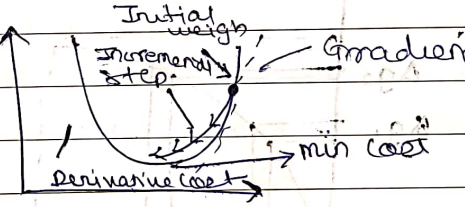
IA 2: 01:03pm

classmate
Date 16/04
Page

# Machine learning.

(1) Gradient Descent.

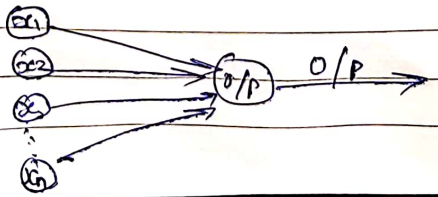
→ - One of the most commonly used iterative optimization algorithms of ML to train ML & DL models.
- Helps in finding the ~~to~~ local minimum of a function.
- If we move towards an negative gradient or away from the gradient at current point it will give the local minimum.
- If we move towards a positive gradient or toward the gradient from at current point it will give the local maximum.




- Objective of Gradient descent is the to minimize the cost function using iterations.

(19) Hebbian Learning Rule.

→ - Hebbian network is a single layer neural network which consist of one input layer with many input units & one output layer with one output unit.
- Architecture used for pattern classification.
- Works by updating the weights between neurons for each training sample.



Algorithm:

(1.) Set all weights to zero.
$w_i = 0$ for $i = 1$ to $n$ and bias to 0

(2.) For each input vector, S(input vect) : t(target o/p vect)
repeat step 3 to 5.

(3.) Set activations for input units with the input vector $x_i = s_i$ for $i = 1$ to $n$

(4.) Set the corresponding o/p value to o/p neurons
$y = t$

(5.) Update weights & bias by applying Hebb rule for $i = 1$ to $n$:

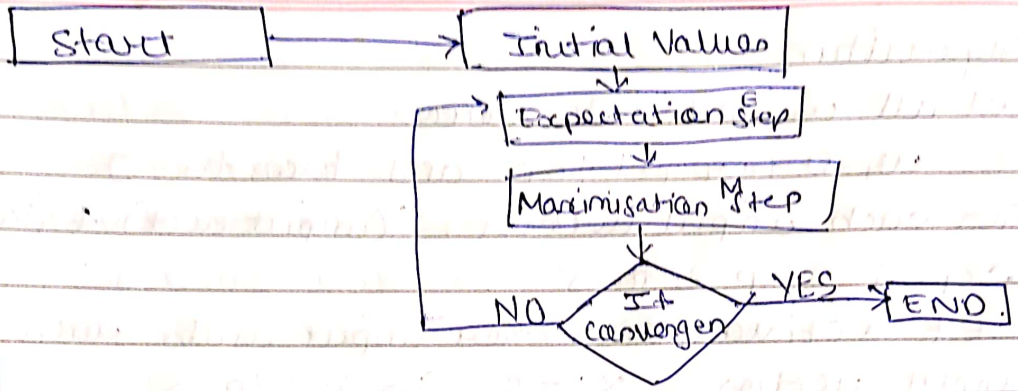
$$w_{i(new)} = w_{i(old)} + x_i y$$
$$b_{(new)} = b_{(old)} + y$$

(14) Expectation - Maximization Algorithm:

→ - Performs maximum likelyhood expectation
- Used to find the latent variable
↳ variable whose val is derived from other variables in form
- Basic for many unsupervised clustering algorithm.

- Steps in EM algorithm:

1. Initially a set of initial values are considered
- A set of incomplete data is given to system.
2. Next - Step - Expectation step - E step
Here, we use observed data to estimate or guess the values of missing / incomplete data
3. Maximisation step / M - step:
Here, we use the complete data generated in preceding e-step to update the values.
4. We check if values are converging or not
If converging - Stop
Otherwise repeat step 2 and 3 till convergence occurs




* Purpose of EM:
- In ML P.S, we obtain many relevant feature to build model, but only few are observable (Non-observable features are latent features)

* Applications of EM:
- Use to calculate Gaussian density of function
- Helps in filling the missing data of sample
- Used in NLP, CV
- Image reconstruction in medical & structural engineering
- Used to estimate parameters of mixed model

* Advantages
- Guaranteed that likelyhood will increase in each iter
- The E-step & M-step are easy to implement
- Solution to the M-steps exist in closed form

* Disadvantages.
- Slow convergence.
- Make convergence to local optimal only.
- Requires both the probabilities, forward & backward.

