

Review for Optimisation of Neural Networks with Genetic Algorithms and Design of Experiments in Stock Market Prediction

Yunus Emre Midilli¹, Sergei Parshutin²
^{1,2}Riga Technical University, Riga, Latvia

Abstract – Neural networks are commonly used methods in stock market predictions. From the earlier studies in the literature, the requirement of optimising neural networks has been emphasised to increase the profitability, accuracy and performance of neural networks in exchange rate prediction. The paper proposes a literature review of two techniques to optimise neural networks in stock market predictions: genetic algorithms and design of experiments. These two methods have been discussed in three approaches to optimise the following aspects of neural networks: variables, input layer and hyper-parameters.

Keywords – Design of experiment, genetic algorithms, neural networks, stock market.

I. INTRODUCTION

A neural network (NN) is a massively parallel-distributed processor made up of simple processing units that have a natural propensity for storing experiential knowledge and making it available for use [1]. NNs are utilised to model the relationships between outputs and inputs of the systems for the purpose of classification, regression, clustering, simulation, pattern recognition etc. They have been used for decades in the areas such as chemistry, energy, finance, operations management, etc. Among these areas, NNs have become a popular tool for financial decision making due to the following reasons [2]:

- There is no need for data distribution assumptions in NN applications.
- NN is an incremental mining technique that permits new data to be submitted to a trained network in order to update the previous network [3].
- NNs are able to decode nonlinear time series data that adequately describe the characteristics of the stock markets [4].
- NNs learn information by training rather than by programming [5].
- NN is data driven and is not restrictive by historical data generation process [6].

One of the earliest studies [7] was performed on the evaluation of strength of NN in the financial time series analysis. IBM common stock daily return was predicted to decode nonlinear regularities in asset price movements. In the study, it was found out that expectation of opening keys of complex financial markets with simple NNs would probably fail. In this context, researcher in [7] considered three-layer perceptron as a simple NN. Therefore, it was suggested to

improve NN structures such as predefined parameters that were used during the training process. During the same period with [7], the ratings of corporate bonds prediction were considered as a generalization problem and NNs were used to solve this problem. In the study [8], it was verified that NNs performed much better than mathematical models.

In the literature, it was verified that NNs gave successful results in exchange rate prediction problems; however, it was emphasised that traditional NNs might sometimes provide controversial results. In the research [9], NNs were used to predict exchange rates of US Dollar (USD) and five other currencies and stated that multi-layer perceptron (MLP), one of the most commonly used NN type, models gave significantly successful results in most of the financial date series analysis but in some cases statistical prediction methodologies provided more accurate results than NN. In case of unsuccessful results, analysts were suggested to check additional factors affecting the performance. Strength of prediction performance of NN was proven even in extreme situations such as financial crisis, political situations etc. Borsa Istanbul-100 (BIST-100) index was predicted with MLP based on macroeconomic indicators such as gold price, oil price, interest rate, money supply and trading volume during the global financial crisis in the time interval between July 2007 and December 2009 [10].

There are many other statistical and mathematical models to forecast exchange rates; however, it has been proven that NNs generally provide better results than statistical and mathematical models. Authors of [8] observed that accuracy of NN was considerably higher than regression analysis, e.g., the success rate during the testing phase for two-layered MLP was 88.3 % as compared to 64.7 % for the regression models. Authors of [11] compared MLP with traditional econometric techniques, including linear and nonlinear analysis, in order to predict net asset values of mutual funds. It was verified that NNs significantly outperformed regression models. Researchers of [12] used probabilistic NN to predict the Taiwan Stock Exchange index return based on some set of buy or hold strategies. Probabilistic NN based performance values were compared with mathematical models and random walk models. They concluded that probabilistic NN based investment strategies obtained higher returns than other traditional strategies. Author of [13] proposed a hybrid forecasting model combining MLP and GA in order to predict Indian Rupee vs. US Dollar (INR/USD), Euro vs. Canadian Dollar (EUR/CAD)

and Australian Dollar and British Pound (AUD/GBP) in the time interval between 2011 and 2014. The proposed model was compared with a linear-trend model, ARMA model and simple MLP. Findings demonstrated that linear-trend models failed with negative Pearson coefficient values. In all exchanges except INR/USD, ARMA model's Pearson coefficient was around 73 %, while simple MLP was around of 95 %. In INR/USD exchange, both MLP and ARMA demonstrated Pearson coefficient around 90 %. INR/USD was predicted successfully by both NN and statistical models; however, statistical models failed to predict other exchanges. The proposed hybrid forecasting model's Pearson coefficient was approximately 5 % higher than MLP model for the exchange rates INR/USD and AUD/GBP.

