International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:7, 2007

Approximate Bounded Knowledge Extraction using Type-I Fuzzy Logic

Syed Muhammad Aqil Burney, Senior Member IEEE, Tahseen Ahmed Jilani, Member IEEE, Cemal Ardil

***Abstract*—**Using neural network we try to model the unknown function f for given input-output data pairs. The connection strength of each neuron is updated through learning. Repeated simulations of crisp neural network produce different values of weight factors that are directly affected by the change of different parameters. We propose the idea that for each neuron in the network, we can obtain quasi-fuzzy weight sets (QFWS) using repeated simulation of the crisp neural network. Such type of fuzzy weight functions may be applied where we have multivariate crisp input that needs to be adjusted after iterative learning, like claim amount distribution analysis. As real data is subjected to noise and uncertainty, therefore, QFWS may be helpful in the simplification of such complex problems. Secondly, these QFWS provide good initial solution for training of fuzzy neural networks with reduced computational complexity.

***Keywords*—**Crisp neural networks, fuzzy systems, extraction of logical rules, quasi-fuzzy numbers.

I. Introduction

FISSION of artificial neural networks and fuzzy inference systems have attracted the growing interest of researchers in various scientific and engineering areas due to the growing need of adaptive intelligent systems to solve the real world problems. A crisp or fuzzified neural network can be viewed as a mathematical model for brain-like systems. The learning process increases the sum of knowledge of the neural network by improving the configuration of weight factors. Fuzzy neural networks are generalization of crisp neural networks to process both numerical information from measuring instruments and linguistic information from human experts, see [2], [14], and [15]. Thus, fuzzy inference systems can be used to emulate human expert knowledge and experience. An overview of different fuzzy neural network architectures is discussed by [5], [7] and classified them as,

1) A fuzzy neural network may take crisp or fuzzy values as inputs and can return crisp or fuzzy output.
2) another class of fuzzy neural networks is feedforward neural networks which are defined from conventional feedforward neural networks by substituting fuzzified neurons for crisp ones. These are named as regular fuzzy neural networks.

It is much more difficult to develop the learning algorithms for the fuzzy neural networks than for the crisp neural networks; this is because the inputs, connections weights and bias terms related to a regular fuzzy neural network are fuzzy sets, see [17], [22] and [24].

The paper is organized as follows. In section II, we made a short study of learning procedures in crisp neural networks. In section III, we present concepts of fuzzy logic and quasi-fuzzy sets. In section IV, simulation experiments using crisp neural network is performed repeatedly to achieve quasi-fuzzy sets. These sets provide the initial solution for type-I neuro-fuzzy networks as discussed by [9], [28] and [29]. To our knowledge, the concept of obtaining fuzzy weights through crisp neural network has not been investigated in the literature.

II. Neural Networks

Using neural network we try to model the unknown function f for given input-output data pairs. The existing algorithms for these problems are regression modeling, neural networks, and wavelet theory. A neural network can be regarded as representation of a function determined by its weight factors and networks architecture [15]. The overall mapping is thus characterized by a composite function relating feedforward network inputs to output. That is

$$\mathbf{O} = \mathbf{f}_{composite}\left(\mathbf{x}\right)$$

Using p-mapping layers in a p+1 layer feedforward net yield

$$\mathbf{O} = \mathbf{f}^{L_p}\left(\mathbf{f}^{L_{p-1}}\ldots\left(\mathbf{f}^{L_1}\left(\mathbf{x}\right)\ldots\ldots\right)\right).$$

Usually, we train a neural network with a training set, present inputs to the neural networks, and interpret the outputs according to the logical rules in the training set see [1], [3],[4] and [21]. The most commonly used technique to adjust weight parameters of a neural network is backpropagation method based on LMS learning defined as

$$J = E\left[\sum_k e_k^2\left(n\right)\right]$$

Manuscript received July, 13. 2005.
S. M. Aqil Burney is Professor in the Department of Computer Science, University of Karachi, Pakistan (phone: 0092-21-9243131 ext. 2447, fax: 0092-21-9243203, e-mail: Burney@computer.org, aqil_burney@yahoo.com).
Tahseen Ahmed Jilani is lecturer and Research fellow in the Department of Computer Science, University of Karachi, Pakistan (e-mail: tahseenjilani@ieee.org).
Cemal Ardil is with National Academy of Aviation, Baku, Azerbaijan (e-mail: cemalardil@gmail.com).

where k= number of output neurons.