~~(6) Different types of Neural network.~~

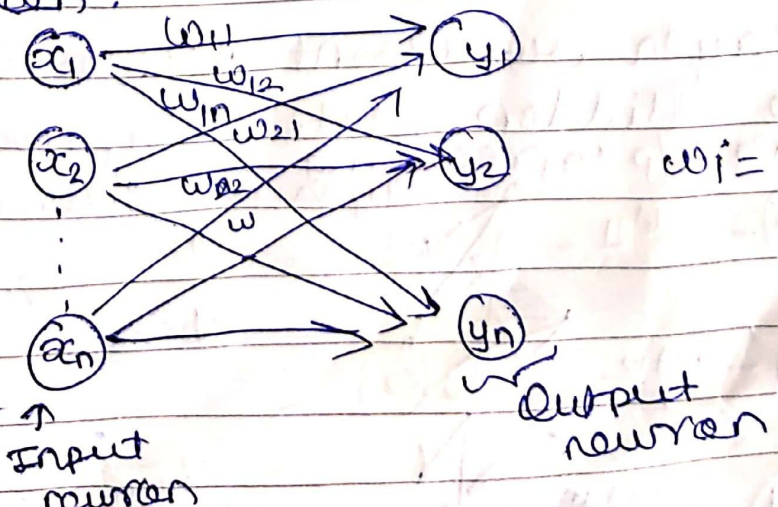
(21) Comparison b/n BNN & ANN.

| | ANN | BNN |
|---|---|---|
| Processing | Sequential & centralised | Parallel & Distributed |
| Rate | Process info in faster pace | Slow in processing information |
| Size | Small | Large |
| Complexity | Incapable to perform pattern recog | Enormous size & complexity of connections |
| Fault tolerance | Intolerance to failure | Implicitly fault tolerant |
| Control unit | Control unit monitor computer activities | All the processing is centrally controlled |
| Feedback | Not provided | Provided |

(11.) (6) Different types of neural networks.

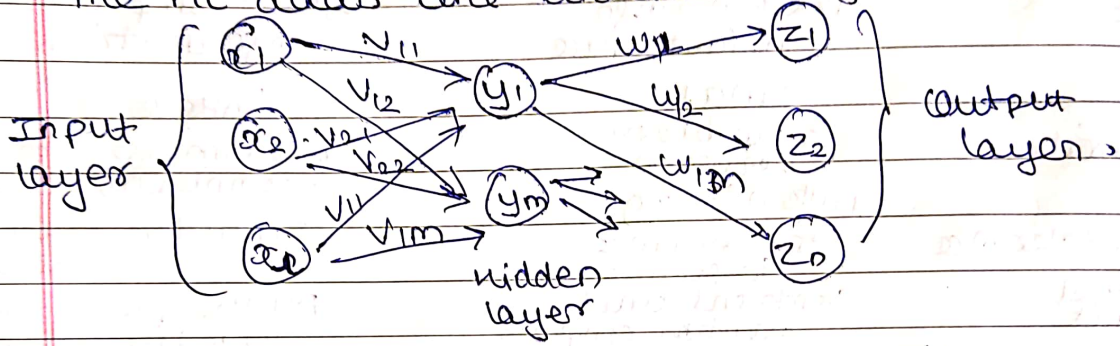
- Five basic types of neural network archi~

1. Single layer feed-forward network:
   - Two layers: Input layers receives Input signals
     Output layers receive output signal
   - Has Synaptic links carrying weights connect i/p n to o/p n but not conversely.
   - Called as single layer as output layer performs computation.



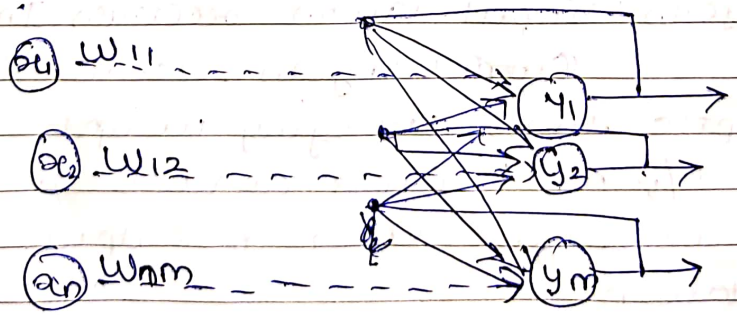

$w_i$ = weights.

2. Multilayer feedforward.
- Made up of multiple layers.
- Intermidiary layers beside i/o & o/p called as hidden layer.
- computational units of h.l are called hidden neuron
- The HL adds the intermidiary computation.

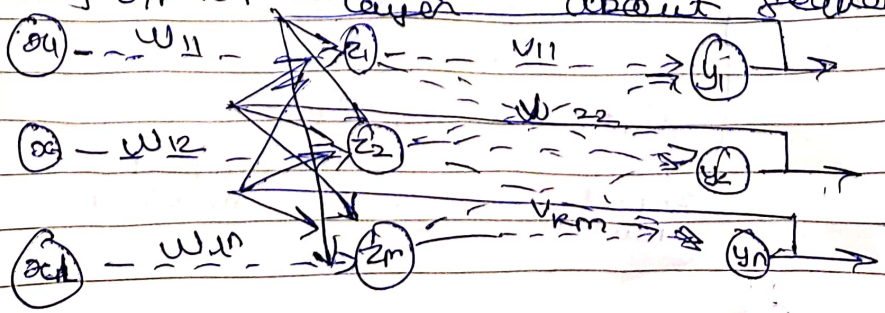



3. Single node with its own feedback.
- Output directs back as input to the same layer or preceding layer, results in feedback network.

