A Proposed Neural Network Based Approach for Analyzing Readings from a Sensor Network

by Sandeep Jain

www.sandeepjain.net

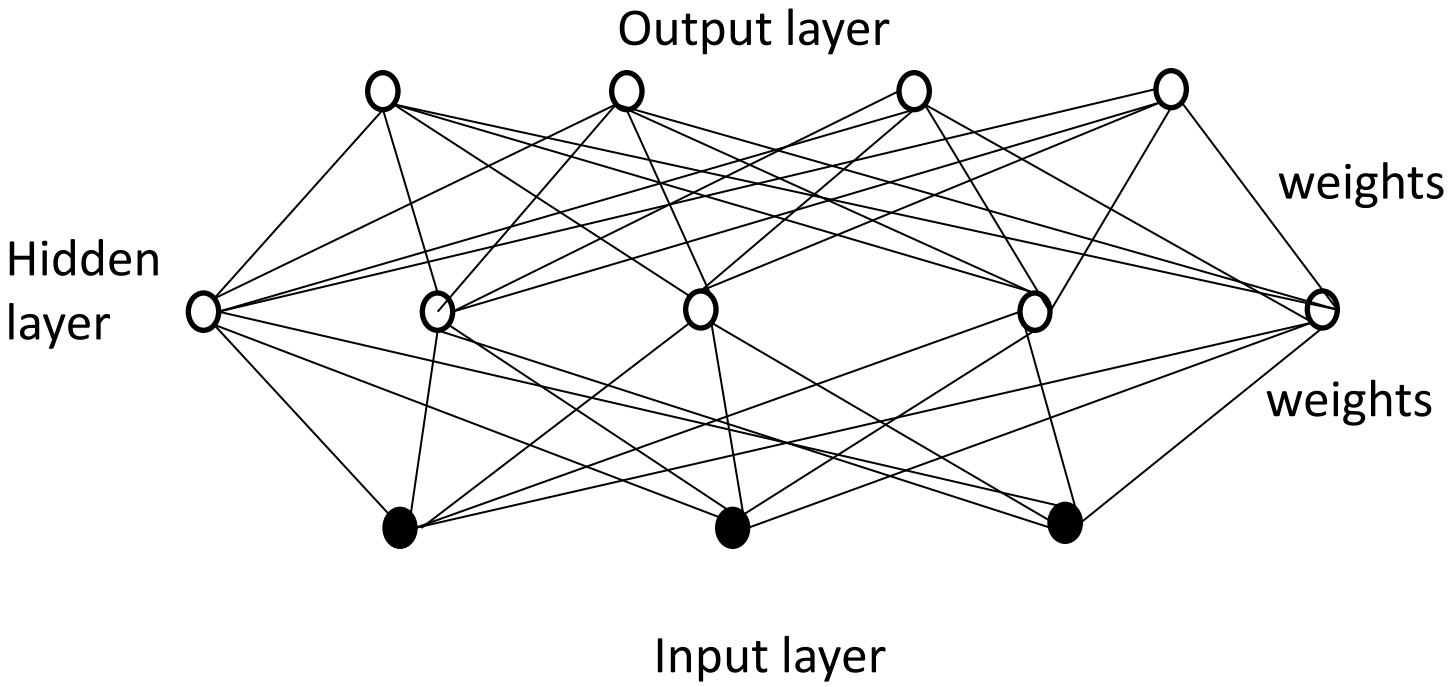
Agenda

- What is a Neural Network?
- Mathematics of a Neural Network in Brief
- The AQHI (Air Quality Health Index) Example
- Relevance to Sensor Networks

What is an Artificial Neural Network?

- Mathematical model of real network of biological neurons.
- There are “neurons” with inputs and outputs, “weights” as in biological synapses, and a “transfer function” which a neuron uses to map its input values to its output values.
- There are a great variety of artificial neural network architectures and algorithms – the most useful of the lot is called a “Multi-Layer Perceptron”.

