

Analysis of international debt problem using artificial neural networks and statistical methods

Berna Yazici · Memmedaga Memmedli ·
Atilla Aslanargun · Senay Asma

Received: 8 May 2009 / Accepted: 28 June 2010 / Published online: 11 July 2010
© Springer-Verlag London Limited 2010

Abstract It is known from the scientific researches that artificial neural networks are alternatives of statistical methods such as regression analysis and classification in recent years. Since multi-layer backpropagation neural network models are nonlinear, it is expected that the neural network models should make better classifications and predictions. The studies on this subject support that idea. In this study, a macro-economic problem on rescheduling or non-rescheduling of the countries' international debts is taken into account. Among the statistical methods, logistic and probit regression, and the different neural network backpropagation algorithms are applied and comparisons are made. Evaluations and suggestions are made depending on the results and different neural network architecture.

Keywords Artificial neural network · Backpropagation algorithm · Conjugate gradient method · Quasi-Newton method · Logistic and probit regression · Rescheduling and non-rescheduling of the international debts

1 Introduction

Artificial neural networks (ANN) are widely used in discrimination, sample recognition and estimation problems as an alternative of statistical models [1, 7, 17, 33]. Economics, finance and business have great importance among the problems examined using ANN techniques [1, 16, 27, 29, 30]. There is increasing number of papers on economics that applied ANN models in recent years. Leung et al. [18] compared the performance of different forecasting techniques and general regression in forecasting exchange rates. Kim et al. [15] mentioned usefulness of neural networks for early warning system of economic crisis. They studied Korean economy as a sample of world economic crisis. More examples can be given on the subject.

One of the other macroeconomic problems that are examined using ANN is international debt problem of the countries. There are also numerous researches using statistical methods on this subject. Frank and Cline [10] used multiple discriminant analysis to predict debt-servicing difficulties. Dhonte [8] used principal components analysis to obtain a description of a country's debt position. Feder and Just [9] used logit analysis in determining debt-servicing capacity. Yoon-Dae [31] made multiple regression analysis in determining of a country's creditworthiness. Kharas [13] made probit analysis to assess the probability of a country's becoming uncreditworthy. Cooper [5] used canonical correlation analysis to examine the relationship between country creditworthiness and a country's recent economic performance.

Interest on macroeconomic problems using ANN has increased since 1990s. Roy and Cosset [26] developed a neural network to estimate sovereign credit ratings. Burell and Folarin [4] used neural networks in order to maintain

B. Yazici · M. Memmedli · A. Aslanargun · S. Asma (✉)
Department of Statistics, Anadolu University,
26470 Eskisehir, Turkey
e-mail: senayolacan@anadolu.edu.tr

B. Yazici
e-mail: bernayazici@yahoo.com

M. Memmedli
e-mail: mmammadov@anadolu.edu.tr

A. Aslanargun
e-mail: aaslanar@anadolu.edu.tr

competitiveness in a global economy and collected the studies on global economy using neural networks until then. Cooper [6] compared the performance of artificial neural networks with logistic, probit regression and discriminant analysis to classify the countries to two classes; reschedule or non-reschedule the international debts. Although there are many studies including classification methods in the literature, the study in this paper differ from them with rich comparisons consist of ANN for whole data and ANN for the partitioned data (training–validation–test). This paper especially pointed out the importance of partitioning the data.

In this paper, rescheduling and non-rescheduling of the international debts of countries is examined using data for 2001. To analyze the data taken into account, logistic and probit regression among the statistical methods and different backpropagation algorithms of multilayer networks as ANN are used. As a result of classification, more successful architectures and algorithms are defined.

2 Statistical methods for classification problems

Researchers usually face with classification problems in application studies. If the problem is discrimination into two groups, logistic regression, probit regression and discriminant analysis are commonly used statistical methods.

Logistic regression is a mathematical modeling approach that can be used to describe the relationship of several independent variables x_1, x_2, \dots, x_k and a binary dependent variable y , where y is coded as 1 or 0 for its two possible categories. The logistic model describes the expected value of y , in other words the probability of occurrence of one of two possible outcomes of y . The general form of the logistic regression model is

$$y_i = E(y_i) + \varepsilon_i, \tag{2.1}$$

where the observations y_i are independent Bernoulli random variables with expected values

$$E(y_i) = p_i = \frac{1}{1 + \exp(-x_i^T \beta)} \tag{2.2}$$

and $x_i^T = (1, x_{i1}, x_{i2}, \dots, x_{ik})$, $\beta^T = (\beta_0, \beta_1, \beta_2, \dots, \beta_k)$, ε_i is i th error.

Let $\eta_i = x_i^T \beta$ be the linear predictor where η_i is defined by the logit transformation of the probability p_i

$$\eta_i = \ln \frac{p_i}{1 - p_i}.$$

The parameters $\beta^T = (\beta_0, \beta_1, \beta_2, \dots, \beta_k)$ of the linear predictor η_i is estimated by the of maximum likelihood

method. Let $\hat{\beta}$ be the final estimate of the model parameters. If the model assumptions are correct, then the estimated value of the linear predictor is constructed as

$$\hat{\eta}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \dots + \hat{\beta}_k x_{ik},$$

where k is the number of the independent variables.

