

NN  
↓  
human  
brain mimic

GA  
↓  
(Genetic Algorithm)  
Combinatorial Optimization

Fuzzy logic  
↓  
0,1 एवं व्याख्येय  
domain

a	b	c
---	---	---

input

hypothesis

function (geometrically interpret रूपात् possible)

 $y = f(x)$ , family of  $f$  द्वाये काळे करावे।  
↳ a, b, c

प्रिले number return करावे, येतो द्वाये तुकाना

यादे ठेण letter.

Book : Neural computing and Introduction

→ Isal, Russel and T jackson.

Machine Learning → Probabilistic graphical model (PGM)  
→ Neural Network (NN)

NN एवं weak point को support करते PGM.

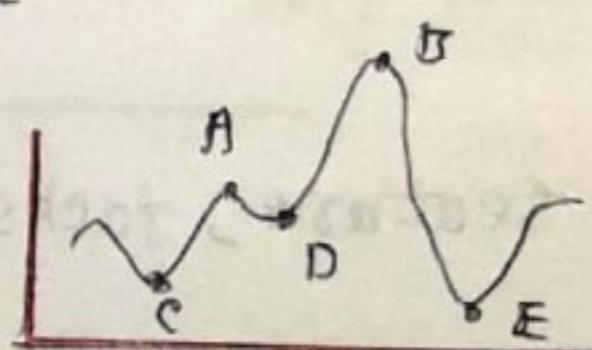
↳ saturated अव्याप्ति आहे। तरीम  $\rightarrow$  हवी capture  
काळे ठेणा एवण्ही algorithm

DL (Deep learning)

Optimization :  $\rightarrow$  Discrete optimization  
 $\rightarrow$  continuous ..

युन challenge optimal

$f^n \rightarrow$  geometric plot  $\rightarrow$

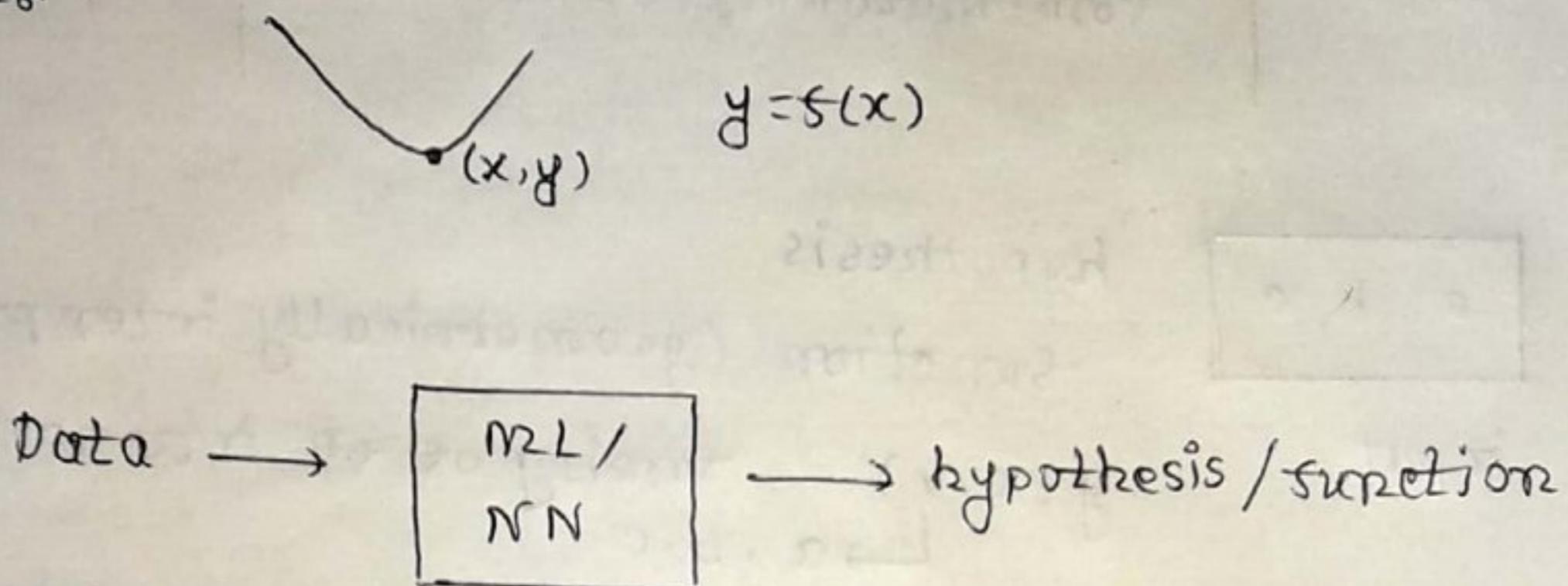


max A/B ?

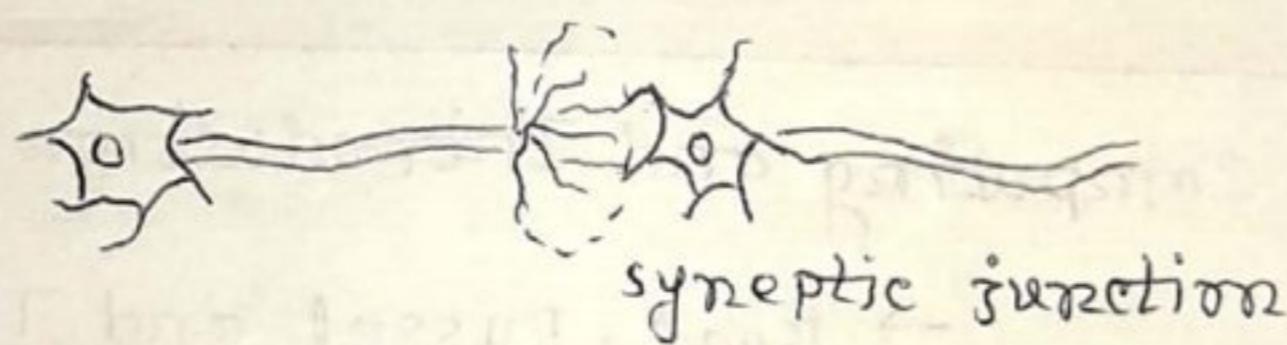
algorithm wise vary करते हैं किंतु max min optimal point