Studies in literature reached a very important level to predict even under very complex market conditions. However, there are many parameters affecting performance of NN. Optimisation of these parameters is required to increase profitability. There are still concerns and challenges to optimise NNs in the financial data analysis. NN design is done through calibrating numerous parameters; however, finding the correct combination of parameter values is challenging given a specific data set [14], [15]. In general, the trial and error process is what most practitioners use for optimisation [16]. Authors of [17] determined a number of hidden neurons and learning rate with a trial and error method to create a predictive model for exchange rate of Taiwan Dollar and US Dollar in the time interval between 1988 and 2007. In the study [6], a number of hidden neurons and a number of hidden layers were determined with a trial and error method. In the study, once predefined error value was achieved with a trial, the trial was accepted as the best combination to predict. Designed NN was used to predict exchange rates of Japanese Yen (JPY), British Pound (GBP), Euro (EUR), Swiss Franc (CHF) against US Dollar (USD) in 2008. Authors of [18] used a trial and error method to identify design parameters of MLP in order to predict NASDAQ stock exchange index values. In the study, a number of hidden layers, number of hidden amounts, training function and transfer function were determined via trying some set of values and applying the combination, which had the best coefficient of determination value (R-Squared). In order to determine the best R-Squared value, average of R-Squared values of training, testing and validation processes were used.

In the study [19], the importance of optimisation of two critical NN design issues faced by financial analysts was emphasised: the selection of appropriate variables and capturing a sufficient quantity of training examples. In the study, it was observed that optimisation of these two parameters caused significant improvement of learning accuracy of NN. In [10], the learning rate and momentum coefficient were determined via utilising a response surface methodology (RSM), i.e., one of the most commonly used design of experiment (DOE) techniques. NN modelling is commonly used with backpropagation (BP) algorithm, but the BP algorithm is often easy to fall into a local optimum point, and the convergence rate is slow to highlight weaknesses, thus affecting the establishment and reliability of the model [20].

II. OPTIMISATION METHODS

There are numerous methods and algorithms used to optimise NNs. Genetic algorithms (GA) and DOE are two of the most common techniques in the literature to optimise NNs in exchange rate predictions.

GA is a generic name of the algorithms that are based on natural selection and imitating genetic process in the optimisation problems [21]. GA is a method that uses natural selection and evolutionary thought to find the best in high dimensional space. Some evolutionary optimisation techniques stack on local optimum, but GA has the ability of global optimisation [20]. Classic GA is a set of processes of coding, selection, cross-over and mutation. Coding is evolving decision variables into the form of chromosomes. Chromosomes are a set of genes. Each gene represents a decision variable of the problem. At the initial step of GA, decision variables of problem are converted into the form of GA via coding. In classic GA, generally binary coding is used. However, in case decision variables are real numbers such as the learning rate of NN, connection weight of NN etc., it is recommended to use real-valued coding operators. In the coding step of GA, GA generates some pre-defined amount of candidate solutions (chromosomes). Once all candidate solutions are generated, the set of chromosomes is called population. After population is created, fitness functions of each chromosome are calculated. Fitness functions are the performance metrics and objective functions of chromosomes according to a solution domain. In NN optimisations, generally statistical performance metrics, such as MSE, MAD, Pearson correlation, correlation coefficient, are used as fitness functions. Once fitness functions of each chromosome are calculated, the best fitted chromosomes are selected in selection steps. Selected chromosomes are performed by cross-over and create child chromosomes for next generations based on a pre-defined cross-over rate. After the cross-over step, child chromosomes are exposed to mutation based on the mutation rate that is significantly lower than the cross-over rate. GAs have also some design parameters like population size, type of coding, cross-over rate, mutation rate etc. After completion of selection, cross-over and mutation steps, these cycles are repeated unless stop criteria are provisioned. Stop criteria can be an expected average fitness rate, minimum difference between chromosomes within the population, epoch size etc.

Design of experiments or experimental design is a test or a set of tests in order to detect effects of factors on outputs via changing inputs in a systematic way. DOE is a statistical technique that is used for process optimisation problems. Traditional trial and error method has a drawback of detecting only one parameter effect on output values but it is possible to identify not only the main effects but also interaction and higher order effects of the factors on response. In the research [22], DOE is described as a set of processes of: (i) problem description, (ii) determination of responses and factors, (iii) determination of factor levels, (iv) choice of type of DOE, (v) performing the experiment, (vi) data analysis. After the detailed description of problem domain, response values and factors are determined. Response is a performance metric or result of the

process. In NN optimisations, generally accuracy and performance metrics are used as response variables. After factors and responses are determined, levels of factors are defined. Factor levels in NN optimisation are generally defined based on learning from literature or preliminary experiments. After the definition of factor levels, a type of DOE is determined. The most common DOE types are full factorial design, central composite design, Box-Behnken design, fractional factorial design, Taguchi design etc. Designed experiments are executed run by run, and response values are noted for each run. Statistical analysis of variance (ANOVA) is performed to identify significance of factor impacts on responses. According to ANOVA analysis, the main and interaction factors are found and the first-order degree mathematical model can be established. After the establishment of the mathematical model, optimum values of the mathematical model are found.

III. APPROACHES OF NN OPTIMISATION

Studies related to optimisation of NNs are categorised into three headings: optimisation of NN variables, optimisation of NN inputs and optimisation of NN hyper-parameters.

