Characteristic Prediction of Wind Tunnel tests using Learning from Examples

Dr. R.Malmathanraj
Assistant Professor
Department of ECE,
National Institute of
Technology, Tiruchirappalli

T. Francis Tyson
UG Final Year Student
Department of ECE,
National Institute of
Technology, Tiruchirappalli

ABSTRACT
This paper presents a systematic neural network approach based on the concept for Learning from Examples for the prediction of aerodynamic characteristics from the Wind tunnel test data. Aerodynamic coefficients are modeled as functions of angle of attack, Normal force coefficient, Mach number, and Lift force Coefficients. The training data which is fed as the input to the neural network is derived from wind tunnel test measurements and numerical simulations. In this paper, a comparative study of the efficiency of neural network prediction based onLFE(Learning from Examples) for different architectures and training dataset sizes is presented. The results of the prediction reflect the sensitivity of the architecture and training dataset size. For a training set of 136 data points and a training set with Mach number ranging from 0.6 to 3, the Generalized Regression Neural network(GRNN) constantly out-performed the Radial basis function neural network and Backpropagation network regression model in time effectiveness. The objective of this paper is to demonstrate that the neural network approach based on the concept of learning from examples is a fast and reliable way for predicting aerodynamic coefficients.

General Terms
Prediction of Aerodynamic Characteristics

Keywords
Wind tunnel test; Radial basis function neural network; Back propagation neural network; Mach number

1. INTRODUCTION
Aerodynamics is the study of different forces acting on an object and its resulting motion through the air. Aerodynamicists characterize flight speed in terms of Mach number and the aerodynamic characteristics depend on individual aircraft design variables. Wind tunnels play a major role in the design and development of space vehicles. It was in 1871, that the first working Wind tunnel was designed by Frank H. Wenham. A Wind tunnel test program was undertaken to define the stage separation aerodynamic environment. The Wind tunnel apparatus is for studying the interaction between a solid body and an air stream. A Wind tunnel simulates the conditions of an aircraft in flight by causing a high speed stream of air to flow past a model of the aircraft being tested. The model is mounted on wires so that lift and drag forces on it can be measured by measuring the tensions in the wire. In the Wind tunnel test the Mach number is an important factor. The graph is plotted for angle of attack (alpha) versus coefficient of frictional drag coefficients (CDF) with the given Mach number as constant[1].

Nowadays the neural network approach to data analysis has received much attention. Neural networks have overcome the theoretical limitations of perceptrons and early linear networks by the introduction of “hidden layers" to represent intermediate processing and to compute nonlinear recognition functions[2].They moreover learn quickly in discriminating amongst equivalent classes of patterns in a holistic manner,within an input domain. They are presented with training sets of representative instances of each class, correctly classified, and they learn to recognize and predict other new instances of these classes. Learning is the phenomenon of readjusting weights in a fixed-topology network via different learning algorithms.

The neural network has been used quite successfully in various engineering and business applications such as analysis of appendicitis and cancer patient data[3], cancer image extraction and classification[4]; studies of soybean diseases[5]; and in pharmaceutical applications such as pharmaceutical production development[6], pharmacodynamic modeling[7], and pharmacological effects of drug concentrations[8].

This paper analyses the neural networks for graph prediction. Aoyoma et al.[9] presented an application of the neural network approach to estimating quantitative structure-activity relationships. The neural network model has always performed better than linear multiple regression analysis. Backpropagation networks (BPN) which are based on fully connected, layered, feedforward networks, in particular, have demonstrated the desirable properties of self-learning, noise-tolerance, and good predicting power. Bodor et al.[10] experimented with a Backpropagation network for solubility prediction. The research showed regression analysis technique to be inferior to the Backpropagation model in mean standard deviation for the training set. However, the neural network as well as the regression technique did not perform for an unknown set of organic compounds.

2. AERODYNAMICS

2.1 Forces of Aeronautics

There are four primary forces that act on an airplane in flight. They are Thrust, Weight, Drag and Lift as shown in Fig 1.

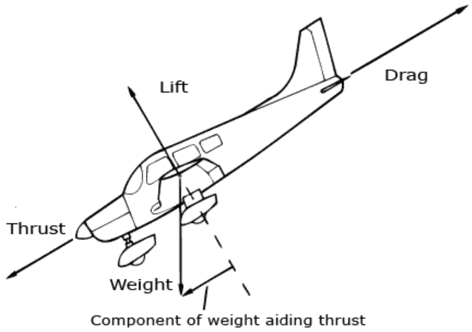


Figure 1. Aerodynamic Forces

Weight is the force that measures the effects of gravity. Force generated for takeoff must be stronger than the weight force. This force is called lift. Thrust propels an object in a particular direction. Drag is the force that resists any object trying to move through a fluid.

A method of computational simulation of the aerodynamic- or hydrodynamic-flow performance features of objects involves the use of neural-network mathematical models implemented in computer hardware. The method can be applied in conjunction with wind-tunnel, water-tunnel, or water-trough testing of scale models of such diverse objects as aircraft or parts of aircraft, sails, fins, turbine blades, and boat hulls. In the case of an aircraft, for example, a neural network can be trained from (1) test input signals (e.g., positions of control surfaces; angle of attack; angles of roll, pitch, and yaw; power settings; and airspeed) and (2) test output signals (e.g., lift, drag, pitching moment, and/or other performance features). In general, the relationships between the input and output variables are nonlinear. The present method harnesses the ability of neural networks to learn nonlinear relationships between input and output variables.

