Artificial Neural Networks (ANN)

INTRODUCTION

- As we have noted, a glimpse into the natural world reveals that even a small child is able to do numerous tasks at once.

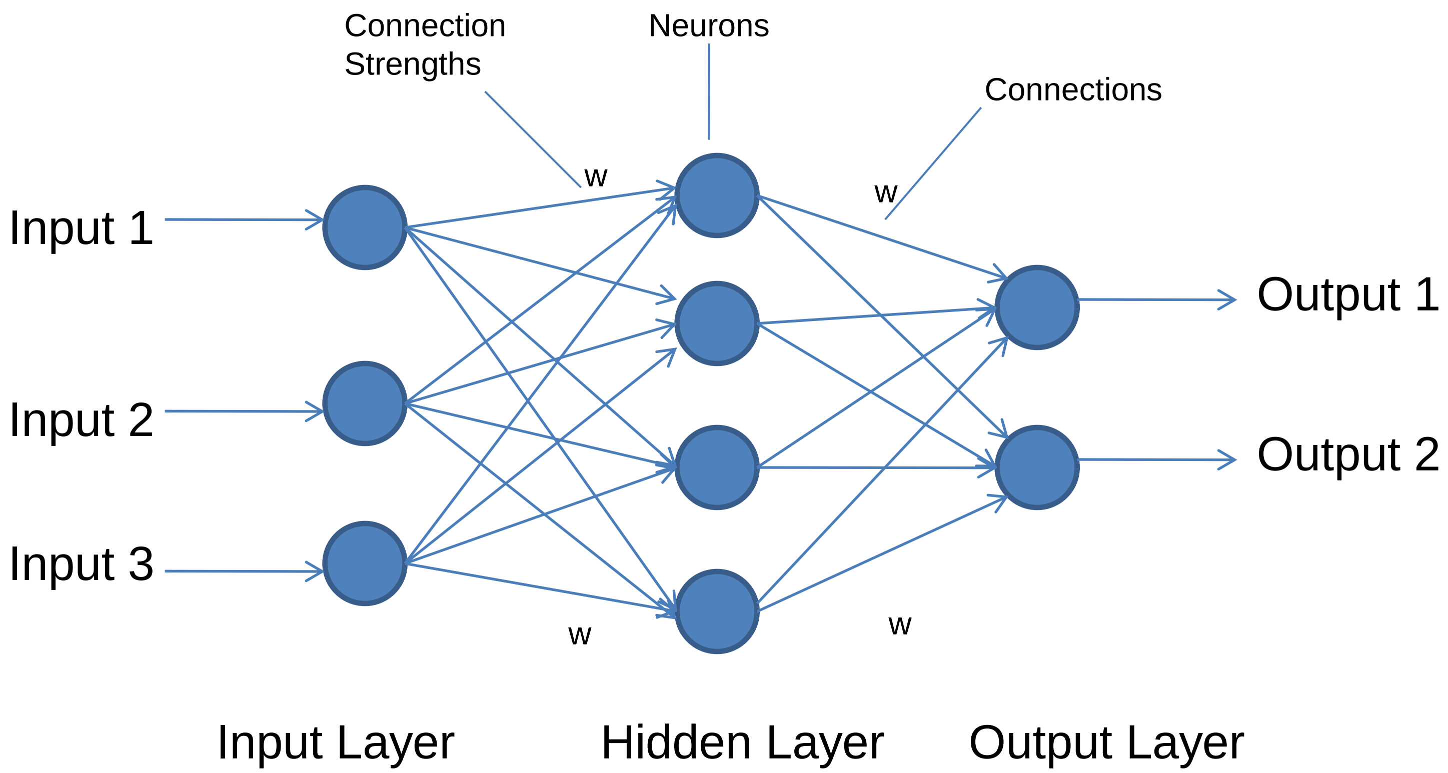
- The example of a child walking, probably the first time that child sees an obstacle, he/she may not know what to do. But afterward, whenever he/she meets obstacles, she simply takes another route.

- It is natural for people to both appreciate the observations and learn from them. An intensive study by people coming from multidisciplinary fields marked the evolution of what we call the artificial neural network (ANN).

ANNs- How They Operate

- ANNs represent a highly connected network of neurons - the basic processing unit
- They operate in a highly parallel manner.
- Each neuron does some amount of information processing.
- It derives inputs from some other neuron and in return gives its output to other neuron for further processing.
- This layer-by-layer processing of the information results in great computational capability.
- As a result of this parallel processing, ANNs are able to achieve great results when applied to real-life problems.