Another mathematical modeling approach is probit models. Again, a nonlinear model in p is transformed so that a monotonic function of p is linear with respect to explanatory variables. The probability p_i in the i th cell or the i th observation is given by the standard cumulative normal distribution function [20, 23]:

$$F(\eta_i) = p_i = \int_{-\infty}^{\eta_i} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right) du. \tag{2.3}$$

The probit transformation is given by the inverse of the standard cumulative normal distribution function F . Solving (2.3) for η_i yields

$$\eta_i = F^{-1}(p_i) = \text{probit}(p_i)$$

Thus, the probit model can be written as

$$F^{-1}(p_i) = \eta_i = \sum_{k=0}^K \beta_k x_{ik}$$

or

$$p_i = F\left(\sum_{k=0}^K \beta_k x_{ik}\right),$$

where parameters of the probit model can be estimated by the method of ordinary least square.

There are several computer programs that help to find these parameters of the model such as STATISTICA, SAS, etc.

Since (2.2) and (2.3) give the probability of belonging one of the two groups, they are commonly used in classification problems. Discriminant analysis is another method used for binary classification problems. Since it does not give CCR for such macro-economic problems as good as logistic regression, probit regression and ANN models, it is not studied in this paper [6].

3 Backpropagation neural network algorithms for classification problems

In this study multi-layer perceptrons (MLP) with one or two hidden layers are taken into account. In fact, usually, one layer in network is sufficient in order to realize a good approximation [3, 11, 12, 33].

Backpropagation is the widespread approximation approach for training of the multi-layer feed forward neural networks based on Widrow-Hoff training rule. The main idea here is, adjusting the weights and the biases that minimize the sum of square error by propagation the error back at each epoch. In order to minimize the sum of square error, different backpropagation algorithms are constructed by applying different numeric optimization algorithms among gradient and Newton methods class.

Sum of squares for q th training sample in supervised training situations with n -inputs, m -outputs and p -hidden units, in two layers (with single hidden layer) neural network is calculated as follows:

$$E^q(w) = \sum_{k=1}^m [y_k - t_k]^2$$

$$= \sum_{k=1}^m \left[f_o \left(\sum_{j=1}^p w_{jk}^o \cdot f_h \left(\sum_{i=1}^n w_{ij}^h \cdot x_i + b_j^h \right) + b_k^o \right) - t_k \right]^2 \tag{3.1}$$

where t_k is the k th target, y_k is the k th output value; w_{ij}^h is the weight that connects the i th input and j th output units, w_{jk}^o is the weight that connects the j th hidden and k th output units, b_j^h is the bias for j th hidden unit, b_k^o is the bias for k th output unit, $f_h(\cdot)$ is activation function applied to the hidden units, and $f_o(\cdot)$ is the activation function that applied to the output units. Here, w is the vector of all weight and bias components. For simplicity of writing q sample indices is not shown up.

A nonlinear (almost an S-shaped sigmoid) $f_h(\cdot)$ function is taken. $f_o(\cdot)$ can also be taken as a linear function. For all N -training sample, the mean square error is defined as:

$$E(w) = \frac{1}{n} \sum_{q=1}^N E^q(w). \tag{3.2}$$

3.1 Standard backpropagation (BP) training algorithm

Gradient descent numeric optimization method is used to decrease error in this algorithm [3, 12]. The iteration of this method is as follows:

$$w_{k+1} = w_k - \alpha \cdot g_k$$

or

$$\Delta w_{k+1} = -\alpha \cdot g_k$$

where w_k is the current vector of weights and biases, g_k is the current gradient of error function (2.2) at the point w_k , and α is the learning (training) rate. The learning rate is crucial for BP since it determines the magnitude of weight changes.

Standard backpropagation training algorithm that includes momentum is given by the improved gradient descent with momentum formula:

$$\Delta w_{k+1} = -\alpha \cdot g_k + \mu \Delta w_k,$$

where $\Delta w_k = w_k - w_{k-1}$ is the weight change in the previous iteration, μ is the momentum coefficient. The weight change in backpropagation training algorithm with momentum is made by the combination of current gradient vector and the previous gradient vector. This change is better for the behavior of the algorithm since it provides the chance of escaping surface local minimums. Of course the better choice of α and μ constants speed up convergence of the algorithm [2, 24, 32].

3.2 Resilient backpropagation (RP) algorithm

MLPs typically use S-shaped sigmoid activation functions in the hidden layers. For this reason, their slope must approach zero as the input gets large. The purpose of the resilient backpropagation (RP) training algorithm is to eliminate these injury effects of the magnitudes of the partial derivatives. In RP algorithm the magnitude of the derivative has no effect on the weight update. Only the sign of the derivative is used to determine the direction of the weight update. If the derivative is zero, then the update value remains the same. Whenever the weights are oscillating the weight change will be reduced. If the weight continues to change in the same direction for several iterations, then the magnitude of the weight change will be increased [25].

Various versions of backpropagation training algorithms to be based on conjugate gradient and Newton methods—second order basic optimization techniques are more efficient in classification and forecasting problems [3, 12].