The coefficients present in the graphs of aerodynamics for Wind tunnel test are Missile Identification Number, Mach number, Dynamic Pressure, Alpha, Control Deflection, Normal Force Coefficient, Side Force Coefficient, Lift Force Coefficient, Frictional Drag Coefficient (CDF), Pitching Moment, Yawing Moment Coefficient, Rolling Moment Coefficient, Base Pressure Coefficient, and Base Drag Coefficient.

The predicted aerodynamic characteristics are:

i. Alpha vs. Frictional Drag Coefficient (CDF)

ii. Alpha vs. Pitching Moment Coefficient

iii. Normal Force Coefficient vs. CDF

iv. Normal Force Coefficient vs. Pitching Moment Coefficient

v. Alpha vs. Lift Force

vi. Normal Force Coefficient vs. Lift Force Coefficient

vii. Alpha vs. Normal Force Coefficient

A chamber through which air is forced at controlled velocities in order to study the effects of aerodynamic flow around airfoils, scale models, or other objects is called a Wind tunnel. Wind tunnels can be divided by several characteristics. Besides the classical types, Eiffel- and Göttinger wind tunnels, several special tunnels, e.g. tailspin and shoot tunnels, exist. Further classification is the type of measuring section, such as the form of the cross section and whether it has an open or closed test section. The Wind tunnel has a Settling Chamber, Contraction Cone, Test Section, Diffuser and Drive Section.

The purpose of the settling chamber is to straighten the airflow. The contraction cone takes a large volume of low velocity air and reduces it to a small volume of high velocity air without creating turbulence. The test section is where the test article and sensors are placed. The main forces to be measured in the test section are Lift and drag. The diffuser slows the speed of airflow in the Wind tunnel. The drive section provides the force that causes the air to move through the Wind tunnel.

To study pressure, velocity distributions around bodies, the Wind tunnel can be made use of to make modification to the body to obtain the aerodynamic forces experienced by body. Aircraft, spacecraft, rockets, cars, trucks and buildings can be tested in Wind tunnels.

3. NEURAL NETWORKS

Characterization of aerodynamic coefficients of an air vehicle has generally been based on wind tunnel tests of scaled models. But this data in its novel form is generally unsuitable in piloted situations since there is no consistency between the data obtained in case of different scale models of the different wind tunnels. Conventionally, fitting of a polynomial function for each aerodynamic coefficient is what was performed. The reason being,iteliminated the scatter from the measurements due to the smooth function[11].

Fitting this smooth function provided smooth derivatives of the data which are crucial in performing stability analyses. In addition,some means of reconciling dissimilar sets of raw data is needed since measurements of the same coefficient from two dissimilar wind tunnels are usually taken at disparate values of angle of attack and lift force coefficient.

Now to collect such amounts of data from numerical simulations or wind tunnel tests is quite expensive; mainly due to the complex model fabrication, intensive power utilization and the high personnel overload. Both steady and unsteady data are required fornumerical simulation of complex vehicles. With simple Euler codes steady aerodynamic data can be obtained at low Mach numbers and angles of attack. In case of a vehicle separation of flow zones Navier-Stokes simulations are required which are costly due to the large processing time required for convergence. Similarly in case of unsteady data, dynamic coefficients can be precisely predicted by using numerical simulations. Yet the advantage of the neural network lies in its ability to combine data from both numerical and experimental simulation to create an efficient archive database. Overall the neural network performs very well in a classification and prediction arena within the data range being utilized during the training phase. The input data for training the neural network can come from either Navier-Stokes simulations or wind tunnel test measurements, or a combination of both.

This research shows the prospective for use of neural networks as a useful analytical technique for graph prediction. The intention behind using neural networks in graph prediction is to have the neural system learn to model a relationship that is represented explicitly in a set of historic data. Thus the objective is to predict a new aerodynamic characteristics graph by performing both interpolation and extrapolation. Several factors influence the performance of such decision support systems.

Simply put different sets of training data produce models withvery different generalization accuracies.

The performance of feed-forward neural network trained with the Backpropagation algorithm and Radial basis function neural network for graph prediction on a wind tunnel test data is analyzed. The neural network techniques is applied for predicting a graph with a new Mach number purely from observing the raw input graph data with a variety of Mach numbers. A new model is thus developed for predicting graphs in Wind tunnel test data. The neural networks used for analysis include Radial basis function (RBF) neural networks and Back propagation neural networks.

The structure of an ANN is basically a topology consists of three layers: input, hidden, and the output layer. The input layer is for extracting knowledge from the environment. The output layer is for the communication with the environment. The hidden layers are responsible for the execution between these two layers. Neurons are connected with unidirectional paths they communicatewith each other in the same layer or in different layers.

4. GRAPH PREDICTION USING LFE

The objective here is to predict the aerodynamic characteristics by using the concept of Learning from examples(LFE). The first step in building a neural network application to predict the aerodynamic characteristics was to develop a test data set for the neural network to use in training itself.

