OPTIMIZATION OF RC COLUMNS USING ARTIFICIAL NEURAL NETWORK

K.KAARTHIKEYAN[a], DR.R.MERCY SHANTHI[b]

[a]*student, School of Civil Engineering, Karunya University Coimbatore,Tamil Nadu, India*
[b]*Associate Professor, School of Civil Engineering, Karunya University Coimbatore,Tamil Nadu,*

ABSTRACT

In the structural design of column, the dimensions of the column and the reinforcement are initially assumed and the interaction formula is used to verify the suitability of chosen dimension and reinforcement. The approach necessitates few trails for coming up with an economical and safe design Using conventional method to design uni axial and biaxial column is long process but using ANN we can do the design easily. In neural network is trained with trained data results of the testing data results of the testing data may be obtained. Result indicates the Neural-network capable of predicting the exact solution with proper training but this ability depends on the complexity of the column optimization itself.

1. INTRODUCTION

1.1 OPTIMIZATION

Structural Engineering involves understanding and modelling of natural phenomenon, material behaviour and laws of mechanics, intuition, past experience or expertise and analysis techniques. The modern computer can bring speed, efficiency and accuracy in analysis of structures. But to computerize the areas such as conceptual design, modelling of natural phenomenon and material behaviour and damage assessments is extremely difficult as it requires human expertise. Structural design is an iterative process. The initial design is the first step in design process.

Artificial Neural Network is a new technology emerged from approximate simulation of human brain and has been successfully applied in many fields of engineering. Neural networks and demonstrate powerful problem solving ability.

Finding an alternative with the most cost effective or highest achievable performance under the given constraints, by maximizing desired factors and minimizing undesired ones. In comparison, maximization means trying to attain the highest or maximum results or outcome without regard to cost or expense. Practice of optimization is restricted by the lack of full information, and the lack of time to evaluate what information is available (see bounded reality for details). In computer simulation (modelling) of business problem, optimization is achieved usually by using linear programming techniques.

1.2 ARTIFICIAL NEURAL NETWORK

Artificial Neural Network are computational models inspired by biological neural network used for processing large number of inputs which are mostly unknown.Human brain is most powerful pattern recognition engine ever invented.

Instead of programming computational system to do specific tasks, teach system how to perform task
To do this, generate Artificial Intelligence System- AI.AI systems must be adaptive – able to learn from data on a continuous basis

1.2.1 NEURAL NETWORK APPLICATION DEVELOPMENT

The development process for an ANN application has eight steps.

- Collection of data
- Training and testing data separation For a moderately sized data set, 80% of the data are randomly selected for training, 10% for testing, and 10% secondary testing.
- Configuring neural network model
- Important considerations are the exact number of perceptrons and the number of layers.
- Parameter tuning and weight initialization
- Data transformation transforms the application data into the type and format required by the ANN.
- Training of the ANN
- Testing of the ANN

