

PAPER • OPEN ACCESS

Use of Artificial Neural Network for Initial design of steel structures

To cite this article: H Mote and S R Satish Kumar 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **660** 012064

View the [article online](#) for updates and enhancements.



ECS **240th ECS Meeting**
Oct 10-14, 2021, Orlando, Florida
**Register early and save
up to 20% on registration costs**
Early registration deadline Sep 13
REGISTER NOW

Use of Artificial Neural Network for Initial design of steel structures

H Mote¹ and S R Satish Kumar²

^{1,2} Department of Civil Engineering, Indian Institute of Technology Madras, Chennai, India.

E-mail: moteharshada2@gmail.com

Abstract. Design of steel structures is an iterative process that often turns complicated when requires initial guess. A good initial design guess that comes from past design experience can considerably reduce the number of subsequent analysis and design cycles. It is difficult to form declarative rules to express human intuitions and past experience. Another problem in design process is arriving at optimal solution, where optimization process in most cases is computationally expensive and time consuming. Artificial Neural Network (ANN) is a promising computational model that can perform cognitive tasks, such as learning and optimization. This paper focuses on application of ANN in design so as to make design process more efficient. The aim is to train ANN to arrive at optimal solution by considering design constraints along with practical. This is illustrated with an example of design of compression member which primarily requires initial guess. The economy in design is tried to obtain by providing training set of optimal solutions to a multilayer neural network. The design process followed to generate training set is in accordance with IS: 800-2007. The solutions given by ANN are approximate but have reasonable agreement with the expected solutions.

1. Introduction

Design of steel structures is an iterative and often a complicated process. Usually, design process starts with assuming some initial solution for the set of governing equations that define the problem statement. The initial solution is used to generate improved approximate solution through a number of iterations that satisfy the design constraints. Furthermore, this process is repeated multiple times with different initial solutions to arrive at optimal solution. As there can be multiple solutions, sometimes the designer has to take decision about the most acceptable solution, based on his /her knowledge and past experience. An alternative is to use mathematical optimization techniques which are time consuming and computationally expensive most of the times. Apart from this, formulating the cost function and practical considerations like availability of material, ease in construction etc. for optimization is difficult as it requires insight and experience in that field. The large number of design and practical constraints makes it difficult to computerize the entire design process.

A good initial guess considerably reduces the number of design and analysis cycles required and helps to arrive at optimal solution in less time. Having a good initial guess is difficult for beginners as it requires knowledge, intuition and experience. Also, it is difficult to write a program using procedural or declarative language that can generalize this experience in order to tackle an entirely new problem and that can learn through its own experience.



Artificial Neural network (ANN) is one of the promising tool that can be used to solve these types of problems. ANN is acknowledged as one of the universal approximators that can approximate any measurable function to any desired degree of accuracy, in a very specific and satisfying sense [1]. It is a superficial model of the functioning of human brain that accumulates experts' knowledge, intuitions and past experience in a generalized form and stores it in the form of network parameters. ANN consists of many parallel interconnected basic computing elements, which have processing ability inspired from biological neural networks. ANN can predict solution based on data presented for training, which can be theoretical, experimental or empirical based on good and reliable past experience or combination of these. Training data can be evaluated, verified or modified by the design professionals to inject human intelligence and intuition. Rogers applied ANN to perform structural analysis and developed guidelines to perform the same and demonstrated that ANN works much faster than mathematical optimization techniques [2]. Significant work in application of ANN in structural engineering has been done by Adeli [3]. Park and Adeli developed a neural dynamics model to solve linear structural optimization problem involving the minimum-weight plastic design of low-rise planar steel frames [4]. Tashakori and Adeli presented optimal (minimum weight) design of cold- formed steel space trusses using the neural dynamics model developed by Adeli and Park [5]. Kang and Yoon in 1994 developed a model for selection of economical truss members that will satisfy stresses requirements of a truss design [6].

