Qual Quant (2016) 50:385–398
DOI 10.1007/s11135-014-0154-0

Usage of artificial neural networks for optimal bankruptcy forecasting. Case study: Eastern European small manufacturing enterprises

T. Slavici · S. Maris · M. Pirtea

Published online: 7 January 2015

Abstract Our study aims to present an optimisation method for the forecasting of bankruptcy. To this end, we elaborate and optimise an artificial neural network (ANN) which, based on the situation of real companies in Eastern Europe, can forecast bankruptcy state. After describing the network structure, the performance is evaluated. Using specific statistical methods, a statistical network optimisation is performed. The conclusion is that ANNs are extremely productive in predicting firm bankruptcy, with the forecast accuracy being higher than the accuracy obtained by traditional methods. The results are applicable at an international level, though the target group of this study contains mainly Eastern European Small Manufacturing Enterprises.

Keywords Forecast accuracy · Artificial neural network · Artificial intelligence · Pattern recognition · Bankruptcy prediction

1 Introduction

Any forecast is characterised by its accuracy. The forecasting accuracy depends on extrinsic factors (i.e., factors depending on the data set subjected to the analysis, such as data accuracy and data availability) and intrinsic factors (i.e., factors depending on the forecasting method). While a person performing a forecast can not control the extrinsic factors, the forecasting method could be chosen and improved in order to obtain higher levels of forecasting accuracy.

T. Slavici (✉) · S. Maris
Ioan Slavici University, Str. Prof. Dr. A. Paunescu- Podeanu Nr. 144, Timisoara, Romania
e-mail: titusslavici@yahoo.com

T. Slavici
Politehnica University of Timisoara, Bd. V. Parvan Nr. 2, Timisoara, Romania

M. Pirtea
West University of Timisoara, Bd. V. Parvan Nr. 4, Timisoara, Romania

This is achieved by using modern tools, such as artificial intelligence, which is today widely recognised as a popular tool for economic forecasting.

Artificial intelligence (AI) is a relatively new branch of computer science. Its origins lie in the mid-1950s, when the growing desire to obtain a machine able to reproduce human intelligence was equalled and surpassed by the development of new technologies. The most challenging AI task was creating an artificial version of the human central nervous system. This was achieved by artificial neural networks (ANNs). An ANN is strongly related to the mathematical optimisation process, i.e., finding a functional minimised by a given dataset. The obtained functional was claimed to extrapolate any dataset related to the initial data. With the development of computers and computer software, ANNs became increasingly complex and efficient and found applicability in various domains.

The learning capacity is probably the fundamental feature of a neural network, as modern algorithms plan to emulate this characteristic within computers. The fundamental ANN features can be divided into two categories: architecture and properties. An ANN's architecture defines the structure, giving the number of neurons and their connections; other features are the Input/Output (I/O), synapse intensity, deviations and activations. An ANN's properties concern the learning method, synapse reactivation, continuous associations and comparison of the new information with the existing information and how the new information is categorised.

Fausett (1994) classified ANNs and highlighted their existing applications in signal processing, car control, pattern recognition (text recognition), medicine (diagnosis and treatment suggestions), speech production and recognition and business (mortgage assessment). Even more applications of ANNs were found (Taylor 1996), including, but not limited to, the automotive and aerospace industry, digital data processing and transmission, texture recognition, clinical psychiatry, business and finance. These applications underlined the advantages of ANNs in solving complex problems, even in real-time. The financial applications considered (estimating means and medians of financial data, an on-line system for such data analysis, an S&P 500 predictor and a more general time series predictor) again demonstrate the power of forecasting and analysis of ANNs in economics. Russell and Norvig (2002) highlighted AI applications used until then: autonomous planning and scheduling, game playing (chess), autonomous car controls, medical diagnosis, logistic planning, robotics (including microsurgery), language understanding and problem solving. By now, AI has applications in almost every field of human endeavor. All of these applications involve algorithms for processing information, decision making and acting optimally based on the feedback of the environment.

Due to its higher accuracy level, AI has been recognised as a resourceful approach in economic forecasting. Balan and Morariu (2008) underline differences between AI forecast methods, including genetic algorithms and ANNs. Their accuracy levels differ slightly, with the main element of comparison being application complexity and their target groups.

The use of ANNs as an economic forecast method regarding companies' evolution dates from 1997. The study developed by Wu (1997) suggested using an ANN for identifying firms that need more complex auditing investigations and underlined the superiority of AI methods over traditional methods in forecast accuracy.

From an investment perspective, Pinson used AI in 2006 to create a multi-expert approach based on a meta-model for business risk assessment. The advantage compared to a traditional method is an adaptable system capable of giving dynamic resolution strategies to the problems. From a similar business forecast approach, AI methods have been used to forecast corporate dividends.

In 2008, Kim et al. characterised the viability of Small Manufacturing Enterprises (SMEs) by employing methods such as adaptive learning networks. The main product and management characteristics of SMEs were investigated to establish the main influence factors for their long-term survival. Their overall accuracy was 61.31 %.