4.1 Defining Inputs and Outputs

This section discusses about the input and output variables. The learning from examples concept (Appendix A) is used for predicting the aerodynamic characteristics. The input variables used are the graphs of Angle of Attack vs. CDF for specific Mach number. Output variables are the predicted characteristic graphs for new Mach number. The samples are provided in sequence as explained in the Fig diagram for the effective implementation of Learning from Examples. The first training pattern consists of all the alpha values and the corresponding Mach number. The one complete presentation of all the training patterns is called one epoch. The Mach number and all the corresponding alpha values are given as input to the neural network. After sufficient presentation of input patterns the network predicts the complete characteristics graph for any Mach number.

$$[\text{Mach Number}][\text{Alpha,NormalForce Coefficient}]^T = [\text{Frictional Drag Coefficient}]^T$$

$$\text{Frictional Drag Coefficient (CDF)} = f(\text{alpha, Mach Number}) \text{ , \&}$$

$$[\text{Frictional Drag Coefficient(CDF)} = f(\text{Normal Force Coefficient, Mach Number})$$

In the neural network design the inputs are alpha, Normal Force Coefficient, Normal Force Coefficient, Mach number and the output is Frictional Drag Coefficient (CDF), Pitching Moment Coefficient and Lift Force.

4.2 Architecture Selection

A variety of neural network architectures are available to process the data from the input data set files. A multi-layerBackpropagation architecture, Radial basis function and Generalized Regression neural network are used for training because of their ability to generalize well when applied to a wide variety of applications and also due to their better regression.

5. IMPLEMENTATION ISSUES

A test data set containing 136 data points for Mach numbers 3.5,3, 2.5,1 and 0.6 was randomly extracted by the neural network to compute average training error used to determine when to stop training. A training set containing all the Mach numbers data was used for network learning.

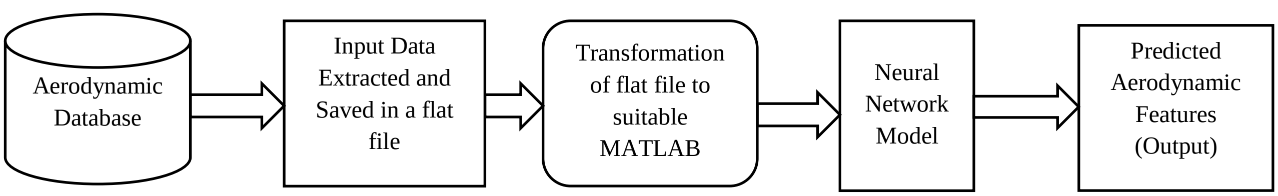


Figure 2. Architecture for graph prediction

The test data set is extracted from the ORACLE Database. But to train and test the different graphs with the Mach numbers given above with Artificial Neural Network the data is converted into sample inputs as (*.dat) file format. A simple Backpropagation network of two input nodes (for the graph parameters), three hidden layers (18, 28, 10 neurons) and one output node, a RBN neural network and GRNN was developed.

Inputs for the project are the learning set of data obtained from the flight manual. The following decisions regarding the neural network were also required as inputs:

i. The number of inputs.
ii. The value for the learning coefficient.
iii. The number of processing elements in the hidden and output layers.

iv. The number of cycles for each run.

MATLAB was used as the modeling language to implement the neural network. In this MATLAB functions in the neural net toolbox will be employed. For example, *initff* initializes a feed-forward model; *trainlm* trains a network using Levenberg-Marquardt Algorithm. Other transfer functions will be employed in this section as well.

Radial basis function networks (RBF) is a type of artificial network for applications to problems of supervised learning e.g. regression, classification and time series prediction. Radial basis function networks are non-parametric models. By non-parametric models, it means that there is a priori knowledge about the function that is to be used to fit the training set. An example of a parametric model would be fitting a straight line to a set of points. The form of the function a straight line is known and it is just a matter of best fitting the line to the training set. RBF networks can be used to solve regression problems.

This project effectively connects the Java front end design, Oracle database and MATLAB. The following steps were performed to connect Java with Oracle:

i. Importing Packages

ii. Registering the JDBC Drivers

iii. Opening a Connection to a Database

iv. Creating a Statement Object

v. Executing a Query and Returning a Result Set Object

vi. Processing the Result Set

vii. Closing the Result Set and Statement Objects

viii. Closing the Connection

In this project we use MATLAB for Neural network and to plot graphs. Matlab is a software package for high performance numerical computation and visualization. It provides easy extensibility with its own high level programming language.

Steps to connect Java with MATLAB

i. Importing Packages

ii. Registering the Matlab Engine

iii. Opening a Connection to a MATLAB

iv. Executing Result Set

In this project the GUI design has the Model, X-axis, Y-axis, Attributes, Value, &these values are retrieved from the database. The training module can be selected from the GUI. After retrieving the data from the Database, the data are stored in the files and are trained with suitable decision module. The plot to show the graphs predicted for Mach number 0.6, 1 and 2.5 by RBF neural network together with statistics is shown in Fig. 11.

A practical problem that occurred during training is that in the BPN Neural Network small changes in the training data set may produce very different models and consequently different performance on unseen data. In this paper we show that this instability means that estimations of the generalization performance of an ANN for a particular task may vary considerably depending on the training data used.