ANN Architecture



Historical Note

- Warren McCulloch and mathematical prodigy Walter Pitts gave **McCulloch-Pitts theory of formal neural networks**.
- In 1949, Donald Hebb further extended the work in this field, when he described how neural pathways are strengthened each time they are used
- In 1954, Marvin Minsky presented his thesis "**Theory of Neural-Analog Reinforcement Systems and Its Application to the Brain-Model Problem**" and also wrote a paper titled "**Steps Toward Artificial Intelligence**."
- Later John von Neumann invented the von Neumann machine.
- 1958, Frank Rosenblatt, a neurobiologist, proposed the perceptron, which is believed to be the **first physical ANN**.

Historical Note

- Between 1959 and 1960, Bernard Wildrow and Marcian Hoff developed the Adaptive Linear Elements (ADALINE) and the Multiple Adaptive Linear Elements (MADELINE) models.

- In 1986, Rumelhart, Hinton and Williams proposed the back propagation algorithm.

The Biological Neuron

- The entire human brain consists of small interconnected processing units called *neurons and are* connected to each other by nerve fibers

- The interneuron information communication makes it possible for multilevel hierarchical information processing, which gives the brain all its problem-solving power.

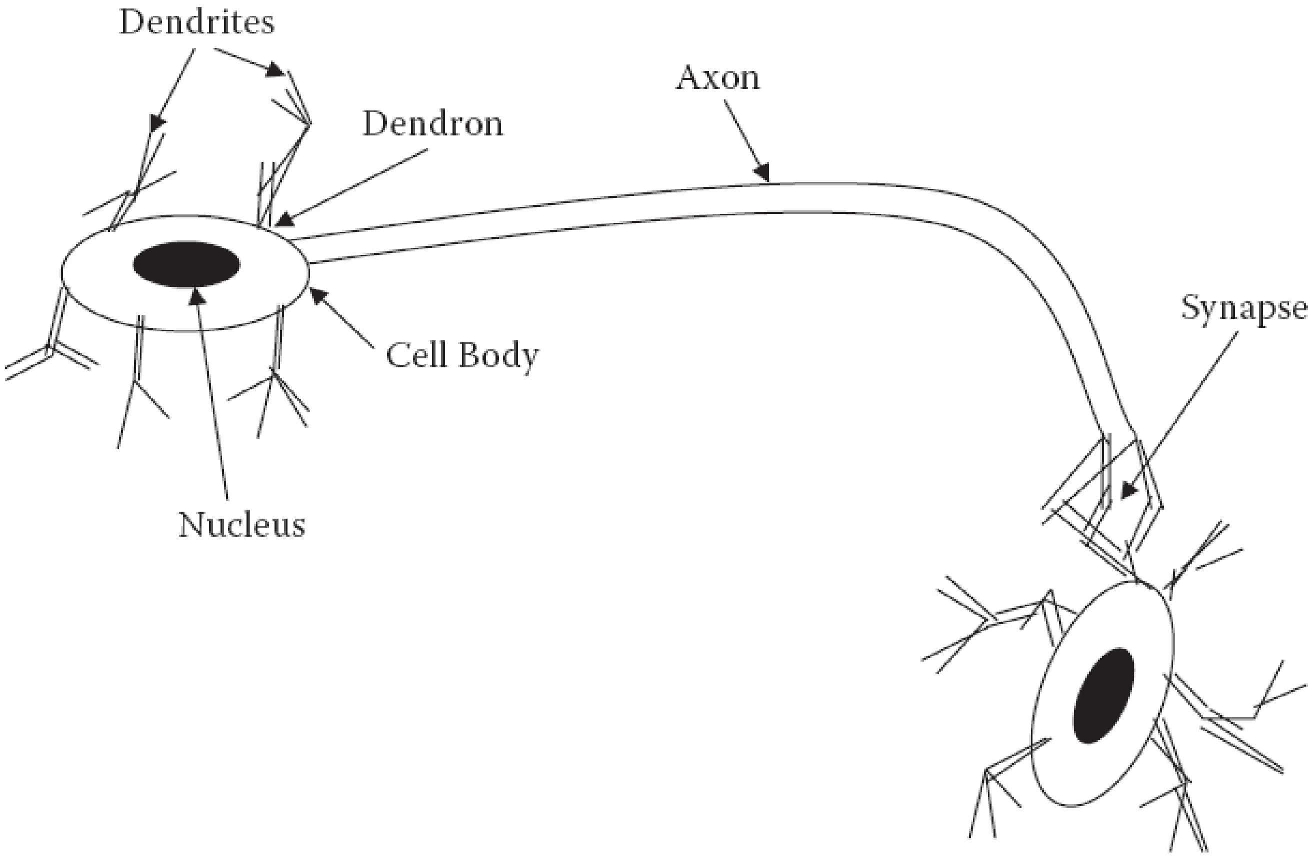
- Each neuron is provided with many inputs, each of which comes from other neurons. Each neuron takes a weighted average of the various inputs presented to it.

