes the forward model. The backward model is thus intuitively defined. The methodology is illustrated as Fig. 4.

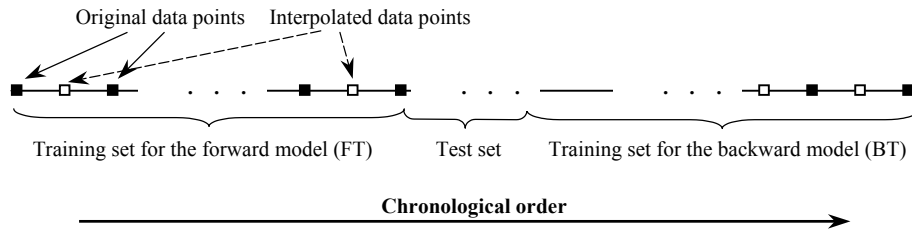


Fig. 4. Methodology to reconstruct “missing” financial data

The same applies for the “reconstruction” of the “missing” financial data of the covariates NINMAR and ROE. Two POPFNN networks are used for this experiment. One network is trained with FT (Fig. 4) as the forward model and the other network with BT as the backward model. The two models are denoted as POPFNN(F) and POPFNN(B) respectively. The reconstructed values of the “missing” points in the test set for the three covariates are shown as Table 4. The reconstructed values computed by the forward model are specified in the column POPFNN(F) while the reconstructed values derived by the backward model are shown in the POPFNN(B) column.

For the final reconstructed outputs (specified under the “Output” column in Table 4), the first two “missing” points in the test set are reconstructed using the forward model POPFNN(F) and the last two “missing” points in the test set are reconstructed using the backward model POPFNN(B). This is done in order to minimize the reconstruction error as the first two “missing” points are closer (in chronological order) to the training instances of POPFNN(F) while the last two “missing” points are closer to the training instances of POPFNN(B). Hence, the respective models would better capture the inherent trend associated with the two sets of “missing” points.

From Table 4, one can conclude that the POPFNN network is fairly efficient in reconstructing the simulated “missing” values for the three specified covariates. This observation is supported by the fairly high *average accuracy* of 81.50%, which can be viewed as a proxy to the predictive capability of the POPFNN network. In addition,

the *correlation* coefficient indicates a reading of 87.07%, which suggests a fairly similar trend between the reconstructed “missing” values of the three covariates (NIEOIN, NINMAR and ROE) and the expected values. The average accuracy and the correlation indicators are computed by augmenting the values derived for each of the three specified covariates. This concludes the experiment on the reconstruction of financial data using the POPFNN network.

Table 4. Reconstruction of “missing” financial data using POPFNN(F) and POPFNN(B) (The four “missing” data points are enumerated as 1-4 in chronological order. Except for columns 2-4, the rest of the values are expressed as percentages. The errors are computed as a percentage of the difference between the network outputs and the expected values. The column “Output” specifies the reconstructed value of the first two “missing” points using the forward model and the last two “missing” points using the backward model. Accuracy is computed by subtracting the error percentage from the whole of 100%)

Covariate NIEOIN (Scaled by 10)							
No	Expected	POPFNN(B)	POPFNN(F)	Error(B)	Error(F)	Output	Accuracy
1	3.6863	4.31035	3.06982	16.92890	-16.7235	3.06982	83.27646
2	3.8422	4.31066	3.06982	12.19249	-20.1025	3.06982	79.89745
3	3.9982	4.31066	3.06982	7.815017	-23.2199	4.31066	92.18498
4	4.1541	4.31092	3.06982	3.775066	-26.1014	4.31092	96.22493
Covariate NINMAR (Scaled by 100)							
No	Expected	POPFNN(B)	POPFNN(F)	Error(B)	Error(F)	Output	Accuracy
1	3.6822	4.43403	3.22196	20.41796	-12.4990	3.22196	87.50095
2	3.8676	4.43403	3.20170	14.64552	-17.2174	3.20170	82.78260
3	4.0529	4.43403	3.18154	9.403884	-21.4997	4.43403	90.59612
4	4.2382	4.43403	3.16801	4.620594	-25.2510	4.43403	95.37941
Covariate ROE (Scaled by 10)							
No	Expected	POPFNN(B)	POPFNN(F)	Error(B)	Error(F)	Output	Accuracy
1	1.5073	3.05191	1.02834	102.4753	-31.7760	1.02834	68.22398
2	1.8531	3.05191	1.12303	64.69214	-39.3972	1.12303	60.60277
3	2.1989	3.05191	1.19664	38.79258	-45.5801	3.05191	61.20742
4	2.5447	3.05191	1.26607	19.93202	-50.2468	3.05191	80.06798
				Average accuracy = 81.50% Correlation = 87.07%			

* Note: Average accuracy is the mean of the values specified in the “Accuracy” column. Correlation is a measure of the trend between the reconstructed and the expected data points.

6. Conclusions

Many statistical models such as the Cox’s model have been applied to the study of bank failure. However, these classical models have not identified the symptoms 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. Hence, this paper attempts to apply a novel neural fuzzy system named GenSoFNN to bank failure analysis. The trained GenSoFNN operates as a bank failure classification system and the formulated fuzzy rule base shed lights on the inherent contributions of the selected covariates to bank failure. In addition, the GenSoFNN consistently outperforms the Cox's model in classifying failed and survived banks using a set of US banking data.

Financial distress does not develop over night and the deterioration of the financial condition of distressed banks can be observed over time. Thus, the solvency of a bank can be traced and studied from its annual financial statements over a period of time. However, it may be impossible to obtain all the required data or there may be missing data during the observation period of the target bank. Hence, the POPFNN network is used as a tool to reconstruct the simulated "missing" financial data. The results are encouraging.