A. Optimisation of NN Variables

NN variables are connection weights that store and transfer information among neurons. Optimisation of NN variables deals with learning algorithms of NN. The most common form of NN is MLP that usually is trained by backpropagation of errors. This can require slow processing, extensive exploration of NN topology and training controls before the implementation of real-life datasets [23]. One of the first studies performed on the strength of NN in the financial time series analysis [7] found out that even though final weight values were the best values of range, which was determined during the used iterations, it was a fact that the existing NN methodologies were using local optimum search and included high risk of stacking on local optimum. Therefore, rather than training with a BP algorithm, it was suggested to use global optimisation methods such as GA or simulated annealing. Generally, evolutionary algorithms such as GA, simulated annealing etc. are used to train NN variables. DOE is not preferred for optimising NN variables.

A number of hidden layers, one of the key design NN parameters, are closely related and positively correlated with the final accuracy rate but negatively correlated with training time, and it may lead to an overfitting issue [24]. However, in their study, this design parameter was defined without any justification and they set other design parameters such as a number of hidden layers, learning rate, iteration number, training error based on historical suggestions. Instead, they focused on optimisation of NN connection weights. They used GA to determine optimum connection weights in order to analyse financial transaction patterns. They trained MLP with a traditional BP algorithm and GA. They observed that while a traditional MLP had 88 % accuracy, optimised MLP had 90.5% accuracy and their learning time was 140 seconds and 125 seconds, respectively. Similarly, [25] proposed an optimisation model for ANN using GA in order to predict the Korea

Composite Stock Price Index (KOSPO). The proposed model optimised four major factors: connection weights, the number of neurons in a hidden layer, feature subset selection and feature transformation. It was verified that GA optimised NNs gave more accurate performance than simple NNs.

B. Optimisation of NN Input Layer

Input layer is a set of an independent dataset that NN maps onto an output layer, which is a dependent dataset. In the context of prediction of exchange rates, generally macroeconomic indicators, historical data, technical indicators or seasonal affects are considered as inputs of the NNs. Independent variables may change according to the predicted market, season or political situations. That is why apart from economic rules, a consistent set of input parameters is not defined in exchange rate prediction with NN. Optimisation of NN inputs deals with the selection of an independent dataset based on the market conditions.

In the research [26], a novel approach is brought to the literature by using GA to optimise technical indicator levels in order to take position of buy, sell or hold. The authors chose Dow 30 index's daily stock prices in the time interval between 1996 and 2016 and calculated technical analysis indicators called relative strength index (RSI) and simple moving average (SMA). RSI values were calculated for the interval of 1 to 20 days and SMA values were calculated for the period of 50 to 200 days. Later, GA was applied to find the best RSI values, which determined buy and sell positions during down and up trends. GA chromosome contained the following genes for both down and up trends: RSI buy value, RSI buy interval, RSI sell value and RSI sell interval. After the determination of optimum RSI values, MLP was trained and the defined RSI values were applied to MLP. MLP output layer contained three neurons: buy, sell or hold. According to the predicted results, success rates of each stock were calculated. The results showed that such a trading system had comparable or better results compared with Buy & Hold and other trading systems for a wide range of stocks even for relatively longer periods. The authors suggested involving more technical indicators as future research.

Author [27] aimed at optimising deep learning MLP with DOE in the time series analysis. Time series data were generated based on a nonlinear series model, called exponential smoothing transition auto-regressive (ESTAR) model, which fit with the financial time series. Number of inputs, number of hidden layers, number of hidden nodes, activation function and length of the series were selected as factors of DOE. It was observed that an increase of input did not significantly improve the forecast performance. It was pointed out that input type choice was much more important than a number of inputs from the perspective of NN performance. In the study, optimisation of deep learning was recommended.

C. Optimisation of NN Hyper-Parameters

Hyper-parameters are a set of the parameters, which determine the network structure, design and the variables. Hyper-parameters are all the parameters that are adjusted before starting to train the NN. Learning rate, momentum rate, number

of hidden layers, number of hidden units, percentage of training dataset, initial weights and biases etc. are considered hyper-parameters since they are set before starting to train the NN. In this context, many studies have been performed to optimise these parameters; however, there is no exactly defined value of these variables. This is the demonstration of a fact that these parameters should be optimised in each learning process.

