

## An Artificial Neural Network Based Digital Differential Protection Scheme for Synchronous Generator Stator Winding Protection

Muhammad Faisal Riaz<sup>1</sup>, Fawwad Hassan Jaskani<sup>2,\*</sup> and Tehreem Awan<sup>1</sup>

<sup>1</sup>NFC Institute of Engineering and Technology, Multan

<sup>2</sup>The Islamia University of Bahawalpur, Bahawalpur

### Abstract

This research depicts another artificial neural network (ANN) based digital differential protection scheme for generator stator winding protection. The scheme incorporates two feedforward neural networks (FNNs). One ANN is utilized for flaw recognition and the other is utilized for inward deficiency grouping. This structure utilizes current examples from the line-side and the unbiased end notwithstanding tests from the field current. Essential and/or second consonant present in the field current during an issue help the ANN, utilized for flaw location, to separate between generator states (typical, outside issue and interior deficiency states). Results demonstrating the performance of the protection scheme are displayed and show that it is quick and solid.

**Keywords:** ANN, Winding Protection, Neural Networks.

Received on 10 January 2019, accepted on 27 January 2019, published on 30 January 2019

Copyright © 423; "O vj co o cf "Hkucn'Tk| et al., licensed to EAI. This is an open access article distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/eai.30-1-2019.160837

\*Corresponding author. Email: Favadhassanjaskani@gmail.com

### 1. Introduction

#### 1.1. Overview

Transformer and generator are the most fundamental components of the power framework with their protection significance. Since most recent three decades, analysts have been dealing with this specific point and rose to numerous new strategies however generally focused on person protection framework. There are assortments of defensive transfers to give solid and secure transformer protection, of which the differential transfers are observed to be more viable [1] in blame segregation than the old harmonic restraint techniques. The differential transfers ought to be planned in a manner that it doesn't mal-work amid polarizing inrush and over excitation states of transformer. The inrush flows produced after blame leeway are additionally to be considered, as in [2], while structuring the hand-off. The greater part of the techniques pursue a deterministic methodology, depending on fixed limit.

#### 1.2. Artificial Neural Networks

The ANN-based algorithms have been effectively actualized in many example or mark acknowledgment issues, as they can distinguish solid states of generator and transformer based on perceiving their wave shapes, all the more decisively, by separating them from the blame current wave shapes [3]-[5]. In [6], Neural Network Principle Component Analysis alongside Outspread Basis Function Neural Networks is utilized as example classifier. At the end of the day, this procedure makes the choice based on the present mark check which is more exact than customary harmonic restraint based techniques utilized for the protection of transformer. This system could deliver the stumbling signal in the occasion of inward blame inside 15ms after blame event. Optimal Probabilistic Neural Network (PNN) utilized in [7] as the center classifier to segregate among inrush and inside blame. Molecule Swarm Optimization is utilized to acquire optimal smoothing factor for PNN. PNN requires bigger capacity for model examples and it is progressively troublesome to prepare attributable to numerical troubles.

### 1.3. Differential Protection

Differential protection, utilizing electromechanical and strong state transfers, is the most widely recognized technique utilized by electric utilities for stator winding protection. Nonetheless, the fast advances in computerized innovation empowered analysts also, originators to gain huge ground in creating chip based protection calculations [1], [2] several chip based calculations for recognizing stator winding flaws have been proposed. A calculation that utilizes immediate contrasts between lineside also, nonpartisan end flows for recognizing stage issues is depicted in [3]. The whole of the quick flows was utilized to give a restriction. Cross-relationship was used to process basic recurrence phasors of the line-side and nonpartisan end flows in [4] the wholes and results of these flows were utilized in the blame discovery calculation. A technique which checked the nearness of the second symphonious in the field winding to distinguish issues is portrayed in [5]. The course of negative-grouping control stream at the generator terminals was utilized to separate among interior and outside flaws. An advanced system that utilizes the positive and negative grouping models of the generator, and voltages and flows estimated at the generator terminals to separate between inward and outside deficiencies is given in [6]. The utilization of multifunctional computerized transfers for generator protection has likewise been examined [7], [8]. These transfers utilize advanced blame location calculations include various flag pre-preparing prerequisites.

### 1.4. Feedforward Neural Networks

The capacity of Artificial Neural (ANNs), in specific Feedforward Neural Networks (FNNs), to learn via preparing any complex input/output their utilization as example classifiers fruitful [ scientists have connected ANNs in various diverse territories [21. In transmission line protection, FNNs have been genius course separation [10], blame classifier versatile reclosing [1a]. ANNs are additionally useful in perceiving charging inrush current waveforms, subsequently transformer protection [13]. An ANN has additionally been propose to distinguish early blames, as turn-to turn stator protection blame and bearing we stage acceptance engines [14]. The utilization of ANN the field current of an alternator and afterward identifying the field winding interterm blame is portrayed.

Another methodology based on decision tree for separation among inrush and inner blame with better precision is exhibited in [8]. This strategy cases to take preparing time of 0.02sec (1 cycle) with characterization precision of 97.77%. Correspondingly, ANN based techniques have been utilized for the protection of generator as well. One such plan with basic ANN is exhibited in [9] for stator winding protection. Three parallel ANNs have been utilized in this plan for characterizing three diverse blame cases. Another such plan is exhibited in [10] where two separate ANNs are utilized for blame discovery and blame grouping. A propelled adaptation of this strategy utilizing fluffy rationale in mix with ANN is exhibited in [11]. In the two cases, blame waveforms are recreated utilizing direct stage amounts strategy. A down to earth protection plot is executed in [12] with ANN created on a digital signal processor (DSP). In spite of the fact that the significance of consolidated/unit protection frameworks has been distinguished in late nineties, not very many have done research on unit protection frameworks from that point forward.

### 1.5. Half Breed Protection

A half breed protection plot is exhibited in [13] for the protection of generator-transformer unit thinking about most of the blame sorts. This plan is created utilizing three microprocessors based on customary harmonic restraint circuit strategy. This gave a base thought for unit protection frameworks. Afterward, two ANN based techniques were introduced in [14], [15] in mix with customary strategies with a blame identification time of 20ms roughly. In the two cases, ANN had been prepared with back spread algorithm. A ground blame unit protection framework is introduced in [16] thinking about as it were the ground deficiencies happening in generator. A considerable lot of the proposed algorithms delivered great results as far as exactness. A superior algorithm can continuously improve the unwavering quality of the protection plot. Be that as it may, utilization of a reinforcement protection framework improves the dependability and usefulness of protection gadgets. This research introduces a model of decision framework based on ANN considering the generator-transformer unit as the ensured object. All the interior blame states of transformer and generator have been reenacted to produce the required database for the preparation of ANN. Likewise, few instances of deficiencies are created utilizing the strategy given in [10]. These cases are utilized just amid testing of the networks. The created ANN has been prepared and tried with RPROP and Genetic Algorithm and the results are analyzed. Amid this procedure, different designs of ANN have been tried by differing the number of concealed neurons and keeping the quantity of info and yield neurons fixed. Definite depiction about these sources of info and yields is examined in later segments.