In 2011, Chen proved the superior accuracy of AI methods over traditional methods in forecasting corporate financial distress. The AI methods used were principle component analysis, decision trees and logistic regression.

The Eastern European economic climate is still influenced by the political development of the region in the second half of the twentieth century. Immediately after 1990, there was a widespread enthusiasm for starting new private companies. This enthusiasm is measured at an economical level through entrepreneurial indicators. In the short term, the number of private companies increased, but, as the enthusiasm was not always a good substitute for managerial skills and experience, many new-founded companies faced bankruptcy (Pirtea 2003). Moreover, the actual state of economic crisis severely affected the young Eastern European private companies. The uncertainty of the medium- and short-term situation of a company caused unwanted market blockages. The need for a good forecasting tool for the bankruptcy of Eastern European companies thus arises. Several authors used neural networks to meet this need, including Darvasi (2010), Dorneanu et al. (2011) and proposed strategies for local use (Mnerie and Mnerie 2011).

Within the framework of European competitiveness policies, developing specific instruments for firm viability is a necessity, especially in the context of economic crisis. There is currently a push to rebuild the trust of economic agents to re-establish normal business trends (Dogaru et al. 2011). The present economic crisis led to the bankruptcy of an important part of the SMEs. Moreover, this decline led to a recession of the global economy, especially in Eastern Europe, as shown by Kim et al. (2010), Dorneanu and Untaru (2011) and also by surveys of Directorate General Economic and Financial Affairs (2011). Such a negative economic event reflected upon less-developed regions, influencing entrepreneurial behaviour (Mnerie and Mnerie 2011). According to the Directorate General of Economic and Financial Affairs, the economic sentiment indicator in the EU has decreased from a value of 106.4 in December 2010 to 92.8 in November 2011. In this view, elaborating more accurate instruments for firm viability and risk assessment gained importance (Kim et al. 2008).

In response to the need for an efficient instrument that could forecast firms' viability, we create an optimised neural network. The network considers four relevant factors regarding companies' financial states and returns a forecast of their bankruptcy state. Our work thus offers valuable information applicable to Eastern European SMEs and potentially beyond.

2 Methods

2.1 Data set

The dataset used for the computations refers to the financial state of 55 Eastern European companies. The NACE codes of the companies, together with their net annual turnover, the value of the assets and the medium number of employees classify them as small manufacturing enterprises. The data was collected from official reports and refers to the period 1994–1998. Based on the collected data, several classes of indicators were further computed. These indicators can be grouped in five main categories:

- Activity rates (5 indicators): stock rotation rate, debt collection rate, payment period of obligations, payment period of suppliers, asset leverage rate
- Cash flow rates (2 indicators): immediate cash flow, current cash flow

– Leverage rates (4 indicators): equity leverage, coverage rate of interests, coverage rate of liabilities with cash-flow, global leverage rate
– Return rates (4 indicators): operation return rate, financial return rate, assets return rate, return revenue rate
– Other economic and financial information (3 indicators): ratio of arrears over actives, ratio of arrears over turnover, dividend rate

A further stage in our work was to determine the indicators most relevant to the financial state of the companies. This stage was best described by Tucu and Rotarescu (2006) and will not be detailed here. The 4 indicators most relevant to the financial state of a company (revenue return rate, coverage rate of liabilities with cash flow, asset leverage rate and payment period of obligations) are computed as:

$$\text{Revenue return rate} = \frac{\text{Net profit}}{\text{Income}}$$

$$\text{Coverage rate of liabilities with cash flow} = \frac{\text{Cash flow}}{\text{Active}}$$

$$\text{Asset leverage rate} = \frac{\text{Liabilities}}{\text{Assets}}$$

$$\text{Payment period of obligations} = \frac{\text{Liabilities}}{\text{Turnover}} \times 360$$

Next, an ANN is constructed and trained to recognise the state of bankruptcy of a company.

As the period covered by the collected data is prior to the period in which our initial studies were performed, it was possible to check the accuracy of the ANN forecast against the reality, i.e., for each of the companies considered, the actual state of bankruptcy or non-bankruptcy was checked in 2004.

The influence of the network parameters is next studied and the ANN is optimised to obtain a high level of forecast accuracy (over 95 %).

2.2 Structure of the ANN

The structure of the ANN refers to the network topology and the algorithms used (see Fig. 1). The neural network we developed is a classification (pattern recognition) network and is implemented using the neural network toolbox of Matlab (2011).

The network topology contains one input layer of four units; one, two or three hidden layers of 10 or 20 units; and one output layer with a single output (Slavici 2009). The four units of the input layer are financial rates, chosen from the classical forecast methods: revenue return rate (X1), coverage rate of liabilities with cash flow (X2), asset leverage (X3) and the payment period of obligations (X4) (Tucu and Rotarescu 2006).

The algorithms used are presented in the following.

