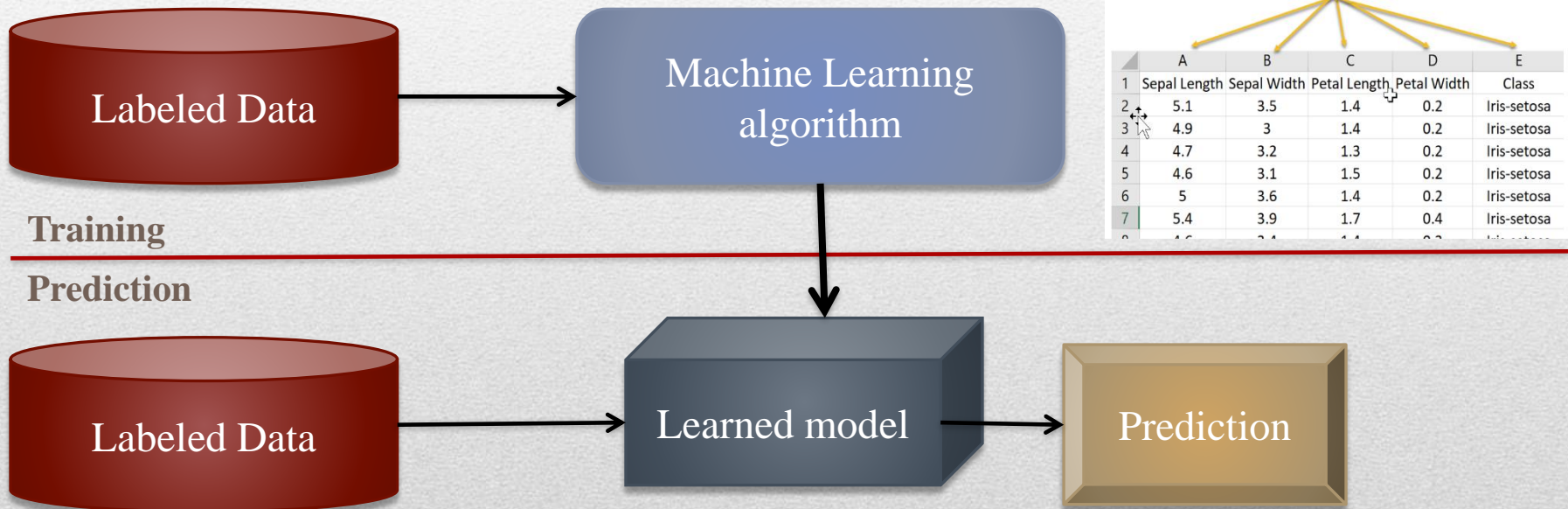


Introduction to Deep Learning

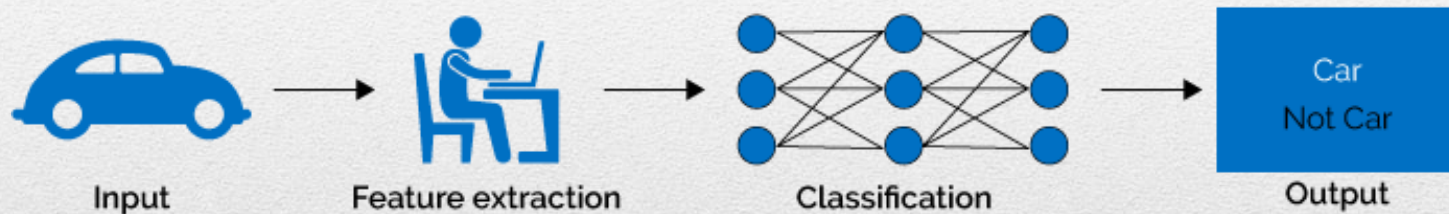
Machine Learning Basics

Machine learning is a field of computer science that gives computers the ability to **learn without being explicitly programmed**

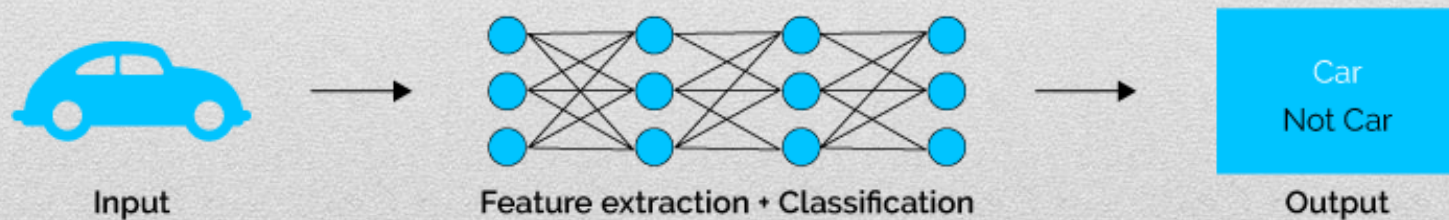


ML vs DL

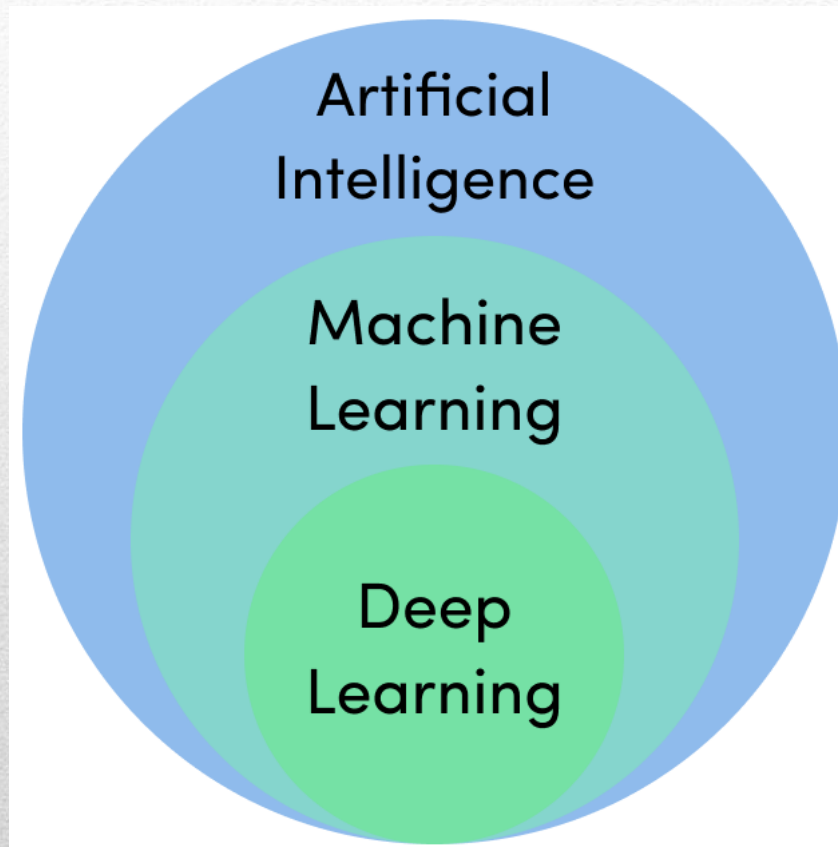
Machine Learning



Deep Learning

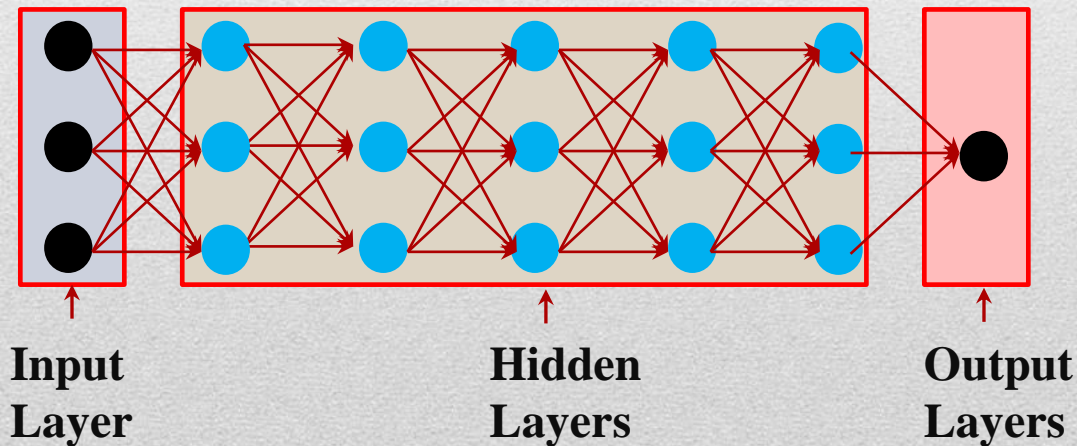


ML vs DL

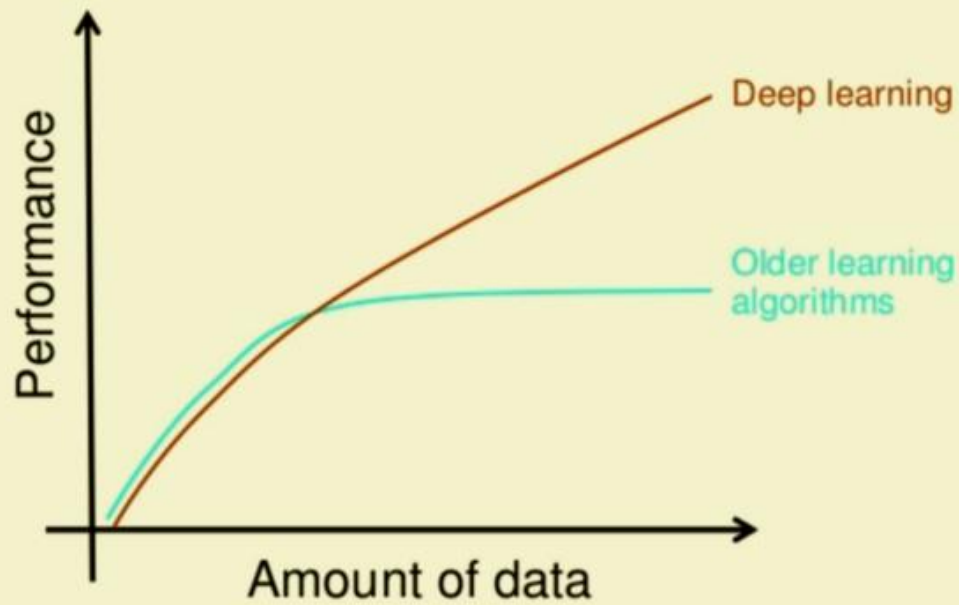


- What exactly is deep learning (DL) ?
- Why is DL almost better than other methods in terms of image, videos, test analysis?

- DL is a normal neural network **BUT with** several layers of nodes **between** input and output
- These layers do **extracting/ processing** the features in a series of stages.



Why deep learning



How do data science techniques scale with amount of data?

Modern Hardware (GPUs)/ Cloud

Data :Social media

<https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063>

Deep Knowledge

How Do Neural Networks Work?

