DS 4400

Machine Learning and Data Mining I

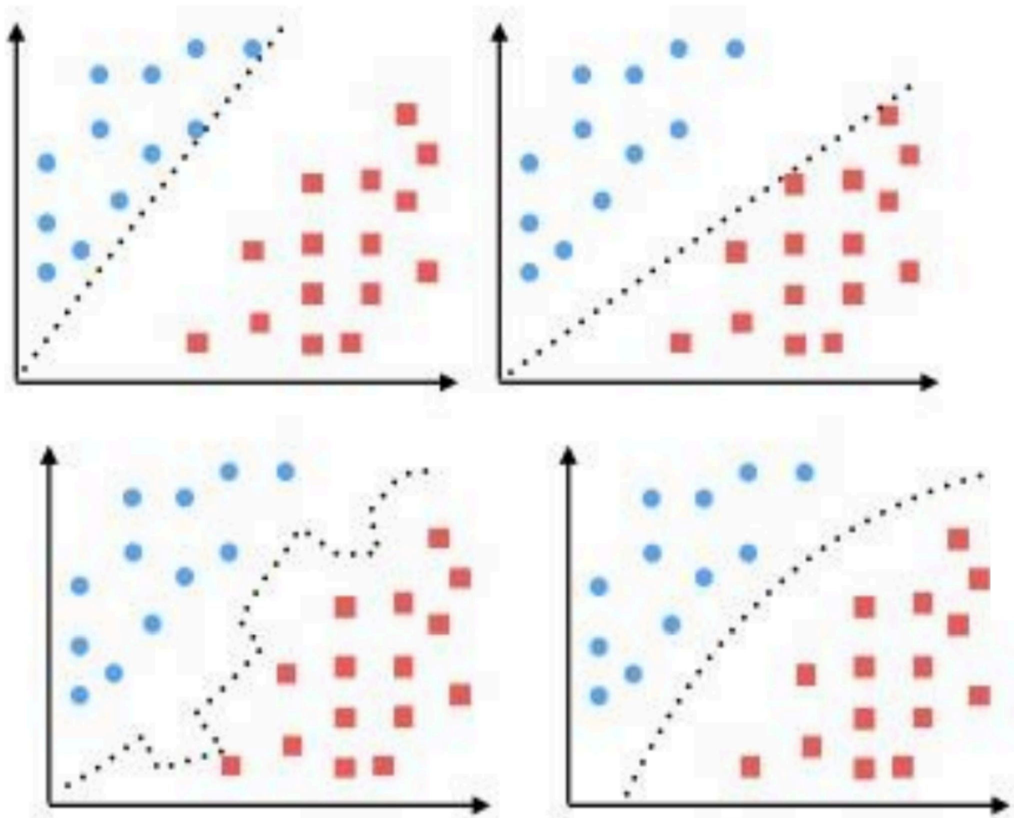
Alina Oprea
Associate Professor
Khoury College of Computer Science
Northeastern University

November 5 2020

Outline

- Introduction to Deep Learning
- Neural Network Architectures
- Feed Forward Neural Networks
  - Forward Propagation
  - Hyper-parameters
  - Activations

Linear vs Non-Linear Classifiers



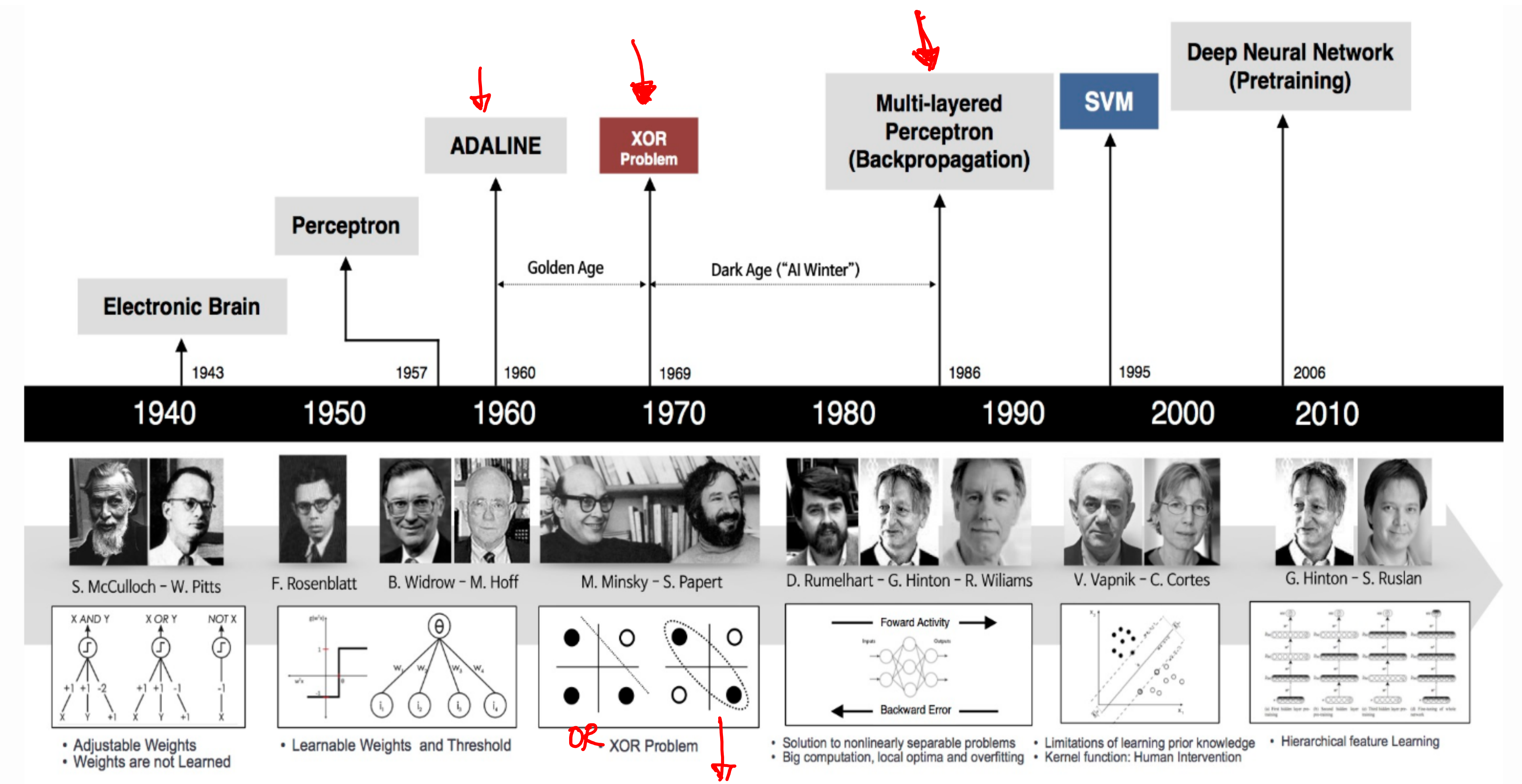
- LOGISTIC REGRESSION
- LDA
- PERCEPTRON

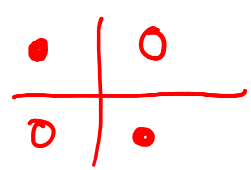
- KNN
- DECISION TREES
- ENSEMBLES
- NAIVE BAYES

Comparing classifiers

Algorithm	Interpretable	Model size	Predictive accuracy	Training time	Testing time
Logistic regression	HIGH	SMALL	LOWER	LOW	LOW
kNN	MEDIUM	LARGE	LOWER	O	HIGH
LDA	MEDIUM	SMALL	LOWER	LOW	LOW
Decision trees	HIGH	MEDIUM	LOWER	MEDIUM	LOW
Ensembles	LOW	LARGE	HIGH	HIGH	HIGH
Naïve Bayes	MEDIUM	SMALL	LOWER	MEDIUM	LOW
Neural Networks	LOW	LARGE	HIGH	HIGH	LOW / MEDIUM