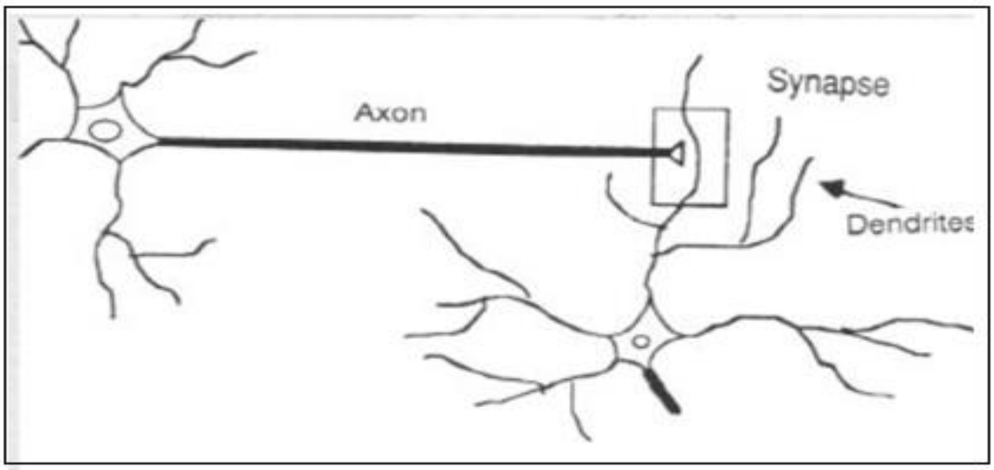


Fig 1 Biological neuron

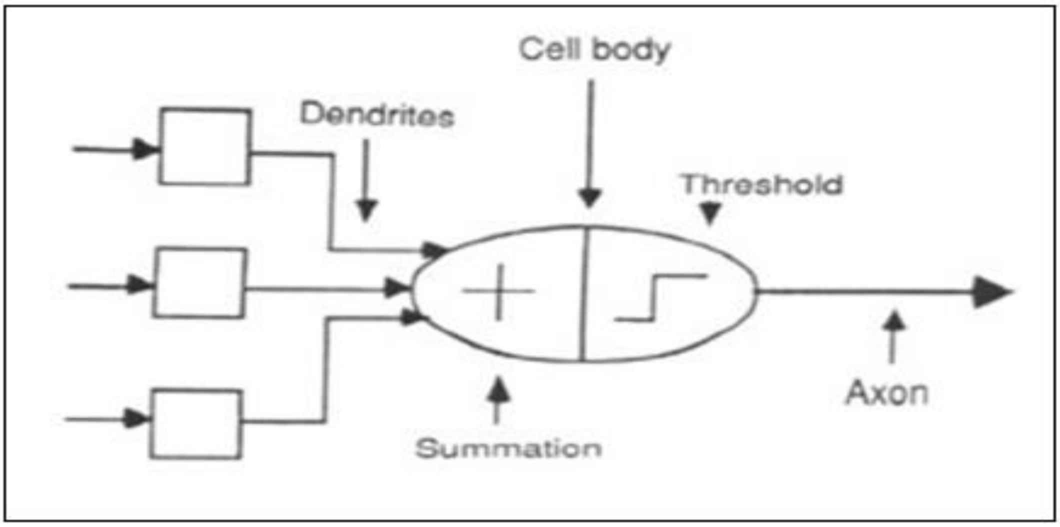


Fig 2 Artificial neuron

1.3 CLASSIFICATION OF LEARNING STRATEGIES

1.3.1 SUPERVISED

- Process of using desired output for training the NN
- It employees a teacher to assist the network by telling the network what the desire response to a given input
- Weights are modified according to the required output
- Not practicable in all cases
- It is based on a labeled training set.
- The class of each piece of data in training set is known.
- Class labels are pre-determined and provided in the training phase.

1.3.2 UNSUPERVISED

- No teacher Required
- Similar to the students learning on their own
- Adaption rules
- Adaption rule generate error signals

1.3.3 REINFORCED

- A teacher is assumed to b present but right answer is not given to the network
- Network is given an indication whether output is right or wrong

Network use this indication to improve performance

2. PARAMETERS USED

P_u Axial load

A_{st} Area of steel

B Breath of beam

D Overall depth of beam or slab

f_{ck} Characteristic compressive strength of concrete

f_y Characteristic strength of steel

M_u Design bending moment

p_t Percentage of steel

3. Analytical study

Table 1 Input values calculated for short columns using excel spread sheets for 9 sets

P_u	350	350	350	350	350	350
Span	700	700	700	700	700	700
Breath	150	155	160	165	170	175
Depth	150	155	160	165	170	175
Cover	25	25	25	25	25	25
Factored load	350	350	350	350	350	350
Grade of concrete	25	25	25	25	25	25
Grade of Steel	415	415	415	415	415	415
% of steel	2.0726	1.7042	1.3698	1.0654	0.7874	0.5329

Table 2 Training data's for short column for 9 sets

P u	350000	350000	350000	350000	350000	350000	350000	350000	350000
Breath	150	155	160	165	170	175	180	185	190
Depth	150	155	160	165	170	175	180	185	190
Moment	5250000	5250000	5250000	5250000	5250000	5250000	5250000	5250000	5250000
Cover	25	25	25	25	25	25	25	25	25
Factored load	3500000	3500000	3500000	3500000	3500000	3500000	3500000	3500000	3500000
Grade of concrete	25	25	25	25	25	25	25	25	25
Grade of Steel	415	415	415	415	415	415	415	415	415
% of steel	2.5	2	1.5	1	1	0.5	0.5	0	0

