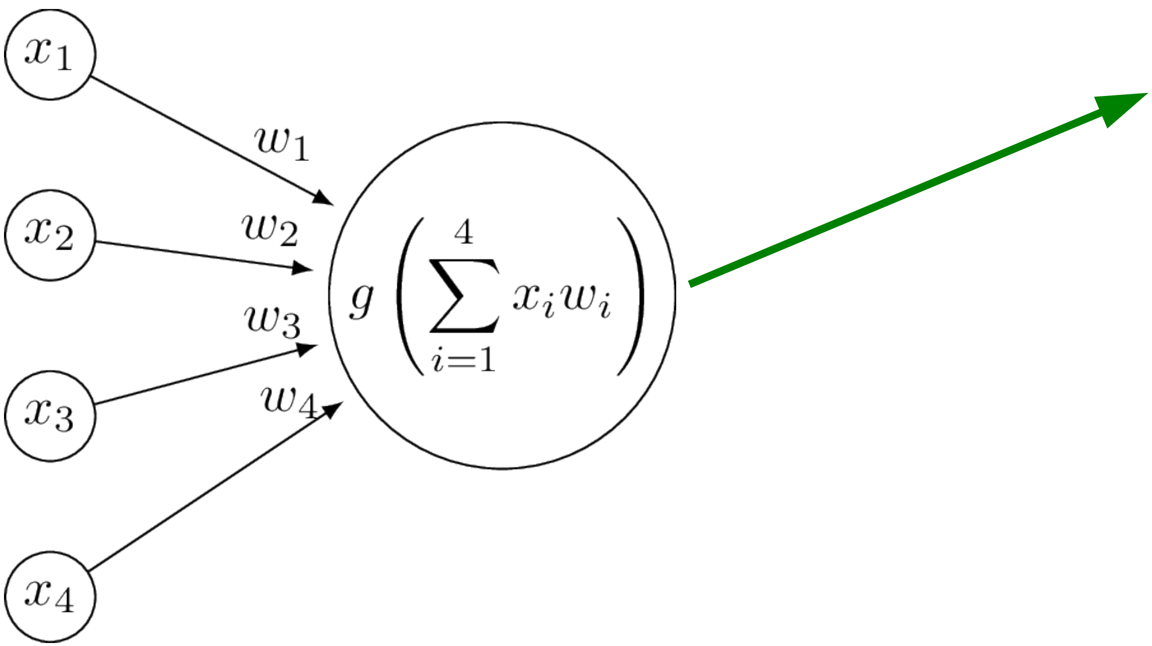
Multi-Layer Neural Networks

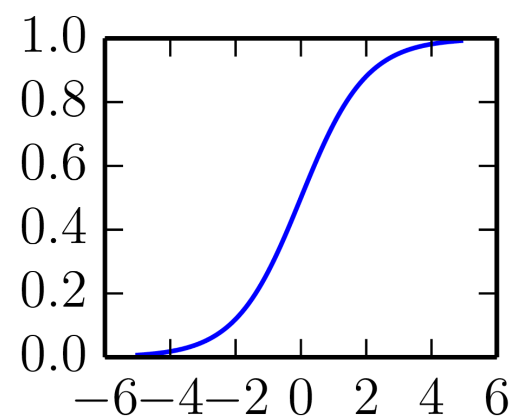
Review

Neuron

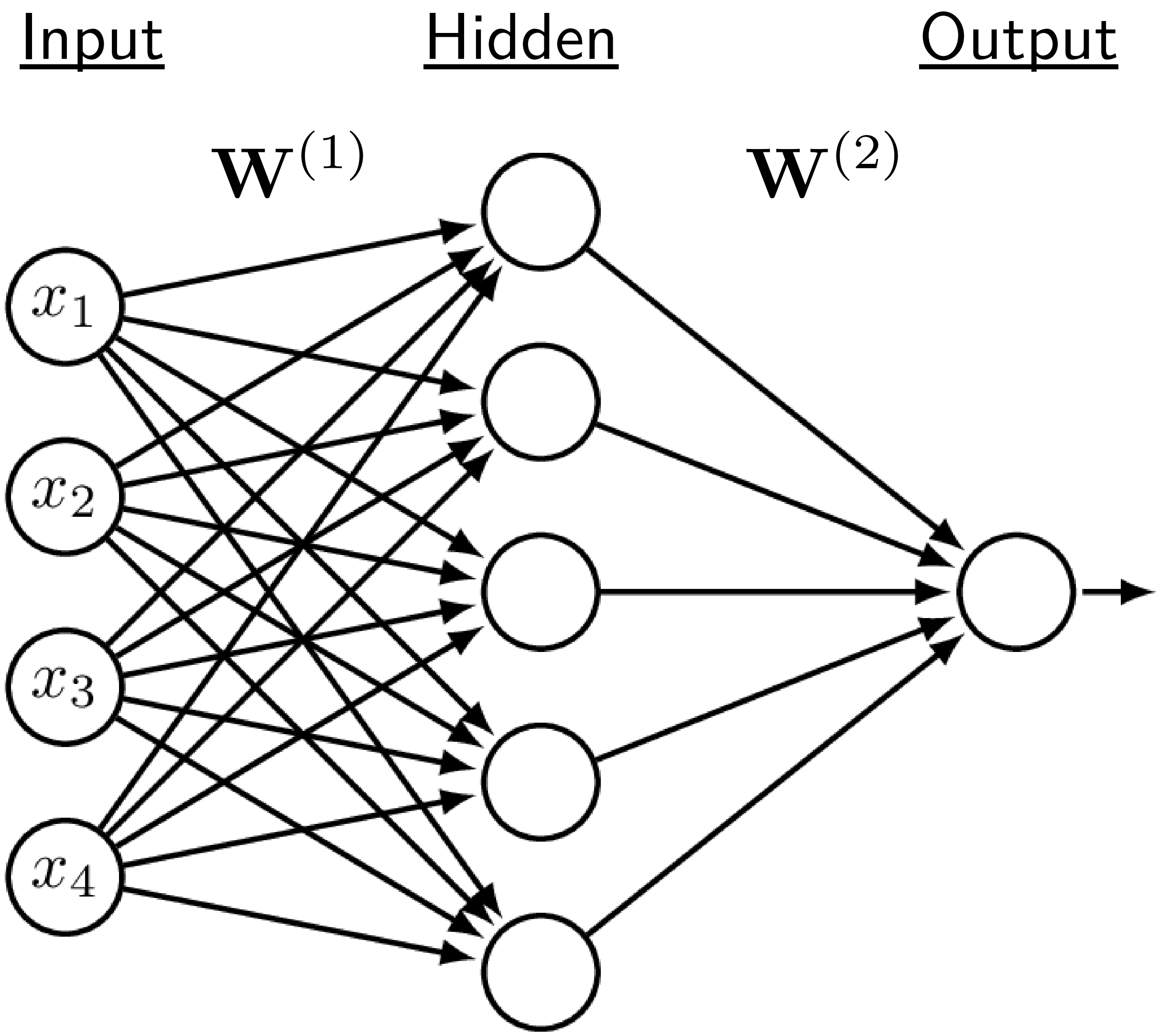


Non-linearity

$$g(a) = \frac{1}{1 + e^{-a}}$$



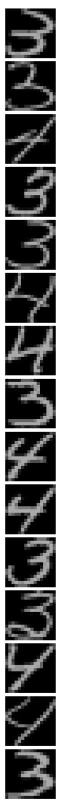
Multi-Layer Networks



Neural Network Example

Training Data

$\mathbf{x}$ y

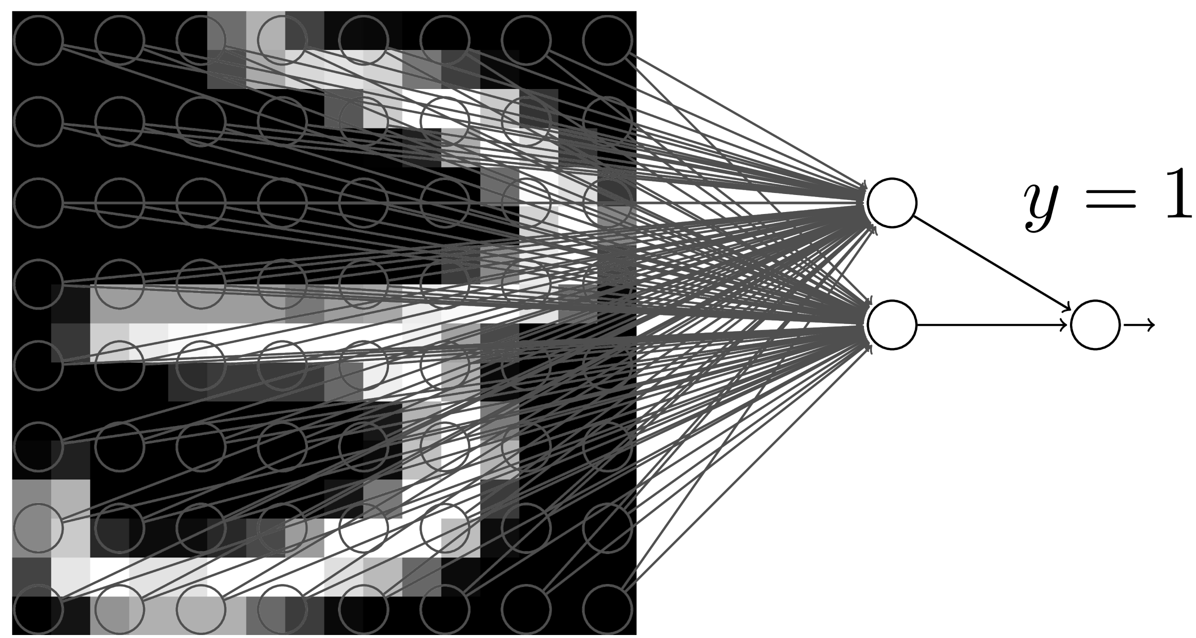


$\rightarrow 1$
$\rightarrow 1$
