2021 European Conference on Computing in Construction
Ixia, Rhodes, Greece
July 25-27, 2021

USING ARTIFICIAL NEURAL NETWORKS TO MODEL BRICKLAYING PRODUCTIVITY

X[1], Y[2], W[1] and Z[1]
[1]U
[2]U

ABSTRACT

The pre-planning phase prior to construction is crucial for ensuring an effective and efficient project delivery. Realistic productivity rates forecasted during pre-planning are essential for accurate schedules, cost calculation, and resource allocation. To obtain such productivity rates, the relationships between various factors and productivity need to be understood. Artificial neural networks (ANNs) are suitable for modelling these complex interactions typical of construction activities, and can be used to assist project managers to produce suitable solutions for estimating productivity. This paper presents the steps of determining the network configurations of an ANN model for bricklaying productivity.

INTRODUCTION

The productivity of the construction industry lags behind other sectors of the world's economy (Barbosa *et al.*, 2017). Due to this, productivity studies are vital, and indeed comprise a significant segment of construction research (Yi and Chan, 2014). Elaborate planning can lead to higher on-site productivity, and ultimately, to better performance for the industry. To this end, more accurate productivity rates are needed, which can be obtained by understanding the effects and functional relationships between various factors.

The relationship between the factors and the productivity rate, and especially the factors' combined effects are complex, thus making modelling challenging (Chao and Skibniewski, 1994). Owing to this, these studies can benefit from artificial neural networks (ANNs). ANNs can be trained to learn from even imperfect datasets, and provide quick and generalised solutions to a problem (Flood and Kartam, 1994a). ANNs can be used for modelling problems in which functional relationships between dependent and independent variables are subject to uncertainty, not understood, or may vary with time (Di Franco and Santurro, 2020). For all the above-mentioned reasons, they can perform better than traditional, statistical methods (Boussabaine, 1996) or even optimisation algorithms, which can operate slowly when the problem at hand involves a large number of variables (Flood and Kartam, 1994a) or when generalisation and patterns extracted from large datasets are the bottom line. Consequently, in this study, ANNs have been selected to analyse the effect of worker and wall characteristics on the bricklayers' labour productivity. Understanding the impact can lead to more realistic schedules and more accurate resource allocation.

After this section, the various applications of ANNs in the field of construction management are presented. Next, comes a short introduction of ANN. Then the steps of determining the network configurations of an ANN model for bricklaying productivity are presented together with the considerations of the various options. Finally, the directions of further model development are presented.

USE OF ANN IN CONSTRUCTION MANAGEMENT STUDIES

Artificial neural networks have been used in construction studies since the late 1980s (Flood and Kartam, 1994a; Adeli, 2001). There is a wide range of applications in the field of construction management. This section gives an overview of these.

Gerek et al. (2015) created two ANN models to study the productivity of bricklaying gangs. They ranked the factors influencing productivity, and found that wall type and working time had the greatest effect (Gerek *et al.*, 2015). Moselhi and Khan (2012) performed significance ranking of influencing factors, as well. However, in their case, the chosen trade was concrete formwork installation. They compared the results gained by applying ANN, fuzzy subtractive clustering, and stepwise regression analysis. Temperature and the type of the structure ranked highest (Moselhi and Khan, 2012). The same data set and input variables were used by Nasirzadeh et al. (2020) and Golnaraghi et al. (2019). The former aimed to use ANN to gain prediction intervals for labour productivity, while the latter compared the results obtained with the help of four different network configurations (Golnaraghi *et al.*, 2019; Nasirzadeh *et al.*, 2020). The output of the ANN by Portas and AbouRizk (1997) was also an interval (referred to as a zone) containing a small range of productivity values for concrete formwork operations. El-Gohary et al. (2017) and Mirahadi and Zayed (2016) sought to gain more accurate productivity rates for concrete works.

Tsehayae and Robinson Fayek (2016) analysed the productivity influencing factors for the same trade. To provide accurate productivity estimates for earthworks, Chao and Skibniewski (1994) used ANN, as well. Oral and Oral (2010) applied self-organising maps to investigate the effects of various influencing factors and to forecast construction productivity in the case of concrete works, formwork installation, and reinforcing works. Oral et al. (2016) compared the application of self-organising maps and artificial bee colony to predict productivity rates for ceramic tiling works. Song and AbouRizk (2008) modelled steel drafting and fabrication with the help of ANN. Heravi and Eslamdoost (2015) analysed the factors affecting productivity for power plant projects. They found supervision, proper coordination, and effective communication to be the most important ones. Moselhi et al. (2005) also investigated projects as a whole, rather than specific trades, and developed a model to understand the effect of change orders on labour productivity.

