Nonlinear Regression of the transfer characteristics of electronic devices: a Neuro Computing Approach

S. N. Kale
Sr. Lecturer
Dept. of Applied Electronics
Sant Gadge Baba Amravati University
Amravati, Maharashtra

S.V.Dudul
Professor and Head
Dept. of Applied Electronics
Sant Gadge Baba Amravati University
Amravati, Maharashtra

ABSTRACT

In this paper, it is shown that Multilayer perceptron Neural Network can elegantly perform nonlinear regression of transfer characteristic of electronic devices. After rigorous computer simulations authors develop the optimal MLP NN models, which elegantly perform such a nonlinear regression. Results show that the proposed optimal MLP NN models have optimal values of MSE (mean square error), r (correlation coefficient) when it is validated on the and transistor non-linearity is observed in the transfer characteristics. The datasets are obtained by performing experiments on a typical p-n junction diode 1N4007, transistor BC107 and Field Effect transistor (FET) BFW10. The number of readings is treated as samples.

Optimal MLP NN (Multilayer Perceptron Neural Network) is developed for regression of electronic devices characteristics.

Other NN configuration Jordan Elman Neural Network has also been considered for this regression.

visual inspection of the plots that the outputs of the estimated MLP NN models closely follow the real one. It is seen that the performance of the proposed MLP NN models clearly outperforms the best Jordan Elman NN models. The simple NN models such as the MLP NN can be employed to solve such a nonlinear regression problem, is a major contribution of this research work.

Keywords

Regression, MLP NN, Jordan Elman neural network.

1. INTRODUCTION

Literature survey [1, 2, 3, 4] shows that Neural Networks (NN) have been effectively used for nonlinear regression. However, there is still enough scope to choose an appropriate NN model so that the performance measures are optimized to approach zero and unity for MSE (mean square error) and correlation coefficient (r), respectively. In regression, both the input data and desired response are experimental variables (normally real numbers) created by a single unknown underlying mechanism. The goal in regression is to find the parameters of the best linear approximation to the input and the desired response pairs. In nonlinear regression, conventional techniques such as least square approach generally do not work reasonably [5]. Therefore NN approach is worth considering for solving nonlinear regression problem [6]. In the electronic devices such as PN junction diode

This paper deals with the nonlinear regression using NN approach. Here datasets are obtained, which are used for regression. As shown in Table 1, there are 32, 33 and 121 training patterns for PN junction diode, transistor and FET respectively. For these datasets (80%) samples are used to train the NN models and (20%) different independent samples are used to assess the performance of estimated network models.

Table 1- Electronic devices datasets used for NN based Models

S. N.	Device type	No.of total samples	No. of training Samples	No. of cross validation Samples
1	P-N junction Diode	32	(80%)26(2 to 27)	(20%)06(28 to 33)
2	Transistor	33	(80%)26(2 to27)	(20%)07(28 to 34)
3	Field Effect Transistor	121	(80%)97(2 to98)	(20%)24(99 to 122)

Independent validation method in statistics is used to evaluate the NN in which the available data are divided into a training set and a cross validation (CV) set. The training data is used to update the weights, in the network. The CV data are then used to assess how well the network has generalized. The learning and generalization ability of the estimated NN model is assessed on the basis of performance measures such as MSE, NMSE (normalized mean square error) and correlation coefficient, r.

The network has been trained at least 15 times starting from different initial weights.

2. COMPUTER SIMULATION

2.1 MLP NN

MLP based NN model is used in this study because it has solid theoretical foundation. The main reason for this is its ability to model simple as well as very complex functional relationship. This has been proven through a large number of practical applications [7]. It is shown that all continuous function can be approximated to any desired accuracy in terms of the uniform norm with a network of one hidden layer of sigmoidal or hyperbolic tangent, hidden units as well as output unit [8]. MLPs are feedforward Neural Networks trained with the standard backpropagation algorithm. They are supervised networks so they require a desired response to be trained. The configuration of MLP NN is determined by number of hidden layers, number of the neurons in each of hidden layer as well as the type of activation function used for the neurons. Backpropagation algorithm with momentum is an improvement to the straight gradient-descent search in the sense that a memory term (the past increment to the weight) is used to speed up and stabilize convergence. Table 2 shows various parameters of the MLP NN models, which are varied for obtaining optimal parameters. PEs are varied from 1 to 20 for the hidden layers.