The initial available data were divided into 3 categories - training, validation and test. Data division could be performed in various modes: randomly, in contiguous blocks, using an interleaved selection or simply by index. However, for our purposes, we choose a random division of the available data, with the ratios of 90 % for the training set, 5 % for the validation set and 5 % for the test set.

It should be underlined that the validation of the ANN was possible, due to the time lapsed between the period studied and the study taken. This time lapse was necessary to verify the

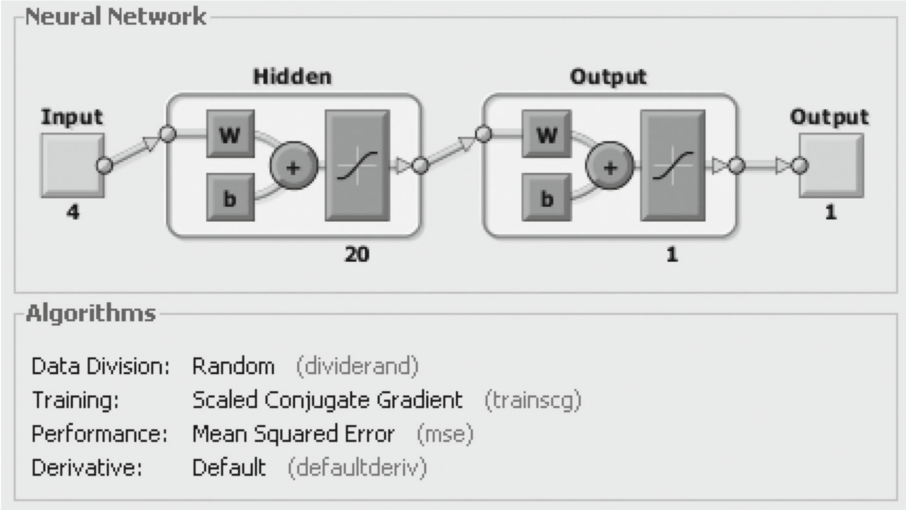

Fig. 1 Neural network (*Source* authors' elaboration with Matlab Program)

bankruptcy state of the firms. Thus, it was possible to automatically check if the validation results were accurate or not.

The training of a neural network involves tuning the values of the weights and biases of the network to optimize network performance. Different algorithms could be used, (e.g., Levenberg-Marquardt, Bayesian regularization, BFGS quasi-Newton, resilient backpropagation, different types of conjugate gradients, one step secant, variable learning rate gradient descent, etc). All these algorithms compute the current weights and biases (x_{k+1}) based on previous known weights, biases (x_k), learning rates (α_k) and gradients (g_k):

$$\mathrm{x}_{k+1} = \mathrm{x}_k - \alpha_k\, \mathrm{g}_k \tag{1}$$

For our purpose, a scaled conjugated gradient back-propagation algorithm is used for the training phase, and the maximum number of epochs to train is set to 1,000. To improve network performance, early stopping of the training algorithm is also used.

The training algorithm ends in one of the following situations: the maximum number of epochs is reached (in our case, this value is set to 1,000), the maximum time is exceeded (which is not our case, as the process is static), performance is minimised to the goal (in our case, the default goal is 0), the performance gradient falls below a predefined value (in our case, the default value is 10^{-6}) or validation performance has increased more than a predefined number of times since the last time it decreased (in our case, the default value 5). In our case, the training algorithm ends because the validation performance increased above the maximum value.

The performance of the training algorithm is measured by the mean squared normalised error function, i.e.,

$$F = \frac{1}{N}\sum_{i=1}^{N}(e_i - a_i)^2 \tag{2}$$

where N is the dimension of the training set, e_i is the normalised value for estimated output corresponding to the variable i (network output), and a_i is the actual (apriori known) normalised output corresponding to the variable i (target output).

The derivative is the default derivative and in our case, it returns the performance gradient needed for the training algorithm.

2.3 Optimisation of the ANN

Using specific statistical tools, like Statgraphics Centurion (2011), one could find the most relevant factors influencing a certain characteristic, as well as the distribution of the required characteristic, as a function of these relevant factors. In the following, we describe this process, applied to the accuracy of forecast.

After performing a statistical analysis, the conclusion was that the factors with most influence on the forecast accuracy are the number of training epochs (e) the number of layers (s) and the number of neurons per layer (n). It should be underlined that the random data division assures the generality of the obtained results.

Next, the forecasting accuracy was computed for various values of the influencing factors:

– For the number of epochs (e): 50, 100, 200 or 500;
– For the number of layers (s): 3, 4 or 5;
– For the number of neurons per layer (n): 10 or 20.

The early stopping condition was not imposed, so the experiments ended when the maximum number of training epochs was reached. Based on the obtained values, the distribution of the variable "forecasting accuracy" was computed and, hence, the optimal values for e, s and n for which the maximum of accuracy should be obtained.

3 Results and discussion

3.1 Training, validating and testing the ANN

To evaluate network performance, the confusion matrices and error rates were studied, for the two phases of the ANN construction:

Phase 1 Training the network using 49 companies.