ANNs can be successfully used for purposes other than construction productivity analysis. Another area of application is cost estimation. Chao and Kuo (2018) used an ANN model to estimate the minimum rate of overheads and markup, while Moselhi et al. (1991) aimed to calculate the optimum markup. Oduyemi et al. (2015) modelled the life-cycle costs of existing buildings with the help of ANN.

Problems in the area of health and safety can also benefit from ANNs. Patel and Jha (2015) modelled the safety climate of construction projects, while Ayhan and Tokdemir (2019) applied ANNs to predict the outcome of construction incidents, thus making their model part of an accident prevention system.

Other applications include using ANNs for selecting the most suitable formwork system (Tam *et al.*, 2005), showing the relationship between human values and motivation of construction managers (Wang *et al.*, 2017), and determining the optimal performance measurement system to be used in off-site sheet metal fabrication shops (Said and Kandimalla, 2018).

To enhance the capabilities of an ANN approach, it is possible to use it combined with another method, thus creating a hybrid model. For instance, there are numerous examples for neuro-fuzzy models, where fuzzy logic is used in the ANN model to better model subjective variables. The models of Portas and AbouRizk (1997) and Ayhan and Tokdemir (2019) have fuzzy output layers, while in addition to that, the input layer of Mirahadi and Zayed's (2016) model also contains a couple of fuzzy variables.

Another option could be to combine ANNs with construction simulation. Song and AbouRizk (2008) embedded ANNs into their discrete-event simulation model to estimate the duration of individual activities, while Chao and Skibniewski (1994) generated the activity durations fed into the ANN model with the help of discrete-event simulation.

The next section presents how ANN models are developed for investigating construction productivity.

ARTIFICAL NEURAL NETWORKS

Artificial neural networks – similar to the human brain and the central nervous system – are able to learn and generalise from examples (Boussabaine and Kirkham, 2008). The components of the network are called neurons, processing elements, or nodes (Moselhi *et al.*, 1991; Boussabaine, 1996). These neurons are organised into three types of layers: input, hidden, and output layers. In any given network, there is one input layer, and one output layer, while the number of hidden layers varies. Figure 1 shows the topology of an ANN model.

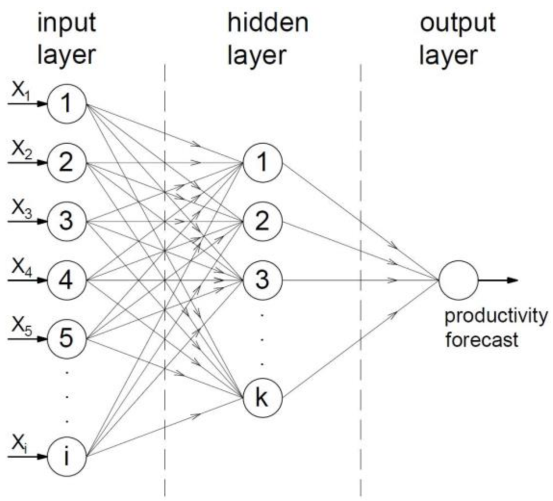

Figure 1: ANN model architecture

As can be seen in Figure 1, the neurons in the network are connected to each other. These links are weighted showing the strength of the connections (Boussabaine, 1996) . The input variables are fed into the input layer, then the signal arrives to the nodes of the hidden layer through the links, and finally, it is transmitted to the output layer. However, the weights of the connections modify the signal that arrives at the output neurons (Flood and Kartam, 1994a). The learning method determines how the weights change over the course of the training (Boussabaine, 1996). Based on what the network has learnt, it will be able to predict the outcome when presented with new input data points (Boussabaine and Kirkham, 2008). ANNs work like a black box, where the magic happens in the hidden layer, hidden from the user (Boussabaine, 1996; Adeli, 2001). In construction management problems, the relationship between the input and the output is typically complex due to unknown combined effects (Chao and Skibniewski, 1994). ANNs are well-suited to handle such cases.

APPLICATION OF ANN FOR PRODUCTIVITY ANALYSIS

The steps of developing the ANN model for analysing the labour productivity of bricklaying works are shown in Figure 2.