ML

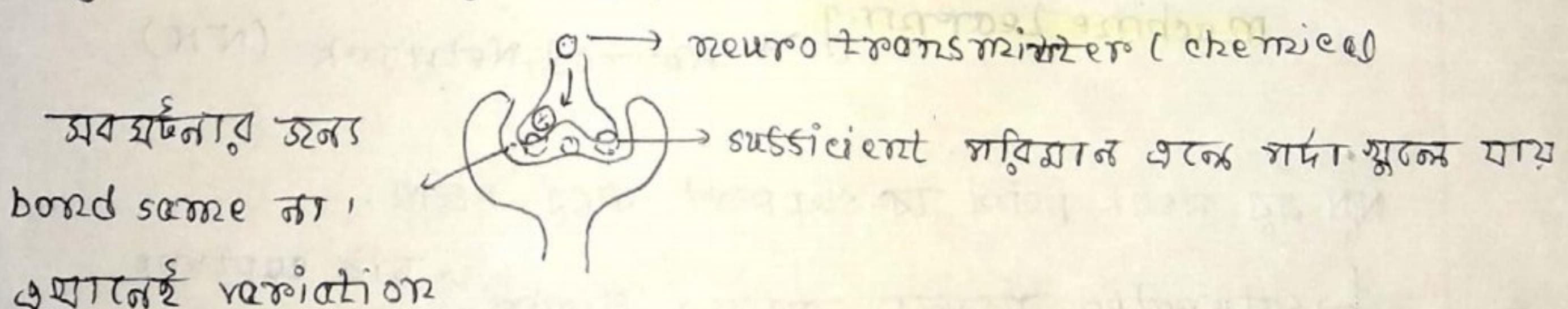
serially (greedy) के problem time योक्ता जाता है तो उसे algo  
use करें।



### Neuron एवं network:



signal के propagate करते किसी?



होता है।

Fig: Synaptic junction

### Synaptic cleft:

Data  $\rightarrow$  Biological concept  $\rightarrow$  algo  $\rightarrow$  SR.

Next class: Lecture, jackson एवं वह chapter शाही होगे।

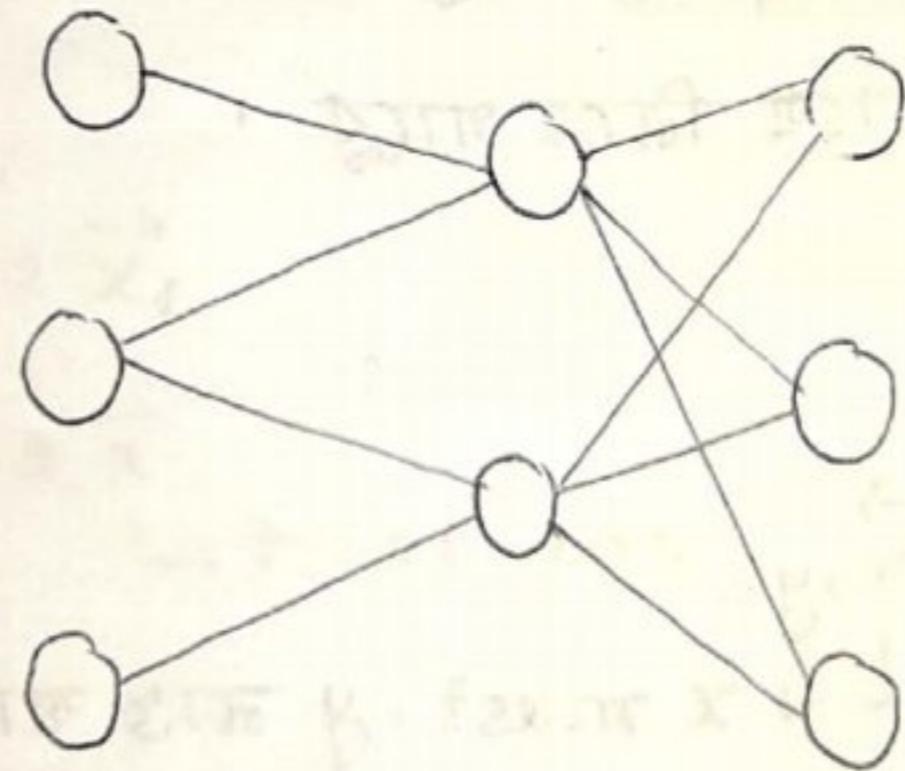
Lab: python :

1. numpy এর datastruct
2. panda input/output
3. matplotlib : lineplot, scatterplot

১০

Neural Network algorithm এর feature:

1. Graceful Degradation
2. Parallel computing
3. Fault identification.

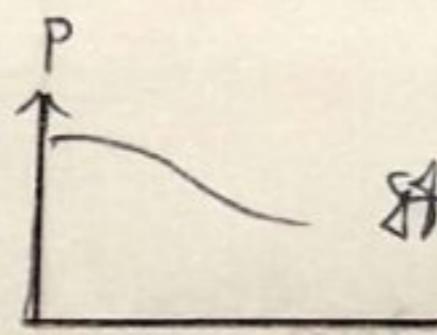


প্রজেক্ট মাত্র এর algo  
ফলোর কয়ে ?

Benefit:

- 1) Graceful degradation.

performance time to time কঠো যায় but NN এ<sup>o</sup> abruptly fall কঠো না ,

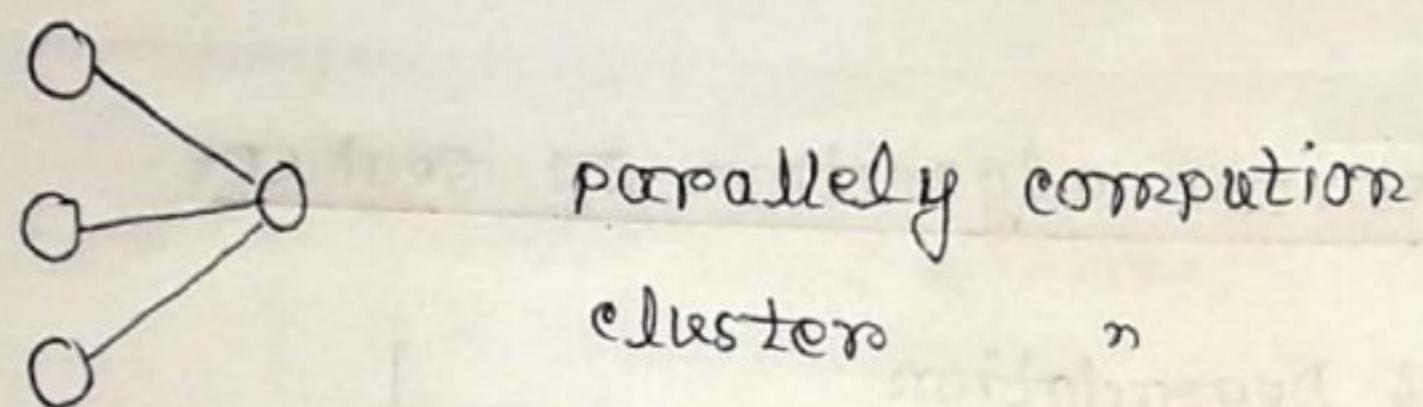


শীঘ্ৰে শীঘ্ৰে fall কঠো ,

একটা neuron শাখিয়ে গচ্ছে বা রাজু রাখে ) একেকের নতুন রাখে  
প্রতিস্থাপন করতে হয় (resetting করা হবে )