Phase 2 Checking the network forecast accuracy using another 6 companies (3 for the validation phase and 3 for the test phase).

In the following, we describe the confusion matrices corresponding to the training and validation testing phases of the neural network build-up procedure, respectively, as well as a global confusion matrix embedding the three matrices. The matrices are structured such that the rows contain the network's output and the columns contain the desired targets (Slavici et al. 2008):

$$\begin{pmatrix} \begin{matrix} true \\ positive \end{matrix} & \begin{matrix} false \\ positive \end{matrix} \\ & \\ \begin{matrix} false \\ negative \end{matrix} & \begin{matrix} true \\ negative \end{matrix} \end{pmatrix}$$

For the training phase, from 49 total subjects, 24 (49.0 %) were marked true positives, 1 (2.0 %) as a false negative, 0 as false positives and 24 (49.0 %) as true negatives, leading to an overall accuracy of 98.0 %.

For the validation phase, from 3 total subjects, 2 (66.7 %) were marked true positive and 1 (33.3 %) true negative, leading to an overall accuracy of 100 %.

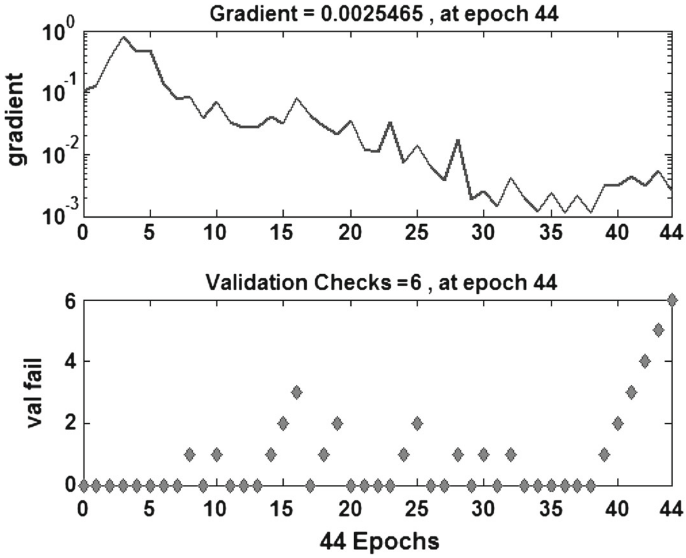

Fig. 2 Evolution of the training phase

The test phase considered 3 subjects, 2 (66.7 %) marked as true positive and 1 (33.3 %) as true negative, with an overall accuracy of 100 %.

The overall accuracy is thus 98.2 %, corresponding to 28 (50.9 %) true positive, 1 (1.8 %) false negative, 0 false positive and 26 (47.3 %) true negative marks.

This result is compared to the prediction marks obtained with classical methods, which has an overall accuracy of 92.0 %, corresponding to 48 (42.8 %) true positive, 1 (0.9 %) false negative, 8 (7.1 %) false positive and 55 (49.1 %) true negative marks.

Figure 2 shows the overall evolution of the training phase. Two aspects were monitored: the performance gradient (i.e., the backward derivative of the performance with respect to weights and biases) and the validation test (i.e., the number of times the error on the validation set increased since the last decrease). In our case, the error on the validation set increased during the last six epochs. This situation means that for the last six epochs, the network begins to overfit the data, memorising the training data rather than learning to generalise from the training set, leading to the necessity of stopping the algorithm early to obtain a network with good generalisation properties. The early stopping technique states that the parameters assigned to the network are those for which the validation error is minimised. In our case, these parameters are those corresponding to the 38th epoch (Fig. 3).

Figure 3 displays the error evolution during all 44 training epochs. The training error (shown in blue) and, even more important, the test error (shown in red) and the validation error (displayed in green) are represented. A small training error shows that the model fits the training data well, while a lower testing error signifies that the neural network displays good performance on real-life data that differs from the training set. In our case, the best validation error was obtained for the parameters corresponding to the 38th training epoch.

The results obtained using the ANN were further compared to the results obtained using the Score A method on the same input data. Tables 1, 2 and 3 gather the obtained results.

Table 1 presents training data, Table 2 presents validation data and Table 3 presents test data. The first four columns contain the first four input variables. The fifth column represents

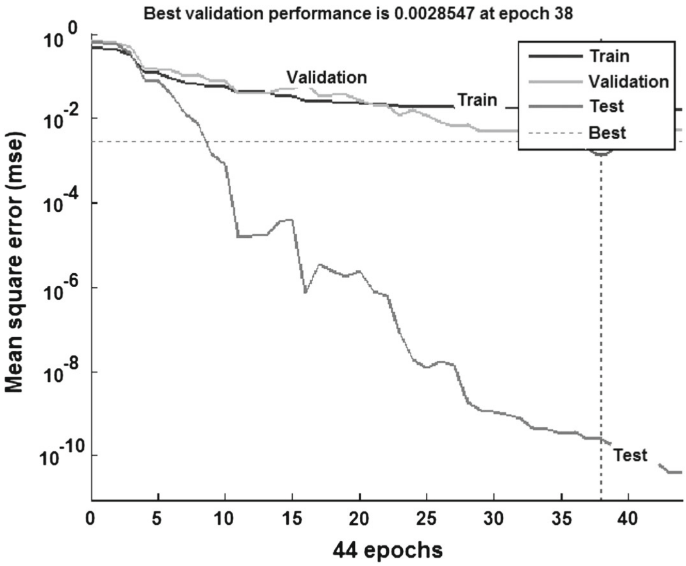

Fig. 3 Error evolution during all training epochs

Table 1 Training data (sample)