MLP is one of the most frequently used and optimised NN types in the literature, but there are also many other NN types, in which optimisation techniques are applied. Utilising GA to optimise hyper-parameters of fuzzy neural network (FNN) in order to obtain the best performance for forecasting Santa Fe time series in the time interval between 1990 and 1991 was mentioned in [28], and it was pointed out that GA-FNN model had $1.0297e^{-6}$ RMSE value. The authors stated that this value had good enough impact for their problem on investors' losses and profits. They suggested extending the work in order to examine the applicability of evolving FNN in an inefficient economic system. Author of [29] worked on the utilisation of GA for optimisation of the following predictive models: MLP, Jordan and Elman Networks (J/E), time-lag recurrent network (TLRN), recurrent neural network (RNN), modular network (MN) and support vector machines (SVM). Each NN type is trained in order to predict daily closing prices of Moscow Times share index in the time interval between 1997 and 2004. GA optimised the following features of the NNs: network topology, performance surface, learning rules, number of hidden neurons, memory gaps, learning rate, and momentum rate. In the study, learning time of optimised NNs was approximately 120 times slower than that of basic NNs; however, improvement on accuracy of NNs increased twice with the proposed method. According to simulation results of the proposed method, it was observed that profitability increased significantly compared to traditional approaches. Researchers of [30] proposed an intelligent trading system based on MLP that was optimised with GA. The proposed trading system was applied to the Hong Kong Hang Seng stock index in the time interval between 1997 and 2007. A number of neurons were determined by a trial and error method. GA was used to optimise initial weights of MLP. After setting optimum initial weights, an optimum NN structure was implemented to the intelligent trading system and results showed that the optimised NN was 7 times more profitable and accurate on trading. Correlation coefficient of the optimised model was 96.34 %. Authors of [31] emphasised radial basis function neural network (RBFNN) as a more effective NN type, which had fast convergence, high precision and strong generalization capability. They used GA to optimise RBFNN centre and width of Gaussian basic function in each hidden node. In the study, a number of hidden neurons and generation number were determined via trial and error. Next day values of the Shanghai Composite Index were predicted based on historical values of the Shanghai Composite Index, Dow Jones Index, Nikkei Index and Hang Seng Index in the time interval between 2008 and 2009. While utilising GA, RBFNN had 3.78 % MSE, basic RBF had 5.80 % MSE. This comparative result was considered successful; however, the authors suggested paying more attention to other influential factors such

as oil price, popularity index and company profit/loss etc. MLP was considered to predict EUR/USD exchange rate in [32]. A Pearson correlation coefficient matrix was established and threshold to accept as an input parameter of MLP was defined as 80 %. The following seven parameters were selected: NASDAQ Index, daily Euro and USD in New Zealand, gold spot prices in the USA, average returns of 5-year USA government bonds, average returns of 5-year Eurozone, crude oil price, exchange rate of EUR/USD of the previous day compared to the day of the output. In the study, GA was used to optimise a number of hidden layers, number of hidden neurons and activation functions of all neurons per layer. The optimised NN revealed 94 % coefficient of determination. In the study [33], it was targeted to implement GA and particle swarm optimisation (PSO) to optimise the ensemble NN architectures in order to predict the Taiwan Stock Exchange Index (TWSE) in the time interval between 2011 and 2014. In order to achieve better prediction accuracy, PSO optimised ensemble NN, where the optimised parameters were the following: the number of modules, number of layers and number of neurons. The aggregation of the responses of the optimised ensemble NN was performed with type-1 and type-2 fuzzy systems. The best prediction error value for GA was found to be 0.0011421 and PSO was 0.0013066. It was concluded that GA gave better performance for solving optimisation and research problems for ensemble NN in stock exchange prediction. In the study [34], initial weights and bias of MLP were optimised in order to predict Nikkei 225 index in the time interval between 2007 and 2013. Hit ratio, average of prediction results of each day was set as performance metric. In the study, the most suggested technical indicators were used as inputs of MLP and the all other hyper-parameters such as a number of hidden neurons, number of iterations, learning rate and momentum rate were determined based on trial and error and historical learning. It was reported that while basic MLP had 60.87 % hit ratio, optimised MLP had 81.27 % hit ratio. Additionally, the study was compared with the previous studies performed in the same problem domain and it was concluded that there was a significant improvement in the proposed methodology. It was suggested that prediction performance could be improved via optimising other design parameters, such as a number of hidden neurons, learning rate etc. Researchers of [20] determined an optimum node transfer function and connection weights; however, a number of hidden layers were determined using a trial and error method. The proposed model was applied to the Shanghai Index data to make simulation and provide the corresponding analysis. Results showed that simple NN had slow convergence and average error values (around of 1.5 %). The two main reasons were indicated as root causes of this failure: one was due to the fact that the stock market was the place of a mental game, especially in short-term operation, and the other was due to the presence of certain difficulty setting parameters of the network. Performance metric of the proposed method was calculated significantly higher than that of simple NNs (1.1 % average error). In addition, the authors applied the proposed method to a different market that had a less market value and volume. They verified that the greater the market

value of the stock market is, the more stable operation is; the stronger the ability to resist interference of external factors is, the higher accuracy of prediction can be achieved.