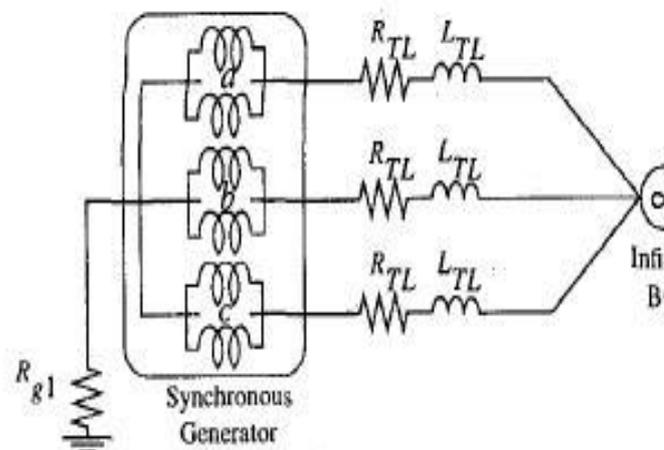
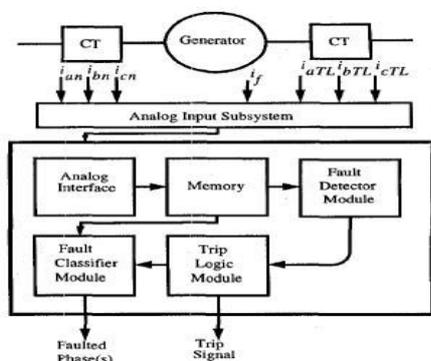


Figure 1. Synchronous Generators with Infinite Bus

To satisfactorily prepare the ANNs to identify blames and arrange inside shortcomings, an exact model for simulating typical task state, outside blame state and inner blame state in a synchronous generator ought to be accessible. For the reason for this venture an electromagnetic transient program for simulating generator states was assembled. Figure demonstrates the reproduced model. It comprises of a multiparallel way synchronous generator associated with an unending transport through a short transmission line (TL), having an obstruction  $R_{TL}$  and an inductance  $L_{TL}$ . Both the generator also, the interminable transport neutrals are grounded through protections  $R_{g1}$  and  $R_{g2}$  individually. Typical task at various power levels and power

variables can be reproduced utilizing this model. Reenactment of the outside issues is based on the algorithm exhibited in [16], by which it is conceivable to recreate a wide range of outside issues that can happen along the TL.



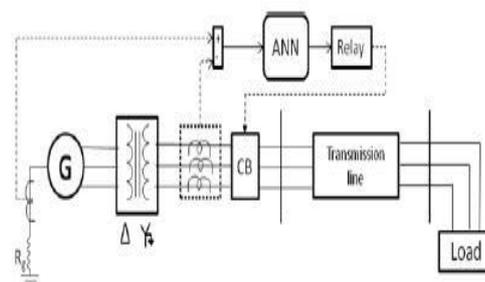
**Figure 2.** an algorithm for simulating interior blames in the synchronous generator of Fig. 1

### 1.6. Proposed Algorithm

An algorithm for simulating interior blames in the synchronous generator of Fig. 1, utilizing the immediate stage amounts, was created. This algorithm pursues the equivalent essential lines followed in [16], for the reenactment of outer flaws. The strategy for ascertaining oneself and shared inductances of the blamed winding of the synchronous machine is based on the examination exhibited in [17], [18], and [19].

The inner deficiencies algorithm can reproduce extraordinary kinds of inner flaws at various rates of the stator winding. A full depiction of that algorithm is accessible in [20]. Henceforth, the created model can reproduce each of the three states. The model gives the momentary estimations of the rotor flows and of the flows in the stator ways, in per unit (Pu).

A three-stage control framework including a 200MVA, 13.8kV Generator and a 200MVA 13.8/132kV -Yg Transformer alongside a 150 km transmission line has been utilized to create the required test and preparing designs. Fig. 2 demonstrates the plan of the unit protection framework and Fig. 2 demonstrates the power framework display made by methods for MATLAB Simulink programming. Extraordinary sorts of deficiencies are made at various areas. All the generator flaws are accepted to happen at 100% of the stator winding. Additionally, inrush present and over excitation conditions are mimicked at various voltage edges and with various burdens. The created waveforms are at that point inspected to bolster the neural networks to be tried with two diverse testing rates.



**Figure 3.** structure of the proposed ANN based differential protection scheme

The structure of the proposed ANN based differential protection scheme, utilized for distinguishing flaws and ordering inward blames, is appeared in Fig. 2. The modules appeared in Fig. 2 are not the total modules of a digital hand-off, yet rather the vital modules required for the proposed plot. In this area the activity of the protection plot as an independent transfer is portrayed. Be that as it may, the ANN modules utilized in the plan can be utilized to help existing algorithms utilized for generator protection, thus expanding the unwavering quality of the protection activity. A. Sources of info the contributions to the transfer are the generator three stage flows from both the line-side (&TL, ibTL, &TL) and the nonpartisan end (I, ibn, IC,) notwithstanding the field current (I f). The second harmonic present in the field current, amid a blame, has been utilized beforehand in a generator digital protection algorithm to show the presence of an irregularity [5]. Albeit crucial as well as second harmonic (depending on the sort of blame) show up in both inner and outside deficiencies, the recreation results appeared that their amplitudes in outer and inward blames are unique. Henceforth, tests of the field current are utilized to help the ANN based blame locator module to separate between the three states.

### 1.7. Simple Input Subsystem

In a real equipment usage the contributions to these modules would be the low dimension signals given by the present transformers (CT) [1]. Be that as it may, in this venture the contributions to the simple information subsystem are the immediate Pu current qualities given by the reenactment demonstrate depicted in area 11. An inspecting recurrence of 1200 Hz (20 tests/cycle) is utilized in this handing-off plot. Subsequently, to abstain from associating issues, an antialiasing low pass channel, with a cut-off recurrence of 570 Hz, is utilized at the simple information subsystem.

### 1.8. Objectives

Design of a new ANN based digital differential protection scheme for generator stator winding protection is presented in this paper. The scheme has two ANNs:

One ANN is used by the fault detector module and the other by the fault classifier module.

The ANN based fault detector module is used to discriminate between three generator states, namely the normal operation state, external fault state and internal fault state.

To adequately train the ANNs to detect faults and classify internal faults, an accurate model for simulating normal operation state, external fault state and internal fault state in a synchronous generator should be available.

For the purpose of this project an electromagnetic transient program for simulating generator states was built. Figure 1 shows the simulated model.

## 2. Literature Review

### 2.1. Overview

Power transformer is a standout amongst the most imperative segments in power framework, for which different kinds of defensive and observing plans have been created for a long time.

### 2.2. Differential protection

Differential protection is a standout amongst the most broadly utilized techniques for securing power transformer against inward blames. The system is based on the estimation and examination of flows at both side of transformer: essential and auxiliary lines. The differential transfer trips at whatever point the distinction of the flows in the two sides surpasses a foreordained edge. This method is precise in the vast majority of the instances of transformer inside deficiencies anyway mal-task of differential hand-off is conceivable because of inrush flows, which result from homeless people in transformer attractive transition. The homeless people in transformer attractive motion may happen because of empowerment of transformer, voltage recuperation after blame leeway or association of parallel transformers.

The presence of such current unsettling influences has made the protection of intensity transformers a difficult issue for protection engineers. In this way, exact characterization of flows in a power transformer is need of this difficult issue, in avoiding maloperation of the differential hand-off under various no-fault conditions including charging inrush, over-excitation, outside blame, and immersion of current transformers [1].