X1 (1)	X2 (2)	X3 (3)	X4 (4)	Y network (5)	A (6)	Y real (7)	True/false (8)
7.10	17.60	41.00	215.00	1.00	2.71	1.00	True
5.20	459.00	0.40	6.00	1.00	30.68	1.00	True
18.20	−9.10	4.20	39.00	0.00	3.40	1.00	False
1.20	8.40	38.40	172.00	1.00	2.42	1.00	True
0.10	0.30	16.00	302.00	1.00	1.70	1.00	True
2.00	13.00	10.60	215.00	1.00	3.70	1.00	True
0.20	50.60	9.30	38.00	1.00	7.54	1.00	True
1.20	6.50	32.60	476.00	1.00	−0.57	1.00	True
0.10	21.80	12.60	143.00	1.00	4.71	1.00	True
14.60	242.60	13.10	26.00	1.00	18.74	1.00	True
–12.00	−35.20	34.70	117.00	0.00	−0.00	0.00	True
–3.40	−13.60	50.50	116.00	0.00	0.91	0.00	True
–10.00	−11.30	57.20	305.00	0.00	−1.71	0.00	True
–42.20	−25.30	47.90	111.00	0.00	−2.01	0.00	True
–16.40	−10.80	65.70	508.00	0.00	−4.66	0.00	True
–27.90	−58.70	41.70	87.00	0.00	−2.33	0.00	True
–34.40	−37.50	52.20	466.00	0.00	−6.12	0.00	True
–57.40	−68.40	57.50	282.00	0.00	−7.59	0.00	True
–31.80	−35.10	70.70	411.00	0.00	−6.19	0.00	True
–61.80	−25.90	114.10	1.48	0.00	−5.54	0.00	True

Table 2 Validation data

X1 (1)	X2 (2)	X3 (3)	X4 (4)	Y network (5)	A (6)	Y real (7)	True/false (8)
2.30	9.00	78.20	54.00	1.00	1.72	1.00	True
18.30	144.00	13.40	90.00	1.00	12.97	1.00	True
−5.10	−6.10	55.10	113.00	0.00	1.00	0.00	True

Table 3 Test data

X1 (1)	X2 (2)	X3 (3)	X4 (4)	Y network (5)	A (6)	Y real (7)	True/false (8)
4.50	17.10	59.50	106.00	1.00	2.71	1.00	True
8.70	33.20	67.90	99.00	1.00	3.49	1.00	True
−2.70	−6.10	56.90	111.00	0.00	1.08	0.00	True

the forecast given by the network, using the following coding: 0 for bankruptcy and 1 for non-bankruptcy. The sixth column is the Score A function computed by classical methods, for which a negative value signifies that the firm is in bankruptcy, a value over 2.05 signals a non-bankruptcy state and a value between 0 and 2.05 signifies uncertainty regarding the bankruptcy state. The seventh column is the real situation of the (non-) existence of the bankruptcy and the eighth column is the accuracy of the forecast.

3.2 Optimising the ANN

The network developed in the previous section can be optimised to obtain forecasts of a higher accuracy. The optimisation can be made using two classes of methods: empirical and statistical methods.

According to Slavici (2009), six relevant factors influence the network's architecture: network topology, learning method, the nonlinearity function used, number of hidden layers, neuron number of each layer and number of training epochs. At this stage, all six factors are considered to be determined a priori. If only four different values are considered for each factor and no additional lines are made for a given set of values, $4^6 = 4,096$ experiments would be required. Using a statistical method for experiment planning is thus necessary.

A first quasi-empirical experiment plan was elaborated and presented with the preliminary conclusions by Slavici (2009). From the variety of existent neural networks typologies (architectures), the achievement of four topologies has been made using Matlab functions (in other cases, conflicts in network exploitation have occurred). The best results obtained corresponded to a feedforward type of neural network. This topology will be used below for the statistical study of the obtained results.

Tables 4 and 5 present the intermediate results of the dispersion bifactorial analysis performed successively for the two factors (epochs and layers). The Fischer criterion value - one of the strongest tools in appreciating the significance of a variable - is thus established. The computed value is compared with the value in the lookup tables. If the computed value is higher than the corresponding lookup table one, the correspondent variable is considered significant for the analysed objective function.