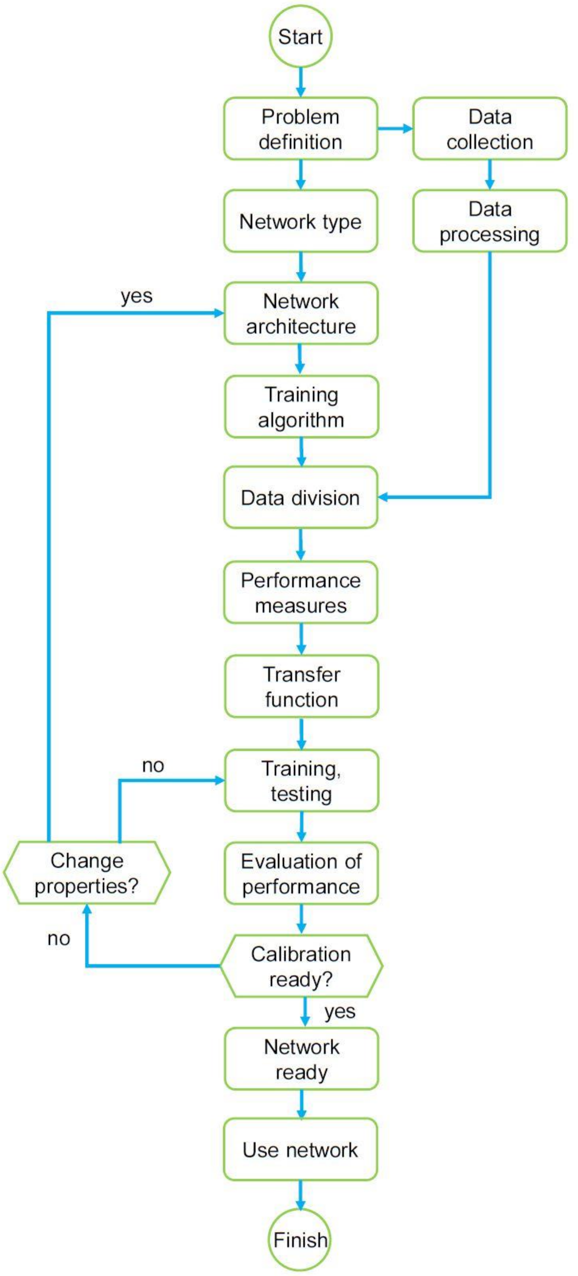

Figure 2: Steps of developing the ANN model

As with any other model, the process started with **problem definition**. In this case, the reason for developing this ANN model is to see how different factors affect the bricklayers' productivity. This goal determines the input and output variables. In this case only those factors that can be known during the pre-planning phase of a construction project are considered, particularly worker and wall characteristics. These factors, which include, for example, the experience of the bricklayers and the type of brick used for the wall, comprise the input neurons of the ANN model. The output neuron is the forecasted productivity rate. Figure 3 shows the ANN model used in this research.