The Multi-Layer Perceptron



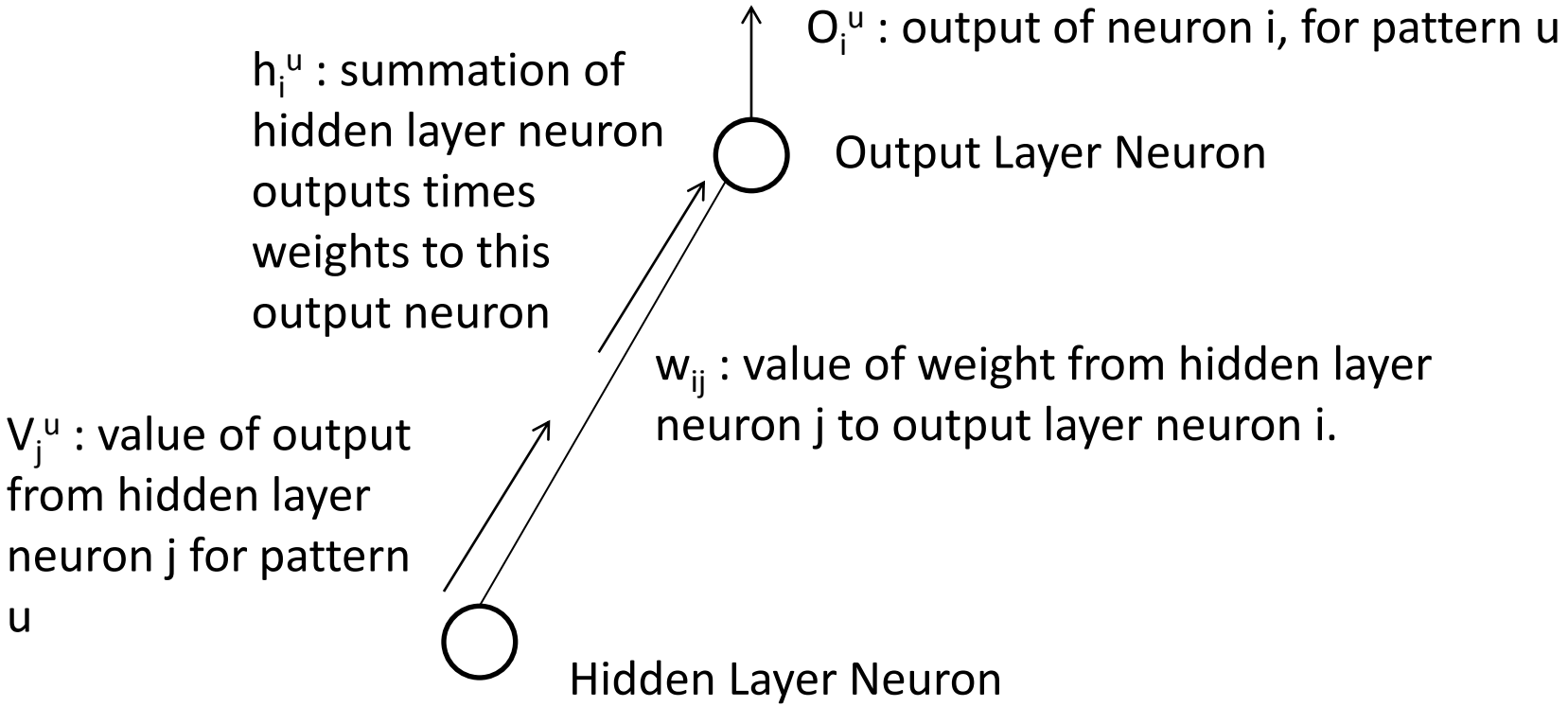
An Example of a Perceptron

The XOR Function:

Inputs		Output
0	0	0
0	1	1
1	0	1
1	1	0

This neural network will take 2 input neurons and 1 output neuron. By presenting the inputs and outputs repeatedly to the network, the network will “learn” the XOR function. The point of the whole exercise is that a neural network can learn far more complex functions than XOR with many more inputs and outputs, including functions for which no explicit mathematical formula can be defined.

