RAMA UNIVERSITY

www.ramauniversity.ac.in

FACULTY OF EGINEERING AND TECHNOLOGY

Soft Computing

LECTURE -06

Umesh Kumar Gera
Assistant Professor
Computer Science & Engineering

OUTLINE

- Back propagation network
- Mathematical Model of unsupervised Learning
- Why used unsupervised learning
- Types of Unsupervised learning
- Advantages and Disadvantages of Unsupervised Learning
- Multiple Choice Question

BACKGROUND OF BACK PROPAGATION

Why Back Propagation?

- ❑ There are various limitation of single level ANN. We can solve very limited classes and task.
- ❑ Multi layer feedback network(MLFN) solved various restriction of single level network, but MLFN cant be solve adjusting the weight of hidden layer weight.
- ❑ Rumelhart Hinton and Williams in 1986 proposed method to solve this problem which named is back propagation method.
- ❑ for designing neural network we have to initialized some initial weight randomly but no one is like god so that you have selected that value which gives incorrect output.
- ❑ So to overcome these type of error we have to change some parameter like weight so that error become minimum.
- ❑ one way to train your model is called back propagation.

WHAT IS BACK PROPOGATION?

Background of Back propagation algorithm

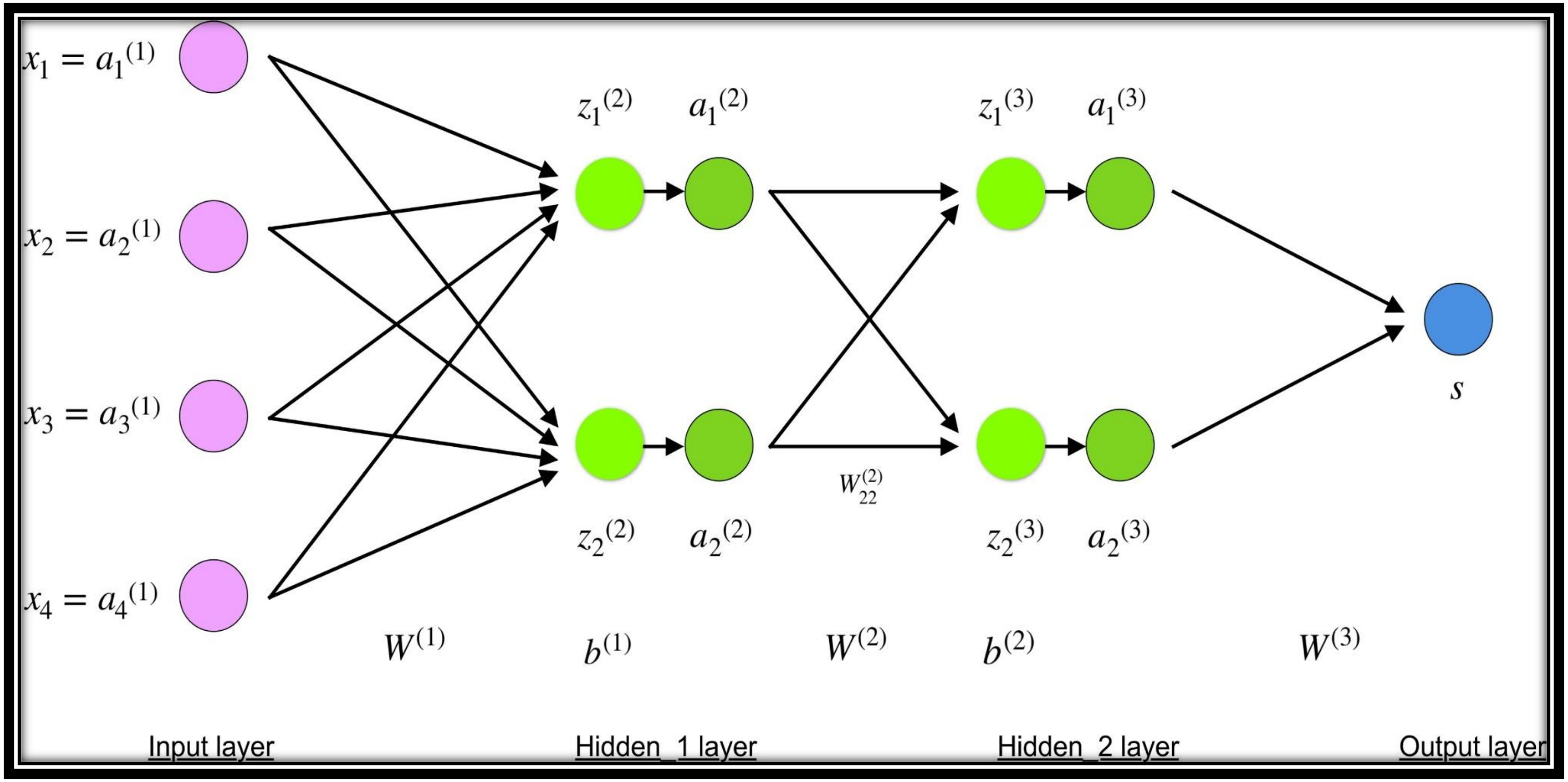
- Back propagation algorithm is probably the most fundamental building block in a neural network.
- It was first introduced in 1960s and almost 30 years later (1989) popularized by Rumelhart, Hinton and Williams in a paper called "Learning representations by back-propagating errors".
- The algorithm is used to effectively train a neural network through a method called chain rule. In simple terms, after each forward pass through a network, back propagation performs a backward pass while adjusting the model's parameters (weights and biases).
- Here I will discuss mathematical process of training and optimizing a simple 4-layer neural network.
- I believe this would help the reader understand how back propagation works as well as realize its importance.

