NEURAL NETWORKS

Lara J. Martin (she/they)
TA: Aydin Ayanzadeh (he)

12/7/2023

CMSC 671

By the end of class today, you will be able to:

1. Perform feed-forward propagation for a given simple neural network
2. Describe how feed-forward networks can be used for classification
3. Extract features & follow backpropagation

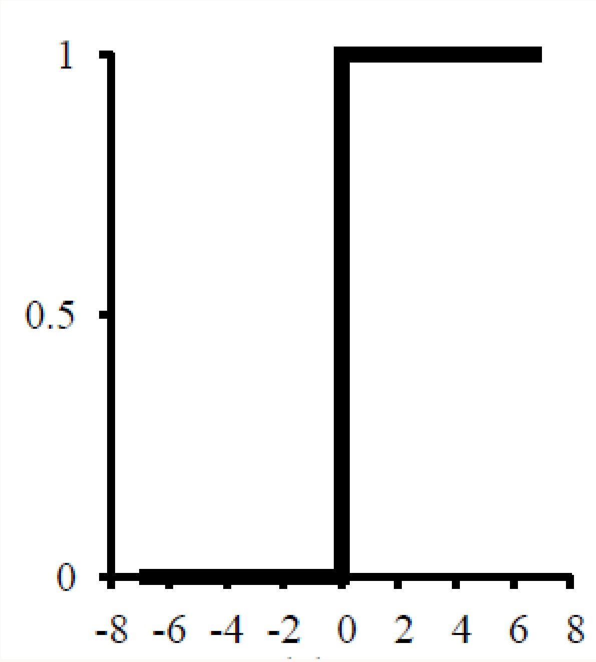
Modified from slides by Dr. Cassandra Kent & Dr. Chris Callison-Burch

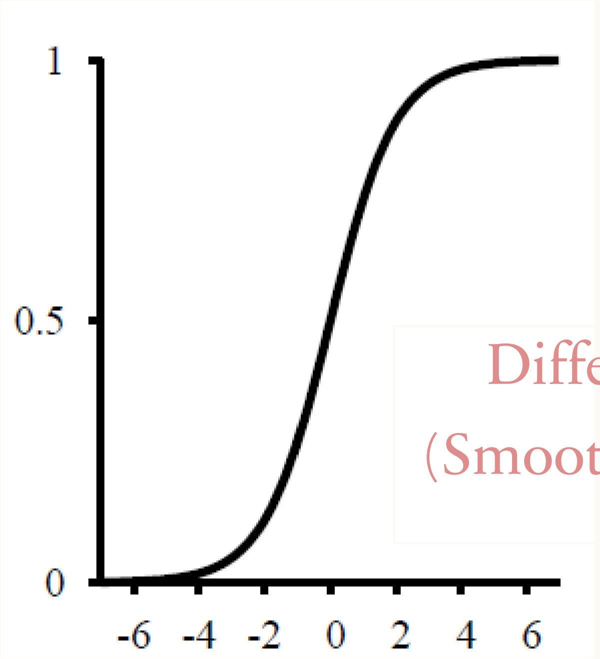
RECAP

Q: What types of problems do perceptrons struggle with?

A: Non-linearly separable problems

REVIEW: NEURON UNIT ACTIVATION FUNCTIONS





Differentiable!
(Smooth derivative)

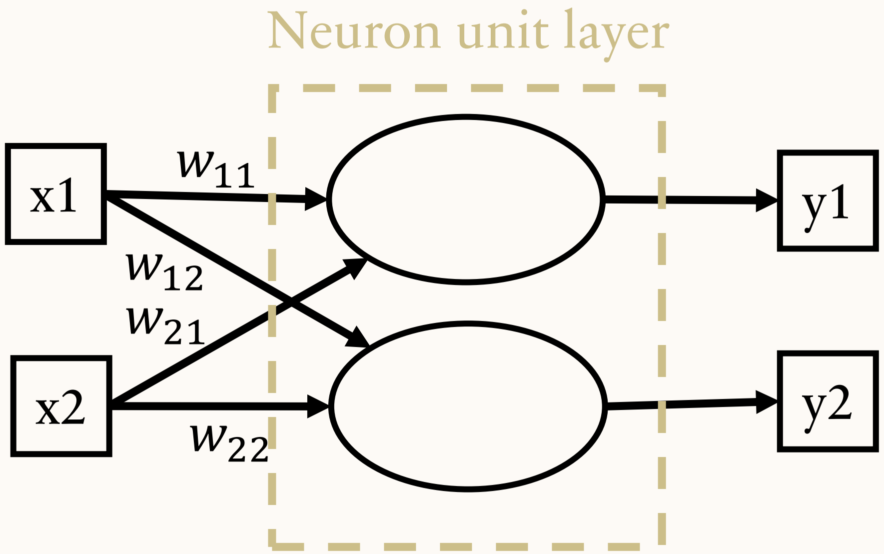
Step function (hard threshold):

