Today lecture contents: (prerequisites: understanding of logistic regression)

- **NN can learn nonlinear decision boundaries**
- **Term related to neural network**
  - **Neurons**
  - **Weights**
  - **Bias**
  - **Layers**
    - **Input**
    - **Hidden**
    - **Output**
  - **No of parameters**
  - **Forward propagation**
  - **Backward propagation**
- **What is the number of networks parameters in the case of xnor NN**
- **Let's see the learning the interesting features of the different layers of Lenet5 (Hidden layers learn the interesting features)**

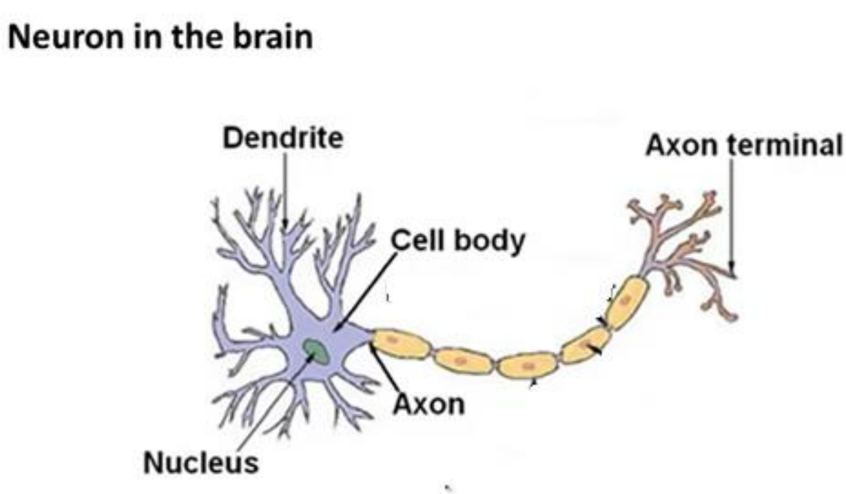
- **Handwritten Digit Classification - Yann Lecun**

https://www.youtube.com/watch?v=yxuRnBEczUU

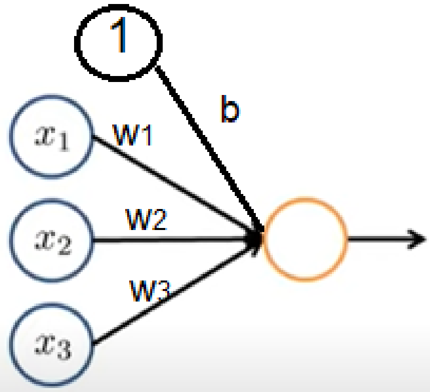
1. Model representation 1

- How do we represent neural networks (NNs)?
  - Neural networks were developed to simulate networks of neurons / human brain

Single Neuron:



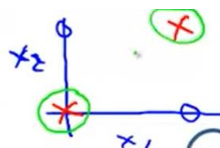
Neuron model: Logistic unit

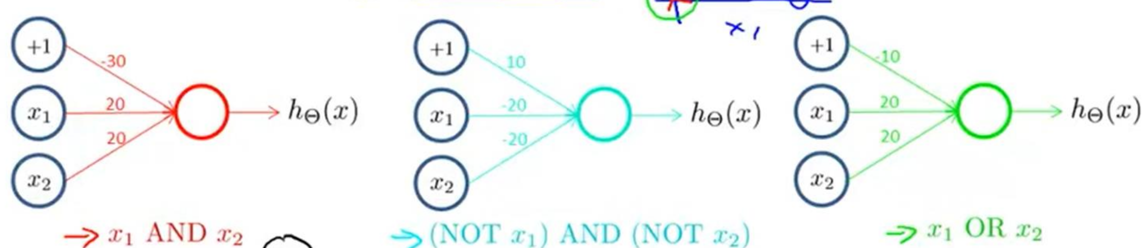


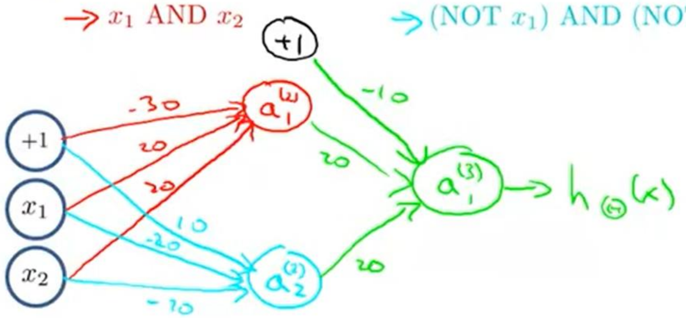
$$f(x) = \sigma(\mathrm{b} + w_1 x_1 + w_2 x_2)$$