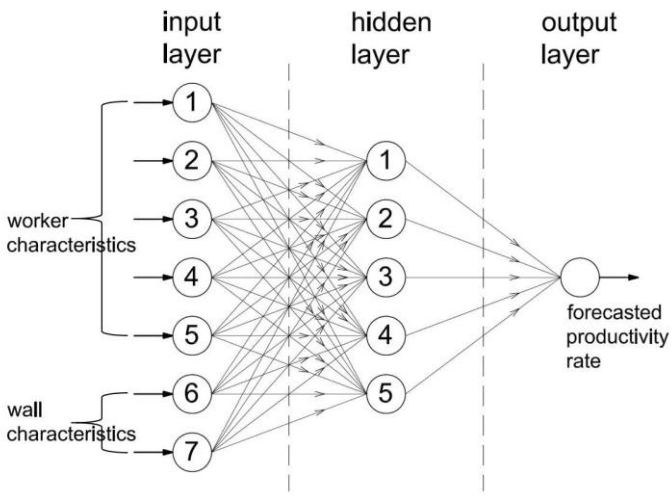

Figure 3. ANN model of bricklaying labour productivity

The selection of the variables informs the **data collection**. Patel and Jha (2015) suggest the minimum number of data points to be equal to the product of the neurons in each layer. According to this, at least 35 samples would be needed for the model in Figure 1. Too few training data points can cause underfitting, meaning that the network is not able to learn properly (Flood and Kartam, 1994a). In the case of productivity studies, especially if the data collection is done through work studies, it can be challenging to amass a substantial dataset. In this research, the data was collected at two construction projects by conducting a traditional work study. When the productivity rates were measured, note was made of the bricklayer working on the course, and the wall section where they worked. There are five worker characteristics, which are ordinal variables measured on a scale of one to three. For example, one is experience. In this case, one represents little, while three substantial experience. There are two wall attributes: difficulty is an ordinal variable measured on a scale of one to three with one being the easiest to construct and three being the most difficult; brick type is a categorical variable.

Data processing includes producing the data table based on the measurements obtained during data collection. In every row of this table, there is one productivity rate measured in bricks/hour together with the corresponding worker and wall attributes. The data table is the basis of determining the input and target matrices. Since the variables are scaled differently, normalisation of the data is needed (Flood and Kartam, 1994a, 1994b).

Data division means that the collected dataset is sorted into training and testing datasets. Most of the papers mentioned in the previous section used one of the following training-testing ratios: 80-20%, 75-25%, or 70-

30%. There often is a third set of data used for validating. In these cases, typically half of the testing set becomes the validating set. The number of data subsets is determined by the selected training algorithm. Normally, the dataset is divided randomly. However, it is essential that all subsets are representative of the collected data (Hagan *et al.*, 2014). In addition, Chao and Skibniewski (1994) found that having extremes in the training dataset helped the performance of the model. In this research, first the dataset obtained through the data collection is divided randomly into training, testing, and validating subsets in a 70-15-15% ratio. Then the divisions producing similar results across the subsets can be used for further analysis.

After the input and output variables are defined, the **network type** has to be chosen. One option is to select a basic network paradigm, another is to define a new one (Moselhi *et al.*, 1991). The network can learn in three ways. In the case of supervised learning, both the input and the output dataset is presented to the network, which calculates a predicted output for each input set, and then it is compared to the desired output (Flood and Kartam, 1994a). Another option is to provide a grade as an output, this is called reinforcement learning (Boussabaine, 1996). In the case of unsupervised learning, the targeted output dataset is not given to the network (Boussabaine, 1996). For example, self-organising maps belong to this category (Oral *et al.*, 2016). Based on the direction of the connections, there are feedforward and recurrent networks. Feedback loops can be found in the latter (Forbes *et al.*, 2004). The networks can also be static or dynamic. In the case of the former, the values of the input variables remain constant, while in the case of the latter, these values change over time (Flood and Kartam, 1994b).

Deterministic and stochastic networks can be distinguished, as well. In probabilistic neural networks probability density functions are used (Specht, 1990). The advantage of probabilistic neural networks is that they can be trained fast on sparse datasets (Sawhney and Mund, 2002; Tam *et al.*, 2005). Feedforward backpropagation networks are the most commonly used ones, see, for example, El-Gohary et al. (2017), or Tsehayae and Robinson Fayek (2016). Moselhi et al. (1991) chose backpropagation for its high accuracy and high interpolative performance. Other types include the radial basis used by, for instance, Moselhi and Khan (2012). Gerek et al. (2015) compared the performance of these two types of networks, and found that the radial basis network was more appropriate for their bricklaying example. Golnaraghi et al. (2019) investigated the application of the general regression network, and the adaptive neuro-fuzzy inference system in addition to the two above-mentioned networks. The backpropagation network suited the formwork assembly activity the best as presented in their paper (Golnaraghi *et al.*, 2019). Oral et al. (2016) used the self-organising map approach for a ceramic tiling activity. Bailey and Thompson (1990) presented the characteristics of many network paradigms.

The previously mentioned input variables are static, they do not change over time. The target output was measured; therefore, the training of the network is supervised. There are no feedback loops in the network, a feedforward network is defined. Due to its accuracy and high interpolative performance, backpropagation is selected.