History of Deep Learning



XOR 

References

- Deep Learning books
  - https://d2l.ai/ (D2L)
  - https://www.deeplearningbook.org/ (advanced)
- Stanford notes on deep learning
  - http://cs229.stanford.edu/notes/cs229-notes-deep_learning.pdf
- History of Deep Learning
  - https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html

Deep Learning

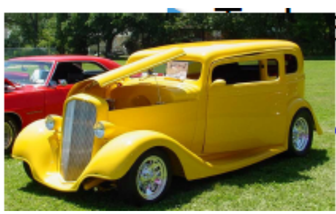
- **The traditional model of pattern recognition (since the late 50's)**
  - Fixed/engineered features (or fixed kernel) + trainable classifier

INPUT

hand-crafted Feature Extractor → "Simple" Trainable Classifier →

FEATURES

- End-to-end learning / Feature learning / Deep learning

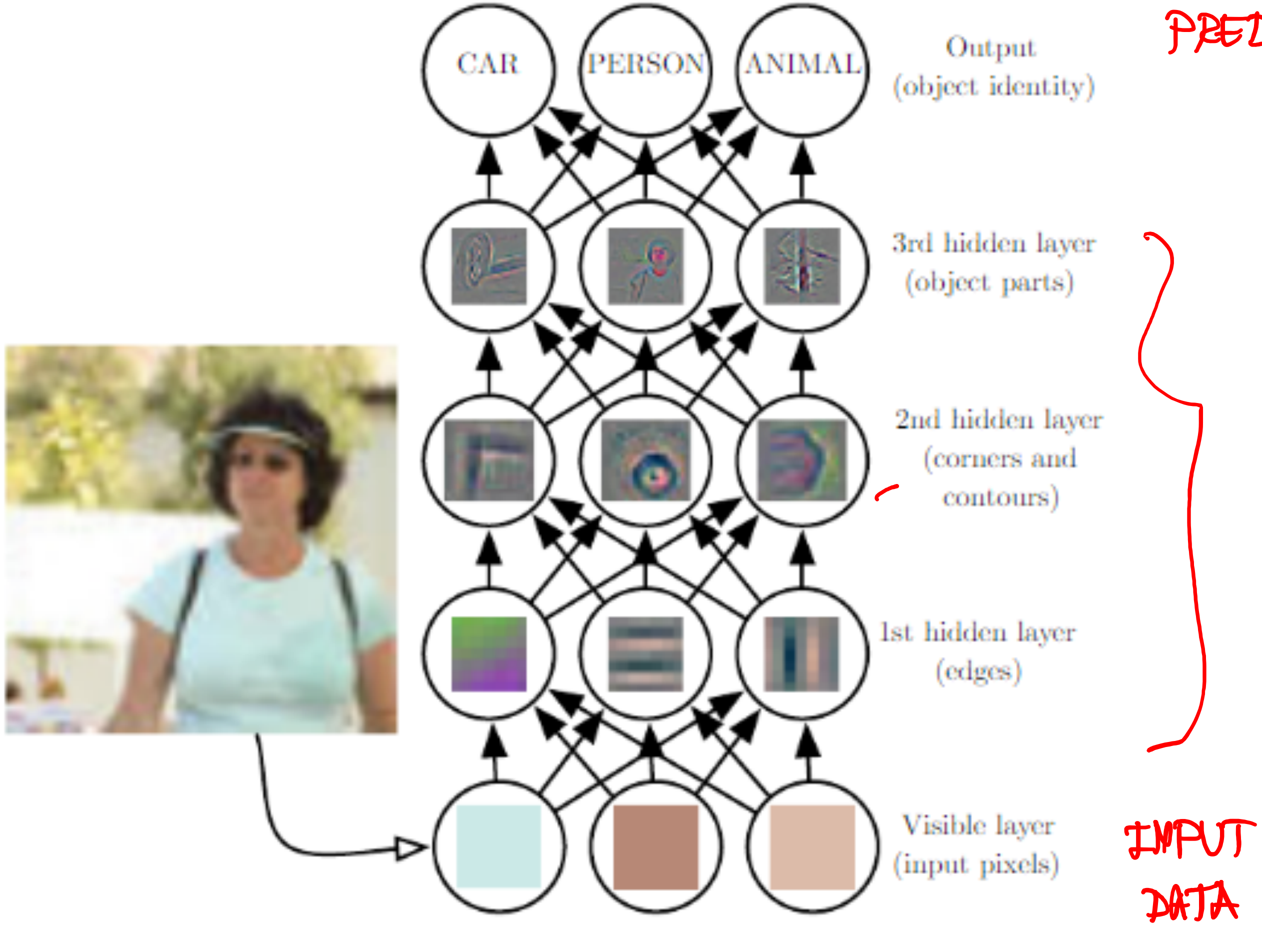


INPUT

→ Trainable Feature Extractor → Trainable Classifier →

PREDICTION

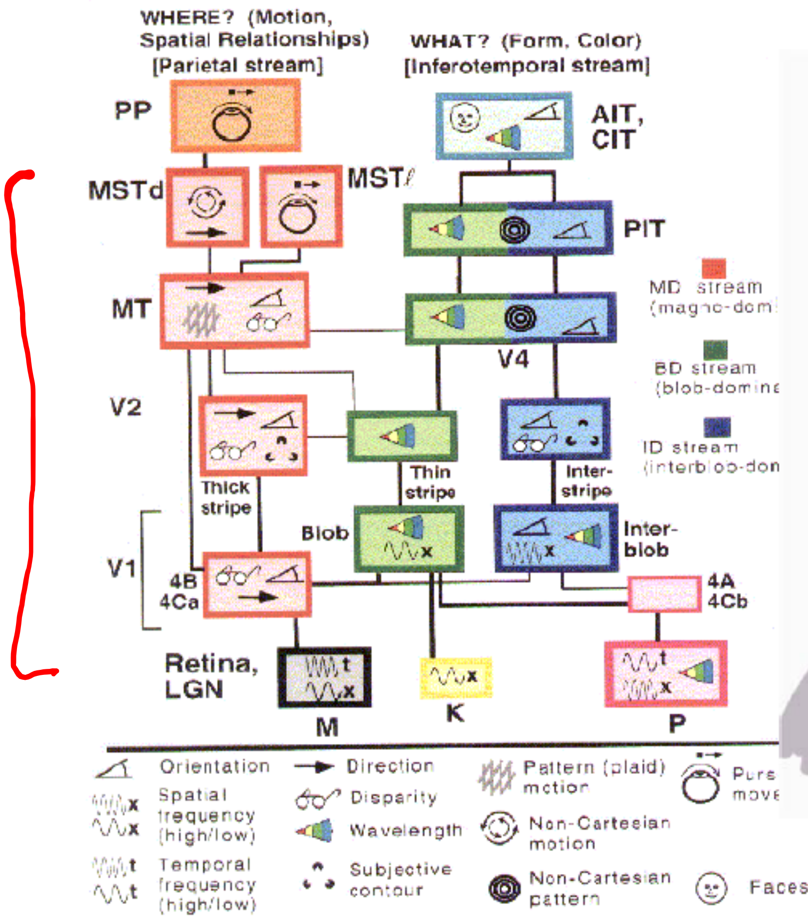
Learning Representations



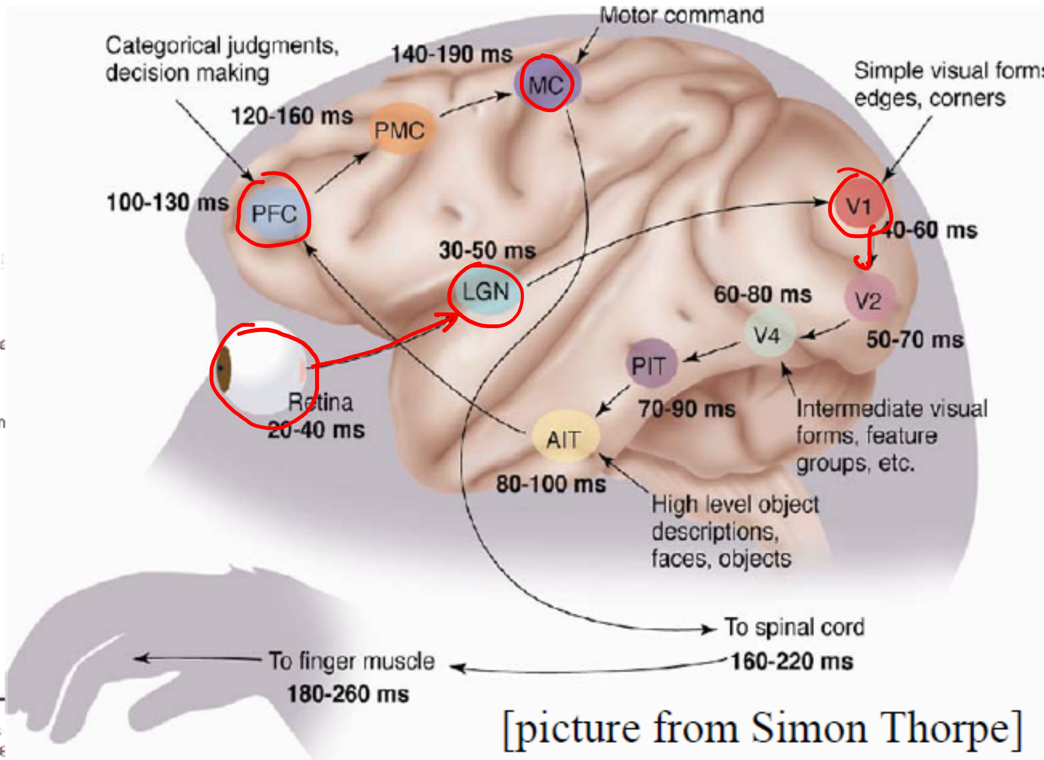
Deep Learning addresses the problem of learning hierarchical representations

The Visual Cortex is Hierarchical

- The ventral (recognition) pathway in the visual cortex has multiple stages
- Retina - LGN - V1 - V2 - V4 - PIT - AIT ....
- Lots of intermediate representations



[Gallant & Van Essen]



[picture from Simon Thorpe]

Neural Function

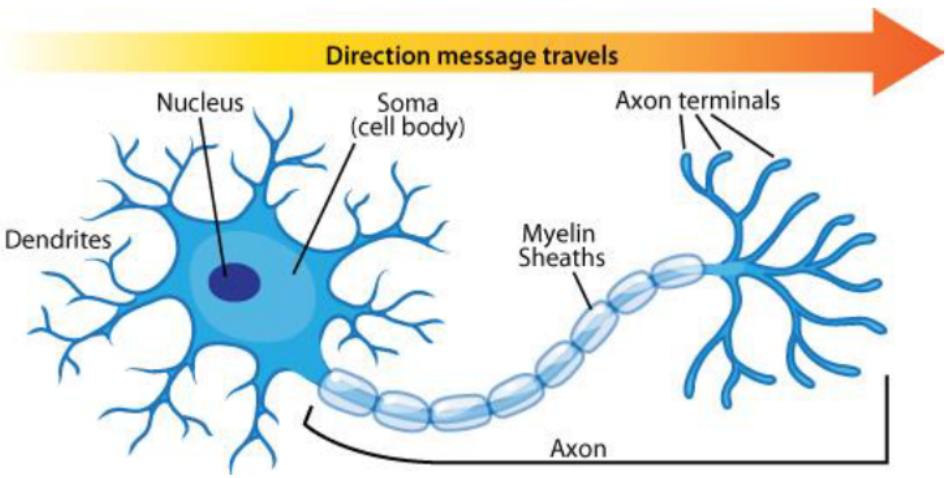
- Brain function (thought) occurs as the result of the firing of **neurons**