Weight	Height	Class
100	180	Man
70	190	Man
50	160	Woman

$$y = \sum_{i=1}^n w_i \cdot x_i + b_i$$

Activation functions

$$h_1 = \sigma(W_1 \text{weight} + W_2 \text{height} + b_1)$$

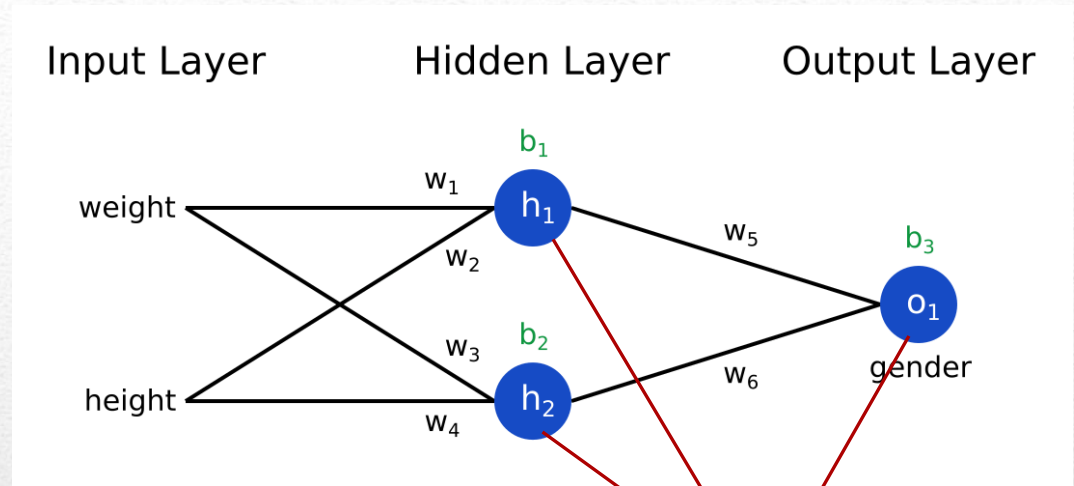
$$h_2 = \sigma(W_3 \text{weight} + W_4 \text{height} + b_2)$$

$$O_1 = \sigma(W_5 h_1 + W_6 h_2 + b_3)$$

$$[2 \times 2] + [2 \times 1] = 6 \text{ weights}$$

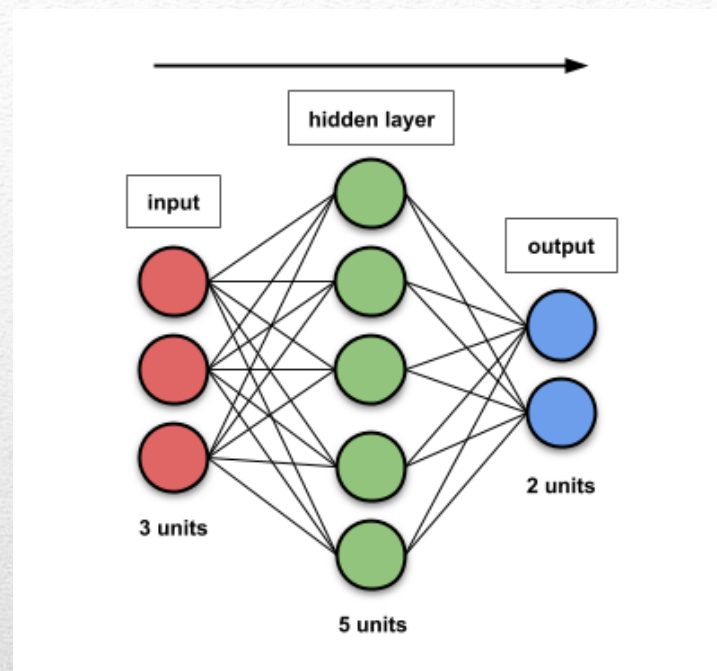
$$2 + 1 = 3 \text{ biases}$$

9 learnable **parameters**



3 neurons (not counting inputs)

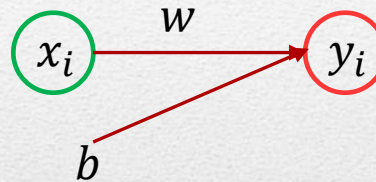
How Do Neural Networks Work?



32 learnable **parameters**

How Do Neural Networks Work?

