ANN - part 2

NN is not used for numerical problems
↳ used for complex problems like image classification

CASE STUDY

Video (1 sec footage) → FPS (Frames Per Second) → set of Images

↳ static image analysis

↓

collection of pixels (in computer lingo)

image analysis ↳ behind the scenes, numerical analysis

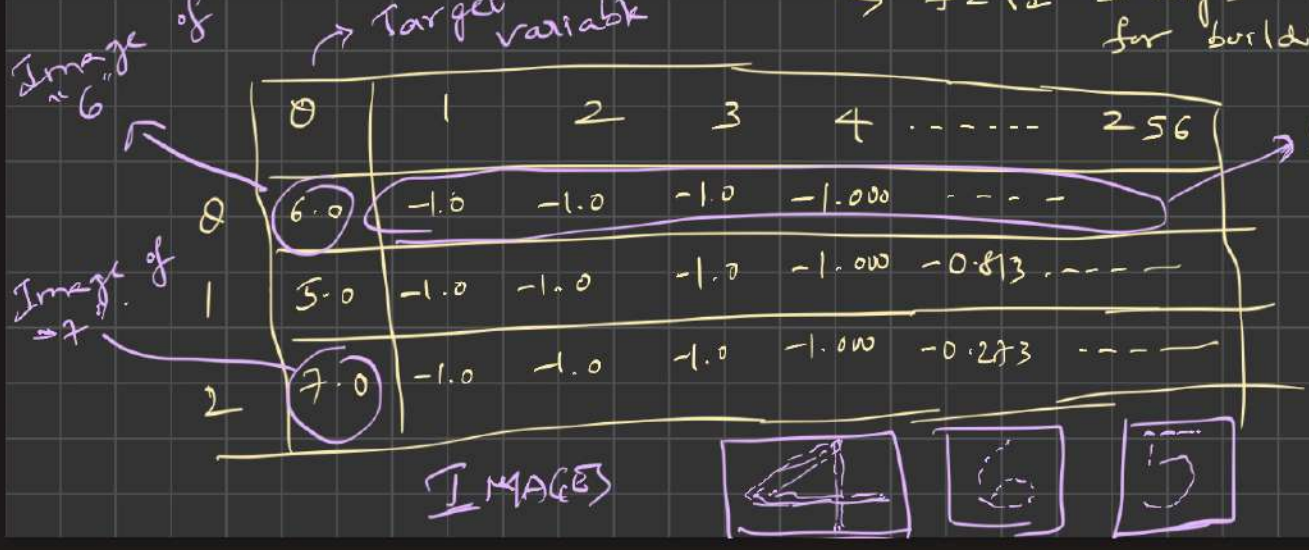
ex: cat image analysis ↳ we are analysing pixel values of the cat image.

ex:
Pixel values
RGB (255, 0, 0)
(0, 0, 255)
(3560, 6512, 0)

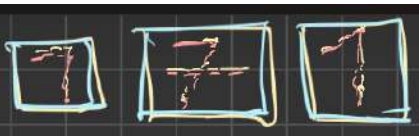
Analyse

ex: train_data.shape (7291, 257) → 256 pixel values

↳ 7291 Images are considered for building a model



Each row indicates image of the number

ex: multiple Images of '7' → 

What is the problem statement?

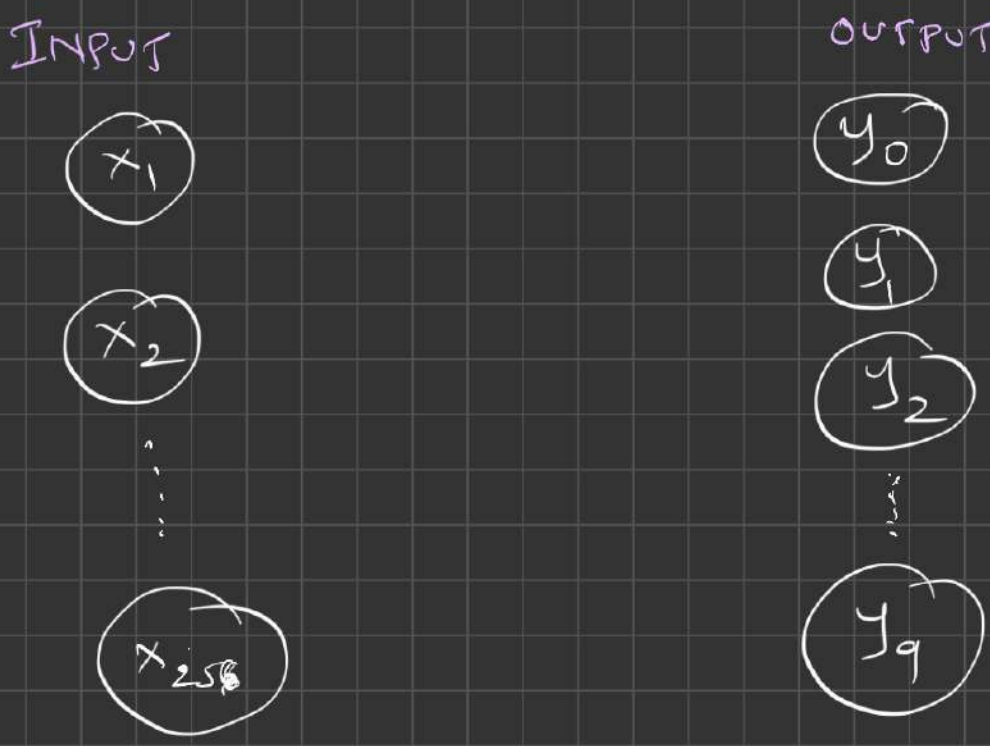
→ We need to predict the numbers on the car plate