The use of ANN for initial design was studied by Mukherjee and Deshpande [7]. The ANN model was built for preliminary design of reinforced concrete single span beams, which predicted good results closer to final solution.

Developing intelligent design process that will cooperate with conventional design process for better and faster results, is the main motivation behind this study. The objective of the study is to use ANN in initial design of steel structures so as to substantially reduce the number of design and analysis cycles and arrive at a feasible solution with practical considerations and other design constraints. The output predicted by ANN can be used as a starting point in design.

The scope of the study includes developing an ANN for predicting initial design of an angle section under compression, loaded through one leg. The training data is in accordance with Indian standard code IS: 800-2007 [8].

2. Basics of Artificial Neural network

ANN is a hypothesized model of functioning of human brain. It resembles the brain due to its ability of acquiring knowledge through learning process and storing it in the form of interconnecting synaptic weights [9].

A typical ANN model consists of:

- Neurons- a number of interconnected processing elements commonly referred to as neurons or nodes.
- Input layer- where data is presented to the network.
- Output layer- that holds the response of the network to the input.
- Hidden layer –These are the intermediate layers that enable network to perform and compute complicated association between input and output.
- Synaptic weights or synapse that are connecting link between two neurons.
- Summing function and activation function are constituents of a neuron. Summing function sums up the input signals from other neurons, weighted by the respective synaptic weight of the neuron. An activation function or a threshold function limits the permissible amplitude range of the output signal to some finite value. Activation function helps to induce nonlinearity in the model. Among the various activation functions available, sigmoid function is chosen because it has very simple derivative which is useful for development of learning algorithm.

General Steps of operating ANN are as follows:

- Specifying the Architecture - Choosing structural initial parameters like number of layers, number of neurons of each layer, initial values of weights, and the type of activation function

etc. To select number of nodes in each layer and number of hidden layers, no fixed guidelines are available. Thus, these parameters are decided by trial and error method. [7].

- Training or learning – It is the major and the most time-consuming part of neural network modelling. In a supervised learning model of ANN, the training process uses given input and output data sets to determine the optimal combination of weights by applying Error Correction Learning rule. In other words, each time an input is presented to the network, it predicts an output which is then compared with the expected output to update the weights so as to minimize predicting error. The training set must reflect all aspects of problem domain.
- Simulation - Simulation means use of the trained neural network to predict outputs of entirely new inputs. This corresponds to the 'recall' function of the brain.

The ANN can be designed to solve the problem by regression as well as by classification approach. In the regression model ANN maps input vector $[X]$ into continuous output variable vector $[y]$. While in classification model ANN maps input vector $[X]$ into discrete output variable vector $[y]$.

3. Application of Neural network for Initial design problem

3.1. Problem definition

Design of compression member which essentially is an iterative process is solved in this article with the application of ANN. The ANN was modelled in MATLAB R2018a. The input layer $[X] = [x_1 \ x_2 \dots \ x_n]$ was decided to have three nodes for Span L , Load P and limiting slenderness ratio (KL/r) . While Output layer $[y] = [y_1 \ y_2 \dots \ y_k]$ predicts width b and thickness t of angle section. The difference between the actual and predicted values is measured by mean of sum of squared errors of prediction and it is called as Cost function J . To minimise this cost function J , advance optimizer *fmincg* a function file provided by Carl Edward Rasmussen was used. The function uses the Polack-Ribiere flavour of conjugate gradients to compute search directions, and a line search using quadratic and cubic polynomial approximations. It uses the Wolfe-Powell stopping criteria with the slope ratio method for guessing initial step sizes. The gradients required for this method were calculated using back propagation algorithms [10]. ANN can be trained to predict size of the angle section for the problem by treating them as continuous variables or as discrete output variables. Hence two different models were implemented for regression and classification approaches.

