International Journal of Innovative Scientific & Engineering

Technologies Research 6(2):59-65, April-June, 2018

 www.seahipaj.org ISSN: 2360-896X

Simulating Adaptive Neuro Fuzzy Inference System (ANFIS) Training Using Student Grade Data

Eti F. I.

Department of Computer Science,
Delta State, Polytechnic, Ozoro, Nigeria
Email: etiefe2018@gmail.com
+234(0)8023392286

ABSTRACT
Machine learning has been identified as an integral facet of Artificial Intelligence (AI), which provides the basis for intelligent model to learn from training data without been intrinsically programmed. The aim of this research paper is geared toward exposing the fundamentals of Adaptive Neuro-Fuzzy Inference System (ANFIS) using Matrix Laboratory (MatLab). This fundamental focuses on identify the ANFIS training error.Sequent to the simulation, the following: Number of nodes: 13181, number of linear parameters: 6561, number of non-linear parameters: 72, total number of parameters: 6633, number of training data pair: 20, number of checking data pair: 0, number of fuzzy rules: 6561. Holistically, the simulation captures a training error of 8.3252e-005 and an average testing error of 8.325e-005.
Keywords: Simulation, Matlab, ANFIS

1.0 INTRODUCTION

Model training and prediction is sub-domain of Artificial Intelligence (AI), built on the premises of computer science. Model training usestraining data encompassing training patterns to understand the structural composition of the data. It predict the output of these data based on the inherent and exhibited features. Model training and prediction is perceived as machine learning process which is the science of applying computer based simulation software's to mimic and learn base on human intuition processes. This learning is autonomous in fashion based on observed entities and real interaction. Machine learning applies learning algorithm to partition data, organize data and structure this data in such a way that the hidden pattern are intuitional enhanced from it (Mohri et al., 2012; Russell and Norvig, 2003).

Machine learning broadly can be classified into three main categories: supervised, unsupervised and reinforced learning (Russell and Norvig, 2003.).Supervised learning apply machine learning to inferring a function from training data (Stevan, 2008). Supervised learning algorithms analyses training data and providing a generalized inferred function. Unsupervised learning algorithms rely hugely on its environment (Mohri et al., 2012). Unsupervised learning perceives learning based on input patterns with the aim of representing the overall statistical structure based on input patterns. This input pattern are obtained from the environment. Reinforced learning can be seen as an intermediate between supervised on unsupervised learning relying hugely on it environment and modeling training data (Bishop, 2006). The continuous adaptations rely feedback responses received from its environment (Steven, 2008; Bishop, 2012).

Adaptive Neuro Fuzzy Inference System (ANFIS) is a supervised learning paradigm, combining Artificial Neural network and Fuzzy Inference System (FIS) (Abraham, 2006; Tahmasebi, 2012)

This research paper provides a simplified implementation, simulating Adaptive Neuro Fuzzy Inference System (ANFIS) training errors using student grade data.

2.0 MATERIAL AND METHOD

Adaptive Neuro-Fuzzy Inference System (ANFIS) is a combination of Artificial Neural Network (ANN) and Fuzzy Inference System (FIS) using Takagi Sugeno model. It is an adaptive network with adaptive capabilities (Mathur, 2006). ANFIS is usually a six layer network implemented as 0 – 5. Each node is designed with a distinct functionalities and processes. ANFIS can be implemented as a single or hybrid training. This training usually implemented as Backward Propagation Gradient Descent (BPGD) and Least Square Estimator (LSE) method. The computational capabilities of these structures are usually optimal, explanatory and power in model training (Al-Hmouz, 2012).

ANFIS as a network includes: Layer 0 (L0) is the first ANFIS layer which interface with it environment. It is seen as the input layer due to its ability in obtaining external input sequence usually used by several layers (Sharma, 2012).Layer 1 (L1), identify and utilize appropriate membership obtain and assigned using an appropriate membership function. It is called the membership function layer. This node operates based on an appropriate membership function mapping using linguistic variables and labels. Layer 2 (L2), also called the rule, processes and producing rule firing output based on the assigned membership and available assigned parameter. This layer calculate the firing strength using either AND or OR operator. Layer 3 (L3), layer also called the normalization layer provides the normalized node values using sequential input variables streaming from rule layer. Layer 4 (L4), also known as the defuzzification provide the consequent output of each variable input using the normalized values, fuzzy values and the membership values. Layer 5 (L5), produces the final consequent output summing all defuzzifcation values obtained from layer 4.