4. single layer recurrent network.
- Recurrent network → feedback net with closed loop.



5. Multilayer recurrent network.
- Feature : Hidden state → captures information about sequence
- Processing o/p to preceding layer

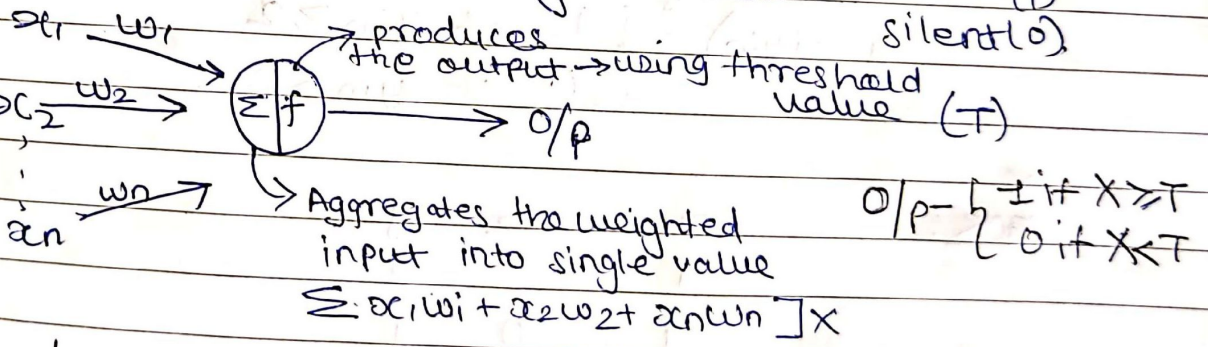


(14) Diff b/w single layer feed forward & multi layer feed forward

| Single layer FF | Multilayer FF |
|---|---|
| - Formation of layer is done by processing & combining elements | - Interconnection of several layers |
| - link b/w i/p & o/p | - ~~link~~ middle layer hidden layer |
| - Not efficient | - Efficient as complexity from HL |

(15) Mc culloch pits model

- First mathematical model of a biological neuron
- Basic building block of neural network
- Directed weight graph is used for connecting
- Two possible states of neuron. → Active (1) Silent (0)




$$o/p = \begin{cases} 1 & \text{if } X \geq T \\ 0 & \text{if } X < T \end{cases}$$

Bias / Threshold: Minimum value of weighted active inp for a neuron to fire

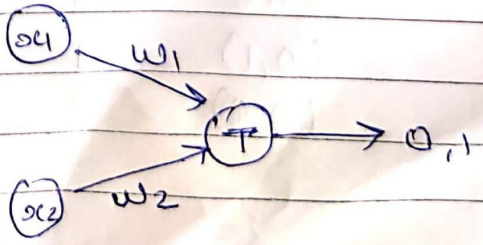
if effective i/p is larger than T.

- then o/p → (1) else (0)

GATES:

OR Gate:

| $x_1$ | $x_2$ | y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |




$w_1 = +1, \quad w_2 = +1$

$(0,0) \Rightarrow \Sigma x_i w_i = 0$

$(0,1) \Rightarrow \Sigma x_i w_i = +1$

$(1,0) \Rightarrow \Sigma x_i w_i = +1$

$(1,1) \Rightarrow \Sigma x_i w_i = +2$

Assume Threshold as +1 (T=1)

∴ Output of neuron y will be

$$y = f(y_{in}) = \begin{cases} 1 & \text{if } y_{in} \geq 1 \\ 0 & \text{if } y_{in} < 1 \end{cases}$$

AND GATE:

$w_1 = 1 \quad w_2 = 1$

$(0,0) \Rightarrow \Sigma x_i w_i = 2$

$(1,0) \Rightarrow \Sigma x_i w_i = 1$

$(0,1) \Rightarrow \Sigma x_i w_i = 1$

$(0,0) \Rightarrow \Sigma x_i w_i = 0$

| $x_1$ | $x_2$ | y |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

Assume Threshold val +2

Output of neuron y will be

$$y = f(y_{in}) = \begin{cases} 1 & \text{if } y_{in} \geq 2 \\ 0 & \text{if } y_{in} < 2 \end{cases}$$

XOR:

| $x_1$ | $x_2$ | y |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

$w_1 = 1$ ; $w_2 = -1$

(1,1) = 0

(1,0) = 1

(0,1) = -1

(0,0) = 0

Threshold T = 3

$1 < 3$

Threshold = 1

when $w_{11} = 1 \quad w_{12} = -1$

-4
-2
-2
0

0

1 -1
0
1
-1
0

NAND:

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 1 |

$w_1 = -2 \quad w_2 = -2$

$T = -3$

T=2

NOR:

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 1 |

$w_1 = -1 \quad w_2 = -1$

$T = -1$