(age এর প্রভাব অস্তিত্ব আছে ) [ audio, video

pruning করে neuron গুলো রাখ দেওয়া আছে ,



### Fault Tolerant:

একটা unit faulty হলে graceful degradation. এবং তা  
আন্তর্ভুক্ত করে রাজু করে নিতে পারে ,

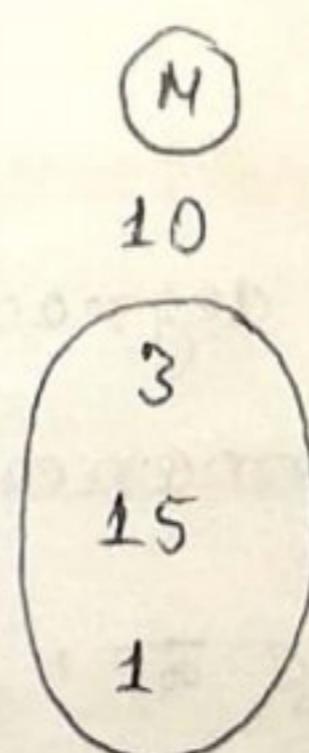
D dimension  $\mathbb{R}^D$   $\rightarrow (x, y)$   
 $\downarrow$   
 x must, y নাও প্রাপ্ত পারে .

4 dimensional space.

$$\begin{aligned} D \\ X \rightarrow 10 \\ Y \rightarrow 5 \end{aligned}$$

$$(\text{cross area}) CA \rightarrow 50 \rightarrow 20$$

$$C \rightarrow 2$$



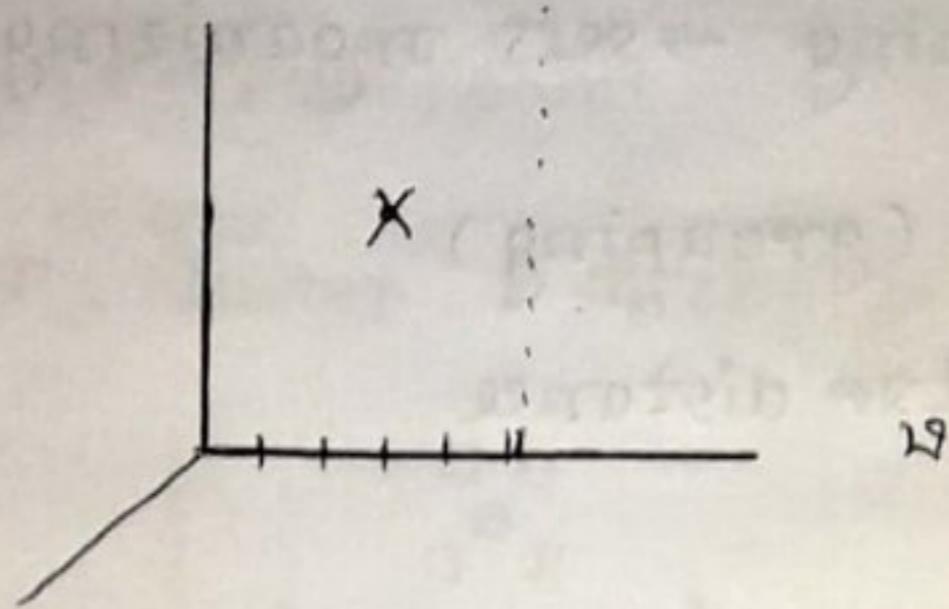
5, 20, 2 উচ্চ দুর্বল

3, 15, 1 " মার্কার "

distinguish feature

dimension reduction: same का reduce करा !

eA

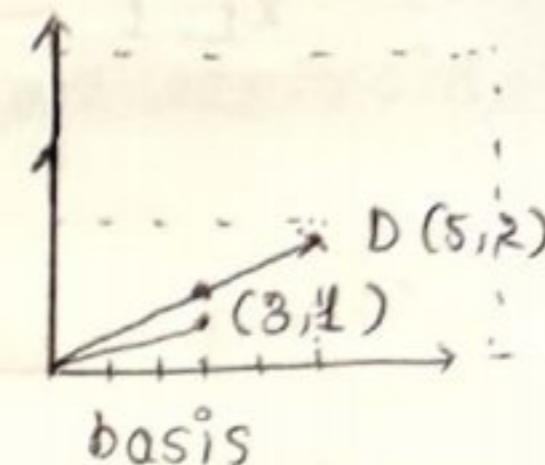


eI

feature space

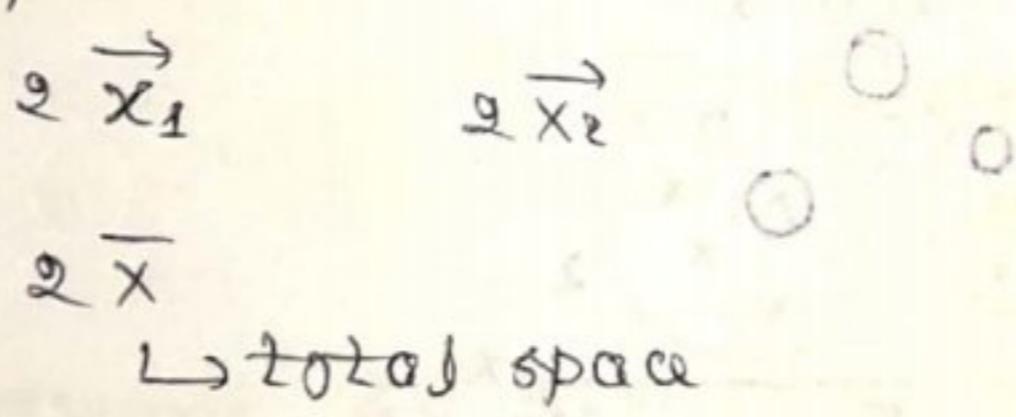
(Duster)

	$x_1$		$x_2$	
$x_{11}$	5		3	$x_{21}$
$x_{12}$	2		1	$x_{22}$
$x_{13}$				



Duster size का राशि देते हो distinguish करा tough. अर्थ scaling  
(dot product)

