**ASSIGNMENT NO. 2**

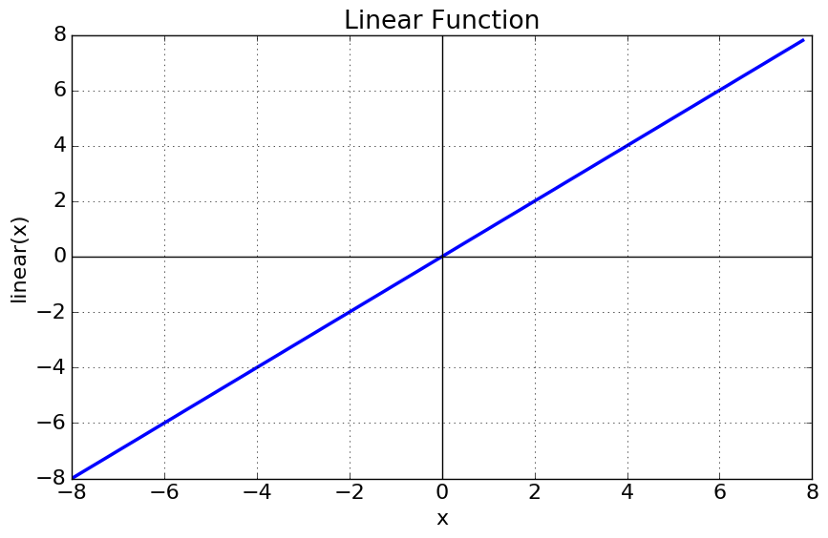
# **Q1. List various activation functions in ANN**

The Activation Functions can be basically divided into 2 types:

1. Linear Activation Function
2. Non-linear Activation Functions

Linear or Identity Activation Function

As you can see the function is a line or linear. Therefore, the output of the functions will not be confined between any range.



Equation: f(x) = x

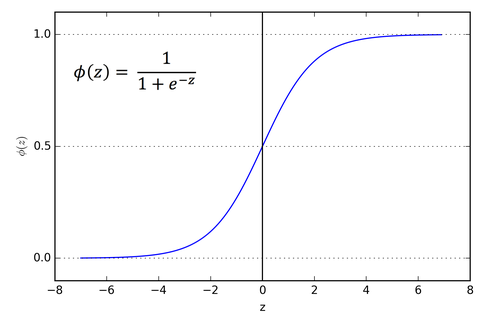
Range: (-∞ to ∞)

It doesn’t help with the complexity or various parameters of usual data that is fed to the neural networks.

Non-linear Activation Function

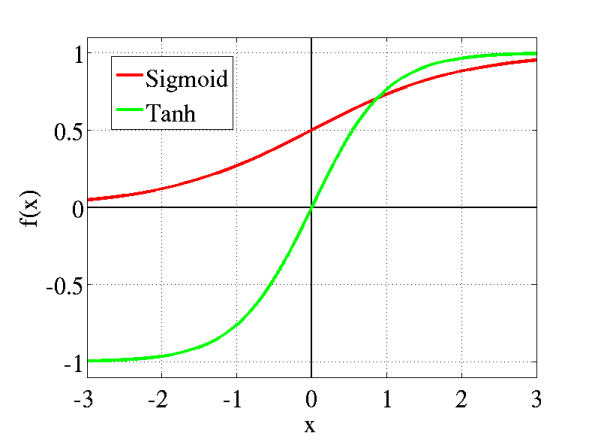
The Nonlinear Activation Functions are the most used activation functions. It makes it easy for the model to generalize or adapt with a variety of data and to differentiate between the output. The Nonlinear Activation Functions are mainly divided on the basis of their range or curves:

1. Sigmoid or Logistic Activation Function



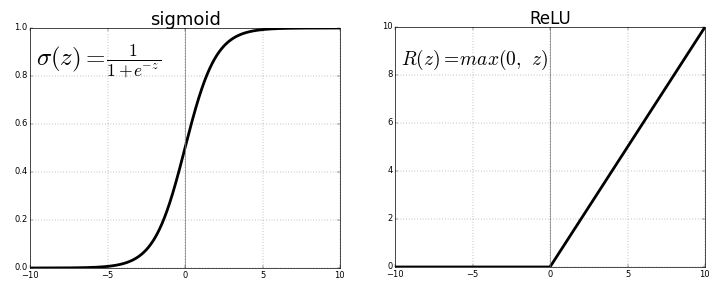
The Sigmoid Function curve looks like a S-shape. The main reason why we use sigmoid function is because it exists between (0 to 1). Therefore, it is especially used for models where we have to predict the probability as an output.