3.1 Mat lab programme for 9 sets of values

```
  p = [350000 350000 350000 350000 350000 350000 350000 350000 350000;
      150 155 160 165 170 175 180 185 190;150 155 160 165 170 175 180 185 190;
      5250000 5250000 5250000 5250000 5250000 5250000 5250000 5250000 5250000;
      25 25 25 25 25 25 25 25 25;350000 350000 350000 350000 350000 350000 350000 350000 350000;
      25 25 25 25 25 25 25 25 25;415 415 415 415 415 415 415 415 415];
t = [2.5 2 1.5 1 1 0.5 0.5 0 0];
%         plot(p,t,'o')
net = newff(p,t,10);
       y1 = sim(net,p)
%         plot(p,t,'o',p,y1,'x')
net.trainParam.epochs = 50;
net.trainParam.goal = 0.01;
net = train(net,p,t);
       r=[350000 155 155 5250000 25 350000 25 415]';
       y2 = sim(net,r)
%         plot(p,t,'o',p,y1,'x',p,y2,'*')
```

Table 3 Input values calculated for short columns using excel spread sheets for 23 sets

b	d	pu	Mu	cover	fck	pu/(fck*bd)	mu/(fckbD^2)	d/d'	p/fck	p
150	150	350000	5250000	25	25	0.0622	0.0622	0.1666	0.1	2.5
155	155	350000	5250000	25	25	0.5827	0.0563	0.1612	0.08	2
160	160	350000	5250000	25	25	0.5468	0.0512	0.1562	0.06	1.5
165	165	350000	5250000	25	25	0.5142	0.0467	0.1515	0.04	1
170	170	350000	5250000	25	25	0.4844	0.0427	0.147	0.04	1
175	175	350000	5250000	25	25	0.4571	0.0391	0.1428	0.02	0.5
180	180	350000	5250000	25	25	0.432	0.036	0.1388	0.02	0.5
150	150	350000	7000000	25	25	0.6333	0.0829	0.1666	0.1	2.5
155	155	350000	7000000	25	25	0.5827	0.07519	0.1612	0.06	1.5
160	160	350000	7000000	25	25	0.5468	0.0683	0.1562	0.06	1.5
165	165	350000	7000000	25	25	0.5142	0.0623	0.1515	0.06	1.5
170	170	350000	7000000	25	25	0.4844	0.0569	0.147	0.02	0.5
175	175	350000	7000000	25	25	0.4571	0.0522	0.1428	0.02	0.5
180	180	350000	7000000	25	25	0.432	0.048	0.1388	0.02	0.5
185	185	350000	7000000	25	25	0.409	0.0442	0.1351	0.02	0.5
150	150	350000	8750000	25	25	0.6222	0.1037	0.1666	0.12	3
155	155	350000	8750000	25	25	0.5827	0.0938	0.1612	0.08	2
160	160	350000	8750000	25	25	0.5468	0.0854	0.1562	0.08	2
165	165	350000	8750000	25	25	0.5142	0.0779	0.1515	0.6	1.5
170	170	350000	8750000	25	25	0.4844	0.0712	0.147	0.04	1
175	175	350000	8750000	25	25	0.4571	0.0653	0.1428	0.04	1
180	180	350000	8750000	25	25	0.432	0.06	0.1388	0.04	1
185	185	350000	8750000	25	25	0.409	0.052	0.1351	0.02	0.5

Table 4 Training values for 23 sets of values

pu	Breath	Depth	Moment	cover	Factored load	Grade of concrete	Grade of Steel	% of steel
350000	150	150	5250000	25	3500000	25	415	2.5
350000	155	155	5250000	25	3500000	25	415	2
350000	160	160	5250000	25	3500000	25	415	1.5
350000	165	165	5250000	25	3500000	25	415	1
350000	170	170	5250000	25	3500000	25	415	1
350000	175	175	5250000	25	3500000	25	415	0.5
350000	180	180	5250000	25	3500000	25	415	0.5
350000	150	150	7000000	25	3500000	25	415	2.5
350000	155	155	7000000	25	3500000	25	415	1.5
350000	160	160	7000000	25	3500000	25	415	1.5
350000	165	165	7000000	25	3500000	25	415	1.5
350000	170	170	7000000	25	3500000	25	415	0.5
350000	175	175	7000000	25	3500000	25	415	0.5
350000	180	180	7000000	25	3500000	25	415	0.5
350000	185	185	7000000	25	3500000	25	415	0.5
350000	150	150	8750000	25	3500000	25	415	3
350000	155	155	8750000	25	3500000	25	415	2
350000	160	160	8750000	25	3500000	25	415	2
350000	165	165	8750000	25	3500000	25	415	1.5
350000	170	170	8750000	25	3500000	25	415	1
350000	175	175	8750000	25	3500000	25	415	1
350000	180	180	8750000	25	3500000	25	415	1
350000	185	185	8750000	25	3500000	25	415	0.5

