Models of Neuron

- A neuron is an information processing unit that is fundemental to the operations of a neural network.
- The block diagram of fig. (4) shows the model of a neuron, which forms the basis for designing a large family of neural networks.

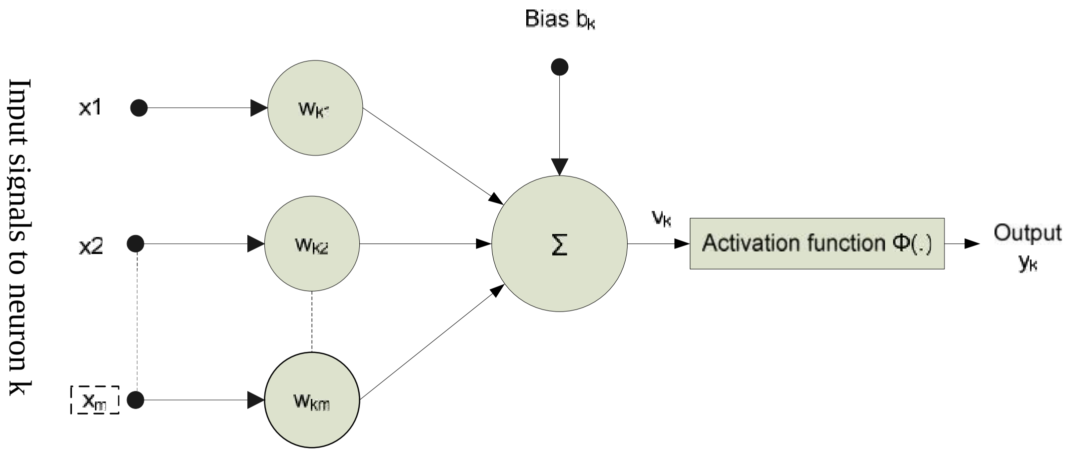


Fig. (4) Nonlinear model of a neuron labeled k

- Here we identify 3 basic elements of the neural model:
  1- A set of synapses or connecting links (j). each of which is characterized by a weight or strength of its own (w). a signal (x_j) at the input of synapse (j) connected to neuron (k) is multiplied by the synaptic weight (w_{kj}).
  2- An adder for summing the input signals weighted by the respective synaptic value of the neuron (the operation here constitute a linear combiner).
  3- An activation function for limiting the amplitude of the output of a neuron. This function is also referred to as squashing function in that it squashes (normalized) the permissible output. Typically, the normalized amplitude range of the output is written as the closed unit interval [0,1], or alternatively , [-1,1]. The neural model of fig. (4) also includes an externally applied ***bias***, denoted by b_k. The bias b_k has the

effect of increasing or lowering the net input of the activation function, depending on whether it is positive or negative, respectively.

In mathematical terms:

$$\boldsymbol{u_k = \sum_{j=1}^{m} w_{kj}\, x_j} \qquad \text{---------------- (1)}$$

$$\boldsymbol{y_k = \varphi(u_k + b_k)} \qquad \text{---------------- (2)}$$

where; $\boldsymbol{x_1, x_2, \ldots, x_m}$ are the input signals,

$\boldsymbol{w_{k1}, w_{k2}, \ldots, w_{km}}$ are the respective synaptic weights of neuron $\boldsymbol{k}$,

$\boldsymbol{u_k}$ (not shown in the figure) is the linear combiner output due to the input signal,

$\boldsymbol{b_k}$ is the bais,

$\boldsymbol{\varphi(\ )}$ is the activation function,

$\boldsymbol{y_k}$ is the output signal of the neuron k

note: some references referred to $\boldsymbol{u_k+b_k}$ as ***"net"*** and $\boldsymbol{\varphi(u_k + b_k)}$ as ***f*(net)**. And referred to the output by "***o***" instead of "***y***".

$$\boldsymbol{v_k = u_k + b_k = net_k} \qquad \text{---------------- (3)}$$

The combination of eq. (1) and eq. (3) can be formulated as folows:

$$\boldsymbol{v_k = \sum_{j=0}^{m} w_{kj}\, x_j} \qquad \text{---------------- (4)}$$

which represents the net of the summing junction of neuron k, i. e.,

$$\boldsymbol{v_k = net_k}$$

$$\boldsymbol{y_k = \varphi(v_k)} \qquad \text{---------------- (5)}$$

in equation (4) we have added a new synaptic. Its input is:

$\boldsymbol{x_0 = 1}$,

and its weight is: $\boldsymbol{w_{k0} = b_k}$

we may therefore formulate the model of nuron k as shown in fig. (5)