TN2576 | AP3762 | MH2819

Identify input variables → 1 to 256 (pixels)

Target variable → 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

↳ Multi class classification



step ①: prepare list of [(min, max)] values for input

step ②: One column [0, 1, 2, 3, 4, 5, 6, 7, 8, 9] should be converted into 10 binary columns

↙

dummy variables

(or) One hot encoding

ex:

y	y_0	y_1	y_2	y_3	y_4	...	y_9
0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0
9	0	0	0	0	0	0	1

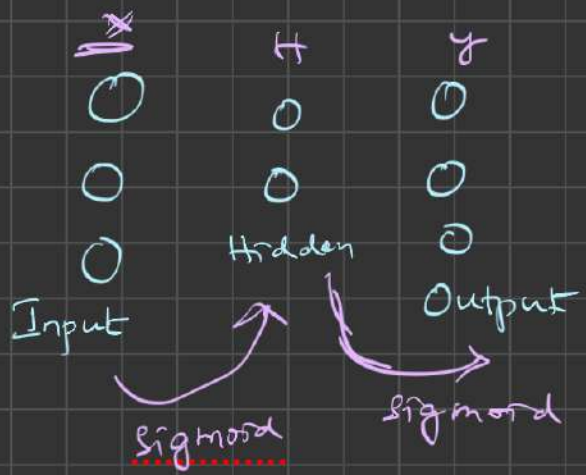
Step ③ : Build Neural Network – Configure Neural Net

net = nl.net.newff(....)

Sigmoid function → logistic function

$$\frac{e^x}{1+e^x}$$

we assume 20 hidden nodes



net.trainf = nl.train.train_rprop → train these models using back propogation

Supply data

net.train(x_train, Y_train, show=0, epochs=300)

show → 0 → final output
1 → for each iteration

epochs → FF → Error Calc → BP → Update weights → one epoch

Explain weights or weight parameters?

In an ANN, the **learned parameters** are:

- Weights
- Biases

same { weights (or) weight parameters

When someone says, "Model parameters"

↓

Parameters = Weights + biases

Note:

ANN learns weights automatically

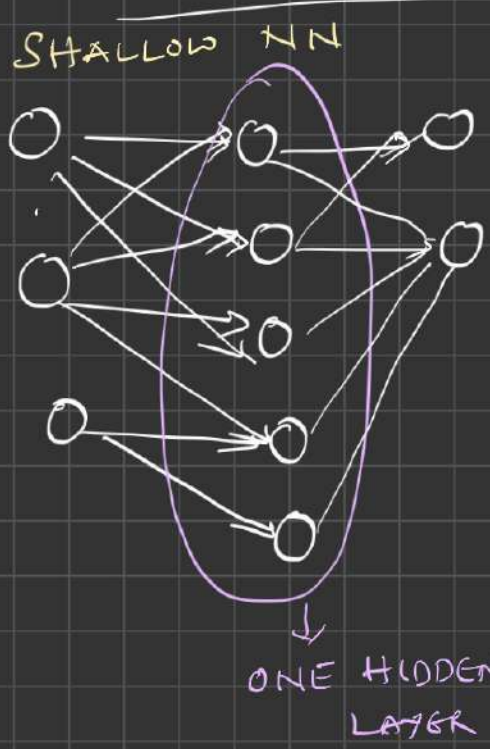
- Back propogation adjusts weights based on mistakes.

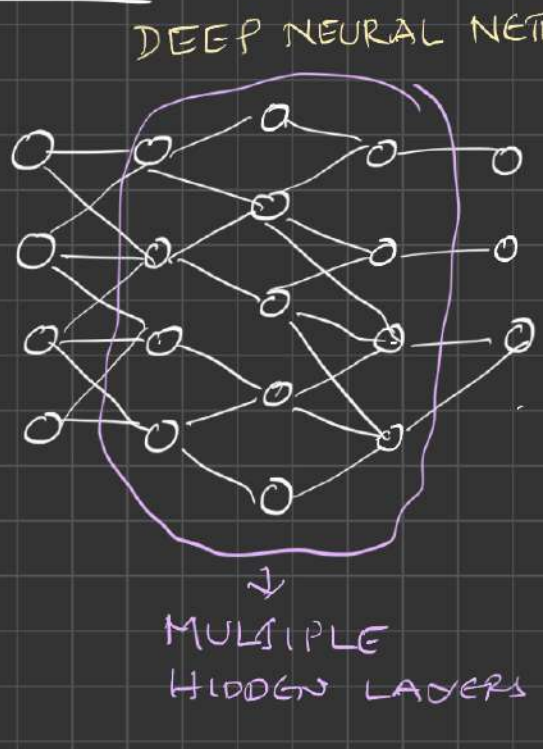
In the above number image example,

Weights are learned importance scores that tell the network which pixels matter, how much they matter, and in what direction for a given classification.

↓

Deep Vs Shallow Neural Networks





NN with one hidden layer → shallow NN

NN with multiple hidden layer → Deep NN

Logistic Reg → Shallow NN → Deep NN

(powerful) (very powerful) (1000 times powerful) than shallow NN → unreasonable accuracy

Package that helps in solving Deep learning models

- Tensorflow → Keras } by google
- pytorch } facebook

→ Python will access TF through an API

All LLM → will have very very deep NN with billions / trillions of parameters.