- Neurons connect to each other through **synapses**, which propagate **action potential** (electrical impulses) by releasing **neurotransmitters**
  - Synapses can be **excitatory** (potential-increasing) or **inhibitory** (potential-decreasing), and have varying **activation thresholds**
  - Learning occurs as a result of the synapses' **plasticicity**: They exhibit long-term changes in connection strength

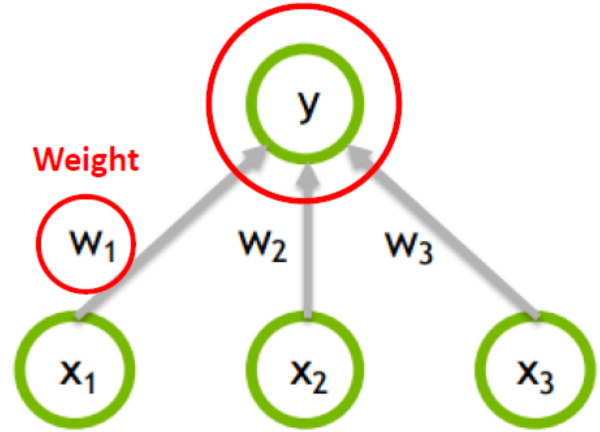
- There are about 10^{11} neurons and about 10^{14} synapses in the human brain!

Analogy to Human Brain

Human Brain



Biological Neuron



$y=F(\underbrace{w_1x_1+w_2x_2+w_3x_3}_{})$

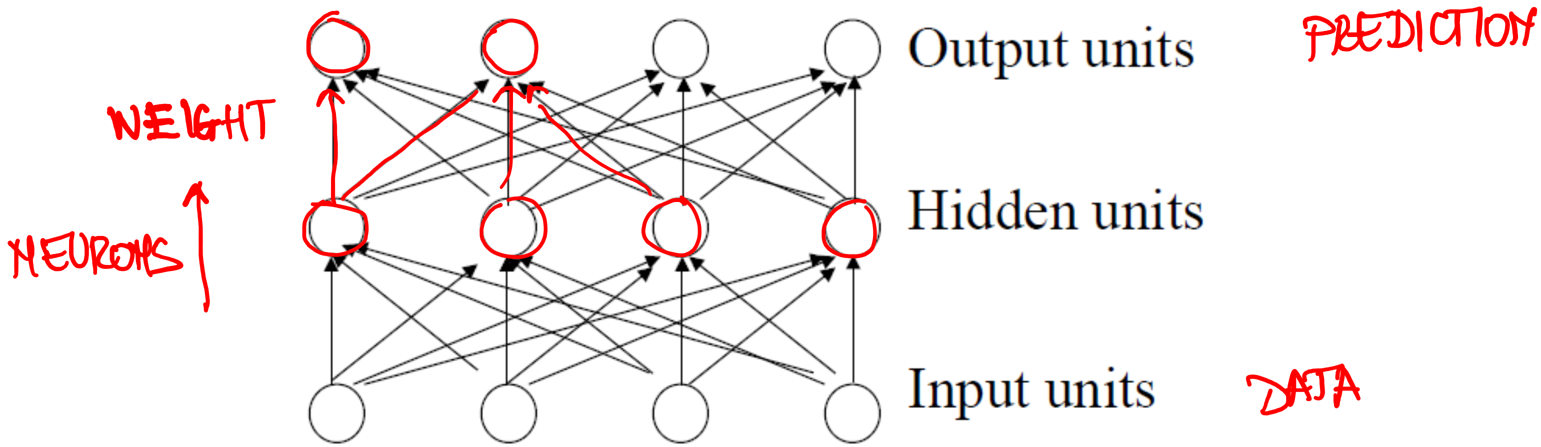
$F(x)=\max(0,x)$ ACTIVATION

Artificial Neuron

Neural Networks

- Origins: Algorithms that try to mimic the brain.
- Very widely used in 80s and early 90s; popularity diminished in late 90s.
- Recent resurgence: State-of-the-art technique for many applications
- Artificial neural networks are not nearly as complex or intricate as the actual brain structure

Neural Networks



Layered feed-forward network

- Neural networks are made up of **nodes** or **units**, connected by **links**
- Each link has an associated **weight** and **activation level**
- Each node has an **input function** (typically summing over weighted inputs), an **activation function**, and an **output**

