

- ① Loss function
- ② Supervised learning Algorithm
- ③ Mean Absolute Error
- ④ Logistic Sigmoid function
- ⑤ True

### Short Questions

① Gradient Descent is a popular optimization technique in machine learning and deep learning, and it can be used with most, if not all, of the learning algorithms. A sub gradient is the slope of a function. It means the degree of change of a variable in response to the changes of another variable.

② The following are the need for ANN's

① To develop a Fault Tolerant system

② Ability to learn

③ Ability to model Non-linear & Complex problems

④ Classification and Clustering Analysis.

④ ③ The following are types:-

- ① Supervised Training
- ② Unsupervised Training
- \* ③ Reinforced Training.

⑤ Types:-

- ① Binary Cross Entropy loss
- ② Categorical Cross Entropy loss
- ③ Hinge loss
- ④ Kullback Leibler Divergence loss

Long Q's

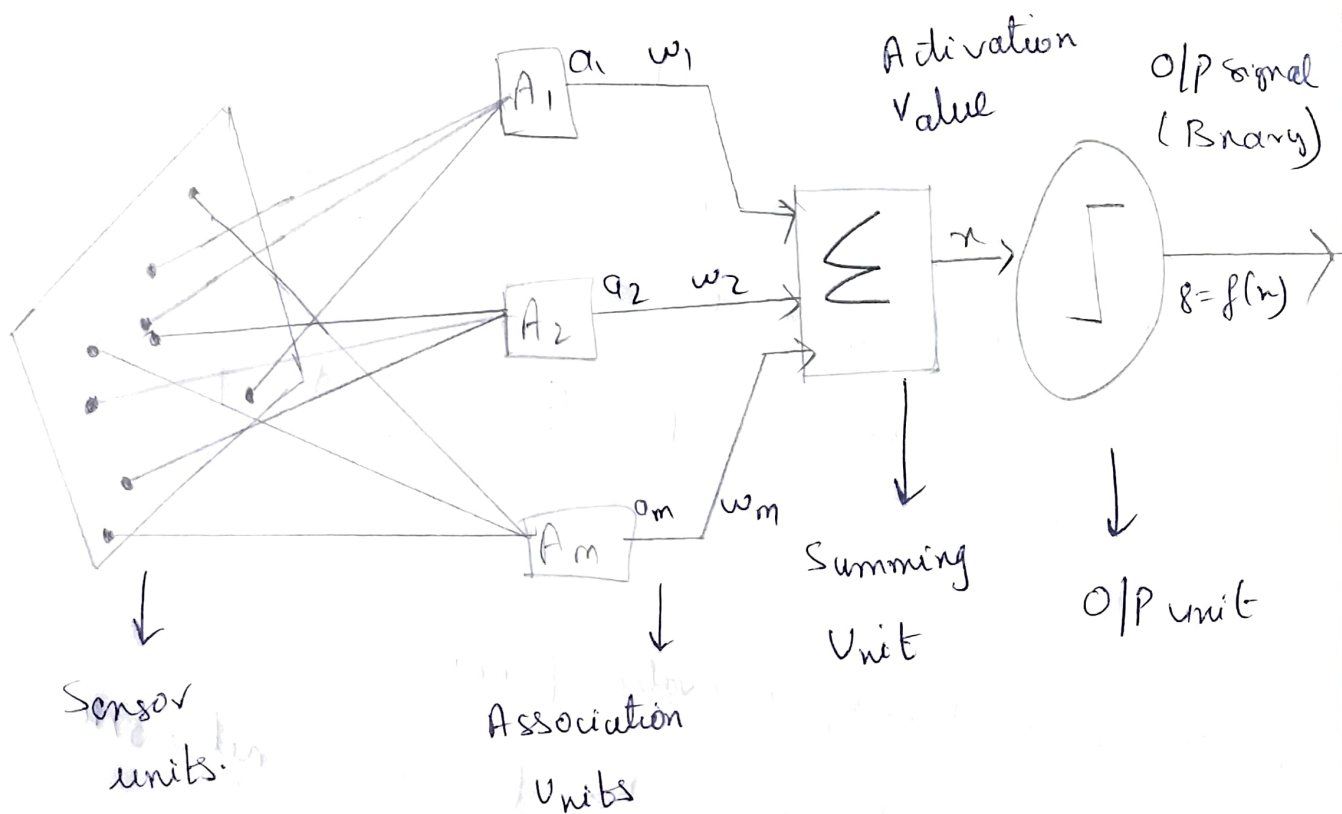
① Q. would suggest the horn single layer perceptron model.

→ ~~Perceptron~~ Perceptron network is capable of programming classification into two / more categories.

→ It is not trained using the perceptron learning rule.

→ Here we will use bipolar neurons. and the simplest architecture that could do the job consists of  $N$  input neurons, an output layer with a single O/P neuron.

Input (Adjustable weights)



### Algorithm

- ① Initialize the weights to some small random values near to zero
- ② Apply the I/P: O/P training patterns (vector pairs)
- ③ Calculate the summing part value  $Net = \sum a_i w_i - \theta$
- ④ Apply Activation function & calculate the O/P.  

$$F(Net) = \begin{cases} 1 & \text{if } Net \geq \theta \\ 0 & \text{if } Net < \theta \end{cases}$$
- ⑤ Calculate the error  $\delta = b_i - S_i$  if error is present then update the weight for that link using

$$\Delta w_j = \eta (f_i - \text{sgn}(w_j^T a)) a_j$$

$$= \eta (b_i - s_j a_j) \quad \text{for } j = 1, 2, \dots, m$$

⑥ Similarly change the bias values

$$\text{New Bias} = \text{Old Bias} + \text{change in bias}$$

⑦ Now repeat from step 3

⑧ Check for error is present repeat the process, else stop.

### Cons

- ① Uses only binary activation function
- ② Can be used only for linear networks
- ③ Since Training time is more.

### Pros

- ① Easy to set up and train
- ② Neuron network can be linked to statistical models which means the model can be used to share Covariance.

### Short Q's

- ④ Shallow neural network gives us basic idea about deep neural network which consist of only 1 or 2 hidden layers.