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

Sigmoid function (soft threshold):

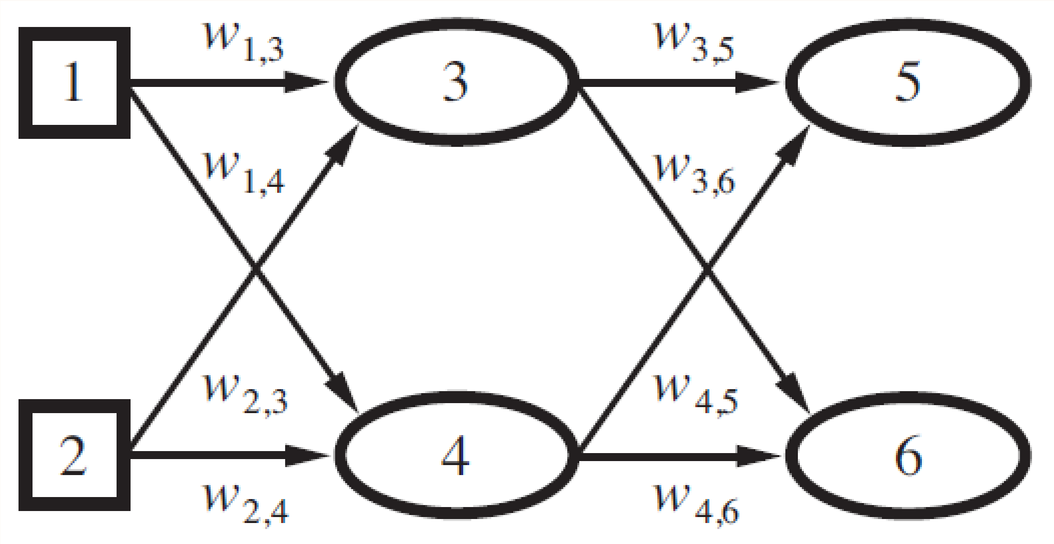
$$g(x) = \frac{1}{1 + e^{-x}}$$

WHAT HAPPENS WHEN WE HAVE MORE THAN ONE HIDDEN LAYER?



MULTI-LAYER NETWORKS: GENERAL STRUCTURE

Mutli-layer perceptrons (aka neural networks) will have **inputs**, one or more **hidden layers**, and an **output layer:**

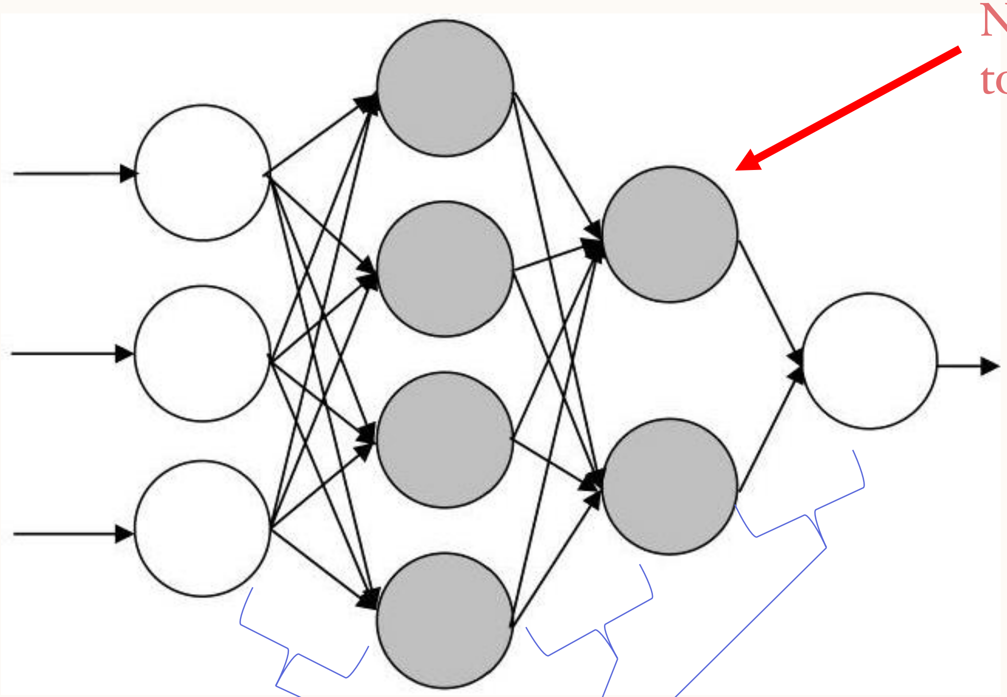


MULTI-LAYER NETWORKS: GENERAL STRUCTURE

Mutli-layer perceptrons (aka neural networks) will have **inputs**, one or more **hidden layers**, and an **output layer:**