RAMA UNIVERSITY UTTAR PRADESH KANPUR

WHAT IS BACK PROPOGATION?

Define the neural network model

The 4-layer neural network consists of 4 neurons for the input layer, 4 neurons for the hidden layers and 1 neuron for the output layer.



WHAT IS BACK PROPOGATION?

Input layer

The neurons, colored in purple, represent the input data. These can be as simple as scalars or more complex like vectors or multidimensional matrices.

$$\boxed{x_i = a_i^{(1)}, i \in 1,2,3,4}$$

Equation for input x_i

The first set of activations (a) are equal to the input values. NB: “activation” is the neuron’s value after applying an activation function. See below.

Hidden layers

The final values at the hidden neurons, colored in green, are computed using z^l — weighted inputs in layer l, and a^l— activations in layer l. For layer 2 and 3 the equations are:

l=2

$$\boxed{\begin{aligned} z^{(2)} &= W^{(1)}x + b^{(1)} \\ a^{(2)} &= f(z^{(2)}) \end{aligned}}$$

Equations for z^2 and a^2

RAMA UNIVERSITY UTTAR PRADESH KANPUR

WHAT IS BACK PROPOGATION?

Hidden layers

The neurons, colored in purple, represent the input data. These can be as simple as scalars or more complex like vectors or multidimensional matrices.

l=3

$$\boxed{\begin{aligned} z^{(3)} &= W^{(2)}a^{(2)} + b^{(2)} \\ a^{(3)} &= f(z^{(3)}) \end{aligned}}$$

Equations for z^3 and a^3

Here W^2 and W^3 are the weights in layer 2 and 3 while b^2 and b^3 are the biases in those layers.

RAMA
UTTAR PRADESH
KANPUR

Activations a^2 and a^3 are computed using an activation function f. Typically, this function f is non-linear and allows the network to learn complex patterns in data.

Looking carefully, you can see that all of x, z^2, a^2, z^3, a^3, W^1, W^2, b^1 and b^2 are missing their subscripts presented in the 4-layer network illustration above. The reason is that we have combined all parameter values in matrices, grouped by layers. This is the standard way of working with neural networks and one should be comfortable with the calculations.

HIDDEN LAYER CONTINUED

Let's pick layer 2 and its parameters as an example. The same operations can be applied to any layer in the network.

