International Journal of Economics & Strategic Management of Business Process (ESMB)
Vol.17 pp.1-5

Neural networks for Predicting supply chain risks: Back propagation training algorithm

Mariem MRAD
Laboratory of Economics and Management
Faculty of Economics Sciences and Management of Sfax
University of Sfax, Tunisia
mradmariem0@gmail.com

Younes BOUJELBENE
Laboratory of Economics and Management
Faculty of Economics Sciences and Management of Sfax
University of Sfax, Tunisia
younes.boujelbene@gmail.com

***Abstract*—Managing supply chain risks has received attention in recent years, with the goal of protecting supply chains from disruptions by predicting their occurrence and mitigating their adverse effects. At the same time, the resurgence of Artificial Intelligence (AI) has led to investigation of machine learning techniques and their applicability in supply chain risk management. However, most works focus on prediction performance and the importance of interpretability so that results can be understood by supply chain practitioners, helping them make decisions that can mitigate or prevents risks from occurring. In this work, we focus on the risk assessment of supply chain based on Back propagation (BP) neural network. The risk assessment procedure is discussed and after the risk factors of supply chain identification and analysis, the risk assessment model is built with BP neural network. Through training of the model using MATLAB, the model shows the preciseness and comprehensive practicability.**

***Keywords*—supply chain risk, risk assessment, BP neural network.**

I. INTRODUCTION

Supply chain management (SCM) is defined as a set of methods used to interconnect suppliers, manufacturers, warehouses and clients so that the merchandise is produced and distributed at the right quantities, to the right places at the right time with the objective of minimizing global system costs and maximizing the customer service levels [1]. There exist numerous quantitative methods for the SCM [2]. The majority of these methods can be regrouped into two classes: methods based on the discrete event simulation and the methods based on the mathematical programming techniques. SCM based on discrete event simulation generally deals with the tactical and operational level decisions such as inventory control, material handling, layout design, and vehicle routing/scheduling, while the mathematical programming techniques are mostly used for long-term strategic decision-making. In modern supply chain networks, in order to increase their competitive edge, the firms employ new strategies such as recentring their activities by outsourcing some part of their production, proposing increased diversity of products to capture the market share and so on. Even though efficient in a stable environment, these strategies augment the vulnerabilities of the firms in an uncertain environment, thus resulting in operational risks to take into account. The field of supply chain risk management (SCRM), which emerged in the early 2000s has now become more than the overlap of directly related areas such as enterprise risk management and supply chain management [3]. As defined in [4], SCRM "encompasses the collaborative and coordinated efforts of all parties involved in a supply chain to identify, asses, mitigate and monitor risks with the aim to reduce vulnerability and increase robustness and reliance of supply chain, ensuring profitability and continuity".

The wide range of decisions and actions that are involved in SCRM have led to an equally wide spectrum of solutions proposed by researchers. These can be broadly classified in three categories: (1) multiple-criteria decision analysis techniques; (2) mathematical modelling and optimization; and (3) Artificial Intelligence (AI) techniques [19].

AI techniques have received relatively little attention in relation to SCRM or supply chain research, in general. Recently, there has been an upsurge in AI due to the availability of increased computing power and large amounts of data, as well as the success of approaches within the broad area of machine learning. This has also led to SCRM researchers considering the potential of AI techniques in relation to tasks such as risk identification, prediction, assessment and response [5-6].

Since artificial neural network is a learning system which can develop the knowledge more than the designers' original level of knowledge, it is well positioned to overcome the subjectivity of risk assessment in which subjectivity is inevitable.[7] In this paper, Back Propagation (BP) neural network, which is currently one of the most widely used and successful neural networks, is applied in supply chain risk assessment to effectively overcome the artificial nature of the assessment and to ensure the accuracy and objectivity of the risk assessment results [18].

II. SUPPLY CHAIN RISK ASSESSMENT PROCEDURES

1. The Procedures

In the main, risks are assessed by a function with "p" and "c" as parameters. Here "p" is the probability of risk event occurring and is decided by the possibility of a threat occurring and a weak point being attacked; and "c" is for the consequences of risk events occurring, or the value of the lost assets. Thus, the risk estimation process includes identification of risk factors and risk factor analysis in two stages.