Example

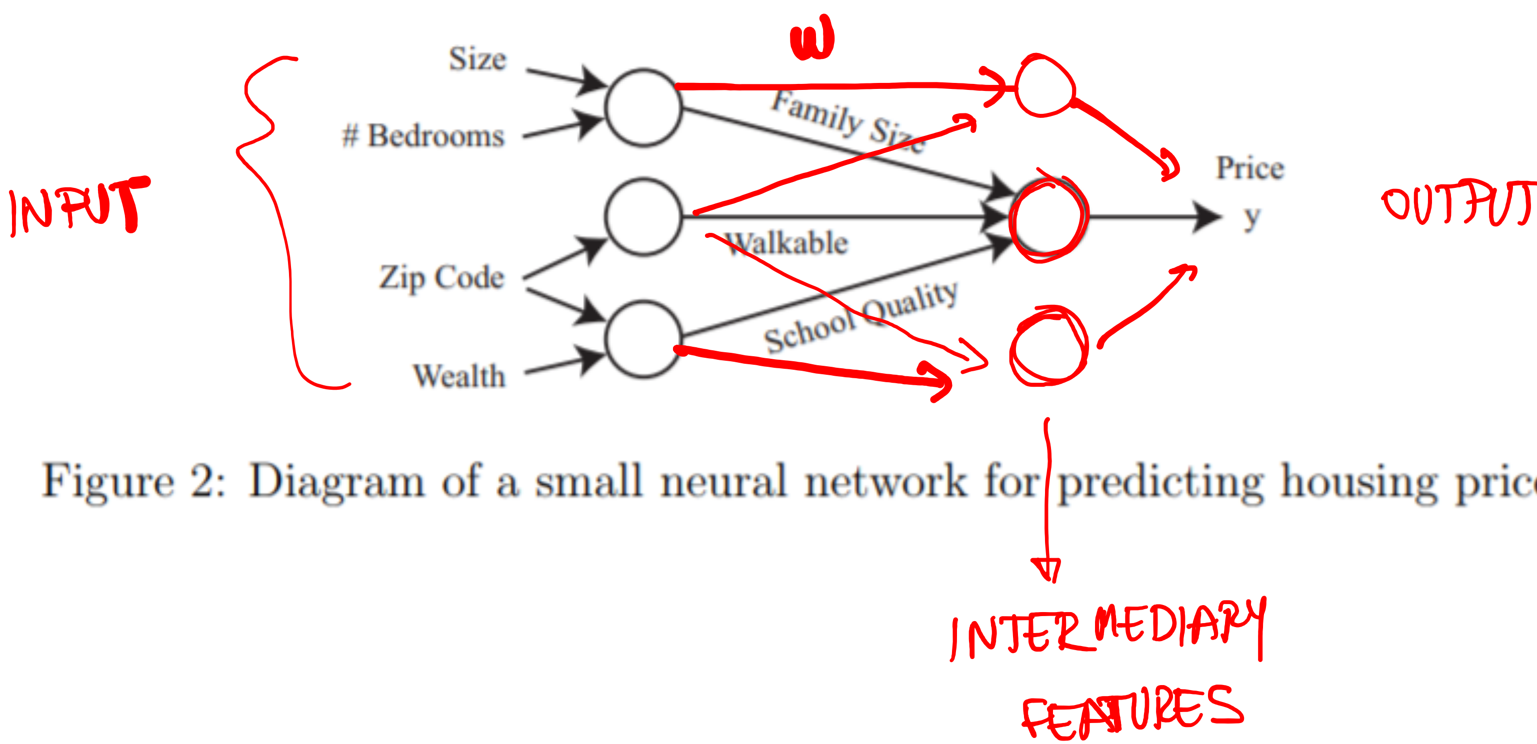
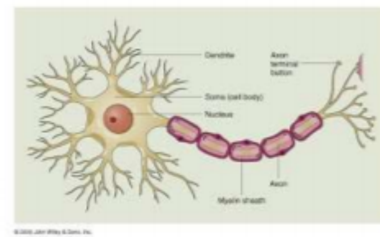


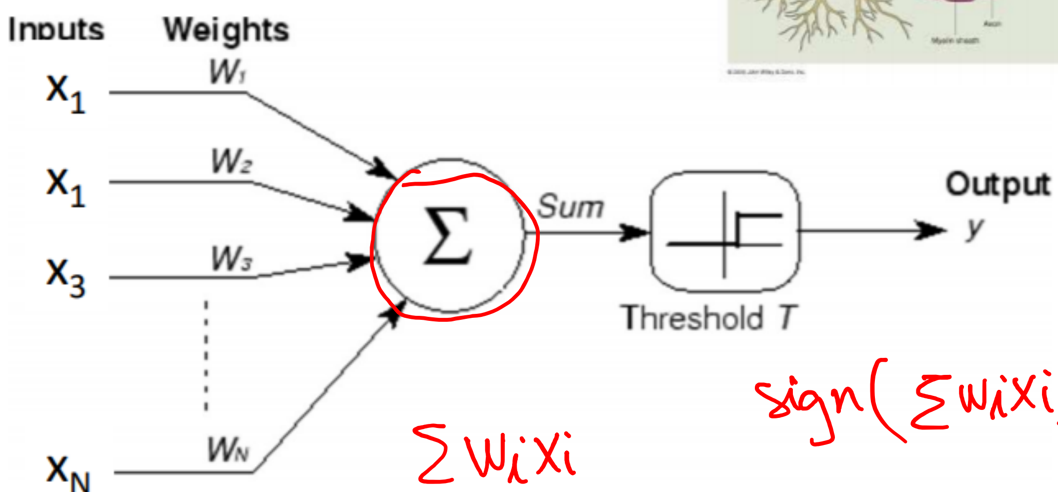
Figure 2: Diagram of a small neural network for predicting housing prices.

- Provide as input only training data: input and label
- Neural Networks automatically learn intermediate features!

Perceptron

$y_i \in \{-1, 1\}$





$\sum w_i x_i$

$sign(\sum w_i x_i)$

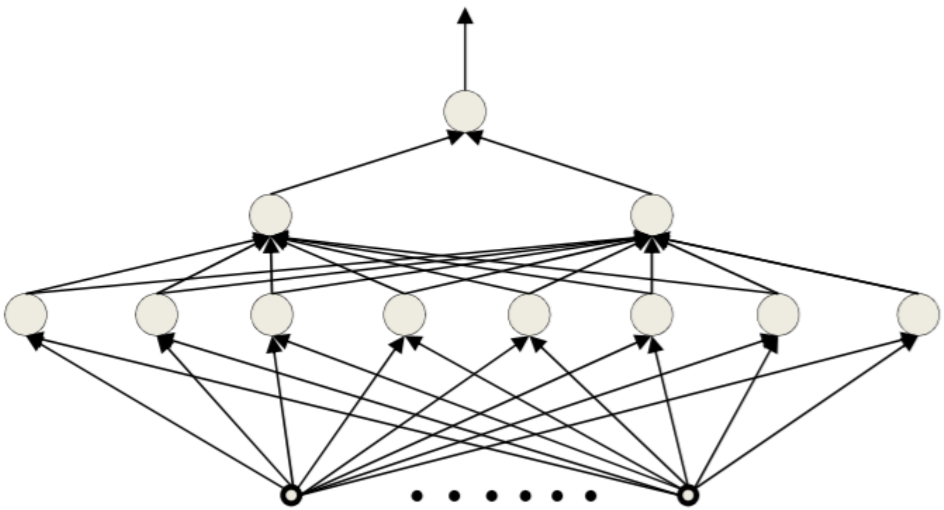
$T = 0$

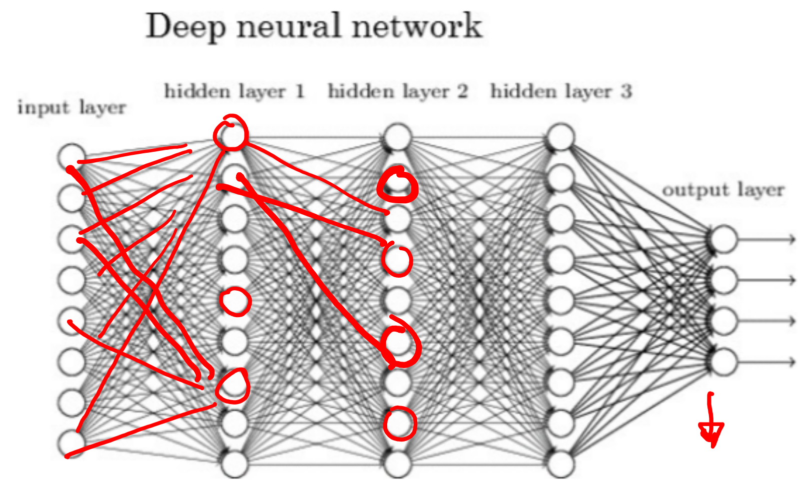
$$y = \begin{cases} 1 \; if \; \sum_i w_i \mathrm{x}_i \geq T \\ 0 \; else \end{cases}$$