Currently, extensive effort has been invested at the Intelligent Systems Laboratory (ISL) [16], Nanyang Technological University, in improving the classification rate and reducing the errors of the GenSoFNN-based bank failure classification system. Furthermore, research to develop a combined strategy for the reconstruction of data and prediction model for bank failure analysis is actively underway. With the use of fuzzy linguistic rules, it become possible for banking analysts to examine hypothetical scenarios by modifying the fuzzy quantifiers to the prediction system. This is also under investigation at the ISL.

The ISL undertakes the investigation and development of advanced hybrid neural fuzzy architectures [18]-[23] for the modeling of complex, dynamic and non-linear systems. These techniques have been successfully applied to numerous novel applications such as automated car control [24], forgery detection [25] and fingerprint identification [26].

References

1. Sinkey, J. Jr.: A multivariate statistical analysis of the characteristics of problem banks. *Journal of Finance* (1975) vXXX 1: 21- 36
2. Martin, D.: Early warning of bank failure: A logit regression approach. *Journal of Banking and Finance* (1977) 1(3): 249-276
3. Lane, W., Looney, S., Wansley, J.: An application of the Cox proportional hazards model to bank failure. *Journal of Banking and Finance* (1986) 10: 511-531
4. Cole, R., Gunther, J.: Separating the likelihood and timing of bank failure. *Journal of Banking and Finance* (1995) 19(6): 1073-1089
5. Tung, W.L.: A Generalized Platform for Fuzzy Neural Network. Technical Report, ISL-TR-01/01, School of Computer Engineering, Nanyang Technological University, Singapore (2001)
6. Quek, C., Zhou, R.W.: POPFNN: A Pseudo Outer-Product Based Fuzzy Neural Network. *Neural Networks* 9(9): 1569-1581, Elsevier Science Ltd. (1996)
7. Cheng, P.Y.K.: Predicting Bank Failures: A Comparison of the Cox Proportional Hazards Model and the Time Varying Covariates Model. Ph.D. Thesis, Nanyang Business School, Nanyang Technological University, Singapore (2002)
8. Zadeh, L.A.: Fuzzy Logic, Neural Networks and Soft Computing. *Communications of ACM* (1994) 37(3): 77-84

9. Lin, C.T., Lee, C.S.G.: *Neural Fuzzy Systems – A Neuro-Fuzzy Synergism to Intelligent Systems*. Englewood Cliffs, NJ, Prentice Hall (1996)
10. Nauck, D., Klawonn, F., Kruse, R.: *Foundations of Neuro-Fuzzy Systems*. Chichester, England; New York, John Wiley (1997)
11. Jang, J.S.: ANFIS: Adaptive-Network-Based Fuzzy Inference Systems. *IEEE Trans. Systems, Man & Cybernetics* (1993) 23: 665-685
12. Lin, C.J., Lin, C.T.: An ART-Based Fuzzy Adaptive Learning Control Network. *IEEE Trans. Fuzzy Syst* (1997) 5(4): 477-496
13. Zadeh, L.A.: Calculus of fuzzy restrictions. *Fuzzy sets and Their Applications to Cognitive and Decision Processes*. Ed. New York: Academic 1-39 (1975)
14. Tung, W.L., Quek, C.: Derivation of GenSoFNN-CRI(S) from CRI-based Fuzzy System. Technical Report, ISL-TR-04/01, School of Computer Engineering, Nanyang Technological University, Singapore (2001)
15. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. In Rumelhart, D.E., McClelland, J.L. et al., eds. *Parallel Distributed Processing*, vol. 1, chap. 8, Cambridge, MA: MIT Press (1986)
16. Federal Reserve Bank of Chicago. URL—<http://www.chicagofed.org>
17. Intelligent System Laboratory, Nanyang Technological University, School of Computer Engineering. URL—<http://www.isl.sas.ntu.edu.sg>
18. Quek, C., Zhou, R.W.: Pseudo-Outer Product based Fuzzy Neural Network (Truth-Value Restriction). *Neural Network* (1996) 9(9): 1569-1581
19. Quek, C., Zhou, R.W.: POPFNN-AARS(S): A Pseudo Outer-Product Based Fuzzy Neural Network. *IEEE Trans. Syst, Man and Cyberns–Part B* (1999) 29(6): 859-870
20. Ang, K.K, Quek, C., Pasquier, M.: POPFNN-CRI(S): Pseudo Outer Product based Fuzzy Neural Network using the Compositional Rule of Inference and Singleton Fuzzifier. To appear in *IEEE Trans. Syst, Man and Cyberns–Part B* (2002)
21. Tung, W.L., Quek, C.: GenSoFNN: A Generic Self-organising Fuzzy Neural Network. To appear in *IEEE Trans. Neural Networks* (2002)
22. Gao, S.Y., Quek, C.: S-TSKfnn: A Novel Self-organising Fuzzy Neural Network based on the TSK Fuzzy Rule Model. Submitted for journal review (2002)
23. Quek, C., Tung, W.L.: A novel approach to the derivation of fuzzy membership functions using the Falcon-MART architecture. *Pattern Recognition Letters* (2001) Elsevier Science 22(9): 941-958
24. Pasquier, M., Quek, C., Toh, M.: Fuzzylot: a novel self-organising fuzzy-neural rule-based pilot system for automated vehicles. *Neural Networks* (2001) 14(8): 1099-1112 Pergamon Press
25. Quek, C., Zhou, R.W.: Antiforgery: A Novel Pseudo-Outer Product based Fuzzy Neural Network Driven Signature Verification System. To appear in *Pattern Recognition Letters* (2002)
26. Quek, C., Loh, K.H.: A Novel Direct Greyscale Minutiae Extraction in Fingerprints Using Fuzzy Neural System. Submitted for journal review (2002)