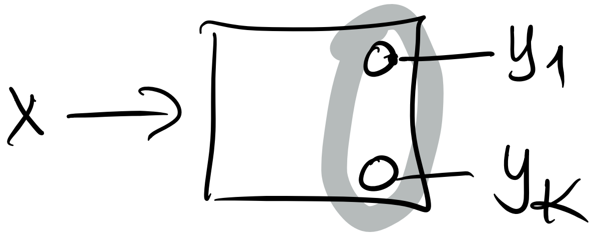
Artificial Neural Networks

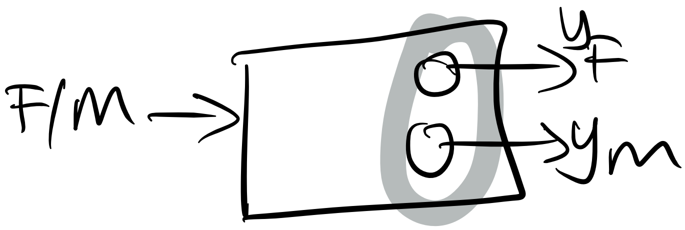
Function Approximation & Network Capabilities

Berrin Yanikoglu

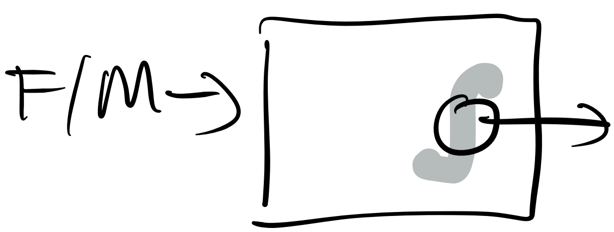
2/2019



$\underline{y}$, t



$\equiv$ above w/ $K=2$



$0.7 = y$

M = 1
F = 0

If x is male, $t=1$

$$P(M|X) = 0.7 = y$$
$$P(F|X) = 0.3 = 1-y$$
$$+$$
$$1$$

$$-\left(t \cdot \log y + (1-t) \cdot \log(1-y)\right)$$

output for 1 class

output for 0 class

if X is female, $t=0$ yellow term remain

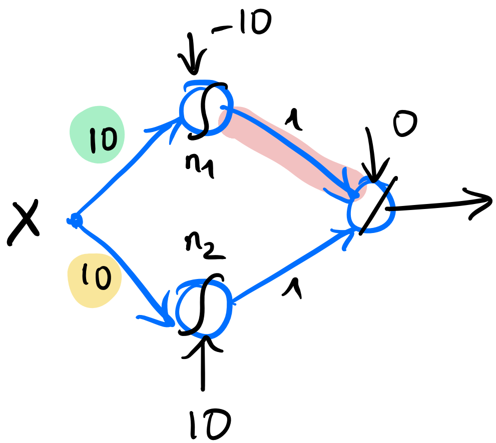
Function Approximation

Neural Networks are intrinsically function approximators:

- we can train a NN to map real valued vectors to real-valued vectors.

Function approximation capabilities of a simple network, in response to its parameters (weights and biases) are illustrated in the next slides.

Function Approximation: Example



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

$$f^2(n) = n$$

Superscripts are layer numbers

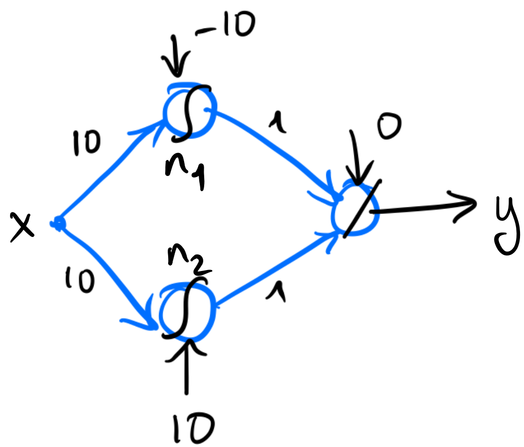
Nominal Parameter Values

$$w^1_{1,1} = 10 \qquad b^1_1 = -10 \qquad w^2_{1,1} = 1$$

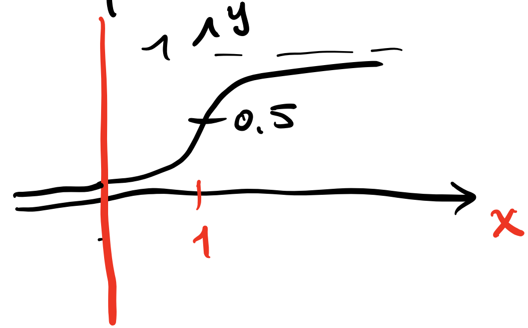
$$w^1_{2,1} = 10 \qquad b^1_2 = 10 \qquad w^2_{1,2} = 1$$