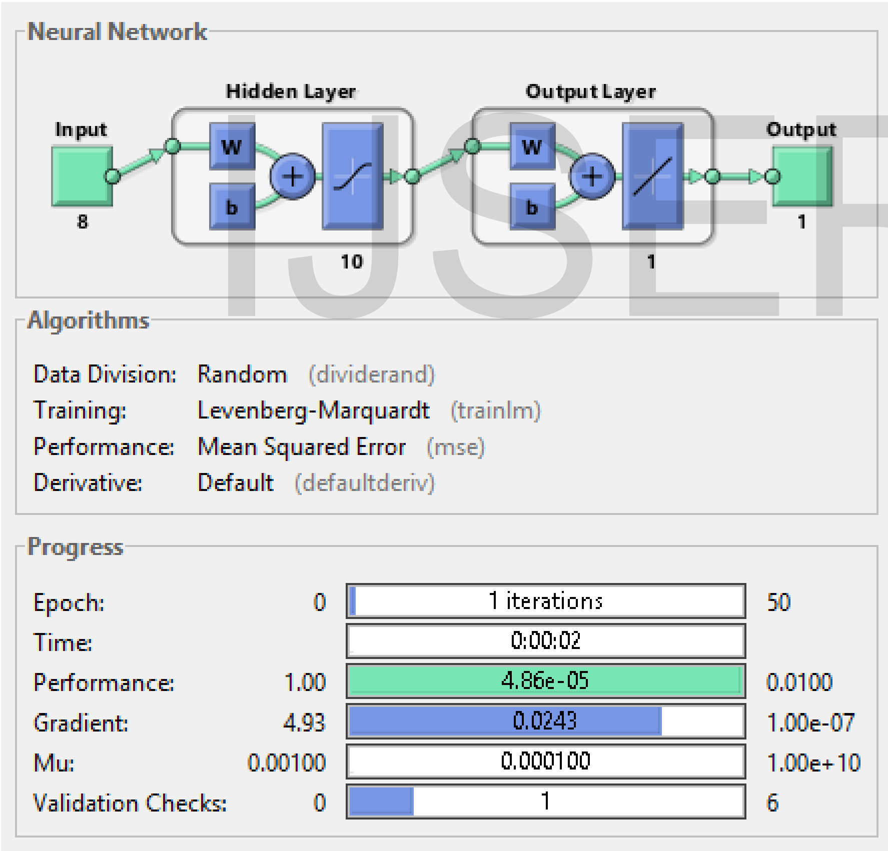
3.2 Mat lab programme for 23 sets of values