One of the most comprehensive studies that included numerous different factors affecting performance of MLP was performed in [35]. In the study, the following parameters were considered as factors of DOE: a number of hidden layers, number of hidden neurons, regression output function, input mode, predicted number of steps, steps used to predict, phase 1 training algorithm, phase 2 training algorithm, epoch size, learning rate, initialisation function, stop criteria, momentum rate, minimum improvement error for a number of epochs, prune units, prune input variables, weight decay regularisation for phase 1, weight decay regularisation for phase 2, backpropagation tuning, quick propagation unit, delta-bar-delta tuning, sample size and sampling method. As response variable, a mean absolute percentage error (MAPE) was selected. Since there were numerous factors considered, establishment and running of a full factorial design of experiments would be a time-consuming activity. In the study, the Taguchi approach was chosen as a screening experiment method due to the following drawbacks in the screening experiment process: (i) fractional factorial designs had drawbacks when a number of factors exceeded twenty, (ii) group screening design [36] had drawbacks related to group information, (iii) sequential bifurcation [37] with a drawback of being limited to quantitative variables, (iv) iterated fractional designs [38] with a drawback of not being able to evaluate interaction, (v) Trocine screening procedure [39] with a drawback of only being able to evaluate three or four critical factors. In the study, eight time series models (STAR1, STAR2, SAR etc.), which were the most frequently seen in the stock exchange analysis, were executed according to the proposed methodology. For each nonlinear model, the main factors and interactions were identified and optimum values were determined. Number of hidden neurons, learning rate, momentum rate and epoch size were defined as factors of DOE in the study [40]. Taguchi design was established. Hit ratio and Pearson correlation were selected as a response variable of the design. The experiment was run for the design of MLP that was utilised to predict the Taiwan Stock Exchange Index prices based on the defined technical indicators. After the execution of DOE was completed, the ANOVA analysis was performed to define the factors that had a significant impact on the responses. According to the ANOVA results, all factors, except the momentum rate, had the main effect on the Pearson correlation value. All factors had the main effect on a hit ratio response. According to the main and interaction effect model, the optimised NN hyper-parameters were used to predict the market data. It was concluded that while traditional models had 53–64 % accuracy, the proposed model had 84 % accuracy. The study [15] identified the design parameters that might impact NN built for forecasting future moves of the UAE MSCI Index in the time interval between 2002 and 2012. Two-level full factorial DOE was established for the design parameters: NN type, number of neurons in hidden layers, output layer transfer function and learning rate. MLP and RNN types were defined

as NN type. Hyperbolic tangent sigmoid and pure linear functions were determined as an output layer transfer function. Numerical parameters (number of hidden neurons and learning rate) used suggestions from literature. According to 2x2x2x2 ANOVA test with 95 % confidence interval, a number of hidden neurons, output layer transfer function and learning were observed as the main effect and significantly decreased MSE. Furthermore, there was interaction between NN type and a number of hidden neurons. When NN type was MLP, the increasing number of hidden neurons caused a significant decrease in MSE, but when NN type was RNN, the number of hidden layers did not have a significant impact on MSE. In addition, interaction between an output layer transfer function and a number of hidden layers was also observed. It was concluded that changing the transfer function affected MSE significantly, while a number of neurons were at a lower level. The results of the study showed that it was possible to find the most significant NN parameters using factorial designs. The proposed model in the study led to a reduction in costs and trial error during building NNs. Researchers in [41] used Taiwan 50 index companies' some financial ratios (e.g., fixed assets to long-term debt, capital-to-total assets etc.) as input of MLP to predict the Taiwan 50 stock index prices in the time interval between 2007 and 2008. In the beginning, there were 54 financial ratios for each company but a preliminary stepwise regression analysis was performed before feeding to MLP. 16 financial ratios were determined as major inputs of index price. Number of hidden neurons, learning rate, momentum rate and epoch size were determined as DOE factors. Levels of the factors were determined based on pre-experiments and learning from the literature. Upper and lower levels of a number of hidden layers were determined based on a rule of thumb [42], [43], which set an upper level not more than twice size of an input layer and not less than an input layer. Upper and lower levels of the learning rate were determined between 0.3 and 0.9 based on suggestions from the literature [44]. Momentum rate levels were set to between 0.5 and 0.9 [40]. The authors applied the fractional factorial experiment. Each run was replicated three times to lower the variation of the experiment and increase the accuracy of the experiment. Six central points were added to the design. Pearson correlation coefficient and RMSE were selected as performance indicators of MLP and response values of DOE. As a result of ANOVA output, it was found out that a number of hidden neurons and learning rate had significant interaction. Optimum combination of four factors was generally lower limits of the factor intervals. This pointed out that the selected factor levels should be extended and curvature should be caught; however, even the curvature was not obvious, the existing DOE model estimated Pearson correlation value as 91.66 %. Taiwan stock exchange financial data were forecasted with six different predictive models: linear regression, least median squares, MLP, RBFN, SVR and DOE based MLP. It was verified that NN techniques were significantly more accurate than statistical techniques. According to comparison between NN techniques, DOE based MLP had approximately 5 % more Pearson correlation value than a basic MLP method. The proposed methodology and other predictive models were

also applied to 40 USA banks from New York Stock Exchange, and a significant difference between methods was observed in the same manner. Istanbul Stock Exchange Index (BIST-100) next day and next week prices were predicted in [10] based on commodities and macroeconomic indicators, such as gold prices, oil prices, interest rates etc. in the time interval between 2007 and 2009 when there was the period of global financial crisis. MLP was used as a prediction algorithm and learning rate and momentum rate parameters were considered to be factors of DOE. The authors determined factor levels according to preliminary trials. While learning rate factor levels were 0.009–0.010 respectively, momentum rate factor levels were defined as 0.2–0.9. MSE was selected as a response variable of the design. RSM was used to establish a mathematical model and according to derivative of the model, optimised values were determined. According to the results, optimum learning rate was 0.0417 and momentum rate was 0.2310. These optimum values proved the fact that there was curvature in the established model. After the optimum design was applied to MLP, it was verified that the usage of DOE increased the performance of MLP. Since prediction period was an extreme situation for global economy, this study proved that the proposed model could work even in extreme chaotic situations.