3.3 Conjugate gradient (CG) algorithms

Search is made conjugate directions in CG algorithms. A set of nonzero n -dimensional vectors $\{p_0, p_1, \dots, p_{n-1}\}$ is said to be conjugate with respect to the symmetric positive definite $n \times n$ matrix A if

$$p_i^T A p_j = 0 \quad \text{for all } i \neq j$$

First, let us take into account the minimum problem of square function:

$$F(w) = \frac{1}{2} w^T H w - b^T w \tag{3.3}$$

where $w \in R^n$ and H is the $n \times n$ symmetric positive definite matrix. For a starting point $w_0 \in R^n$, the method defined by the formulas below called conjugate direction method [21]:

$$w_{k+1} = w_k + \alpha_k p_k \tag{3.4}$$

$$\alpha_k = -\frac{p_k^T g_k}{p_k^T H p_k} \tag{3.5}$$

Here, (3.5) is defined by the minimum problem of one variable function of $\varphi(\alpha) = F(w_k + \alpha p_k)$.

For any starting point $w_0 \in \mathbb{R}^n$, the sequence $\{w_k\}$ generated by the conjugate direction algorithm (3.4), (3.5) converges to the minimum point w of the problem (3.3) in at most n steps [21].

The conjugate gradient method is a conjugate direction method; start out by searching in the gradient direction on the first iteration

$$p_0 = -g_0 \quad (3.6)$$

conjugate current p_k vector is calculated by using the previous p_{k-1} vector and current gradient

$$p_k = -g_k + \beta_k p_{k-1} \quad (3.7)$$

Coefficient β_k in (3.7) is selected by the condition of p_{k-1} and p_k vectors' being conjugate with respect to the symmetric positive definite $n \times n$ matrix H ($p_{k-1}^T H p_k = 0$):

$$\beta_k = \frac{g_k^T H p_{k-1}}{p_{k-1}^T H p_{k-1}} \quad (3.8)$$

There are various CG algorithms depending on different calculation formulas of β_k coefficients. Some important ones of those are as follows [3, 12, 21].

3.3.1 Fletcher-Reeves conjugate gradient algorithm (CGF)

For this algorithm β_k is defined by

$$\beta_k = \frac{g_k^T g_k}{g_{k-1}^T g_k}$$

This is the ratio of the norm squared of the current gradient to the norm squared of the previous gradient.

3.3.2 Polak–Ribere conjugate gradient algorithm (CGP)

Another version of the CG algorithm was proposed by Polak and Ribere (CGP), for which β_k is defined by

$$\beta_k = \frac{g_k^T (g_k - g_{k-1})}{g_{k-1}^T g_{k-1}}$$

Now, (3.2) error function is taken into account. Since activation functions are nonlinear, sum of mean squared error function $E(w)$ is nonlinear function of w . Using first two terms of the Taylor series expansion of $E(w)$ around w_k , we can approximate function $E(w)$ to a squared function

$$E(w_{k+1}) - E(w_k) = \nabla E(w_{k+1}) \approx g_k^T \Delta w_k + \frac{1}{2} \Delta w_k^T H_k \Delta w_k \quad (3.9)$$

where H_k is the Hessian matrix in current point w_k :

$$H_k = \frac{\partial^2 E(w_k)}{\partial w^2}$$

So, in each current iteration taking function (3.9) instead of function (3.3) suitable minimization problem is taken into account and using (3.4)–(3.8), local conjugate gradient is realized. In this case, Hessian matrix H_k is assumed positive definite.

3.3.3 Scaled conjugate gradients (SCG) algorithm

CG algorithms reviewed above makes a linear search in each iteration and that makes calculations very expensive.

Moller [19] put forward scaled conjugate gradient algorithm that does not use line search procedure in traditional CG algorithms. The basic idea of SCG is to combine the model trust region approach with the conjugate gradient approach [3, 21].

Instead of using line minimization which involves few error function evaluations, Moller proposed $H p_k$ evaluation with $O(W)$ order operations (W is the total number of weights and biases). This approach can be unsuccessful when Hessian matrix H in local approximation is not positive definite. In this case, the denominator of (3.5) $p_k^T H p_k$ will be negative and finally weight update can increase the value of error function. The problem can be overcome by modifying the Hessian matrix to ensure that it is positive definite, for example

$$H_{\text{mod}} = H + \lambda I$$

where I is the unit matrix, $\lambda \geq 0$ is a *scaling* coefficient. We can make Hessian matrix positive definite by choosing the big positive value of λ . So, the step length formula (3.5) is changed as follows:

$$\alpha_k = \frac{p_k^T g_k}{p_k^T H p_k + \lambda_k \|p_k\|^2} \quad (3.10)$$

while λ_k increases α_k decreases. The size of the trust region is adjusted by λ_k parameter. While λ_k increases trust region gets smaller. The choice of appropriate λ_k is very important for each step of SCG algorithm.