- A threshold unit
  - "Fires" if the weighted sum of inputs exceeds a threshold

LINEAR MODEL

Multi-Layer Perceptron





MULTIPLE NEURONS IN PARALLEL

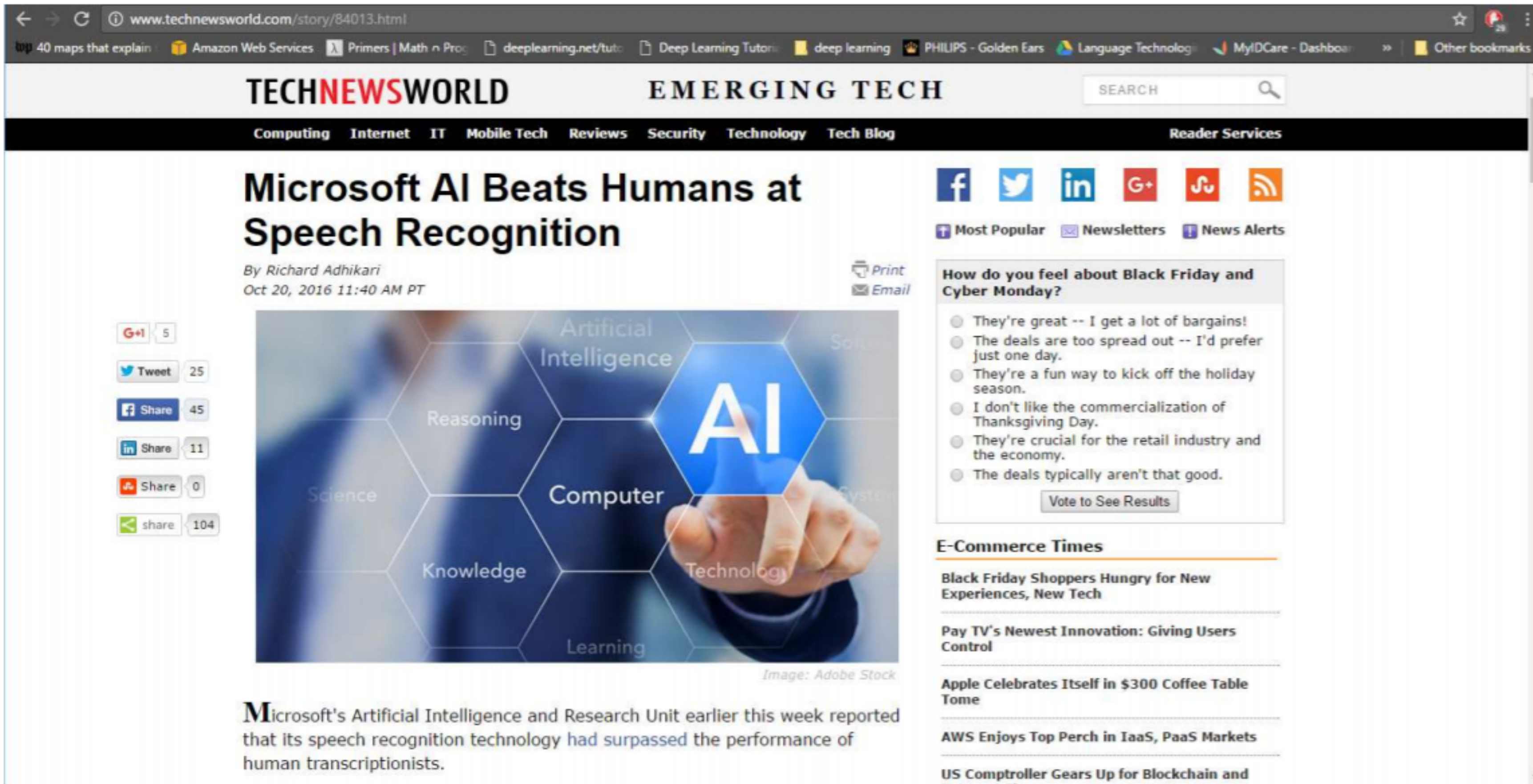
- A network of perceptrons
  - Generally "layered"

NON-LINEAR MODEL

FULLY CONNECTED

Deep Learning Applications

Success stories: Speech recognition



www.technewsworld.com/story/84013.html

TECHNEWSWORLD EMERGING TECH

Computing Internet IT Mobile Tech Reviews Security Technology Tech Blog Reader Services

Microsoft AI Beats Humans at Speech Recognition

By Richard Adhikari
Oct 20, 2016 11:40 AM PT

Image: Adobe Stock

Microsoft's Artificial Intelligence and Research Unit earlier this week reported that its speech recognition technology had surpassed the performance of human transcriptionists.

How do you feel about Black Friday and Cyber Monday?

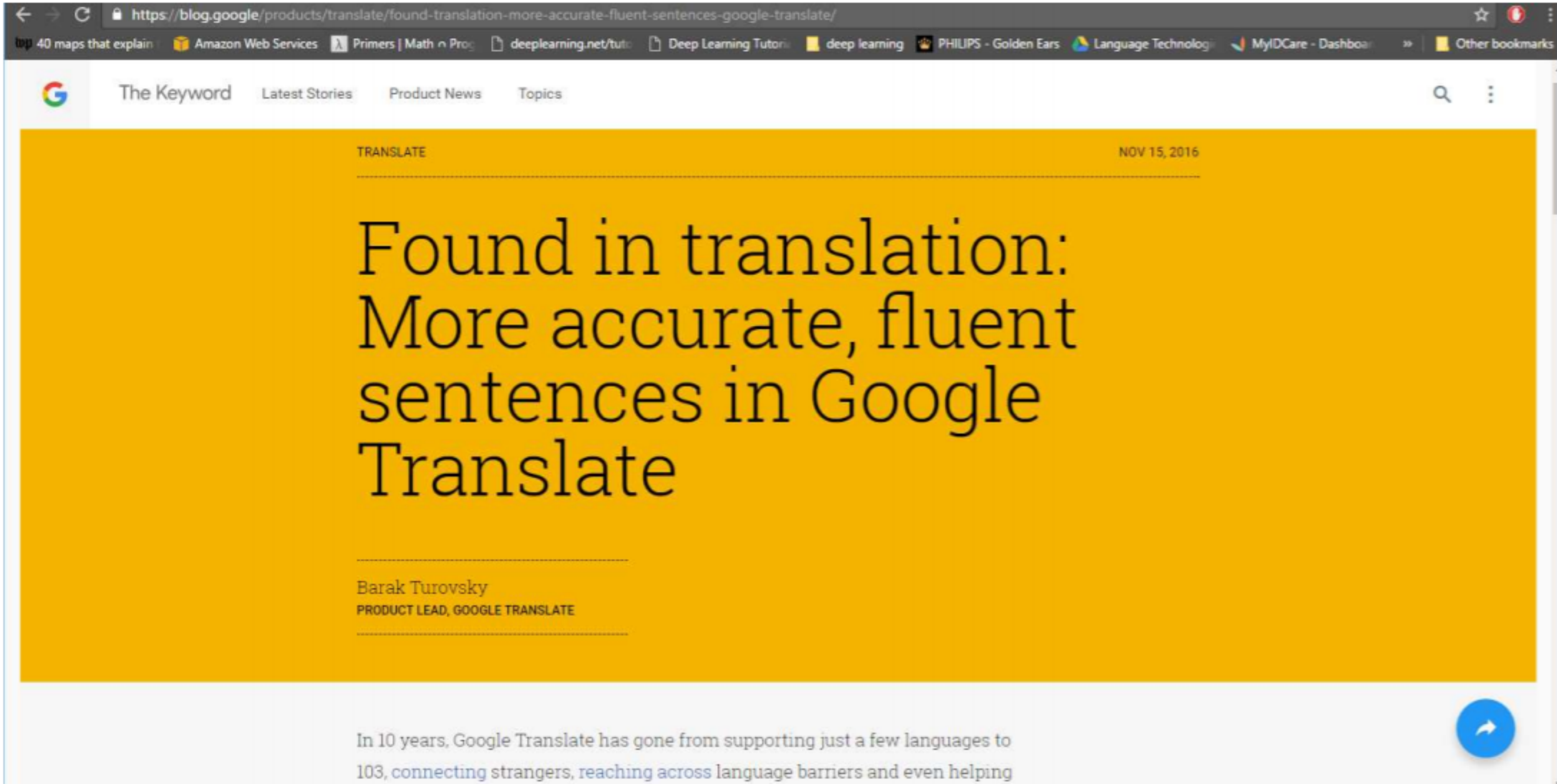
- They're great -- I get a lot of bargains!
- The deals are too spread out -- I'd prefer just one day.
- They're a fun way to kick off the holiday season.
- I don't like the commercialization of Thanksgiving Day.
- They're crucial for the retail industry and the economy.
- The deals typically aren't that good.