$$w_{ji}^{l}(n+1)=w_{ji}^{l}(n)+\eta\,\delta_{j}^{l}(n)\,y_{i}^{l-1}(n),$$

η is the learning rate and $\delta_j^l(n)$ is the local change made at each neuron in the learning, see [15]

$$\delta_{j}^{l}(n)=\begin{cases} e_{j}^{L}(n)\Phi_{j}^{'}\left(\upsilon_{j}^{L}(n)\right) \\ (\textit{for neuron } j \textit{ in output layer } L) \\ \Phi_{j}^{'}\left(\upsilon_{j}^{l}(n)\right)\sum_{k}\delta_{k}^{l+1}(n)\,w_{kj}^{l+1}(n) \\ (\textit{for neuron } j \textit{ in output layer } l) \end{cases}.$$

But to deal with noisy and uncertain information, a crisp neural network has to use concepts of fuzzy interference systems [27].

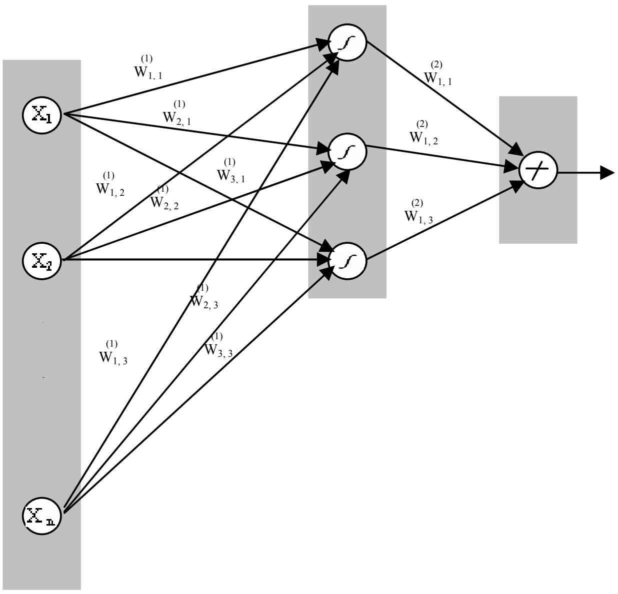

Fig. 1 Structure of a crisp artificial neural network

III. Fuzzy Logic

Fuzzy logic was originally proposed by Prof. Lotfi A. Zadeh to quantitatively and effectively handle problems involving uncertainty; ambiguity and vagueness see [12] and [13]. The theory which is now well-established was specifically designed to mathematically represent uncertainty and vagueness and provide formalized tools for dealing with the imprecision that is intrinsic to many real world problems. The ability of fuzzy logic is inherently robust since it does not require precision and noise-free inputs. Fuzzy inference systems are the most reliable alternative if the mathematical model of the system to be controlled is unavailable [11],[18] and [26]. The fuzzy sets and fuzzy rules can be formulated in terms of linguistic variables. Methods of fuzzy logic are commonly used to model a complex system by a set of rules provided by the experts. But fuzzy rules can also be applied in reverse problems: given the input-output behavior of a system, what are the rules which are governing the behavior.

We cite definitions of fuzzy set and membership function cross over points, alpha-cut sets and convexity of a fuzzy set see [10].

Definition 1: If X is a collection of objects denoted generically by x, then a fuzzy set **A** is defined as a set of ordered pairs,

$$A=\{(x,\mu_A(x)),|x\in X\},$$

Where $\mu_A(x)$ is called the membership function for the fuzzy set **A**. The membership function maps each element of X to a membership grade value between 0 and 1.

Definition 2: The $\alpha-cut$ or α - level set is a non-fuzzy set of a fuzzy set A denoted by $\mathbf{A}^{\alpha}$ and defined as

$$\mathbf{A}^{\alpha}=\{x \mid \mu_A(x)\geq\alpha\} \tag{1}$$

Thus every fuzzy set can be represented as a set of it's $\alpha-cuts$ as

$$\mathbf{A}=\{\mathbf{A}^{\alpha_1},\mathbf{A}^{\alpha_2},....,\mathbf{A}^{\alpha_m}\} \tag{2}$$

Definition 3: A fuzzy set is convex if and only if for any $x_1,x_2\in X$ and any $\lambda\in[0,1]$

$$\mu_{\mathbf{A}}(\lambda x_1+(1-\lambda)x_2)\geq\min\ (\mu_{\mathbf{A}}(x_1)+\mu_{\mathbf{A}}(x_2))$$

Alternatively, **A** is convex if all of its $\alpha-cut$ sets are convex.