X	Y
1	2
3	4
5	6
7	8
8	9
	?



$$b = 1$$

$$\hat{y} = x_i * w + b$$

$$\hat{y} = 8 * .5 + 1 = 5$$



$$\hat{y} = 8 * 0.7 + 1 = 8.2$$



$$\hat{y} = 8 * 1 + 1 = 9$$



Deep Learning Platforms



Training the neural network

Fields *class*

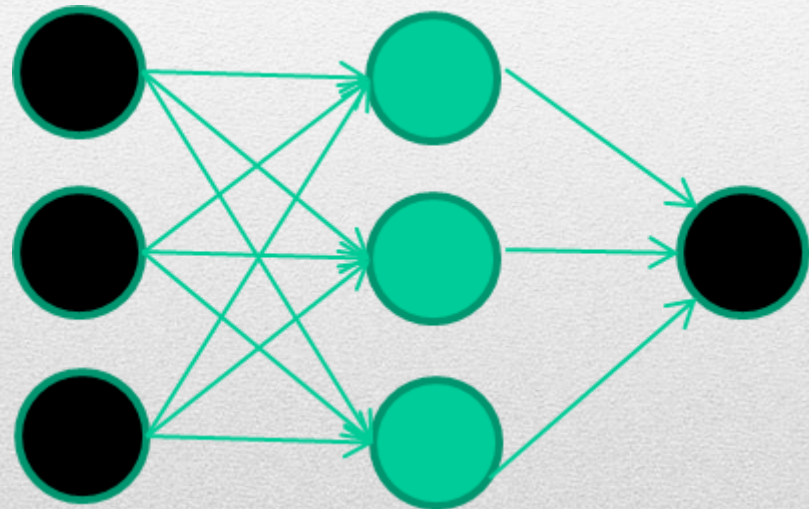
1.4 2.7 1.9 0

3.8 3.4 3.2 0

6.4 2.8 1.7 1

4.1 0.1 0.2 0

etc ...



Training data

Fields *class*

1.4 2.7 1.9 0

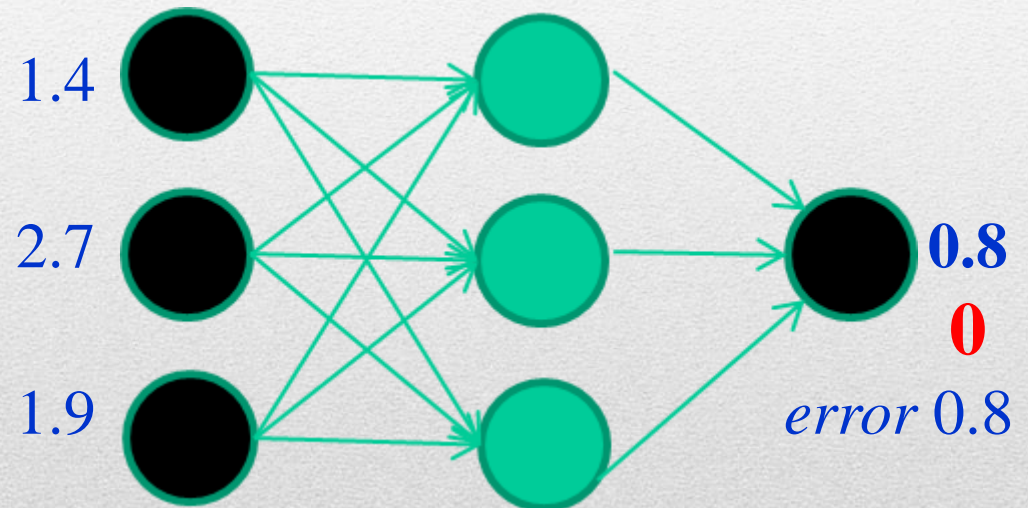
3.8 3.4 3.2 0

6.4 2.8 1.7 1

4.1 0.1 0.2 0

etc ...

Compare with target output



Training data

Fields *class*

1.4 2.7 1.9 0

3.8 3.4 3.2 0

6.4 2.8 1.7 1

4.1 0.1 0.2 0

etc ...

Compare with target output