$$b^2 = 0$$

Function Approximation: Example



For n_1: net input = 1 output =



For n_2: when $x = -1$, net = 0, y = 0.5

Nominal Parameter Values

$$w^1_{1,1} = 10 \qquad b^1_1 = -10 \qquad w^2_{1,1} = 1$$

$$w^1_{2,1} = 10 \qquad b^1_2 = 10 \qquad w^2_{1,2} = 1$$

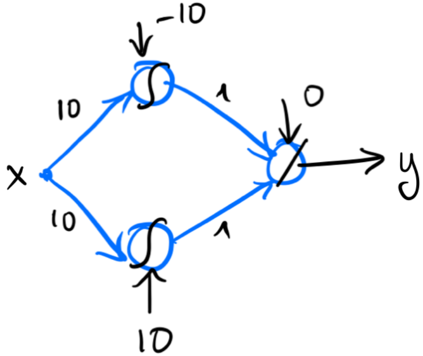
$$b^2 = 0$$

-1 0

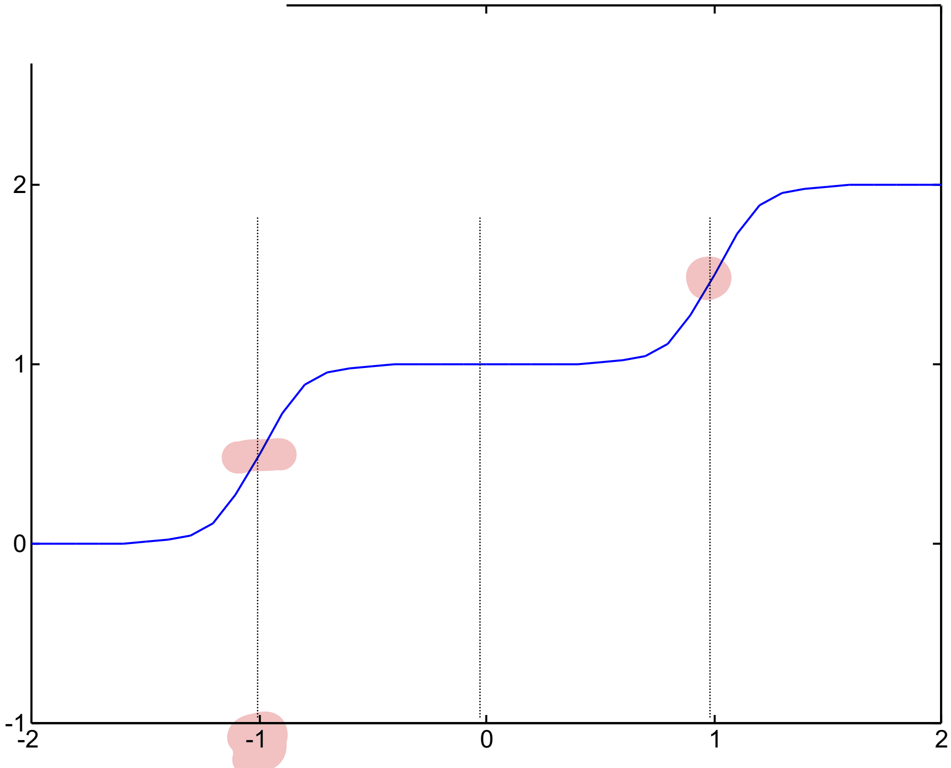
$$f^1(n) = \frac{1}{1+e^{-n}}$$

$$f^2(n) = n$$

Superscripts are layer numbers



minal Response



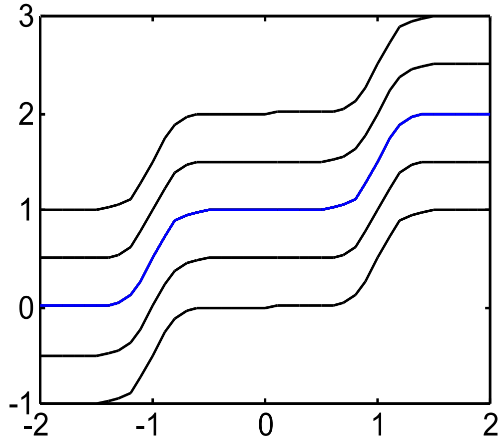
only n_2 is active

both n_1 & n_2 are "active"