Definition 4: A quasi-fuzzy number **A** is a fuzzy set of the real line with a normal, fuzzy convex and continuous membership function satisfying the following conditions,

$$\lim(t\to-\infty)\,\mathbf{A}(t)=0\ ,\ \lim(t\to\infty)\,\mathbf{A}(t)=0 \tag{3}$$

Let **A** be a fuzzy number. Then $\mathbf{A}^{\gamma}$ is a closed convex subset of $\mathbf{R}$ for all $\gamma\in[0,1]$ defined as

$$a_l(\gamma)=\min(\mathbf{A}^{\gamma}),\ a_r(\gamma)=\max(\mathbf{A}^{\gamma}) \tag{4}$$
$$a_l:[0,1]\to\mathbf{R}\ \ a_r:[0,1]\to\mathbf{R}$$

Then $\mathbf{A}^{\gamma}=[a_l(\gamma),a_r(\gamma)]$. The support of A is the open interval $(a_l(\gamma),a_r(\gamma))$.

Definition 5: A triangular membership function is specified by three parameters (a_m,a_l,a_r) as follows:

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:7, 2007

$$trian\ (x; a_l, a_m, a_r) = \max\left(\min\left(\frac{x-a_l}{a_m-a_l}, \frac{a_r-x}{a_r-a_m}\right), 0\right) \quad (5)$$

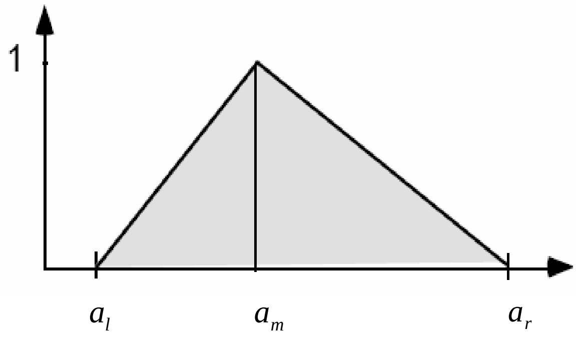

Fig. 2 Quasi triangular fuzzy set

In order to reduce computational expense, we use triangular fuzzy numbers $\tilde{a} = (a_m, a_l, a_r)_{trian}$ to define the fuzzy weight. These quasi-fuzzy weights sets follow fuzzy arithmetic, and thus can be used for fuzzy neural networks.

IV. Experiment

In this paper we demonstrate the learning and obtaining fuzzy membership functions of weight vectors to obtain quasi-fuzzy weight sets. The input/target pair presented to the network is $\{\underline{\mathbf{X}}, \underline{t}\}$ where $\underline{\mathbf{X}} = [\underline{x}_1, \underline{x}_2, \underline{x}_3, \underline{x}_4, \underline{x}_5]$. A crisp neural network with 3 hidden and one output neuron is trained with performance function 1e-06 and repeated the simulation for first 100 successes.

Daily close share prices are considered from Karachi stock exchange for 200 trading days and are preprocessed. For each of the hidden neuron and output neuron, the simulated weight values may be plotted.

The QFWS of first input connected to all the three neuron are shown in figure 4. The triangular membership is constructed due to its reduced complexity [8] and [19] and [20]. For $w^1_{1,1}$ as shown in fig. 3, using (5) the parameters of triangular-mf will be,

$$a_l = \min(w^1_{1,1}) \; a_r = \max(w^1_{1,1}) \; a_m = {(a_l + a_r)}/{2} \quad (6)$$

Secondly, [15] defines that each hidden weight connection of neuron lies approximately in the interval