3.2. Architecture of Neural Network – Regression Model

In this case, both b and t were treated as continuous variables. A multilayer neural network was initially built with one input layer, one output layer and one hidden layer. Later number hidden layer and nodes were increased as per the requirement. The input layer consist of three units $[X] = [x_1 \ x_2 \ x_3]$ each representing the Design span L , Design load P , and Limiting (KL/r) . The output layer was of only two nodes $[y] = [y_1 \ y_2]$ representing b and t of the angle section respectively. Here, the output values predicted by outer layer nodes do not have any limit. The cost function was calculated using (1).

$$J = \frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \left[\left(h_{\theta} (x^{(i)}) \right)_j - y_j^{(i)} \right]^2 \quad (1)$$

Where, m is total number of training examples, k is number of output nodes, $\left(h_{\theta} (x^{(i)}) \right)_j$ represents predicted j^{th} output node of ANN for given i^{th} training example $[x^{(i)} \ y^{(i)}]$.

The sigmoidal function was used as an activation function in the hidden layer but not in the output layer which is given by (2).

$$\sigma(z_j) = \frac{1}{[1 + e^{-z_j}]} \tag{2}$$

Where, $z_j = \sum \theta_{kj}x_k$ is the summation of product of synaptic weight θ_{kj} between j^{th} receiving node and all k nodes of previous layer including bias. The bias which is a parameter used to modulate output, was treated as a node with fixed value of 1. Bias controls the value at which activation function triggers. It is like an intercept added in a linear equation.

The values of b and t obtained from this network were mapped into discrete output by using Discrete mapping scale shown in figure 1, while figure 2 shows network architecture,

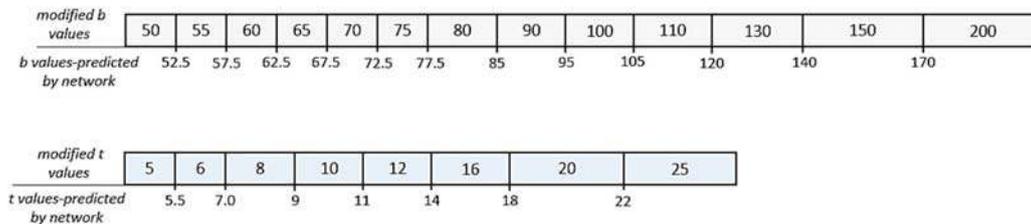


Figure 1. Discrete mapping scale for regression model.

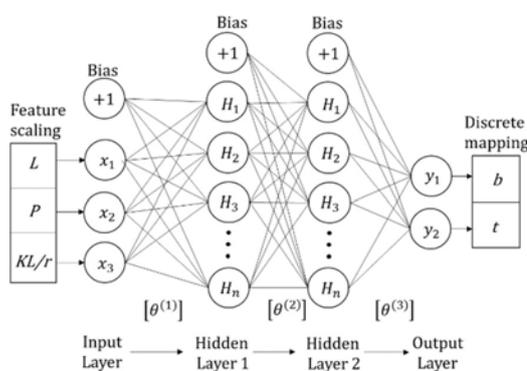


Figure 2. ANN -Regression model.

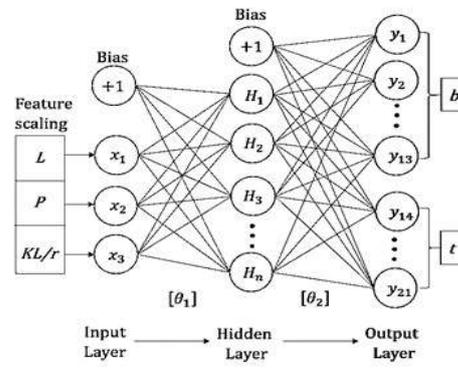


Figure 3. ANN -Classification model.