- The weighted average is then made to pass over a nonlinear inhibiting function that limits the neuron’s output. The nonlinearity in biological neurons is provided by the presence of potassium ions within each neuron cell and sodium ions outside the cell membrane.

- The difference in concentrations of these two elements causes an electrical potential difference, which in turn causes a current to flow from outside the cell to inside the cell. This is how the neuron takes its inputs

Structural Components of a Neuron

- A neuron has four main structural components - the *dendrites(input)*, the *cell body*, the *axon(output)*, and the *synapse.*



Structural Components of a Neuron

- Dendrites are responsible for receiving the signals from other neurons.

- These signals are passed through a small, thick fiber called a *dendron.*

- *The received signals collected at* different dendrites are processed within the cell body, and the resulting signal is transferred through a long fiber called the *axon.*

- *At the other end of the axon exists an inhibiting unit called a synapse,* which controls the flow of neuronal current from the originating neuron to the receiving dendrites of neighboring neurons.

The Artificial Neuron

- The neural network, by its simulating a biological neural network, is a novel computer architecture and a novel algorithmization architecture relative to conventional computers.

- It allows using very simple computational operations (additions, multiplication, and fundamental logic elements) to solve complex, mathematically ill-defined problems, nonlinear problems, or stochastic problems.

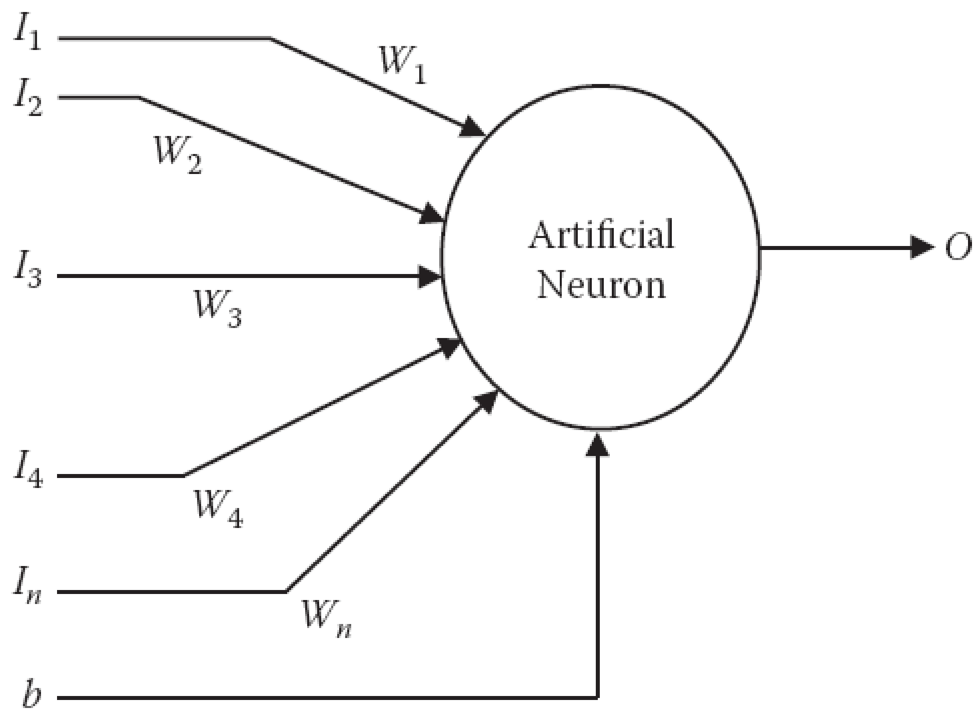
- The artificial neuron is the most basic computational unit of information processing in ANNs.