Vote to See Results

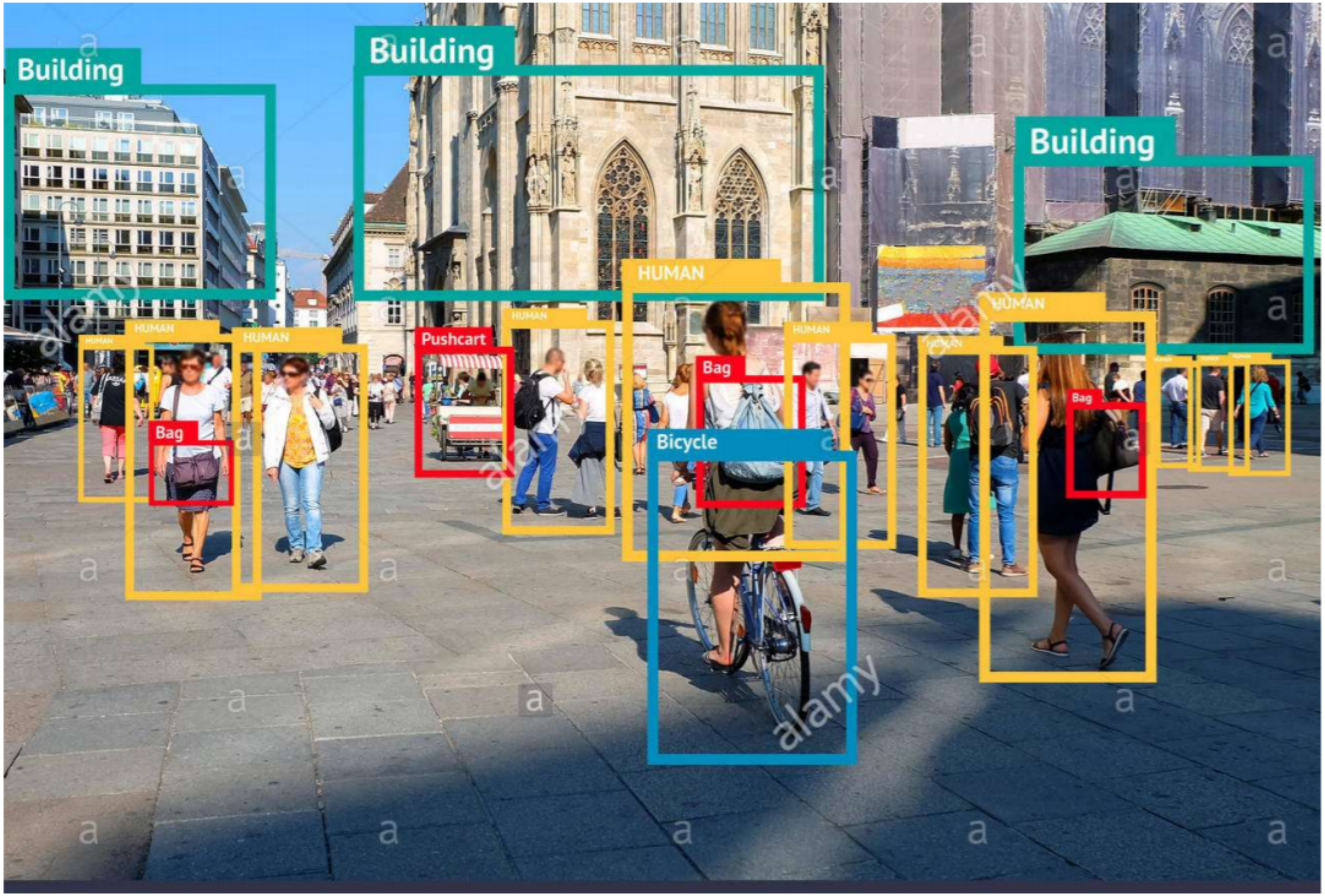
E-Commerce Times

- Black Friday Shoppers Hungry for New Experiences, New Tech
- Pay TV's Newest Innovation: Giving Users Control
- Apple Celebrates Itself in $300 Coffee Table Tome
- AWS Enjoys Top Perch in IaaS, PaaS Markets
- US Comptroller Gears Up for Blockchain and

Success stories: Machine Translation



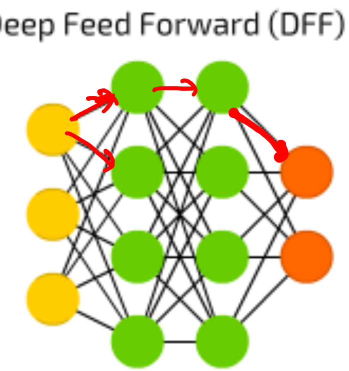
Success stories: Image segmentation



Neural Network Architectures

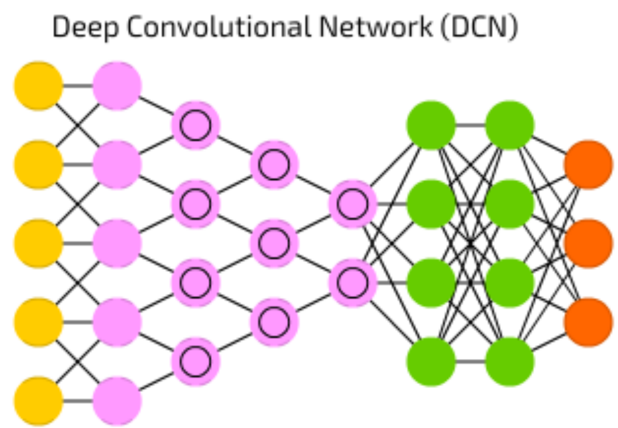
Feed-Forward Networks

- Neurons from each layer connect to neurons from next layer



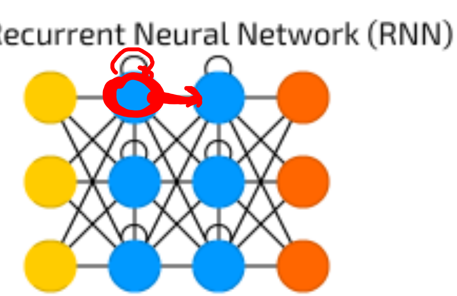
Convolutional Networks

- Includes convolution layer for feature reduction
- Learns hierarchical representations