2. *Supply Chain Risk Identification*

Risk identification refers to the process of identifying the different risk factors in the supply chain, which is made to identify and record the risk process thence all risk factors and their relationships are identified. In a supply chain, materials

ISSN 1737-9288

flow through a large number of production enterprises to customers, and business flow, logistics, and information flow is generated in the process. The supply chain process is associated with many processes including the transportation, storage, loading and unloading, Handling, packaging, distribution processing, distribution, information processing and many other processes, in which any one link will cause problems with supply chain risks. Therefore, many risk factors are affecting the normal operation of the supply chain. It is generally believed that the supply chain risks usually come from the natural environment and social environment in two aspects. [8] Through them, the risk factors caused by the natural environment are generally difficult to control and forecast. Therefore, we only consider the risk factors caused by the social environment in this paper. The risk factors caused by the social environment include the sole supplier risks, the information transmission risks, the logistics and distribution risks, the financial risks, market volatility risks, partner risks and profit distribution risks. For the sake of brevity, this paper takes the information risks, financial risks, logistics risks, time risks, and organizational risks as the supply-chain risk assessment factors.

3. Supply Chain Risk Analysis

On the basis of risk identification, risk analysis can further improve risk. Its goal is to collect enough information on the risks so as to determine the probability of the occurrence of various types of risk, as well as the consequences in case the risk events happen. Risks can be estimated with this information. In this paper, we take the risk ratings as the input vectors of the BP neural network, and the risk assessment value as the output of the neural network. After the network training with some samples which are evaluated successfully using traditional methods, the weight values of the network become correct through the adaptive learning process. Thus, the well-trained neural network can be used as an effective tool of risk assessment [9].

III. NEURAL MODELING AND SYNTHESIS

1. Neural networks

Neural networks, like powerful computational tools, have been typically applied to a wide range of tasks, such as function approximation, pattern recognition, identification of complex systems and times series prediction [10]. Neural network Modeling requires the step of model selection, which is an important phase in the design of a neural network.

There are different kinds and architectures of neural networks depending fundamentally on the manner how they learn. In this work, the multi-layer perception approach is used. Many researchers have focused on learning neural networks problems and several algorithms have been developed. Higher accuracy and faster convergence are a crucial issue in choosing the appropriate training algorithm.

The most popular technique for training multilayered is known as the back propagation (BP) algorithm [11]. The use of this algorithm is not always successful due to its sensitivity to learning parameters, initial state and perturbation [12]. There has been much work on the convergence of (BP) algorithm by using the gradient method [13]. Also, different versions of (BP) learning algorithms have been proposed, such as on-line algorithm for dealing with varying inputs [14] and the Levenberg-Marquardt-algorithm [15].

In this paper, training algorithm with one hidden layer have been presented. According to various studies [11], a single layer of hidden neurons is used to resolve most problems and is made the approximation of some functions thanks to its universal approximation capability.

Consider that the system model is describes by the following recurrent equation:

$$y(k+1) = F[y(k), y(k-1), \ldots\ldots, y(k-n+1), u(k), u(k-1), \ldots\ldots\ldots\ldots, u(k-m+1)] \qquad (1)$$

Where u and y represent respectively the input and the output vector of the network, m is the order of u, F is assumed unknown nonlinear function.

$[y(k), y(k-1), \ldots\ldots, y(k-n+1), u(k), u(k-1), \ldots\ldots\ldots\ldots, u(k-m+1)]$ represents the input vector.

The neural model can be used therefore to provide the estimated output $y_m(k+1)$ of the process at time k+1 based on the input values and output values at time k.

Each neuron is connected to all of the next layer by connections whose weights are arbitrary real numbers. A neuron in the hidden layer or in the output layer combines its inputs into single value, that which it transforms afterwards to produce the output This transformation is called the activation function.

For multi-layer networks, the sigmoid activation function is defined as: $f(x) = \frac{1}{1+e^{-x}}$.

2. Direct Neural Network « DNM »

A simple (DNM), with a single output may be described in Fig. 1.

