Neural Networks : A study on the topic of Artificial Neural Networks

Monesa Thoguluva Janardhanan
Masters of Computer Science
University of North Carolina at Charlotte
mthogulu@uncc.edu

***Abstract*—The main aim of this paper is to present the ideas grasped from the study of Artificial Neural Networks and their Activation function. Neural network helps the machine to learn, memorize and correct itself based on the experience of previous learning. This paper talks about Artificial Neural Network, their applications, their types and their architecture. This paper also discusses about activation functions, their types, their advantages and disadvantages and why is it used. It briefly touches some of the learning methods in ANN and some of the optimization methods to carry out neural network. Also 5 research papers were randomly selected for this study which uses ANN algorithms for their research. A brief explanation of their findings, algorithms used are mentioned in the Literature Review of this paper.**

***Index Terms*—Artificial Neural Network, ANN Architecture, ANN Learning Methods, ANN Elements, Activation Function, Backward Propagation.**

I. Introduction

Machine Learning Techniques are numerously used in every field as this empowers the software to learn, explore and analyses any given data without any human intervention. "While successes have been achieved in modeling biological neural systems, there are still no solutions to the complex problem of modeling intuition, consciousness and emotion - which form integral parts of human intelligence" - Alan Turing, 1950. Artificial Neural Network comes under Machine Learning Algorithms which makes the system to basically mimic how a human brain learns. Human Brain has about 100 billion basic units called neurons. Each neuron has Soma which is the nucleus of neuron, Dendrites are long irregular filaments attached to Soma and Axon which is connected to Soma in the other end. Axon terminates in a specialized contract called the synaptic junction which is the electrochemical contact between neurons. The size of the synapses are believed to be linked with learning. Coming back to neural network there are three elements in neural network which is the input layer, hidden layer and the output layer. The input layer basically accepts input features from the outside world to the network. No computations and functions are performed in this layer. It just passes on the information received to the next set of layer which is the hidden layer. Here the nodes are exposed to the outside world and all the computations and calculations are performed in this layer. It then transfers the computed result to the next layer which is the output layer. This layer displays the information learned by the machine to the output world. The nodes present in the layers are nothing but neurons. ANN is nothing but a network on interconnected neurons. Each neuron is characterized by its weight, bias and activation function which is discussed in detail in the later sections of this paper. Fig 1 displays a sample image of neural network with their elements and interconnected neurons.

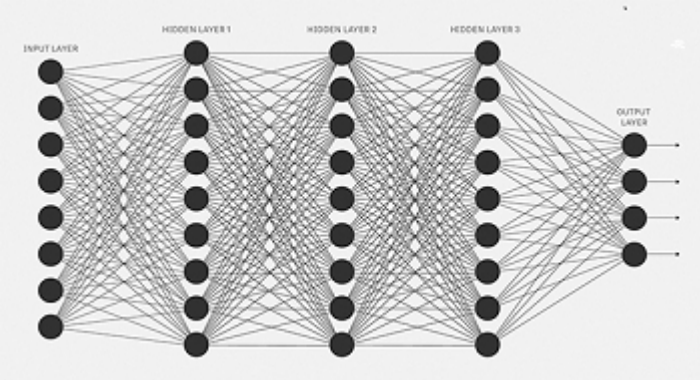

Fig. 1. Sample ANN with input layer, 3 hidden layers and output layer [1]

II. Artificial Neural Network

Artificial Neural Network consists of processing units called neurons. As discussed earlier about human brain, ANN tries to imitate the human brain. Here the Dendrites are the Input, Synapse via Axon is the output and there is an Activation function which determines whether to activate or deactivate the neurons. So the overall model consists of input, a bias which is by default 1 is added to the input and weight. A sample model of ANN model is displayed as shown in the Fig 2. The product of the weight and the input gives the