- Number of inputs, outputs, and number and size of hidden layers can vary
- Combination of **different weights** and **different structures** represent different **functions**
- We will treat each layer as **fully-connected**
  - Each unit in one layer connects to every unit in the next layer

MULTI-LAYER NETWORKS: GENERAL STRUCTURE EXAMPLE

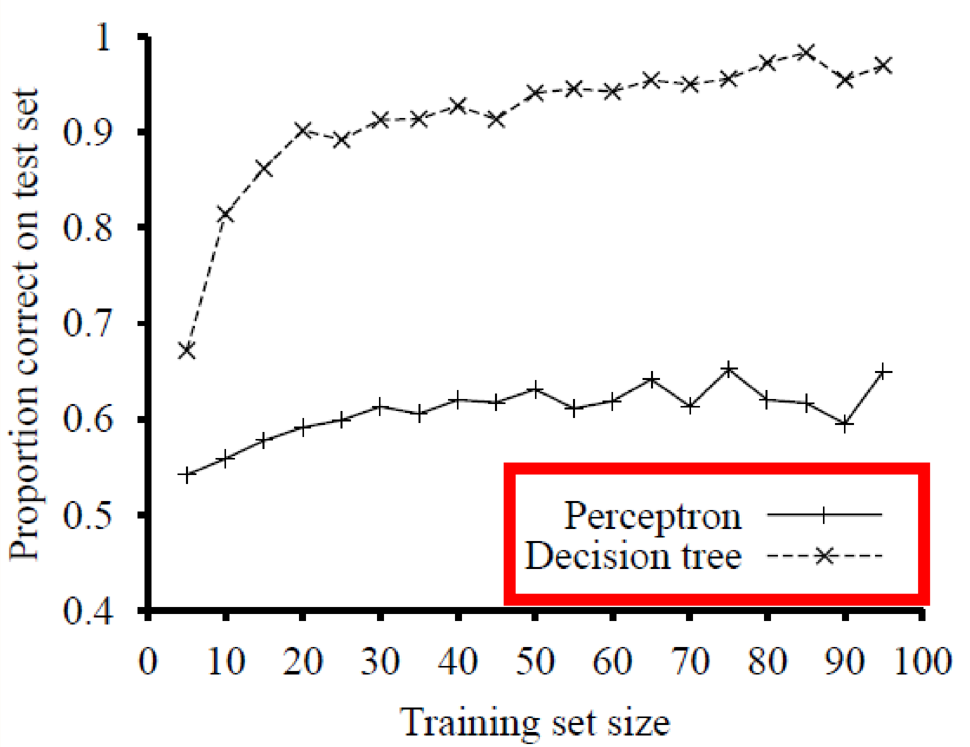


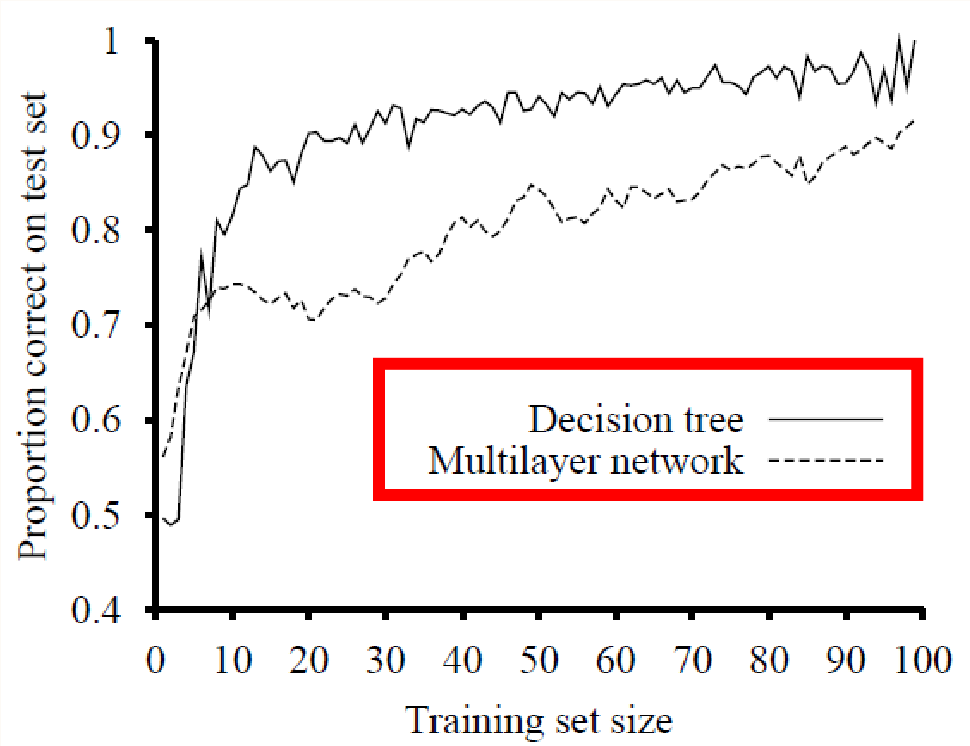
Note that the layers don’t have to be the same size

