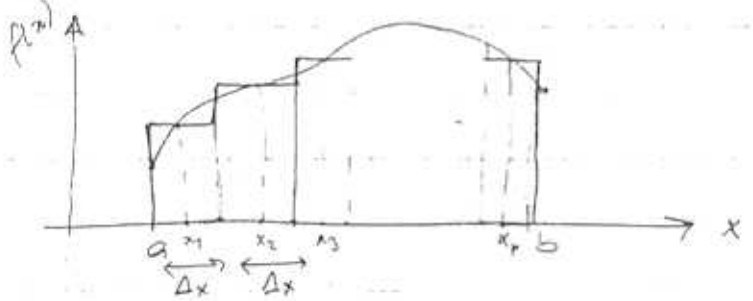
NEURAL NETWORKS AS UNIVERSAL APPROXIMATORS

Suppose that we want to approximate a function $h(x)$, defined on a compact interval. Hence, we try to find a new function $H(w,x)$ where weights w are adjustable, such that

$$\| h(x_i) - H(w^*, x_i) \| \text{ is minimum}$$

A simple choice, which is widely used in approximation theory, is staircase function (also used in numerical evaluation of integrals)



Obviously this staircase function is an approximation to original function $h(x)$, and the difference can be made arbitrarily small by making Δx arbitrarily small. In fact

$$\int_a^b (h(x) - H(a,x))^2 dx \to 0 \quad \text{as} \quad \Delta x \to 0 \quad \text{and}$$

$$\left(\sum_{i=1}^{p} |h(x_i) - H(w,x_i)|^2 \Delta x \right) = \Delta x \cdot E_{cum}$$

For any $\varepsilon > 0$, we can find $\Delta x > 0$ such that

$$\int_a^b |h(x) - H(w,x)|^2 dx < \varepsilon \qquad \text{(obvious)}.$$

Now, let us see if we can realize this staircase function by using neural networks. Input space is divided into P intervals, the length of each interval is $\Delta x = \frac{b-a}{P}$, and we have:

$$x_{i+1} - x_i = \Delta x \qquad i = 1, 2, \ldots P$$

And the intervals are $\left(x_i - \frac{\Delta x}{2}, \; x_i + \frac{\Delta x}{2}\right)$

(i.e. x_i is at the middle of i^{th} interval)

The values $h(x_i)$ should be known (by experiments). We define the unit-step function $1(x)$ as usual:

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

Note that in terms of sgn function:

$$1(x) = \frac{1}{2}\,(\operatorname{sgn}(x) + 1)$$

$1(x-T) \rightarrow$

$1(x-T) - 1(x-T-\Delta) \rightarrow$

Hence the staircase function can be written as follows:

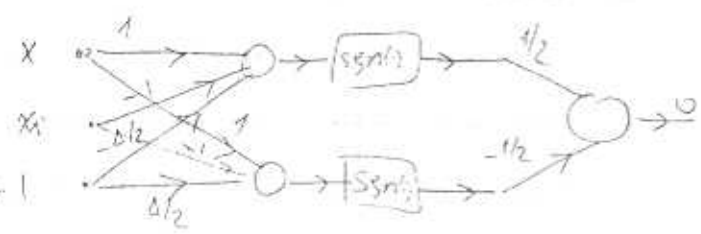
$$H(w,x) = h(x_1)\left[1\left(x - x_1 + \frac{\Delta}{2}\right) - 1\left(x - x_1 - \frac{\Delta}{2}\right)\right]$$

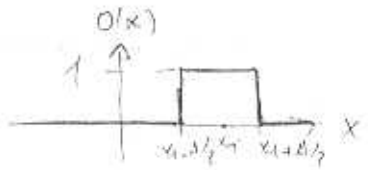
$$+ h(x_2)\left[1\left(x - x_2 + \frac{\Delta}{2}\right) - 1\left(x - x_2 - \frac{\Delta}{2}\right)\right]$$

$$+ \cdots + h(x_p)\left[1\left(x - x_p + \frac{\Delta}{2}\right) - 1\left(x - x_p - \frac{\Delta}{2}\right)\right]$$

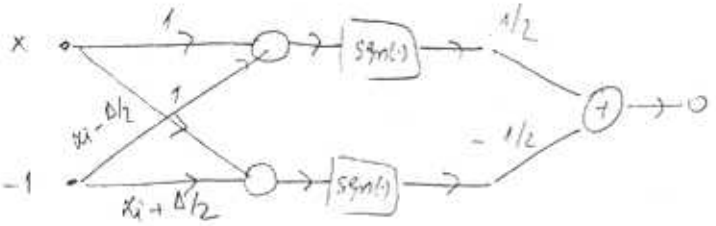
$$= \sum_{i=1}^{p} h(x_i)\left[1\left(x - x_i + \frac{\Delta}{2}\right) - 1\left(x - x_i - \frac{\Delta}{2}\right)\right]$$

$$1\left(x - x_i + \frac{\Delta}{2}\right) - 1\left(x - x_i - \frac{\Delta}{2}\right) = \frac{1}{2}\operatorname{sgn}\left(x - x_i + \frac{\Delta}{2}\right) - \frac{1}{2}\operatorname{sgn}\left(x - x_i - \frac{\Delta}{2}\right)$$