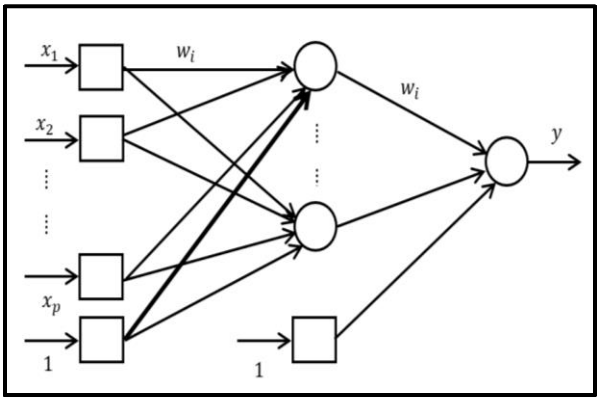

Fig 1: Feedforward Neural Network (FNN)

In this work we mean by the name the direct neural model the global neural model. To obtain the global model we use one neural network this is called the mono-network neural modeling.

2.1 Learning

Artificial neural networks (ANN) can be considered as a black box containing learned and memorized information. But at startup this box is empty, it contains neither information nor knowledge, learning is necessary. The learning process is a phase during which the behavior of the network is modified until the desired behavior is achieved. Learning of "ANN" consists of adapting its different parameters (weight) according to an iterative algorithm of adjustment or adaptation. The learning process takes into consideration all the examples provided at the entrance. Learning algorithms yield better results for multiple and varied input examples [16].

The learning procedure requires:

- A training set consisting of N examples, each one consisting of a vector applied to the input network and a vector of desired values of outputs. This training set must be rich

ISSN 1737-9288

enough that it covers the maximum possible area of operation for the network.

- Defining a cost function which measures the difference between network outputs and desired outputs present in the training set.
- An algorithm for minimizing the cost function.

$$E_r = \frac{1}{2}\left[y_p(k+1) - y_m(k+1)\right] \quad (2)$$

$y_p(k+1)$ and $y_m(k+1)$ respectively represent the output of the process and that of the model at k+1.

Most "neural network" learning algorithms are optimization algorithms: they seek to minimize, by nonlinear optimization methods, a cost function that is a measure of the gap between the network responses and the desired responses. This optimization is done iteratively, by modifying the weights as a function of the gradient of the cost function: the gradient is estimated by a method specific to neural networks, the so-called backpropagation method. The backpropagation algorithm is one of the most used supervised algorithms for learning multilayer perceptions (MLP). Generally, the learning is done over a relatively long period, and includes four stages of calculation:

Generally, learning requires a relatively long period, and includes four stages of calculation:

1. Initialization of synaptic weights of the network.
2. Presentation of the input vector and propagation of states.
3. Calculation of the error at the output of the network.
4. Calculation of the vector of correction.

For this type of network, we use the back-propagation algorithm as a learning algorithm.

2.2 The back propagation algorithm

In an "MLP", for each example presented to the network, an estimated output is calculated by propagating the calculation from one layer to another to the output layer, then an error will be computed and then backpropagated in the network to end. adjust each weight. The same procedure is repeated for all learning examples. This process is repeated until the outputs of the network are sufficiently close to the desired outputs.

Denote that $y(y_1, \ldots, y_m)$ is the vector of desired outputs and the sigmoid function f, its derivative is: $f(x) = f(x)\big(1 - f(x)\big)$.

The input and output of a neuron v are respectively denoted by I_v and O_v.

$$\begin{cases} I_v = \sum_j W_{jv} O_j \\ O_v = f(I_v) \end{cases} \quad (3)$$

The error corresponding of an example p is given by:

$$E_p(w) = \sum_{i=1}^{m} \frac{1}{2}(S_i^p - Y_i^p)^2 \quad (4)$$

The back-propagation algorithm is an approximation of the gradient method. It is defined by:

$$W_{uv}(t+1) = W_{uv}(t) - \varepsilon(t)\frac{\partial E^p(w)}{\partial W_{uv}} \quad (5)$$

t denotes the numbers of iterations, $\varepsilon(t)$ is the step of gradient. This equation requires the calculation of W_{uv}.