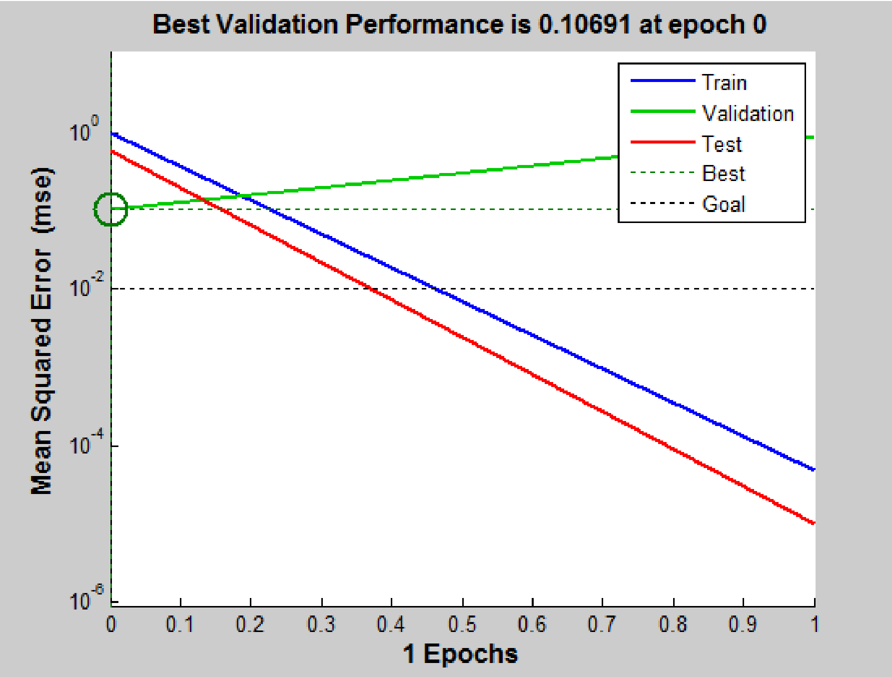
```
 p =[350000 350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  350000  35
    150 155 160    165 170 175 180 150 155 160 165 170 175 180 185 150 155 160 165 170 175 180 185;
    150    155 160 165 170 175 180 150 155 160 165 170 175 180 185 150 155 160 165 170 175 180 185;
    5250000     5250000 5250000 5250000 5250000 5250000 5250000 7000000 7000000 7000000 7000000 7000000 7000000 7000000 7000000 8750000 8750000 8750000 875000
    25 25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25;
    3500000     3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 3500000 35000
    25 25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25  25;
    415    415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415 415];
t = [2.5000 2   1.5000  1   1   0.5000  0.5000  2.5000  1.5000  1.5000  1.5000  0.5000  0.5000  0.5000  0.5000  3   2   2   1.5000  1   1   1   0.5000];
       plot(p,t,'o')
net = newff(p,t,10);
       y1 = sim(net,p)
       plot(p,t,'o',p,y1,'x')
net.trainParam.epochs = 50;
net.trainParam.goal = 0.01;
net = train(net,p,t);
       r=[350000 170 170 8750000 25 3500000 25 415]';
       y2 = sim(net,r)
%        plot(p,t,'o',p,y1,'x',p,y2,'*')
```

4. Results and discussions

4.1 Results for 9 set of inputs

```
>> fdf

y1 =

    3.7471    3.2169    2.4576    1.6896    0.6730    1.3312    1.7150    0.7723    0.6616


y2 =

    3.2169

>>
```