W^1 is a weight matrix of shape (n, m) where n is the number of output neurons (neurons in the next layer) and m is the number of input neurons (neurons in the previous layer). For us, n = 2 and m = 4.

$$\boxed{W^{(1)} = \begin{bmatrix} W_{11}^{(1)} & W_{12}^{(1)} & W_{13}^{(1)} & W_{14}^{(1)} \\ W_{21}^{(1)} & W_{22}^{(1)} & W_{23}^{(1)} & W_{24}^{(1)} \end{bmatrix}}$$

Equation for W^1

The first number in any weight's subscript matches the index of the neuron in the next layer (in our case this is the Hidden_2 layer) and the second number matches the index of the neuron in previous layer (in our case this is the Input layer).

x is the input vector of shape (m, 1) where m is the number of input neurons. For us, m = 4.

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Equation for x

HIDDEN LAYER CONTINUED

b^1 is a bias vector of shape (n , 1) where n is the number of neurons in the current layer. For us, n = 2.

$$b^{(1)} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \end{bmatrix}$$

Equation for b^1

Following the equation for z^2, we can use the above definitions of W^1, x and b^1 to derive "Equation for z^2

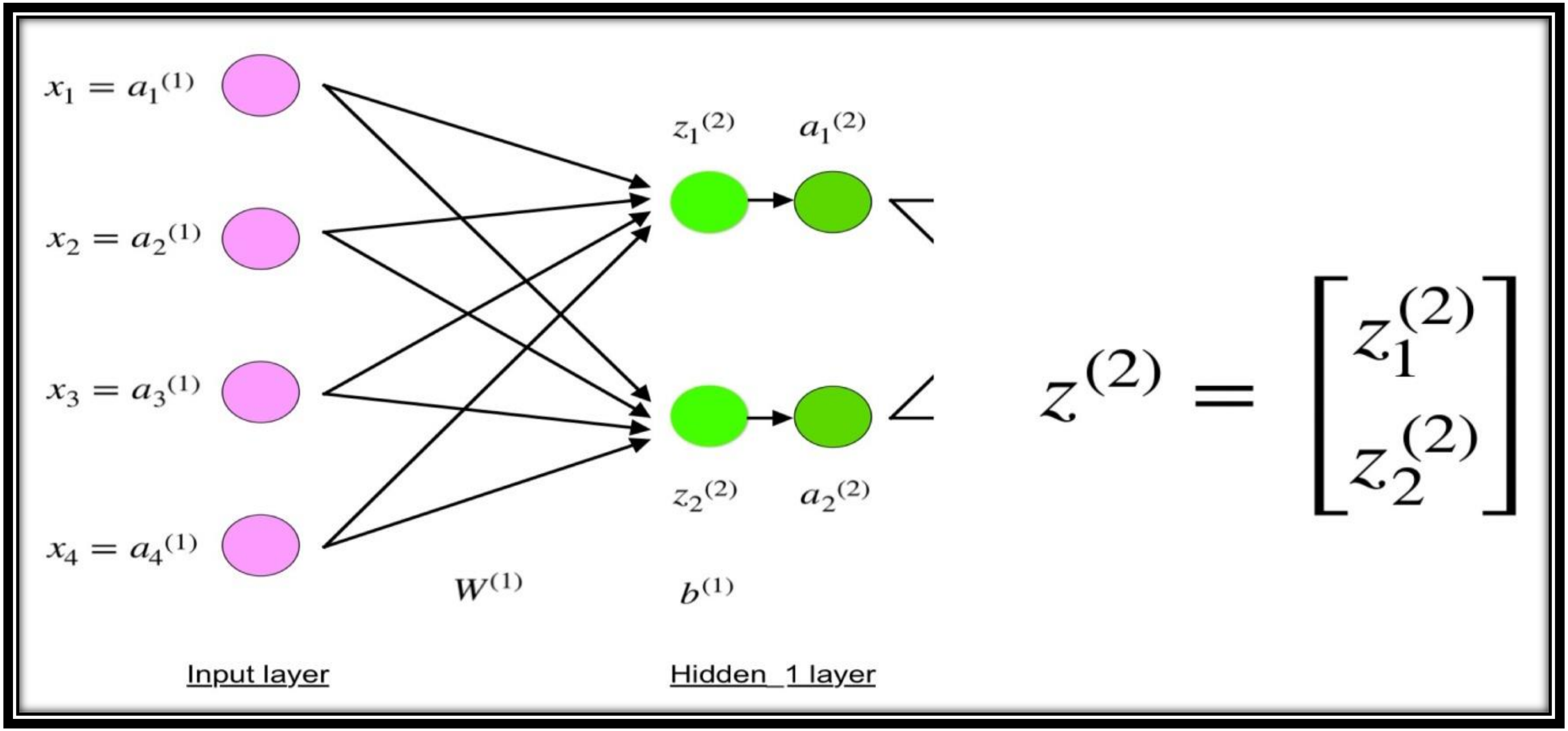
$$z^{(2)} = \begin{bmatrix} W_{11}^{(1)} x_1 + W_{12}^{(1)} x_2 + W_{13}^{(1)} x_3 + W_{14}^{(1)} x_4 \\ W_{21}^{(1)} x_1 + W_{22}^{(1)} x_2 + W_{23}^{(1)} x_3 + W_{24}^{(1)} x_4 \end{bmatrix} + \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \end{bmatrix}$$