करा होना,



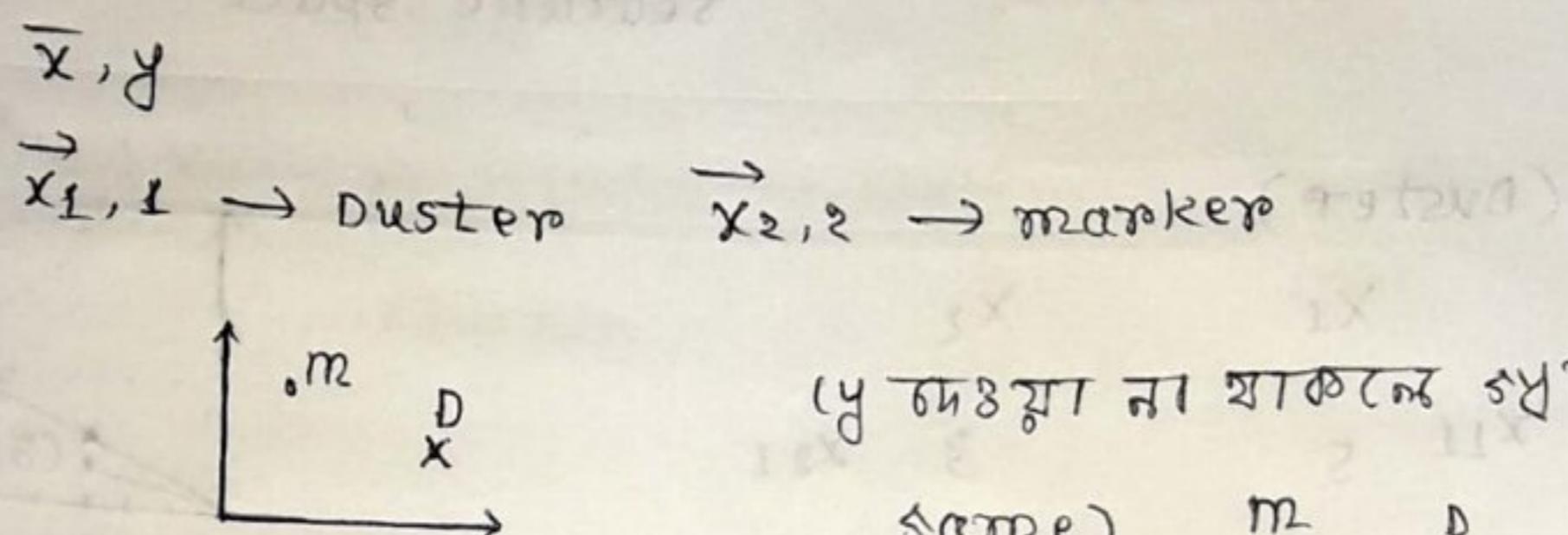
linear algebra rotation and transformation अपूर्ण गुणिता होता !

गणितीय रूप से होते हो property यह ठिक आते,

Unsupervised Learning  $\rightarrow$  self organizing map.

clustering (grouping)  
↳ vector distance

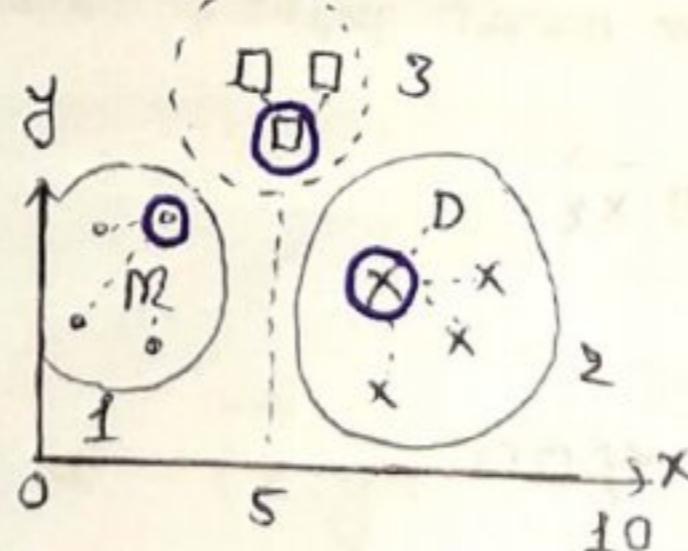
20



Why unsupervised?

- Label collect কোটা costly
- ex: rare diseases এর image

(যদিয়ে না আমার স্বাক্ষর করা  
হৈলে)       $m$        $D$   
computer এর কাছে কৃতৃপক্ষ হৈলে,  
না কোনটা বি?



ধৰ্য,  $K=3$

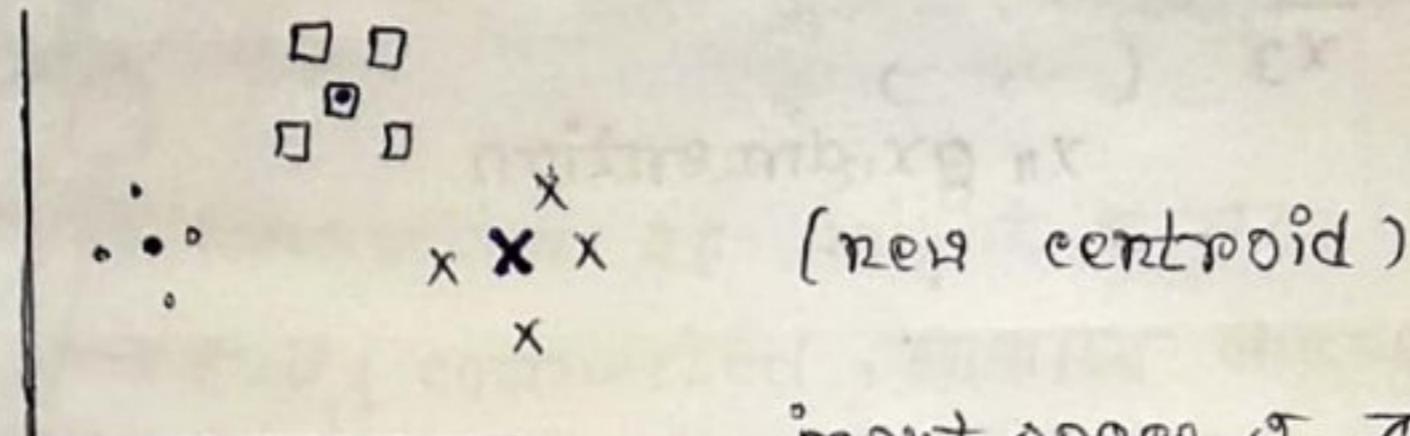
[multiple clusters একি জিনিয়েও represent কোটে গাঢ়ে]

- select  $K$  random point (centroid)
- Assign nearest samples to corresponding centroid.