ANFIS can be classified among a class of predictive model which learn from training data based on target values. The simulation of ANFIS is made possible through the provision of input parameter and rules (Sharma, 2012). Using IF-TEHN- Rules obtained from fuzzified input with appropriate membership, the consequent parameters are produced. Through the forward and backward pass produced through the hybrid algorithm (Least Square Estimator: LSE and Back Propagation Gradient Descent: BPGD) and output target is obtained (Al-Hmouz, 2012; Sharma, 2012).

Mathwork is the consortium responsible for maintaining Matlab,In fact matlab was designed and developed by mathwork. Matlab allow algorithm design and implementation, data representation, interface design and functional programming. These are usually accomplished exploring the toolbox available within matlab enhancing symbolic data representation. Functions and subroutine written in C, C++, C#, Pascal and FORTRAN could be called using Matlab. This is usually achieved using a wrapper function usually allowing matlab data types to be passed and returned.

3.0 ANFIS Simulation Interfaces

Simulation data were obtained employing a structured interview approach applied anonymously to thirty students (30) randomly selected within four (400) level in other to obtain .student academic data (students' scores). The student grade courses cut across eight (08) 400L courses: 401, 403, 408, 409, 402, 420, 412, and 417. A total of thirty student were interviewed randomly. Table 3.1, provides the simulation results.

Table 3.1: Student Grade Data for Simulation

Samples	**401**	**403**	**408**	**409**	**402**	**420**	**412**	**417**
Sample 1	40	45	52	50	40	42	42	50
Sample 2	40	40	60	71	50	40	40	54
Sample 3	40	60	54	66	48	60	65	60
Sample 4	40	40	45	65	57	40	40	53
Sample 5	50	45	60	68	50	50	69	40
Sample 6	55	50	55	60	38	60	45	30
Sample 7	70	38	67	40	55	62	54	40
Sample 8	75	55	70	63	40	50	40	40
Sample 9	40	40	40	71	40	60	60	40
• •	• •	• •	• •	• •	• •	• •	• •	• •
Sample 30	30	55	70	74	75	54	54	60

Matrix Laboratory (MATLAB): MATLAB 9.5 (2010b) was employed extensive in this study as a simulation tool. It was also supported by interactive and computing environment. It provides a platform in solving computing problem possibly faster than conventional programmable components.
Microsoft Excel was also employed in exporting dataset into Matlab environment.
A laptop system was employed with a minima Ram of 2gb, 300gb hard disk and a Microsoft window 7 as operating system.
The simulation data was structured as: 67% for training while the remaining 33% was used for validation or testing. Figure 3.1 – Figure 3.3, depicts the ANFIS simulations.