$$f(x) = \frac{1}{1 + e^{-(\mathrm{b} + w_1 x_1 + w_2 x_2)}}$$

Putting it together: x_1 XNOR x_2





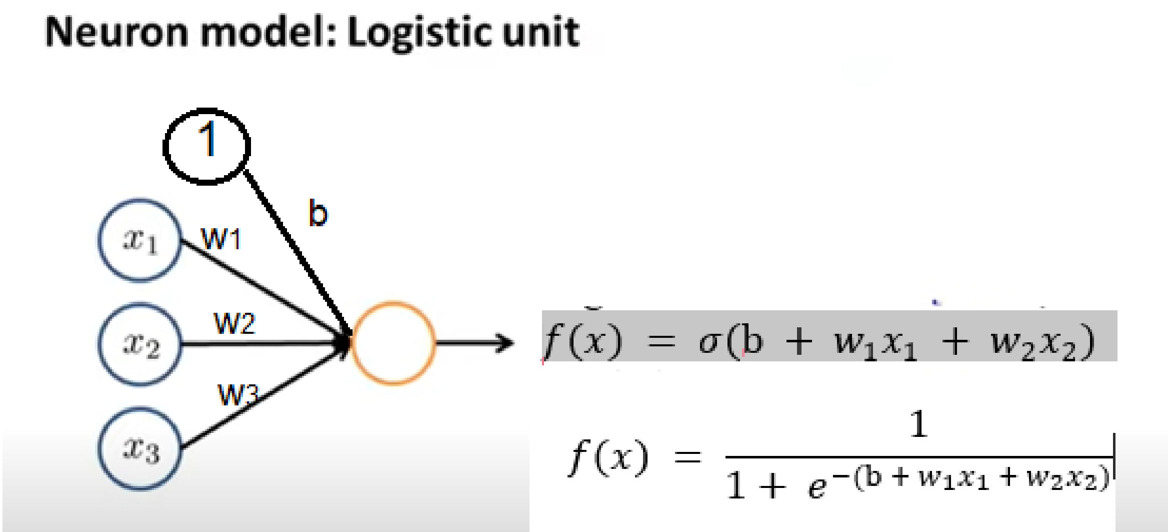


x_1	x_2	$a_1^{(2)}$	$a_2^{(2)}$	$h_\Theta(x)$
0	0	0	1	1
0	1	0	0	0
1	0	0	0	0
1	1	1	0	1

Andrew Ng

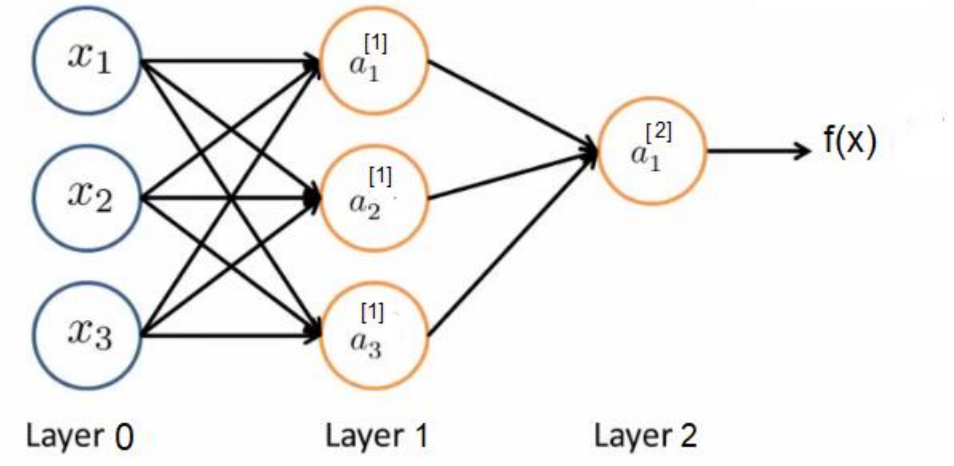
2. Artificial neural network - representation of a neuron

- TERMINOLOGY: bias, weights, layers (input, hidden, output)



- Very simple model of a neuron's computation
    - Often good to include an x_0 input - the **bias unit**
        - This is equal to 1
- This is an artificial neuron with a sigmoid (logistic) activation function
    - W vector is the **weights (later represents by b and w)** of a model

Below we have a group of neurons together



- Here, input is **x_1, x_2 and x_3** => belong to **input layer (layer 0)**

- Three neurons in **layer 1**
  - We could also call output first layer - i.e. ($a_1^{[1]}$, $a_2^{[1]}$ and $a_3^{[1]}$)

- Final fourth neuron which produces the output is **the output layers**
  - Which again we *could* call $a_1^{[2]}$

- The first layer is the **input layer**
- The final layer is the **output layer** – which produces value computed by a hypothesis
- Middle layer(s) are called the **hidden layers**

  - Parameter's matrices sizes are:

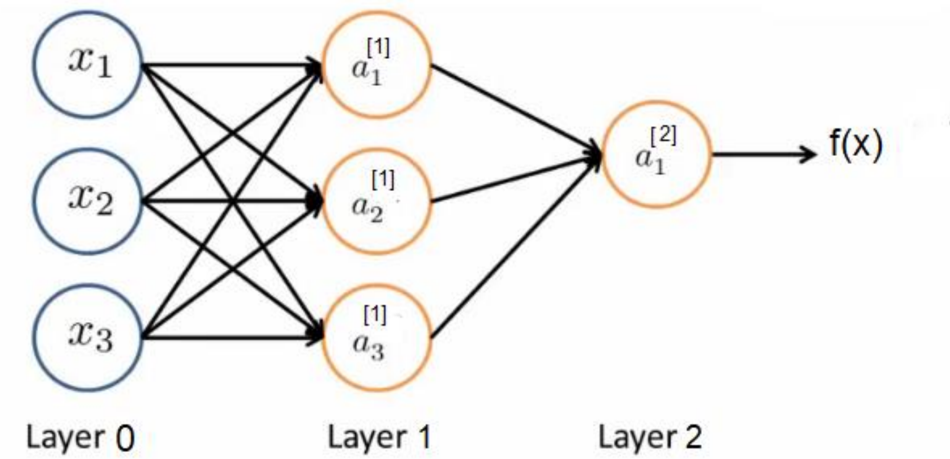
3. Neural networks – notation:

- We have an example of a network, with the **associated calculations for the four nodes below.**

$a_i^{[j]}$ - activation of unit *i* in layer *j*

- So, a_1^2 - is the **activation** of the 1st unit in the second layer
- By activation, we mean the value which is computed and output by that node

$W^{[j]}$ - matrix of parameters controlling the function mapping from layer *j* to layer *j* + 1



Suppose you train a logistic regression function/classifier, and the learned hypothesis function is

$$a_1^{[1]} = \sigma(b_1^{[1]} + w_{11}^{[1]}x_1 + w_{12}^{[1]}x_2 + w_{13}^{[1]}x_3) = \sigma(W_1^{[1]}X)$$
$$a_2^{[1]} = \sigma(b_2^{[1]} + w_{21}^{[1]}x_1 + w_{22}^{[1]}x_2 + w_{23}^{[1]}x_3) = \sigma(W_2^{[1]}X)$$
$$a_3^{[1]} = \sigma(b_3^{[1]} + w_{31}^{[1]}x_1 + w_{32}^{[1]}x_2 + w_{33}^{[1]}x_3) = \sigma(W_3^{[1]}X)$$
$$f(x) = a_1^{[2]} = \sigma(b_1^{[2]} + w_{11}^{[2]}a_1^{[1]} + w_{12}^{[2]}a_2^{[1]} + w_{13}^{[2]}a_3^{[1]}) = \sigma(W_1^{[2]}A_1^{[1]})$$

The process of calculating the f(x) is called the **forward propagation** => (calculation of the different steps from input to hidden to output activation)

Dimension of vectors and matrices. $W_1^{[1]}, W_2^{[1]}, W_3^{[1]}$, $W_1^{[2]}$, $W^{[1]}$, $W^{[2]}$, $\mathbf{b_1^{[2]}}$, $\mathbf{B^{[1]}}$

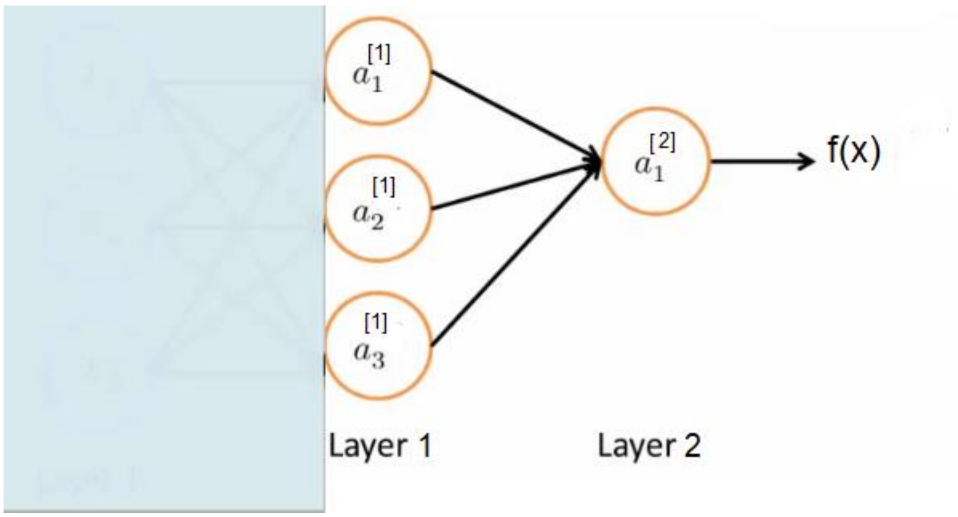
- Then $w^{[1]}$ would be = [3 x 4]
- Then $w^{[2]}$ would be = [1 x 4]

4. Quiz: what are the total number of parameters for XNOR NN?

Answer:

5. Neural networks learning its own features

- Diagram below looks a lot like logistic regression

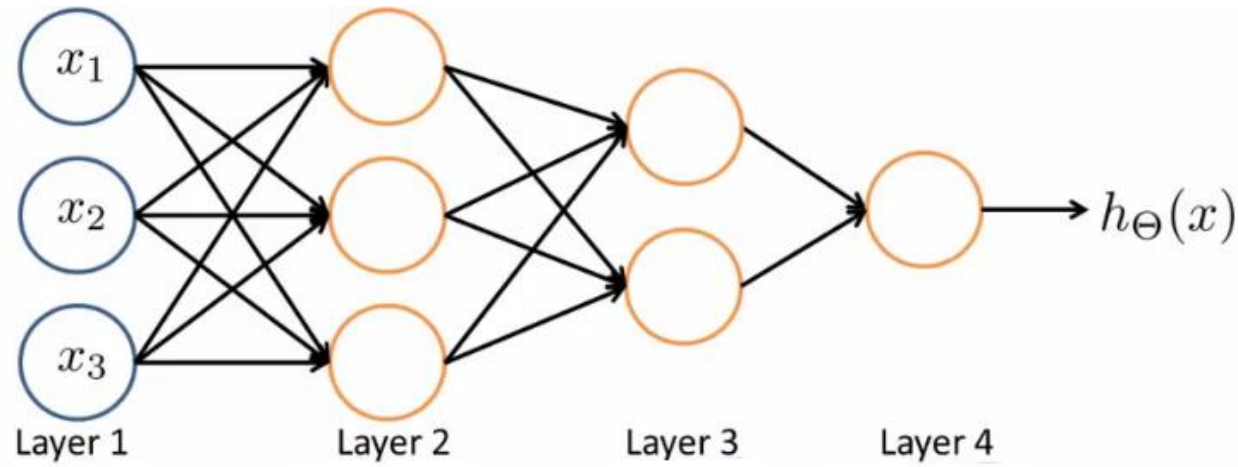


- **Layer 2 is act like a logistic regression node**
  - The function output
  - **$f(x) = g(w_{10}^{[2]} a_0^{[1]} + w_{11}^{[2]} a_1^{[1]} + w_{12}^{[2]} a_2^{[1]} + w_{13}^{[2]} a_3^{[1]} + b_1^{[2]})$**
  - This is just logistic regression
    - The only difference is, instead of input a feature vector, the features are calculated by the hidden layer
- **The features $a_1^{[2]}$, $a_2^{[2]}$, and $a_3^{[2]}$ are calculated/learned by network using $w^{[1]}$ parameters not original features**
- So the mapping from layer 0 to layer 2 (i.e. the calculations which generate the $a^{[2]}$ features) is determined by another set of parameters – $w^{[1]}$
-
  - Depending on the $w^{[1]}$ parameters you can learn some interesting things
    - Flexibility to learn whatever features it wants to feed into the final logistic regression calculation
      - So, if we compare this to previous logistic regression, you would have to calculate your own exciting features to define the best way to classify or describe something
      - Here, we're letting the hidden layers do that, so we feed the hidden layers our input values, and let them learn whatever gives the best final result to feed into the final output layer

6. Other architecture of Neural networks:

- As well as the networks already seen, other architectures (topology) are possible
  - More/less nodes per layer
  - More layers
  - Once again, layer 2 has three hidden units, layer 3 has 2 hidden units by the time you get to the output layer you get very interesting non-linear hypothesis

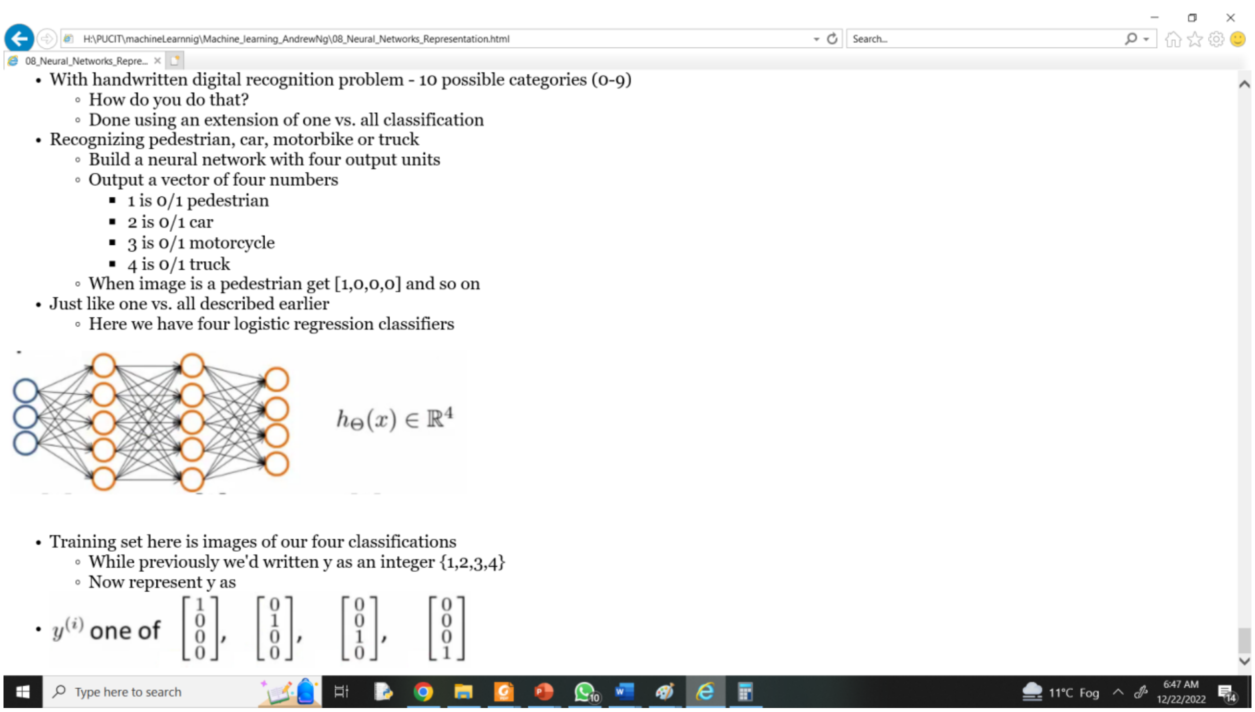
  - Once again, layer 2 has three hidden units, layer 3 has 2 hidden units by the time you get to the output layer you get very interesting non-linear hypothesis



- Some of the intuitions here are complicated and hard to understand
  - In the following lectures we're going to go through a detailed example to understand how to do non-linear analysis
  - **We will discuss the case study of different NN (AlexNet, ResNet etc) that help us how to fix the number of hyper parameter of NN**

7. Multiclass classification

- Multiclass classification is, unsurprisingly, when
  you distinguish between more than two categories (i.e. more than 1 or 0)
- With handwritten digital recognition problem - 10 possible categories (0-9)
  - How do you do that?
  - Done using an extension of one vs. all classification
- Recognizing pedestrian, car, motorbike or truck
  - Build a neural network with four output units
  - Output a vector of four numbers
    - 1 is 0/1 pedestrian
    - 2 is 0/1 car
    - 3 is 0/1 motorcycle
    - 4 is 0/1 truck
  - When image is a pedestrian get [1,0,0,0] and so on
- Just like one vs. all described earlier
  - Here we have four logistic regression classifiers



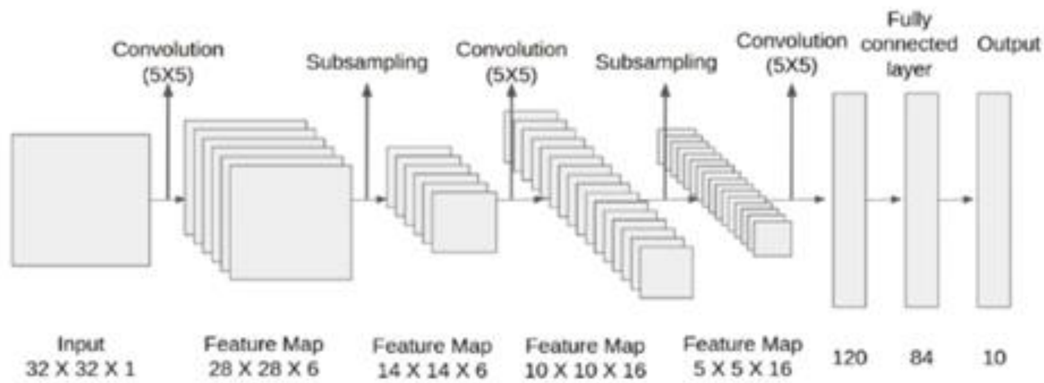
- With handwritten digital recognition problem - 10 possible categories (0-9)
  - How do you do that?
  - Done using an extension of one vs. all classification
- Recognizing pedestrian, car, motorbike or truck
  - Build a neural network with four output units
  - Output a vector of four numbers
    - 1 is 0/1 pedestrian
    - 2 is 0/1 car
    - 3 is 0/1 motorcycle
    - 4 is 0/1 truck
  - When image is a pedestrian get [1,0,0,0] and so on
- Just like one vs. all described earlier
  - Here we have four logistic regression classifiers

$h_\Theta(x) \in \mathbb{R}^4$

- Training set here is images of our four classifications
  - While previously we'd written y as an integer {1,2,3,4}
  - Now represent y as
- $y^{(i)}$ one of $\begin{bmatrix}1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$

8. Demo what the hiddent layer learns:

The Architecture of Lenet-5



- LeNet-5 CNN architecture is made up of 7 layers. The layer composition consists of 3 convolutional layers, 2 subsampling layers and 2 fully connected layers

- Let’s see the learning the interesting features by the different layers of Lenet5
- Handwritten Digit Classification - Yann Lecun https://www.youtube.com/watch?v=yxuRnBEczUU

- Demo of Yann LeCun work=> is pioneer of NN
- His early work for handwritten digit classification: see the
- Input area
- Column show visualization of first layer output features visualization
- Next layer
- Finally and what the hidden layers learns is feed next layers and ultimate to the final layer to recognize the handwritten digits.
- **See more detail about arichitecture https://www.analyticsvidhya.com/blog/2021/03/the-architecture-of-lenet-5/**

Quiz, activation function, NN paper results, why need CNN, lesser no of parameters and due to weight sharing.