এই interpretation,  
 $x_1 \cdot 1$   
 $x_2 \cdot 2$   
 $x_3 \cdot 3$

- select  $k$  (average) point

അതോക cluster ന് നിങ്ങളേ മുമ്പേ avg.



iteration കൂട്ടെ കൂട്ടെ ഏതൊക്കെയുള്ള exact centroid ശാന്തിയാ യാണോ

അതണ ആറു change ഹാരെ ശാന്തിയോ നാ ! stable position

$$\sum_{i=1}^n \sum_{k=1}^K \text{distance}(c_k, x_i)$$

Intra cluster distance (minimize)

Inter cluster distance (maximize)

$$\sum_{k=1}^K \sum_{i=1}^n \text{distance}(c_k, c_i)$$

Neural Network എ തechnique ഇന്തോ self organizing map

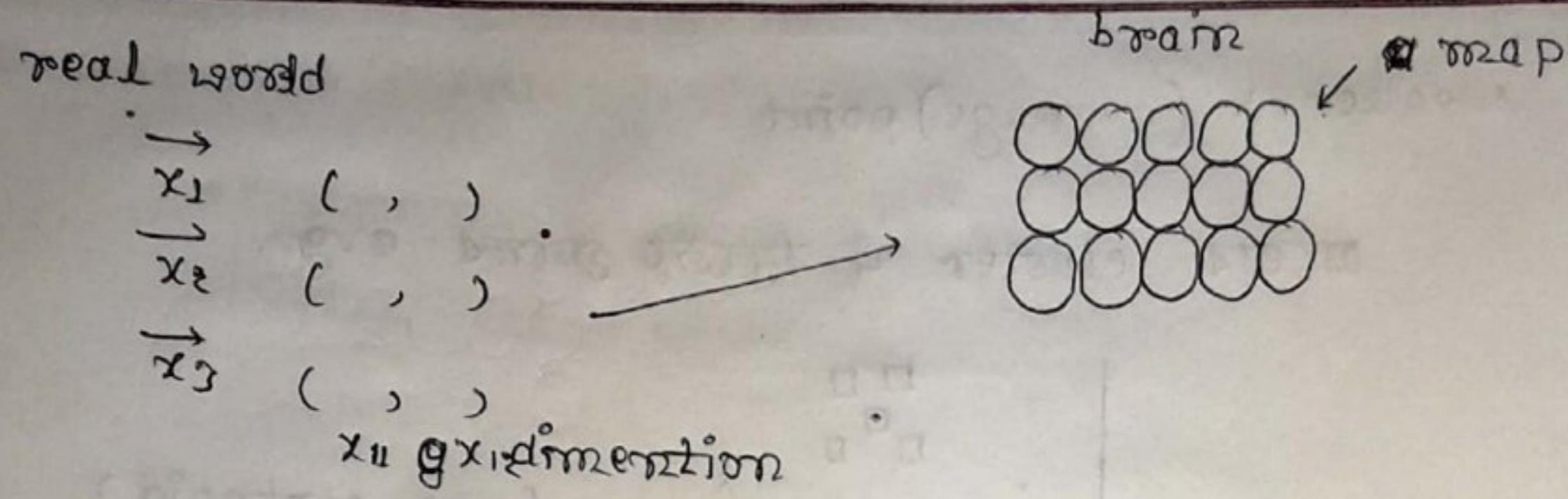
- unsupervised
- map input vectors to a map of different dimensions
- Patterns close to one another in the input space, should be close to one another in the map. (cluster)

$\rightarrow$   
 $x_i$

clustering ദി ബ്രൈൻ സൈസ് എ ഇല്ല ,

pattern : cluster type അവഗുണം മപ എ ഏകദി ജാമ്പഗാമ്പ ശാന്തിയോ !

അവഗാമി ദി വിവരിച്ചി ബോർഡി ബോർഡി ബോർഡി



### Kohonen Self Organising map

- unsupervised learning methods
  - We have to make two basic assumptions about the network.
    - class membership is broadly defined as input patterns that share common feature
    - network will be able to identify common across the range of input pattern.
- SOM 3rd step :
- Find the BMU (Best matching Unit) for a pattern.
    - ↳ competitive learning
  - is selected according to similarity, between current input values and all the nodes in the grid.

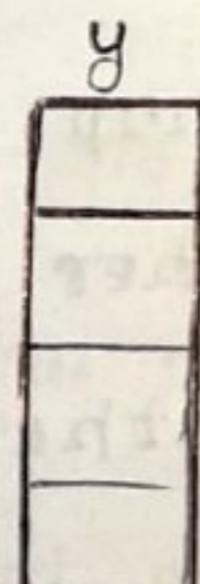
Input (	$x_1$	$x_1$	
	:		
	$x_{10}$	$x_{10}$	

2 dimension

$10 \times 2$

Centroid		

2 dimension



1 dimensional

K means

- Only a single node is activated at each iteration

self-organizing maps differ from other artificial neural network as they apply competitive learning as opposed to error correcting learning (such as back propagation with gradient descent), and in the sense that they use a neighborhood function to preserve the topological properties of input space.

neurotransmitter charge / weight change

single layer linear  
2D grid of neurons

connected

जा

node don't know  
value of its  
neighbors

$\vec{x}$

object  
duster  
marker

$x_{11}$   $x_{12}$  duster

markers

$\circ \rightarrow$  एवं एक वर्ते वज्रे वज्रे

$\square \rightarrow$  एवं एक अवृत्तिरूपी मान्यता

$d(e - x_i)$   $d(\vec{v} - x)$  BMU find

similar object मुले काढाले हाते नेट नोड ए

प्रथम duster एवं इन्हें एक नोड, लेकिन duster एवं इसके बाहरे

object हाते, so better assumption एवं इन्हें region बनवा हए

lab: color node

After clustering, each node has its own  $(i, j)$  coordinate which allows one to calculate the Euclidean distance between 2 nodes.