### 2.3. Background Studies

Since last 1960s, analysts have extensive enthusiasm for the region of digital protection of intensity contraption [1]. The primary highlights which have pulled in specialists to explore the attainability of structuring digital transfers for power framework protection are its speed of activity, steadfastness, solidness, economy, adaptability, and probability of incorporating a digital hand-off into the progressive PC framework inside the substation and with the lattice. Further the digital transfer gives protection, yet in addition is utilized for status observing of intensity device. Also, with the utilization of artificial insight in defensive gadgets, the decision-production capacity of the transfers is improved.

### 2.4. Early Techniques

Early techniques were based on desensitizing or postponing the transfer to defeat the homeless people [2]. These strategies are unacceptable by the by, since the transformer were presented to since quite a while ago unprotected occasions. Improved security and trustworthiness at that point was acknowledged when the second harmonic substance concerning the key one was presented as a distinguishing proof model, known as harmonic restraint differential protection [3]. Be that as it may, a few scientists revealed the presence of a lot of the second harmonic in some winding deficiencies [4, 5]. Moreover, the new ages of intensity transformers utilize low-misfortune indistinct material in their centre, which can create inrush flows with lower harmonics substance and higher extents [5]. In such cases, a few creators have changed the proportion of second harmonic to essential

limiting rule by utilizing different proportions characterized at a higher recurrence [6]. While different scientists proposed concealed Markov's model [7], fluffy rationale based techniques [4, 8], wave formed acknowledgment strategy [1, 9], and furthermore artificial neural networks-(ANNs-) [10, 11] based learning design way to deal with show signs of improvement characterization exactness, low computational weight, and quick reaction of the transfer. In any case, these techniques rely upon fixed limit list (either in time area or in recurrence space) and these may require extensive computational weight. Besides, the execution of an ANN especially relies upon its speculation ability, which thus is needy upon the information portrayal. One critical normal for information portrayal is uncorrelated. At the end of the day, a lot of information introduced to an ANN should not to comprise of associated data. This is on the grounds that corresponded information diminish the uniqueness of information portrayal and consequently acquaint disarray with the ANN display amid learning process and henceforth, delivering one that has low speculation capacity to determine inconspicuous information.

## 3. Methodology

### 3.1. Artificial Neural Network

The ANN-based algorithms have been successfully implemented in many pattern or signature recognition problems, as they can detect healthy conditions of generator and transformer based on recognizing their wave shapes, more precisely, by differentiating them from the fault current wave shapes [3]-[5]. In [6], Neural Network Principle Component Analysis along with Radial Basis Function Neural Networks is used as pattern classifier. In other words, this technique makes the decision based on the current signature verification which is more accurate than traditional harmonic restraint based techniques used for the protection of transformer. This technique could produce the tripping signal in the event of internal fault within 15ms after fault occurrence. Optimal Probabilistic Neural Network (PNN) used in [7] as the core classifier to discriminate between inrush and internal fault. Particle Swarm Optimization is used to obtain optimal smoothing factor for PNN. PNN requires larger storage for exemplar patterns & it is more difficult to train owing to numerical difficulties.

### 3.2. Our Approach

A new approach based on decision tree for discrimination between inrush and internal fault with better accuracy is presented in [8]. This method claims to take processing time of 0.02sec (1 cycle) with classification accuracy of 97.77%. Similarly, ANN based techniques have been used for the protection of generator too. One such scheme with simple ANN is presented in for stator winding protection. Three parallel ANNs have been used in this scheme for classifying three different fault cases. Another such scheme is presented in where two separate ANNs are used for fault detection and fault classification. An advanced version of this method using fuzzy logic in combination with ANN is presented in [11]. In both cases, fault waveforms are simulated using direct phase quantities method. A practical protection scheme is implemented in [12] with ANN developed on a digital signal processor (DSP).

Although the importance of combined/unit protection systems has been identified in late nineties, very few have carried out research on unit protection systems since protection of generator-transformer unit considering most of the fault types. This scheme

is developed using three microprocessors based on conventional harmonic restraint circuit method. This gave a base idea for unit protection systems. Later, two ANN based techniques were presented in [14], [15] in combination with conventional methods with a fault detection time of 20ms approximately. In both cases, ANN had been trained with back propagation algorithm. A ground fault unit protection system is presented in [16] considering only the ground faults occurring in generator.

### 3.3. Research Gap

Many of the proposed algorithms produced good results in terms of accuracy. A better algorithm can always improve the reliability of the protection scheme. However, use of a backup protection system improves the reliability and functionality of protection devices. This research presents a model of decision system based on ANN considering the generator-transformer unit as the protected object. All the internal fault conditions of transformer and generator have been simulated to generate the required database for the training of ANN. Also, few cases of faults are generated using the method given in [10]. These cases are used only during testing of the networks. The developed ANN has been trained and tested with RPROP and Genetic Algorithm and the results are compared. During this process, various architectures of ANN have been tested by varying the number of hidden neurons and keeping the number of input and output neurons fixed. Detailed description

### 3.4. Neural Network Design and Simulation

The first step to formulate the problem is identification of proper input and output set. Various architectures and combination of input sets were attempted to arrive at the final configuration with a goal of maximum accuracy. Keeping the number of outputs fixed at 2, the number of input neurons and the number of hidden neurons are varied on trial and error basis until it produced minimum error. Two configurations are finalized for testing. Both the proposed ANNs generate 2 outputs to represent 4 classifications as shown in Table I. The basic architecture of the ANN is shown in Fig. 3.

#### A. Resilient Backpropagation (RPROP) Algorithm

Resilient Backpropagation is a modification of the ordinary gradient descent back-propagation. To overcome the inherent disadvantages of pure gradient-descent, Resilient Backpropagation (RPROP). This algorithm was pioneered by Martin Riedmiller [17]. The basic principle of RPROP is to eliminate the harmful influence of the size of the partial derivative on the weight step. As a consequence, only sign of the derivative is considered to indicate the direction of the weight update but not the magnitude.

The update value for each weight and bias is increased by a factor whenever the derivative of the performance function with respect to that weight has the same sign for two successive iterations. The update value is decreased by a factor whenever the derivative with respect to that weight changes sign from the previous iteration. If the where  $0 < \eta < 1 < \eta +$  RPROP is generally much faster than the standard steepest descent algorithm as it converges quickly and it is said to be the best training algorithm for pattern recognition & classification problems [18].

#### B. GA Based Training of ANN

The genetic algorithm (GA) is a well-known optimization technique based on the principles of genetics and natural selection and doesn't require derivative information for optimization. Unlike back propagation algorithm, it provides global minima of optimization function. In the proposed method, GA has been used for finding weights and biases of Artificial Neural Network. Then the next part is to define a fitness function which can be used as an evaluation function to optimize the weight set. The fitness function used here is mean square error (MSE), which has been obtained by applying all training sets (Input and Target) for each weight set in the population. The algorithm of fitness function used with GA is given below.

Let  $(I_i, T_i)$ ,  $i=1, 2, N$ , where  $I_i = (I_{i1}, I_{i2}, \dots, I_{in})$  and  $T_i = (T_{i1}, T_{i2}, \dots, T_{in})$  represents the input-output pairs of the problem to be solved by ANN with configuration l-m-n. For each chromosome  $C_i = 1, 2, \dots, p$  belonging to the current population  $P_i$  whose size is  $p$

#### C. Training and Testing of ANN

Both the ANNs are trained separately with both above algorithms. During RPROP based training, 10% sets of total samples are used for validation and another 10% are used for testing purpose. During GA based training, the ANN is trained by optimizing the weights and biases of the network to minimize MSE. The total number of variables is calculated as given below.