Optimal values of MSE, r-correlation coefficient are obtained with 4 PEs for transistor in the single hidden layer. For P-N junction diode best results are obtained using two hidden layers. First and second hidden layers contain 9 and 6 PEs respectively. In case of FET 6 PEs with single hidden layer is used. Different supervising learning rules are attempted such as momentum, conjugate gradient, quick propagation, Levenberg Marquardt, step and delta bar delta. In each case, a NN model is trained for fifteen times with different initialization of connection weights. The best results are obtained for Levenberg Marquardt learning rule in hidden as well as output layers. In addition 'linear', 'lineartanh' and 'tanh' transfer functions are varied in the output layer. The optimal values are found for linear transfer function for PN diode and transistor whereas in FET it is Tanh. For the MLP NN models, are used.

Table 2 – Variable parameters of MLP NN Model

supervised learning epochs= 1000, Error threshold = 0.01, Transfer function in hidden layer= tanh, No. of PEs in input layer = 01, No. of PEs in output layer =1

	Parameter	Typical Range	Optimal parameter		
			PN junction Diode	Transistor	FET
1	Hidden Layer	1 to 4	**2**	**1**	**1**
2	PE	1 to 20	**Hidden I- 9** **HiddenII-6**	**4**	**6**
3	Learning Rule	Momentum (Mom), Conjugate gradient (CG), Levenberg Marquardt (LM), Quick propagation (QP), Step, Delta bar delta	**LM**	**LM**	**LM**
4	Transfer Function in output layer	Linear, Lineartanh, Tanh	**Linear**	**Linear**	**Tanh**

For transistor dataset the performance measures are found better for single hidden layer and that of for PN junction diode it is for two hidden layers as shown in Table 3. With increase in number of hidden layers the performance of the network has not improved significantly for transistor. In case of P-N junction diode two hidden layers give optimal performance.

Table 3– Number of Hidden layer and r

No. of Hidden layer	r		
	P-N junction diode	Transistor	FET
1	0.9926	**0.9929**	**0.9748**
2	**0.9966**	0.9787	0.9707
3	0.9894	0.9908	0.9685
4	0.9742	0.9874	0.9546

The number of epochs for training the datasets is also varied to obtain optimum response. Table 4 illustrates that optimum performance for 1000 epochs, as there is no significant improvement in performance in case of P-N junction diode whereas in transistor dataset there is deterioration

Table 4 – Number of epochs and performance measures

S.N.	No. of epochs	r		
		P-N Junction Diode	Transistor	FET
1	1000	**0.9966**	**0.9929**	**0.9748**
2	2000	0.9955	0.9732	0.9749
3	3000	0.9931	0.9898	0.9743
4	4000	0.9967	0.9731	0.9254
5	5000	0.9954	0.9880	0.9745

Figure 1,2 and 3 give regression capability of MLP NN on cross validation datasets of P-N junction diode, transistor and FET respectively**,** which portray desired output and actual output of the MLP NN on cross validation data set. It is seen that actual output follows the desired output very closely.

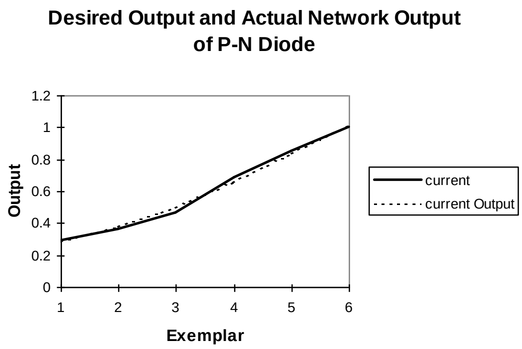

Fig. 1 Regression capability of MLP NN on cross validation Dataset of P-N junction Diode [r=0.9966]

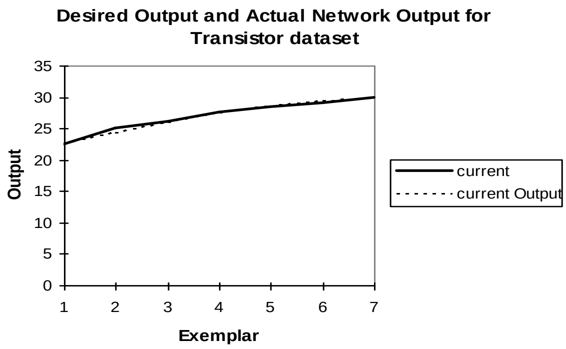

Fig. 2 Regression capability of MLP NN on cross validation Dataset of Transistor [r=0.9929]

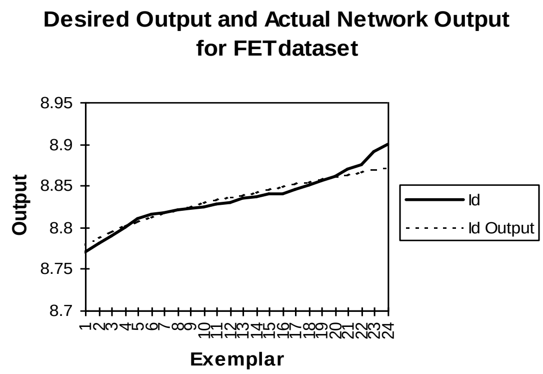

Fig. 3 Regression capability of MLP NN on cross validation Dataset of FET [r=0.9948]

2.2 Jordan Elman Neural Network

Recurrent networks are neural networks with one or more feedback loops. The Recurrent networks are used as input-output mapping networks and also as associative memories [9]. By definition, the input space of a mapping network is mapped onto an output space, a recurrent network responds temporarily to an externally applied input signal. Recurrent networks can be considered as dynamically driven recurrent networks. Because of global feedback memory requirement reduces significantly [10].

