**Assignment 4.2**

**1) Explain different layers in neural network?**

Ans)

Basically, there are 3 different types of layers in a neural network

**Input Layer**

It is a layer where all the inputs are fed to the Neural Network or model.

**Hidden Layers**

Hidden Layers are the layers which are in between input and output layers which are used for processing inputs. A Neural Network can have more than one Hidden layer.

**Output Layer**

It is the Last layer of the Neural Network. It takes the information from Hidden layers after data processing. Thus, the processed data is made available at output layer.

The four most common types of neural network layers are Fully connected, Convolution, Deconvolution, and Recurrent, and below you will find what they are and how they can be used.

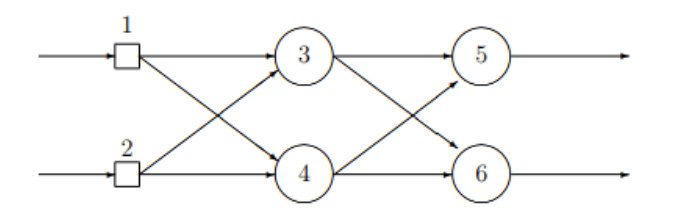
Fully connected layers connect every neuron in one layer to every neuron in the next layer. Fully connected layers are found in all different types of neural networks ranging from standard neural networks to convolutional neural networks (CNN).

A Convolution Layer is an important type of layer in a CNN. Its most common use is for detecting features in images, in which it uses a filter to scan an image, a few pixels at a time, and outputs a feature map that classifies each feature found.

A Deconvolution Layer is a transposed convolution process that effectively upsamples data to a higher resolution. This can include image data and/or feature maps generated from a convolution layer, or other types of data. For image data, the upsampling resolution output by deconvolution may be the same as the original input image, or may be different.

A Recurrent Layer includes a “looping” capability such that its input consists of both the data to analyze as well as the output from a previous calculation performed by that layer. Recurrent layers form the basis of recurrent neural networks (RNNs), effectively providing them with memory (i.e., maintain a state across iterations), while their recursive nature makes RNNs useful for cases involving sequential data like natural language and time series. They’re also useful for mapping inputs to outputs of different types and dimensions.

2)



Table

Description automatically generated A picture containing text

Description automatically generated

A picture containing graphical user interface

Description automatically generated

**Calculate the output of the network for each of the input patterns?**

Ans)

First input 0,0

Y3’ = activation(y3) = activation(0) = 1

Y4’ = activation(y4) = activation(0) = 1

Y5’ = activation(y5) = activation(w35\*y3’+w45\*y4’) = activation(1-1)=1

Y6’ = activation(y6) = activation(w36\*y3’+w46\*y4’) = activation(-1+1)=1

So the outputs for 0,0 is 1,1

Second input 1,0

Y3’ = activation(-2\*1+3\*0) =activation(-2)=0

Y4’ = activation(4\*1 + -1\*0) = activation(4) = 1

Y5’ = activation(1\*0 + -1\*1) = activation(-1) =0

Y6’ = activation(-1\*0+1\*1) = activation(1) = 1

So the outputs for 1,0 are 0,1

**3) What will happen if the learning rate of a Neural Network is too high?**

Ans)

Learning rate (λ) is one such hyper-parameter that defines the adjustment in the weights of our network with respect to the loss gradient descent. It determines how fast or slow we will move towards the optimal weights

The Gradient Descent Algorithm estimates the weights of the model in many iterations by minimizing a cost function at every step.

Learning rate (λ) is one such hyper-parameter that defines the adjustment in the weights of our network with respect to the loss gradient descent. It determines how fast or slow we will move towards the optimal weights

The Gradient Descent Algorithm estimates the weights of the model in many iterations by minimizing a cost function at every step.

But with a low learning rate, he would consider the white dog as an outlier and would continue to believe that all dogs are black.

And if the learning rate is too high, he would instantly start to believe that all dogs are white even though he has seen more black dogs than white ones.

The point is it’s’ really important to achieve a desirable learning rate because:

both low and high learning rates results in wasted time and resources

A lower learning rate means more training time

more time results in increased cloud GPU costs

a higher rate could result in a model that might not be able to predict anything accurately