5 को चारे 100 एवं इसके बाहरे फैलते हैं  
duster एवं region form

real world एवं इसके region form करना जैसा। region create  
कामार इन्हें वज्रे वज्रे वज्रे duster एवं इसके बाहरे फैलता है

Suppose 5 - marker [marker]

best match 25 उत्तम, best match एवं उत्तम region है

मात्र

5 एवं convert करते हैं, मात्रे जिसे 5 उत्तम

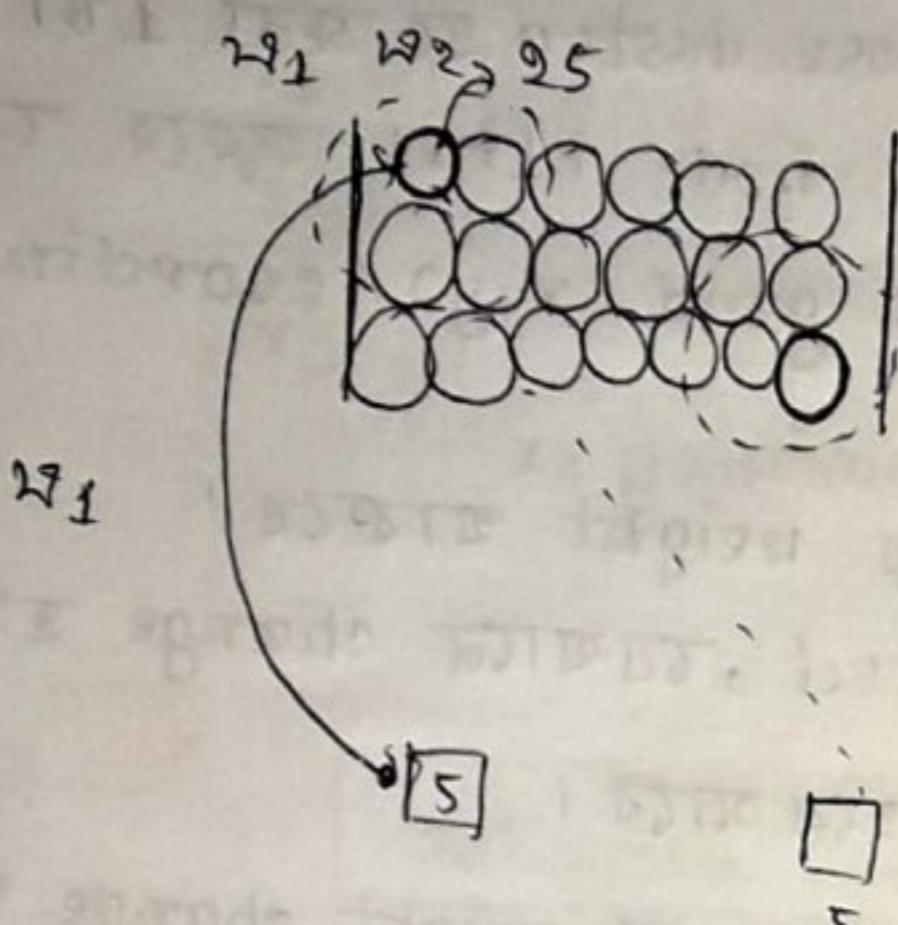
→ change BMU

निजे चारे, आगे निजे चारे

→ change neighbours (radius तक निजे चारे निजे चारे)

charge is max

radius = min dist / 2



$$25 - 5 = 20$$

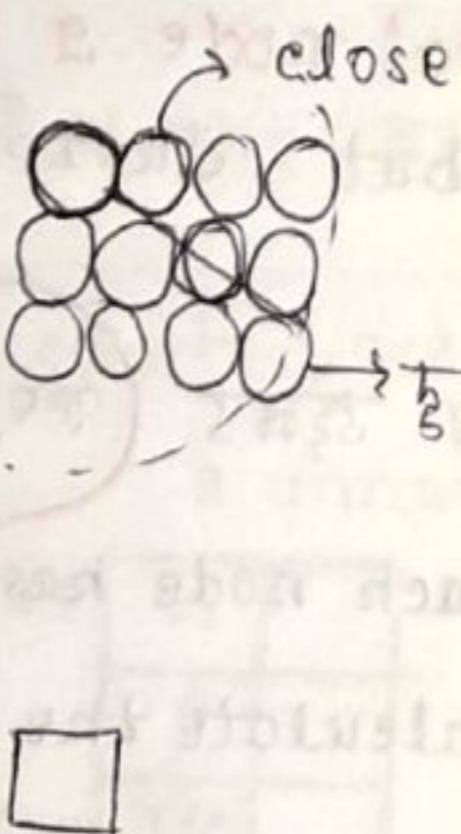
$$n = [0 : 1]$$

$$27_2 + 1 = 27_2 + 4 \times n$$

iteration

O এর মান right থেকে আয়ে,

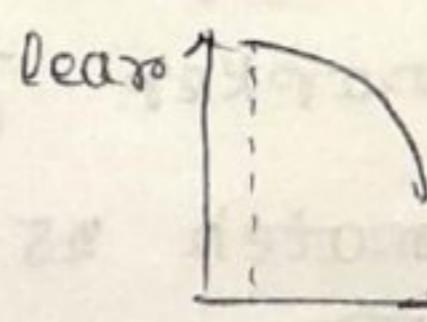
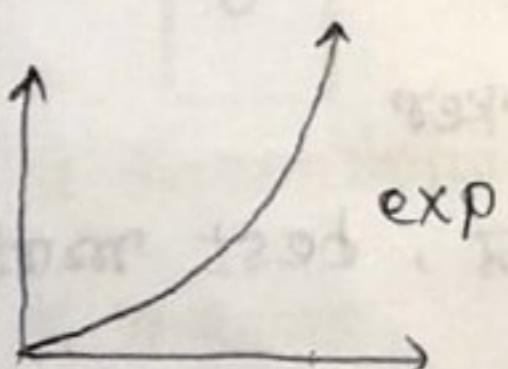
neighbourhood:



node ক্ষেত্রের x, y -এর dist measure করা

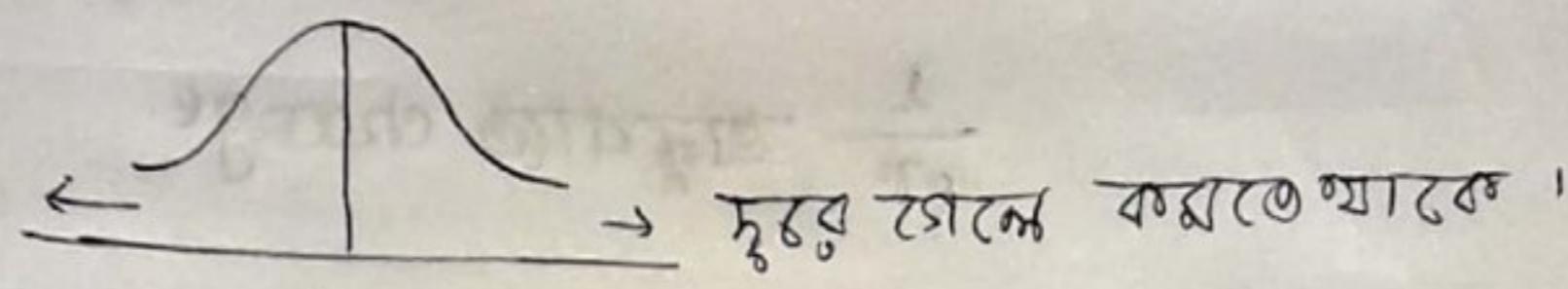
charge neighbours এ তুরেও টা দেখি charge ইন্ডে,