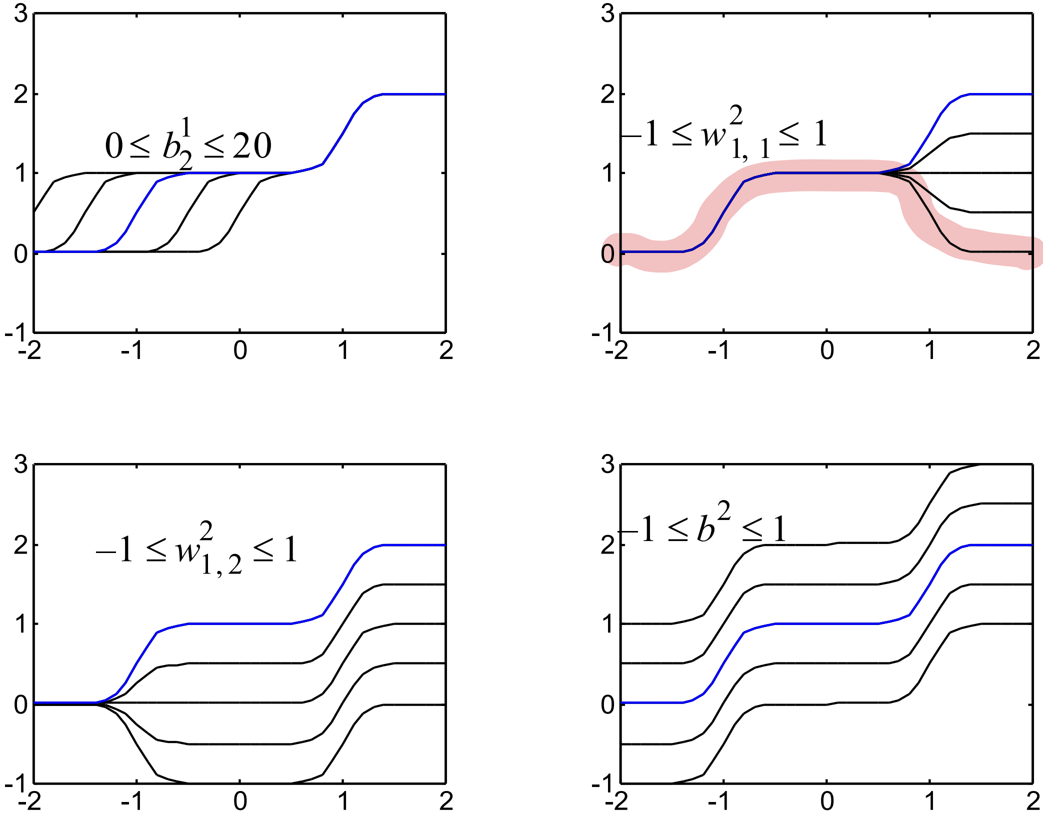
Parameter Variations

What would be the effect of varying the bias of the output neuron?