Table 4 Intermediate results for the first factor (number of epochs)

Stat. General monova	Main effect: no epochs (t1. Sta) 1—No. Epochs, 2—No. Layers				
Dispersion source	Square sum	Freedom degrees	Estimated dispersions	Fischer criterion	Level
Error effect	2055.222	3	685.0741	216.3392	000000
	76.000	24	3.1667		

Table 5 Intermediate results for the second factor (number of layers)

Stat. General monova	Main effect: no epochs (t1. Sta) 1—No. Epochs, 2—No. Layers				
Dispersion source	Square sum	Freedom degrees	Estimated dispersions	Fischer criterion	Level
Error effect	24.05556	2	12.02778	3.7982246	036887
	76.000	24	3.16667		

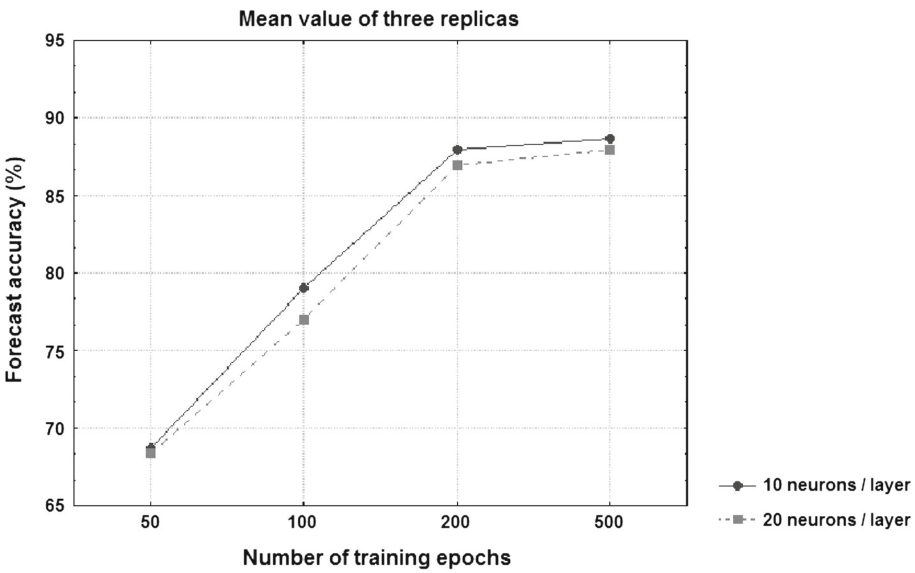

Fig. 4 Forecast accuracy versus number of neurons per layer

Figure 4 shows the forecast accuracy depending on the number of training epochs, the number of neurons per layer being considered a parameter (10 or 20 neurons per layer). It can be observed that a better accuracy is achieved using 10 neurons per layer than using 20 neurons per layer. The accuracy obtained for 500 training epochs is also slightly higher than that obtained for 200 training epochs.

Figure 5 illustrates the forecast accuracy as a function of the number of training epochs, with the number of layers considered a parameter (3, 4 or 5 layers). These values are computed based on the data included in Table 6. Table 6 reflects the forecast accuracy obtained for a specified number of training epochs and layers. For each pair of values (s,e), three repetitions