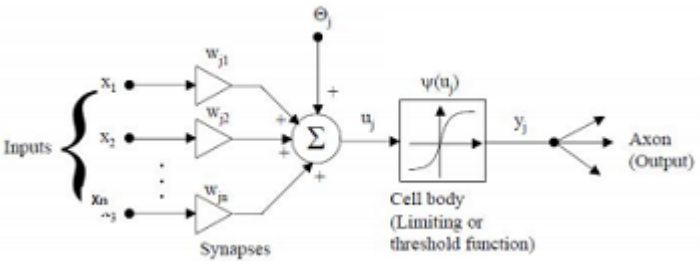

Fig. 2. Artificial Neuron Model [2].

strength of the signal. The weight indicates the connection

between the neurons. If the weight is positive then the neurons are connected and if it is negative the signal strength of the neurons are reduced and if the weight is 0 there are no connection between neurons. This weight changing is based upon the Activation function which will be discussed in this paper in the later sections. This process of adjusting weights of the ANN to get the required output is called learning or training. So the model can be represented as shown in the below equation. Here x is the input, y is the output, w is

$$y_j = \psi\left(\sum_{i=1}^{n} w_{ji}x_i + \Theta_j\right)$$

Fig. 3. ANN representation [2].

the weight, θ is the bias. Feedforward Networks, Feedback Networks and Lateral Networks are the architecture of ANN which is discussed below.

A. Feedforward Networks

In feedforward networks the information flows in the forward direction. There are no feedbacks returned within the network.

- Input Layer - The number of neurons in this layer corresponds to the number of inputs in the network. The nodes in this layer are passive nodes which means that this nodes do not take part in the signal modification i.e, all neurons will be active and there are no neurons which will be in off state. It only tranmits the information to the next layer.
- Hidden Layer - This has random number of layers and random number of neurons. The number depends upon the dataset. The nodes in this layer are active which means they take part in the signal modification.
- Output Layer - As input layer, here the number of neurons in the output layer is the number of output in the output layer. The nodes in this layer is active as well.

Fig. 4 explains a sample image of Feedforward Network.

B. Feedback Networks

In the feedback networks as the name indicates the output of this network is directly or indirectly connected to the input and the results are sent as a feedback. Basically the system will get trained from this as it knows what will be the output using this weight value and bias values. Based on the output it changes the values and it recomputes again. Here the input and the output layer are linear but the hidden layer is decided by the Activation functions which will be discussed in the upcoming sections in this paper. Fig. 5 represents a sample image of Feedback Networks.

C. Lateral Networks

In the Lateral networks the neurons are interconnected in every layer. The input neurons are connected to other input neurons. The hidden are connected only to other hidden

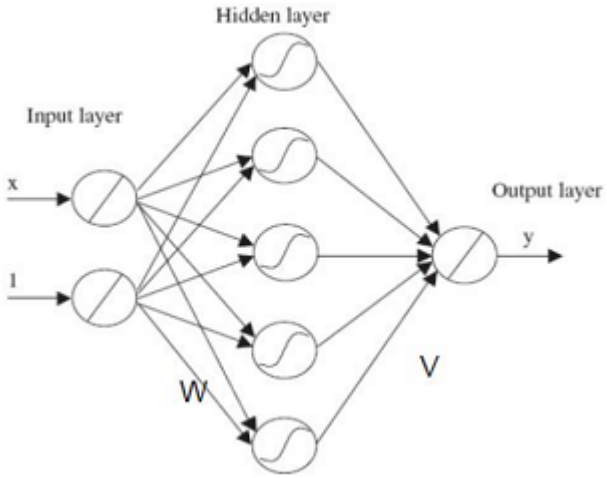

Fig. 4. Feedforward Network [2].

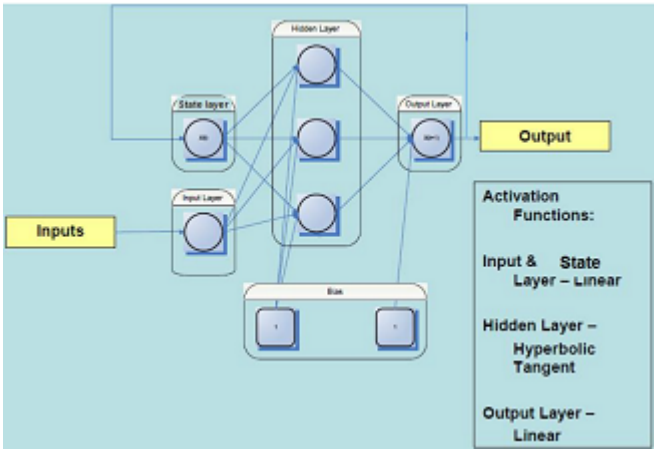

Fig. 5. Jordan Recurrent Network [2].

neurons. The output neurons are connected to the other output neurons. This can be thought as a combination of both Feedforward and Feedback Networks as this is connected to each other but in the same layer and in the forward direction. Fig. 6 represents a sample image of Lateral Network.

III. ANN Learning Methods

There are three different learning methods in ANN. Supervised learning, Unsupervised Learning and Reinforced Learning. The below figure displays the algorithms that are used for the learning methods.

A. Supervised Learning

As the name indicates, the learning is under supervision. The machine is given an input and their desired output and it has to learn and create a model based on the given output. We will be talking more about gradient descent and backprobagation in the next sections.

- Gradient Descent - It aims to find the weight values which minimises the error in the next epoch. Backprobagation

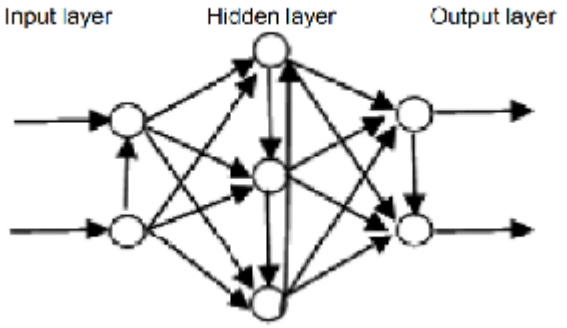

Fig. 6. Lateral Network [2].

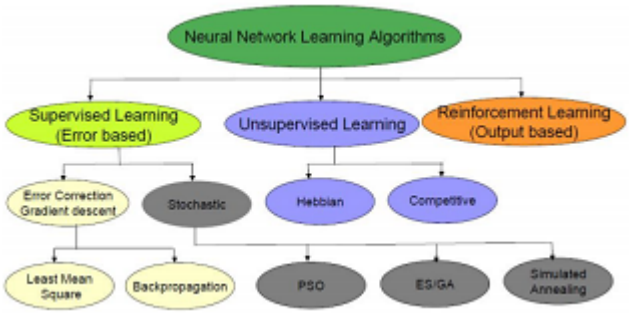

Fig. 7. ANN Learning Methods [2].

training algorithm is based on Gradient Descent. Backprobagation Algorithm is most commonly used method in training neural network. It comes under supervised machine learning algorithms with feedforward architecture. It is one of the most frequently utilized neural network techniques for Classification and Prediction. Here once the output is calculated the output is matched with the desired output and the difference between the targeted output and the output obtained on propagated back to the layers. As the flow again starts it changes the weight of the neurons/nodes in the hidden layers. This cycle of going forward from input to output and from output to input is called epoch. This cycle stops until the output obtained is matched with the targeted output with some tolerance.

B. Unsupervised Learning

As the name implies, unsupervised learning is a learning without any supervision. It can also be stated as the learning experience in adulthood. Adults learn and do the actions based on their past experience and knowledge. Unsupervised learning is similar to it. Hebb's rule helps the neural network or neuron assemblies to remember specific patterns much like memory. From that stored knowledge, similar sort of incomplete or spatial patterns could be recognized. This is even faster than the delta rule or the backprobagation algorithm because there is no repetitive presentation and training of input-output pairs [2].

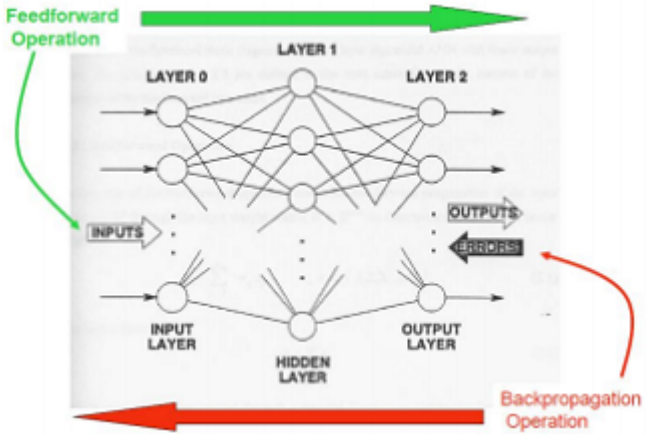

Fig. 8. Feedforward neural network with Gradient Descent [2].

C. Reinforcement Learning

This model does not give the machine a desired output but tells the machine whether the output is correct or not. A reward is given for a correct answer computed and a penalty is given for for wrong answer.

IV. Activation Function

Activation Function are mainly used in Neural Network for introducing a non-linear transformation in it. A Neural Network without an activation function is just a linear regression model [3]. Activation function decides whether a neuron should be activated or not. It does the non-linear transformation to the input making it capable to learn and perform complex tasks. Below is the detailed description of different types of activation function and their graphical representation.

A. Binary / Heavisied Step Function

Binary / Heavisied Step function can be used as an activation function for binary classifier. For example, to say if the patient has cancer(1) or not(0) or to classify the email as spam(1)or not(0). So the result of this function is either 0 or 1. This function is not useful when there are multiple classes in the input. As the Gradient is 0 for all x, during backprobagation process the weight and biases dont update.

- Equation : f(x) = 1, x $\geq$ 0 or 0 if x $<$ 0
- Range : There is no range in this. The value is either 0 or 1.
- Gradient : f'(x) = 0 for all x.

Fig. 9 is the graphical representation of the Binary Step function / Heaviside Step Function

B. Linear Function

Linear function might be an ideal task where interpretability is highly desired. Although this doesn't become zero after differentiation but it is a constant. So during backprobagation process the weight and biases don't get affected.

- Equation : f(x) = ax

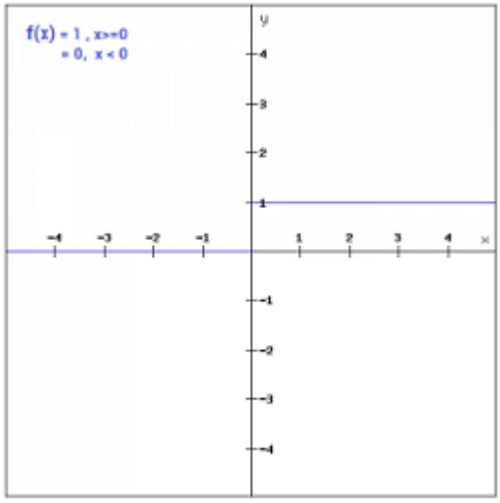

Fig. 9. Binary/Heaviside Step Function [3].

- Range : -inf to +inf
- Gradient : f'(x) = a.

Fig. 10 is the graphical representation of the Linear Function. It gives an example if a is 4.

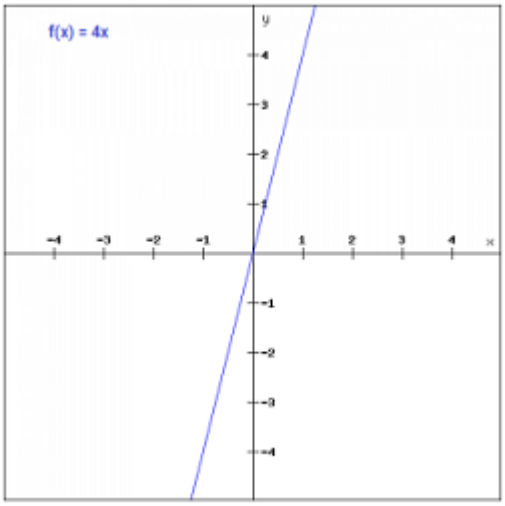

Fig. 10. Linear Function [3].

C. Sigmoid Function

Sigmoid function is one of the most widely used non linear activation function. As this is a non-linear function which means when there are multiple neurons their output will be non-linear as well. The gradient values are significant in the range of -3 to 3. As the gradient value approaches to zero the network is not really learning. Additionally the output of the sigmoid function will always have the same sign. This can be addressed by changing the sigmoid function which is what happens in tanh function.

- Equation : f(x) = 1/(1+e^x)
- Range : Value is from 0 to 1.
- Gradient : f'(x) = 1 - Sigmoid(x).

Fig. 11 is the graphical representation of the Sigmoid Function Function. As you can see in the figure this is a smooth S-shaped function and is continuously differentiable.

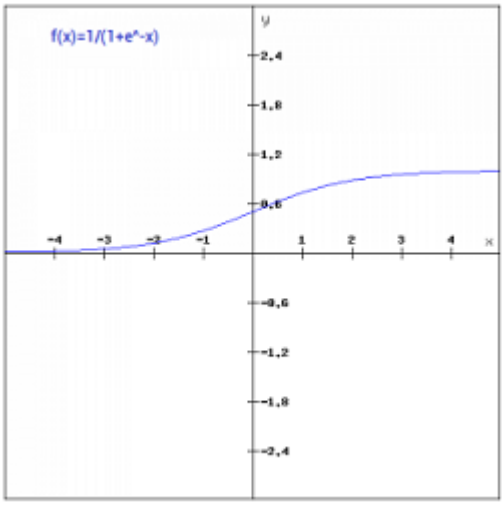

Fig. 11. Sigmoid Function [3].

D. Tanh Function

Tanh function is very simliar to the Sigmoid function except there is a symmetry between the origin. Thus the two input layers will not use the same sign. It helps in centering the data by bringing close to zero. Usually Tanh is preferred over Sigmoid function.

- Equation : f(x) = 2Sigmoid(2x)-1
- Range : -1 to 1.
- Gradient : f'(x) = 1 - f(x)$\hat{2}$(x).

Fig. 12 is the graphical representation of the Tanh Function Function.

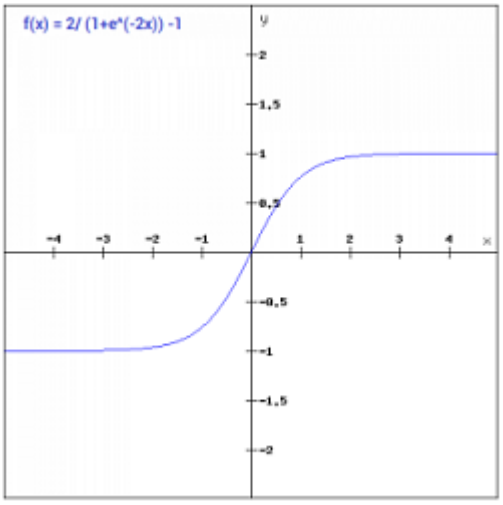

Fig. 12. Tanh Function [3].

E. ReLU Function

ReLU stands for Rectified Linear Unit. This is another non-linear function which is deeply used in the deep learning domain. The main advantage of using ReLU function is that it does not activate all the neurons at the same time. It gives an output of x if x is positive otherwise it is zero. Since only certain neurons are activated, the ReLU function is more computationally efficient compared to sigmoid and tanh function.One limitation is the gradient value is 0 in the negative side of the graph so during the backprobagation process the weight and biases are not updated. This can create dead neurons which can never be activated again. This problem is hanldled in leaky ReLU which is discussed in the next section.

- Equation : f(x) = max(0, x)
- Range : 0 to inf
- Gradient : f'(x) = 1 if x $\geq$ 0 or 0 if x = 0.