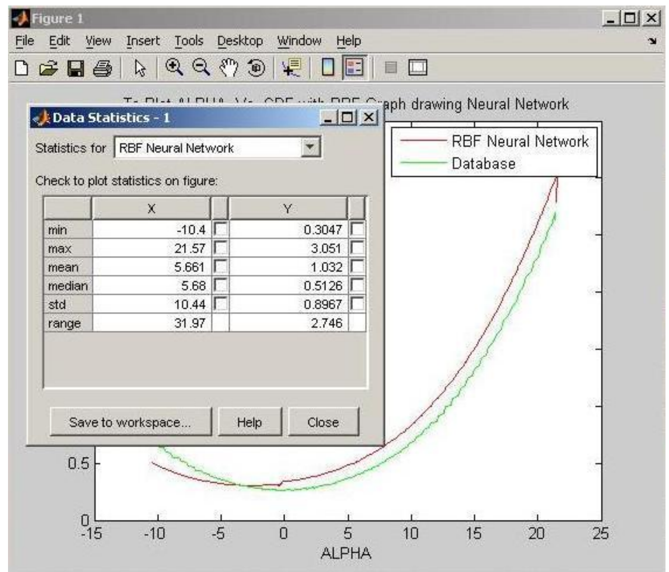


Figure 3. Plot to show the comparison between graph predicted by neural network and graph drawn from database

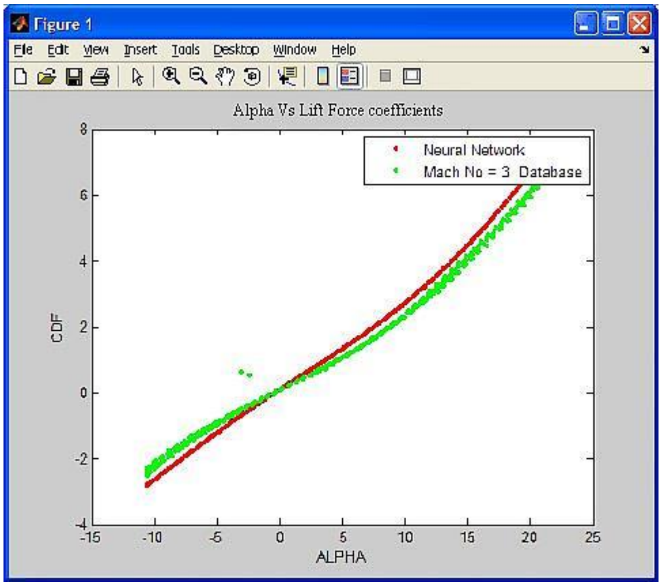


Figure 4. Alpha vs. Lift Force Coefficients

It can be seen that as the training of the network proceeds the error on the training data continues to drop but after 200 epochs (200 presentations of all the training data) the error on unseen test data starts to rise. After this point the network is overfitting to peculiarities in the training data and is losing generalization accuracy.

Up to a certain point additional training data will produce appreciable increases in accuracy. However, beyond the knee point in the graph additional data produces little increase in accuracy. At the knee point the learning system has seen a useful cross section of data samples that represent a good coverage of the problem domain.

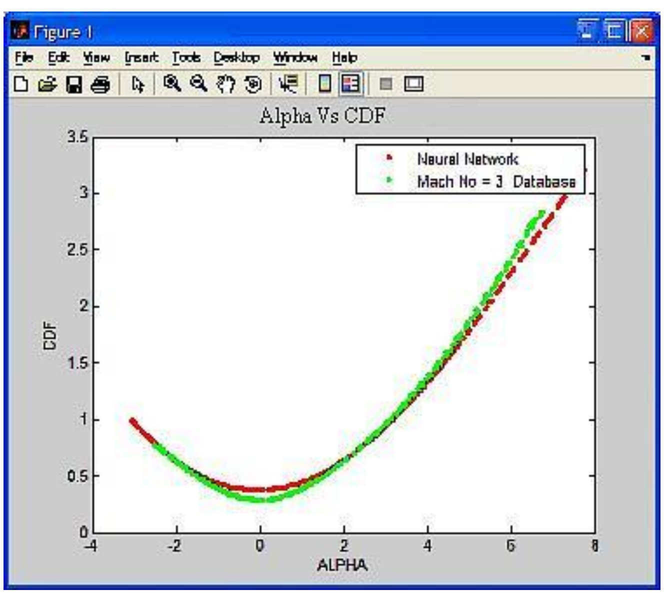
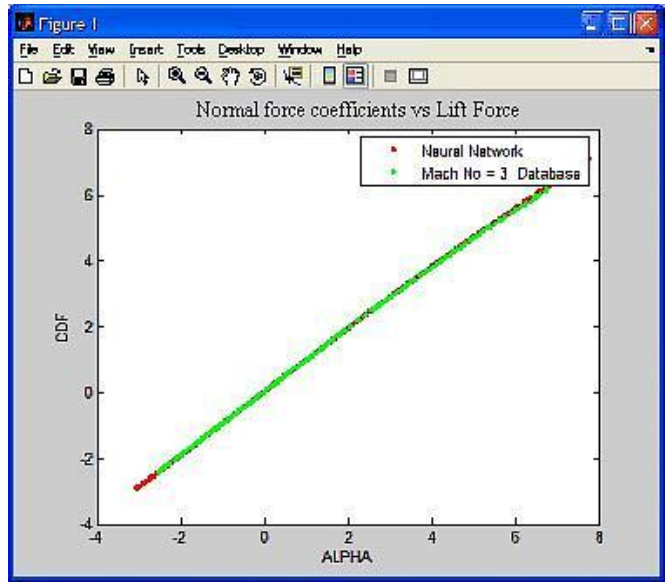


Figure 5. Alpha vs. CDF

Figure 6. Normal Force Coefficient vs. Lift Force



The solution to this problem is to hold out some of the available data from training and stop training when error on this validation set starts to rise. In situations where an abundance of training data is available, all the details of the problem will be well represented in the training data and overfitting is unlikely to be observed.