2. Tanh or hyperbolic tangent Activation Function



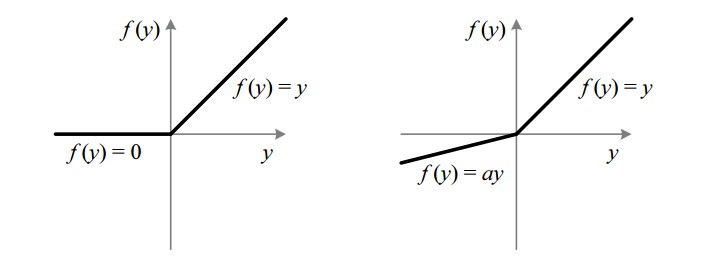
The tanh is also like logistic sigmoid but better. The range of the tanh function is from (-1 to 1). tanh is also sigmoidal (s - shaped). The advantage is that the negative inputs will be mapped strongly negative and the zero inputs will be mapped near zero in the tanh graph.

3. ReLU (Rectified Linear Unit) Activation Function



The ReLU is the most used activation function in the world right now.Since, it is used in almost all the convolutional neural networks or deep learning. As you can see, the ReLU is half rectified (from bottom). f(z) is zero when z is less than zero and f(z) is equal to z when z is above or equal to zero.

4. Leaky ReLU



It is an attempt to solve the dying ReLU problem. The leak helps to increase the range of the ReLU function. Usually, the value of a is 0.01 or so. When a is not 0.01 then it is called Randomized ReLU.

# **Q2. Difference between ANN, CNN and RNN**

|  | **ANN** | **CNN** | **RNN** |
| --- | --- | --- | --- |
| Type of Data | Tabular Data, Text Data | Image Data | Sequence data |
| Parameter Sharing | No | Yes | Yes |
| Fixed Length Input | Yes | Yes | No |
| Recurrent Connections | No | No | Yes |
| Spatial Relationship | No | Yes | No |
| Performance | ANN is considered to be less powerful than CNN, RNN. | CNN is considered to be more powerful than ANN, RNN. | RNN includes less feature compatibility when compared to CNN. |
| Application | Facial recognition and Computer vision. | Facial recognition, text digitization and Natural language processing. | Text-to-speech conversions. |
| Main advantages | Having fault tolerance, Ability to work with incomplete knowledge. | High accuracy in image recognition problems, Weight sharing. | Remember each and every information, Time series prediction. |
| Disadvantages | Hardware dependence, Unexplained behavior of the network. | Large training data needed, don’t encode the position and orientation of objects. | Gradient vanishing, exploding gradient. |

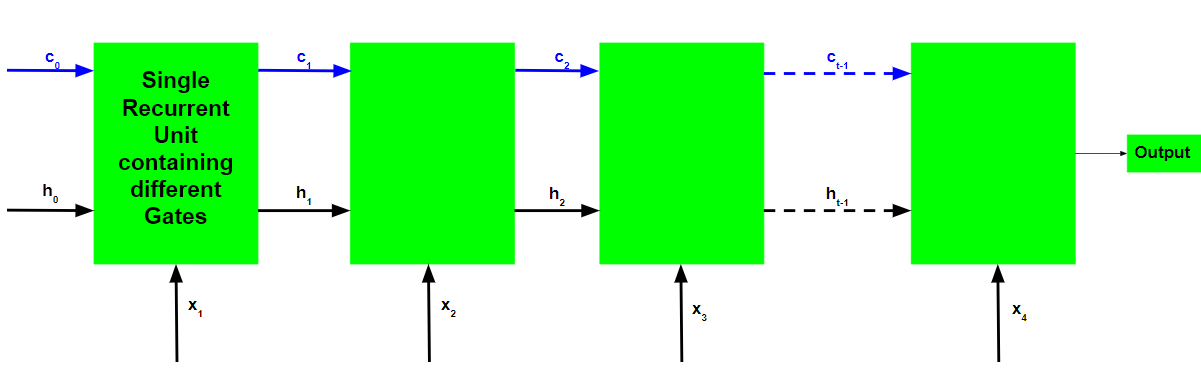
# **Q3. Explain Long Short-Term Memory Networks**

