Neural Networks

Training Algorithm

- The algorithm which is commonly used now, to train an ANN is **Backpropagation Algorithm**.

Contd...

- Before understanding the working of the backpropagation algorithm, let us understand
  1. **How a NN is working in the forward direction**
  2. **The steps in the supervised learning of a NN**

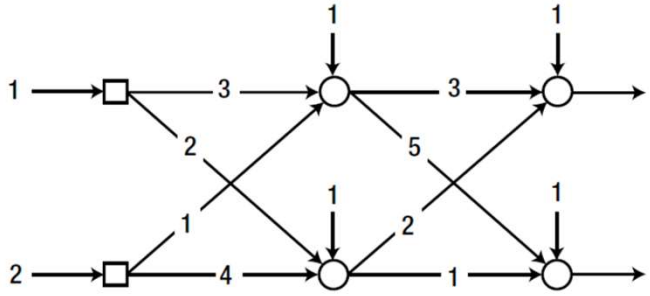
Contd...

1. How a NN is working in the forward direction

- In a feedforward NN
  - the signal enters the input layer,
  - passes thorough the hidden layers,
  - and, leaves through the output layer.
- In other words, the nodes on one layer receive the signal simultaneously and send the processed signal to the next layer at the same time.

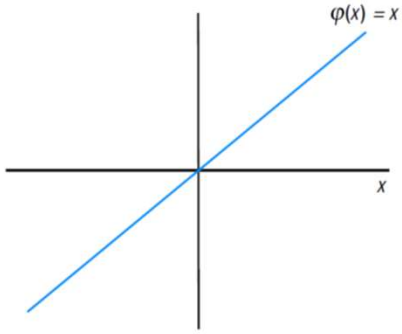
Contd...

- We follow a simple neural network to see how the input data is processed as it passes through the layers.



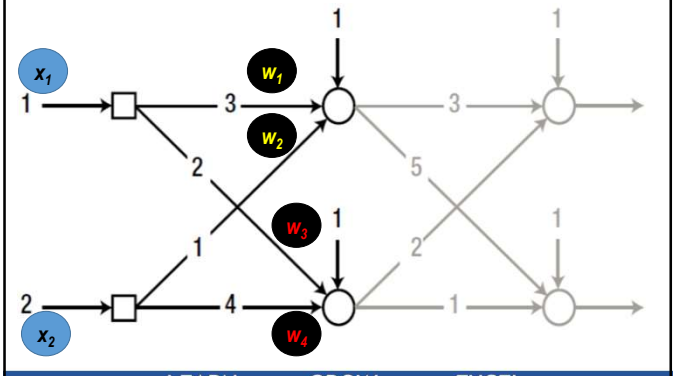
Contd...

- Just for convenience, the activation function applied to each node is assumed to be a **linear function**.



Contd...

- We **calculate the output from the hidden layers**, as follows.



Contd...

- The **first node of the hidden layer** calculates the output as follows…

$$\text{Weighted sum: } v = (3\times1) + (1\times2) + 1 = 6$$
$$\text{Output: } y = \varphi(v) = v = 6$$

- The **second node of the hidden layer** calculates the output as follows…

$$\text{Weighted sum: } v = (2\times1) + (4\times2) + 1 = 11$$
$$\text{Output: } y = \varphi(v) = v = 11$$

Contd...

- The weighted sum calculations can be combined in a matrix equation as follows…

$$v = \begin{bmatrix} 3\times1+1\times2+1 \\ 2\times1+4\times2+1 \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 2 & 4 \end{bmatrix}\begin{bmatrix} 1 \\ 2 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 11 \end{bmatrix}$$

Weighted sum of the first node

Weighted sum of the second node

Contd...

- i.e.,

$$v = \begin{bmatrix} w_1 & w_2 \\ w_3 & w_4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

- The weights of the first node of the hidden layer are in the first row.
- The weights of the second node of the hidden layer are in the second row.

Contd...