The optimal network configuration can be obtained by following a trial-and-error approach, as there are no formal rules concerning this (Boussabaine and Kirkham, 2008; El-Gohary *et al.*, 2017). To determine the **network architecture**, decisions have to be made concerning the number of hidden layers and the number of neurons in each of these layers. It is worth starting with one hidden layer (Boussabaine and Kirkham, 2008). Two layers, however, can provide greater flexibility (Flood and Kartam, 1994a). Having too few hidden neurons in the network might lead to underfitting, and produce high error values (Flood and Kartam, 1994a; El-Gohary *et al.*, 2017). On the other hand, too many hidden nodes can lead to overfitting, in which case the error values are low; however, the network cannot work well outside the training patterns (Flood and Kartam, 1994a; El-Gohary *et al.*, 2017). At the start, the number of hidden neurons can be set at 2/3 or 70-90% of the input neurons, or at the average of the number of input and output nodes (Boussabaine and Kirkham, 2008; El-Gohary *et al.*, 2017). Having more than 2-2.5 times as many hidden neurons as input nodes might cause instability in the network (Patel and Jha, 2015; Ayhan and Tokdemir, 2019). Probabilistic neural networks typically have one hidden layer with as many neurons as training patterns (Sawhney and Mund, 2002; Tam *et al.*, 2005). One of the chosen network configurations can be seen in Figure 3. There are seven input neurons (one for each input variable mentioned before) and one output neuron (the forecasted productivity rate). There is one hidden layer. Based on the above recommendations, the number of hidden neurons is between 4 and 20. However, networks with a higher number of neurons (40, 100, 150) are also tested in the case of certain training algorithms. Furthermore, networks with two hidden layers with 5-40 neurons per layer are also examined.

The **training algorithm** or learning rule determines the way in which the weights are recalculated over the course of training. Selection depends on many factors, including the network type, and the dataset. In the case of backpropagation networks, the application of the generalised delta rule used to be widespread (Bailey and Thompson, 1990; Adeli, 2001). Adeli (2001) recommended choosing the adaptive conjugate gradient algorithm instead. Several models use the Levenberg-Marquardt algorithm due to it being fast and powerful, see, for example, Gerek et al. (2015). Another option is the Bayesian Regularisation algorithm suggested for small and noisy datasets, see, for example, Golnaraghi et al. (2019). Heravi and Eslamdoost (2015) compared the application of Bayesian Regularisation and scaled conjugate gradient learning rule, and found that the former had better generalisation performance. In the case of radial basis networks, the Gaussian function is the most commonly used (Adeli, 2001). For examples, see Gerek et al. (2015) and Moselhi and Khan (2012). Performance can be improved by allowing the learning rate to be modified during the training process; therefore use can be made of algorithms with adaptive learning rates (MathWorks United Kingdom, no date b). In this research, altogether six training algorithms are selected. The most frequently used Levenberg-Marquardt algorithm is the first choice for its speed and power. The Bayesian Regularisation algorithm recommended for noisy and small datasets is selected also for preventing overfitting. Two training algorithms (gradient descent with adaptive learning rate backpropagation and gradient descent with momentum and adaptive learning rate backpropagation) with adaptive learning rates are chosen, as well, for their ability to enhance performance by amending the learning rate. The Broyden-Fletcher-Goldfarb-Shanno quasi-Newton backpropagation is selected for its speed (MathWorks United Kingdom, no date a). The scaled conjugate gradient algorithm is chosen for its efficiency (Hagan *et al.*, 2014).

Over the course of the training of the network, the difference between the targeted and the predicted output is calculated, typically, with the help of statistical tools. This will be later used to evaluate the performance of the given network configuration. The most commonly used **performance measures** are the mean squared error (or the root-mean-square error), the mean absolute percentage error, the mean absolute error, and the correlation coefficient. In this study, the mean squared error, the mean absolute percentage error, and the correlation coefficient are used to evaluate the network configurations.

The output of the neurons is calculated based on the weights of the connections. Then a **transfer** or activation **function** is applied to this result (Flood and Kartam, 1994a). These functions can be linear, threshold, or sigmoid, which is the most widely used (Boussabaine and Kirkham, 2008). Portas and AbouRizk (1997) selected a sigmoid, while Tsehayae and Robinson Fayek (2016) applied a hyperbolic sigmoid transfer function. Heravi et al. (2015) experimented with different combinations of log-sigmoid, tangent sigmoid, and linear functions. They found that the log-sigmoid functions performed well with Bayesian Regularisation, while the tangent sigmoid function failed with the same algorithm (Heravi and Eslamdoost, 2015). Gerek et al. (2015) used saturating linear and linear activation functions in their two-layer feedforward network. Sigmoid functions are the most commonly used and best resemble the behaviour of biological neurons (Boussabaine and Kirkham, 2008). They are advised in the case of backpropagation by Bailey and Thompson (1990). Therefore, in this study, log-sigmoid and tangent sigmoid activation functions are selected for the first, or in the case of two hidden layers, the first two layers, and a linear transfer function is applied for the final layer.