IV. CONCLUSION

In this review, GA and DOE based optimisation of NNs in exchange rate prediction have been examined. It has been found out that the requirement of optimising NNs in exchange rate prediction occurred even in the first trials in the literature. It has been pointed out that simple NNs are not good enough for profitability and accuracy even though they are successful in some cases. There are various types of NNs used in exchange rate prediction. GA and DOE are also used to optimise NNs; however, the main focus is placed on MLP. It is emphasised that deep learning gives better performance in prediction problems; in this context there are not so many studies performed to optimise deep learning parameters in exchange rate prediction. Another finding is that generally researchers focus on optimisation of connection weights or NN hyper-parameters. There are not so many studies to determine an optimum independent dataset (input layer) of NN specific to exchange rate prediction. It has been observed that it is not very common to use one exchange rate as input of technical indicators or prices of another exchange rate. On the other hand, there are studies concerning profitability of NN, but profitability is generally not an output of the NN but it is calculated after processing the outputs of NNs. Besides, almost all studies have verified the strength of NNs in exchange rate prediction. Studies have proven that the determination of parameters of NN increases the efficiency of profitability and accuracy. In the optimisation of NN, generally statistical accuracy performance metrics are used. Pearson coefficient, correlation coefficient, mean square error (MSE), and mean absolute deviation (MAD) are the most commonly used performance metrics. Apart from statistical performance metrics, in exchange rate prediction, profitability or financial indicators are used as prediction performance.

To sum up, there are many studies performed to optimise NN parameters, but still some of the major parameters such as learning rate, number of hidden layers etc. are determined with a trial and error method. Awareness of optimisation requirement of NN should be increased among the practitioners. In addition, it has been observed that there is not a single comprehensive software system developed to optimise NNs in exchange rate prediction. Implementation of optimisation tools to NN software in exchange rate prediction will increase awareness and result in detailed studies in the future.

REFERENCES

- [1] S. Haykin, *Neural Networks and Learning Machines*. New Jersey: Pearson Education Inc., 2009.
- [2] M. Jasemi, A. M. Kimiagari and A. Memariani, "A modern neural network model to do stock market timing on the basis of the ancient investment technique of Japanese Candlestick," *Expert Systems with Applications*, vol. 38, no. 4, pp. 3884–3890, 2011. <https://doi.org/10.1016/j.eswa.2010.09.049>
- [3] M. Lam, "Neural Network techniques for financial performance prediction," *Decision Support Systems*, vol. 37, no. 4, pp. 567–581, 2004. [https://doi.org/10.1016/S0167-9236\(03\)00088-5](https://doi.org/10.1016/S0167-9236(03)00088-5)
- [4] J. Yao, C. Tan and H. Poh, "Neural networks for technical analysis: A study on KLCL," *International Journal of Theoretical and Applied Finance*, vol. 2, no. 2, pp. 221–241, 1999. <https://doi.org/10.1142/S0219024999000145>
- [5] B. P. Nasutin and A. Agah, "Currency exchange rate forecasting with neural networks," *Journal of Intelligent Systems*, vol. 10, no. 3, pp. 219–253, 2000. <https://doi.org/10.1515/JISYS.2000.10.3.219>
- [6] A. Emam and H. Min, "The Artificial Neural Network for Forecasting Foreign Exchange Rates," *International Journal of Services and Operations Management*, vol. 5, no. 6, pp. 740–757, 2009. <https://doi.org/10.1504/IJSOM.2009.026772>
- [7] H. White, "Economic Prediction Using Neural Networks: The Case of IBM Daily Stock Returns," in *Second Annual IEEE Conference on Neural Networks*, San Diego, CA, USA, 1988. <https://doi.org/10.1109/ICNN.1988.23959>
- [8] S. Dutta and S. Shekhar, "Bond Rating: A Non-Conservative Application of Neural Networks," in *Proceedings of the IEEE International Conference on Neural*, San Diego, CA, USA, 1988. <https://doi.org/10.1109/ICNN.1988.23958>
- [9] J. Yao and C. L. Tan, "A case study on using neural networks to perform technical forecasting of forex," *Neurocomputing*, vol. 34, no. 1–4, pp. 79–98, 2000. [https://doi.org/10.1016/S0925-2312\(00\)00300-3](https://doi.org/10.1016/S0925-2312(00)00300-3)
- [10] S. Sakarya, M. Yavuz, A. D. Karaoglan and N. Ozdemir, "Stock Market Index Prediction with Neural Network During Financial Crises: A Review on BIST-100," *Financial Risk and Management Reviews*, vol. 1, no. 2, pp. 53–67, 2015. <https://doi.org/10.18488/journal.89/2015.1.2/89.2.53.67>
- [11] W.-C. Chiang, T. Urban and G. Baldrige, "A Neural Network Approach to Mutual Fund Net Asset Value Forecasting," *Omega*, vol. 24, no. 2, pp. 205–215, April 1996. [https://doi.org/10.1016/0305-0483\(95\)00059-3](https://doi.org/10.1016/0305-0483(95)00059-3)
- [12] M. T. Leung, A.-S. Chen and H. Daouk, "Application of Neural Networks to Emerging Financial Market: Forecasting and Trading the Taiwan Stock Index," *Computers and Operations Research*, vol. 30, no. 6, pp. 901–923, 2003. [https://doi.org/10.1016/S0305-0548\(02\)00037-0](https://doi.org/10.1016/S0305-0548(02)00037-0)
- [13] M. Rather, "Computational intelligence based hybrid approach for forecasting currency exchange rate," in *2nd International Conference on Recent Trends in Information Systems*, Kolkata, 2015. <https://doi.org/10.1109/ReTIS.2015.7232846>
- [14] M. A. Ruggiero, "Neural Networks: Tahiti or Bust," *Artificial Intelligence in Finance*, vol. 2, pp. 15–20, 1995.
- [15] Lasfer, H. El-Baz and I. Zuolkernan, "Neural Network Design Parameters for Forecasting Financial Time Series," in *5th International Conference on Modeling, Simulation and Applied Optimization*, IEEE, 2013. <https://doi.org/10.1109/ICMSAO.2013.6552553>
- [16] G. Zhang and B. Patuwo, "Forecasting with Artificial Neural Networks: The State of the Art," *International Journal of Forecast*, vol. 14, no. 1, pp. 35–62, 1998. [https://doi.org/10.1016/S0169-2070\(97\)00044-7](https://doi.org/10.1016/S0169-2070(97)00044-7)