To mark from δ_k , the denominator of (3.10) Moller [19] chooses λ_k^* as follows,

$$\lambda_k^* = 2 \left(\lambda_k - \frac{\delta_k}{\|p_k\|^2} \right).$$

For this reason, $\delta_k^* > 0$ is realized. That value $\delta_k^* = -p_k^T H_k p_k$ is used instead of denominator in calculation of α_k step length in (3.10).

3.4 Quasi-Newton (QN) algorithms

The basic step of Newton's method is

$$w_{k+1} = w_k - H_k^{-1} g_k \quad (3.11)$$

where H_k is the Hessian matrix in current point w_k . Newton's method often converges faster than conjugate gradient methods. Unfortunately, it is complex and expensive to compute the Hessian matrix. Quasi-Newton (or secant) methods are based on Newton's method, but which does not require calculation of second derivatives. They update (the update is computed as a function of the gradient) an approximate Hessian matrix at each iteration of the algorithm.

3.4.1 The Broyden, Fletcher, Goldfarb and Shanno (BFGS) procedure

This is one of the successful QN algorithms. In BFGS algorithm, the approximation G of the inverse Hessian matrix in (3.11) is constructed as

$$G_{k+1} = G_k + \frac{pp^T}{p^T v} - \frac{G_k v v^T G_k}{v^T G_k v} + (v^T G_k v) u u^T,$$

where $p = w_{k+1} - w_k$, $v = g_{k+1} - g_k$, $u = \frac{p}{p^T v} - \frac{G_k v}{v^T G_k v}$,

G is a positive definite matrix, $-Gg$ is the descent way at each step of the algorithm. Weights update is made as follows:

$$w_{k+1} = w_k + \alpha_k G_k g_k,$$

where α_k is found by line minimization.

3.4.2 The one-step secant (OSS) algorithm

This is an attempt to bridge the gap between the CG algorithms and Quasi-Newton algorithms. That algorithm accepts that the previous Hessian is an identity matrix and does not calculate inverse matrixes in choosing the new search direction. This algorithm involves very light storage and fewer calculations in each step than the CG algorithms.

4 Definition and importance of rescheduling the international debts

International debt crisis occurred after financial crisis, riveted attention on political issues at the international level. If global economic justice is to be achieved, debt crises must be assessed within the broader context of the international financial system. This system has led to instability and recurrent financial crises that have severely harmed the interests of poor countries and their people [22].

The financial crises lived in the world badly affect the economies of the countries and debt crises follow economic crises, eventually the debt problem dominates all policy decisions.

Since late 1982, debt crisis has been almost universal as a result of the developing countries' fail to service their debt. Governments and multilateral agencies then stepped in, but this official activity after August 1982 should not beguile one into thinking that the major governments were absent from the game earlier [28]. The definition of the crisis in the Northern industrial countries was uniform: the onset of widespread difficulties in servicing the mountain of developing country debt threatened the stability of the international financial system [14].

At the end of 1990s, another financial turmoil in Asia that erupted in mid-1997 has abated since January, and markets have partially recovered from their troughs. Nevertheless, currency and asset values in the countries involved have been left far below precrisis levels and considerable uncertainty remains about the resolution of the crisis and its global repercussions. The crisis and its effects were apparent in December 1997. So the last economic crisis lived in the world is Asian economic crisis of 1998.

Since international debt crisis effects all economical and political balances of countries, in this study, countries' rescheduling and non-rescheduling of their international debts for 2001 data for 96 countries after Asian economic crisis of 1998 are taken into account. The effects of that crisis have occurred a few years later. So, the data related with 96 countries' rescheduling and non-rescheduling of the international debts for 2001 are used.

The data are taken from The World Bank Group web site and The World Bank e-library. The decision of which country is in rescheduled class is given by observing the tables of Stand-By Arrangements published in the Journals of IMF Survey over 1997–2001. Hence, if a country made Stand-By Arrangement by IMF, then it is chosen as a country reschedules its debt and the value of related dependent variable is 1, else that country is non-reschedule and the value of related dependent variable is 0.

The factors that affect the rescheduling and non-rescheduling of the international debts defined as follows and those factors are taken as independent variables for logistic and probit regression, and inputs for ANN:

- X_1 : Average increase in GDP growth (annual %) over 1960–2001
- X_2 : The ratio of short term debt (%GDP) to export of goods and services (%GDP)
- X_3 : Debt service ratio, the interest on total external debt plus amortization on long-term debt as a percentage of the exports of goods and services in 2001

X_4 : The import of cover and equals international reserves divided by total imports in 2001 (US \$ million)

X_5 : The ratio of export of goods and services (%GDP) to import of goods and services (%GDP).

The number of input variables is five for MLP model since all those variables are taken as input variables. The output variable can either be 1 for rescheduled debts if probability is equal or greater than 0.5 or 0 for non-rescheduled debts if probability is less than 0.5. So in output layer of the network, there is only one neuron.

5 Estimation of the results and comparison of different methods

In this study, first the architectures with two hidden layers are used and since they do not have much contribution, they are not involved. It is seen that MLP with one hidden layer gives high CCR. So MLP for 5:p:1 is taken into account. The undefined parameter here is the number of units in the hidden layer. Since the initial weights and biases are random for any network and algorithm taking different values of p between 1 and 15, 33 experiments are made and as a result of that, the networks for p ranging between 6 and 10 are preferred. The CCR for different training algorithms for these networks are given in Table 1.