To solve the problem of Vanishing and Exploding Gradients in a Deep Recurrent Neural Network, many variations were developed. One of the most famous of them is the Long Short Term Memory Network (LSTM). In concept, an LSTM recurrent unit tries to “remember” all the past knowledge that the network has seen so far and to “forget” irrelevant data. This is done by introducing different activation function layers called “gates” for different purposes. Each LSTM recurrent unit also maintains a vector called the Internal Cell State which conceptually describes the information that was chosen to be retained by the previous LSTM recurrent unit.

A Long Short Term Memory Network consists of four different gates for different purposes as described below:

1. Forget Gate (f): It determines to what extent to forget the previous data.
2. Input Gate (i): It determines the extent of information to be written onto the Internal Cell State.
3. Input Modulation Gate (g): It is often considered as a sub-part of the input gate and much literature on LSTM does not even mention it and assume it is inside the Input gate. It is used to modulate the information that the Input gate will write onto the Internal State Cell by adding non-linearity to the information and making the information Zero-mean.
4. Output Gate (o): It determines what output(next Hidden State) to generate from the current Internal Cell State.

The basic workflow of a Long Short Term Memory Network is similar to the workflow of a Recurrent Neural Network with the only difference being that the Internal Cell State is also passed forward along with the Hidden State.



1. Take input the current input, the previous hidden state, and the previous internal cell state.
2. Calculate the values of the four different gates by following the below steps:-

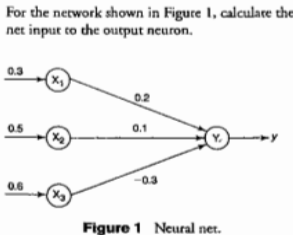
* For each gate, calculate the parameterized vectors for the current input and the previous hidden state by element-wise multiplication with the concerned vector with the respective weights for each gate.
* Apply the respective activation function for each gate element-wise on the parameterized vectors. Below given is the list of the gates with the activation function to be applied for the gate.

1. Calculate the current internal cell state by first calculating the element-wise multiplication vector of the input gate and the input modulation gate, then calculate the element-wise multiplication vector of the forget gate and the previous internal cell state and then adding the two vectors.



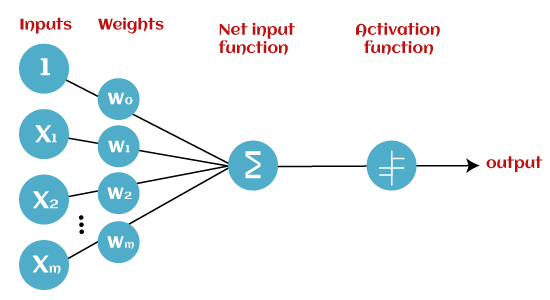
1. Calculate the current hidden state by first taking the element-wise hyperbolic tangent of the current internal cell state vector and then performing element-wise multiplication with the output gate.

# **Q4. Discuss in detail the Perceptron Networks and calculate the net input (yin) of the given network**



Perceptron is a Machine Learning algorithm for supervised learning of various binary classification tasks. Further, Perceptron is also understood as an Artificial Neuron or neural network unit that helps to detect certain input data computations in business intelligence.

The Perceptron model is also treated as one of the best and simplest types of Artificial Neural networks. However, it is a supervised learning algorithm of binary classifiers. Hence, we can consider it as a single-layer neural network with four main parameters, i.e., input values, weights and Bias, net sum, and an activation function.



Input Nodes or Input Layer

This is the primary component of Perceptron which accepts the initial data into the system for further processing. Each input node contains a real numerical value.