After making the decisions regarding the initial settings of the network, the **training** and the **testing** could start. Next came the **evaluation of the performance** of the network configuration using the selected measures. If the performance is not satisfactory, there are two options. If the performance is below expectations, the network attributes need to be changed and the training and testing run again. The modifications are made one at a time to be able to observe the effect of the change. The other option, in the case of better performing networks, is to retrain the network with the same configuration to see if using different weights during training could help enhance the performance. This cycle continues until the optimal network configuration is found; the **calibration** is **ready**. When that happens, the **network** is **ready**, it can be **used** with new datasets to predict solutions and values (see Figure 2).

CONCLUSIONS

Investigating how different factors affect productivity is crucial for achieving higher levels of productivity on construction projects, and ultimately improving the performance of the sector. More realistic productivity rates lead to better schedules, cost calculations, and resource allocation. Artificial neural networks can be well used for such problems as they are able to provide solutions for complex problems involving non-linear relationships (Boussabaine, 1996). To help with model development for productivity studies, this paper summarises the decisions that need to be made and presents the considerations in the case of a model for bricklaying.

With the definition of the exact problem, the input variables are determined. The output variable is the productivity rate. After choosing the most suitable network type, the possible network configurations need to be listed. This includes choosing the number of hidden layers, and the neurons in each layer, the division of the dataset into training, testing, and validating datasets, the selection of the learning rule, and the transfer function. The interested reader is referred to the detailed explanation on how to choose the parameters for the network configuration in the previous section.

When the training and testing of all the networks are finished, they need to be compared and evaluated based on the chosen performance measures. In this way, the most optimal network configuration can be selected and used for predicting the productivity rate of the bricklaying activity.

In further stages of model development, the ANN model can be part of a discrete-event simulation model, where the bricklaying activity durations come from the ANN model component. This will be achieved based on the framework for creating hybrid simulation models, in which various simulation methods (such as discrete-event simulation) are combined with each other or other techniques (such as ANN), developed by Bokor et al. (2019).

As discussed in this paper, ANN and algorithms based on ANN are approaches that look into capturing knowledge from datasets. These models have the potential to transform the construction industry with the use of data-based solutions that can improve the way projects are delivered. In this particular case, ANN can be used to determine more realistic productivity rate predictions for accurate time and cost estimates, and improved project planning in bricklaying.

REFERENCES

Adeli, H. (2001) 'Neural networks in civil engineering: 1989-2000', *Computer-Aided Civil and Infrastructure Engineering*, 16(2), pp. 126–142. doi: https://doi.org/10.1111/0885-9507.00219.

Ayhan, B. U. and Tokdemir, O. B. (2019) 'Predicting the outcome of construction incidents', *Safety Science*. Elsevier, 113(March 2019), pp. 91–104. doi: https://doi.org/10.1016/j.ssci.2018.11.001.

Bailey, D. L. and Thompson, D. (1990) 'How to develop neural-network applications', *AI Expert*, 5(6), pp. 38–47.

Barbosa, F., Woetzel, J., Mischke, J., Ribeirinho, M. J., Sridhar, M., Parsons, M., Bertram, N. and Brown, S. (2017) *Reinventing Construction: A Route To Higher Productivity*. Available at: https://www.mckinsey.com/~/media/McKinsey/Industries/Capital Projects and Infrastructure/Our Insights/Reinventing construction through a productivity revolution/MGI-Reinventing-construction-A-route-to-higher-productivity-Full-report.ashx.

Bokor, O., Florez, L., Osborne, A. and Gledson, B. J. (2019) 'Overview of construction simulation approaches to model construction processes', *Organization, Technology and Management in Construction*, 11, pp. 1853–1861. doi: 10.2478/otmcj-2018-0018.

Boussabaine, A. H. (1996) 'The use of artificial neural networks in construction management: A review', *Construction Management and Economics*, 14(5), pp. 427–436. doi: https://doi.org/10.1080/014461996373296.

Boussabaine, A. and Kirkham, R. (2008) 'Artificial neural network modelling techniques for applied civil and construction engineering research', in Knight, A. and Ruddock, L. (eds) *Advanced Research Methods in the Built Environment*. Hoboken: Wiley-Blackwell, pp. 155–170.