The tabulation to show the efficiency of neural networks in predicting the performance characteristics of aerodynamic database in case of GRNN, BPN and Radial basis function is shown in Tables 1,2& 3. The tabulation calculates the Deviation between the values from the Wind tunnel test database and the values predicted from the Neural network.

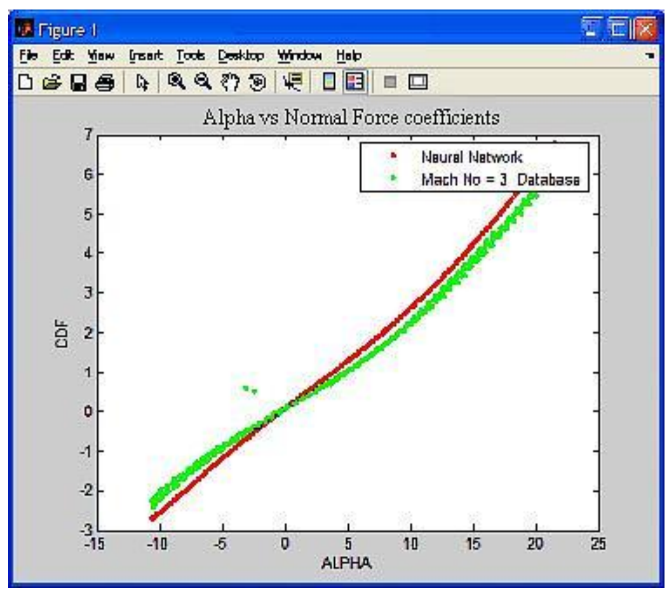


Figure 7. Alpha vs. Normal Force Coefficients

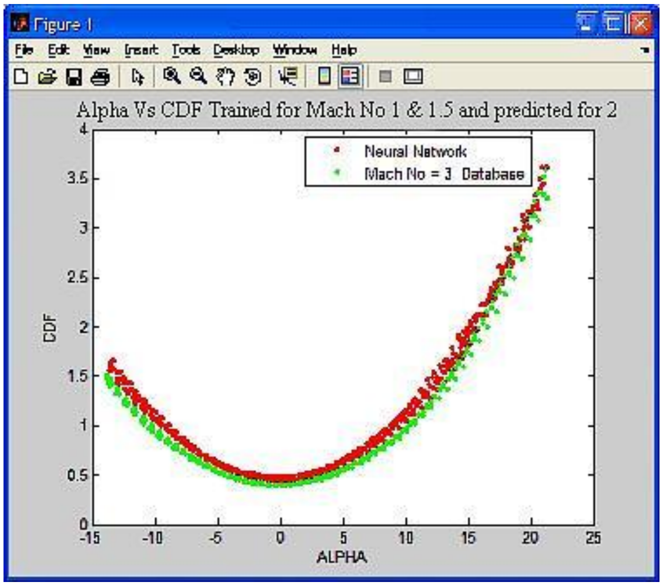