Wight and Bias

Weight parameter represents the strength of the connection between units. This is another most important parameter of Perceptron components. Weight is directly proportional to the strength of the associated input neuron in deciding the output. Further, Bias can be considered as the line of intercept in a linear equation.

Activation Function

These are the final and important components that help to determine whether the neuron will fire or not. Activation Function can be considered primarily as a step function.

Perceptron model works in two important steps as follows:

1. In the first step first, multiply all input values with corresponding weight values and then add them to determine the weighted sum. Mathematically, we can calculate the weighted sum as follows:

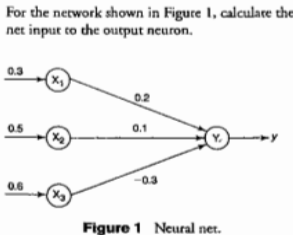
∑wi\*xi = x1\*w1 + x2\*w2 +…wn\*xn

Add a special term called bias 'b' to this weighted sum to improve the model's performance.

∑wi\*xi + b

1. In the second step, an activation function is applied with the above-mentioned weighted sum, which gives us output either in binary form or a continuous value as follows:

Y = f(∑wi\*xi + b)



Net Input = w1 \* x1 + w2 \* x2 + w3 \* x3 = (0.3 \* 0.2) + (0.5 \* 0.1) + (0.6 \* -0.3) = 0.06 + 0.05 - 0.18 = **-0.07**

# **Q5. Implement AND function using perceptron network for bipolar inputs and bipolar targets. (Initial values are w1=w2=b=0, learning rate=1, threshold=0.2)**

AND Gate

| **X1** | **X2** | **Target** |
| --- | --- | --- |
| 1 | 1 | 1 |
| 1 | -1 | -1 |
| -1 | 1 | -1 |
| -1 | -1 | -1 |

Step 1

Initialized all weights and bias as 0

w1 = w2 = w3 = b = 0

Step 2

For first input,

x1 = 1, x2 = 1 and t = 1

w1 = 0, w2 = 0 and b = 0

Calculating the net input,

yin = b + w1 \* x1 + w2 \* x2 = 0

Step 3

Using activation function (Sigmoid)

y = f(yin) = { 1 ( yin > ) ; 0 ( - <= yin <= ) ; -1 ( yin < )

y = f(yin) = f(0) = 0

Since t = 1 and y = 0, so t y

Step 4

Updating the weights,

w1new = w1old + \* t \* x1 = 0 + 1 \* 1 \* 1 = 1

w2new = w2old + \* t \* x2 = 0 + 1 \* 1 \* 1 = 1

bnew = bold + \* t = 0 + 1 \* 1 = 1

Step 5

Performing above steps for each input value,

| **EPOCH 1** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | | **t** | **yin** | **y** | **New Weights** | | |
| x1 | x2 |  |  |  | w1 | w2 | b |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 1 | -1 | -1 | 1 | 1 | 0 | 2 | 0 |
| -1 | 1 | -1 | 2 | 1 | 1 | 1 | -1 |
| -1 | -1 | -1 | -1 | -1 | 1 | 1 | -1 |

| **EPOCH 2** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | | **t** | **yin** | **y** | **New Weights** | | |
| x1 | x2 |  |  |  | w1 | w2 | b |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | -1 | -1 | -1 | -1 | 0 | 0 | 0 |
| -1 | 1 | -1 | -1 | -1 | 0 | 0 | 0 |
| -1 | -1 | -1 | -3 | -1 | 0 | 0 | 0 |

So final weights and bias are

* w1 = 1
* w2 = 1
* b = -1

# **Q6. Explain Applications of Data Science**

In Search Engines

The most useful application of Data Science is Search Engines. As we know, when we want to search for something on the internet, we mostly use Search engines like Google, Yahoo, Safari, Firefox, etc. So Data Science is used to get Searches faster.

For Example, When we search something suppose “Data Structure and algorithm courses ” then at that time on the Internet Explorer we get the first link of GeeksforGeeks Courses. This happens because the GeeksforGeeks website is visited most in order to get information regarding Data Structure courses and Computer related subjects. So this analysis is Done using Data Science, and we get the Topmost visited Web Links.