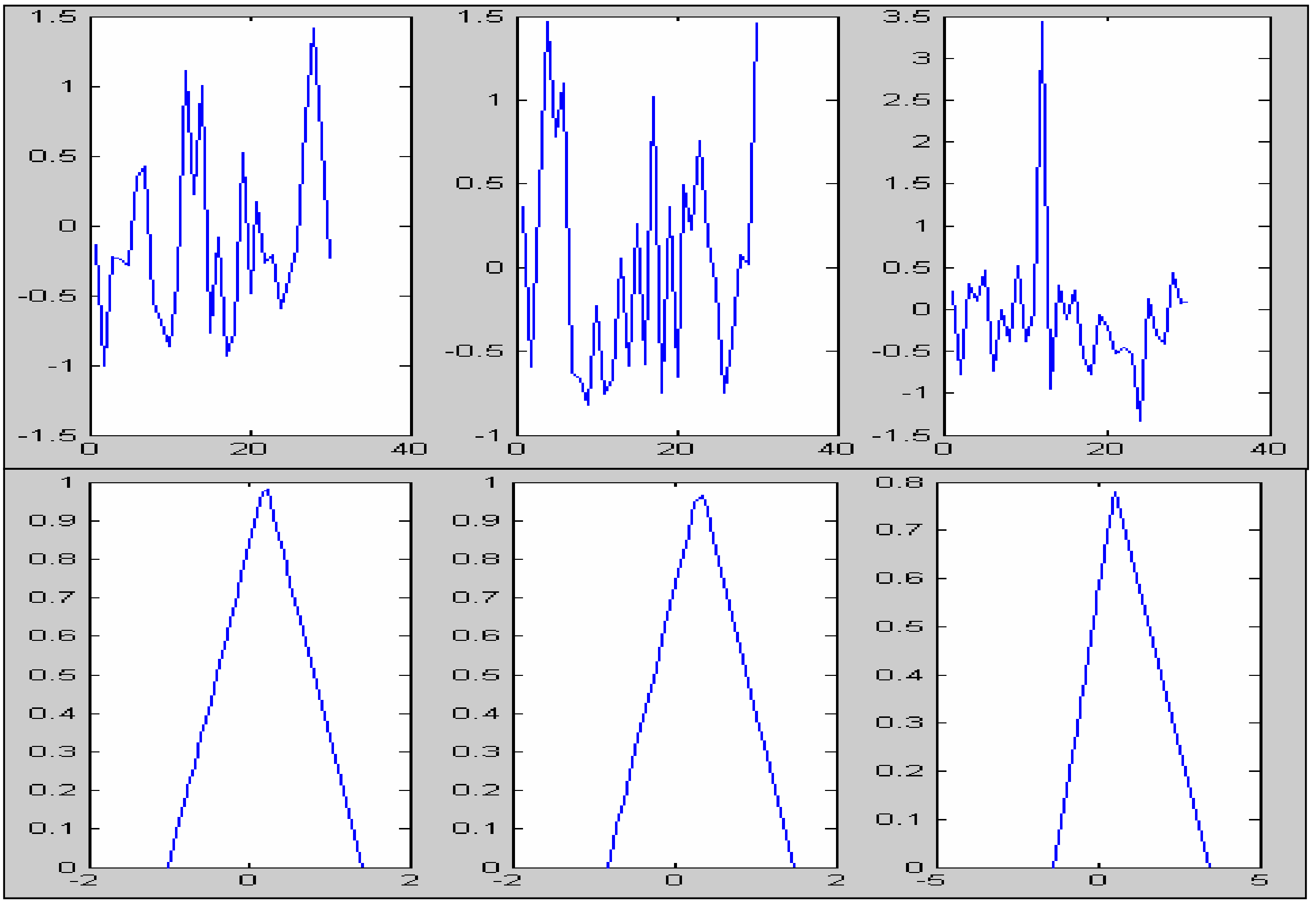

Fig. 3 (a) Input weight matrix for first input vector, (b) Triangular-mf for first weight matrix $\mathbf{w}^{i,1}$, $i = 1,2,3$ $(no.of\ neurons)$

$$-\frac{1}{\sqrt{n_h}} < w_{ij} < \frac{1}{\sqrt{n_h}} \tag{7}$$

Our proposed interval based weight set in eq. (6) provides little large interval to search for weights of hidden part of a fuzzy neural network.

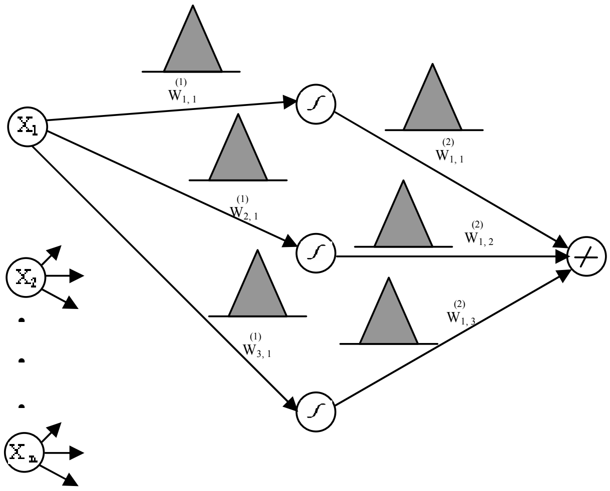

Fig. 4 Proposed quasi-fuzzy weight neural network

V. Conclusion

We described the architecture of QFWS based fuzzified neural networks and presented a general framework of learning algorithms of fuzzified neural networks. Learning in neuro-fuzzy learning with fuzzy weights requires initialization of an interval based fuzzy sets, which require higher computing than for crisp learning to deal with uncertainty, vagueness and linguistic behaviors of some real life situations see [6], [16], [23] and [25].