Figure 8. Alpha vs. CDF trained for Mach 1 & 1.5 and predicted for 2

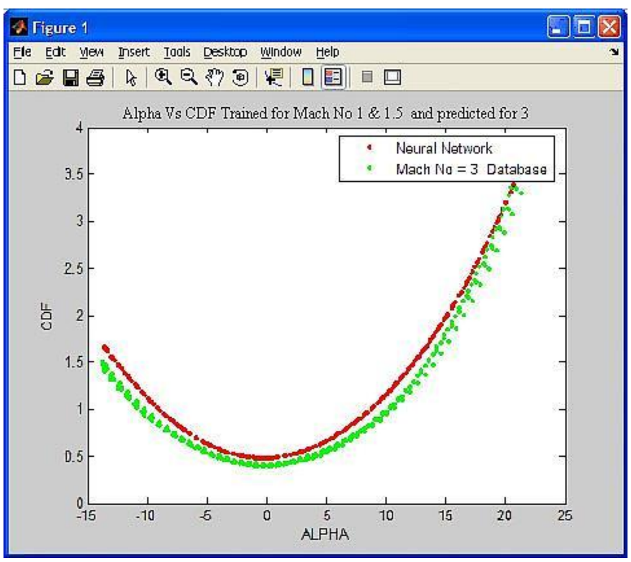


Figure 9.Alpha vs. CDF trained for Mach 1 & 1.5 and predicted for 3

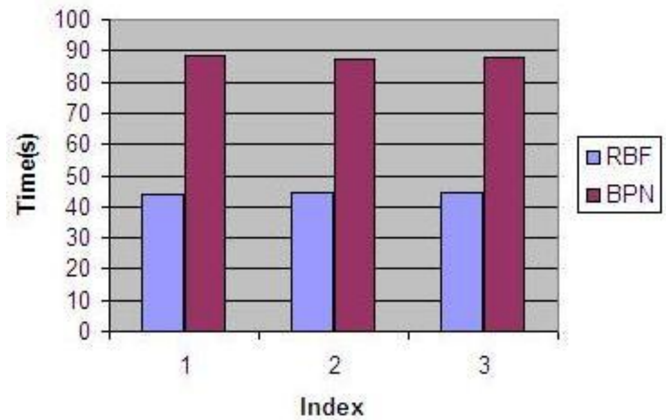


Figure 10. Plot to show time for Convergence

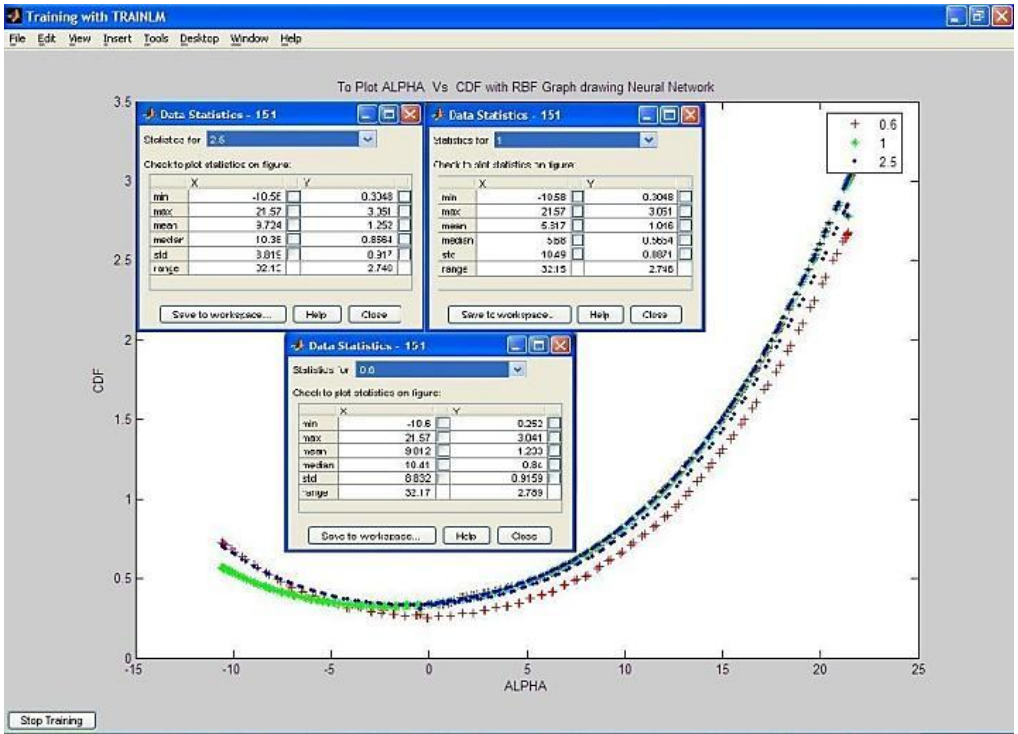


Figure 9. Alpha vs. CDF trained for Mach 1 & 1.5 and predicted for 3

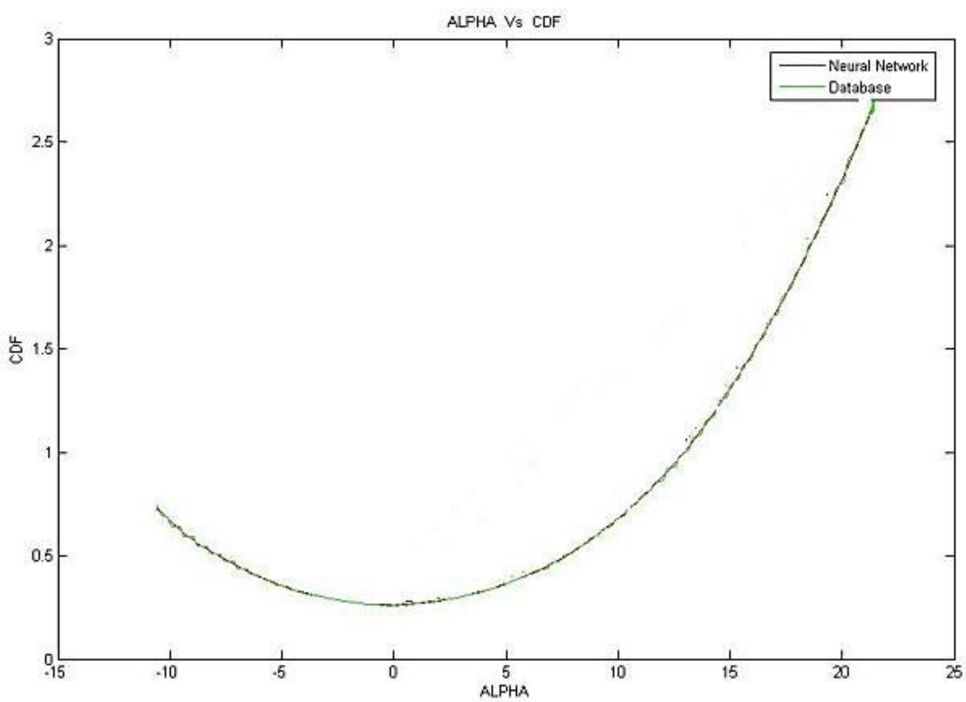


Figure 12. Plot to show the sample graph drawn during the learning trial of BPN neural network (with rectification)