- Each neuron takes information from a set of neurons, processes it, and gives the result to another neuron for further processing.

- These neurons are a direct result of the study of the human brain and attempts to imitate the biological neuron.

Structure of an Artificial Neuron

- The neuron consists of a number of inputs. The information is given as inputs via input connections, each of which has some weight associated with it.
- An additional input, known as *bias, is given to* the artificial neuron.



Structure of an Artificial Neuron (continued)

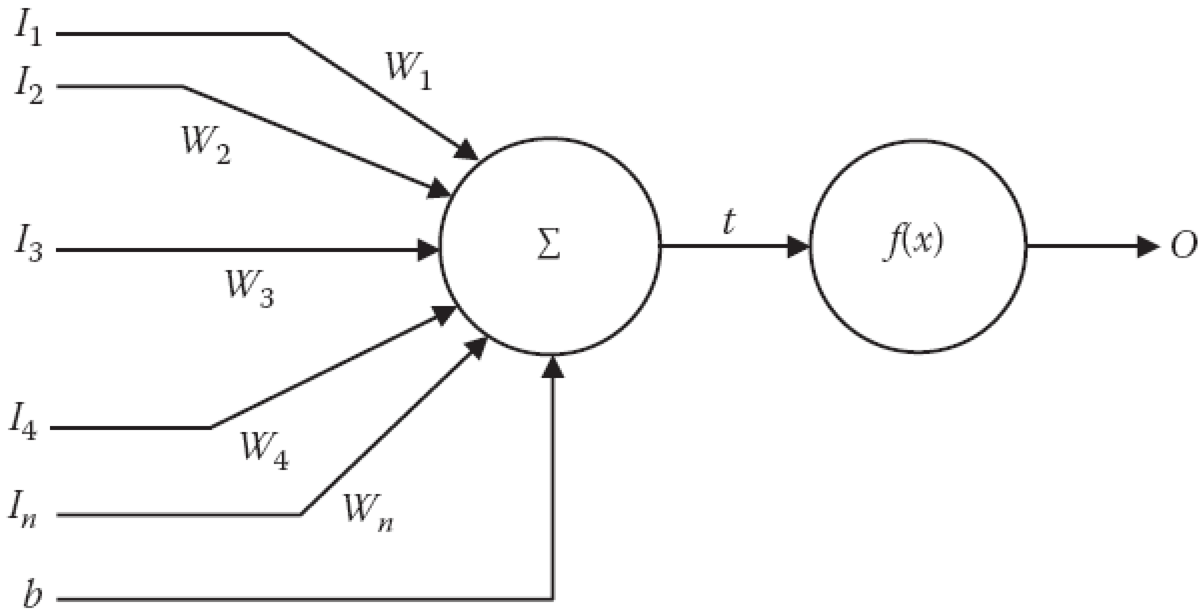


Figure 2.4 The processing in a single artificial neuron.

the inputs are marked *I1, I2, I3, … , In; the weights associated with each Connection* are given by *W1, W2, W3, … , Wn; b denotes the bias; and the output is denoted by O. Because there is* one weight for every input, the number of inputs is equal to the number of weights in a neuron.

The Processing of the Neuron

- Functionality of neuron that is performed can be broken down into two steps. The first is the weighted summation, and the second is the activation function. The two steps are applied one after the other, as shown in the previous slide.

- The function of the summation block is given by Equation

$$t = \sum_{i=1}^{n} W_i * I_i + b$$

- The summation forms the input to the next block. This is the block of the activation function, where the input is made to pass through a function called the *activation function.*

The Activation Function

- The activation function performs several important tasks
- One of the more important of which is to introduce nonlinearity to the network.
- Another important feature of the activation function is its ability to limit the neuron’s output.
- The complete mathematical function of the artificial neuron can be given as:

$$O = f(t)$$

where *f is any activation function*

$$O = f\left(\sum_{i=1}^{n} W_i * I_i + b\right)$$

The Perceptron

- The perceptron is the most basic model of the ANN. It consists of a single neuron. The perceptron may be seen as a binary classifier that maps every input to an output that is either 0 or 1.