Suppose that $d_v = \frac{\partial E^p}{\partial I_v}$, so we have:

$$W_{uv}(t+1) = W_{uv}(t) - \varepsilon(t) d_v O_u \quad (6)$$

This learning algorithm depends on several factors [17]:

The initialization of the network parameters: the choice of the initial values of the weights and the iteration step must be done in a way that ensures a fast and stable convergence.

The complexity of the learning base and the order of presentation of the examples.

The structure of the "Neural Networks" considered must be chosen in an appropriate way, especially the number of neurons of the hidden layer which must be optimal.

The main purpose of learning is to make a network capable of generalizing. It is dangerous to continue indefinitely the learning phase without control [16]. The learning capacity of a neural network is so strong that after a certain number of iterations, the synaptic weights are able to predict almost the data without error. At this stage, the neural network is no longer mistaken in its predictions, but there is a great risk that these predictions are just as good as the data on which learning was based. This is called learning by heart or over-learning. This phenomenon is then prevented by adding a phase to the modeling procedure called the generalization or validation phase.

2.3 Generalization

The concept of generalization for a neural network is used to accurately measure model performance for a given problem once learning is complete. The generalization is manifested by an input basis, unknown to the network, given to the model to test its ability to generalize; if error is minimal, we say that the process has learned otherwise, so we speak of over learning. It represents one of the features of neural networks because they are able to generalize from a test.

This phase is influenced mainly by three factors: the number and quality of the learning examples, the complexity of the learning algorithm used and the size of the network. [11]

IV. SIMULATION RESULTS

In this section, we present the simulation results. The application of the modeling approach is showed. We use this modeling approach for the neural identification of Supply Chain Risk. The neural modeling was done through a neural network with one hidden layer and one output. The activation function used in the example is the sigmoid function and the network generates the output $y_m(k+1)$.

4.1 Experiment Data

The database used in experiment, including the risk assessment value and the risk level of the risk factors, can be collected by expert evaluation. The sample data as neurons in the input layer and the output layer is *listed in Table 1, where X1* is for information risk rating value, *X2* is for the level of financial risk, *X3* is the logistics risk rating value, *X4* is the time risk rating, *X5* is the value of organization risk rating.

4.2 BP neural network training

An artificial neural network models with BP training has been developed using MATLAB software for supply chain risk assessment, and the experiment dataset in Table 1 is used to train the network. The performance of the BP neural network training is indicated in Fig.2. Here the variable learning rate back propagation algorithm is used in the network training, which can adjust the learning rate in accordance with the error curve change.

The architecture of the neural network generating the database is illustrated in Fig. 2.

ISSN 1737-9288

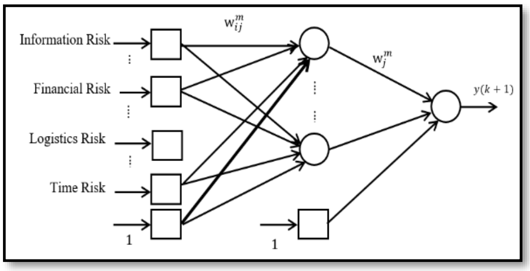

Fig 2: The architecture of the neural network generating the database.

TABLE I. DESCRIPTION AND MEASUREMENT UNIT OF THE METRICS OF THE PROPOSED PREDICTION SYSTEM

No	Parameters in the input layer					Sample
	X_1	X_2	X_3	X_4	X_5	results
1	0.4	0.3	0.4	0.5	0.3	0.2
2	0.3	0.5	0.8	0.4	0.2	0.41
3	0.4	0.3	0.2	0.3	0.4	0.36
4	0.6	0.5	0.2	0.7	0.5	0.59
5	0.3	0.2	0.4	0.5	0.4	0.35
6	0.2	0.2	0.2	0.3	0.4	0.28
7	0.5	0.4	0.3	0.6	0.4	0.46
8	0.8	0.7	0.6	0.8	0.7	0.77
9	0.2	0.3	0.3	0.4	0.2	0.26
10	0.4	0.5	0.6	0.4	0.35	0.48
11	0.7	0.6	0.8	0.5	0.73	0.74
12	0.4	0.3	0.2	0.3	0.5	0.38
13	0.87	0.7	0.63	0.6	0.8	0.80
14	0.3	0.2	0.4	0.4	0.5	0.34
15	0.6	0.5	0.7	0.5	0.6	0.62