Chao, L.-C. and Kuo, C.-P. (2018) 'Neural-Network-Centered Approach to Determining Lower Limit of Combined Rate of Overheads and Markup', *Journal of Construction Engineering and Management*, 144(2), pp. 04017117-1–8. doi: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001440.

Chao, L. and Skibniewski, M. J. (1994) 'Estimating Construction Productivity: Neural-Network-Based Approach', *Journal of Computing in Civil Engineering*, 8(2), pp. 234–251. doi: https://doi.org/10.1061/(ASCE)0887-3801(1994)8:2(234).

El-Gohary, K. M., Aziz, R. F. and Abdel-Khalek, H. A. (2017) 'Engineering Approach Using ANN to Improve and Predict Construction Labor Productivity under Different Influences', *Journal of Construction Engineering and Management*, 143(8), pp. 1–10. doi: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001340.

Flood, I. and Kartam, N. (1994a) 'Neural Networks in Civil Engineering. I: Principles and Understanding', *Journal of Computing in Civil Engineering*, 8(2), pp. 131–148. doi: https://doi.org/10.1061/(ASCE)0887-3801(1994)8:2(131).

Flood, I. and Kartam, N. (1994b) 'Neural Networks in Civil Engineering. II: Systems and Application', *Journal of Computing in Civil Engineering*, 8(2), pp. 149–162. doi: https://doi.org/10.1061/(ASCE)0887-3801(1994)8:2(149).

Forbes, D. R., McGurnaghan, H., Graham, L. D. and Smith, S. D. (2004) 'Concrete Placing Productivity Using a Novel Neural Network Design', in Khosrowshahi, F. (ed.) *Proceedings of the 20th Annual ARCOM Conference*. Edinburgh, UK: ARCOM, pp. 1053–62. Available at: http://www.arcom.ac.uk/-docs/proceedings/ar2004-1053-1062_Forbes_et_al.pdf.

Di Franco, G. and Santurro, M. (2020) 'Machine learning, artificial neural networks and social research', *Quality and Quantity*. Springer Netherlands. doi: https://doi.org/10.1007/s11135-020-01037-y.

Gerek, İ. H., Erdis, E., Mistikoglu, G. and Usmen, M. (2015) 'Modelling masonry crew productivity using two artificial neural network techniques', *Journal of Civil Engineering and Management*, 21(2), pp. 231–238. doi: https://doi.org/10.3846/13923730.2013.802741.

Golnaraghi, S., Zangenehmadar, Z., Moselhi, O. and Alkass, S. (2019) 'Application of Artificial Neural Network(s) in Predicting Formwork Labour Productivity', *Advances in Civil Engineering*, 2019, pp. 1–11. doi: http://dx.doi.org/10.1155/2019/5972620.

Hagan, M. T., Demuth, H. B., Beale, M. H. and De Jesús, O. (2014) *Neural Network Design*. 2nd edn. Available at: https://hagan.okstate.edu/NNDesign.pdf.

Heravi, G. and Eslamdoost, E. (2015) 'Applying Artificial Neural Networks for Measuring and Predicting Construction-Labor Productivity', *Journal of Construction Engineering and Management*, 141(10), pp. 1–11. doi: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001006.

MathWorks United Kingdom (no date a) *BFGS quasi-Newton backpropagation*. Available at: https://uk.mathworks.com/help/deeplearning/ref/trainbfg.html (Accessed: 21 March 2021).

MathWorks United Kingdom (no date b) *Gradient descent with adaptive learning rate backpropagation*.

Available at: https://uk.mathworks.com/help/deeplearning/ref/traingda.html (Accessed: 22 March 2021).

Mirahadi, F. and Zayed, T. (2016) 'Simulation-based construction productivity forecast using Neural-Network-Driven Fuzzy Reasoning', *Automation in Construction*, 65(May 2016), pp. 102–115. doi: https://doi.org/10.1016/j.autcon.2015.12.021.

Moselhi, O., Assem, I. and El-Rayes, K. (2005) 'Change orders impact on labor productivity', *Journal of Construction Engineering and Management*, 131(3), pp. 354–359. doi: https://doi.org/10.1061/(ASCE)0733-9364(2005)131:3(354).