- The perceptron is given by the function represented by Equation

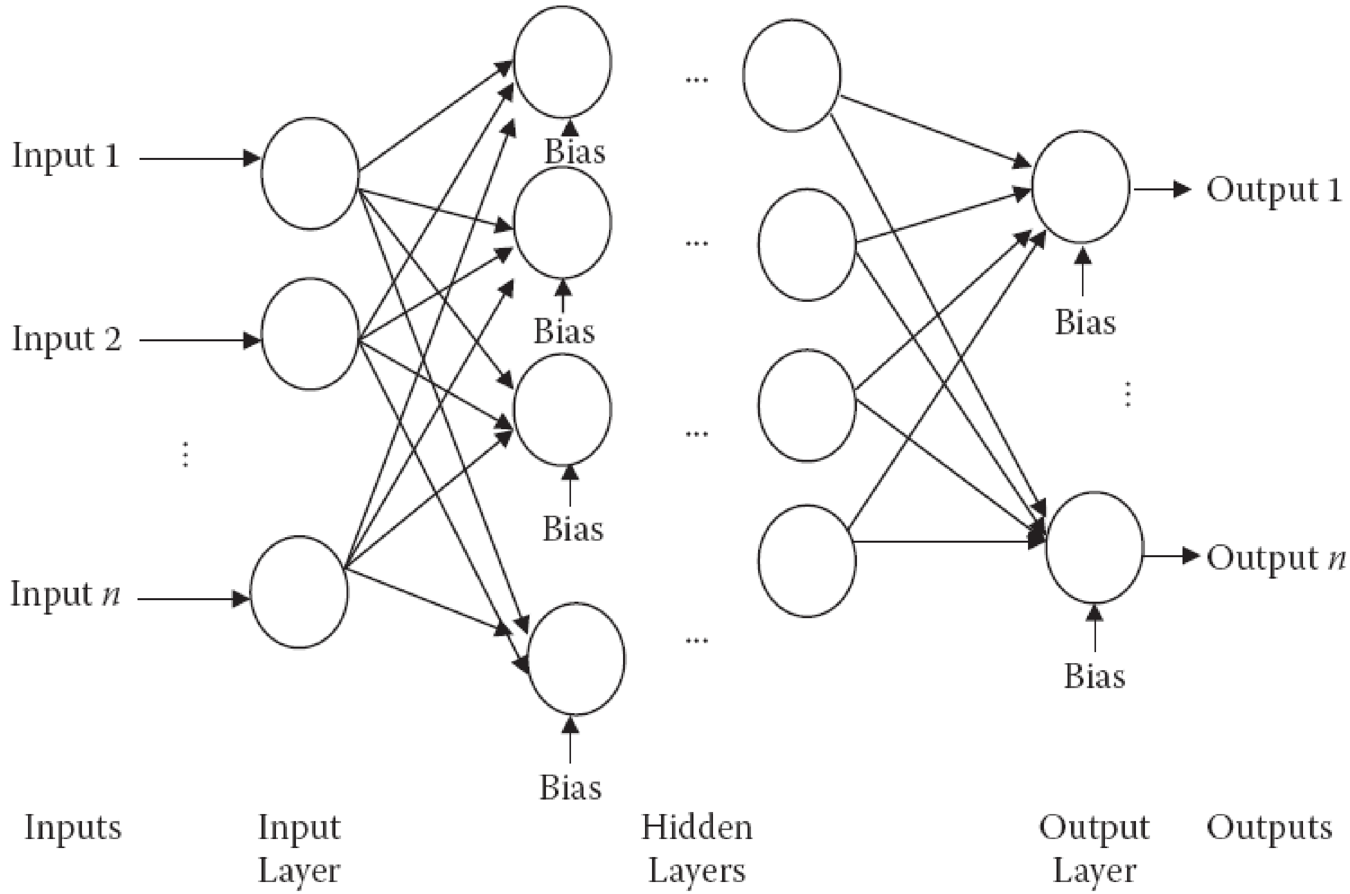
$$f(x) = \begin{cases} 1 & \text{if } w.x + b > 0 \\ 0 & \text{otherwise} \end{cases}$$

- where *w is the vector of real-world weights, x is the input, "." is the dot product of the two vectors,* and *b is the bias.*

- The perceptron has learning capabilities in that it can learn from the inputs to adapt itself. As a result, the perceptron is able to learn historical data.

Multilayer Perceptron

- Multilayer perceptrons are networks used to overcome the linear separability limitation of the perceptrons.



Multilayer Perceptron

- An MLP consists of
  1. An input layer,
  2. At least one intermediate or “hidden” layer,
  3. And one output layer

- The neurons from each layer being fully connected (in some particular applications, partially connected) to the neurons from the next layer.