4.3 BP neural network testing

After proper training, the network is simulated with other input parameters combinations and the network responses are compared with experimental response. The comparison of predicted values by using the developed model with experimental data was shown in Table II. It shows that predicted accuracy of the model is quite good and can be used in the supply chain risk assessment.

TABLE II. COMPARAISON BETWEEN SAMPLE RESULTS AND THE SIMULATION RESULTS

No	Sample results	Simulation results
11	0.74	0.7452
12	0.38	0.0390
13	0.80	0.801
14	0.34	0.342
15	0.62	0.625

A. Experiments and discussions

In this section, we present the simulation results. The sufficiency of the suggested algorithm is analyzed. We utilize this algorithm for the neural identification of a supply chain risk assessment.

The goal of our simulation is to find the adequate structure of the input-output neural model which describes the dynamics of the supply chain risk by using the approach presented in previous section.

The supply chain risk parameters used in simulations are illustrated in "Table. I". This model, assumed that:

- The number of hidden neurons equal to 6 and a step iteration ε equal to 0.2. These values are determined with tacking into account a good approximation and a good generalization.
- The learning algorithm used is the back-propagation algorithm.
- The learning time is equal to 192.792545 seconds.
- The value of error is equal to .00001.

The results of training phase are of the model is given by this figure:

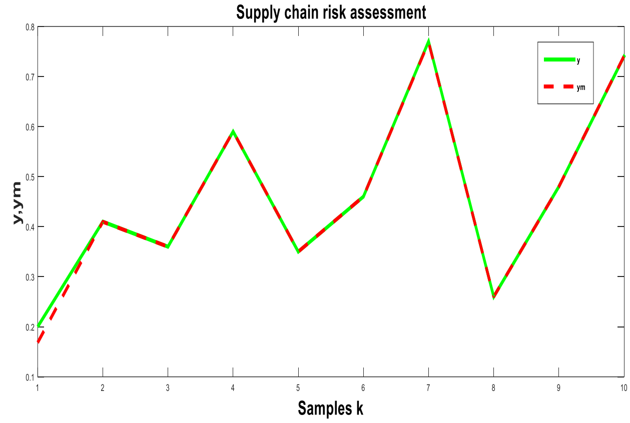

Fig 3: The evolution of the outputs

It can be shown that the suggested algorithm offers a good convergence characteristic in the training and validation phases.

"Fig. 3" illustrates the importance the PB algorithm. Indeed, we can see that the PB algorithm provides a good performance but with a slow convergence time. In the same way, the PB algorithm has a minimum convergence time with degradation of the performance of the model acquired. The resulting algorithm guarantees both rapid convergence and higher learning and generalization capabilities.

V. CONCLUSION

In this paper, an artificial neural network training algorithm has been designed and applied in Supply Chain Risk assessment. The proposed approach develops a BP algorithm based on artificial neural networks techniques to avoid subjectivity factors in the risk assessment process. The network simulation results shows that the model is accurate and practical.

An essential finding in this study is that the construction of the neural network is an empirical process and requires several trials to find the suitable parameters giving the best performances of the neural training. These parameters include weights initialization, number of hidden neurons, MSE (mean square error), numbers of iterations, etc. The learning pedagogy affects also the accuracy rate.

The limitation faced in this research is the collection of the database that's need to be larger as the accuracy of the ANN training increases by increasing the number of selected samples per training class. Moreover, the correctness of the sample become difficult to testified.

Various research directions and challenges could be considered for future research. Firstly, the approach proposed in this article can be applied to forecast supply chain performance in order to automate the whole logistics management processes.

Another direction of research concerns the application of other artificial neural approaches like fuzzy logic in supply chain

ISSN 1737-9288

risk estimation in order to compare the advantages and limitations of using each of them.