Recurrent Networks

- Keep hidden state
- Have cycles in computational graph



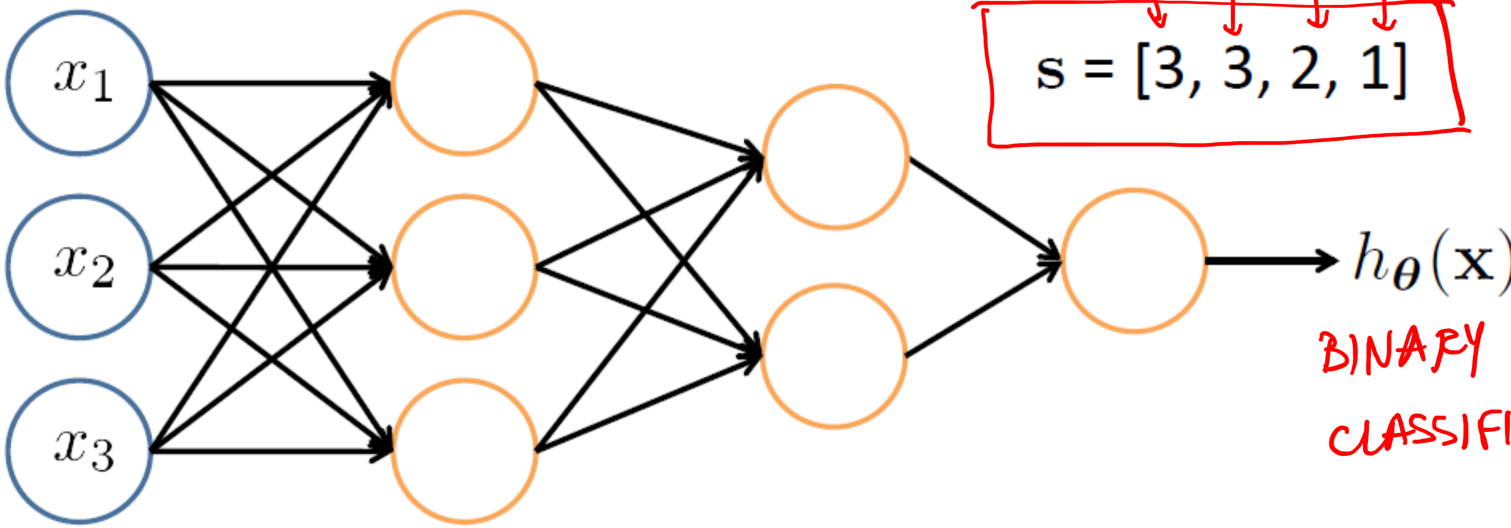
Feed-Forward Networks

L0 L1 L2 L3

$\boxed{\mathbf{s} = [3, 3, 2, 1]}$

X

INPUT



BINARY CLASSIFICATION

Layer 0 Layer 1 Layer 2 Layer 3

OUTPUT

L denotes the number of layers

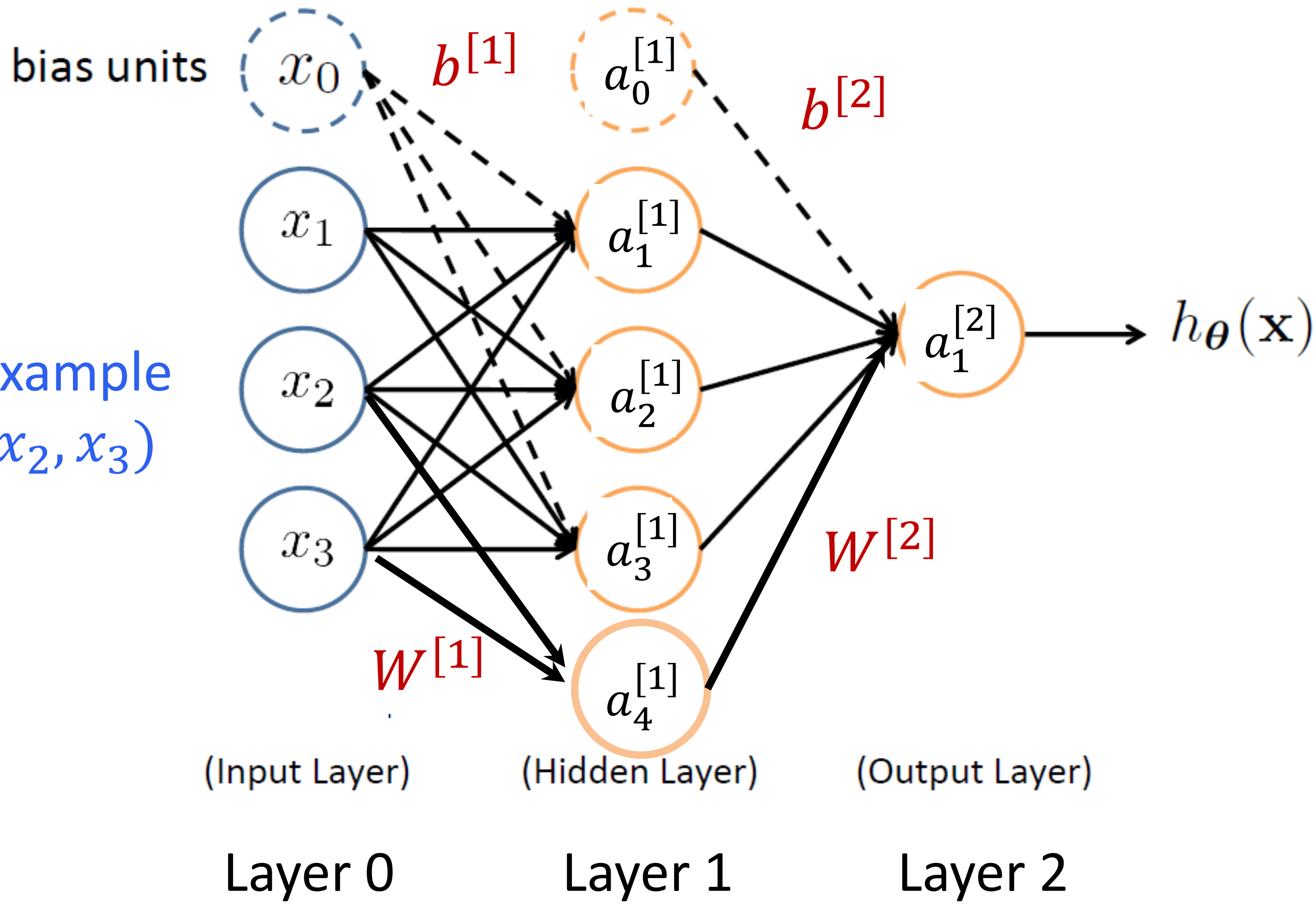
$\mathbf{s} \in {\mathbb{N}^+}^L$ contains the numbers of nodes at each layer

- Not counting bias units
- Typically, $\boxed{s_0 = d}$ (# input features) and $\boxed{s_{L\text{-}1} = K}$ (# classes)