3.3. Architecture of Neural Network –Classification model

A multilayer neural network was built with one input layer, one output layer and one hidden layer. The input layer consist of three units $[X] = [x_1 x_2 x_3]$ representing Design span L , Design load P , and limiting (KL/r) . The output consists of 21 units i.e. $[y] = [y_1 y_2 \dots y_k]$ where $k = 21$, out of which first 13 nodes were for different discrete widths of available angle sections and rest 8 were for different discrete available thickness as shown in figure 3. Node with maximum probability among the first 13 output nodes gives width of the section while the one with maximum probability among the last 8 nodes gives thickness of the section. Initially, number of nodes in hidden layer was arbitrarily set to 12 and was modified later as per the requirement. The cost function for ANN classification model was calculated using (3),

$$J = \frac{1}{m} \sum_{i=1}^m \sum_{j=1}^k \left[-y_j^{(i)} \log \left(\left(h_{\theta} \left(x^{(i)} \right) \right)_j \right) - \left(1 - y_j^{(i)} \right) \log \left(1 - \left(h_{\theta} \left(x^{(i)} \right) \right)_j \right) \right] \quad (3)$$

3.4. Obtaining Training data set

The problem used for illustration is a design of compression member using Indian standard equal angle section (ISA). The material availability and minimum weight are used as design optimization constraints to arrive at economical section. The angle was assumed as loaded through one leg with end condition of number of bolts connected to gusset plate being greater than 2 and Gusset/connecting member fixity as Fixed. The design procedure followed Indian Standard code IS 800:2007. Out of all 72 Indian standard equal angle sections, 50 most commonly available sections were selected. The design load P is a set of load values ranging from 10 kN to 1620 kN with an interval of 10kN and the Design Span (L) is set of spans ranging from 0.25 m to 10 m with 0.25 m interval. For every combination of P and L the capacity of all sections was checked. The angle section with least weight and design load carrying capacity greater than P was selected. To calculate the design compressive load carrying capacity of section, Clause 7.5.1.2 of pg. 48 was used. The two design constraints of maximum allowable slenderness ratio of 180 and of 250, as per Clause 3.8 of IS: 800-2007 was used. According to this clause, limiting slenderness ratio is 180 for member carrying compressive load due to dead and imposed loads while it is 250 for member subjected to compressive load resulting from combination of wind and earthquake. The algorithms to generate training data was coded using Pandas Library of Python Programming language. The algorithm generated more than 8500 possible examples. The data was independently and identically distributed into training, cross validation and test set in the ratio 3:1:1[11].

3.5. Deciding the Architecture of Neural Network

To evaluate the performance of the network, the data set consisting of inputs and outputs was divided into three sets namely training set, Cross Validation Set and Test set in the ratio 3:1:1. The network is trained using training set only but the cost function was calculated for both training and cross validation set. This training cost function and cross validation cost function are plotted with respect to different model parameters such as number of iterations, regularization, number of hidden layer units, number of training sets etc. These plots are referred as learning curves. These learning curves were used to classify model into high-bias (under-fitting) or high-variance model (over-fitting) and to get the values of properties which give optimal fit. Though the over-fitted model shows least error, it lacks ability to generalize. Hence the over-fitting of the model is avoided. After adjusting network model, the final performance of model is assessed using test set [12].

Both training error and validation error were high in the initial network architecture of single hidden layer regression model, proving it to be under high bias category (under-fitting). Hence additional hidden layer was introduced in the model. The cost function reduced significantly after introducing second hidden layer. Though the error was lower than before, the model failed to identify smaller angle sections. Least size of section predicted by regression model was ISA 60x5 mm.

In case of classification model both training error and validation error were low but the least size section predicted by this model was ISA 90x6mm section (higher than the least size of ISA 60 X 5 mm, predicted by regression model). This happened because the percentage of examples of angle sections of size 150x16, 150x20, 200x12, 200x16, 200x20 and 200x25 was more than 60% in the training set (with total of 5145 examples) as shown in figure 4. Same was the case with validation and test set. This resulted skewness in data which is understandable since large range of load and span is satisfied by these sections. Because of which all the smaller sections (b<130mm) were treated as outliers by both networks. The resulting models were more biased towards the higher sections. Hence, these networks were able to predict higher section accurately but failed in predicting smaller sections.