$\rightarrow 0$
$\rightarrow 1$
$\rightarrow 1$
$\rightarrow 0$
$\rightarrow 0$
$\rightarrow 1$
$\rightarrow 0$
$\rightarrow 0$
$\rightarrow 1$
$\rightarrow 1$
$\rightarrow 0$
$\rightarrow 0$
$\rightarrow 1$

$\vdots$

Network

$\mathbf{x}$



Backpropagation

- Activation at the output layer:

$$a_k = o\left(\sum_j w_{j,k}^{(2)}\, g\left(\sum_i w_{i,j}^{(1)} x_i\right)\right)$$

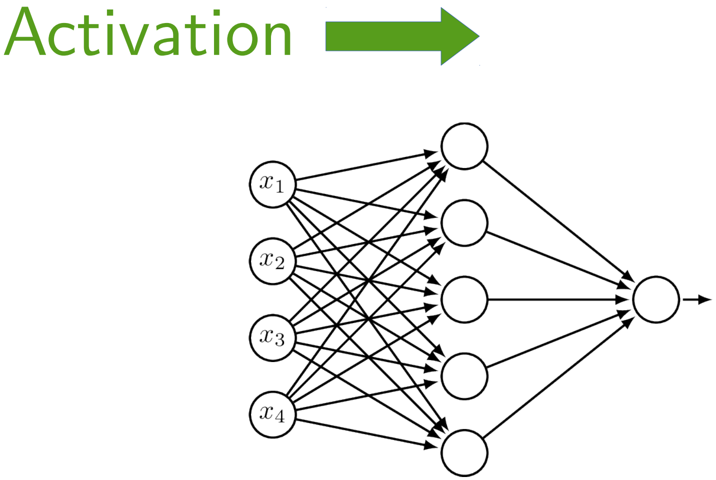