In additional, in this study we have consider only some risk variable. Future studies can take into account other risk variable and can also involve other external features that will lead to better risk assessment.

REFERENCES

[1] D. Simchi-Levi, P. Kaminsky, E. Simchi-Levi, Designing and Managing the Supply Chain, McGraw Hill, Boston, 2000.

[2] S. Tayur, R. Ganeshan, M. Magazine (Eds.), Quantitative Methods for Supply Chain Management, Kluwer's International Series, Boston, 2000.

[3] M.S. Sodhi, B.-G. Son, C.S. Tang, Researchers' perspectives on supply chain risk management, Prod. Oper. Manage. 21 (1) (2012) 1–13.

[4] G. Baryannis, S. Validi, S. Dani, G. Antoniou, Supply chain risk management and artificial intelligence: state of the art and future research directions, Int. J. Prod. Res. 57 (7) (2019) 2179–2202, http://dx.doi.org/10.1080/ 00207543.2018.1530476.

[5] Y. Fan, L. Heilig, S. Voss, Supply chain risk management in the era of big data, Lecture Notes in Comput. Sci. 9186 (2015) 283–294, http://dx.doi. org/10.1007/978-3-319-20886-2_27.

[6] S. Ye, Z. Xiao, G. Zhu, Identification of supply chain disruptions with economic performance of firms using multi-category support vector machines, Int. J. Prod. Res. 53 (10) (2015) 3086–3103, http://dx.doi.org/10. 1080/00207543.2014.974838.

[7] Kaili Zhou, Yaohong Kang, "The neural network Model and MATLAB Programming", 2005, pp. 69-82.

[8] Yan-ling Ni, Hai-yin Li, Xiang Yan, "Comparison of SC Risk Control with Enterprise Risk Control", Logistics Technology, Dec. 2004, pp. 40-42

[9] Dong-mei Zhao, Hai-feng Liu, Chen-guang Liu, "Risk Assessment of Information Security Based on BP Neural Network", Computer Engineering and Applications, vol 43. Jan. 2007, pp. 139-141.

[10] S, B., J, B., & Hassani. H, E. (2013). Causal Method and Time Series Forecasting model based on Artificial Neural Network. International Journal of Computer Applications, 75(7), 37–42.

[11] Narendra, K. S., & Parthasarathy, K. (1991). Gradient methods for the optimization of dynamical systems containing neural networks. IEEE Transactions on Neural Networks, 2(2), 252–262.

[12] Xinghuo Yu, Efe, M. O., & Kaynak, O. (2002). A general backpropagation algorithm for feedforward neural networks learning. IEEE Transactions on Neural Networks, 13(1), 251–254.

[13] Frasconi, P., Gori, M., Maggini, M., & Soda, G. (1996). Representation of finite state automata in Recurrent Radial Basis Function networks. Machine Learning, 23(1), 5–32.

[14] Zhao, X., Xie, J., & Leung, J. (2002). The impact of forecasting model selection on the value of information

[15] Hagan, M.T. and Menhaj, M.B. (1994) Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks, 5(6), 989-993.

[16] Solla S.A., « Learning and Generalization in Layered Neural Networks: the Contiguity Problem» In L. Personnas and G. Drefus (eds), Neural Networks: from Models and Applications. Paris. I.D.S.E.T,1989.

[17] Rivals I. Modélisation et commande de processus par réseaux de neurones; application au pilotage d'un véhicule autonome, Thèse de Doctorat de l'Université Paris 6 (1995).

[18] Silva, N., Ferreira, L. M. D. F., Silva, C., Magalhães, V., & Neto, P. (2017). Improving Supply Chain Visibility With Artificial Neural Networks. Procedia Manufacturing, 11, 2083–2090. doi:10.1016/j.promfg.2017.07.329

[19] Koo, K., Govindarasu, M., & Tian, J. (2019). Event prediction algorithm using neural networks for the power management system of electric vehicles. Applied Soft Computing, 105709. doi:10.1016/j.asoc.2019.105709

ISSN 1737-9288