According to Table 1, the best CCR are obtained by BFGS and OSS algorithms (97.92). The CCR for logit (85.42) and probit (86.46) are quite lower than the NN results.

Even if the results of training of the neural network for the whole data set are good, the results of extrapolation may be bad for the whole data. In this case the network becomes memorizer and even the small changes in data can cause big mistakes. To avoid this drawback, the data are divided into three parts for training, validation and test. At the stage of training, the neural network (adjusting the weights) only training data are used but with the help of

validation data, the performance of the network is checked at each epoch. Training automatically stops when the total error square starts to increase for validation data and in this case the parameters of neural network are obtained. At this stage, test data are taken as extraneous data. The network with minimum total square error for each of training, validation and test data is accepted the one that gives the best performance.

For the problem in question, the data set, obtained from World bank's data bank, with 96 countries is divided to 3 parts; 48 units for training, 24 units for validation and 24 units for test. Experiments are made by applying different algorithms for MLPs with one and two hidden layers. The models that give better performance and are suitable for each algorithm are chosen by changing the number of units in hidden layer. Thirty-three experiments are made for each of architecture. CCR is taken as choice criteria. As a result, the chosen models with one hidden layer are shown in Table 2 and with two hidden layers are shown in Table 3. The values in parenthesis indicate the CCR of the test data.

As results of experiments for training data set CCR is calculated as 88.54 and for test data 79.17. That means RP algorithm gives the best performance. The architectures which give that result for RP algorithm are (the number of neurons in first and second hidden layers are shown, respectively) as follows: 3:4, 4:3, 4:5, 4:6, 4:8, 4:9, 5:1–5:3, 5:5–5:8, 5:11, 6:1, 6:3–6:5, 6:7, 6:9, 6:11, 7:1–7:9, 8:1–8:6, 8:8–9:5, 9:8, 9:9, 9:11–10:6, 10:8–11:4, 11:7, 11:9. Examining the results, the second best performance is obtained by SCG algorithm.

The applications of ANN are made using MATLAB 6.0 software.

According to CCR results, it can be concluded that, when the whole data is used for training in neural network, ANN gives much better performance than logit and probit models. Especially CG algorithms make much better classifications. However, QN algorithms have the best performance for 5:9:1 (BFGS) and 5:10:1 (OSS) architected MLP with 97.92 CCR.

Table 1 Correct classification ratio for neural networks and for statistical methods

Structure of MLP	ANN for whole data							Statistical methods	
	Algorithms							Logit	Probit
	GDM	RP	SCG	CGF	CGP	BFGS	OSS		
5:6:1	92.71	95.83	96.88	96.88	94.79	93.75	96.88	85.42	86.46
5:7:1	94.79	95.83	96.88	96.88	93.75	92.71	96.88		
5:8:1	95.83	96.88	96.88	96.88	95.83	96.88	96.88		
5:9:1	94.79	95.83	96.88	96.88	95.83	97.92	96.88		
5:10:1	96.88	95.83	95.83	96.88	96.88	95.83	97.92		
5:11:1	95.83	95.83	96.88	96.88	96.88	95.83	95.83		

Table 2 Correct classification ratio of neural network algorithms with one hidden layer

Structure of MLP	Algorithms						
	GDM	RP	SCG	CGF	CGP	BFGS	OSS
<i>Neural network methods (training–validation–test)</i>							
5:6:1	88.54 (79.17)	88.54 (79.17)	88.54 (79.17)	82.29 (83.33)	86.46 (79.17)	87.50 (83.33)	85.42 (79.17)
5:7:1	88.54 (79.17)	88.54 (79.17)	87.50 (79.17)	80.21 (79.17)	86.46 (79.17)	87.50 (79.17)	87.50 (79.17)
5:8:1	86.46 (79.17)	88.54 (79.17)	88.54 (79.17)	82.29 (79.17)	82.29 (83.33)	87.50 (79.17)	86.46 (79.17)
5:9:1	87.50 (79.17)	88.54 (79.17)	87.50 (79.17)	81.25 (79.17)	84.38 (79.17)	88.54 (79.17)	85.42 (79.17)
5:10:1	86.46 (79.17)	88.54 (79.17)	87.50 (79.17)	83.33 (83.33)	86.46 (79.17)	86.46 (79.17)	85.42 (79.17)
5:11:1	86.46 (79.17)	88.54 (79.17)	88.54 (79.17)	82.29 (83.33)	87.50 (79.17)	87.50 (79.17)	87.50 (79.17)

Table 3 Correct classification ratio of neural network algorithms with two hidden layer

