MAE 552 Heuristic Optimization

Instructor: John Eddy

Lecture #30

4/15/02

Neural Networks

Neural Networks

Non-Linearity: (In response to a question asked)

1st – We'll define a non-linear system as one in which the outputs are not proportional to the inputs.

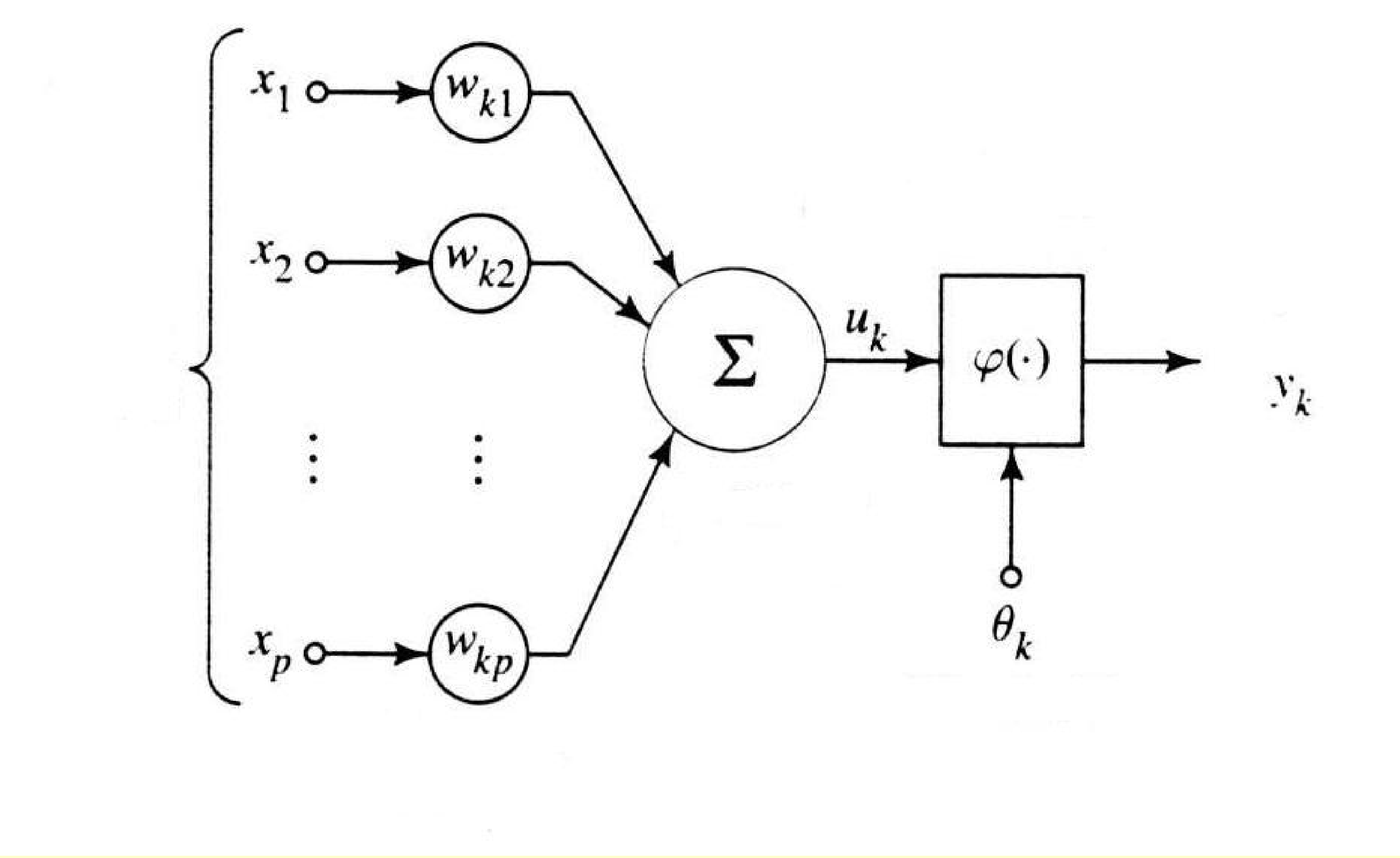
NN's incorporate non-linearity at each neuron by means of the neurons activation function which is typically a sigmoid.