$$-1 \le b^2 \le 1$$

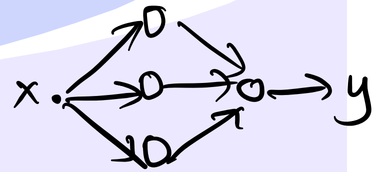


Parameter Variations



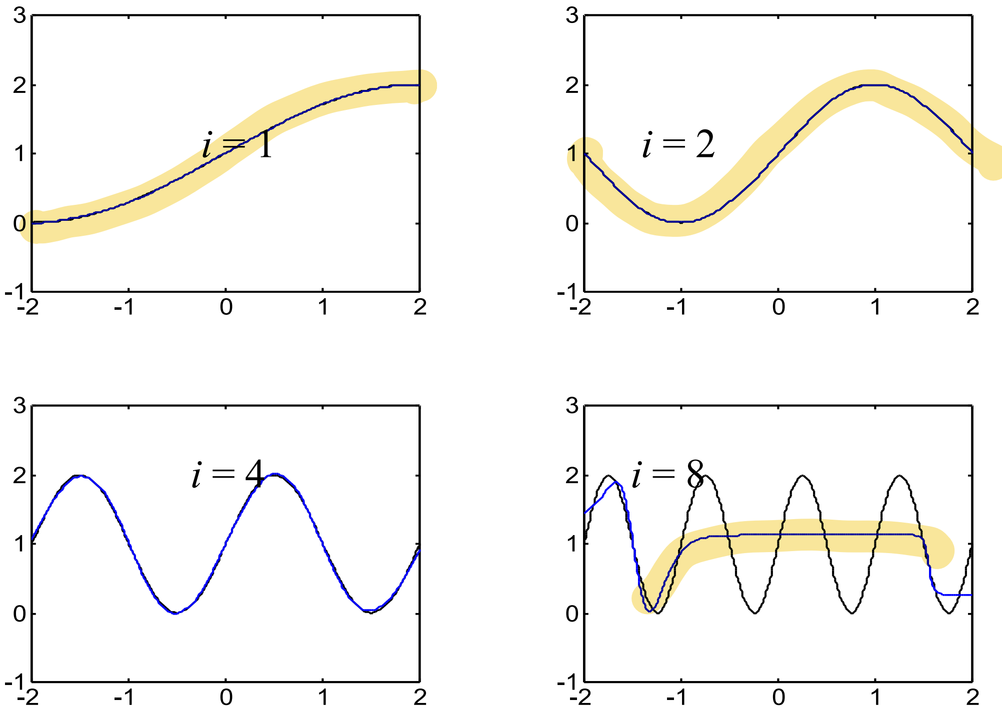
Network Complexity

Choice of Architecture



$$g(p) = 1 + \sin\left(\frac{i\pi}{4}p\right)$$

1-3-1 Network (1 input, 3 hidden, 1 output nodes)



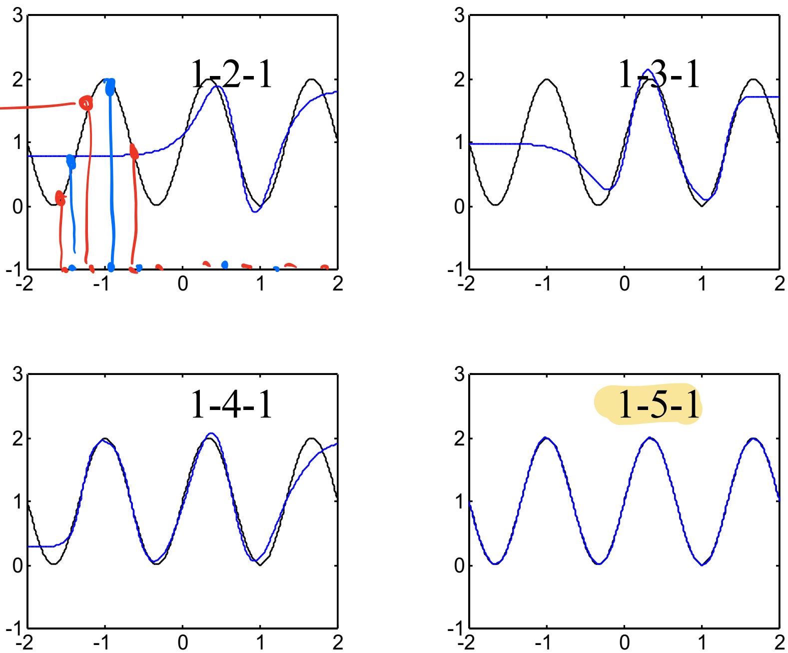
Choice of Network Architecture

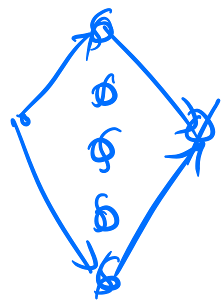
Fix the fn :

