Assignment 1 –Vrunda Patel

1. What is the function of a summation junction of a neuron? What is threshold activation function?

   Ans 1. Binary step function is a threshold-based activation function which means after a certain threshold neuron is activated and below the said threshold neuron is deactivated

2. What is a step function? What is the difference of step function with threshold function?

   Ans 2. Step Function is one of the simplest **kinds of activation** functions. In this, we consider a threshold value and if the value of net input says y is greater than the threshold then the neuron is activated, A threshold transfer function is **used to quantify the output of a neuron in the output layer**

3. Explain the McCulloch–Pitts model of neuron.

   Ans3. This is **simplified model of real neurons**, known as Threshold Logic Unit. A set of synapsesc (i.e connections) brings the activations from the other neurons. A processing unit sums the inputs, the applies the non-linear activation funcation (i.e threshold / transfer function).

4. Explain the ADALINE network model.

   Ans4. Adaline architecture network. ADALINE (Adaptive Linear Neuron) is one of many artificial neural network methods that have only one output It is an early single-layer artificial neural network and the name of the physical device that implemented this network. It is based on the McCulloch–Pitts neuron It uses bipolar activation function. It uses delta rule for training to minimize the Mean-Squared Error (MSE) between the actual output and the desired/target output. The weights and the bias are adjustable.

5. What is the constraint of a simple perceptron? Why it may fail with a real-world data set?

   Ans 5. Perceptron networks have several limitations. First, the output values of a perceptron can take on only one of two values (0 or 1) due to the hard-limit transfer function. Second, perceptrons can only classify linearly separable sets of vectors. Perceptrons only represent linearly separable problems. They fail to converge if the training/ real world examples are not linearly separable.

6. What is linearly inseparable problem? What is the role of the hidden layer?

   Ans6. Clearly not all decision problems are linearly separable: they cannot be solved using a linear decision boundary. Problems like these are termed linearly inseparable.

   Hidden layer in artificial neural networks a layer of neurons, whose output is connected to the inputs of other neurons and therefore is not visible as a network output.

   For Exapmle first hidden layer detects pixels of light and dark, they are not very useful for face recognition, but they are extremely useful to identify edges and simple shapes on the second hidden layer. The third hidden layer knows how to comprise more complex objects from edges and simple shapes. Finally, at the end, the output layer will be able to recognize a human face with some confidence.

7. Explain XOR problem in case of a simple perceptron.

   Ans 7. XOR is where if one is 1 and other is 0 but not both. ... A "single-layer" perceptron can't implement XOR. The reason is because the classes in XOR are not linearly separable. You cannot draw a straight line to separate the points (0,0),(1,1) from the points (0,1),(1,0). An XOR (exclusive OR gate) is a digital logic gate that gives a true output only when both its inputs differ from each other.

8. Design a multi-layer perceptron to implement A XOR B.

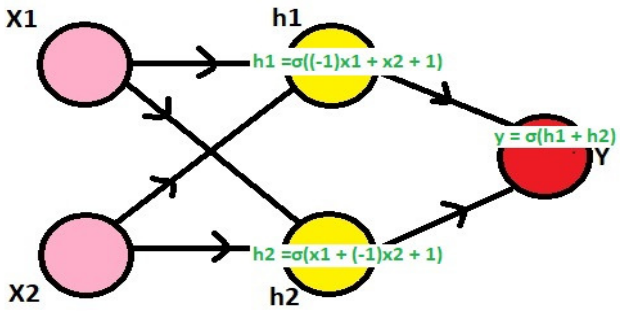
   Ans 8. we create a Multi Layered Perceptron (or MLP). We call this extra layer as the Hidden layer. To build a perceptron, we first need to understand that the XOr gate can be written as a combination of AND gates, NOT gates and OR gates in the following way:

   a XOr b = (a AND NOT b)OR(bAND NOTa)

we need to observe that our inputs are 0s and 1s. To make it a XOr gate, we will make the h1 node to perform the (x2 AND NOT x1) operation, the h2 node to perform (x1 AND NOT x2) operation and the y node to perform (h1 OR h2) operation. The NOT gate can be produced for an input a by writing (1-a), the AND gate can be produced for inputs a and b by writing (a.b) and the OR gate can be produced for inputs a and b by writing (a+b). Also, we'll use the sigmoid function as our activation function σ, i.e., σ(x) = 1/(1+e^(-x)) and the threshold for classification would be 0.5, i.e., any x with σ(x)>0.5 will be classified as 1 and others will be classified as 0.

Now, since we have all the information, we can go on to define h1, h2 and y. Using the formulae for AND, NOT and OR gates, we get:

1. h1 = σ((1-x1) + x2) = σ((-1)x1 + x2 + 1)
2. h2 = σ(x1 + (1-x2)) = σ(x1 + (-1)x2 + 1)
3. y = σ(h1 + h2) = σ(h1 + h2 + 0)



9. Explain the single-layer feed forward architecture of ANN.

Ans 9. A single-layer neural network represents the most simple form of neural network, in which there is only one layer of input nodes that send weighted inputs to a subsequent layer of receiving nodes, or in some cases, one receiving node.

10. Explain the competitive network architecture of ANN.

Ans :Competitive learning is a form of unsupervised learning in artificial neural networks, in which nodes compete for the right to respond to a subset of the input data. A variant of Hebbian learning, competitive learning works by increasing the specialization of each node in the network.