- [17] A.-P. Chen, Y.-C. Hsu and K.-F. Hu, "A Hybrid Forecasting Model for Foreign Exchange Rate Based on a Multi-neural Network," in *Fourth International Conference on Natural Computation*, Jinan, China, 2008. <https://doi.org/10.1109/ICNC.2008.298>
- [18] H. Moghaddam, M. H. Moghaddam and M. Esfandyari, "Stock Market Index Prediction Using Artificial Neural Network," *Journal of Economics, Finance and Administrative Science*, vol. 21, no. 41, pp. 89–93, 2016. <https://doi.org/10.1016/j.jefas.2016.07.002>
- [19] S. Walczak, "An Empirical Analysis of Data Requirements for Financial Forecasting with Neural Networks," *Journal of Management Information Systems*, vol. 17, no. 4, pp. 203–222, 2001. <https://doi.org/10.1080/07421222.2001.11045659>
- [20] Luo, Y. Chen and W. Jiang, "Stock Market Forecasting Algorithm Based on Improved Neural Network," in *Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA)*, Macau, China, 2016. <https://doi.org/10.1109/ICMTMA.2016.154>
- [21] M. H. Satman, *Genetik algoritmalar*, İstanbul: Türkmen Kitabevi, 2016.
- [22] C. Montgomery, *Statistical quality control: A modern introduction*. New Jersey: John Wiley & Sons Singapore Pte. Ltd., 2013.
- [23] W. M. Jenkins, "Neural Network Weight Training by Mutation," *Computers & Structures*, vol. 84, no. 31–32, pp. 2107–2112, 2006. <https://doi.org/10.1016/j.compstruc.2006.08.066>
- [24] T. Jun and H. Lei, "Genetic Optimization of BP Neural Network in the Application of Suspicious Financial Transactions Pattern Recognition," in *International Conference on Management of e-Commerce and e-Government*, Beijing, 2012.
- [25] K.-j. Kim and H. Ahn, "Simultaneous optimization of artificial neural networks for financial forecasting," *Applied Intelligence*, vol. 36, no. 4, pp. 887–898, June 2012. <https://doi.org/10.1007/s10489-011-0303-2>
- [26] O. B. Sezer, M. Ozbayoglu and E. Dogdu, "A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters," *Procedia Computer Science*, vol. 114, pp. 473–480, 2017. <https://doi.org/10.1016/j.procs.2017.09.031>
- [27] Suhartono and N. Suhermi, "Design of Experiments to Optimize the Architecture of Deep Learning for Nonlinear Time Series Forecastin," *Procedia Computer Science*, vol. 144, pp. 269–276, 2018. <https://doi.org/10.1016/j.procs.2018.10.528>
- [28] Muhammad and G. King, "Foreign Exchange Market Forecasting Using Evolutionary Fuzzy Networks," in *IEEE/IAFE 1997 Computational Intelligence for Financial Engineering (CIFEr)*, New York City, NY, USA, 1997.
- [29] S. Hayward, "Setting up Performance Surface of an Artificial Neural Network With Genetic Algorithm Optimization: in Search of an Accurate and Profitable Prediction for Stock Trading," in *Congress on Evolutionary Computation*, Portland, OR, USA, 2004.
- [30] M. Zhu and L. Wang, "Intelligent Trading Using Support Vector Regression and Multilayer Perceptrons Optimized with Genetic Algorithms," in *The 2010 International Joint Conference on Neural Networks (IJCNN)*, Barcelona, Spain, 2010. <https://doi.org/10.1109/IJCNN.2010.5596301>
- [31] P. Du, X. Luo, Z. He and L. Xie, "The Application of Genetic Algorithm-Radial Basis Function (GA-RBF) Neural Network in Stock Forecasting," in *Chinese Control and Decision Conference*, Xuzhou, China, 2010.
- [32] Pacelli, V. Bevilacqua And M. Azzollini, "An Artificial Neural Network Model to Forecast Exchange Rates," *Journal of Intelligent Learning Systems and Applications*, vol. 3, no. 2, pp. 57–69, 2011. <https://doi.org/10.4236/jilsa.2011.32008>
- [33] M. Pulido and P. Melin, "Genetic Algorithm and Particle Swarm Optimization of Ensemble Neural Networks with Type-1 and Type-2 Fuzzy Integration for Prediction of The Taiwan Stock Exchange," in *IEEE 8th International Conference on Intelligent Systems (IS)*, Sofia, Bulgaria, 2016. <https://doi.org/10.1109/IS.2016.7737413>
- [34] M. Qiu and Y. Song, "Predicting the Direction of Stock Market Index Movement Using an Optimized Artificial Neural Network Model," *PLoS ONE*, vol. 11, no. 5, pp. 1–11, 2016. <https://doi.org/10.1371/journal.pone.0155133>
- [35] P. Balestrassi, E. Popova, A. Paiva and J. M. Lima, "Design of Experiments on Neural Network's Training for Nonlinear Time Series Forecasting," *Neurocomputing*, vol. 72, no. 4–6, pp. 1160–1178, 2009. <https://doi.org/10.1016/j.neucom.2008.02.002>
- [36] J. Kleijnen, *Statistical Tools for Simulation Practitioners*. New York: Marcel Dekker Inc. Publ., 1987.
- [37] L. Trocine, L. Malone, A. Trocine and L. Malone, "Finding Important Independent Variables Through Screening Designs: A Comparison of Methods," in *Winter Simulation Conference*, Orlando, FL, USA, USA, 2000.
- [38] T. Andres and W. Hajas, "Using Iterated Fractional Factorial Design to Screen Parameters in Sensitivity Analysis of a Probabilistic Risk Assessment Model," in *Joint International Conference on Mathematical Models and Supercomputing in Nuclear Applications*, Karlsruhe, Germany, 1993.
- [39] L. Trocine, "An Efficient, Effective, and Robust Procedure for Screening More Than 20 Independent Variables Employing a Genetic Algorithm," University of Central Florida, Orlando, FL, USA, 2001.
- [40] L.-F. Hsieh, S.-C. Hsieh and P.-H. Tai, "Enhanced stock price variation prediction via DOE and BPNN-based optimization," *Expert Systems with Application*, vol. 38, no. 11, pp. 14178–14184, 2011. <https://doi.org/10.1016/j.eswa.2011.04.229>
- [41] M.-Y. Chen, M.-H. Fan, Y.-L. Chen and H.-M. Wei, "Design of Experiments on Neural Network's Parameters Optimization for Time Series Forecasting in Stock Markets," *Neural Network World*, vol. 4, no. 13, pp. 369–393, 2013. <https://doi.org/10.14311/NNW.2013.23.023>
- [42] A. Blum, *Neural Networks in C++: An Object-Oriented Framework for Building Connectionist Systems*, 1st ed. Michigan: John Wiley & Sons, 1992.
- [43] K. Swingler, *Applying Neural Networks: A Practical Guide*. Morgan Kaufmann, p. 1761, 1996.
- [44] S. Saarinen, R. Bramley and G. Cybenko, "III-Conditioning in Neural Network Training Problems," *Journal on Scientific Computing*, vol. 14, no. 3, pp. 693–714, 1993. <https://doi.org/10.1137/0914044>