No. of variables = input weights + input biases + layer weights + layer biases

$$=(I*H) + H + (H*O) + O$$

$$=(I+O+1)*H+O$$

where I = No. of inputs; H = No. of Hidden neurons; O= No. of outputs.

Once the training process is completed the network is ready for testing. The network is then fed with new samples that are not used for training. For this purpose few test cases of generator have been developed using the direct phase quantities method given in [10]. For transformer fault cases, database is created in MATLAB only..

## 4. Results

### 4.1. Network Performance and Numerical Results

The designed ANN has been trained and tested with Resilient Back Propagation (RPROP) algorithm and Genetic Algorithm (GA). Further, ANN1 with 30 inputs (half cycle data) give less error than the ANN2 with 48 inputs (full cycle data). However, further decreasing the inputs didn't produce good results as the data less than half cycle is insufficient to reproduce the required wave shape to take the decision.

### 4.2. Training Time State

The training time also depends on the processor used in the PC. Present methods are implemented on the latest Intel core i7 processor based system. To further increase the training speed of the GA algorithm, parallel processing technology has been used with the help of parallel processing toolbox available in MATLAB. This allows GA to use best speed of multi-core

technology of the processor. The Intel i7 processor has 8 cores which can be used in 12 clusters or workers mode.

It is worth mentioning that both algorithms (RPROP and GA) take almost same time to detect the occurrence of fault, i.e., about 10ms for ANN1 and about 13ms for ANN2. This time is calculated based on the number of the sample at which the ANN produce a value above 0.98 at the output for a target value of '1' after the first sample of the fault wave is fed to it. Although the results are not very good when the method is applied as a primary protection system, the results can be considered satisfactory when this system is used as a backup protection unit, which generally operates after some delay from the primary protection unit.

### 4.3. Simulations

We have got 100 neural networks dataset to test our validation after training neural network for different epochs. Following are the results for different epochs at interval of 6 to test the validation performance of training time state

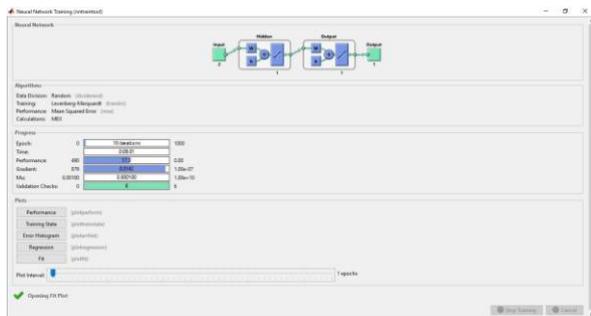


Figure 4. Complete neural network of project

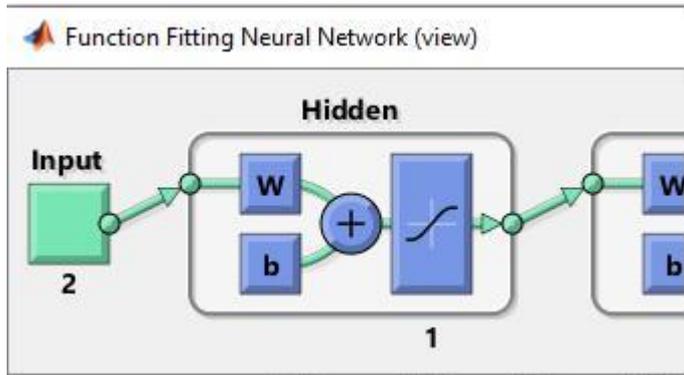


Figure 5. Hidden layers and neurons of the projects

Case 1: For epoch 12

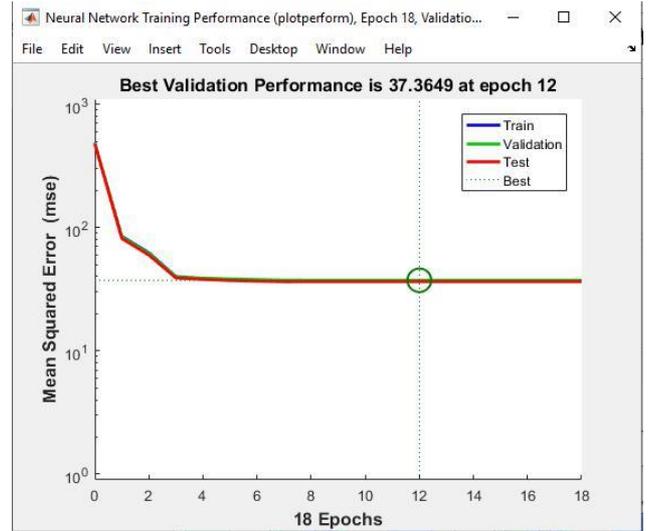


Figure 6. best validation performance is at epoch 12 and 37.3649 where the exact solution obtains

MSE is the mean (average) magnitude of the squares of the error: i.e., the distance between the model's estimate of your test values and the actual test value. (Squaring just converts things to an absolute value rather than fiddling with under or overshooting).

The physical interpretation would be that this is how close, on average, the hyperplane drawn by your network gets to the actual cloud of data in your validation set. Lila's MSE of 36.6 shows that this network can essentially guess arbitrarily close to the targets.

Now this is only half of judging performance. The other half is how validation is done (is the model any good at guessing out-of-sample values?), and how much regularization the network has done (i.e., did the network use a ton of free parameters and overfit? or is it tuned to a small set of actual regularities in the data).

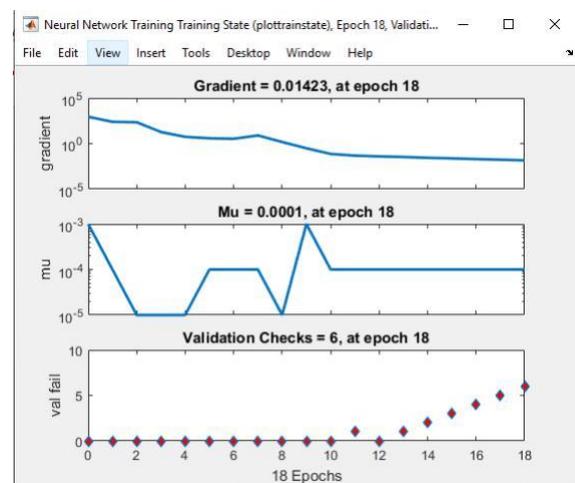
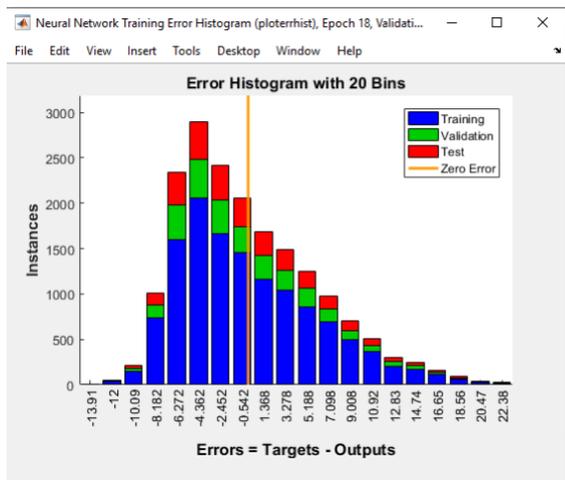
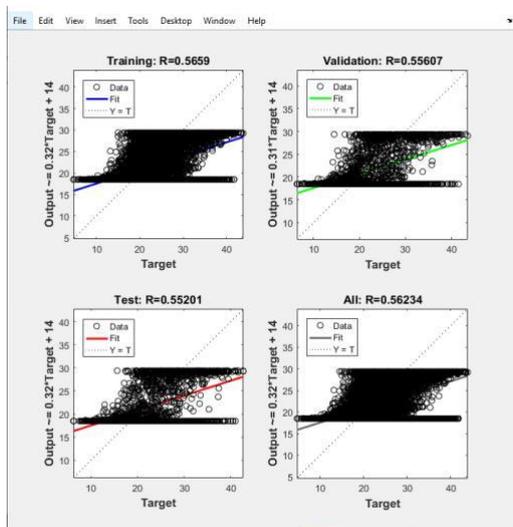


Figure 7. After 18 epochs the level of gradient, Mu and Validation Fail approaches to stability



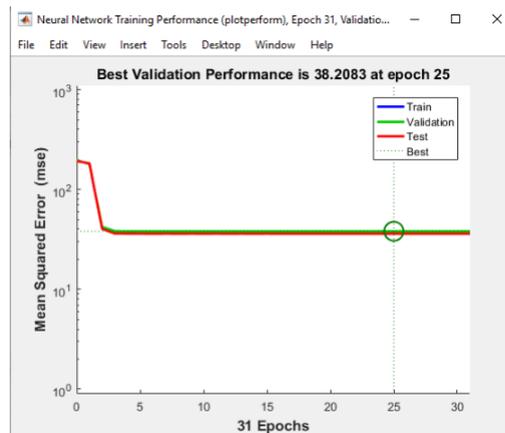
**Figure 8.** Error histogram, after validation of training, the zero error shows there is no negative feedback



**Figure 9.** the three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets.

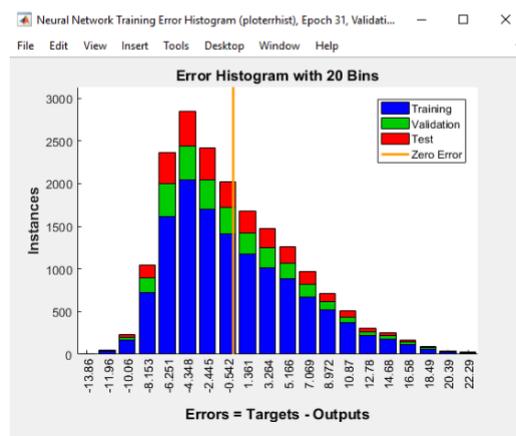
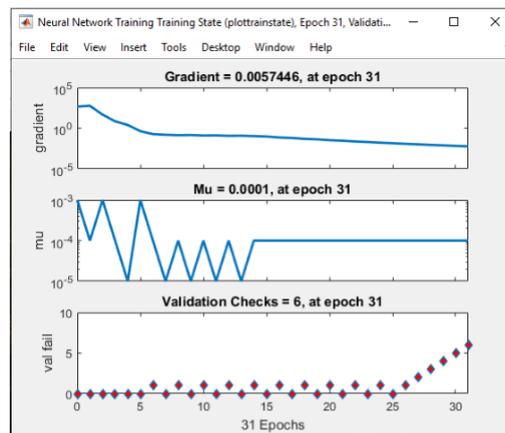
The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If  $R = 1$ , this indicates that there is an exact linear relationship between outputs and targets. If  $R$  is close to zero, then there is no linear relationship between outputs and targets.

**Case 2: For epoch 25**



**Figure 10.** MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase, equals to 38.

As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase, which in this example, equals to 38.2 at 31 epochs. The meaning of MSE being very small (close to zero) is that the desired outputs and the ANN's outputs for the training set have become very close to each other.



**Figure 11.** shows the error histogram of the trained neural network for the training, validation and testing steps. This figure shows

that the data fitting errors are distributed within a reasonably good range around zero.

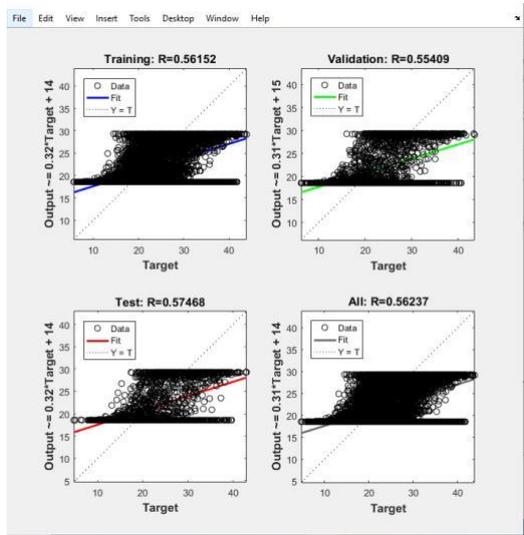


Figure 12. As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 35.7

As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 35.7 in 25 epochs. The meaning of MSE being very small (close to zero) is that the desired outputs and the ANN's outputs for the training set have become very close to each other.

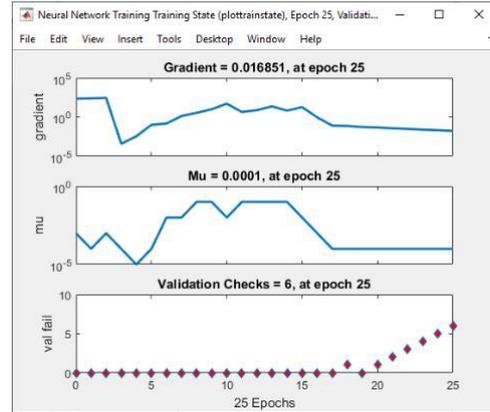


Figure 13. shows variation in gradient coefficient with respect to number of epochs. The final value of gradient coefficient at epoch number 25 is 0.01 which is approximate near to zero.

shows variation in gradient coefficient with respect to number of epochs. The final value of gradient coefficient at epoch number 25 is 0.01 which is approximate near to zero. Minimum the value of gradient coefficient better will be training and testing of networks. From figure it can be seen that gradient value goes on decreasing with increase in number of epochs.