Relevant Variables



Neural Network Mathematics (I)

Given a neuron i in the output layer, the input to that neuron will be:

$$h_i^u = \sum_j w_{ij} \; V_j^u$$

where Vj is the output of each hidden layer, and w_{ij} is the weight from each hidden neuron to the given output neuron. Given this input to the output neuron, the output of the neuron is:

$$O_i^u = g(h_i^u) = g(\sum_j w_{ij} \; V_j^u)$$

where the transfer function g is $g(x) = 1 / (1 + e^{-x})$

This is a sigmoidal “S” shaped function.

Neural Network Mathematics (II)

So given a set of inputs to the neural network, for each pattern or example u, we are able to calculate an output O_i^u where i is the subscript of each output of the neural network, and P_i^u is the corresponding known pattern value on which the network has to be trained. Given these two values, we can calculate an error function:

$$E(W) = 0.5 \sum_{iu} [\ P_i^u - O_i^u\ ]^2$$

This measures the distance between the actual output vector and the desired pattern vector. The E(W) indicates that this error function is a function of the weights of the neural network.

Neural Network Mathematics (III)

We now want to figure out how to continue updating the weights in a way that will keep on reducing the error function such that the discrepancy between the actual and desired output becomes so low that we can consider the neural network as "trained". Since E is a "surface" which is a mathematical function of the weights, we can do a gradient descent downhill of the surface until E is minimized. The formula for this is:

$$\Delta w_{ij} = - n \, \partial E / \partial w_{ij}$$

If you calculate the derivative of E with respect to w_{ij}, you will get, for a given neuron i at the output, and a given pattern u,

$$\Delta w_{ij} = - n \sum_{u} \delta_i^u V_j^u$$

where

$$\delta_i^u = O_i^u (1 - O_i^u) (P_i^u - O_i^u)$$

A similar calculation can be used to update the weights between the input layer and the hidden layer or layers.

Some Applications of Neural Networks

Numerous applications – a rich area of research:

- NETTalk – Training a neural network to convert text to speech
- Predicting the secondary structure of proteins
- Sonar target recognition
- Navigation of a car
- Image compression
- Backgammon
- Recognizing hand written zip codes
- Speech recognition

The AQHI Example

- AQHI: Air Quality Health Index, invented in Canada
- Raw pollutant values are collected from multiple receiving stations: NO_2, O_3, and $PM_{2.5}$ (particulate matter).
- They are translated via an empirically derived mathematical formula to an index between 1 and 10, which is a measure of the short term health impact of the pollutant recordings
- In my work, this data has been trained on a neural network, and the neural network is now predicting the output values for a given set of inputs.
- The value of this work is that if one wants to experiment with new mappings of inputs to outputs, for which a mathematical formula is not readily available, one can use neural network technology to generate implicit mappings of inputs to outputs

AQHI Neural Network

- 3 input neurons (NO_2, O_3, $PM_{2.5}$)
- 10 output neurons (AQHI Index Values from 1 to 10
- 20 hidden layer neurons
- 70 training examples (more are needed)
- At the end of 20,000 training iterations, training accuracy is 80% – 90%

Relevance to Sensor Networks

- A sensor network in all likelihood consists of multiple distributed sensors
- It is very likely that the sensors are taking numerical recordings of selected data values
- It is quite likely that the data reaches some kind of base station for the storage and analysis of the collected data
- Since the sensors can record continuously, there is very likely to be a large volume of data
- It is not at all obvious that the readings from the sensors can be analyzed by plugging them into an explicit mathematical formula
- Such a scenario is ideal for experimenting with neural networks
- A neural network can be very time-consuming to train if there large input and/or output vectors. But for data sets where the input and output vectors are not large, as with the AQHI example, a neural network can be an ideal technology for interpreting sensor data