Chg. the network

$$g(p) = 1 + \sin\left(\frac{6\pi}{4}p\right)$$

Sampled pts in train set

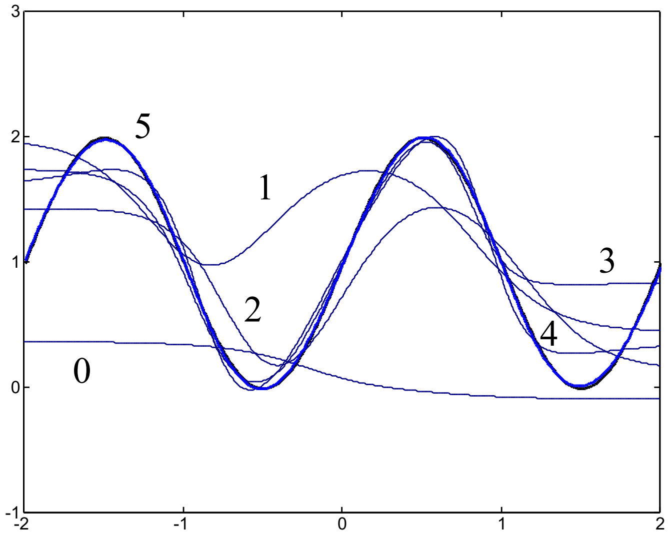


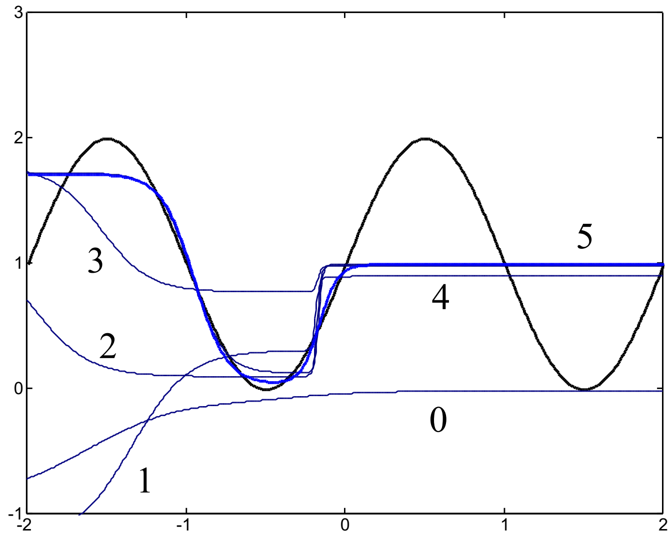


Residual error decreases with O(1/M) where M is the number of hidden units

Convergence in Time

$$g(p) = 1 + \sin(\pi p)$$

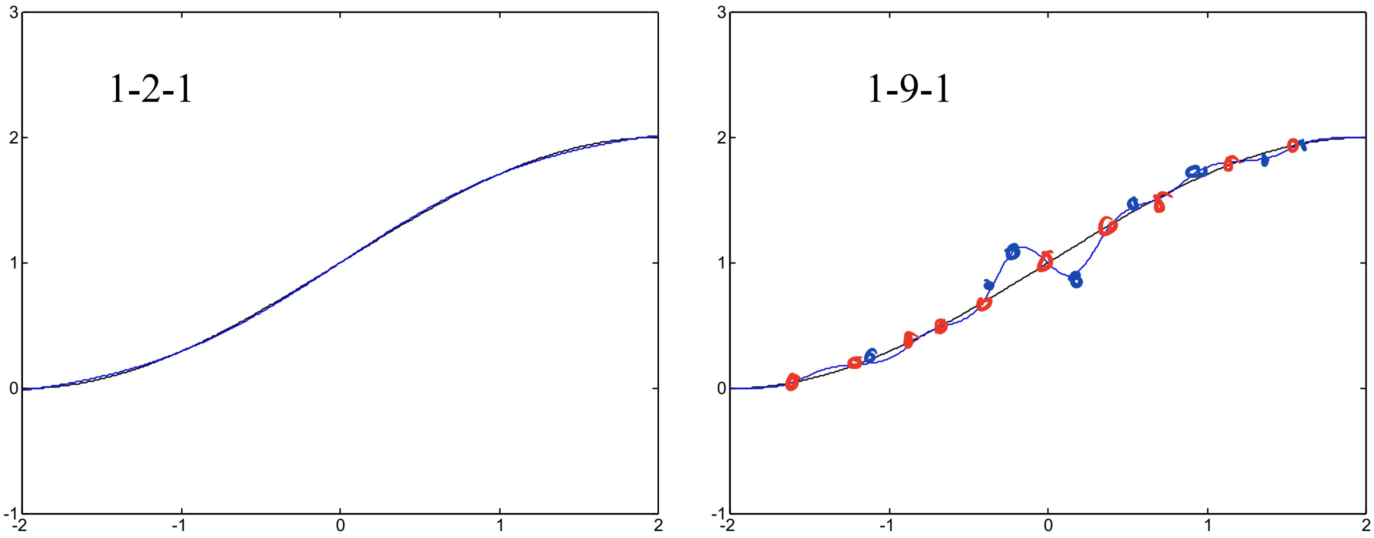




Generalization

$$\{\mathbf{p}_1, \mathbf{t}_1\},\ \{\mathbf{p}_2, \mathbf{t}_2\},\ \dots,\ \{\mathbf{p}_Q, \mathbf{t}_Q\}$$

$$g(p) = 1 + \sin\left(\frac{\pi}{4} p\right) \qquad p = -2, -1.6, -1.2, \dots, 1.6, 2$$



We will see later how to use complex models (e.g. a higher order polynomial or a MLP with large number of nodes) together with regularization to control model complexity.

- E.g. to keep the weight small, so that even if we use complex models, the weight are kept in check so that the overall model is not prone to overfit.