Jordan and Elman networks extend the multilayer perceptron with context units, which are processing elements (PEs) that remember past activity. Context units provide the network with the ability to extract temporal information from the data.

Table 5 shows various parameters of the Jordan Elman NN models, which are varied for obtaining optimal parameters. Here also different supervising learning rules are attempted. It is found that the best results are obtained for Levenberg Marquardt learning rule in hidden as well as output layers. Transfer functions are varied in the output layer and optimal parameters are found for linear transfer function.

Table 5 – Variable parameters of Jordan Elman NN Model

S.N.	Parameter	Typical Range	Optimal parameter		
			P-N Diode	Transistor	FET
1	Learning Rule	Momentum (Mom), Conjugate gradient (CG), Levenberg Marquardt (LM), Quick propagation (QP), Step, Delta bar delta	**Levenberg Marquardt**	**Levenberg Marquardt**	**Quick propagation**
2	Transfer Function in output layer	Linear, Lineartanh, Tanh	**Linear**	**Linear**	**Tanh**
3	Context Unit Transfer Function	Integrator Axon, Tanh Integrator Axon, Sigmoid Integrator Axon, Context axon, Tanh Context axon, Sigmoid Context axon	**Integrator Axon**	**Integrator Axon**	**Sigmoid Integrator Axon,**
4	Context Unit time constant	0.1 to 0.9	**0.6**	**0.7**	**0.8**

In Jordan Elman NN, context unit transfer functions are varied for the optimal performance and it is found for integrator axon. Time constants are varied from 0.1 to 0.9 and optimal performance is obtained with time constant 0.8 for FET whereas for P-N junction diode and transistor it is optimal for 0.6 and 0.7 respectively. For the given dataset Jordan Elman NN model is trained for five times. Fig 4,5 and 6 depict regression capability of Jordan Elman NN on cross validation dataset for all the three devices. It is seen that actual output barely follows the desired output.

For the given dataset MLP NN model is trained for five times. The performance measures such as MSE, NMSE and r on training dataset and cross validation dataset are obtained

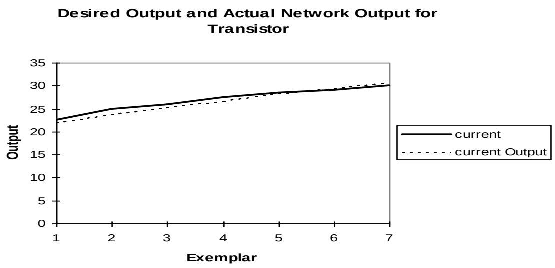

Fig. 4 Regression capability of Jordan Elman NN on CV Dataset for PN junction diode [r=0.9966]

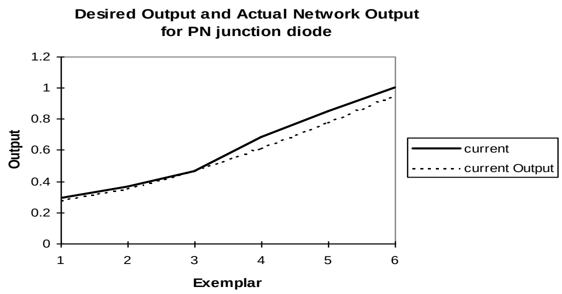

Fig.5 Regression capability of Jordan Elman NN on CV Dataset for Transistor[r=0.9872]

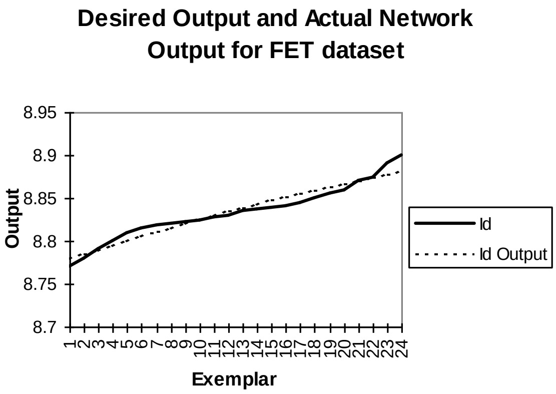

Fig. 6 Regression capability of Jordan Elman NN on CV Dataset for FET[r=0.9687]

The optimal performance of architectures of every NN for every device is shown in Table 6. The optimal performance is obtained for MLP NN.