Further improved identification of suitable membership functions is possible by determining the underlying probability structure of synaptic connections of a crisp neural network. Thus based on this idea, we can form fuzzy inference systems with varying rules. This may provide new research directions to compare different QFWS based fuzzy neural networks.

Acknowledgement

Very thankful to Mr. M. Najam-ul-Hasnain of Department of Computer Science, University of Karachi for his computing support.

The authors would like to thank the referees for their helpful suggestions and comments.

References

[1] Aqil Burney S.M., Jilani A. Tahseen and Cemal Ardil, “A comparative study of first and second order training algorithms for artificial neural networks”, Int. Journal of Computational Intelligence, vol. 1, no.3, 2004, pp. 218-224.

[2] Aqil Burney S.M., Jilani A. Tahseen and Cemal Ardil, “Levenberg-Marquardt algorithm for Karachi Stock Exchange share rates forecasting”, Int. J. of Computational Intelligence, vol. 1, no. 2, 2004, pp 168-173.

[3] Aqil Burney S.M., Jilani A. Tahseen, “Time Series forecasting using artificial neural network methods for Karachi Stock Exchange”, A Project in the Dept. of Computer Science, University of Karachi. 2002.

[4] F. Scarselli and A. C. Tosi, “Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results,” Neural Networks, vol. 11, no. 1, 1998, pp. 15-37.

[5] G. Castellano and A.M. Fanelli, “Fuzzy inference and rule extraction using a neural network", Neural Network World Journal, vol. 3, 2000, pp. 361-371.

[6] H. Ishibuchi and M. Nii, “Numerical analysis of the learning of fuzzified neural networks from if-then rules,” Fuzzy Sets Syst., vol. 120, no. 2, 2001, pp. 281-307.

[7] H. Ishibuchi, Fujioka, and Tanaka, (1993), “Neural networks that learn from fuzzy If-then rules”, IEEE Transactions on Fuzzy Systems, vol. 1. no. 2. 1993.

[8] J. Dunyak and D. Wunsch, “Training fuzzy numbers neural networks with alpha-cut refinement,” in Proc. IEEE Int. Conf. System, Man, Cybernetics, vol. 1, 1997, pp. 189-194.

[9] J. J. Buckley and Y. Hayashi, “Neural networks for fuzzy systems,” Fuzzy Sets and Systems 1995, pp. 265-276.

[10] Jang, Sun and Mizutani, Neuro-fuzzy logic and Soft Computing; A computational approach to learning and machine intelligence. New York: Practice-Hall, 2003, Chap. 2-4

[11] Jerry M. Mendel, Uncertainly Rule-Based Fuzzy Logic Systems. Introduction and new Directions. New York: Prentice Hall PTR, NJ.2001, chapter 1-7.

[12] L. A. Zadeh, “The concept of linguistic variable and its applications to approximate reasoning”, Parts I,II,III, Information Sciences, 8(1975) 199-251; 8(1975) 301-357; 9(1975) 43-80. 30.

[13] L. A. Zadeh, “Outline of a new approach to the analysis of complex systems and decision processes”, IEEE Trans. Systems, Man and Cybernetics, 1973, vol. 3, pp. 28-44.

[14] Mir F. Atiya, Suzan M. El-Shoura, Samir I. Shaken, “A comparison between neural network forecasting techniques- case study: river flow forecasting". IEEE Trans. on Neural Networks. Vol. 10, No. 2. 1999.

[15] M. Bishop, Neural networks for pattern recognition. United Kingdom: Clarendon Press, 1995, chapter 5-7.

[16] Nauck and R. Kruse, “Designing neuro-fuzzy systems through backpropagation”, in Fuzzy Modeling: Paradigms and Practice, Kluwer, Boston, 1996. pp. 203-228.