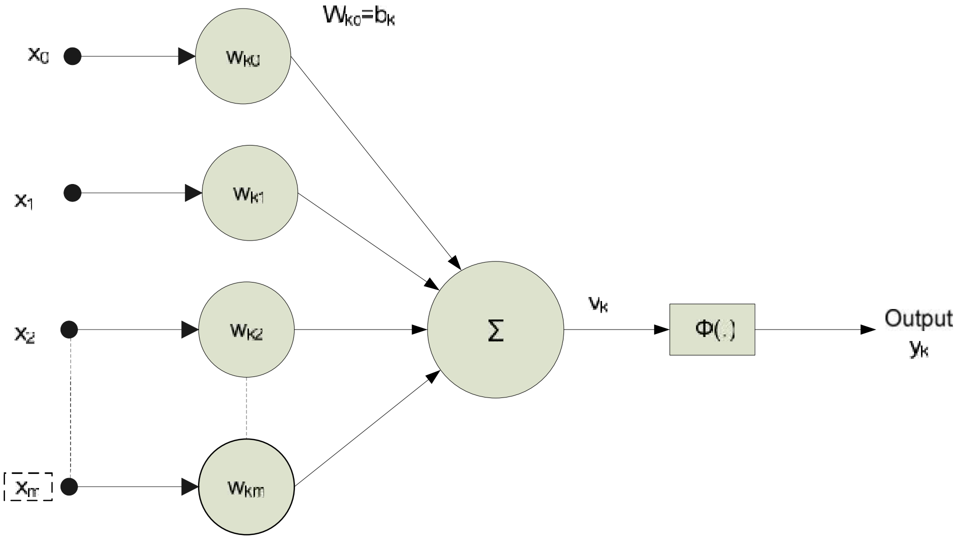
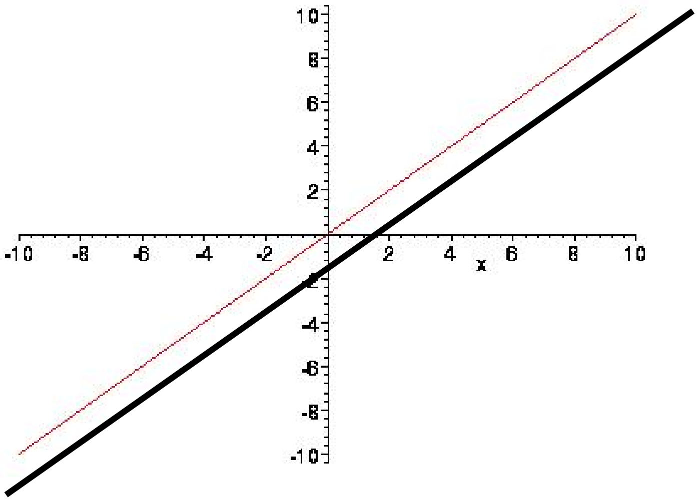


Fig. (5) Another nonlinear model of a neuron, $\boldsymbol{w_{k0}}$ accounts for the bias

Types of activation functions

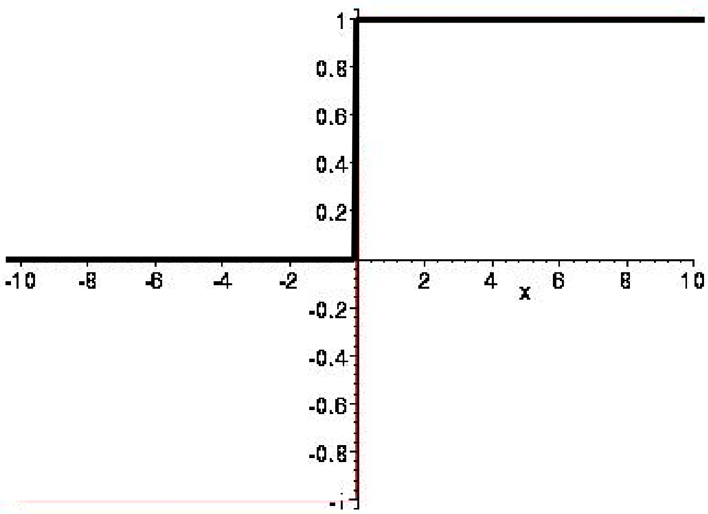
1- Identity function:

$$f(x) = x$$



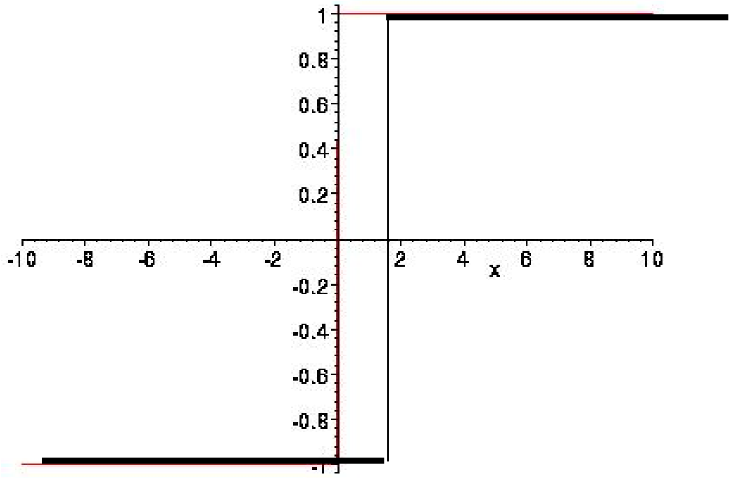
2- Step Function or Threshold function or Binary step function

$$f(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \le 0 \end{cases}$$



This function is also called the (Heaviside) function. Correspondingly, the output of neuron k will take the value of 1 if the induced local field is nonnegative, and 0 otherwise. In neural computation such neuron is referred to as the ***McCulloch-Pitts model***.

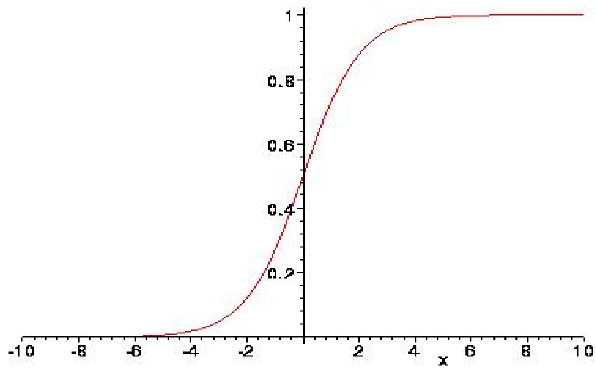
- Another common variation for the threshold function is to take the values -1 and +1 (bipolar function) as shown below.



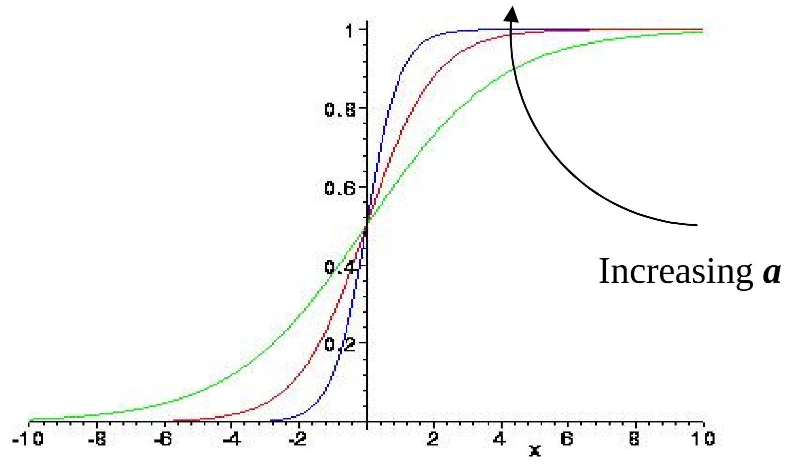
3- Logistic or Sigmoid function (S shape):

It is the most common form of activation function used in N.N. The **logistic function is also called binary sigmoid function**, and has the form

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



Where, ***a*** is the slope parameter of the sigmoid function. When ***a*** → ∞, the sigmoid function becomes a threshold function .



4- Symmetric Sigmoid or Bipolar Sigmoid

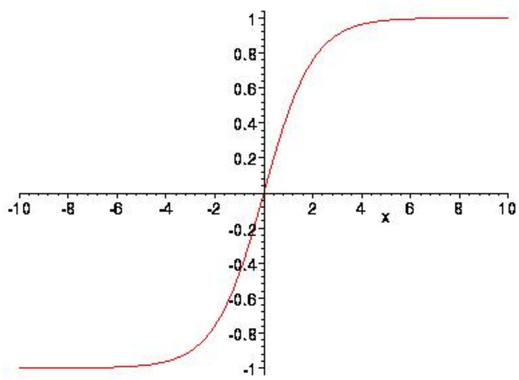
Some times it is desirable to have the activation function range from -1 to +1. In which case, the activation function will be