Equation for Z^2

Now carefully observe the neural network illustration from above.

RAMA

HIDDEN LAYER CONTINUED



You will see that z² can be expressed using (z_1)² and (z_2)² where (z_1)² and (z_2)² are the sums of the multiplication between every input x_i with the corresponding weight (W_ij)¹.

This leads to the same "Equation for z² and proofs that the matrix representations for z², a², z³ and a³ are correct.

OUTPUT LAYER CONTINUED

Output layer

The final part of a neural network is the output layer which produces the predicated value. In our simple example, it is presented as a single neuron, colored in blue and evaluated as follows:

$$s = W^{(3)}a^{(3)}$$ Equation for output X

Forward propagation and evaluation

The equations above form network's forward propagation. Here is a short overview:

$$x = a^{(1)} \qquad \textit{Input layer}$$

$$z^{(2)} = W^{(1)}x + b^{(1)} \qquad \textit{neuron value at Hidden}_1 \textit{ layer}$$

$$a^{(2)} = f(z^{(2)}) \qquad \textit{activation value at Hidden}_1 \textit{ layer}$$

$$z^{(3)} = W^{(2)}a^{(2)} + b^{(2)} \qquad \textit{neuron value at Hidden}_2 \textit{ layer}$$

$$a^{(3)} = f(z^{(3)}) \qquad \textit{activation value at Hidden}_2 \textit{ layer}$$

$$s = W^{(3)}a^{(3)} \qquad \textit{Output layer}$$

HIDDEN LAYER CONTINUED

❑ The final step in a forward pass is to evaluate the predicted output *s* against an expected output *y*.

❑ The output y is part of the training dataset (x, y) where x is the input (as we saw in the previous section).

❑ Evaluation between s and y happens through a cost function. This can be as simple as MSE (mean squared error) or more complex like cross-entropy.

❑ We name this cost function C and denote it as follows:

$$C = cost(s, y)$$

Equation for cost function C

❑ were cost can be equal to MSE, cross-entropy or any other cost function.

❑ Based on C’s value, the model “knows” how much to adjust its parameters in order to get closer to the expected output y. This happens using the back propagation algorithm.

RAMA UNIVERSITY KANPUR

MULTIPLE CHOICE QUESTION

1. What is the objective of back propagation algorithm?

a) to develop learning algorithm for multilayer feed forward neural network

b) to develop learning algorithm for single layer feed forward neural network

c) to develop learning algorithm for multilayer feed forward neural network, so that network can be trained to capture the mapping implicitly

d) none of the mentioned

2. The back propagation law is also known as generalized delta rule, is it true?

a) yes

b) no

3. What is true regarding back propagation rule?

a) it is also called generalized delta rule

b) error in output is propagated backwards only to determine weight updates

c) there is no feedback of signal at nay stage

d) all of the mentioned

4. What is true regarding back propagation rule?

a) it is a feedback neural network

b) actual output is determined by computing the outputs of units for each hidden layer

c) hidden layers output is not all important, they are only meant for supporting input and output layers

d) none of the mentioned

RAMA UNIVERSITY UTTAR PRADESH KANPUR

REFERENCES

- https://towardsdatascience.com/understanding-backpropagation-algorithm-7bb3aa2f95fd
- https://www.sanfoundry.com/neural-networks-questions-answers-backpropagation-algorithm/