(sigmoid meaning curved like the letter C or curved like the letter S).

More on this later.

Neural Networks

Modeling of a Neuron.



Elements of a Neuron Model

1 - *Synapses* or *Connecting links*: it is at these sites that input enters the neuron.

Each synapse has an associated weight. This weight will multiply the input prior to entering the summing junction (to be discussed next).

Elements of a Neuron Model

2 – An *adder* (linear combiner) for summing the input signals (which have been weighted by their respective synapses prior to entering). This is our *summing junction*.

So we have as output from our adder:

Elements of a Neuron Model

3 – An *activation function* (or squashing function) for limiting and otherwise controlling the output (y_k) of a neuron.

The result of this function is usually normalized between 0 and 1 or perhaps -1 to 1.

Neural Networks

So what is θ_k?

> **θ_k is referred to as the threshold value. It alters the value of the input to the activation function.**

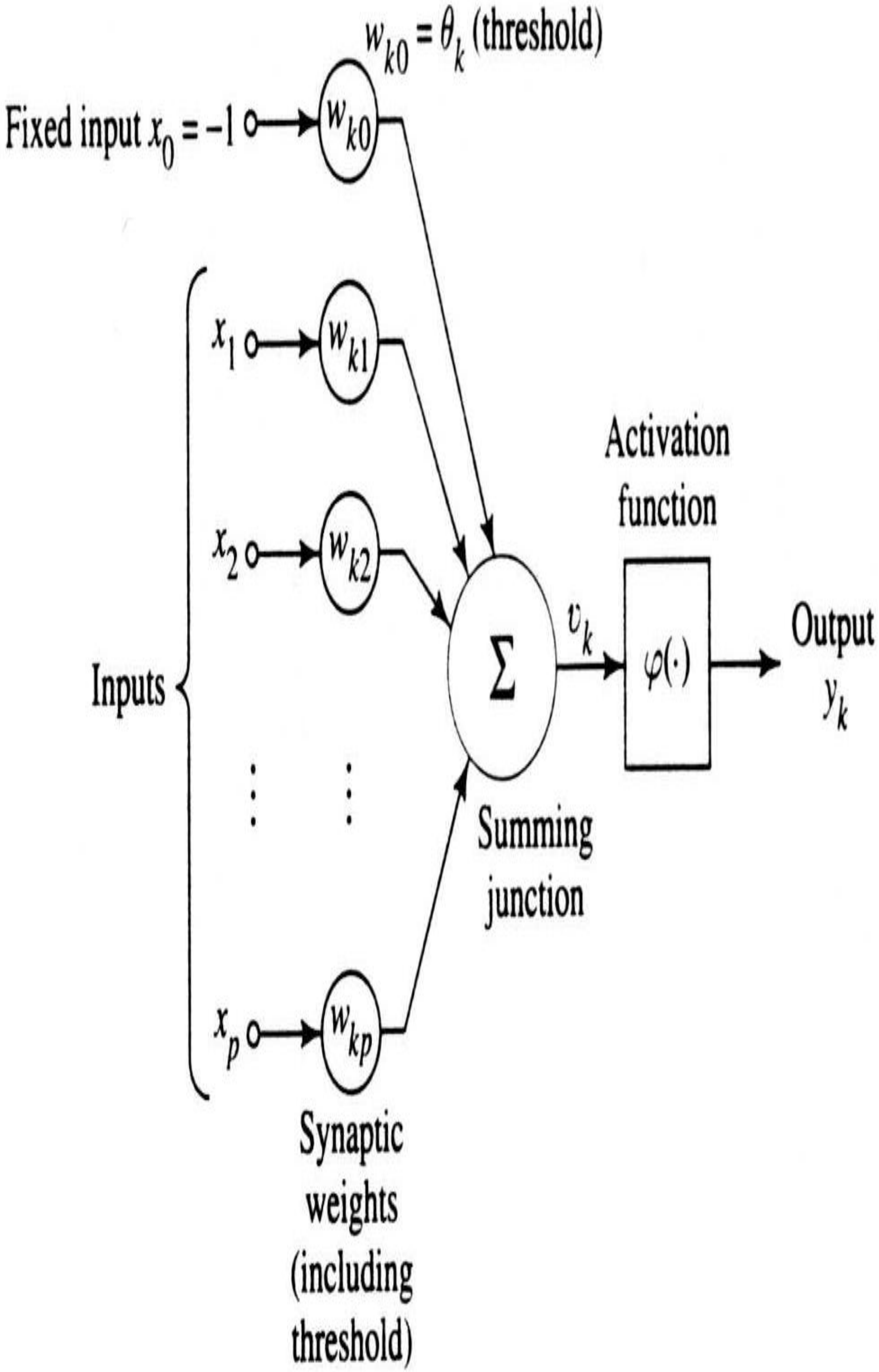
Neural Networks

On the other hand, a given neuron may implement a bias term formulation.

A bias term would typically be thought of as increasing the activation energy for a neuron.

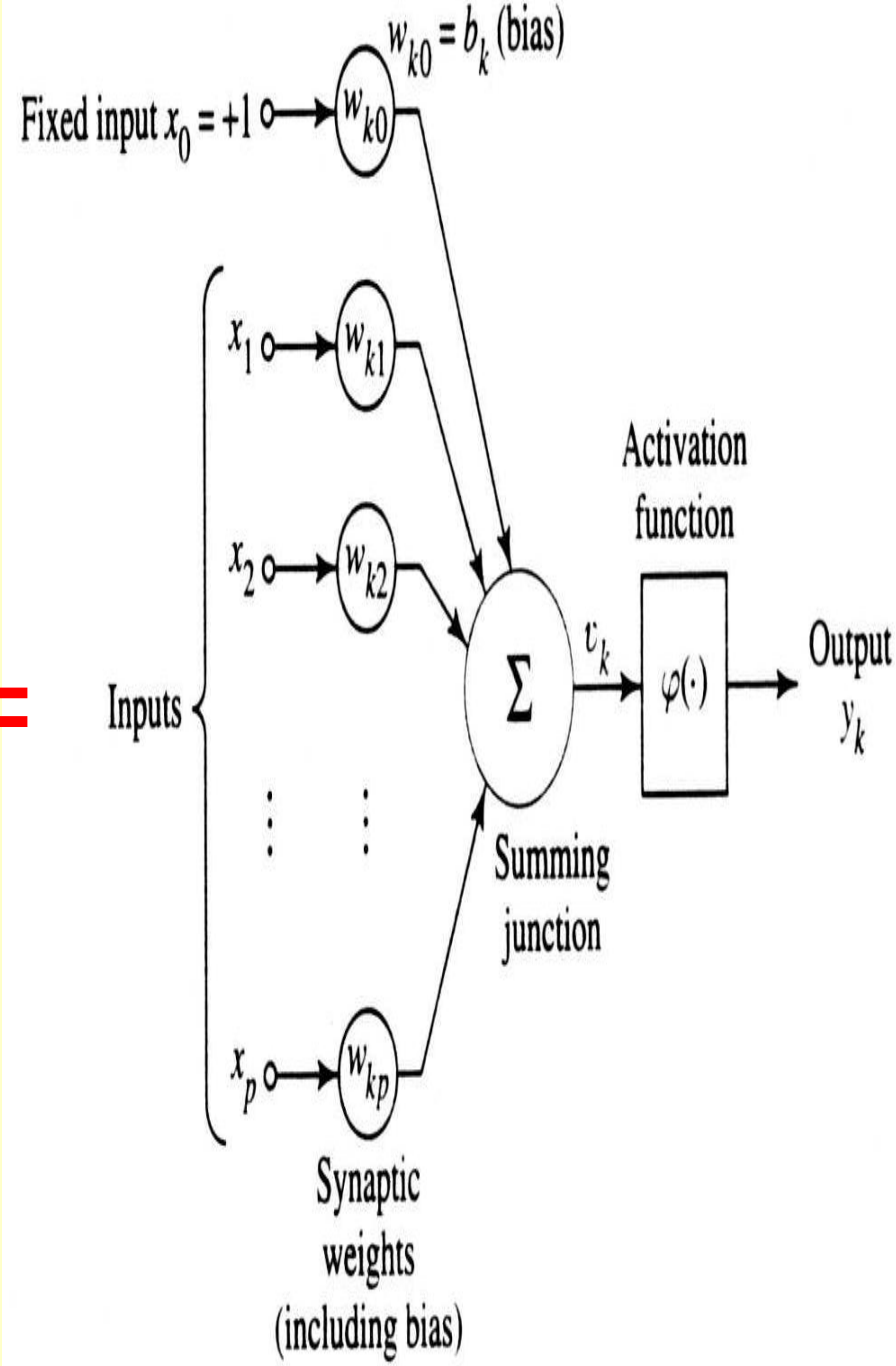
Mathematically, a neuron may be modeled equivalently using either a threshold or a bias if you keep in mind that either can take on positive and negative values.