- This result can be generalized as the following equation.

$$v = Wx + b$$

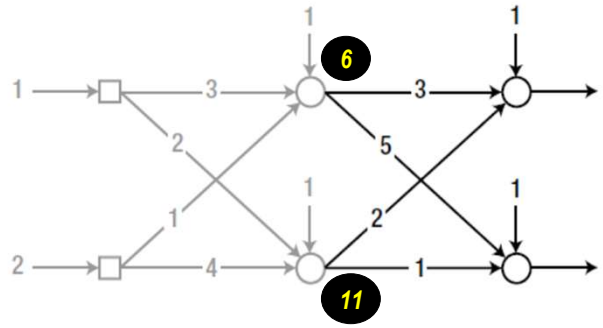
- x – is the input signal vector
- b – is the bias vector of the nodes
- W – is the matrix of weights

- For the example network, W is given as

$$W = \begin{bmatrix} -- \text{ weights of the first node } -- \\ -- \text{ weights of the second node } -- \end{bmatrix} = \begin{bmatrix} 3 & 1 \\ 2 & 4 \end{bmatrix}$$

Contd...

- Since we have the outputs from the hidden layer nodes, we can determine the outputs of the next layer, which is the output layer.



Contd...

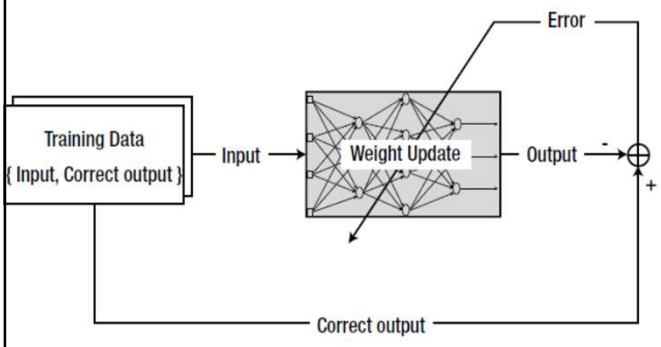
- We use the matrix form for computation of the output.

$$\text{Weighted sum: } v = \begin{bmatrix} 3 & 2 \\ 5 & 1 \end{bmatrix} \begin{bmatrix} 6 \\ 11 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 41 \\ 42 \end{bmatrix}$$

$$\text{Output: } y = \varphi(v) = v = \begin{bmatrix} 41 \\ 42 \end{bmatrix}$$

Contd...

2. The steps in the supervised learning of a NN



Contd...

1) Initialize the weights with adequate values.
2) Give the input into the neural network. Obtain the output from the neural network and calculate the error from the correct output.
3) Adjust the weights to reduce the error.
4) Repeat steps 2 – 3 for all the training data.

Backpropagation Algorithm

1. Initially the weights are assigned at random.
2. Then the **algorithm iterates through many cycles** of two processes (*a forward phase* and *a backward phase*) **until** a stopping criterion is reached. Every cycle is known as an **epoch**.

   Each epoch includes the two processes:

   a) Forward Phase

   b) Backward Phase

Contd...

a) Forward Phase

Here, the neurons are activated in sequence from the input layer to the output layer, applying each neuron's weights and activation function along the way. Upon reaching the final layer, output signals are generated.

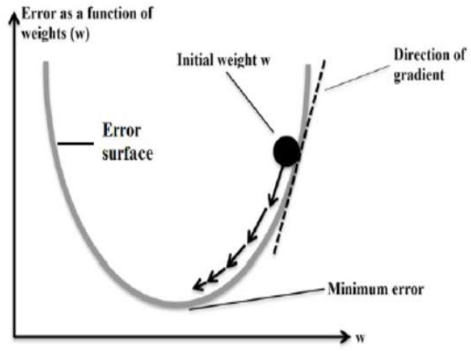
b) Backward Phase

Here, the network's output signals (from the forward phase) are compared against to the true target values in the training data. The difference between the network's output signals and the true values result in an **error** that is propagated backwards in the network **to modify the connection weights** between neurons and reduce the future errors.

Contd...