Structure of MLP	Algorithms						
	GDM	RP	SCG	CGF	CGP	BFGS	OSS
<i>Neural network methods (training–validation–test)</i>							
5:5:11:1	85.42 (79.17)	88.54 (79.17)	87.50 (79.17)	78.13 (83.33)	83.33 (79.17)	89.58 (83.33)	88.54 (79.17)
5:6:9:1	78.13 (83.33)	88.54 (83.33)	88.54 (79.17)	78.13 (83.33)	87.50 (79.17)	87.50 (79.17)	87.50 (79.17)
5:7:6:1	88.54 (79.17)	88.54 (79.17)	88.54 (79.17)	78.13 (83.33)	89.58 (79.17)	87.50 (79.17)	85.42 (79.17)
5:9:6:1	87.50 (79.17)	88.54 (83.33)	88.54 (79.17)	78.13 (83.33)	84.38 (79.17)	87.50 (79.17)	87.50 (79.17)
5:11:5:1	88.54 (79.17)	89.58 (83.33)	87.50 (79.17)	78.13 (83.33)	80.21 (83.33)	87.50 (79.17)	87.50 (79.17)
5:11:8:1	88.54 (79.17)	89.58 (79.17)	88.54 (79.17)	78.13 (83.33)	88.54 (79.17)	87.50 (79.17)	85.42 (79.17)
5:11:9:1	88.54 (79.17)	88.54 (79.17)	85.42 (79.17)	78.13 (83.33)	89.58 (79.17)	87.50 (79.17)	87.50 (79.17)
5:11:11:1	80.21 (83.33)	89.58 (79.17)	88.54 (79.17)	78.13 (83.33)	86.46 (83.33)	87.50 (79.17)	86.46 (79.17)

As mentioned before, if the network is trained for whole data, memorizing property comes out and causes bad estimates for extraneous data. It is seen from Table 2 that, when the data divided to 3 parts, for one hidden layer MLP, generally the CCR for training result is 88.54 and test result is 79.17. Each RP algorithm for moderate or small amount of data, gives the best and SCG gives the second best performance.

Examining Table 3, it can be seen that from two different results the CCR for whole data is 89.58 and for test

data is 83.33. As a result, those ratios are obtained for BFGS algorithm with 5:5:11:1 architected MLP and RP algorithm for 5:11:5:1 architected MLP. At this stage again, RP algorithm makes better estimations.

6 Conclusions

In this paper, rescheduling and non-rescheduling of the international debts of countries is examined using data for

2001. To analyze the data taken into account, logistic and probit regression among the statistical methods and different backpropagation algorithms of multilayer networks as ANN are used. As a result of classification, more successful architectures and algorithms are defined.

Different ANN models are compared with the logit and probit models for classification problem and results are evaluated using CCR criterion.

CCR results show that, when the whole data is used for training in neural network, ANN gives much better performance than logit and probit models. Especially, CG algorithms make much better classifications.

If the network is trained for whole data, memorizing property comes out and causes bad estimates for extraneous data. Each RP algorithm for moderate or small amount of data, gives the best and SCG gives the second best performance. At this stage again, RP algorithm makes better estimations. This result especially points out the importance of partitioning the data.

General results for the problem in question are quite high, and those can be obtained by RP algorithm. In this method, misclassifications are occurred only for Cape Verde, Nigeria and Sri Lanka among 96 countries. The second choice that gives good results for the problem is SCG algorithm. Misclassified countries here are Bosnia and Herzegovina, Cape Verde and Sri Lanka.

Consequently, ANN, as an alternative to statistical methods, usually shows better performance in estimation and classification for many empirical studies. Also ANN has a great advantage of not requiring the assumptions which are necessary for statistical methods. Moreover, ANN has another advantage for extraneous data and prediction. However, the choice of suitable architecture and algorithm has a great importance. Depending on the idea of getting better results using RP and SCG algorithms for classification problems, those algorithms are suggested for that kind of problems. It is also necessary to be cautious about the QN algorithms that can give good results at the stage of choosing the suitable network algorithms.

Acknowledgments This study was supported by the Research Fund of Anadolu University Eskisehir, Turkey. We appreciate to Prof. Dr. Ilyas SIKLAR from Department of Economics for his very valuable comments.

Appendix

See Table 4.