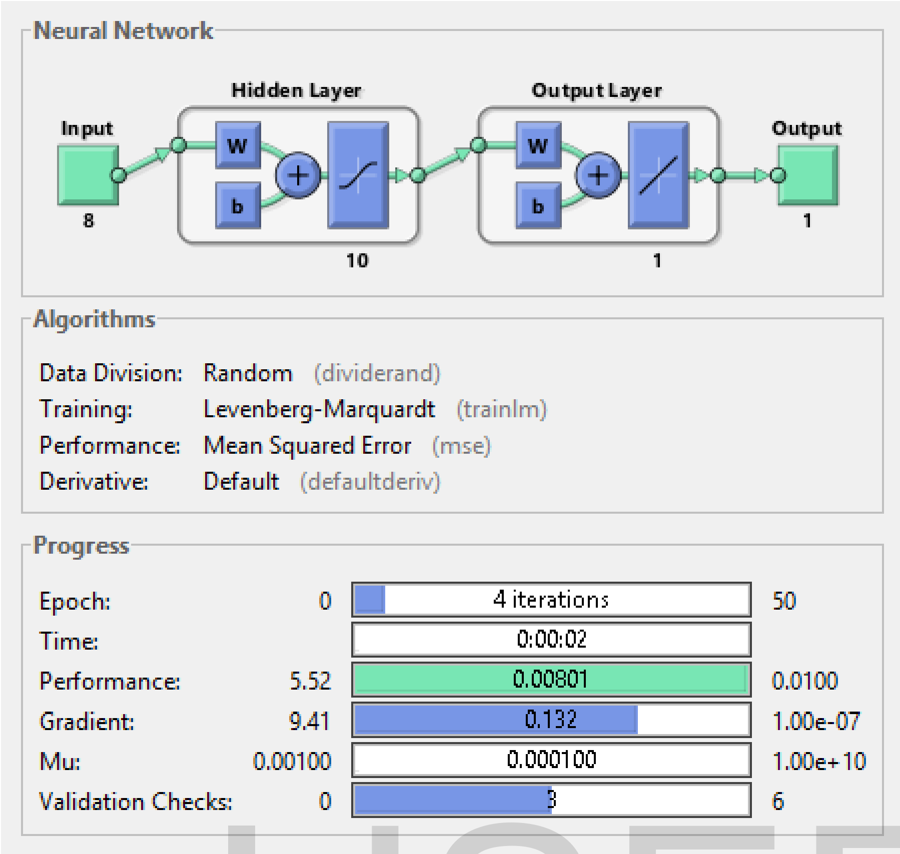




4.2 Results for 23 set of inputs

```
>> sos

y1 =

  Columns 1 through 12

    3.6854    2.5725    2.1629    2.1565    1.2275   -0.3318   -1.5920    5.3109    4.4130    4.0249    2.8262    0.7207

  Columns 13 through 23

   -1.0260   -2.7773   -4.0344    5.5938    5.8455    4.1926    1.5150   -0.0154   -1.3466   -2.0584   -2.6027


y2 =

    1.3381

>>
```

International Journal of Scientific & Engineering Research, Volume 7, Issue 4, April-2016
ISSN 2229-5518