Table 1. Tabulation showing the deviation between the values predicted by BPN neural network and drawn from database

Sl. No.	PredicitonCharacteristics	BPN											
		WTT Max Deviation		ANN Max Deviation		Difference in Max Deviation		Cumulative Difference in Max Deviation		AverageDifference		% Deviation	
		X	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**
1	Alpha vs Normal Force	24.01	7.92	25.12	29.147	0.4	0.71	0.39	0.33	0.001	0.025	0.1728	0.1785
2	Alpha vs Pitching Moment Coefficient	10.6	20.58	10.32	28.76	0.22	8.87	0.47	7.895	0.1373	2.7110	0.0017	0.5816
3	Normal Force Coefficient vs CDF	6.763	2.93	7.001	2.902	0.278	0.018	0.199	0.018	0.2541	0.1708	0.0399	0.019
4	Normal Force Coefficient vs Pitching Moment Coefficient	2.312	21.36	2.256	24.36	0.004	5.12	0.011	9.589	0.362	2.9653	0.0368	0.0352
5	Alpha vs Lift Force	8.3	2.1	9.65	2.123	0.01	0.372	0.32	1.653	0.1758	0.3045	0.0019	0.0254
6	Normal Force Coefficient vs Lift Force Coefficient	6.73	6.14	6.80	6.24	0.044	0.09	0.048	0.10	0.275	0.2045	0.040	0.399
7	Alpha Vs CDF	10.6	0.724	10.58	0.899	0.02	0.19	0.017	0.14	0.1356	0.174	0.00141	0.0115

Table 2. Tabulation showing the deviation between the values predicted by RBF neural network and drawn from database

Sl. No.	PredictionCharacteristics	RBF											
		WTT Max Deviation		ANN Max Deviation		DIFF in Max Deviation		Cumulative Diff in Max Deviation		AvgDiff		% Deviation	
		X	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**
1	Alpha vs Normal Force	21.42	6.49	20.92	7.089	0.5	0.59	0.5	0.59	0.0020	0.0378	0.1947	0.2146
2	Alpha vs Pitching Moment Coefficient	10.6	20.23	10.48	27.95	0.12	7.72	0.12	7.72	0.1947	2.2388	0.0020	0.0475
3	Normal Force Coefficient vs CDF	6.763	2.831	6.998	2.809	0.2350	0.022	0.235	0.022	0.2166	0.1501	0.0414	0.0384
4	Normal Force Coefficient vs Pitching Moment Coefficient	2.531	22.41	2.526	25.49	0.005	3.08	0.013	8.328	0.2166	2.2685	0.0414	0.0499
5	Alpha vs Lift Force	10.6	2.25	10.58	2.714	0.02	0.464	0.0404	1.218	0.1947	0.2004	0.0020	0.0391
6	Normal Force Coefficient vs Lift Force Coefficient	6.76	6.15	6.81	6.26	0.05	0.11	0.05	0.11	0.2166	0.2022	0.0414	0.0414
7	Alpha Vs CDF	10.6	0.724	10.58	0.894	0.02	0.17	0.02	0.17	0.1947	0.1468	0.0020	0.0122

Table 3. Tabulation showing the deviation between the values predicted by GRNN neural network and drawn from database

Sl. No.	PredictionCharacteristics	GRNN											
		WTT Max Deviation		ANN Max Deviation		Difference in Max Deviation		Cumulative Difference in Max Deviation		AverageDifference		% Deviation	
		X	Y	X	Y	X	Y	X	Y	X	Y	X	Y
1	Alpha vs Normal Force	9.11	2.06	8.51	2.307	0.6	0.247	1.41	3.13	0.383	0.043	0.3719	0.3536
2	Alpha vs Pitching Moment Coefficient	10.6	20.22	10.58	25.78	0.02	5.56	0.040	4.25	0.383	0.710	0.3719	0.3616
3	Normal Force Coefficient vs CDF	6.763	2.83	6.881	2.792	0.118	0.038	0.99	0.038	0.041	0.108	0.3499	0.1919
4	Normal Force Coefficient vs Pitching Moment Coefficient	6.763	60.6	6.934	61.9	0.171	1.3	0.170	1.3	0.044	0.710	0.3499	0.3552
5	Alpha vs Lift Force	21.42	5.91	21.36	6.546	0.06	0.63	0.06	0.63	0.383	0.041	0.3719	0.3591
6	Normal Force Coefficient vs Lift Force Coefficient	6.763	6.154	6.812	6.3	0.049	0.146	0.049	0.146	0.044	0.042	0.3499	0.3552
7	Alpha Vs CDF	10.6	0.724	10.58	0.882	0.02	0.15	0.040	0.563	0.383	0.109	0.3719	0.1895

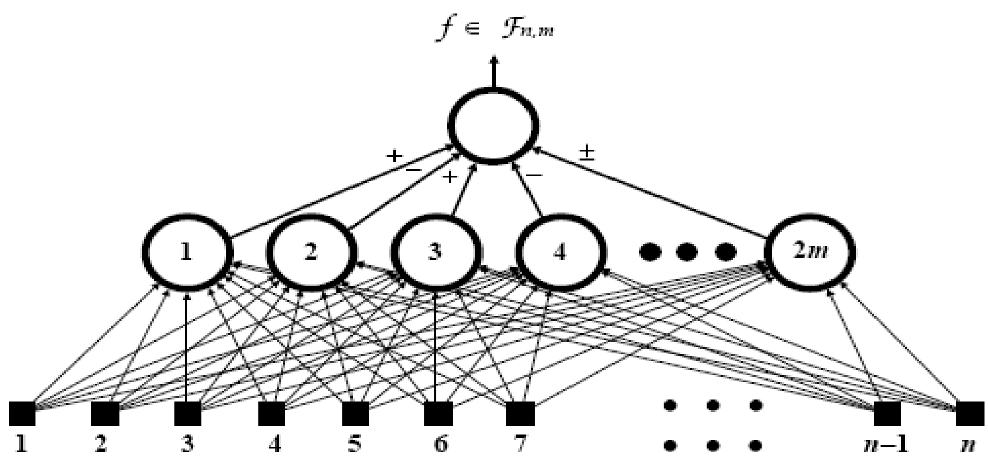


Figure 13. Architecture for learning from examples

6. Appendix A

Proof : Learning From Examples

Any set of k functions of n variables specified by m examples can be computed by a neural network with ±1 weights having

$$size_{kf}{}^{*}(n,m\Delta) = o(m^{*}(n+k)/\Delta)$$, and

$$depth_{kf}{}^{*}(n,m\Delta) = o(\lg(mn)/\lg(\Delta))$$,

and occupying an area of:

$$A_{kf}^{*}(n,m\Delta) = o(m(n+k))$$

for all values of the fan-in(Δ) in the range *2*to *n.*

The construction is:

M TGs of n variables in the first layer (atmost); as they are MAJORITY gates of n inputs (AND- equivalent gates),

and

K TGs of m/2 variables in the second layer which are MAJORITY gates of m/2 inputs (OR- equivalent gates)

We can compute:

$$depth_{kf}^{*}(n,m\Delta)=[\lg n/\lg\Delta]+[\lg m/\lg\Delta]=O(\lg(mn)/\lg(\Delta))$$

$$size_{kf}^{*}(n,m\Delta)=m[n\text{-}1/\Delta\text{-}1]+k[m\text{-}1/\Delta\text{-}1]=O(m\Delta(n+k))$$

and occupying:

$$A_{kf}^{*}(n,m\Delta)=m\Delta[n\text{-}1/\Delta\text{-}1]+k\Delta[m\text{-}1/\Delta\text{-}1]=O(m(n+k))$$

7. CONCLUSION

This research paper showed the applicability of the neural network approach to graph prediction in the Wind tunnel test using the angle of attack and coefficient of frictional drag coefficients. The back propagation network performed similar tothe Radial basis function neural network but the time factor involved with the Backpropagation network is high. The back propagation neural network topology of 18,28,10,1 and systematically selected network parameters (learning rate of 0.35, no momentum factor, and about 150 epochs), performed equally better with the radial basis function neural network in both recall and generalization. The Generalized regression neural network provides optimal solution with quicker learning duration. This paper presents a complete neural network development process of training, recall, and generalization for an interesting Wind tunnel test application. The research provides fruitful results to make neural network computing more suitable to draw new graphs in Wind tunnel tests with any Mach number.

8. REFERENCES

[1] R. Malmathanraj and S. ThamaraiSelvi , 2008. Prediction of Aerodynamic Characteristics Using Neural Networks. Asian Journal of Information Technology, 7: 19-26.

[2] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. 1986. Learning internal representations by error propagation. In Parallel Distributed Processing, pages 318-362, D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, Editors, The MIT Press, Cambridge, MA.

[3] S. M. Weiss and I. Kapouleas. 1989. An empirical comparison of pattern recognition, neural nets, and machine learning classification methods. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89), pages 781-787, Detroit, MI, August 20-25.

[4] C. Moallemi. 1991. Classifying cells for cancer diagnosis using neural networks IEEE Expert, pages 8-12, December.

[5] R. Mooney, J. Shavlik, G. Towell, and A. Gove. 1989. An experimental comparison of symbolic and connectionist learning algorithms. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89), pages 775-780, Detroit, MI, August 20-25.

[6] A. S. Hussain, X. Yu, and R. D. Johnson. 1991. Application of neural network computing in pharmaceutical product development. Pharm. Res., 8:1248-1252.

[7] P. Veng-Pedersen and N. B. Modi. 1992. Neural networks in pharmacodynamic modeling. Is current modeling practice of complex kinetic systems at a dead end, J. Pharmacokinet. Biopharm., 20:397-412.

[8] R. J. Erb. 1992. Neural network computation of nonlinear pharmacological effects from concentration of drug in the central compartment independent of equilibrium status. Pharm. Res., 9:293.

[9] Y. Suzuki, T. Aoyama, and H. Ichikawa. 2007. Neural networks applied to pharmaceutical problems. J. Med. Chem., 33(9):2583-2590 (1990). Tavel, P. Modeling and Simulation Design. AK Peters Ltd.

[10] Bodor, N.; Huang, M.-J. 1991. Neural Network Studies. 1. Estimation of the Aqueous Solubilty of Organic Compounds. J. Am. Chem. Soc.113, 9480-9483

[11] RajkumarThirumalainambi and Jorge Bardina, "Training data requirement for a neural network to predict aerodynamic coefficients", Proc. SPIE 5102, 92 (2003); doi:10.1117/12.486343