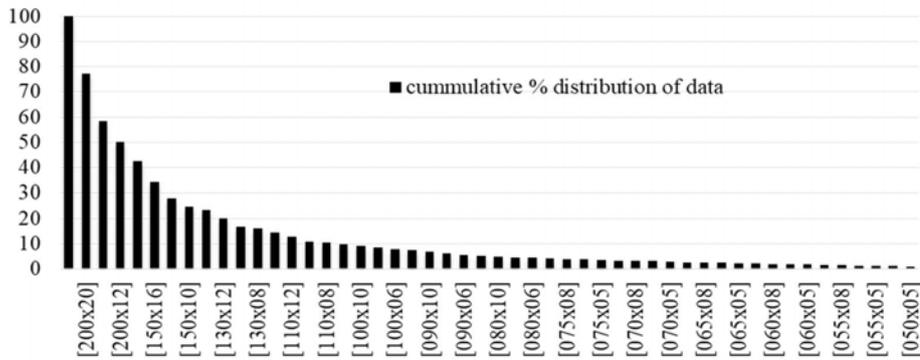


Figure 4. Distribution of cumulative percentage data in training set.

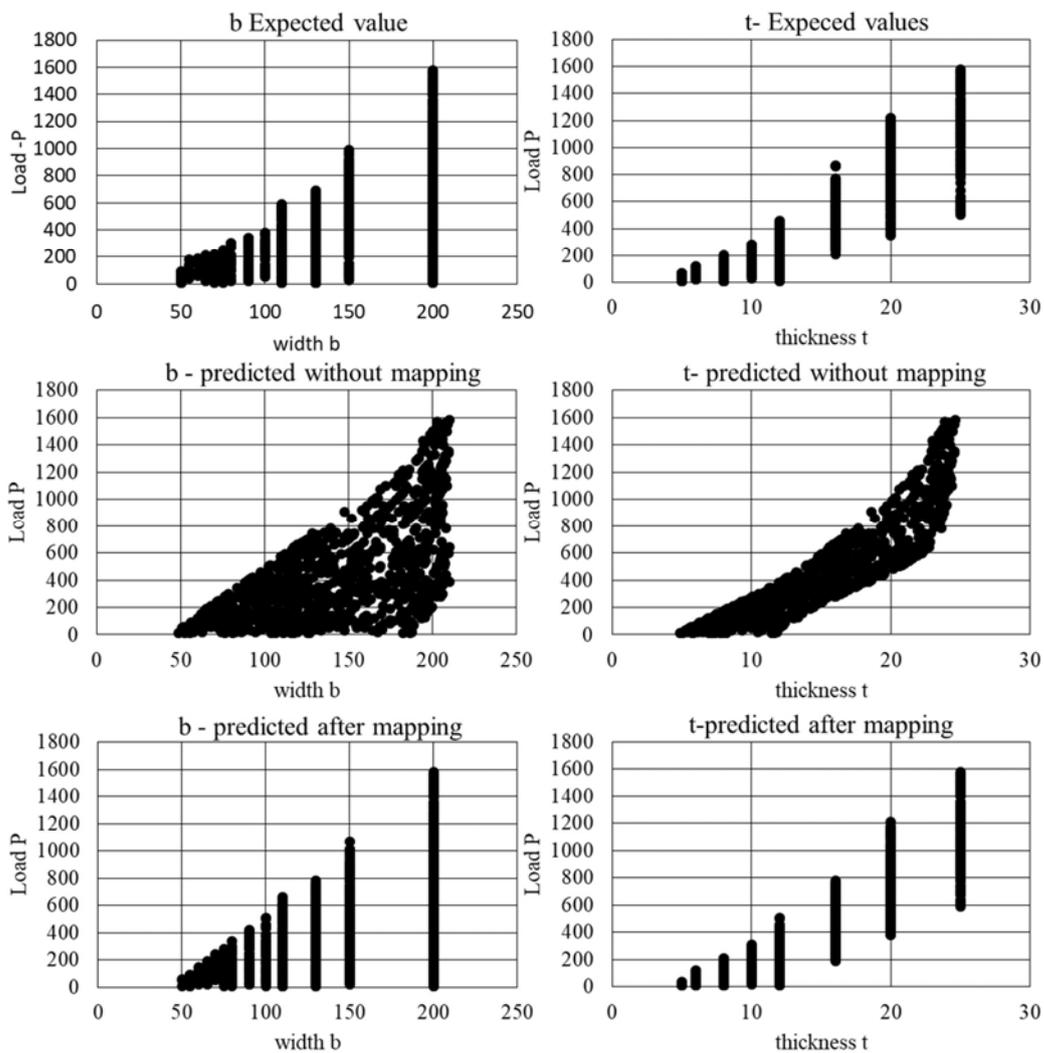


Figure 5. Width and thickness of angle sections predicted using ANN -Regression model.

To overcome this problem, large number of examples of these larger sections were removed randomly from the training set. The reduced training set was of only 1430 examples. The neural networks were trained freshly with this reduced training set. Performance of both the networks improved in predicting smaller sections. This shows that, minimization of Cost function is essential for any neural network but is not the only criteria of predicting its accuracy, especially when data contains more examples of one class. Thorough inspection of the data predicted by network for various cases is also important. It is important to understand that neural networks are approximators and hence we cannot expect