Moselhi, O., Hegazy, T. and Fazio, P. (1991) 'Neural Networks as Tools in Construction', *Journal of Construction Engineering and Management*, 117(4), pp. 606–625. doi: https://doi.org/10.1061/(ASCE)0733-9364(1991)117:4(606).

Moselhi, O. and Khan, Z. (2012) 'Significance ranking of parameters impacting construction labour productivity', *Construction Innovation*, 12(3), pp. 272–296. doi: https://doi.org/10.1108/14714171211244541.

Nasirzadeh, F., Kabir, H. M. D., Akbari, M., Khosravi, A., Nahavandi, S. and Carmichael, D. G. (2020) 'ANN-based prediction intervals to forecast labour productivity', *Engineering, Construction and Architectural Management*, 27(9), pp. 2335–2351. doi: https://doi.org/10.1108/ECAM-08-2019-0406.

Oduyemi, O., Okoroh, M. and Dean, A. (2015) 'Developing an artificial neural network model for life cycle costing in buildings', in Raidén, A. B. and Aboagye-Nimo, E. (eds) *Proceedings of the 31st Annual ARCOM Conference*. Lincoln, UK: ARCOM, pp. 843–852. Available at: http://www.arcom.ac.uk/-docs/proceedings/2b4b7af225597aadc924a5ecf9b42b1a.pdf.

Oral, E. L. and Oral, M. (2010) 'Predicting construction crew productivity by using Self Organizing Maps', *Automation in Construction*. Elsevier B.V., 19(6), pp. 791–797. doi: https://doi.org/10.1016/j.autcon.2010.05.001.

Oral, E. L., Oral, M. and Andaç, M. (2016) 'Construction crew productivity prediction: Application of two novel methods', *International Journal of Civil Engineering*, 14(3), pp. 181–186. doi: https://doi.org/10.1007/s40999-016-0009-2.

Patel, D. A. and Jha, K. N. (2015) 'Neural Network Approach for Safety Climate Prediction', *Journal of Management in Engineering*, 31(6), pp. 05014027-1–11. doi: https://doi.org/10.1061/(ASCE)ME.1943-5479.0000348.

Portas, J. and AbouRizk, S. (1997) 'Neural Network Model for Estimating Construction Productivity', *Journal of Construction Engineering and Management*, 123(4), pp. 399–410. doi: https://doi.org/10.1061/(ASCE)0733-9364(1997)123:4(399).

Said, H. M. and Kandimalla, P. (2018) 'Performance Measurement of Building Sheet-Metal Ductwork Prefabrication under Batch Production Settings', *Journal of Construction Engineering and Management*, 144(2), pp. 04017107-1–13. doi: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001423.

Sawhney, A. and Mund, A. (2002) 'Adaptive Probabilistic Neural Network-based Crane Type Selection System', *Journal of Construction Engineering and Management*, 128(3), pp. 265–273. doi: https://doi.org/10.1061/(ASCE)0733-9364(2002)128:3(265).

Song, L. and AbouRizk, S. M. (2008) 'Measuring and Modeling Labor Productivity Using Historical Data', *Journal of Construction Engineering and Management*, 134(10), pp. 786–794. doi: https://doi.org/10.1061/(ASCE)0733-9364(2008)134:10(786).

Specht, D. F. (1990) 'Probabilistic neural networks', *Neural Networks*, 3(1), pp. 109–118. doi: https://doi.org/10.1016/0893-6080(90)90049-Q.

Tam, C. M., Tong, T. K. L., Lau, T. C. T. and Chan, K. K. (2005) 'Selection of vertical formwork system by probabilistic neural networks models', *Construction Management and Economics*, 23(3), pp. 245–254. doi: https://doi.org/10.1080/0144619042000205912.

Tsehayae, A. A. and Robinson Fayek, A. (2016) 'System model for analysing construction labour productivity', *Construction Innovation*, 16(2), pp. 203–228. doi: https://doi.org/10.1108/CI-07-2015-0040.

Wang, D., Arditi, D. and Damci, A. (2017) 'Construction Project Managers' Motivators and Human Values', *Journal of Construction Engineering and Management*, 143(4), pp. 04016115-1–10. doi: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001261.

Yi, W. and Chan, A. P. C. (2014) 'Critical Review of Labor Productivity Research in Construction Journals', *Journal of Management in Engineering*, 30(2), pp. 214–225. doi: https://doi.org/10.1061/(ASCE)ME.1943-5479.0000194.