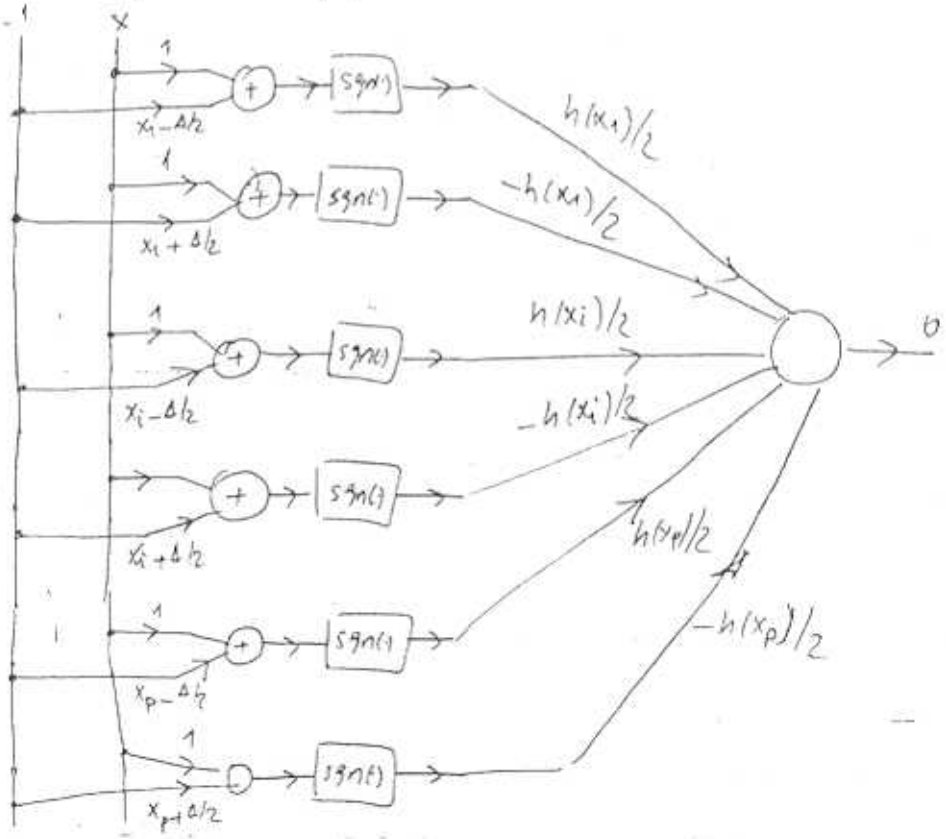




elementary window



Then we get the approximator:



Here we have one hidden layer and at the output we only use a linear sum ($h(x)$ need not be restricted to unity). In the hidden layer we need $2P$ neurons. Obviously, with this approach if see want to get finer approximation, Δ should be small, hence P, hence number of hidden neurons should be large.

Same analogy works even when we have multi inputs and/or multi outputs.

Here we have signum function at neuron outputs. If we have continuous activation function $\varphi(v) = \frac{1-e^{\lambda v}}{1+e^{\lambda v}}$, the situation is similar:

as $\lambda \to +\infty$, bump becomes closer to pulse

Hence even if we use continuous neurons, same analysis holds (i.e. function is approximated by staircase, staircase is approximated by bumps ...).

This analogy tells that, any continuous function on a compact interval can be approximated by a neural network with a desired degree of accuracy. This property can be proven rigorously as given by the following universal approximation theorem:

THEOREM (Cybenko, 1989): Let $\varphi(\cdot)$ be a nonconstent, bounded and monotone increasing function. Let I_n denote the unit hypercube $[0,1]^n$, (i.e. $x \in I_n \Rightarrow x = (x_1, \dots, x_n)$, $0 \le x_i \le 1$). Let $f: I_n \to \mathbb{R}$ be an arbitrary continuous function (Domain is compact). For any $\varepsilon > 0$ there exist an integer M, and a set of real constants α_i, θ_i and w_{ji}, $i = 1, 2, \dots, M$, $j = 1, 2, \dots, n$, such that

$$F(x_1, \dots, x_n) = \sum_{j=1}^{M} \alpha_j \varphi\left(\sum_{i=1}^{n} w_{ji} x_i - \theta_j \right) \Rightarrow \text{Picture of this NN.}$$

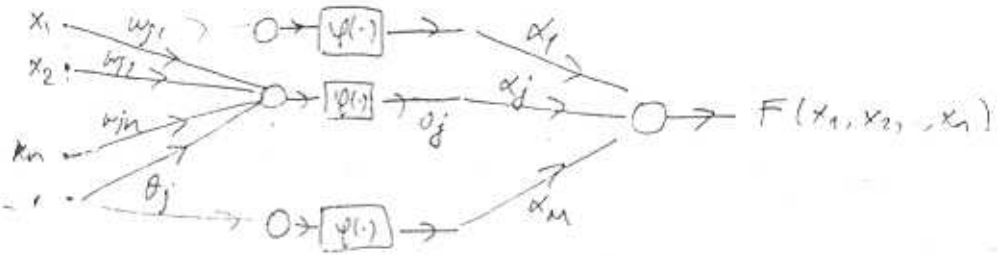
is an uniform approximation of f on I_n. i.e.

$$|F(x_1, \dots, x_n) - f(x_1, \dots, x_n)| < \varepsilon \qquad \forall (x_1, \dots, x_n) \in I_n.$$

1°) Network has n input nodes $x_1, \dots, x_n$, and one hidden layer of M neurons.

2°) Synaptic weights of j^{th} hidden neuron are $w_{j1}, \dots, w_{jn}$ and θ_j as threshold.

3°) Output is linear combination of hidden neuron outputs.



This theorem is in fact an EXISTENCE type of theorem. It says that theoretically, NN's can approximate any continuous function with any desired degree of accuracy. In a given compact set. Moreover, the theorem states that for such an approximation, a single layer is sufficient. However, the theorem does not say that a single layer structure is optimum in the sense of learning time, ease of implementation or total number of connections and/or neurons.