Fig. 13 is the graphical representation of the ReLU Function Function.

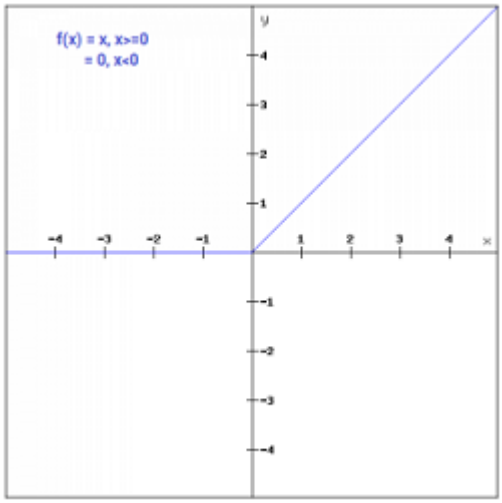

Fig. 13. ReLU Function [3].

F. Leaky ReLU Function

Leaky ReLU function is nothing but an improved version of ReLU function. Leaky ReLU has a small linear component for x less than 0 so the gradient of this will not be zero and the neurons will not be deactivated. As we are implementing this the system will no longer encounter dead neurons in the negative regions.

- Equation : f(x) = 0.01x for x $<$ 0 and x for x $\geq$ 0.
- Range : -inf to inf
- Gradient : f'(x) = 1 if x $\geq$ 0 and 0.0.1 if x $<$ 0.

Fig. 14 is the graphical representation of the Leaky ReLU Function.

G. Parameterised ReLU Function

This is another activation function which solves the problem with Leaky ReLu. Here a constant variable is introduced when

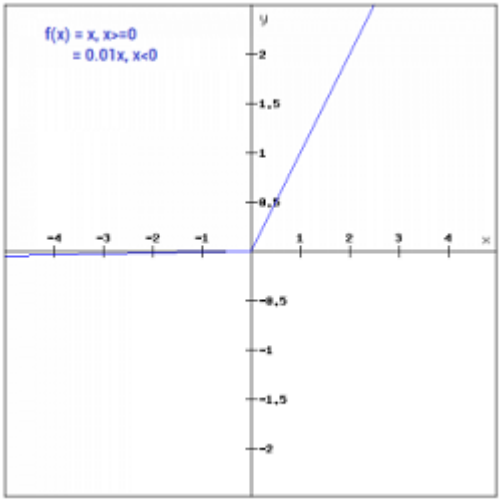

Fig. 14. Leaky ReLU Function [3].

x is less than 0. So, when the value is set to 0.01 it acts exactly like a Leaky ReLU function. In this case the constant is a trainable parameter. This is used when Leaky function fails to solve the problem of dead neuron which cannot be passed to the next layer.

- Equation : f(x) = x, x $\geq$ 0 and ax, x $<$ 0
- Range : -inf to inf.
- Gradient : f'(x) = 1, x $\geq$ 0 and a, x $<$ 0.

Fig. 15 is the graphical representation of the Parameterised ReLU Function. Apart from the above activation functions

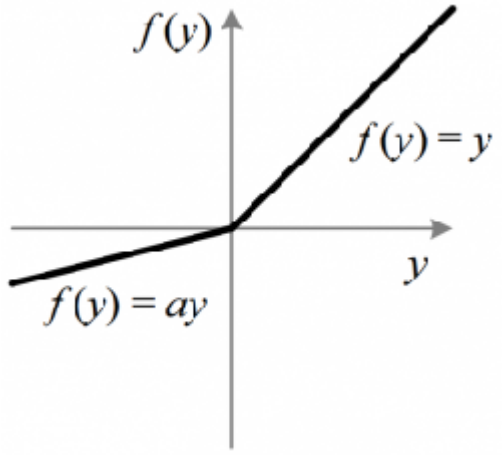

Fig. 15. Parameterised ReLU Function [3].

there are Exponential Linear Unit Function, Swish Function and Softmax Function. Sigmoid Function, ReLU function are most widely used Activation functions. Sigmoid functions show their greatness in case of classification. It is sometimes avoided due to the vanishing gradient problem as discussed. ReLU function is a general function which are used these days. As we encounter dead neurons this function is the best. As a rule of thumb, if we dont know what activation function

to choose, first choose ReLU then move to other activation function in case ReLU doesn't provide optimum results.