$$27_2 + 1 = 27_2 + \theta n \propto n$$



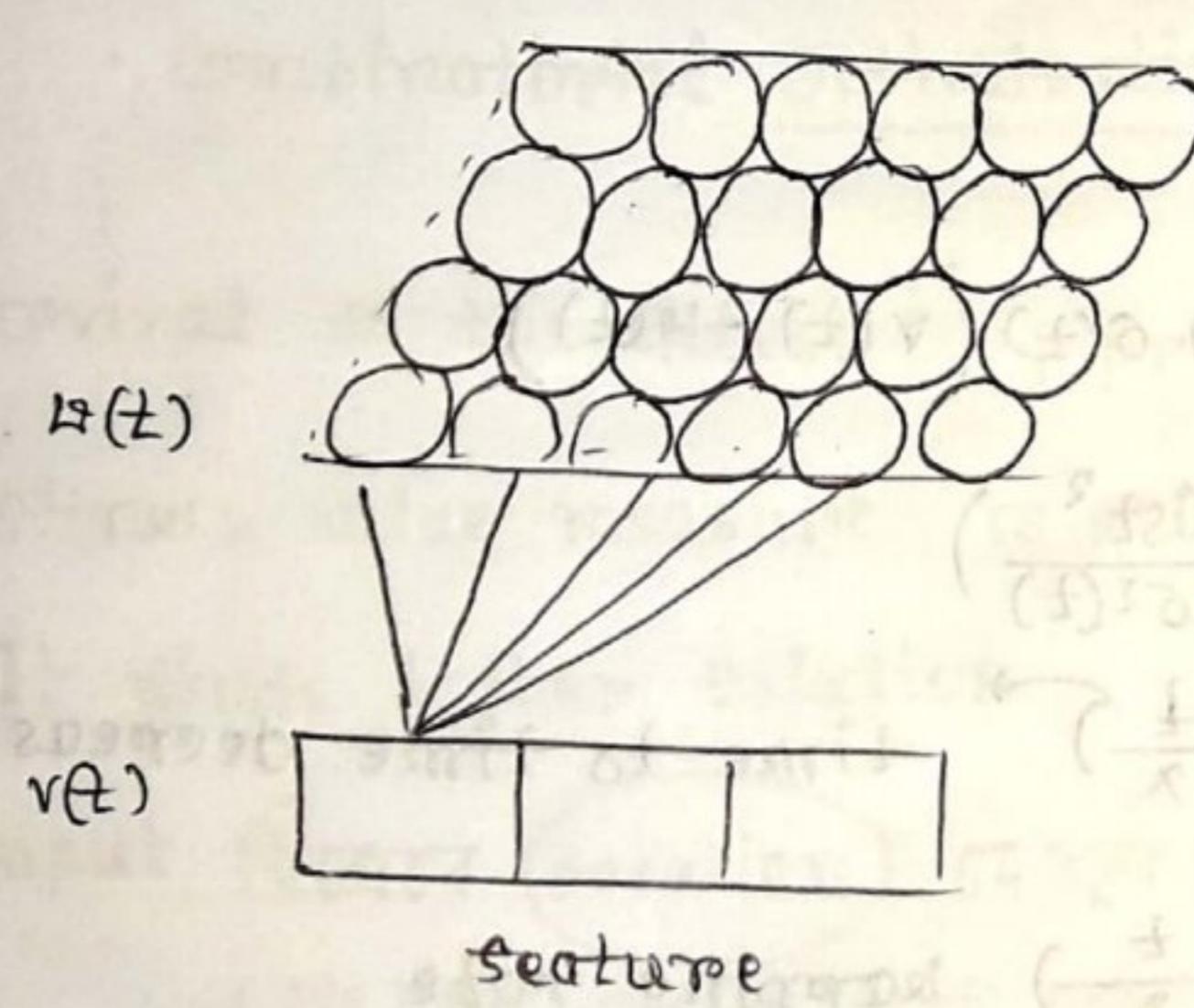
negative exp on

radius যখন কম হওয়ার অ্যন্তর্ভুক্ত,



→ Decrease the radius ( $r \downarrow$ )

SOM learning:



map ହତେ ହତେ ଏବେର୍ଟ୍ଟା  
region ଏ ଏବେର୍ଟ୍ଟା color ହରେ

1. weight 0-1 ପର୍ଯ୍ୟେ ମୁହଁ ମଧ୍ୟ ମଧ୍ୟ ମଧ୍ୟ initial value;

2. ( weight - weight ) time consuming ତାହିଁ color change

ଥିବେ determine ବଳୀ ଯାଏ । similar color ଓ କ୍ରୂଣେ

3. Iteration. input pattern ଆଜେକୁ ଫୁଲୋପ୍ରାପ୍ତ ମଧ୍ୟ ଏବେର୍ଟ୍ଟା choose

- radius change ( $r \downarrow$ )