3. The technique used to determine how much a weight should be changed is known as **gradient descent method**. The gradient suggests, **how steeply** the **error will be reduced or increased** for ***a* change in the weight**. The **algorithm will attempt to change the weights** that results in the **greatest reduction in error** by an amount known as the **learning rate.**

Contd...



The Cost Function

- Loss / Cost functions depict by how much the algorithm (here, the backpropagation algorithm) has missed the target.
- There are commonly used cost functions in machine learning.
- Two such cost functions are
  - **SSE (Sum of Squared Errors)**
  - **MSE (Mean Squared Errors)**

Contd...

1. SSE (Sum of Squared Errors)

- SSE is a cost function used in machine learning and is defined as

$$\text{SSE} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- y_i is the actual value.
- $\hat{y}_i$ is the predicted value from the NN.

Contd...

2. MSE (Mean Squared Errors)

- MSE is a cost function used in machine learning and is defined as

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- y_i is the actual value.
- $\hat{y}_i$ is the predicted value from the NN.

Deep Learning

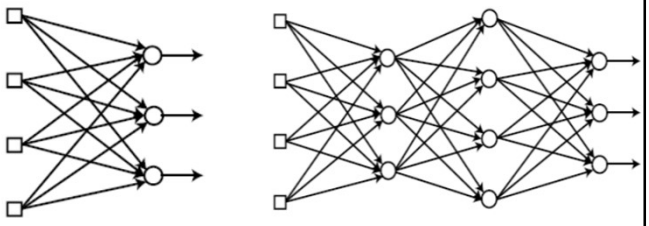
- An ANN with multiple hidden layers **is called** a **Deep Neural Network (DNN)**, and
- The practice of training such a network is referred to as **Deep Learning**.

Contd...

- The term ***deep*** refers to the number of layers in a neural network.

Contd...

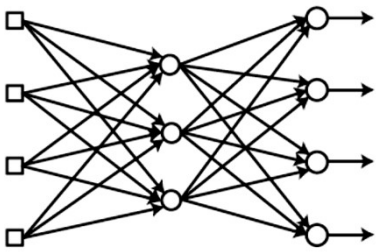
- When hidden layers are added to a **single-layer neural network**, it produces a **multi-layer neural network**.
  - Therefore, the multi-layer neural network consists of an input layer, hidden layer(s) and output layer.



Single-Layer Neural Network | Multi-Layer Neural Network

Contd...

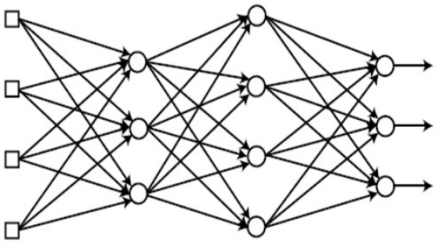
- The neural network that has a single hidden layer is called a **shallow neural network**.



Shallow Neural Network

Contd...

- As noted already, a multi-layered neural network that contains two or more hidden layers is called a **deep neural network**.



Deep Neural Network

Applications of Deep Learning

1. Automated Driving

Automotive researchers are using deep learning to automatically detect objects such as stop signs and traffic lights. In addition, deep learning is used to detect pedestrians, which helps decrease accidents.

Contd…

2. Aerospace and Defense

Deep learning is used to identify objects from satellites that locate areas of interest, and identify safe or unsafe zones for troops.

Contd…

3. Medical Research

Cancer researchers are using deep learning to automatically detect cancer cells. Teams at UCLA built an advanced microscope that yields a high-dimensional data set used to train a deep learning application to accurately identify cancer cells.

Contd…

4. Industrial Automation

Deep learning is helping to improve worker safety around heavy machinery by automatically detecting when people or objects are within an unsafe distance of machines.

Contd…

5. Electronics

Deep learning is being used in automated hearing and speech translation. For example, home assistance devices that respond to your voice and know your preferences are powered by deep learning applications.

References

1. Machine Learning with R, Second Edition, Brett Lantz, PACKT Publishing.
2. MATLAB Deep Learning, Phil Kim, Apress Publishing.