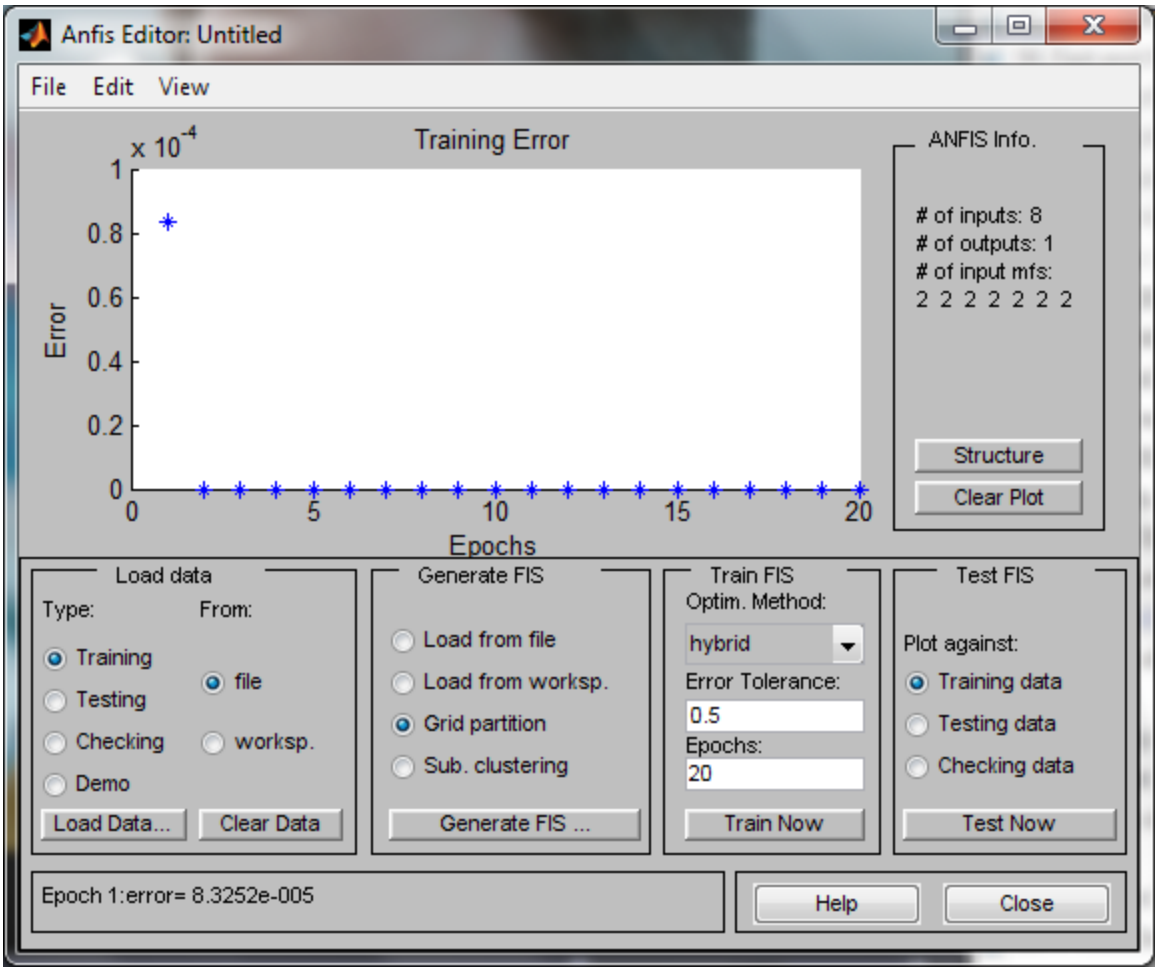

Figure 3.1:ANFIS Training Error

Figure 3.1, shows the ANFIS training error. The horizontal plot shows the total number of training data while the vertical bar shows the training values with the highest been 0.004. The training was initially set with epoch of 20 and tolerance error of 0.5. Overall training was attained at epoch 1 with 8.3252e-005.