V. Features of ANN

ANN is a form of computer program in different ways. Some of its strength and limitation are [3]

- Adaptive Learning - The ability to replicate the human brain and learn from the mistakes to give a desired output.
- Self Organization - ANN can create its own organization while learning. As discussed above the it have a set of elements which runs and performs multiple tasks. A normal program can perform only fixed task.
- Parallel Operation - It involves parallel operation unlike a normal program.
- Fault Tolerance - ANN runs no matter what the data set is. It tries to find some desired output based on the data even if the data is not normal. The time between each process keeps changing although given the same sets of input.

VI. Applications of ANN

ANN are being used in may fields [1]

- Classification - The aim of this is to classify an object from an input. For example, image classification
- Pattern Matching - The aim of this is to match a pattern in the input given an input vector.
- Pattern Completion - The aim of this is to complete the missing parts of the given pattern in the input vector.
- Optimization - The aim of this is to find the optimal values in the Optimization problem.
- Control - A function or appropriate action based on the given input.
- Data Mining - The aim is to discover knowledge in data.

VII. Literature Review

Reference [4] focuses on classification of GPS satellites based on Neural Network, Genetic Algorithm and Particle Swarm Optimization. GDOP values which is Geometric dilution of precision factor is used to calculate a suitable satellite from a set of satellites. Theoretically satellites which are close together are not able to provide accurate information as compared to satellites which are widely separated from each other. Low GDOP values provide accurate GPS accuracy. The GDOP calculation has a time burden because it includes complicated transformation and inversion of measurement matrices. To tackle this problem neural network and evolutionary algorithms namely Genetic Algorithm and Particle Swarm Optimization is used. The GDOP value is spitted into 6 classes and the result of the neural network is the classification of the satellite based on the classes. This paper states that the highest classification accuracy has been achieved by the use of LevenBerg-Marquardt training algorithm with Genetic Algorithm.

Reference[5] demonstrates classification of hand movements by using Artificial Neural Network. It states that generally there are two approaches for extracting feature. 4 steps were involved in this research. The first step is to filter out the sEMG signals. The second step is to extract the features from the first step. Then the features were reduced to two dimension using Diffusion Map. Lastly seven different motions were classified by Artificial neural Network, Gustafson Kessel Algorithm. Finally their classifications were compared and this experiment shows that ANN outperforms Diffusion Map + Gustafson Kessel Algorithm. For feature extraction 8 time domain and 2 frequency domain were used. For ANN Feedforward Backprobagation architecture is used. 20 set of hidden neuron and 30 set of hidden neuron were used and the research shows that the accuracy was greater when used 20 hidden neurons.

Reference[6] presents a network based Automatic Fingerprints Classification Algorithm. This paper states that the fingerprints can be generally classified into 5 kinds: left loop, right loop, whorl, tented arch and arch. As a first step fingerprint features are extracted and are directed into the input layer and then the weight coefficients are are studied based on genetic algorithm. This research has an accuracy upto 93.12%. Three layer forward feedback network is used for this research. And the activation function is represented by f(x) = C - J, J $<$ C and 0, J $\geq$ C. This study used 100 fingerprints for the weight coefficent training. Then 480 original fingerprints are used for testing. The fingerprints are classified based on the above discussed kinds.