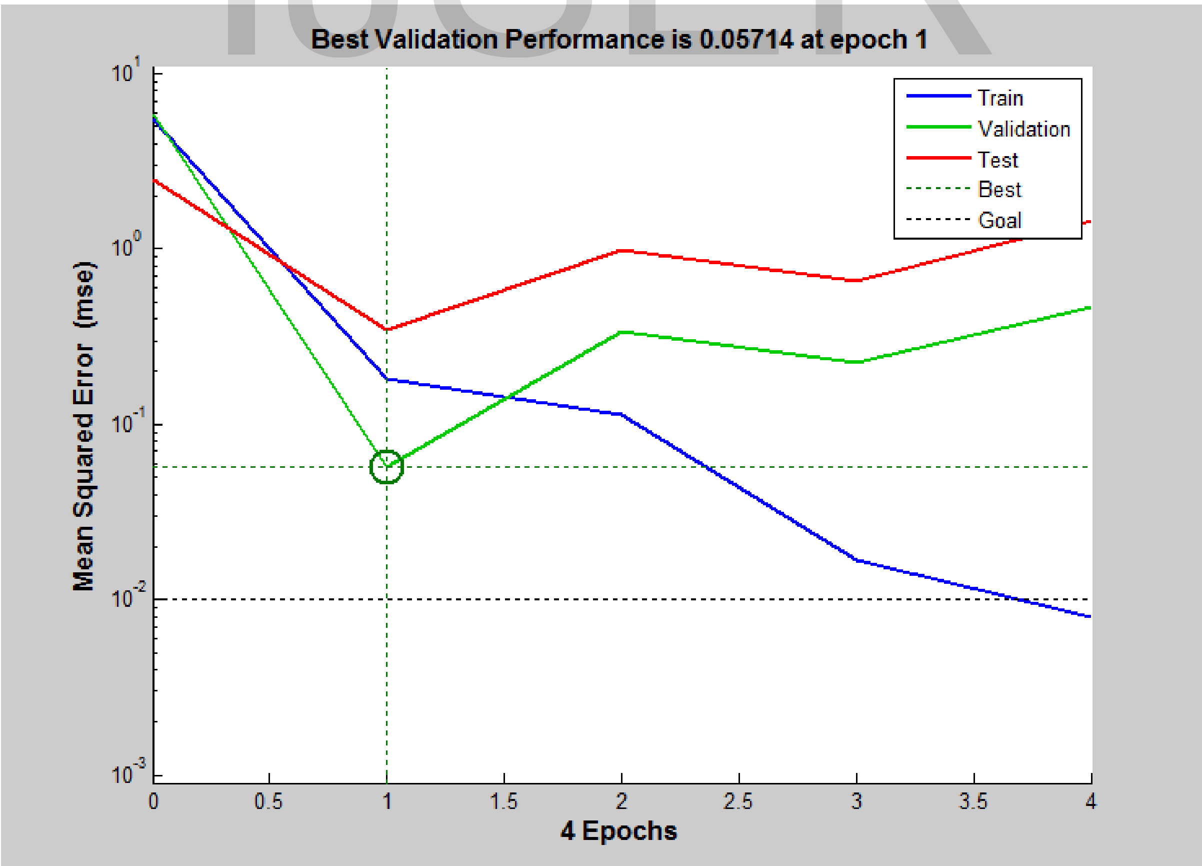


Table 5 Percentage of errors in the result

pu	b	d	Mu	Cover	Factored load	fck	fy	Percentage of steel		
								Desired value	Predicted value	% of error
350000	150	150	5250000	25	3500000	25	415	2.5	1.3381	-0.46476
350000	155	155	5250000	25	3500000	25	415	2	1.3381	-0.33095
350000	160	160	5250000	25	3500000	25	415	1.5	1.3381	-0.1079333
350000	165	165	5250000	25	3500000	25	415	1	1.3381	0.3381
350000	170	170	5250000	25	3500000	25	415	1	1.3381	0.3381
350000	175	175	5250000	25	3500000	25	415	0.5	1.3381	1.6762
350000	180	180	5250000	25	3500000	25	415	0.5	1.3381	1.6762
350000	150	150	7000000	25	3500000	25	415	2.5	1.3381	-0.46476
350000	155	155	7000000	25	3500000	25	415	1.5	1.3381	-0.1079333
350000	160	160	7000000	25	3500000	25	415	1.5	1.3381	-0.1079333
350000	165	165	7000000	25	3500000	25	415	1.5	1.3381	-0.1079333
350000	170	170	7000000	25	3500000	25	415	0.5	1.3381	1.6762
350000	175	175	7000000	25	3500000	25	415	0.5	1.3381	1.6762
350000	180	180	7000000	25	3500000	25	415	0.5	1.3381	1.6762
350000	185	185	7000000	25	3500000	25	415	0.5	1.3381	1.6762
350000	150	150	8750000	25	3500000	25	415	3	1.3381	-0.5539667
350000	155	155	8750000	25	3500000	25	415	2	1.3381	-0.33095
350000	160	160	8750000	25	3500000	25	415	2	1.3381	-0.33095
350000	165	165	8750000	25	3500000	25	415	1.5	1.3381	-0.1079333
350000	170	170	8750000	25	3500000	25	415	1	1.3381	0.3381
350000	175	175	8750000	25	3500000	25	415	1	1.3381	0.3381
350000	180	180	8750000	25	3500000	25	415	1	1.3381	0.3381
350000	185	185	8750000	25	3500000	25	415	0.5	1.3381	1.6762

5. Conclusion

Neural network model was used for short column under uni axial bending, Excel spread sheet is prepared for the design of column under uni axial bending and using the above spread sheet training data for ANN model was prepared, Mapping for input and output for ANN model is done, Desired output for percentage of steel is obtained using MATLAB, The percentage of error for the predicted values are less than zero for most of the inputs.

6. Acknowledgements

The authors are thankful to the Management of Karunya University for providing necessary facilities to carry out the work reported in this paper.

References

1. H.Sudarsana Rao, B.Ramesh Babu, "Optimized column design using genetic algorithm based neural networks" Indian journal of Engineering and Material science . 1 November 2006
2. N. Jayaramappa*, A. Krishna, B. P. Annpurna and T. Kiran " Prediction of Base Shear for Three Dimensional RC Frame Subjected to Lateral Load using Artificial Neural Network" Indian Journal of Science and Technology, *June 2014*
3. Abu-Bakr A. A. Aga1, Fathelrahman M. Adam2,3 "Design Optimization of Reinforced Concrete Frames "Open Journal of Civil Engineering, 2015, 5, 74-83 Published March 2015
4. Mohd Suhairil Meon1, Muhammad Azhan Anuar2, Mohd Hanif Mohd Ramli3, Wahyu Kuntjoro4,
5. Zulkifli Muhammad5 "Frame Optimization using Neural Network" international journals on science and technology
6. Application of Neural Network in Civil Engineering Problems" by D.-S. Jeng, D. H. Cha and M. Blumenstein in 2003.
7. Applications of Artificial Neural Network in Construction Engineering and Management by Megha Jaina KK Pathakb in August 2014
8. Counterpropagation Neural Networks in Structural Engineering" by Hojjat Adeli in Dec 2006.
9. Analysis of infilled frames – A Study using NN" N Muralikrishna, Dr. Gangadharan in 1999.
10. Predicting the life of concrete structures using neural networks" by N.R. Buenfeld & et. Al in 1998.
11. Neural Networks approaches to weight simple truss design problem by Hyeong – taek & C. John yoon in 1994