Consider the following neuron models.



$w_{k0} = \theta_k$ (threshold)
Fixed input $x_0 = -1$
w_{k0}
x_1
w_{k1}
x_2
w_{k2}
Inputs
x_p
w_{kp}
Synaptic weights (including threshold)
Σ
Summing junction
v_k
Activation function
$\varphi(\cdot)$
Output y_k

=



$w_{k0} = b_k$ (bias)
Fixed input $x_0 = +1$
w_{k0}
x_1
w_{k1}
x_2
w_{k2}
Inputs
x_p
w_{kp}
Synaptic weights (including bias)
Σ
Summing junction
v_k
Activation function
$\varphi(\cdot)$
Output y_k

Elements of a Neuron Model

So for example: our previous output equation written for a threshold was:

Elements of a Neuron Model

So we can show that the two neuron models just presented are not only equivalent to each other, but they are also equivalent to the first neuron model where the threshold was input directly to the activation function.

Elements of a Neuron Model

So with the zero indexed term, we account for the threshold or bias and the activation function representation becomes:

Neural Networks

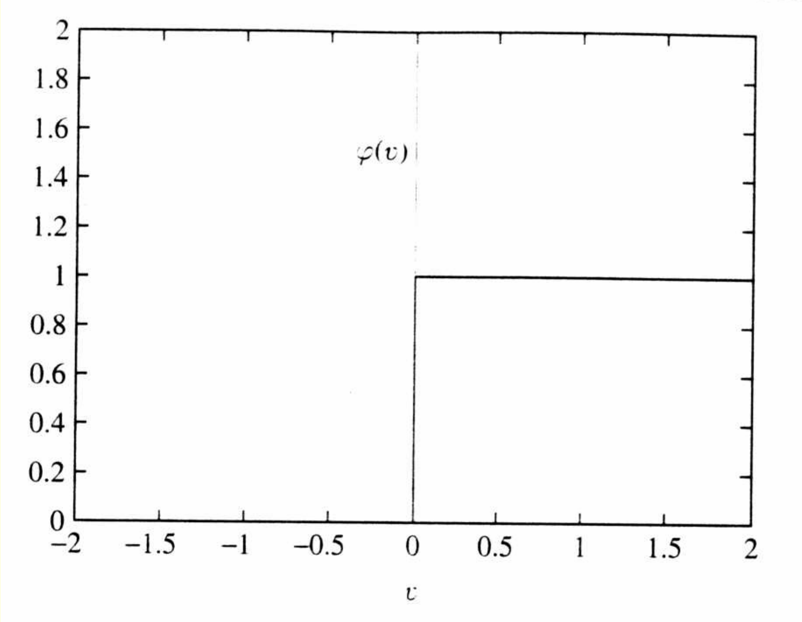
So what is does the activation function φ(*) look like?

There are three common implementation of the activation function. The first is referred to as a

Threshold Function:

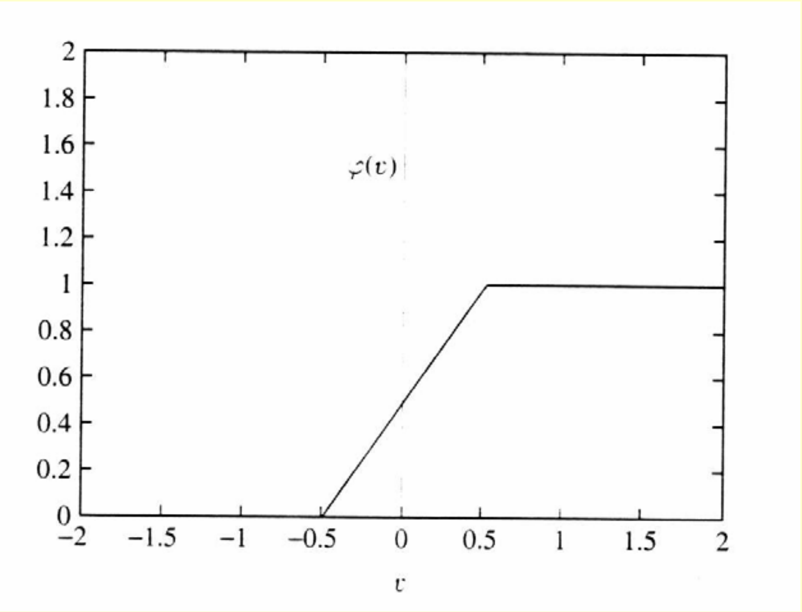
Neural Networks

And thus the output from one of our neurons will be:



Neural Networks

Piecewise-Linear

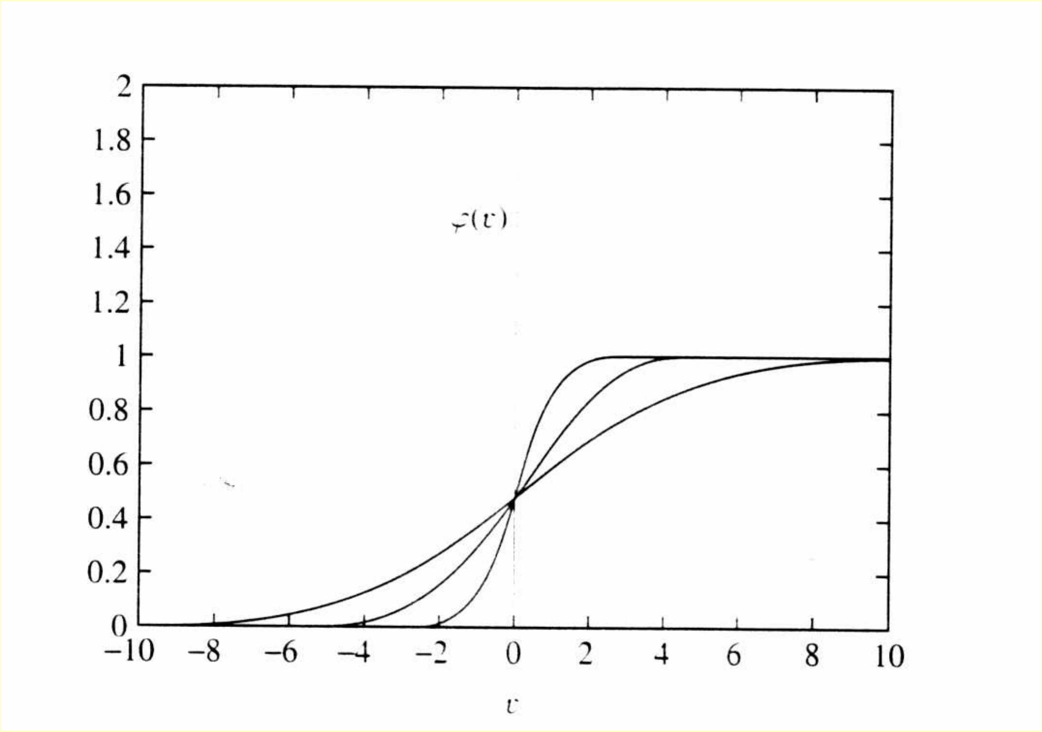


Neural Networks

Sigmoid Function: (by far the most common)

Common Example is the Logistic Function.

Neural Networks



Learning

Ok, so we know something about what a neuron looks like. How does a collection of such constructs "Learn"?

Def: Learning – in the context of a neural network…

Learning

There are 3 steps in the general case learning process:

Learning

Mathematically:

Learning

There are 3 learning paradigms that we will consider:

1 – Supervised Learning:

2 – Reinforcement Learning:

3 – Self-Organized Learning: