

Neural Network

NN refers to a method that teaches to process data in a way that is inspired by Human brain

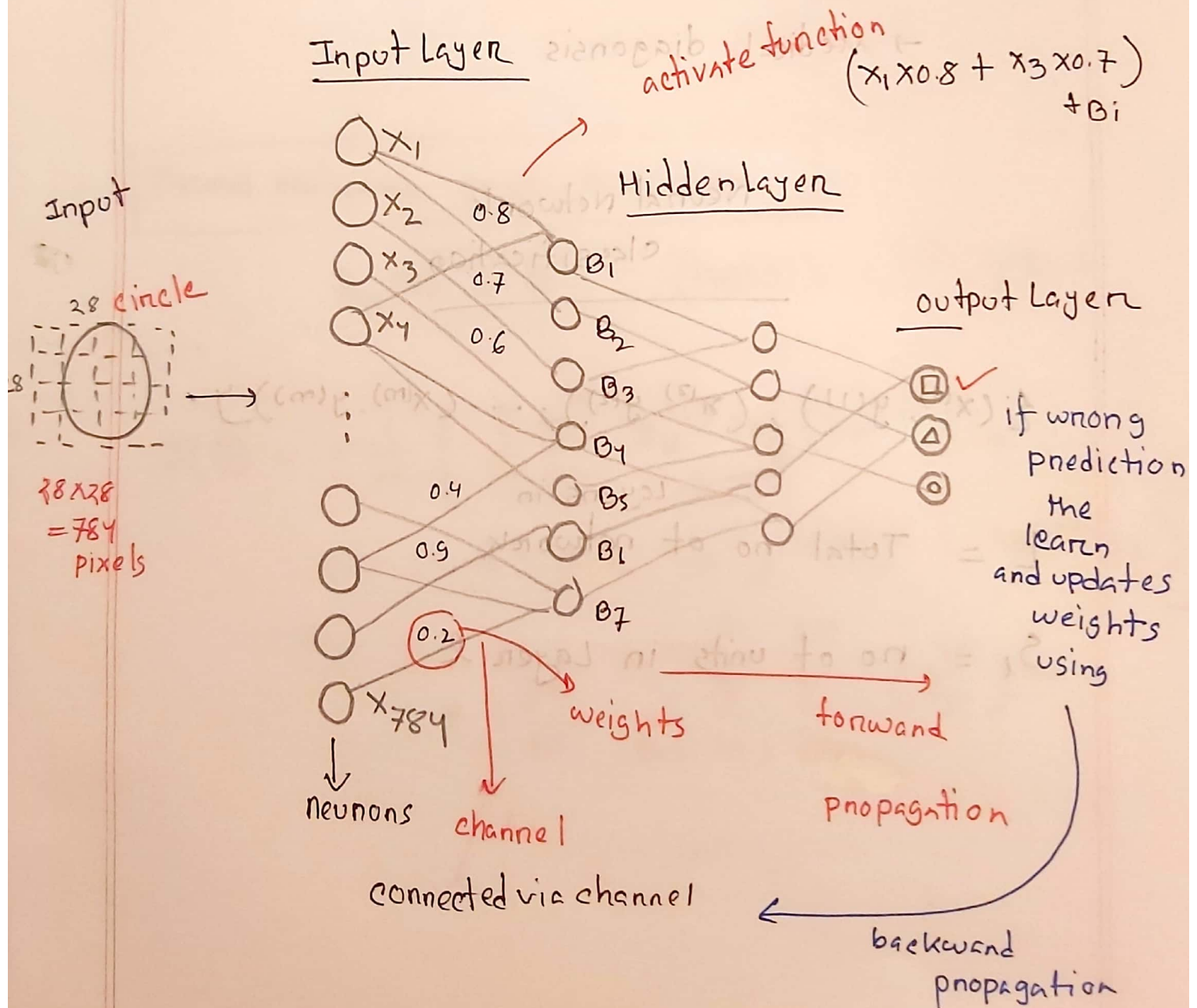
→ a neural network consists of interconnected processing unit called neuron or nodes, organized in Layer. These Layer include, one input layer, one or more hidden layer one output layer. Each connection between neurons has an associated weights, which are adjusted during training to learn patterns and relationship in data.

① Input Layer

② Hidden Layer :- process input data
combines them using weights
applies activation function
passes to next layer

- ③ Output Layer
- ④ weights
- ⑤ Activation function

Detecting a shape



What is Binary classification?

involves classifying data into two classes,
/categories they can be labeled either 0 or 1,
true or false etc.

→ spam detection

→ Fraud detection

→ Medical diagnosis

Neural Network
classification

$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}) \dots (x^{(m)}, y^{(m)})\}$

$L =$ Total no of layers in network

$S_l =$ no of units in Layer l

Regularized Logistic Regression cost function

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \log(h_{\theta}(x^{(i)})) + (1-y^{(i)}) \log(1-h_{\theta}(x^{(i)})) \right] + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$$

Neural Network cost function

$$h_{\theta}(x) \in \mathbb{R}^k \quad (h_{\theta}(x))_i = i^{\text{th}} \text{ output}$$

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{k=1}^K y_k^{(i)} \log((h_{\theta}(x^{(i)}))_k) + (1-y_k^{(i)}) \log(1 - (h_{\theta}(x^{(i)}))_k) \right] + \frac{2\lambda}{m} \sum_{l=1}^{L-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (\theta_{ji}^{(l)})^2$$

CNNs

~~Common~~ convolutional Neural Network is a type of deep learning that structured to learn complex pattern and feature from input

It consist convolutional layers that apply filter to local patterns, pooling layers to reduce dimension, and fully connected layer to make final decision

① Input Layer

② Convolution Layer

→ set of filter

→ search for patterns

→ dot product with filter

→ creates feature map

③ Activation Layer

→ Activation function ReLU
(Rectified Linear Unit)

④ Pooling Layer

→ Reduce dimension of
previous layer

⑤ Flattening

→ convert pooling layer into
a vector

⑥ Fully connected Layer

→ decision making final

⑦ Output

why convolutional Neural Networks (CNNs)?

Image
1920 x 1080

\therefore Image size = $1920 \times 1080 \times 3$ RGB

\therefore 1st Layer = $1920 \times 1080 \times 3$

= 6 million neurons

\therefore suppose

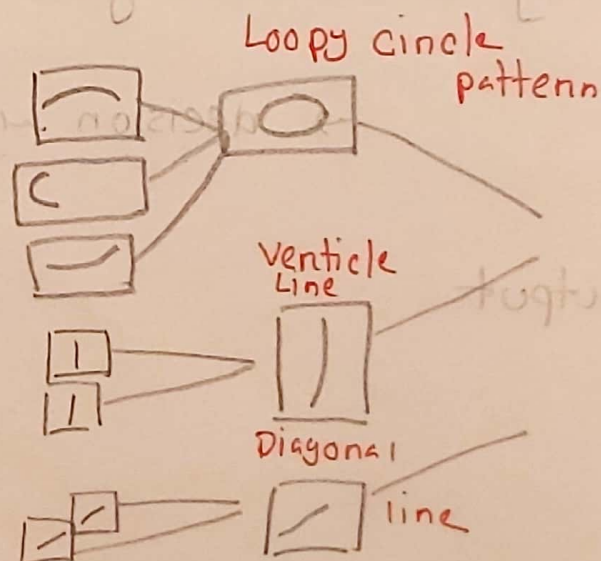
1 Hidden layer = 4 million neurons

\therefore Total weight = $6 \times 4 = 24$ million

Issue :- \rightarrow Too much computation

\rightarrow Treats Local pixels same as pixels far Apart

How Identity?



checking a if a digit has a loopy filter?
circle

-1	1	1	1	-1
-1	1	-1	1	-1
-1	1	1	1	-1
-1	-1	-1	1	-1
-1	-1	-1	1	-1
-1	-1	1	-1	-1
-1	1	-1	-1	-1

*

3x3 filter

1	1	1
1	-1	1
1	1	1

filter for loopy circle



$$\begin{aligned}
 &(-1 \times 1) + (1 \times 1) + (1 \times 1) \\
 &+ (-1 \times 1) + (1 \times -1) + (-1 \times 1) \\
 &+ (-1 \times 1) + (1 \times 1) + (1 \times 1) = -1/9
 \end{aligned}$$

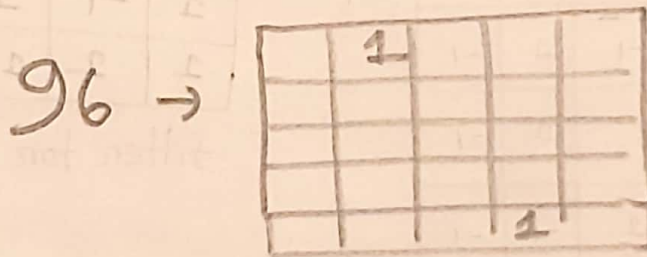
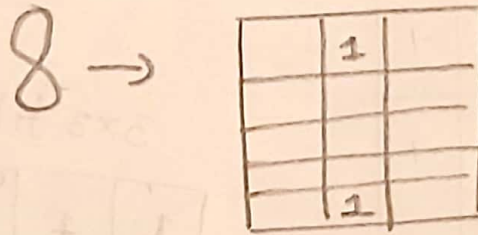
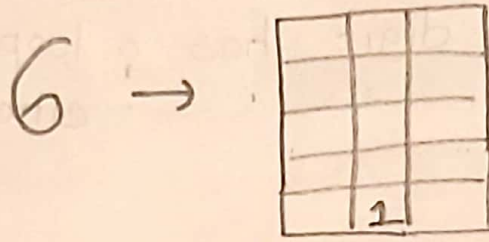
$$= -1/9$$

feature map

that means it has circle

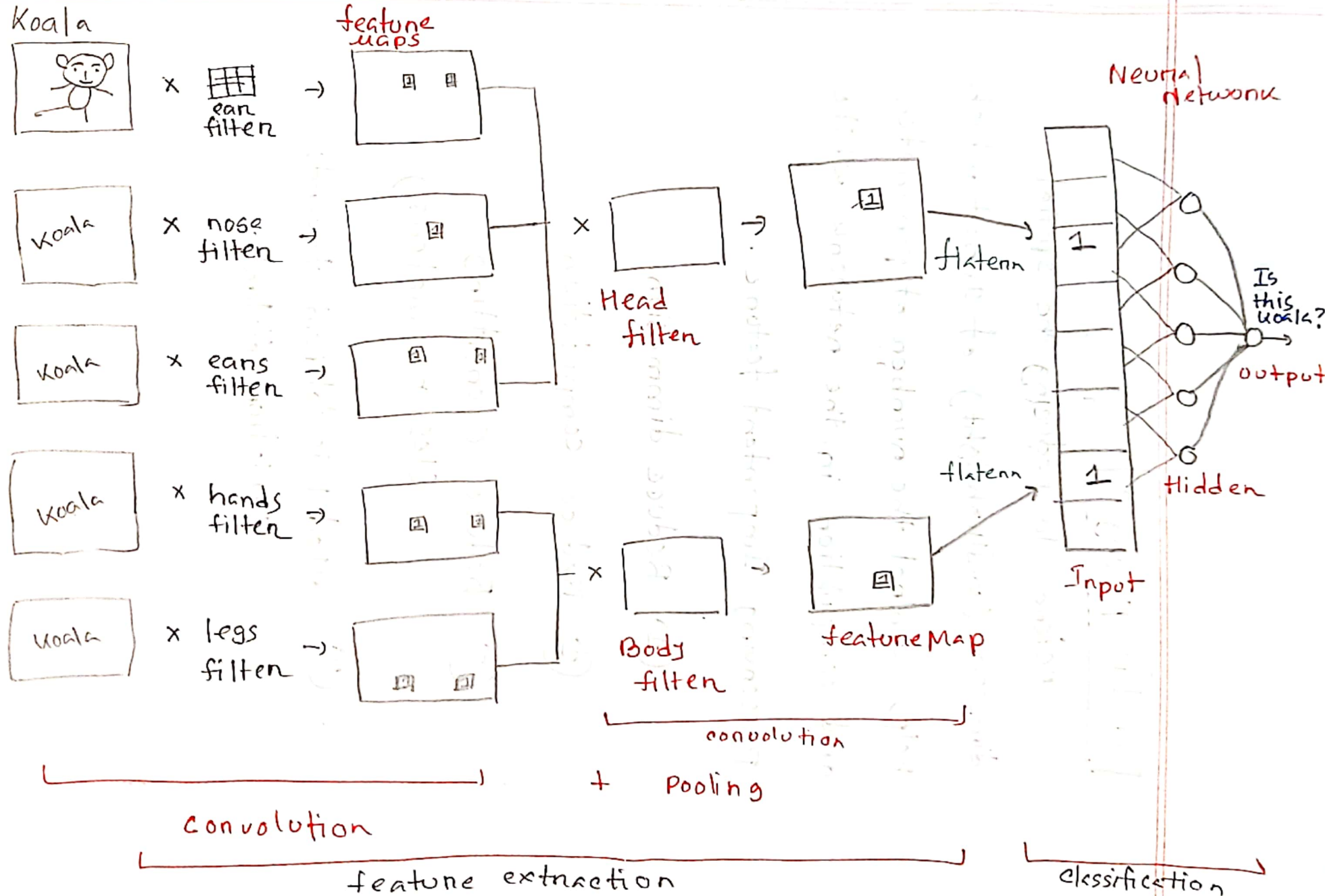
-0.11	1	-11
-0.55	.11	-0.33
-0.33	-0.33	-0.33
-0.22	-0.11	-0.222
-0.33	-0.33	-0.33

feature dectons



- ① Loop dectons — feature map-1
- ② Venticle Line dectons — map-2
- ④ diagonal Line dectons — map-3

convolution Filter



what is pooling ?

refers to reducing the spatial dimension (width \times height) of Input which helps control the number of parameters and computation in the network while preserving Important features.

- ① Reduce dimension
- ② Reduce computation
- ③ Reduce Overfitting
(less parameters)
- ④ Model is tolerant towards variation, distortion

max pooling

5	1	3	4
8	2	9	2
1	3	0	1
2	2	2	0



8	9
3	2

2x2 filter