- Here o is the activation function at the output layer. Units at the input layer are indexed with *i*, hidden with *j* and output with *k*.
- Error metric, assuming multiple output units:

$$Error = \frac{1}{2}\sum_k (y_k - a_k)^2$$

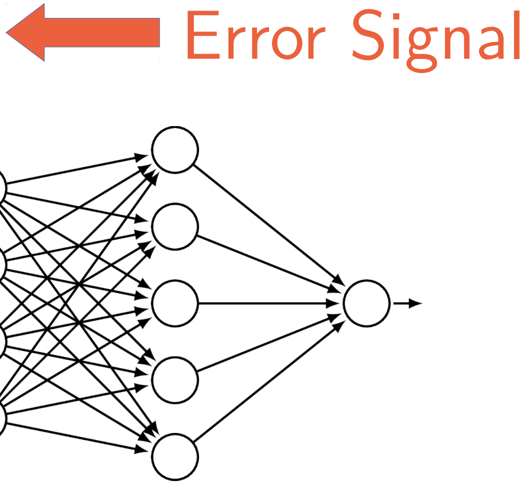
- Now just compute $\dfrac{\partial\, Error}{\partial\, w_{j,k}^{(2)}}$ and $\dfrac{\partial\, Error}{\partial\, w_{i,j}^{(1)}}$.

