PA10. Krishnaraj Thadesar

29th NOV 2023

AI Lab - Assignment - 7

(A) Title: Write a program to implement Naive Bayes classifier / neural network classifier.

(A) Aim: Write a program on Neural Network classification.

(A) Objective: To study Neural Network classifier

(A) Platform: Linux x86_64

(T) Theory:

(1) Neural Networks:

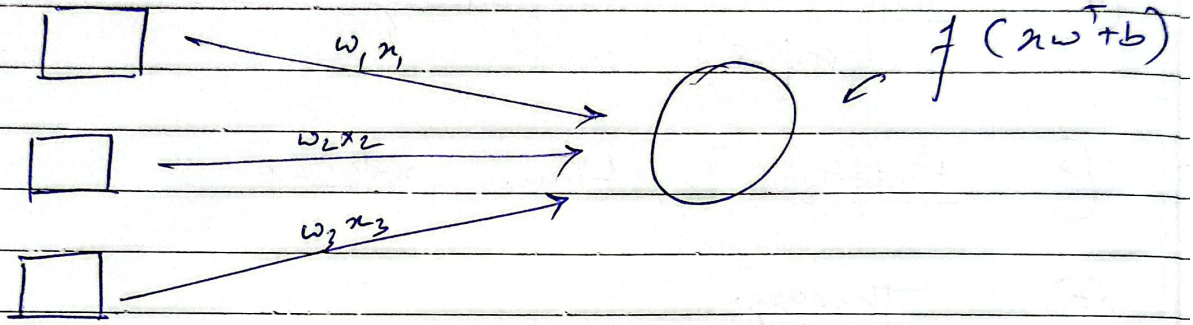
→ A neural network is a computational model inspired by the human brain structure & functioning. It's a composition of interconnected nodes, called neurons, organized into layers.

→ Each neuron processes data, performs computations, and generates an output signal.

→ Through training, neural networks learn to recognize patterns and relationships in data by adjusting the strengths of connections between neurons, known as weights, to produce desired output for given inputs.

(2) Structure and function of a single neuron in a neural network.

→ Structure: A single neuron comprises of fundamental units. It receives input signals (representing features or data) through weighted connections, aggregates them, applies an activation function to produce an output signal.



→ Function: Process information by transforming inputs into meaningful outputs. They play a crucial role in the learning pattern and relationships within data through the adjustment of connection weights during training; aiding in tasks like pattern recognition, classification and prediction.

(3) Basic Architecture of a feed forward Neural Network.

→ A feedforward neural network is a type of artificial neural network that has the following basic architecture.