Reference[7] shows weakness and drawbacks of a password authentication scheme using Neural Networks for Multiserver Architecture. It states that in 2001, Li et al. proposed a password authentication scheme for multiserver architecture by using pattern classification scheme based on neural networks. In Li et al. multierver architecture there are three participants which are users, servers and system administrator. This neural network used backprobagation network. First it encodes the password and then normalizes it. It selects a secret key which is securely shared between SA and Servers. This paper states that this method is vulnerable for attacks and if the user chooses a password he cannot change the password for after registration. In addition since the password is encrypted with a key users who choose the same password will have same identity. So even if the user deletes his login or if he is removed from the server, he can still login back again. So this paper states that Li et al, deserves some further researches to improve the way of applying neural networks for password authentication.

Reference[8] focuses on Hyperspectral Feature Selection and Classification with a RBF-based Novel Double Parallel Feedforward Neural network and Evolution Algorithm. Band Selection is a very important preprocessing procedure for analysis of hyperspectral data which helps in reducing the great amount of time for computation. This paper states that in recent years feature selection using Neural Network such as Multi-Layer Forward neural Network (MLFNN), Radial Basis Function Neural Network (RBFNN) and Double Parallel Feedforward Neural Network (DPFNN) becomes a promising method for dimensionality reduction. This research uses

Improved DPFNN (IDPFNN) where it is the RBFNN not MLFNN to keep the parallel connection with the Single-Layer Feedforward Neural Network (SLFNN). A total of 1245 samples were used for training and 6232 samples were used for testing. As RBFNN is very good in local optimization it helps in better performance of IDPFNN.

VIII. Conclusion

This paper demonstrates about Artificial Neural Networks, their elements, architecture, types. It touches some topics on ANN learning methods and covers Backprobagation algorithm. Then it focuses on Activation function of ANN and why is it used and some of their types. Some features and application of ANN are listed in this study. 5 research papers were randomly selected for this study and are explained in brief in the Literature Review section. From this study we can learn that ANN makes the system to basically mimic like a human brain. It consists of processing units called neurons. Input layer, Hidden Layer and Output layer are the elements present in any Neural Network. Backprobagation algorithm is most commonly used method in training neural network. It comes under supervised machine learning algorithms and frequently used techniques for classification and prediction. An epoch is nothing but the cycle of going forward from input to output and then going backward from output to input after calculating the weight. Activation function introduces a non-linear transformation in neural network. It decides whether a neuron/node in the hidden layer should be activated or not. Different types of Activation function their Equation, Range Gradient and their uses and limitations are discussed in this paper.

References

[1] Graduate School of Science and Technology, Kumamoto University - Japan cs.kumamoto-u.ac.jp

[2] Dishashree Gupta. (2020) Fundamentals of Deep Learning – Activation Functions and When to Use Them? analyticsvidhya.com

[3] Shiruru, Kuldeep. (2016). AN INTRODUCTION TO ARTIFICIAL NEURAL NETWORK. International Journal of Advance Research and Innovative Ideas in Education. 1. 27-30.

[4] U. Baspinar, H. Selcuk Varol and K. Yildiz, "Classification of hand movements by using artificial neural network," 2012 International Symposium on Innovations in Intelligent Systems and Applications, Trabzon, 2012, pp. 1-4.

[5] H. Azami, M. D. Soltani and I. Tavakkolnia, "An intelligent approach for classification of GPS satellites based on neural network, genetic algorithm and particle swarm optimization," 2016 International Conference for Students on Applied Engineering (ICSAE), Newcastle upon Tyne, 2016, pp. 70-74.

[6] X. Li, G. Wang and X. Lu, "Neural Network Based Automatic Fingerprints Classification Algorithm," 2010 International Conference of Information Science and Management Engineering, Xi'an, 2010, pp. 94-96.

[7] Wei-Chi Ku, "Weaknesses and drawbacks of a password authentication scheme using neural networks for multiserver architecture," in IEEE Transactions on Neural Networks, vol. 16, no. 4, pp. 1002-1005, July 2005.

[8] R. Huang and L. Zhou, "Hyperspectral feature selection and classification with a RBF-based novel Double Parallel Feedforward Neural Network and evolution algorithms," 2009 4th IEEE Conference on Industrial Electronics and Applications, Xi'an, 2009, pp. 673-676.