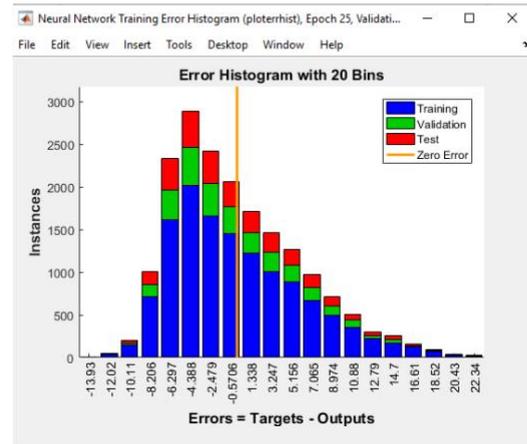
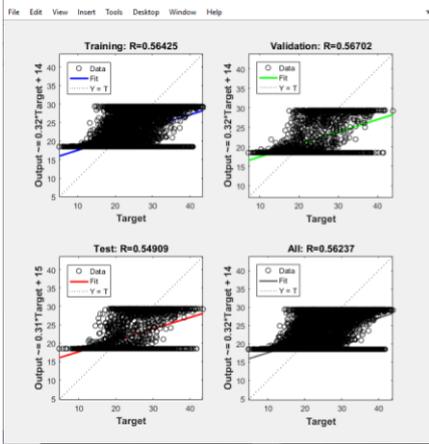


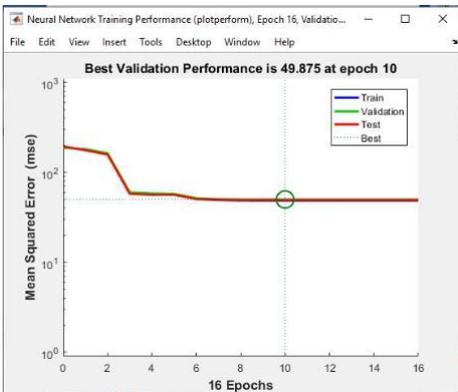
Figure 14. shows the error histogram of the trained neural network for the training, validation and testing steps. This figure shows that the data fitting errors are distributed within a reasonably good range around zero.



**Figure 15.** The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets.

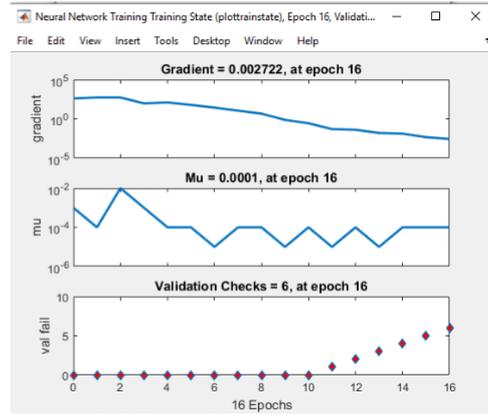
The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If  $R = 1$ , this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets.

**Case 2: For epoch 10**

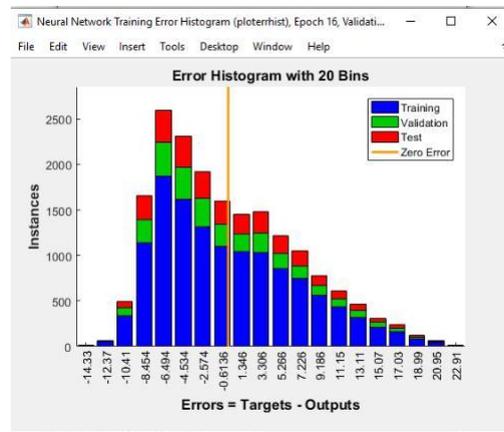


**Figure 16.** As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 49.8.

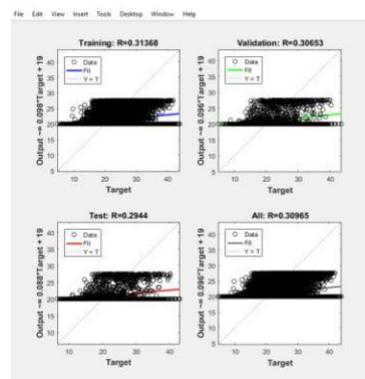
As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 49.5 in at 10 epochs in 16 epochs. The meaning of MSE being very small (close to zero) is that the desired outputs and the ANN's outputs for the training set have become very close to each other.



**Figure 17.** shows variation in gradient coefficient with respect to number of epochs. The final value of gradient coefficient at epoch number 16 is 0.02 which is approximate near to zero. Minimum the value of gradient coefficient better will be training and testing



**Figure 18.** shows the error histogram of the trained neural network for the training, validation and testing steps. This figure shows that the data fitting errors are distributed within a reasonably good range around zero.

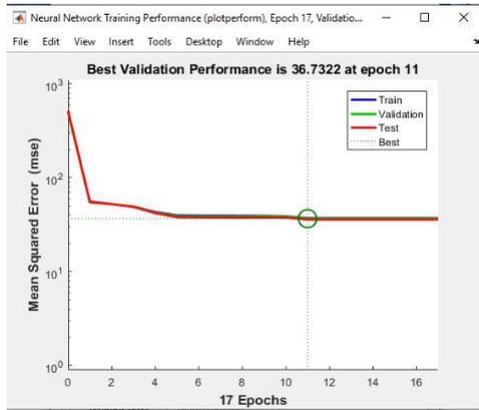


**Figure 19.** The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets.

The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs

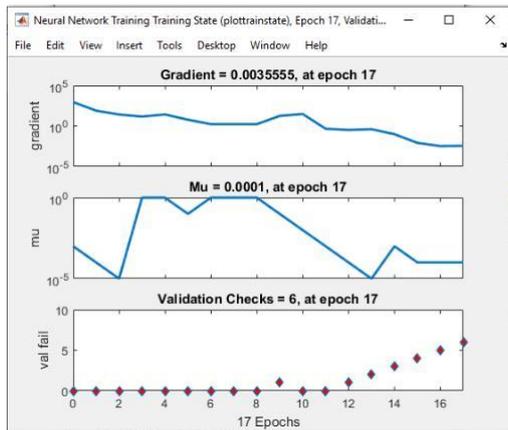
= targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If R = 1, this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets.

**Case 2: For epoch 17**

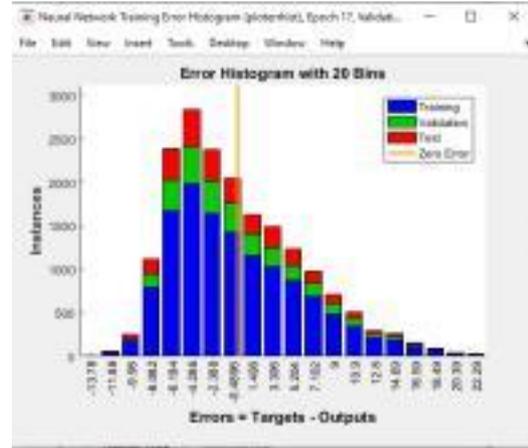


**Figure 20.** As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 36.7

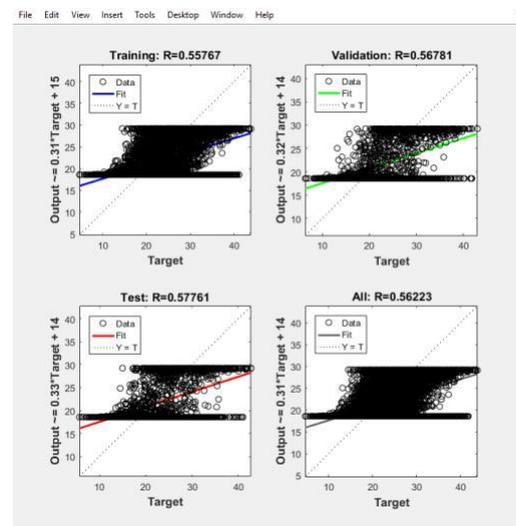
As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 36.7 at 11 epochs in 17 epochs. The meaning of MSE being very small (close to zero) is that the desired outputs and the ANN's outputs for the training set have become very close to each other.