[17] Nauck, Detlef and Kruse, Rudolf, “Designing neuro-fuzzy systems through backpropagation”, In Witold Pedryz, editor, Fuzzy Modeling: Paradigms and Practice, 1996. pp. 203-228, Kluwer, Boston.

[18] Nilesh N. Karnik, Jerry M. Mendel and Qilian Liang, “Type-2 Fuzzy Logic Systems”, IEEE Trans. Fuzzy Syst., 1999, vol. 15, no. 3,pp. 643-658.

[19] P. Eklund, J. Forsstrom, A. Holm, M. Nystrom, and G. Selen, “Rule generation as an alternative to knowledge acquisition: A systems architecture for medical informatics”, Fuzzy Sets and Systems, vol. 66 1994, pp. 195-205.

[20] P. Eklund, “Network size versus preprocessing, Fuzzy Sets, Neural Networks and Soft Computing” (Van Nostrand, New York, 1994, pp. 250-264.

[21] Puha, P. K. H. Daohua Ming, “Parallel nonlinear optimization techniques for training neural networks.”, IEEE Trans. on Neural Networks, vol. 14, no. 6, 2003, pp 1460-1468.

[22] Puyin Liu and Hongxing, ”Efficient learning algorithms for three-layer regular feedforward fuzzy neural networks”, IEEE Trans. Fuzzy Syst., vol. 15, no. 3, 2004, pp. 545-558.

[23] S. Mitra and Y. Hayashi, “Neuro-fuzzy rule generation: Survey in soft computing framework,” IEEE Trans. Neural Networks., vol. 11, no. 3, 2000, pp. 748–768.

[24] S. M. Chen, “A weighted fuzzy reasoning algorithm for medical diagnosis”, Decision Support Systems, vol. 11, 1994, pp.37-43.

[25] Sungwoo Park and Taisook Han, “Iterative Inversion of Fuzzified Neural Networks”, IEEE Trans. Fuzzy Syst., vol. 8, no. 3, 2000, pp. 266-280.

[26] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control", IEEE Trans. Syst. Man Cybernet., 1985, pp. 116-132.
[27] Włodzisław Duch," Uncertainty of Data, Fuzzy Membership Functions, and Multilayer Perceptrons", IEEE, Trans. on Neural Network, vol. 16, no.1, 2005.
[28] Xinghu Zhang, Chang-Chieh Hang, Shaohua Tan and -Pei Zhuang Wang," The Min-Max Function Differentiation and Training of Fuzzy Neural Networks", IEEE Trans, Neural Networks, vol. 7. no. 5, 1996, pp. 1139-1149.
[29] Y. Hayashi, J. J. Buckley, and E. Czogala, "Fuzzy neural network with fuzzy signals and weight, "in Proc. Int. Joint Conf. Neural Networks, vol. 2, Baltimore, MD, pp. 1992, pp. 696-701.

S. M. Ail Burney received the B.Sc, first class first M.Sc. M.Phil. from Karachi University in 1970, 1972 and 1983 respectively. He received Ph.D. degree in Mathematics from Strathclyde University, Glasgow with specialization in estimation, modeling and simulation of multivariate Time series models using algorithmic approach with software development.
He is currently professor and approved supervisor in Computer Science and Statistics by the High Education Commission, Government of Pakistan. He is also member of various higher academic boards of different universities of Pakistan. His research interest includes AI, soft computing, neural networks, fuzzy logic, data mining, statistics, simulation and stochastic modeling of mobile communication system and networks and network security. He is author of three books, various technical reports and supervised more than 100 software/Information technology projects of Masters level degree programs and project director of various projects funded by Government of Pakistan.
He is member of IEEE (USA), ACM (USA) and fellow of Royal Statistical Society United Kingdom and also a member of Islamic society of Statistical Sciences. He is teaching since 1973 in various universities in the field of Econometric, Bio-Statistics, Statistics, Mathematic and Computer Science He has vast education management experience at the University level. Dr. Burney has received appreciations and awards for his research and as educationist including NCR-2002 award for Best National Educationist.

Tahseen A. Jilani received the B.Sc., first class second M.Sc. (Statistics) and M.A. (Economics) from Karachi University in 1998, 2001 and 2003 respectively. Since 2003, he is Ph.D. research fellow in the Department of Computer Science, University of Karachi.
He is member of IEEE-Computational Intelligence Society. His research interest includes AI, neural networks, soft computing, fuzzy logic, Statistical data mining and simulation. He is teaching since 2002 in the fields of Statistics, Mathematic and Computer Science.