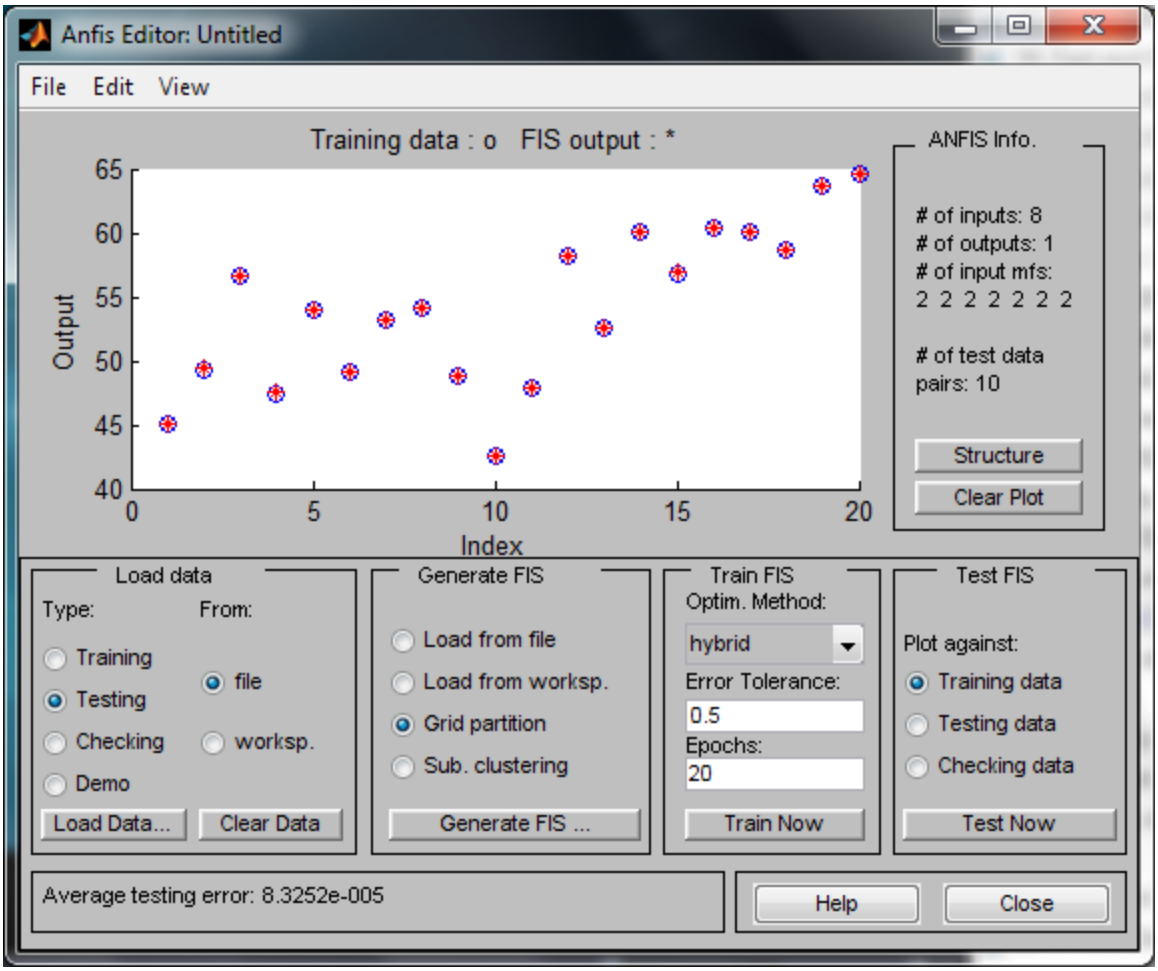

Figure 3.2:ANFIS Training Data vs. Fuzzy Inference Output

Figure 3.2, shows ANFIS Training Data vs. Fuzzy Inference Output. The horizontal plot shows the total number of training data while the vertical bar shows the output and associated values. The perfectly pairing of ANFIS output with training data show the optimality of training and testing which was aligned with the training epoch of 20 and tolerance error of 0.5. The overall average testing error: 8.325e-005.