Yunus E. Midilli graduated from Istanbul University in 2016. He continued his educational career at Ondokuz Mayıs University as a MSc degree student in intelligent systems engineering. He has been a PhD student at the Information Technology Department of Riga Technical University since 2018. Yunus E. Midilli worked as a System Supervisor of the Quality Assurance Department of a global tobacco manufacturing company between 2016 and 2018. During this period of time, he developed his Master Thesis regarding optimisation of neural networks with a case study in tobacco industry. After relocating to Riga for PhD studies, he has been continuing his business career as an Operations Specialist at Nordic bank since 2018. His PhD research is related to optimisation of neural networks in exchange rate prediction systems. His research interests include neural networks, data mining, optimisation, quality improvement.
E-mail: yunusemremidilli@gmail.com

Sergei Parshutin, Dr. sc. ing., Assistant Professor of Riga Technical University, Faculty of Computer Science and Information Technology, Institute of Information Technology. In 2011, he successfully defended his Doctoral Thesis at Riga Technical University. His research interests include data mining, machine learning, decision support systems, fuzzy logic and fuzzy systems, neural networks, intelligent agent technologies and automation control. He is the author and co-author of 24 publications, including 2 monographs and teaching aids.
E-mail: sergei.parshutin@rtu.lv
ORCID iD: <https://orcid.org/0000-0002-8689-3043>