Backpropagation Algorithm

- Forward Pass:



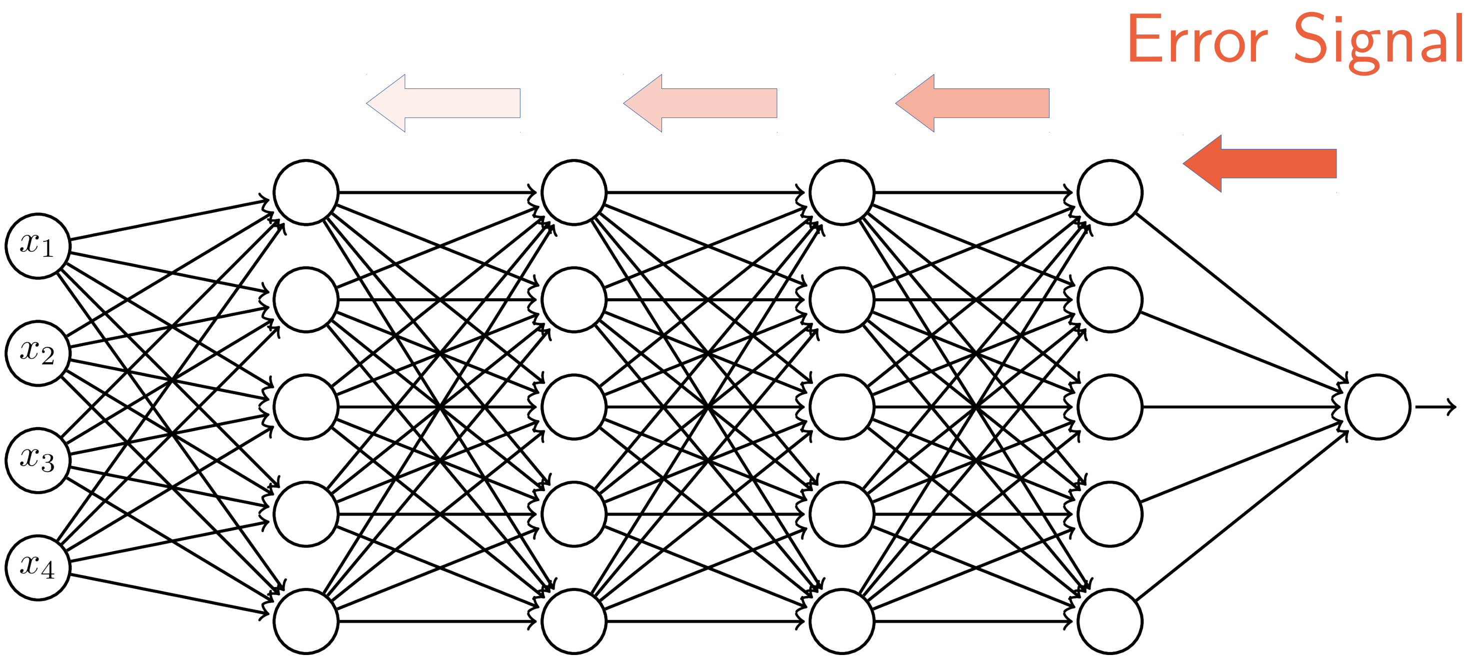
- Backward Pass:



Deep vs. Shallow Networks

- How best to add capacity?
  - More units in a single hidden layer?
    - Three layer networks are universal approximators: with enough units any continuous function can be approximated
    - Adding layers makes the learning problem harder...

Vanishing Gradients



Advantages of Deep Architectures

- There are tasks that require exponentially many hidden units for a three-layer architecture, but only polynomially many with more hidden layers
- The best hand-coded image processing algorithms have deep structure
- The brain has a deep architecture

The Deep Learning "Revolution"

- Geoff Hinton introduced a simple idea in 2006
- Greedy, Layer-Wise, Unsupervised Pre-Training
  - Train the first hidden layer to re-represent the input.
  - Train the second hidden layer to re-represent the first hidden layer
  - …
  - Fine-tune the entire network using backpropagation on labeled data

G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006.

The Flood Gates Open

- Better Hardware

  GPGPU

  Cluster Computing

- Massive Data Sets

  Street View House Numbers

  Kaggle

- Better Training Algorithms

  RMSProp

  Dropout

- New Architectures

  Maxout

  Rectified Linear Units