- The number of neurons in the input layer is equal to the number of inputs of the system. Each neuron corresponds to one input.

- In output layer, the number of neurons is equal to the number of outputs that the system is supposed to provide.

- The number of layers decides the computational complexity of the ANN.

Common Activation Functions

No.	Function Name	Function Equation
1.	Identity	$f(x) = x$
2.	Logistic	$f(x) = \dfrac{1}{1 - e^{-x}}$
3.	Sigmoid	$f(x) = \dfrac{1}{1 - e^{x}}$
4.	Tanh	$f(x) = \tan h(x/2)$
5.	Signum	$f(x) = \begin{matrix} +1 & x > 0 \\ -1 & x < 0 \\ \text{undefined} & x = 0 \end{matrix}$
6.	Hyperbolic	$f(x) = \dfrac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$
7.	Exponential	$f(x) = e^{-x}$

Example

- Calculate the output for a neuron. The inputs are (0.10, 0.90, 0.05), and the corresponding weights are (2, 5, 2). Bias is given to be 1. The activation function is logistic. Also draw the neuron architecture.

Solution:

Using Equation, we have

$$O = f\left(\sum_{i=1}^{n} W_i * I_i + b\right)$$

where *I1 = 0.1, W1 = 2, I2 = 0.9, W2 = 5, I3 = 0.05, W3 = 2, and b = 1.*

$$f(x) = \frac{1}{1 - e^{-x}}$$

Hence the neuron's output is

*f(W1 * I1 + W2 * I2 + W3 * I3)*

*= f(2 * 0.1 + 5 * 0.9 + 2 * 0.05)*

= f(0.2 + 4.5 + 0.1)

= f(4.8)

= 1.008

Types of Data Involved

The data is generally divided into three sets

- **Training data :** These data are used by the training algorithm to set the ANN's parameters, weights, and biases. Training data make up the largest set of data, comprising almost 80 percent of the data.

- **Testing data:** This data set is used when the final ANN is ready. Testing data, which are completely unknown, are used to test the system's actual performance. The testing data set consists of about 10 percent of the data.

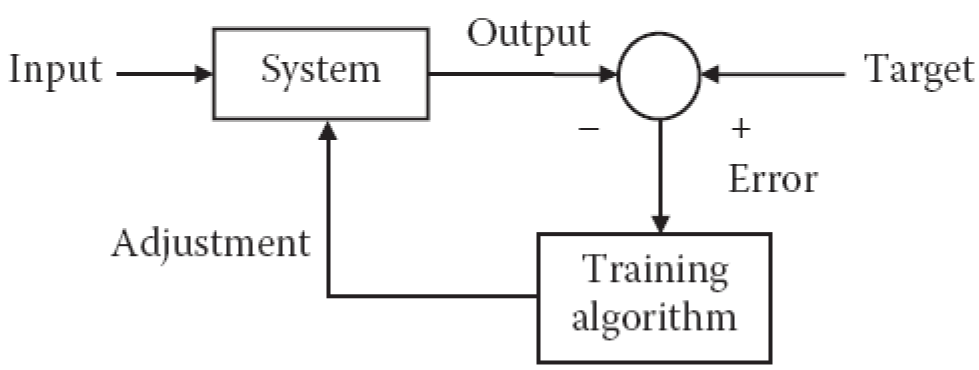
- **Validation data:** These data are used for validation purposes during the training period. At any point during training, we measure the error on both the training data and the validation data. The size of this data set is about 10 percent of the entire data set.

Training

- ANNs are able to learn the data presented in a process known as *training*
- The training algorithm tries to adjust the various weights (and biases) and set them to a value such that the ANN performs better at the applied input.
- Note that each ANN might not be able to train itself well for all architectures. This performance and training depends on the number of hidden layers as well as on the neurons in these hidden layers.
- Changing the architecture cannot be performed during training.
- Because this entire activity is manual, it may take a long time and a great deal of experience before a designer is able to selects a final design.