Table 4 Data for dependent and independent variables

Countries	Y	X ₁	X ₂	X ₃	X ₄	X ₅
Albania	0	1.82	0.04	2.4	0.41	0.47
Angola	0	1.04	0.02	26.3	0.13	1.01
Argentina	1	1.26	3.9E-07	109.6	0.53	0.44
Armenia	0	3.26	0.08	7.4	0.34	0.55
Azerbaijan	0	0.13	0.04	5.3	0.42	1.10
Bangladesh	0	3.23	0.04	7.3	0.13	0.40
Belarus	0	0.02	0.18	2.8	0.05	0.95
Belize	0	3.46	0.11	20	0.20	0.74
Benin	0	2.88	0.24	13.4	0.87	0.40
Bolivia	0	2.76	0.24	30.8	0.56	0.77
Bosnia and Herzegovina	1	2.83	0.05	19	0.47	0.45
Botswana	0	4.34	0.01	2.4	3.12	1.50
Brazil	1	2.04	0.05	76.4	0.49	0.35
Bulgaria	0	0.35	0.05	16.6	0.42	0.88
Cambodia	0	5.65	0.01	1.2	0.29	0.84
Cameroon	0	1.73	0.35	12.3	0.14	1.09
Cape Verde	1	2.51	0.12	5.5	0.30	0.48
Chad	0	0.85	0.09	10	0.15	0.30
Chile	0	3.29	0.11	29.5	0.77	1.06
China	0	-0.08	0.01	7.7	0.81	0.67
Comoros	0	1.85	0.71	6.1	0.97	0.54
Congo. Dem. Rep.	0	0.02	0.38	1.7	0.02	1.07
Congo. Rep.	0	2.39	0.37	4	0.05	1.50
Costa Rica	0	4.49	0.02	9.8	0.15	0.93
Cote d'Ivoire	0	2.01	3.73	19.9	0.21	1.27
Croatia	1	0.97	0.03	27.6	0.44	0.90
Czech Republic	0	1.30	0.02	11.2	0.34	0.96
Dominica	0	2.14	0.02	11.3	0.19	0.72
Dominican Republic	0	2.92	0.03	8.2	0.19	0.75
Ecuador	1	2.96	16.56	21.6	0.18	1.22
Egypt. Arab Rep.	0	4.59	0.02	8.9	0.59	0.79
Eritrea	0	2.77	0.15	2.2	0.09	0.26
Ethiopia	0	2.50	0.06	18.5	0.25	0.49
Gabon	1	2.54	1.16	14.3	0.01	1.49
The Gambia	0	3.55	0.01	6.7	0.37	0.73
Georgia	0	2.81	0.10	6.2	0.10	0.63
Ghana	0	2.55	23.19	12.8	0.13	0.70
Guatemala	0	3.59	0.03	9.5	0.39	0.65
Guinea	0	2.71	0.25	15.7	0.32	1.32
Guyana	0	1.68	0.19	6.6	0.37	0.85
Haiti	0	0.79	1.28E-04	5.2	0.13	1.00
Honduras	0	3.19	0.14	11.1	0.39	0.69
India	0	4.06	0.04	11.7	0.73	0.35
Indonesia	1	5.04	0.04	25.9	0.55	1.21
Iran. Islamic Rep.	0	0.81	4.5E-03	8.1	0.81	1.78

Table 4 continued

Countries	Y	X ₁	X ₂	X ₃	X ₄	X ₅
Jamaica	0	1.22	0.03	15.6	0.42	0.70
Kazakhstan	0	0.01	0.08	31.1	0.23	0.97
Kenya	0	3.22	2.1E-03	13.9	0.28	0.74
Lithuania	1	1.07	0.03	30.6	0.25	0.90
Macedonia FYR	0	-0.76	0.05	12.8	0.40	0.76
Madagascar	0	1.25	6.6E-04	6.8	0.27	1.18
Malawi	0	2.90	0.11	14.9	0.30	0.72
Maldives	0	5.78	1.01	4.6	0.21	1.24
Mali	0	1.50	0.12	9.4	0.34	0.66
Mauritania	0	1.73	0.07	18.7	0.49	0.65
Mauritius	0	4.69	0.03	6.6	0.29	1.04
Mexico	1	3.41	0.09	26.3	0.24	0.92
Moldova	0	0.68	0.03	14.7	0.21	0.67
Mongolia	0	1.80	0.21	7.4	0.26	0.80
Morocco	0	2.44	0.15	18.2	0.70	0.91
Mozambique	0	0.51	0.01	7.6	0.46	0.61
Niger	0	1.71	0.12	7.2	0.24	0.69
Nigeria	1	2.28	0.09	19	0.61	1.11
Pakistan	1	4.49	0.13	32.2	0.43	0.93
Panama	1	4.14	1.28	24.4	0.31	1.06
Paraguay	0	3.23	2.75	10.6	0.19	0.61
Peru	1	2.60	3.6E-03	22	0.94	0.90
Philippines	1	3.64	6.3E-04	21.6	0.42	0.94
Poland	0	0.02	0.01	40.4	0.57	1.20
Romania	1	0.09	0.03	18.5	0.29	0.81
Russian Federation	1	-1.31	0.02	14.6	0.50	1.54
Rwanda	0	1.51	4.7E-04	24.6	0.48	1.18
Sao Tome and Principe	0	0.98	0.99	21.2	0.45	0.39
Senegal	0	2.20	0.14	12.6	0.34	0.76
Seychelles	0	2.03	0.02	3.3	0.05	0.82
Slovak Republic	0	2.14	0.02	10.8	0.25	0.90
Sri Lanka	1	3.42	0.08	10.1	0.17	0.86
St. Lucia	0	1.66	0.02	7.1	0.25	0.97
St. Vincent & The Grenadines	0	1.78	0.02	7.9	0.27	0.82
Sudan	0	1.94	0.04	3.2	0.04	1.07
Swaziland	0	2.54	6.15E-05	2.7	0.21	0.89
Syrian Arab Republic	0	1.81	0.07	3.3	0.58	1.25
Tajikistan	0	0.02	0.12	11.4	0.12	0.84
Tanzania	0	3.57	0.39	10.3	0.33	0.40
Thailand	1	4.60	0.02	25	0.48	1.11
Trinidad and Tobago	0	2.12	0.02	4.7	0.49	1.25
Tunisia	0	3.74	0.07	13.9	0.19	0.91
Turkey	1	3.14	0.03	44	0.66	1.08
Uganda	0	4.58	0.07	4.2	0.51	1.49
Ukraine	0	-0.99	0.04	10.6	0.15	1.03
Uruguay	1	1.91	0.01	36	0.91	0.94
Uzbekistan	0	0.89	0.02	26.1	0.38	1.02