$$f(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}$$

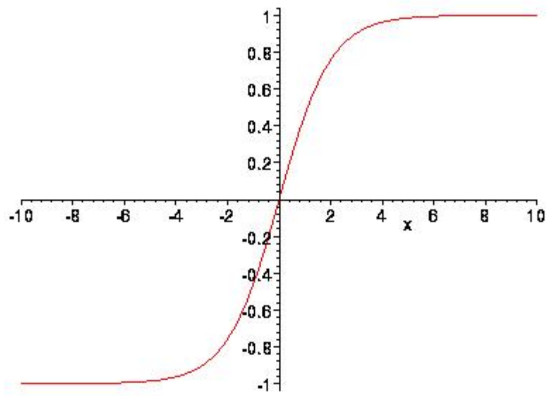
For the corresponding form of a sigmoid function we may use either the symmetric sigmoid function or the hyperbolic function. The symmetric sigmoid is a sigmoid function multiplied by 2 and then shifted down by 1 so that it ranges between -1 and 1.

If g(x) is the standard sigmoid then the symmetric sigmoid is

$$f(x) = 2g(x) - 1$$



The symmetric sigmoid differs from the hyperbolic tangent (***tanh(x)***) by a constant factor. As we can see, the graph below is identical to the graph for the symmetric sigmoid.

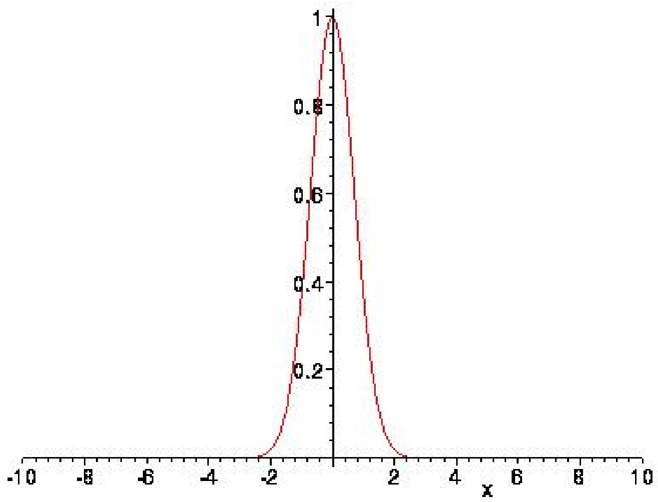


Radial Basis Functions

A radial basis function is simply a Gaussian,

$$f(x) = e^{(-a x^2)}$$

It is called local because, unlike the previous functions, it is essentially zero everywhere except in a small region.



Derivatives

The derivative of the **identity function** is just 1. That is, if f(x) is the identity then

$$\frac{df}{dx} = 1.$$

The derivative of the **step function** is not defined which is exactly why it isn't used.

The nice feature of **sigmoids** is that their derivatives are easy to compute. If f(x) is the logistic function above then

$$\frac{df}{dx} = f(x)(1 - f(x))$$

$$\frac{a\,e^{(-a\,x)}}{(1 + e^{(-a\,x)})^2}$$

for a = 1

$$\frac{e^{(-a\,x)}}{(1 + e^{(-a\,x)})^2}$$

This is also true of **hyperbolic tangent** . If f(x) is tanh then

$$\frac{df}{dx} = 1 - f(x)^2$$

$$= 1 - (\tanh(x))^2$$

Stochastic Model of a Neuron

The neural model describes in fig. (5) is deterministic in that its input-output behavior is precisely defined for all inputs. For some applications of N.Ns, it is desirable to base the analysis on stochastic neural model. The activation function of the McCulloch – Pitts model is given probabilistic interpretation. Specifically, a neuron is permitted to reside in only one two states +1 or -1, say the decision for a neuron to fire (i. e., switch its state from “off” to “ on”) is probabilistic.

- Let ***x*** denote the state of the neuron, and ***p(v)*** denote the probability of firing, where v is the induced local field of the neuron, we may then write

$$x = \begin{cases} +1 & \text{with probability } p(v) \\ -1 & \text{with probability } 1-p(v) \end{cases}$$

- A standard choice for ***p(v)*** is the sigmoid-shaped function

$$p(v) = \frac{1}{1 + \exp(-v/T)}$$

Where ***T*** is the pseudo temperature used to control the noise level and then the uncertainty in firing. When ***T*** → 0 then the above two equations reduced to a noiseless form.