11. What is the Ann Architecture explain with type?

Ans: ANN architecture is based on the structure and function of the biological neural network. Similar to neurons in the brain, ANN also consists of neurons which are arranged in various layers

The three major types of Ann Architecture are

- Artificial Neural Networks (ANN)
- Convolution Neural Networks (CNN)
- Recurrent Neural Networks (RNN) RNN captures the sequential information present in the input data i.e. dependency between the words in the text while making predictions

ANN can be used to solve problems related to:

- Tabular data
- Image data
- Text data

We can use recurrent neural networks to solve the problems related to:

- Time Series data
- Text data
- Audio data

- CNN learns the filters automatically without mentioning it explicitly. These filters help in extracting the right and relevant features from the input data
- Though convolutional neural networks were introduced to solve problems related to image data, they perform impressively on sequential inputs as well.
- CNN captures the spatial features from an image. Spatial features refer to the arrangement of pixels and the relationship between them in an image. They help us in identifying the object accurately, the location of an object, as well as its relation with other objects in an image
- CNN also follows the concept of parameter sharing. A single filter is applied across different parts of an input to produce a feature map

	MLP	RNN	CNN
Data	Tabular data	Sequence data (Time Series,Text, Audio)	Image data
Recurrent connections	No	Yes	No
Parameter sharing	No	Yes	Yes
Spatial relationship	No	No	Yes
Vanishing & Exploding Gradient	Yes	Yes	Yes

12. Consider a multi-layer feed forward neural network. Enumerate and explain steps in the backpropagation algorithm used to train the network.

- A multilayer feedforward neural network is an interconnection of perceptrons in which data and calculations flow in a single direction, from the input data to the outputs. The number of layers in a neural network is the number of layers of perceptrons
- The algorithm is used to effectively train a neural network through a method called chain rule. In simple terms, after each forward pass through a network, backpropagation performs a backward pass while adjusting the model's parameters (weights and biases)

Below are the steps involved in Backpropagation:

- Step – 1: Forward Propagation.
- Step – 2: Backward Propagation.
- Step – 3: Putting all the values together and calculating the updated weight value.
- Backpropagation is the essence of neural network training. It is the method of fine-tuning the weights of a neural network based on the error rate (i.e. loss) obtained in the previous epoch (i.e., iteration). Proper tuning of the weights allows you to reduce error rates and make the model reliable by increasing its generalization

13.What are the advantages and disadvantages of neural networks?

Advantages of Artificial Neural Networks (ANN)

- Problems in ANN are represented by attribute-value pairs.
- ANNs are used for problems having the target function, the output may be discrete-valued, real-valued, or a vector of several real or discrete-valued attributes.
- ANN learning methods are quite robust to noise in the training data. The training examples may contain errors, which do not affect the final output.
- It is used where the fast evaluation of the learned target function required.
- ANNs can bear long training times depending on factors such as the number of weights in the network, the number of training examples considered, and the settings of various learning algorithm parameters.

Disadvantages of Artificial Neural Networks (ANN)

- Hardware Dependence:
- Artificial Neural Networks require processors with parallel processing power, by their structure.
- For this reason, the realization of the equipment is dependent.
- Unexplained functioning of the network:
- This the most important problem of ANN.
- When ANN gives a probing solution, it does not give a clue as to why and how.
- This reduces trust in the network.
- Assurance of proper network structure:
- There is no specific rule for determining the structure of artificial neural networks.
- The appropriate network structure is achieved through experience and trial and error.
- The difficulty of showing the problem to the network:
- ANNs can work with numerical information.
- Problems have to be translated into numerical values before being introduced to ANN.
- The display mechanism to be determined will directly influence the performance of the network.
- This is dependent on the user's ability.
- The duration of the network is unknown:
- The network is reduced to a certain value of the error on the sample means that the training has been completed.
- The value does not give us optimum result

14. Write short notes on any two of the following:

Single-layer feed forward network

In this type of network, we have only two layers input layer and output layer but the input layer does not count because no computation is performed in this layer. The output layer is formed when different weights are applied on input nodes and the cumulative effect per node is taken.

Artificial Neural Network is capable of learning any nonlinear function. Hence, these networks are popularly known as Universal Function Approximators. ANNs have the capacity to learn weights that map any input to the output.One of the main reasons behind universal approximation is the activation function. Activation functions introduce nonlinear properties to the network. This helps the network learn any complex relationship between input and output

Gradient descent

What is gradient descent explain with example?
Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. It is basically used for updating the parameters of the learning model. ... But if the number of training examples is large, then batch gradient descent is computationally very expensive

Gradient descent is an optimization algorithm which is commonly-used to train machine learning models and neural networks. Training data helps these models learn over time, and the cost function within gradient descent specifically acts as a barometer, gauging its accuracy with each iteration of parameter updates.

Recurrent networks

Recurrent neural networks (RNN) are the state of the art algorithm for sequential data and are used by Apple's Siri and and Google's voice search. It is the first algorithm that remembers its input, due to an internal memory, which makes it perfectly suited for machine learning problems that involve sequential data
Recurrent neural networks (RNN) are the state of the art algorithm for sequential data and are used by Apple's Siri and and Google's voice search. It is the first algorithm that remembers its input, due to an internal memory, which makes it perfectly suited for machine learning problems that involve sequential data
It is commonly used in speech recognition and natural language processing. Recurrent neural networks recognize data's sequential characteristics and use patterns to predict the next likely scenario