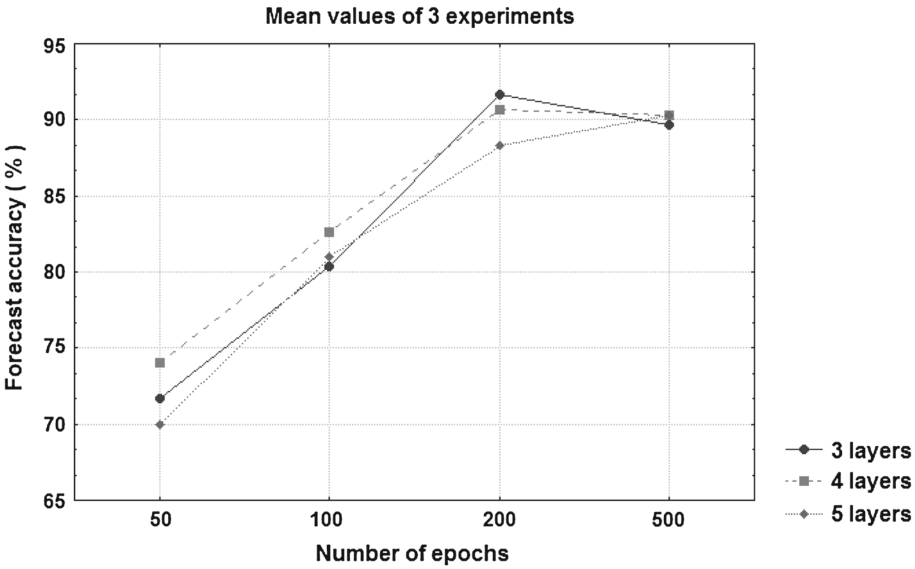

Fig. 5 Forecast accuracy versus number of layers

Table 6 Forecast accuracy as a function of e and s

Number of epochs	Number of layers	Forecast accuracy	Mean value	Dispersion
50	3	70.000	71.66666	1.527525
50	3	72.000		
50	3	73.000		
50	4	70.000	74.00000	3.605551
50	4	75.000		
50	4	77.000		
50	5	71.000	70.00000	2.645751
50	5	72.000		
50	5	67.000		
100	3	80.000	80.33334	0.577350
100	3	81.000		
100	3	80.000		
100	4	82.000	82.66666	0.577350
100	4	83.000		
100	4	83.000		
100	5	81.000	81.00000	1.000000
100	5	82.000		
100	5	80.000		
200	3	90.000	91.66666	1.527525
200	3	92.000		
200	3	93.000		
200	4	91.000	90.66666	1.527525
200	4	92.000		

Table 6 continued

Number of epochs	Number of layers	Forecast accuracy	Mean value	Dispersion
200	4	89.000		
200	5	90.000	88.33334	1.527525
200	5	88.000		
200	5	87.000		
500	3	90.000	89.66666	1.527525
500	3	91.000		
500	3	88.000		
500	4	90.000	90.33334	0.577350
500	4	91.000		
500	4	90.000		
500	5	88.000	90.33334	2.081666
500	5	91.000		
500	5	92.000		

of the experiment were performed and the mean values were computed. While a lower number of training epochs (50 or 100) requires 4 layers for better accuracy, 3 or 4 layers are required for better accuracy for a higher number of training epochs (over 200). The best accuracy obtained is for 200 training epochs and 3 layers.

Figures 4 and 5 refer to the discrete dependence of the forecast accuracy on two given sets of data: the number of training epochs and either the number of layers or the number of neurons per layer. These dependences give information about the forecast accuracy value for specific values of the parameters (n,e), respectively (s,e). To obtain more accurate answers about these dependences, two continuous functions describing the evolution of the forecast accuracy on the parameters (n,e), respectively (s,e), are required. The maximum value of these functions is then computed to obtain the values of the parameters for which the forecast accuracy is highest.

Based on the least squares method, two objective functions have been constructed, describing the dependence of the forecast accuracy on its most influencing factors. The first objective function describes the dependence of the forecast accuracy on the number of neurons per layer n and epochs e and it has the following expression:

$$\mathrm{P(n, e)} = 21.648 + 5.92 \times \mathrm{n} + 0.194 \times \mathrm{e} - 0.201 \times \mathrm{n}^2 + 0.0001 \times \mathrm{n} \times \mathrm{e}$$

The second objective function describes the dependence of the forecast accuracy on the number of layers s and epochs e and it is given as:

$$\mathrm{P(s, e)} = 43.543 + 11.501 \times \mathrm{s} + 0.182 \times \mathrm{e} - 1.542 \times \mathrm{s}^2 + 0.002 \times \mathrm{s} \times \mathrm{e}$$

Figure 6 shows the forecast accuracy for the first objective function. Good forecast accuracy for the first function (over 97 %) can be achieved using a number of neurons per layer close to 15 and a number of training epochs close to 350. Figure 7 shows the forecast accuracy for the second objective function. In this case, the best accuracy (over 95 %) can be achieved for a number of training epochs close to 350 and the number of layers has little influence on this maximum value. A good forecast accuracy, over 95 %, can be achieved with a neural network having the following structure: close to 350 training epochs, close to 15 neurons per layer and close to 4 layers.