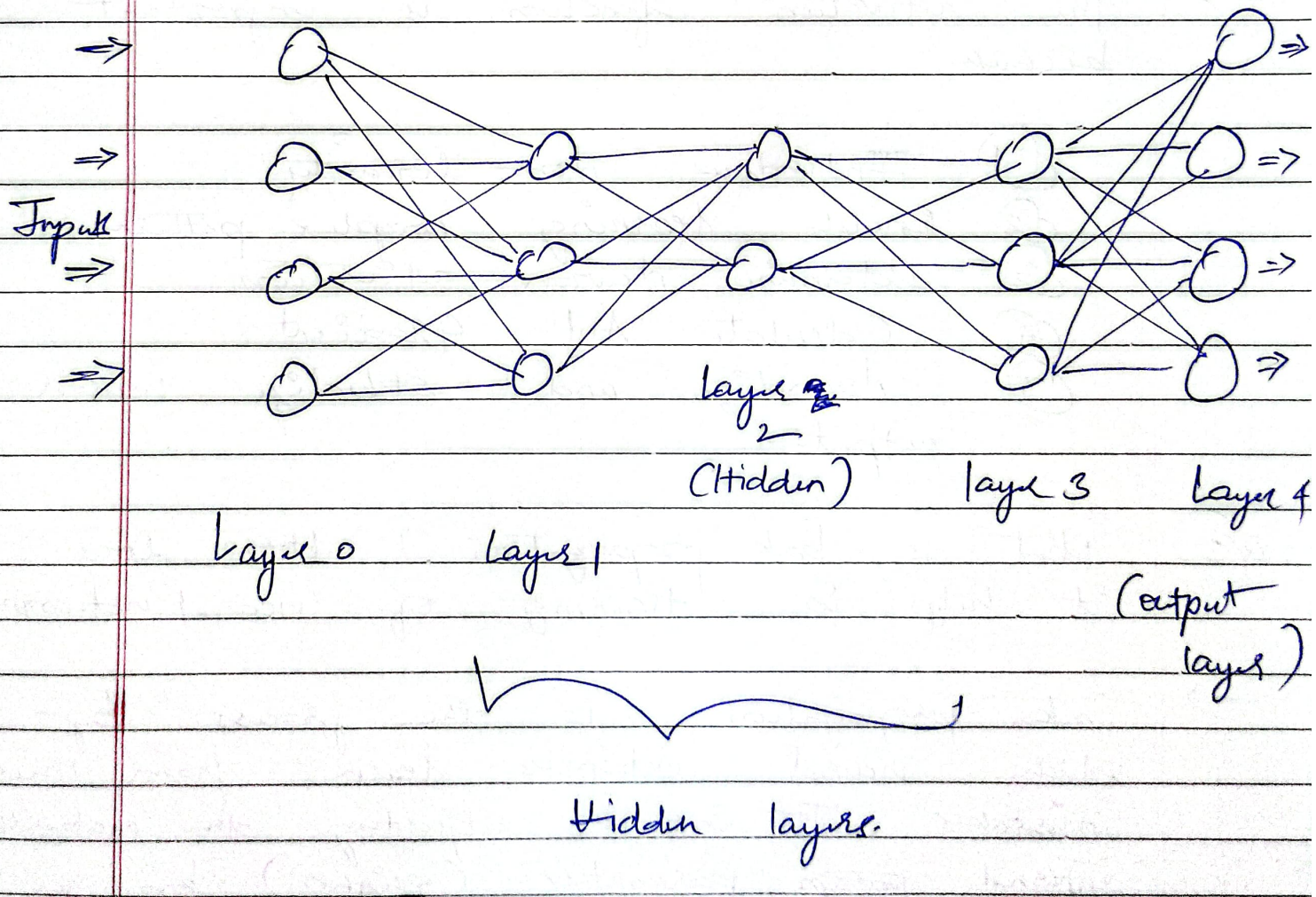
* An input layer that receives the data and passes it through the ~~next~~ next layer.

* One or more hidden layers that process the data and perform non-linear transformation

* An output layer that produces the final output of the network.

→ Each layer consists of units called neurons which are connected by weights that are representing the strength of the connection.

→ The data flows in one direction from the input layer to the output layer, without any feedback loop.



FAQ's

Q1) What is the activation function, and why is it important in neural networks?

→ The activation function decides if the neuron should be activated or not by calculating the weighted sum & further adding bias to it. The purpose of the activation function is to introduce non-linearity into the output of the neuron.

→ The activation function is important because:

1. Introduce Non-linearity
2. Enable learning complex patterns
3. Determine Neuron activation
4. Calculate And Gradient
5. Normalize and stabalize neuron outputs.

Q2) What is back propagation? How does it help in training of neural networks?

→ Back propagation is the process by which neural networks learn from their mistakes. It involves adjusting the networks internal parameters (weights) by calculating how much they contributed to the prediction error and updating them to minimize that error.

Working / Training:

1. Forward Pass: Input data is passed through the neural network to produce a prediction.
2. Error Calculation: Networks output is compared to expected result.
3. Backward Pass: Error is propagated backward, calculating the contribution of each weight to that error.
4. Adjust weight: To minimize error.
5. Iteration: Repeat 1 → 4, so as to refine networks predictions towards multiple cycles.
6. Convergence: Through this process the network learns to make better predictions by minimizing errors iteratively.

Q. Define and compare different loss functions used in training neural networks. ?

→ 1. Mean Square Error:

→ Measure the average of squared differences between predictions and actual values.

→ sensitive to outliers due to squaring

(2) Mean Absolute Error:

→ Computes the average of absolute differences between predictions and actual values.

→ Less sensitive to outliers compared to MSE.

(3) Binary Cross Entropy:

→ Used for binary classification, evaluates the performance of model providing probability outputs.

→ Penalizes significant deviations between predicted probabilities and actual classes.

(4) Categorical Cross Entropy:

→ Ideal for multi class identification tasks measures difference between predicted and actual class distributions.

→ Penalizes misclassification by assigning higher loss for larger deviation across multiple classes.