Fully connected

MULTI-LAYER NETWORKS

- Mutli-layer neural networks can effectively classify data that’s not linearly separable.
- Example: restaurant task





COMPUTING VALUES: FORWARD PROPAGATION

Forward propagation calculates the output values for a given set of input values

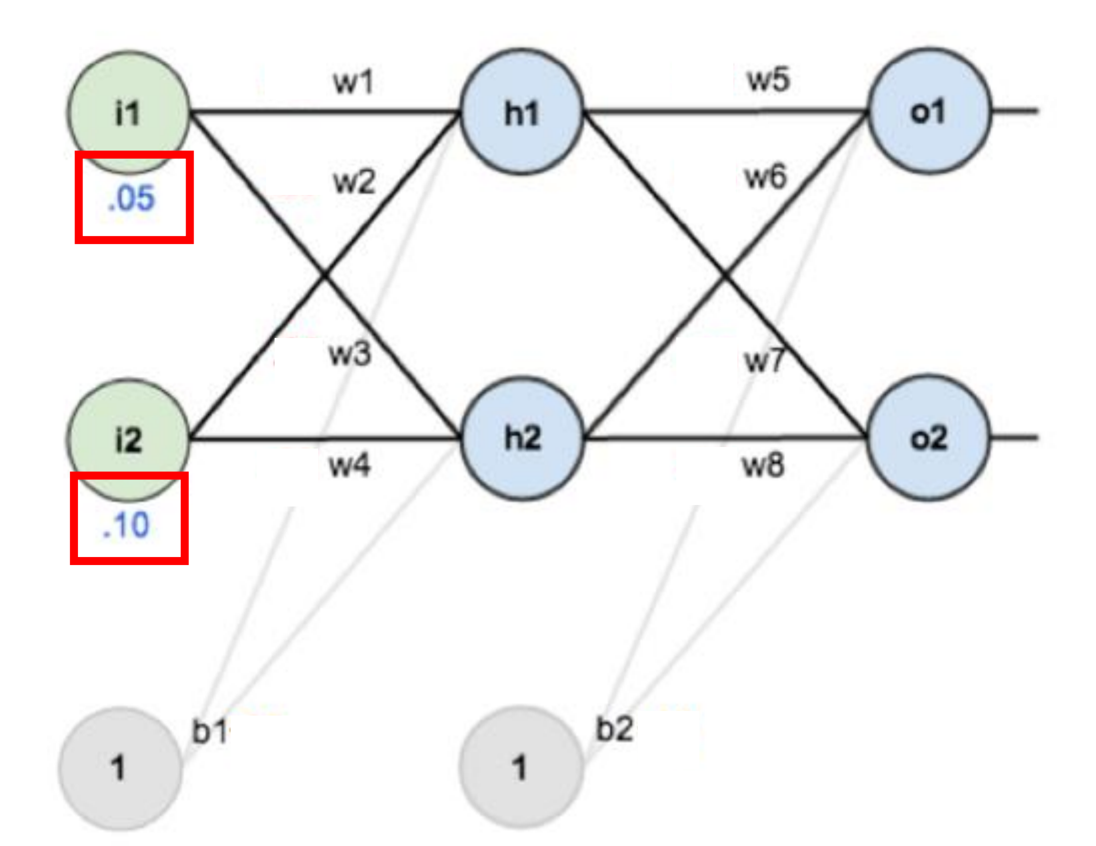
Algorithm

For each layer:

1. Calculate the weighted sum of inputs to each neuron unit
2. Evaluate the activation function to determine the output of each neuron unit
3. Use outputs as inputs for the next layer

FORWARD PROPAGATION EXAMPLE

- Calculate the output of the network below, assuming each neuron uses a sigmoid activation function, given 0.05 and 0.1 as inputs.

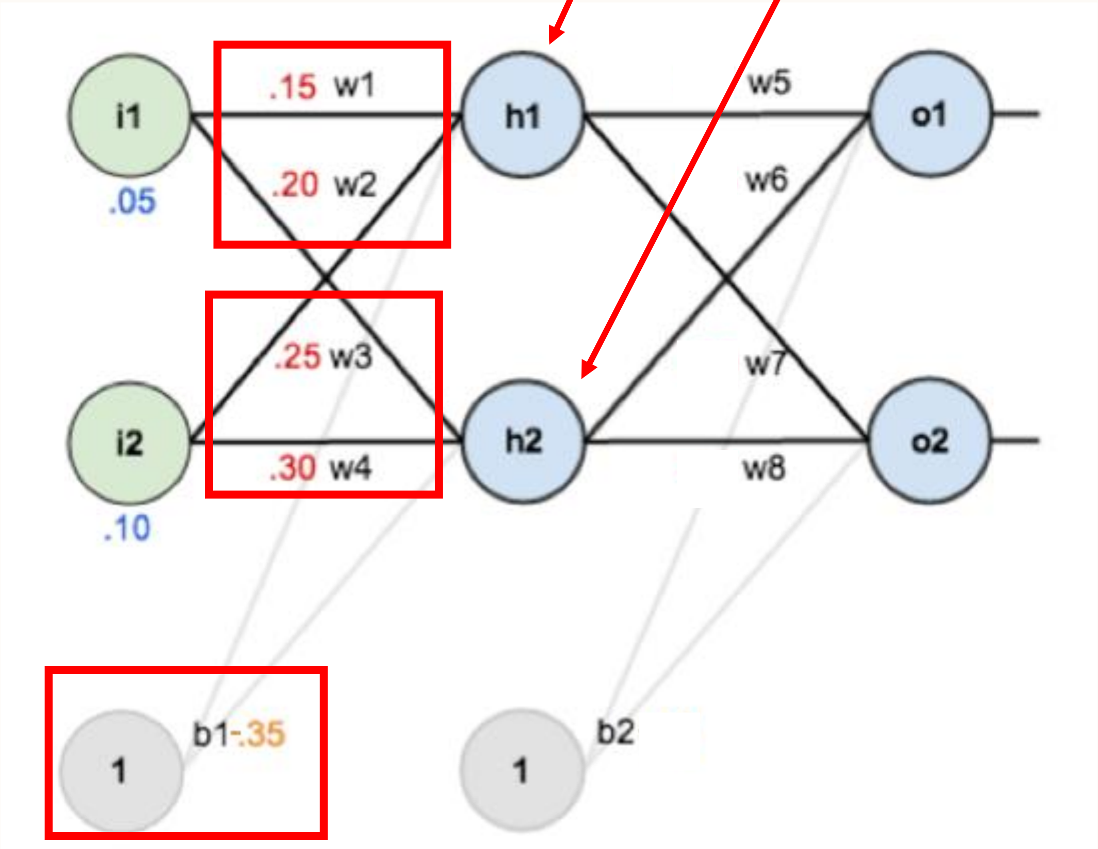


For each layer:

1. Calculate the weighted sum of inputs to each neuron unit
2. Evaluate the activation function to determine the output of each neuron unit
3. Use outputs as inputs for the next layer

FORWARD PROPAGATION EXAMPLE

- Calculate inputs to the hidden layer (units h1 and h2):



$$
\begin{aligned}
in_{h1} &= w_1 i_1 + w_2 i_2 + b_1 \\
&= .15(.05)+.2(.1)-.35 \\
&= .0075+.02-.35 \\
&= -.3225
\end{aligned}
$$

$$
\begin{aligned}
in_{h2} &= w_3 i_1 + w_4 i_2 + b_2 \\
&= .25(.05)+.3(.1)-.35 \\
&= .0125+.03-.35 \\
&= -.3075
\end{aligned}
$$

For each layer:

1. Calculate the weighted sum of inputs to each neuron unit
2. Evaluate the activation function to determine the output of each neuron unit
3. Use outputs as inputs for the next layer

FORWARD PROPAGATION EXAMPLE

- Calculate <u>outputs</u> to the hidden layer (units h1 and h2):

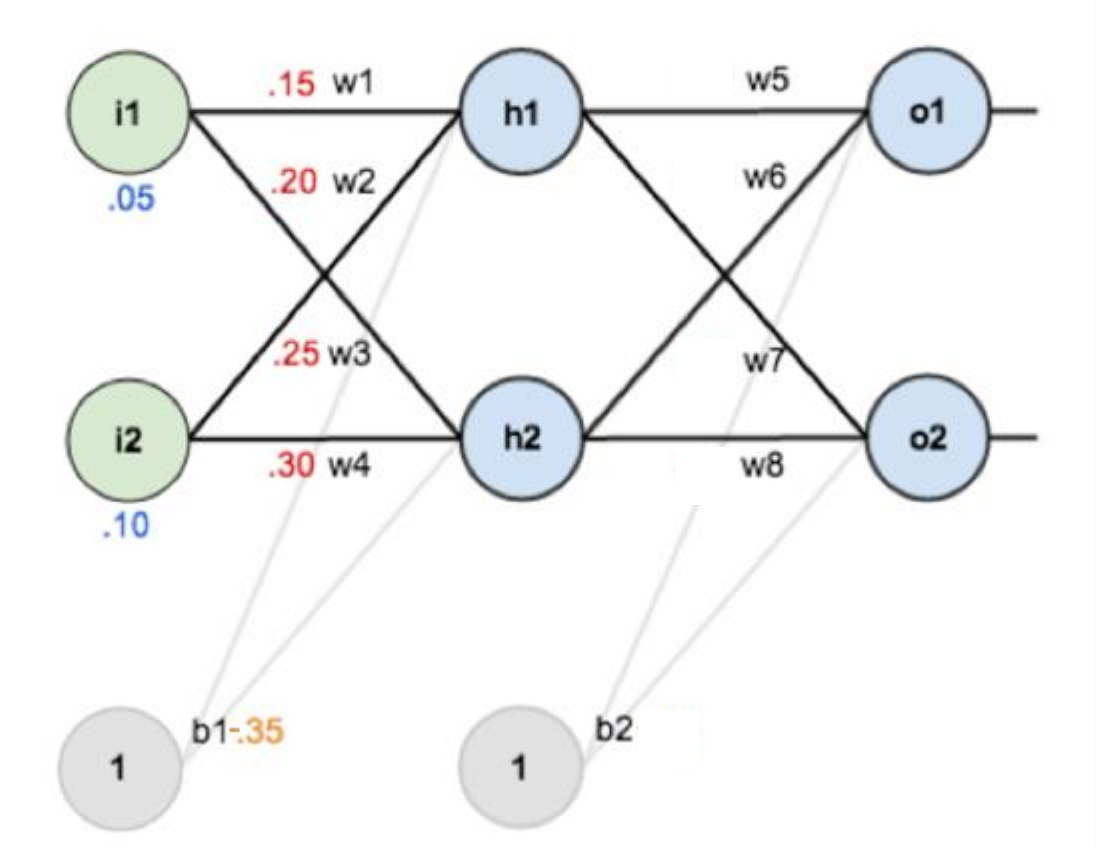
How do we do this?
Use our activation function!

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

What will be our *x*?

$\text{in}_{h1} = -.3225$
$\text{in}_{h2} = -.3075$

For each layer:
1. Calculate the weighted sum of inputs to each neuron unit
2. Evaluate the activation function to determine the output of each neuron unit
3. Use outputs as inputs for the next layer



$$\text{out}_{h1} = g(\text{in}_{h1}) = \frac{1}{1+e^{-in_{h1}}} = \frac{1}{1+e^{-(-.3275)}} = .4188$$

$$\text{out}_{h2} = g(\text{in}_{h2}) = \frac{1}{1+e^{-in_{h2}}} = \frac{1}{1+e^{-(-.3075)}} = .4237$$

YOUR TURN: FORWARD PROPAGATION EXAMPLE

1. What would the input and output values be for the **output layer**?
2. Which nodes would activate, if any?

$out_{h1} = .4188$

$out_{h2} = .4237$

Activation function:

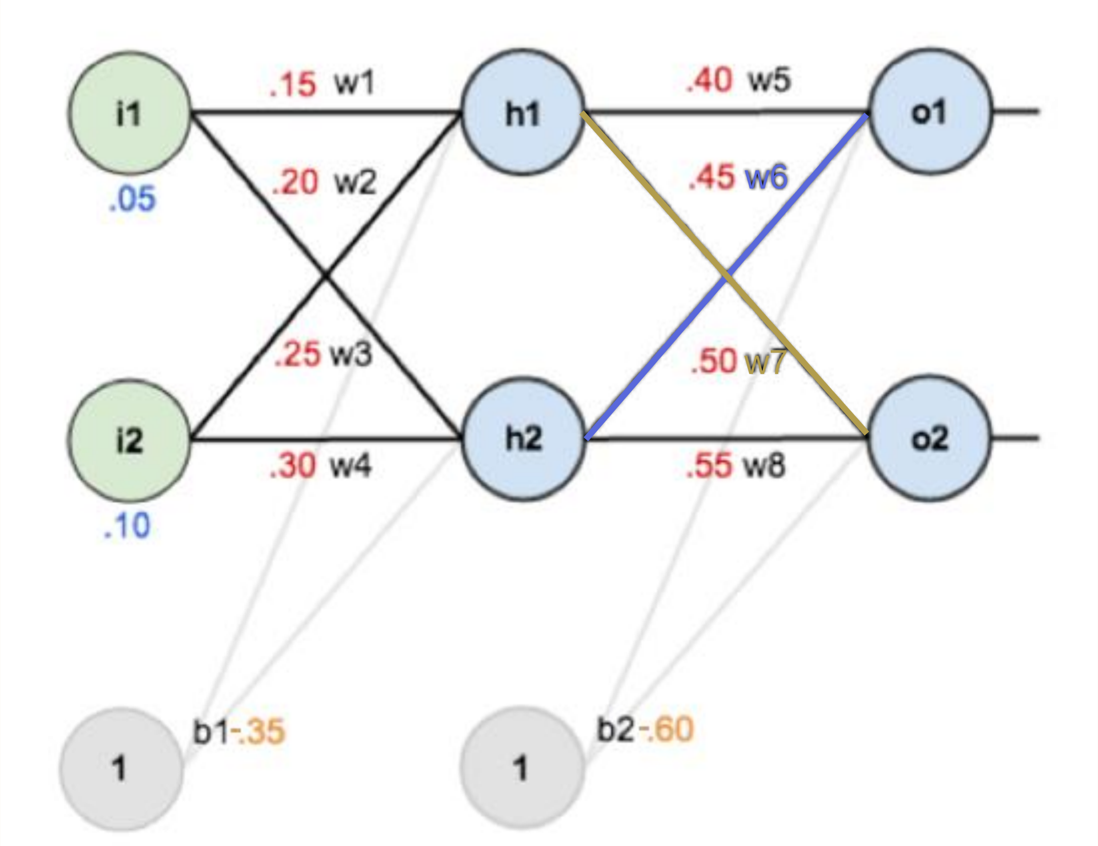
$$g(x) = \frac{1}{1 + e^{-x}}$$

Decision boundary:

$$\hat{y} = \begin{cases} 1, if\ P(y|x) > 0.5 \\ 0, otherwise \end{cases}$$

For each layer:

1. Calculate the weighted sum of inputs to each neuron unit
2. Evaluate the activation function to determine the output of each neuron unit
3. Use outputs as inputs for the next layer



HOW ARE NEURAL NETWORKS USED?

- Are neural networks supervised or unsupervised learning?
  - Inputs to the network are features of our data set
  - Outputs to the network are our labels

- Can they be used for classification or regression?
  - Either!

EXAMPLE – IMAGE CLASSIFICATION

Example:

- Classifying images of dogs and muffins
  (it's harder than you might think)

EXAMPLE – IMAGE CLASSIFICATION

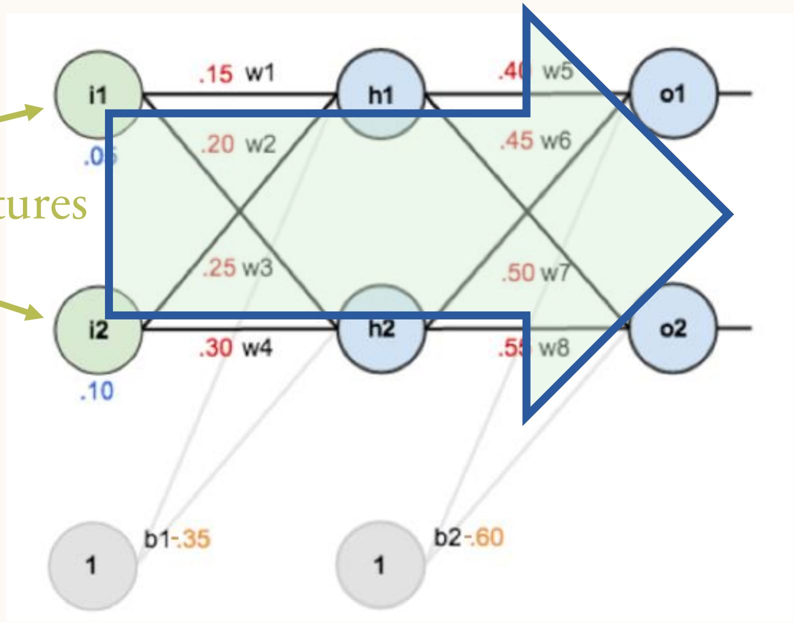
- What does are training data look like?

Input	Output1 (dog likelihood)	Output2 (muffin likelihood)
	.99	.01
	.05	.95

TRAINING

- For each training example, take a network with some **initial weights**, and use forward propagation to see what outputs we get:

Extract features



$.24 \neq .01$

$.62 \neq .99$

- Calculate error
- Then learn new weights for the network so that we get the outputs that we expect!

EXAMPLE – SENTIMENT CLASSIFICATION

Identify whether a given piece of text (like a review) is positive or negative:

Input: "Spiraling away from narrative control as its first three episodes unreel, this series, about a post-apocalyptic future in which nearly everyone is blind, wastes the time of Jason Momoa and Alfre Woodard, among others, on a story that starts from a position of fun, giddy strangeness and drags itself forward at a lugubrious pace."

Output: positive (1) or negative (0)

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Variable	Definition	Value
x_1	Count of positive lexicon words	
x_2	Count of negative lexicon words	
x_3	Does “no” appear? (binary feature)	
x_4	Number of 1st and 2nd person pronouns	
x_5	Does ! Appear? (binary feature)	
x_6	Log of the word count for the document	

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Variable	Definition	Value
x_1	Count of positive lexicon words	3
x_2	Count of negative lexicon words	
x_3	Does “no” appear? (binary feature)	
x_4	Number of 1st and 2nd person pronouns	
x_5	Does ! Appear? (binary feature)	
x_6	Log of the word count for the document	

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Variable	Definition	Value
x_1	Count of positive lexicon words	3
x_2	Count of negative lexicon words	2
x_3	Does “no” appear? (binary feature)	
x_4	Number of 1st and 2nd person pronouns	
x_5	Does ! Appear? (binary feature)	
x_6	Log of the word count for the document	

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Variable	Definition	Value
x_1	Count of positive lexicon words	3
x_2	Count of negative lexicon words	2
x_3	Does "no" appear? (binary feature)	1
x_4	Number of 1st and 2nd person pronouns	
x_5	Does ! Appear? (binary feature)	
x_6	Log of the word count for the document	

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Variable	Definition	Value
x_1	Count of positive lexicon words	3
x_2	Count of negative lexicon words	2
x_3	Does “no” appear? (binary feature)	1
x_4	Number of 1st and 2nd person pronouns	3
x_5	Does ! Appear? (binary feature)	
x_6	Log of the word count for the document	

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Variable	Definition	Value
x_1	Count of positive lexicon words	3
x_2	Count of negative lexicon words	2
x_3	Does “no” appear? (binary feature)	1
x_4	Number of 1st and 2nd person pronouns	3
x_5	Does ! Appear? (binary feature)	0
x_6	Log of the word count for the document	

EXTRACTING FEATURES

It's hokey. There are virtually no surprises , and the writing is second-rate . So why was it so enjoyable? For one thing , the cast is great . Another nice touch is the music . I was overcome with the urge to get off the couch and start dancing . It sucked me in , and it'll do the same to you .

Word count = 64, ln(64) = 4.15

Variable	Definition	Value
x_1	Count of positive lexicon words	3
x_2	Count of negative lexicon words	2
x_3	Does "no" appear? (binary feature)	1
x_4	Number of 1st and 2nd person pronouns	3
x_5	Does ! Appear? (binary feature)	0
x_6	Log of the word count for the document	4.15

CALCULATING "IN"

$$in = \sum_i w_i x_i + b$$

$$in = 0.805$$
$$\sigma(0.805) = 0.69$$

Variable	Definition	Value	Weight	Product
x_1	Count of positive lexicon words	3	2.5	7.5
x_2	Count of negative lexicon words	2	-5	-10
x_3	Does "no" appear? (binary feature)	1	-1.2	-1.2
x_4	Number of 1st and 2nd person pronouns	3	.5	1.5
x_5	Does ! Appear? (binary feature)	0	2	0
x_6	Log of the word count for the document	4.15	.7	2.905
b	Bias	1	.1	.1

https://playground.tensorflow.org/