**Figure 21.** shows variation in gradient coefficient with respect to number of epochs. The final value of gradient coefficient at epoch number 17 is 0.03 which is approximate near to zero. Minimum the value of gradient coefficient better will be training and testing



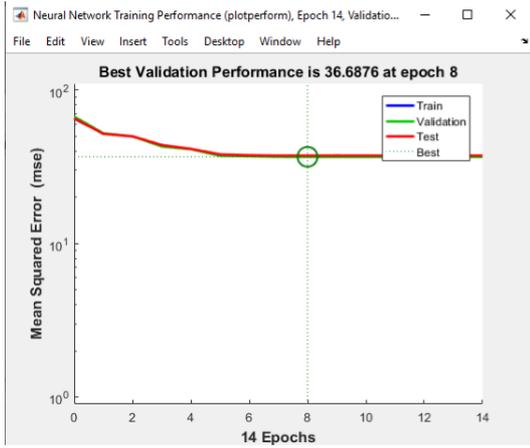
**Figure 22.** shows the error histogram of the trained neural network for the training, validation and testing steps. This figure shows that the data fitting errors are distributed within a reasonably good range around zero.



**Figure 23.** The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets.

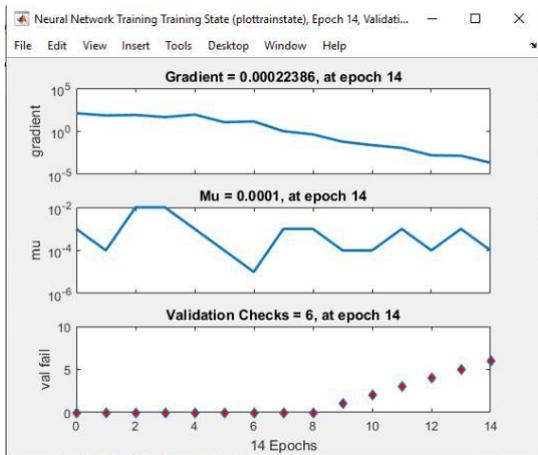
The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If R = 1, this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets.

**Case 2: For epoch 14**

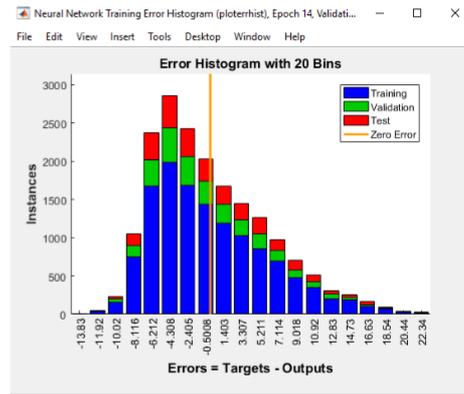


**Figure 24.** As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 36.6

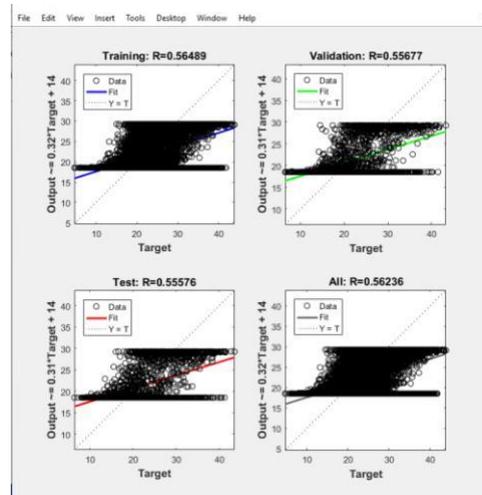
As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 36.6 at 8 epochs in 14 epochs. The meaning of MSE being very small (close to zero) is that the desired outputs and the ANN's outputs for the training set have become very close to each other.



**Figure 25.** shows variation in gradient coefficient with respect to number of epochs. The final value of gradient coefficient at epoch number 14 is 0.0002 which is approximate near to zero. Minimum the value of gradient coefficient better will be training and testing



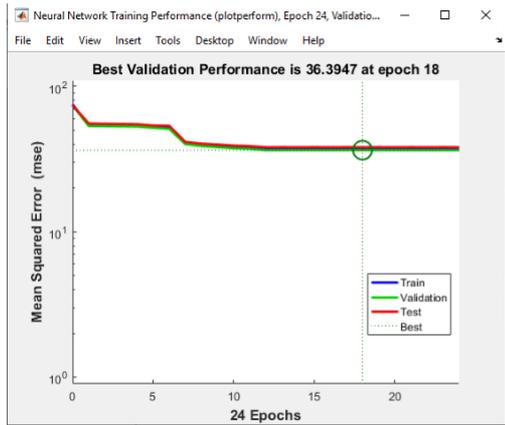
**Figure 26.** shows the error histogram of the trained neural network for the training, validation and testing steps. This figure shows that the data fitting errors are distributed within a reasonably good range around zero.



**Figure 27.** The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets.

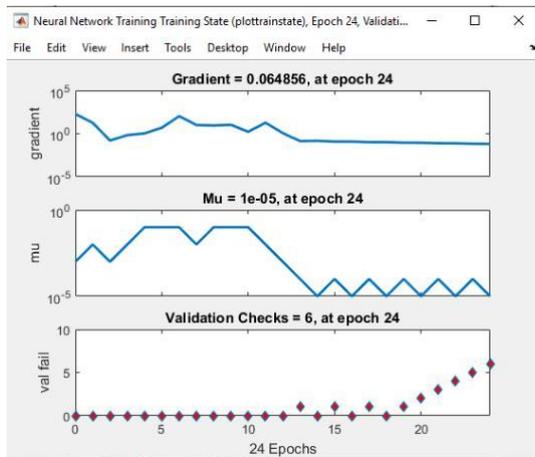
The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If R = 1, this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets.

**Case 2: For epoch 24**

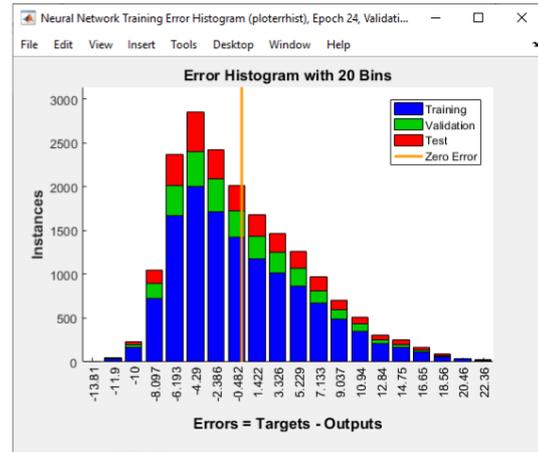


**Figure 28.** As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 36.3

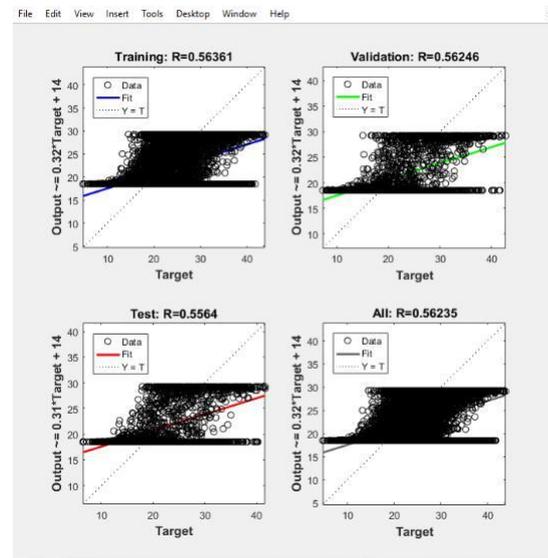
As you can see in the performance plot, with the epochs the MSE (Mean Square Error) of the ANN has decreased. A well trained ANN should have a very low MSE at the end of the training phase , which in this example, equals to 36.3 at 18 epochs in 24 epochs. The meaning of MSE being very small (close to zero) is that the desired outputs and the ANN's outputs for the training set have become very close to each other.