prediction of exact solution. Further, classification model was checked for different number of hidden nodes ranging from 3 to 30. It was found that 15 number of hidden nodes gave better results. Cost function converged after 80 number of iterations. Further increase in hidden layer nodes did not show any significant improvement in prediction capacity of the model. Similarly, regression neural network model with 2 hidden layers each containing 5 hidden layer nodes and 60 number of iterations gave better prediction capacity. Percentage errors in prediction of b and t before mapping into discrete values was 45.4% and 54.5% respectively. These errors were brought down to 3.8 % and 7.0% respectively after the use of mapping scale. The conversion can be observed from the figure 5.

3.6. Comparison between ANN Regression and Classification model

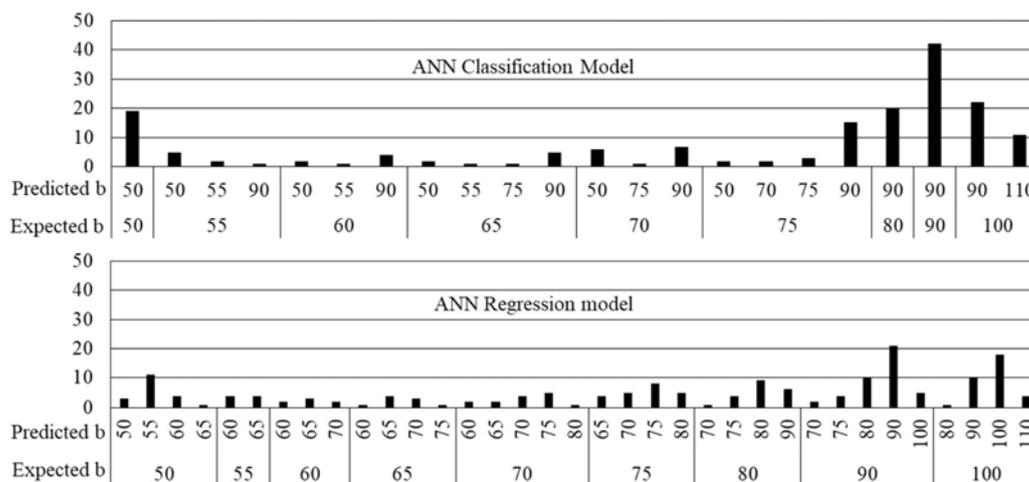


Figure 6. Performance in predicting smaller sections.