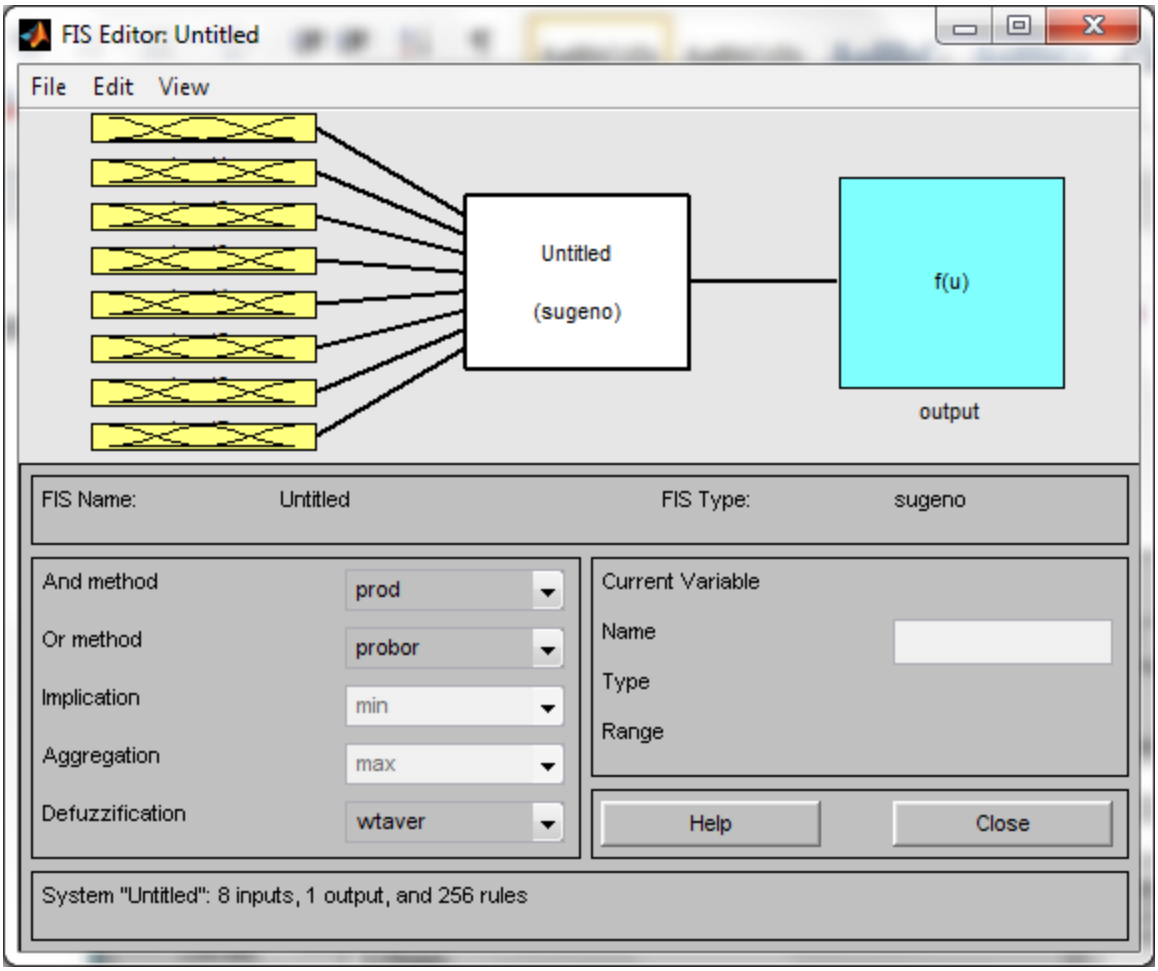

Figure 3.3: Fuzzy Inference System

Figure 3.3 shows the Fuzzy Inference System (FIS) as fuzzified input variables and defuzzified output. The sugeno fuzzy system produces a weighted consequent output with the sugeno implication process utilizes Or operator in returning weighted output.

4.0 SIMULATION RESULT ANALYSIS

The simulation resulted presented through the Matlab simulation offers the following results: Number of nodes: 13181, number of linear parameters: 6561, number of non-linear parameters: 72, total number of parameters: 6633, number of training data pair: 20, number of checking data pair: 0, number of fuzzy rules: 6561.

From the simulation the following were noted:

a. The number of node produces is largely dependent on the number of input parameters
b. The number of linear parameter is dependent of number of input variables, fuzzy membership function.
c. The non-linear is dependent on membership assigned.
d. Number of parameter is explored from input variables and simulation components.
e. The number of training pair is a function of the training data and it permutation
f. Number of checking data is dependent on the size of training data
g. The number of fuzzy rules is a permutation of the fuzzy membership and parameter.

5.0 CONCLUSION

The simulation of student grade data using Adaptive Neuro Fuzzy Inference System (ANFIS) has been explored in this research paper. Microsoft Excel, A predefined laptop and Matrix Laboratory (Matlab) were identified as simulation tools. Matlab as a simulator provided the training error interface, the validation interface and the fuzzy inference interfaces. The ANFIS training provide a training error of with8.3252e-005 confirmed at the first epoch. This was subsequently followed by an average testing error of 8.325e-005. The minute errors show the optimality of training using ANFIS.

REFERENCES

Abraham, A. (2005), "Adaptation of Fuzzy Inference System Using Neural Learning", in Nedjah, Nadia; de Macedo Mourelle, Luiza, Fuzzy Systems Engineering: Theory and Practice, Studies in Fuzziness and Soft Computing, 181, Germany: Springer Verlag, pp. 53–83, doi:10.1007/11339366_3

Al-Hmouz A. (2012), Modeling and Simulation of an Adaptive Neuro-Fuzzy Inference System (ANFIS) for Mobile Learning, https://ieeexplore.ieee.org/document/6104032/

Bishop, C. M. (2006), Pattern Recognition and Machine Learning, Springer, ISBN 0-387-31073-8.

Stevan H. (2008), "The Annotation Game: On Turing (1950) on Computing, Machinery, and Intelligence", in Epstein, Robert; Peters, Grace, The Turing Test Sourcebook: Philosophical and Methodological Issues in the Quest for the Thinking Computer, Kluwer

Tahmasebi, P. (2012). "A hybrid neural networks-fuzzy logic-genetic algorithm for grade estimation"

Russell, S.; Norvig, P. (2003) Artificial Intelligence: A Modern Approach (2nd ed.). Prentice Hall. ISBN 978-0137903955.

Mohri, M., Rostamizadeh, A.; Talwalkar, A. (2012). Foundations of Machine Learning. USA, Massachusetts: MIT Press. ISBN 9780262018258.

Mathur N. (2016),*Comparison of adaptive neuro-fuzzy inference system (ANFIS) and Gaussian processes for machine learning (GPML) algorithms for the prediction of skin temperature in lower limb prostheses*, https://www.sciencedirect.com/science/article/pii/S1350453316301515.

Sharma M. (2012),Artificial Neural Network Fuzzy Inference System (ANFIS), retrieved https://arxiv.org/pdf/1212.0059