In Transport

Data Science also entered into the Transport field like Driverless Cars. With the help of Driverless Cars, it is easy to reduce the number of Accidents.

For Example, In Driverless Cars the training data is fed into the algorithm and with the help of Data Science techniques, the Data is analyzed like what is the speed limit in Highway, Busy Streets, Narrow Roads, etc. And how to handle different situations while driving etc.

In Finance

Data Science plays a key role in Financial Industries. Financial Industries always have an issue of fraud and risk of losses. Thus, Financial Industries needs to automate risk of loss analysis in order to carry out strategic decisions for the company. Also, Financial Industries uses Data Science Analytics tools in order to predict the future. It allows the companies to predict customer lifetime value and their stock market moves.

For Example, In Stock Market, Data Science is the main part. In the Stock Market, Data Science is used to examine past behavior with past data and their goal is to examine the future outcome. Data is analyzed in such a way that it makes it possible to predict future stock prices over a set timetable.

In E-Commerce

E-Commerce Websites like Amazon, Flipkart, etc. use data Science to make a better user experience with personalized recommendations.

For Example, When we search for something on the E-commerce websites we get suggestions similar to choices according to our past data and also we get recommendations according to most buy the product, most rated, most searched, etc. This is all done with the help of Data Science.

In Health Care

In the Healthcare Industry data science acts as a boon. Data Science is used for:

* Detecting Tumor.
* Drug discoveries.
* Medical Image Analysis.
* Virtual Medical Bots.
* Genetics and Genomics.
* Predictive Modeling for Diagnosis etc.

In Image Recognition

Currently, Data Science is also used in Image Recognition. For Example, When we upload our image with our friend on Facebook, Facebook gives suggestions Tagging who is in the picture. This is done with the help of machine learning and Data Science. When an Image is Recognized, the data analysis is done on one’s Facebook friends and after analysis, if the faces which are present in the picture matched with someone else profile then Facebook suggests us auto-tagging.

In Targeting Recommendation

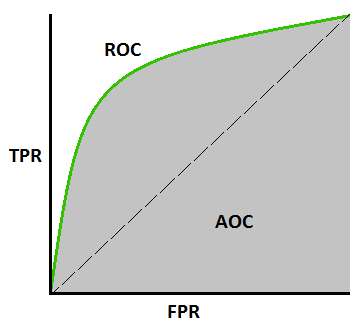
Targeting Recommendation is the most important application of Data Science. Whatever the user searches on the Internet, he/she will see numerous posts everywhere. This can be explained properly with an example: Suppose I want a mobile phone, so I just Google search it and after that, I changed my mind to buy it offline. Data Science helps those companies who are paying for Advertisements for their mobile. So everywhere on the internet, in social media, in the websites, in the apps everywhere I will see the recommendation of that mobile phone which I searched for. So this will force me to buy online.

In Delivery Logistics

Various Logistics companies like DHL, FedEx, etc. make use of Data Science. Data Science helps these companies to find the best route for the Shipment of their Products, the best time suited for delivery, the best mode of transport to reach the destination, etc.

# **Q7. Comparing classifiers based on ROC Curve.**

ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. By analogy, the Higher the AUC, the better the model is at distinguishing between patients with the disease and no disease. The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.



For a perfect classifier the ROC curve will go straight up the Y axis and then along the X axis. A classifier with no power will sit on the diagonal, whilst most classifiers fall somewhere in between. The AUC for a classifier with no power, essentially random guessing, is 0.5, because the curve follows the diagonal. The AUC for that mythical being, the perfect classifier, is 1.0. Most classifiers have AUCs that fall somewhere between these two values.

The AUC can be used to compare the performance of two or more classifiers. A single threshold can be selected and the classifiers’ performance at that point compared, or the overall performance can be compared by considering the AUC. Most published reports compare AUCs in absolute terms: “Classifier 1 has an AUC of 0.85, and classifier 2 has an AUC of 0.79, so classifier 1 is clearly better“. It is, however, possible to calculate whether differences in AUC are statistically significant.