Regression model size is much lesser than size of classification model. In ANN knowledge is stored in the form of synaptic weight $[\theta]$ which is much smaller for regression model as compared to classification model. The smaller sections sizes predicted by regression model were closer to exact sizes compared to classification model. This can be seen from figure 6, where classification model failed in most of the cases where b is between 60-80mm. Table 1 shows that percentage error of regression model in predicting t is more than Classification model. From above comparison, it can be concluded that for the above design problem regression model is better than classification model because of its size and better predicting capacity for smaller sections as well as larger sections. Using test set, the regression model displayed error of 5.1% and 7.25% in predicting b and t respectively, while for classification model, these errors were 5.35% and 5.01% respectively.

It is to be noted that cost function of regression model was lower than that of classification model. The output values range from 0 to 1 for classification model but not for regression model. Hence cost function calculated using training set J_{train} , validation set J_{train} and test set J_{train} cannot be used for comparison.

Table 1. Comparison of prediction error between final selected ANN regression and classification model models.

	Input layer	Hidden layers		Output layer	Cost functions			Average % Error in validation set	
		(1)	(2)		J_{train}	J_{val}	J_{test}	b	t
ANN-Regression	3	5	5	2	0.0573	0.0646	0.0754	3.8	7.0
ANN-Classification	3	15	-	21	1.6300	1.6570	1.7277	4.8	4.9

4. Conclusion

ANN is a flexible tool that can be used to map complex, nonlinear input -output relation. It can capture human knowledge in generalized form. It can predict solutions that are close to exact solution with acceptable margins. But they are sensitive to training set. The training data set should reflect all the aspects of the problem. Skewness of training data sets significantly affects predictions of networks making it biased towards majority. This paper demonstrates the competence of generalization of ANN in initial design problem of an angle compression member loaded through one leg. Both ANN solving problem with regression and classification approaches were studied. It can also be concluded that ANN can be used to design solutions which are close to optimal design solution provided that ANN are trained using data that reflects optimal solutions.

References

- [1] Hornik K, Stinchcombe M and White H 1989 Multilayer feedforward networks are universal approximators *Neural Networks* **2** 359-66
- [2] Rogers J L 1994 Simulating structural analysis with neural network *J. Computing in Civil Engineering* **8** (2) 252-65
- [3] Adeli H 2001 Neural networks in civil engineering:1989-2000 *Computer-Aided Civil and Infrastructure Engineering* **16** (2) 126-42
- [4] Adeli H and Park H S 1995 A neural dynamics model for structural optimization: Theory *Computers and Structures* **57** (3) 391-9
Adeli H and Park H S 1995 Optimization of space structures by neural dynamics model *Neural Networks* **8** (5) 769-81
- [5] Tashakori A and Adeli H 2002 Optimum design of cold-formed steel space structures using neural dynamics model *Journal of Structural Engineering* **58** (12) 1545-66
- [6] Kang H T and Yoon C J 1994 Neural network approaches to aid simple truss design problems *Microcomputers in Civil Engineering* **9** (3) 211-18.
- [7] Mukherjee A and Deshpande J M 1995 Modelling initial design process using artificial neural networks *J. Computing in Civil Engineering* **9** (3) 194-200
- [8] Bureau of Indian Standards (BIS) 2007 General Construction in Steel- Code of practice: IS 800
Bureau of Indian Standards (BIS) 1989 Dimensions for hot rolled steel beam, column, channel and angle sections: IS 808
- [9] Kapania R K and Liu Y 1998 *Application of Artificial Neural Networks in Structural Engineering with Emphasis on continuum models* (Virginia Polytechnic Institute and State University)
- [10] Ng A Machine learning *Coursera online course* Standford viewed Nov 23 2018 <<https://www.coursera.org/learn/machine-learning>>
- [11] Pedregosa F *et. al.* 2011 Scikit-learn: Machine Learning in Python *Journal of Machine Learning Research* **12** 2825-30.
- [12] Hastie T Tibshirani R and Friedman J 2009 *The Elements of Statistical Learning: Data Mining, Inference, and Prediction Second Edition* (New York: Springer-Verlag) pp 37-39