**Figure 29.** shows variation in gradient coefficient with respect to number of epochs. The final value of gradient coefficient at epoch number 24 is 0.06 which is approximate near to zero. Minimum the value of gradient coefficient better will be training and testing



**Figure 30.** shows the error histogram of the trained neural network for the training, validation and testing steps. This figure shows that the data fitting errors are distributed within a reasonably good range around zero.



**Figure 31.** The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets.

The three plots represent the training, validation, and testing data. The dashed line in each plot represents the perfect result – outputs = targets. The solid line represents the best fit linear regression line between outputs and targets. The R value is an indication of the relationship between the outputs and targets. If R = 1, this indicates that there is an exact linear relationship between outputs and targets. If R is close to zero, then there is no linear relationship between outputs and targets.

Table 1. Number of Iteration and Epochs will lesser the Gradient value means the train set is almost near to solution

Epochs	Validation Performance	Error Histogram	Gradient	Mu	Validation Check
--------	------------------------	-----------------	----------	----	------------------

14	37.6	-0.482	0.01	1e-05	6
17	37.2	-0.482	0.0112	1e-05	6
18	37.5	-0.482	0.002	1e-05	6
21	37.8	-0.482	0.006	1e-05	6
22	38.4	-0.482	0.0001	1e-05	6
24	38.9	-0.482	0.0001	1e-05	6
25	39.5	-0.482	0.0001	1e-05	6

## 5. Conclusion

After comparing the results, it is found that the ANN with half cycle data input is found more suitable than the remaining 3 combinations in terms of accuracy, training speed, precision and speed in fault detection. The RPROP based pattern recognition method is efficient in solving classification problems and a differential relay can be considered as a classifier which identifies what kind of event occurs on the power system network.

## References

- [1] M. Habib and M. Marin, "A comparative analysis of digital relaying algorithms for the differential protection of three phase transformers," *IEEE Trans. Power Syst.*, vol. 3, no. 3, pp. 1378-1384, 1988.
- [2] A. Wiszniewski, W. Rebizant, and L. Schiel, "New algorithms for power transformer inter-turn fault protection," *Electr. Power Syst. Res.*, vol. 79, no. 10, pp. 1454-1461, Oct. 2009
- [3] M. R. Zaman and M. A. Rahman, "Experimental testing of the artificial neural network based protection of power transformers," *IEEE Trans. Power Deliv.*, vol. 13, no. 2, pp. 510-517, Apr. 1998.
- [4] Z. Moravej and D. Vishwakarma, "ANN-Based harmonic restraint differential protection of power transformer," *Journal-Institution Eng. India Part EL Electr. Eng. Div.*, vol. 84, pp. 1-6, 2003.
- [5] E. C. Segatto and D. V. Coury, "A differential relay for power transformers using intelligent tools," *IEEE Trans. Power Syst.*, vol. 21, no. 3, pp. 1154-1162, Aug. 2006.
- [6] M. Tripathy, "Power transformer differential protection using neural network principal component analysis and radial basis function neural network," *Simul. Model. Pract. Theory*, vol. 18, no. 5, pp. 600-611, May 2010.
- [7] M. Tripathy, R. P. Maheshwari, and H. K. Verma, "Power transformer differential protection based on optimal probabilistic neural network," *IEEE Trans. Power Deliv.*, vol. 25, no. 1, pp. 102-112, Jan. 2010.
- [8] S. R. Samantaray and P. K. Dash, "Decision tree based discrimination between inrush currents and internal faults in power transformer," *Int. J. Electr. Power Energy Syst.*, vol. 33, no. 4, pp. 1043-1048, May 2011.
- [9] A. Taalab, H. A. Darwish, and T. A. Kawady, "ANN-Based novel fault detector for generator windings protection," *IEEE Trans. Power Deliv.*, vol. 14, no. 3, pp. 824-830, Jul. 1999.
- [10] A. I. Megahed and O. P. Malik, "An artificial neural network based digital differential protection scheme for synchronous generator stator winding protection," *IEEE Trans. Power Deliv.*, vol. 14, no. 1, pp. 86-93, 1999.
- [11] B. Bhalja, R. P. Maheshwari, S. Nema, and H. K. Verma, "Neuro-Fuzzy-Based scheme for stator winding protection of synchronous generator," *Electr. Power Components Syst.*, vol. 37, no. 5, pp. 560-576, Apr. 2009.
- [12] H. A. H. A. Darwish, A.-M. I. A. I. Taalab, and T. A. T. A. Kawady, "Development and implementation of an ANN-based fault diagnosis scheme for generator winding protection," *IEEE Trans. Power Deliv.*, vol. 16, no. 2, pp. 208-214, Apr. 2001.
- [13] I. Korbasiewicz, "A microprocessor based protective system for generator-transformer units," in *Proc. The Fourth International Conference on Developments in Power Protection.*, 1989, pp. 56-60.
- [14] A. Halinka and M. Szweczyk, "ANN based detection of electrical faults in generator-transformer units," in *Proc. The Eighth IEE International Conference on Developments in Power System Protection*, 2004, vol. 2004, pp. 348-351.
- [15] Y. Lu, L. Lai, and G. Tang, "Neural network based generator-transformer protection," in *Proc. 2004 International Conference on Machine Learning and Cybernetics*, 2004, vol. 7, pp. 4295-4301.
- [16] M. Zielichowski and T. Szlezak, "A new digital ground-fault protection system for generator-transformer unit," *Electr. Power Syst. Res.*, vol. 77, no. 10, pp. 1323-1328, Aug. 2007.
- [17] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," in *Proc. International Conference on Neural Networks*, San Francisco, 1993, pp. 586-591.
- [18] M. Shiblee, B. Chandra, and P. K. Kalra, "Learning of geometric mean neuron model using resilient propagation algorithm," *Expert Syst. Appl.*, vol. 37, no. 12, pp. 7449-7455, Dec. 2010
- [19] M. Habib and M. Marin, "A comparative analysis of digital relaying algorithms for the differential protection of three phase transformers," *IEEE Trans. Power Syst.*, vol. 3, no. 3, pp. 1378-1384, 1988.
- [20] A. Wiszniewski, W. Rebizant, and L. Schiel, "New algorithms for power transformer inter-turn fault protection," *Electr. Power Syst. Res.*, vol. 79, no. 10, pp. 1454-1461, Oct. 2009
- [21] M. R. Zaman and M. A. Rahman, "Experimental testing of the artificial neural network based protection of power transformers," *IEEE Trans. Power Deliv.*, vol. 13, no. 2, pp. 510-517, Apr. 1998.
- [22] Z. Moravej and D. Vishwakarma, "ANN-Based harmonic restraint differential protection of power transformer," *Journal-Institution Eng. India Part EL Electr. Eng. Div.*, vol. 84, pp. 1-6, 2003.