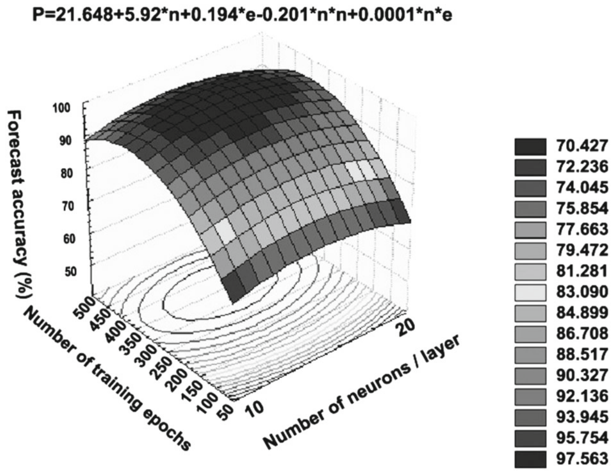

Fig. 6 The forecast accuracy of the first objective function

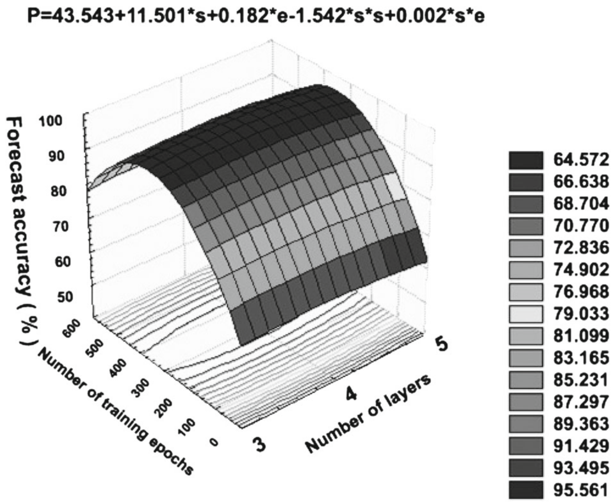

Fig. 7 The forecast accuracy of the second objective function

4 Conclusions

Using ANNs to predict company bankruptcy is extremely efficient, as the forecast accuracy is higher than in traditional methods. A data volume higher than the network training set can thus be considered. The data processing has been made using AI methods and not mathematical modelling (with statistical tools) for the following reasons:

(a) A mathematical model of the process is either unknown or too complex and is associated with insufficient accuracy; it cannot even be determined in some cases.
(b) The available data are incomplete or subject to noise in some cases.
(c) There are a number of constraints applied to the process that require being simultaneously optimised.

By studying the dependence of the forecast accuracy on the network topology, an optimised structure with a forecast accuracy over 95 % has been developed. This forecast, obtained by AI methods, is superior to those obtained by traditional methods and does not require statistical tests or research planning with factorial type tests.

The benefits brought by ANN and expert systems compared to the classical economic forecast and decision making methods include: higher forecast accuracy, faster decision making and the possibility of dynamical update of the wide databases constructed. The increasing use of AI methods is thus justified and represents an added value for the financial environment.

References

Balan, I., Morariu, N.: Genetic algorithms vs artificial neural networks in economic forecasting process. J. Appl. Comput. Sci. Math. **2**(4), 9 (2008)

Chen, M.: Predicting corporate financial distress based on integration of decision tree classification and logistic regression. Expert Syst. Appl. **38**(9), 11261–11272 (2011)

Darvasi, D.: Foundation and association accounting using traditional and classical methods and artificial intelligence systems. "Ioan Slavici" Foundation for culture and education Publishing House, Timisoara (2010)

Directorate General Economic and Financial Affairs.: Business and Consumer Surveys (2011). http://ec.europa.eu/economy_finance/db_indicators/surveys/index_en.htm Accessed on November 1, 2011

Dogaru, T., Van Oort, F., Thissen, M.: Agglomeration economies in European regions: perspectives for objective 1 regions. J. Econ. Soc. Geogr. **102**(4), 486–494 (2011)

Dorneanu L., Untaru M.: Law and financial new challenges in agribusiness. In: Proceedings of the 7th International Conference of Integrated Systems for Agri-food production, Nyiregyhaza, pp. 309–312 (2011)

Dorneanu L., Untaru M., Darvasi D., Rotarescu V., Cernescu, L.: Using Artificial Neural Networks in Financial Optimization. In: Proceedings of the Fifth Wseas International Conference on Business Administration (ICBA '11), WSEAS University Press, Puerto Morelos, Mexico, pp. 93–96 (2011)

Fausett, L.: Fundamentals of Neural Networks: Architectures, Algorithms and Applications. Prentice Hall, Upper Saddle River (1994)

Kim, J., Won, C., Bae, J.K.: A knowledge integration model for the prediction of corporate dividends. Expert Syst. Appl. **37**(2), 1344–1350 (2010)

Kim, K.S., Knotts, T.L., Jones, S.C.: Characterizing viability of small manufacturing enterprises (SME) in the market. Expert Syst. Appl. **34**(1), 128–134 (2008)

Mnerie, D., Mnerie, G.: Marketing strategies for economic and social development of Banat region. In: Proceedings of Transnational Sustainable Methods for Quality in Higher Education, Timisoara, pp. 254–259 (2011)

Neural Network Toolbox Documentation (2011). http://www.mathworks.com/help/toolbox/nnet/ Accessed on December 2, 2011

Pinson, S.: Credit risk assessment and meta-judgement. BT Technol. J. **24**(3), 49–56 (2006)

Pirtea, M.: Treasury Management for Companies. Mirton Publishing House, Timisoara (2003)

Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice Hall Series in Artificial Intelligence, Upper Saddle River (2002)

Slavici, T.: Artificial Intelligence. "Ioan Slavici" Foundation for culture and education Publishing House, Timisoara (2009)

Slavici, T., Gubencu, D., Groza, I.: Introduction in Informatics. "Ioan Slavici" Foundation for culture and education Publishing House, Timisoara (2008)

Statgraphics Centurion Documentation (2011). http://www.statlets.com/documents.htm Accessed on December 2, 2011

Taylor, T.G. (ed.): Neural Networks and Their Applications. Wiley, Hoboken (1996)

Tucu, D., Rotarescu, V.: Economy and Company's Management. Orizonturi Universitare Publishing House, Timisoara (2006)

Wu, R.C.: Neural network models: foundations and applications to an audit decision problem. Ann. Oper. Res. **75**, 291–301 (1997)