Table 4 continued

Countries	Y	X ₁	X ₂	X ₃	X ₄	X ₅
Venezuela RB	0	2.46	0.01	24.1	0.56	1.27
Vietnam	0	4.74	0.44	6.7	0.21	0.96
Yemen. Rep.	0	3.93	0.12	5	1.06	1.03
Zambia	0	1.33	1.26	12	0.07	1.02

References

- Aslanargun A, Mammadagha M, Yazici B, Yolacan S (2007) Comparison of ARIMA, neural networks and hybrid models in time series: tourist arrival forecasting. *J Stat Comput Simul* 77(1):29–53
- Bhaya A, Kaszkurewich E (2004) Steepest descent with momentum for quadratic functions is a version of the conjugate gradient method. *Neural Netw* 17:65–71
- Bishop CM (1995) *Neural networks for pattern recognition*. Clarendon Press, Oxford
- Burell PR, Folarin BO (1997) The impact of neural networks in finance. *Neural Comput Appl* 6(4):193–200
- Cooper JCB (1985) Country creditworthiness: a canonical correlation analysis. Glasgow Caledonian University economics discussion paper, no. 3
- Cooper JCB (1999) Artificial neural networks versus multivariate statistics: an application from economics. *J Appl Stat* 26:909–921
- Cheng B, Titterington D (1994) Neural networks: a review from a statistical perspective. *Stat Sci* 9(1):2–54
- Dhonte P (1975) Describing external debt situations: a roll over approach. *IMF Staff Papers* 24:159–186
- Feder G, Just R (1977) A study of debt servicing capacity applying logit analysis. *J Dev Econ* 4:25–38
- Frank CR, Cline WR (1971) Measurement of debt service capacity: an application of discriminant analysis. *J Int Econ* 1:327–344
- Hagan MT, Demuth HB, Beale MH (1996) *Neural network design*. PWS, Boston
- Haykin S (1999) *Neural networks: a comprehensive foundation*. Prentice Hall, New Jersey
- Kharas H (1984) The long run creditworthiness of developing countries: theory and practice. *Quart J Econ* 99:415–439
- Kahler M (1985) Politics and international debt: explaining the crisis. *Int Organ* 39(3):357–382
- Kim TY, Oh KJ, Sohn I, Hwang C (2004) Usefulness of artificial neural networks for early warning system of economic crisis. *Expert Syst Appl* 26(4):583–590
- Kuan CM, White H (1994) Artificial neural networks: an economic perspective. *Econ Rev* 13(1):1–91
- Kumar UA (2005) Comparison of neural networks and regression analysis: a new insight. *Expert Syst Appl* 29:424–430
- Leung MT, Chen A, Daouko H (2000) Forecasting exchange rates using general regression neural networks. *Comput Oper Res* 27:1093–1110
- Moller M (1993) Scaled conjugate gradient algorithm for fast supervised learning. *Neural Netw* 6(4):525–533
- Montgomery DC, Peck EA, Vining GG (2001) *Introduction to linear regression analysis*. Wiley, New York
- Nocedal J, Wright SJ (1999) *Numerical optimization*. Springer, New York
- Pettifor A (2003) Resolving international debt crises fairly. *Ethics Int Aff* 17(2):2–9

23. Powers DA, Yu X (2000) Statistical methods for categorical data analysis. Academic Press, London
24. Qian N (1999) Efficient backpropagation learning using optimal learning rate and on the momentum term in gradient descent learning algorithms. *Neural Netw* 102(1):145–151
25. Riedmiller M, Braun H (1993) A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In: Proceedings of the IEEE international conference on neural networks
26. Roy J, Cosset J (1990) Forecasting country risk ratings using a neural network. In: Proceedings of the 23rd Hawaii international conference on systems sciences. 4:327–334
27. Sharda R (1994) Neural networks for the MS/OR analyst: an application bibliography. *Interfaces* 24(2):116–130
28. Wellons PA (1985) International debt: the behavior of banks in a politicized environment. *Int Organ* 39(3):441–471
29. Wilson R, Sharda R (1992) Neural networks. *OR/MS Today*, August, 36–42
30. Wong B, Bodnovich T, Selvi Y (1995) A bibliography of neural networks business application research: 1988–September 1994. *Expert Syst* 12(3):253–262. http://www.worldbank.org/data/country_data
31. Yoon-Dae E (1979) Commercial banks and the creditworthiness of less developed countries. UMI Research Press, Ann Arbor, MI
32. Yu XH, Chen GA (1997) Efficient backpropagation learning using optimal learning rate and momentum. *Neural Netw* 10(3):517–527
33. Zhang GP, Patuwo EB, Hu MY (1998) Forecasting with artificial neural networks: the state of the art. *Int J Forecast* 14:35–62