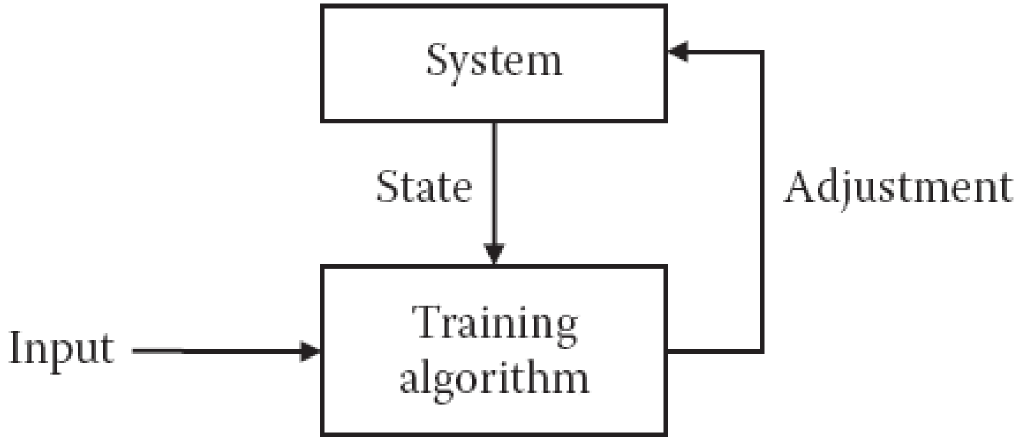
Types of Learning

- **Supervised learning :** In this kind of learning, both the inputs and the outputs are well determined and supplied to the training algorithm. Hence whenever an input is applied, we can calculate the error. We try to adjust the weights in such a way that this error is reduced.



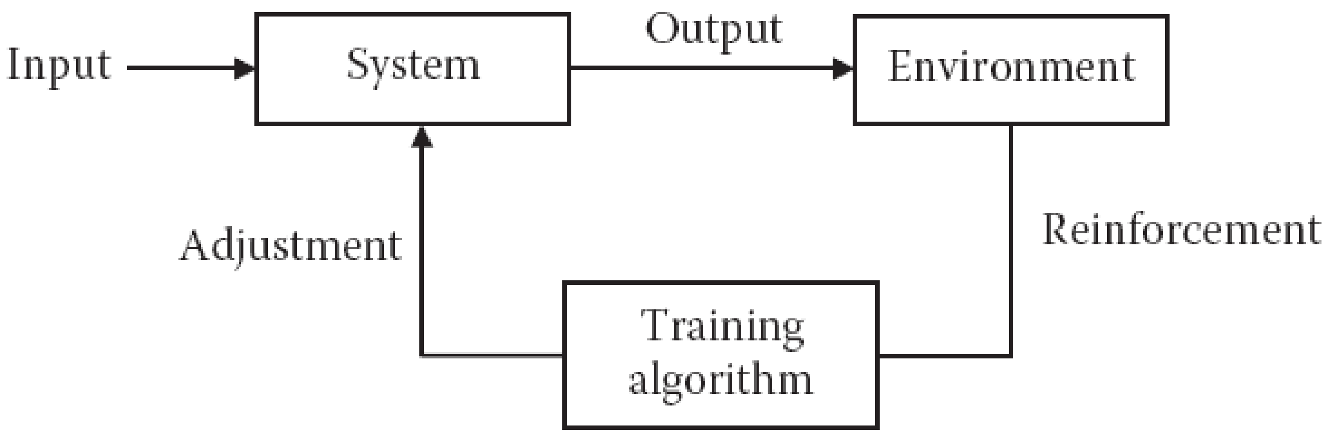
Types of Learning

- **Unsupervised learning:** In this type of learning, the target outputs are unknown. The inputs are applied, and the system is adjusted based on these inputs only. Either the supporting weights of the problem are added or the dissipative nodes are decreased. In either case, the system changes according to the inputs.



Types of Learning

- **Reinforcement learning:** This type of learning is based on the reinforcement process. In this system, the input is applied. Based on the output, the system either gives some reward to the network or punishes the network. In this learning technique, the system tries to maximize the rewards and minimize the punishment. The basic block diagram is given



Stopping Condition

- The algorithm stops according to the stopping condition. Normally one or more of the following criteria are used as stopping conditions:
  - **Time:** The algorithm may be stopped when the time taken to execute exceeds more than a threshold.
  - **Epoch**: The algorithm has a specified maximum number of epochs. Upon exceeding this number, the algorithm may be stopped
  - **Goal**: The algorithm may be stopped if the error measured by the system reduces to more than a specific value. It may not be useful to continue training after this point.
  - **Validating data:** If the error on validation data starts increasing, even if there is a decrease in the testing data, it would be better to stop further training.
  - **Gradient:** Gradient refers to the improvement of the performance or the lowering of the error in between epochs.