For FET, it is observed that for MLP NN the time required for training the network per epoch per exemplar is 20.5 microseconds. Connection weights (free parameters) are 7 for all the NN architectures.

In case of PN junction diode, free parameters are 85 and time elapsed per epoch per exemplar is 884 microseconds, which is largest amongst all the NNs. However the percentage error is 4.6% that is the least for MLP NN.

When transistor dataset is considered, free parameters are 13 and time elapsed per epoch per exemplar is 38.5 microseconds. For MLP NN 0.87% error is the least amongst all other NNs.

Table 6: Comparison of all the NN Architectures on cross validation dataset of the three devices

P= No. of connection weights, N= Total no. of exemplars in training dataset

S.N.	Device Type	NN	% Error	Time elapsed /epoch/ Exemplar in microseconds	N/P
1	PN junction Diode 1N4007	MLP (1-9-6-1)	4.6	884	0.31
		Jordan Elman (1-4-1)	5.5	38.5	2
2	Transistor BC107	MLP (1-4-1)	0.87	38.5	2
		Jordan Elman (1-4-1)	2.5	38.5	2
3	FET	MLP (1-6-1)	0.09	20.5	0.196
		Jordan Elman (1-4-1)	0.07	10.3	0.134

3. CONCLUSION

Results show that a MLP NN is able to solve Nonlinear Regression of the transfer characteristics of electronic devices problem with sensible accuracy. When the performance of MLP and Jordan Elman NN based regression are carefully examined for PN junction diode, transistor and FET data sets, MLP NN has clearly outperformed its Jordan Elman NN counterpart with respect to the performance measures such as MSE, NMSE, and r. Moreover, from visual inspection of graphs the actual output of the estimated MLP NN model follows the desired output more closely when compared with other NN models. It is also seen that the time elapsed per epoch per exemplar is sensibly less for MLP NN amongst all the networks except in PN junction diode. In case of transistor the time elapsed per epoch per exemplar is same for both the NNs. For PN junction diode it is 22.96 times higher in MLP NN model than that of Jordan Elman NN model. In FET, the time elapsed per epoch per exemplar for MLP NN model is 1.99 times than that of Jordan Elman NN model. The MLP NN models for all the three devices have achieved the least percentage error on cross validation dataset. Ratio of number of exemplars to in training dataset to number of connection weights is higher for MLP NN among all devices except PN junction diode. This ratio gives the complexity of NN. Proposed MLP NN models are able to accomplish the

Nonlinear Regression of the transfer characteristics of electronic devices.

4. REFERENCES

[1] S. I. Wu and H. Zheng, Stock Index Forecasting using Recurrent Neural Networks. *Proc. The IASTED Conference on Artificial Intelligence and Applications*, Innsbruck, Austria, 2006

[2] Richard D. de Veaux, Jennifer Schumi, Jason Schweinsberg, Lyle H. Ungar, Prediction Intervals for Neural Networks via Nonlinear Regression, *Technometrics*, Vol. 40, No. 4, pp. 273-282, Nov.1998

[3] Abderrahmane Amrouche, and Jean Michel Rouvaen, Efficient System for Speech Recognition using General Regression Neural Network, International Journal of Intelligent Systems and Technologies 1;2 Spring 2006

[4] Seyed Mehdi Sadat-Hashemi et. al. ,Predicting the type of pregnancy using artificial neural networks and multinomial logistic regression: a comparison study, *Journal-Neural computing and* applications,*Volume 14*, Number 3, pp 198-202, Springer London, September, 2005

[5] Ham M. Fredric, Ivica Kostanic, *Principles of neurocomputing for science & engineering* (Tata McGraw Hill)

[6] Haykin S., *Neural Networks: A Comprehensive Foundation*, Pearson Education.

[7] Demuth H. and Beale M. *Neural Network Toolbox: User's Guide, Version 3.0* (The MathWorks, Inc., Natick, MA,1998)

[8] Cybenko G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control,Signals, and Systems*, 2(4), 303-314, 1989.

[9] [Principe J.C., Euliano N Lefebvre W.C. *Neural & Adaptive Systems: Fundamentals Through Simulations* (John Wiley & Sons,Inc.) 2000

[10] Quin S. Z., Su H. T., and McAvoy T. J. Comparison of Four Neural Net Learning Methods for Dynamic System Identification. *IEEE Transactions on Neural Networks*, vol. 3, no. 1, pp. 122-30, 1992