Feed-Forward NN

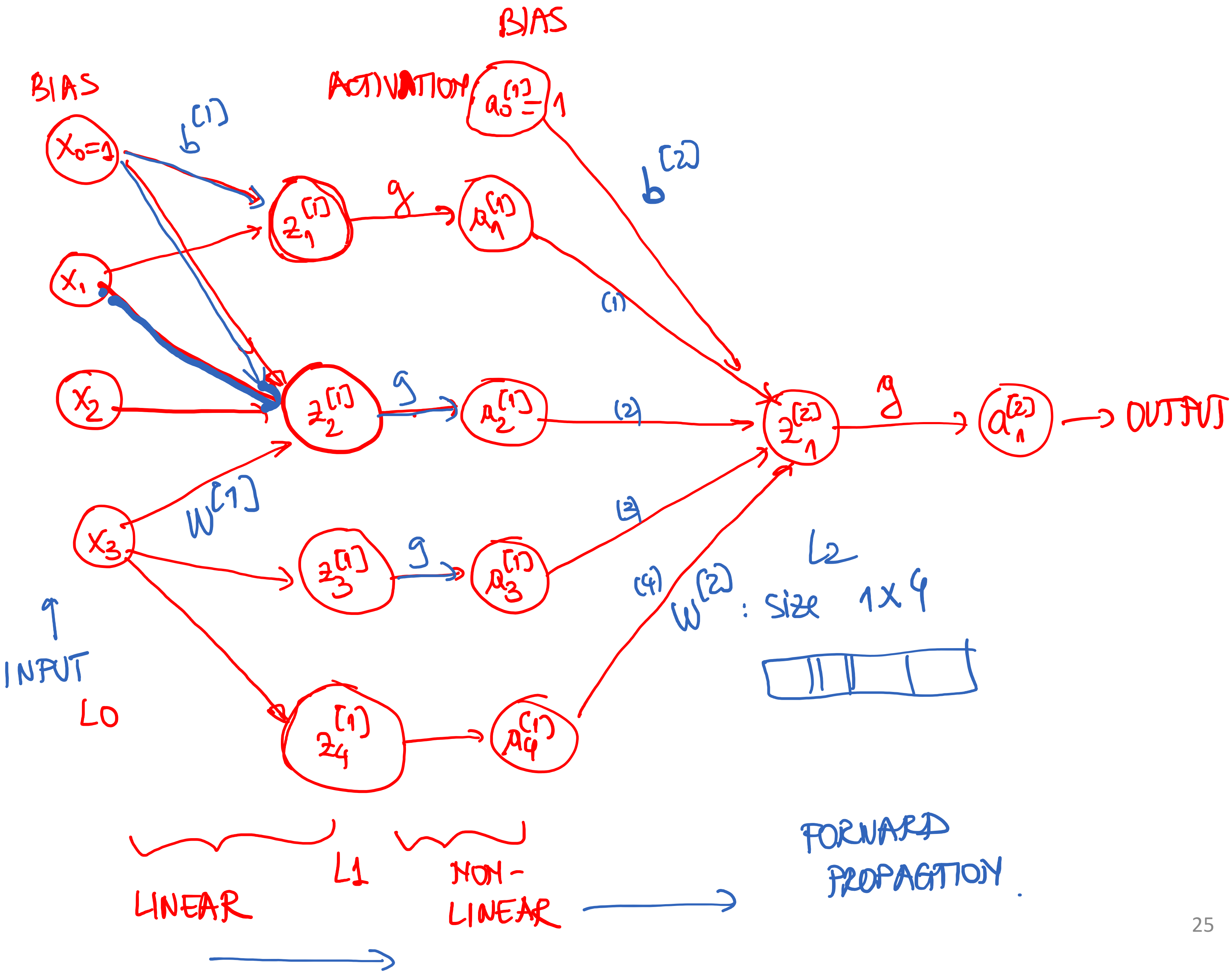
- Hyper-parameters
  - Number of layers
  - Architecture (how layers are connected)
  - Number of hidden units per layer
  - Number of units in output layer
  - Activation functions
- Other
  - Initialization
  - Regularization

Feed-Forward Neural Network



Training example
$x = (x_1, x_2, x_3)$

No cycles $\theta = (b^{[1]}, W^{[1]}, b^{[2]}, W^{[2]})$


BIAS
BIAS
ACTIVATION
$a_0^{[1]} = 1$
$x_0 = 1$
x_1
x_2
x_3
$b^{[1]}$
$W^{[1]}$
$z_1^{[1]}$
$z_2^{[1]}$
$z_3^{[1]}$
$z_4^{[1]}$
g
$a_1^{[1]}$
$a_2^{[1]}$
$a_3^{[1]}$
$a_4^{[1]}$
$b^{[2]}$
(1)
(2)
(3)
(4)
$z_1^{[2]}$
$a_1^{[2]}$
OUTPUT
INPUT
L0
L1
L2
$W^{[2]}$: size 1 x 4
LINEAR
NON-LINEAR
FORWARD PROPAGATION

COMPUTATIONS (L1)

$\boxed{\text{LINEAR}}$

$$\begin{cases} z_1^{[1]} = W_1^{[1]} \cdot x + b_1^{[1]} \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \\ \vdots \\ z_4^{[1]} = W_4^{[1]} \cdot x + b_4^{[1]} \end{cases}$$

3

$$\underset{(4\times1)}{z^{[1]}} = \begin{bmatrix} z_1^{[1]} \\ \vdots \\ z_4^{[1]} \end{bmatrix} \underset{4}{=} \underbrace{\begin{bmatrix} W_1^{[1]} \\ W_2^{[1]} \\ \vdots \\ W_4^{[1]} \end{bmatrix}}_{\text{MATRIX } W^{[1]} : \text{ size } (4\times3)} \cdot \underset{(3\times1)}{x} + \underset{(4\times1)}{b^{[1]}} \; ; \quad b^{[1]} = \begin{bmatrix} b_1^{[1]} \\ \vdots \\ b_4^{[1]} \end{bmatrix}$$

ACTIVATIONS: $a_1^{[1]} = g\left(z_1^{[1]}\right), \ldots, a_4^{[1]} = g\left(z_4^{[1]}\right)$

$$a^{[1]} = \begin{bmatrix} a_1^{[1]} \\ \vdots \\ a_4^{[1]} \end{bmatrix} = g\left(z^{[1]}\right)$$

Vectorization

$$\left\{\begin{array}{lcl} z_1^{[1]} = W_1^{[1]} \; x + b_1^{[1]} & \text{and} & a_1^{[1]} = g(z_1^{[1]}) \\ \vdots & \vdots & \vdots \\ z_4^{[1]} = W_4^{[1]} \; x + b_4^{[1]} & \text{and} & a_4^{[1]} = g(z_4^{[1]}) \end{array}\right.$$

$$\underbrace{\begin{bmatrix} z_1^{[1]} \\ \vdots \\ \vdots \\ z_4^{[1]} \end{bmatrix}}_{z^{[1]} \in \mathbb{R}^{4\times 1}} = \underbrace{\begin{bmatrix} - & W_1^{[1]} & - \\ - & W_2^{[1]} & - \\ & \vdots & \\ - & W_4^{[1]} & - \end{bmatrix}}_{W^{[1]} \in \mathbb{R}^{4\times 3}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{x \in \mathbb{R}^{3\times 1}} + \underbrace{\begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \\ \vdots \\ b_4^{[1]} \end{bmatrix}}_{b^{[1]} \in \mathbb{R}^{4\times 1}}$$

$$z^{[1]} = W^{[1]}x + b^{[1]}$$

Linear

$$a^{[1]} = g(z^{[1]})$$

Non-Linear

COMPUTATION LAYER 2

$$z_1^{[2]} = W_1^{[2]} \cdot a^{[1]} + b_1^{[2]}$$ LINEAR

$\uparrow$

FROM L1

$$a_1^{[2]} = g\left(z_1^{[2]}\right)$$ ACTIVATION

$$\underset{1}{z^{[2]}} = \underset{\text{Size: } (1 \times 4)}{W^{[2]}} \cdot \underset{(4 \times 1)}{a^{[1]}} + \underset{1}{b^{[2]}}$$

Review

- Feed-Forward Neural Networks are the common neural networks architectures
  - Fully connected networks are called Multi-Layer Perceptron
- Input, output, and hidden layers
  - Linear matrix operations followed by non-linear activations at every layer
- Activations:
  - Non-linear functions
- Forward propagation: process of evaluating input through the network

Acknowledgements

- Slides made using resources from:
  - Andrew Ng
  - Eric Eaton
  - David Sontag
  - Yann LeCun
- Thanks!