- color change : ନିଜେର , radius ଏବ୍ୟା ମଧ୍ୟ ଯାଏ

• राशिएँ पूळाने charge देतीं ।

$$\frac{1}{\epsilon^{\eta}} \text{ अनुप्राप्त charge}.$$

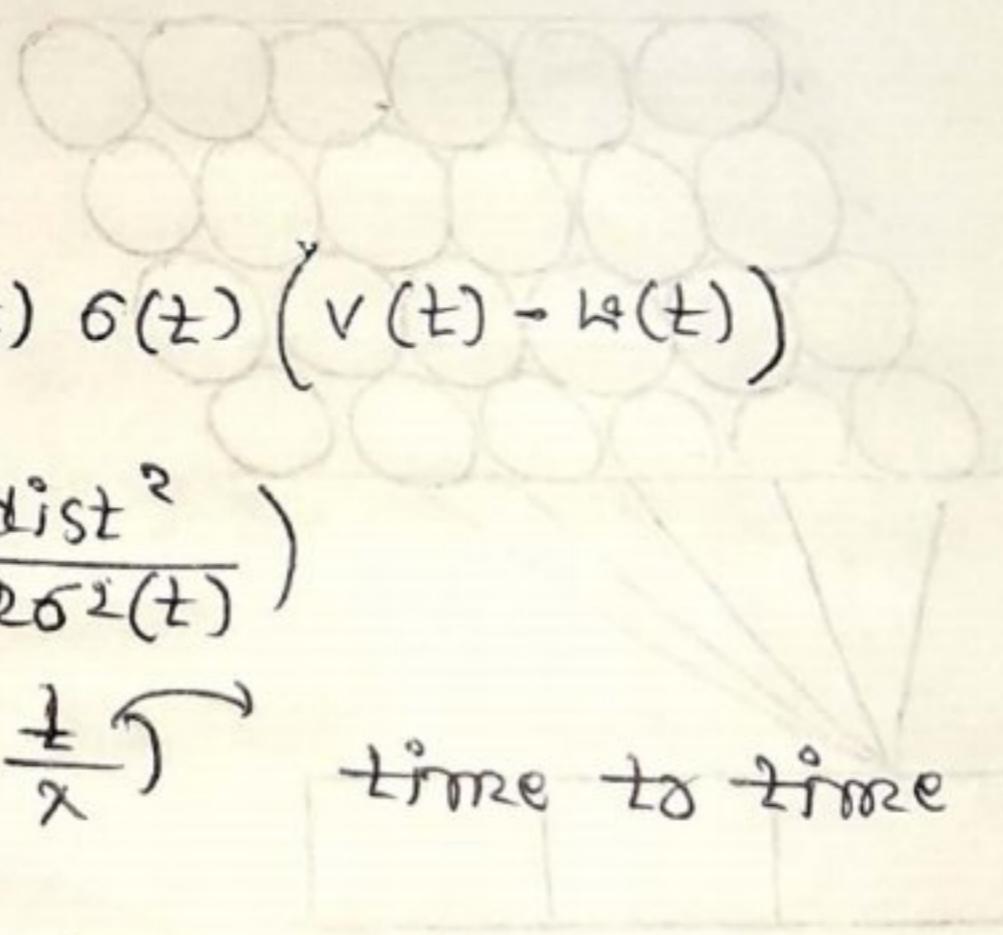
$$M_{t+1} = M_t + (V(t) - M(t))$$

$\hookrightarrow$  यह कोणता correction ले 0.001

मुद्रोंकर्ता change करते ना, फ्राक्चन नियंत्रे change ।

काढते मवालो एकी छाड़े change हय ना, नुस्खे पूळो करा!

Human brain ने एडाप्टे करते ।



$$M_{t+1} = M_t + \Theta(t) \sigma(t) (V(t) - M(t))$$

influence rate  $\rightarrow \Theta(t) = \exp\left(-\frac{\text{dist}^2}{2\sigma^2(t)}\right)$

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\lambda}\right) \quad \text{time to time decreasing}$$

$$L(t) = L_0 \exp\left(-\frac{t}{\lambda}\right) \quad \text{learning rate}$$

$\hookrightarrow$  1 दृष्टेष्ठ राहा यावे ।

$$\left( -\frac{\text{iteration}}{\text{time constant}} \right)$$

radius = radius<sub>0</sub> \* e

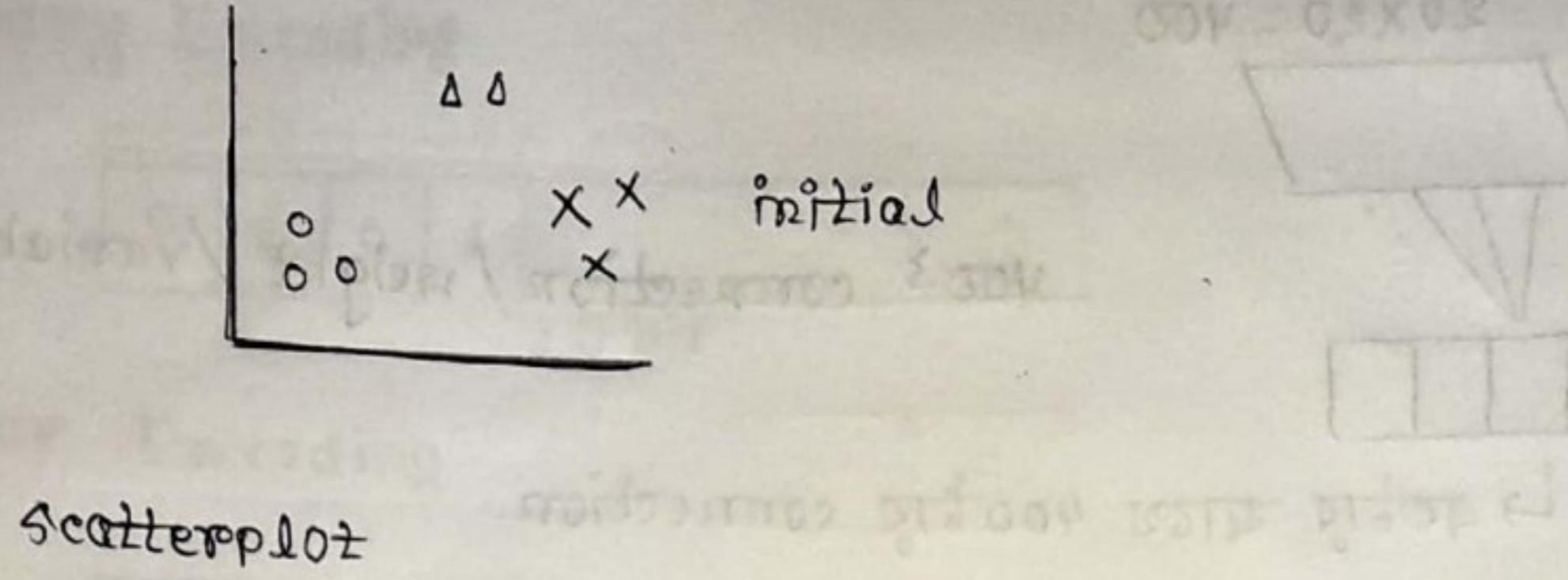
$$\text{time-constant} = \frac{\text{iteration}}{\log(\text{radius}_0)}$$

Lab 1 : K-means

(↓) update centers.

प्राच रेखा गुरु विभाग इंजीनियरिंग एवं विज्ञान

GA is a searched base optimization technique based on the "survival of the fittest"



3D

Genetic Algorithm finds better solution

- Combinatorial Optimization

Iteration by iteration property  
change करे optimization तरीका

"survival of the fittest" , problem के तरीके solution

• fitness value measure , to solution के fitness value वाले तरीके choose

• It finds better solution

• input हितवे solution क्या है, best fit solution कि

output. (We have population of possible solutions to the given problem.  
These solution then undergo recombination and mutation, producing new children, and the process is repeated)

$$f(a, b) = a + b - c^2 + \frac{1}{d} + \frac{1}{e^2} - \frac{1}{f^3} - g^3$$

तरीके combination के तरीके  $f(x)$  maximum?

• differentiation

• pos के max

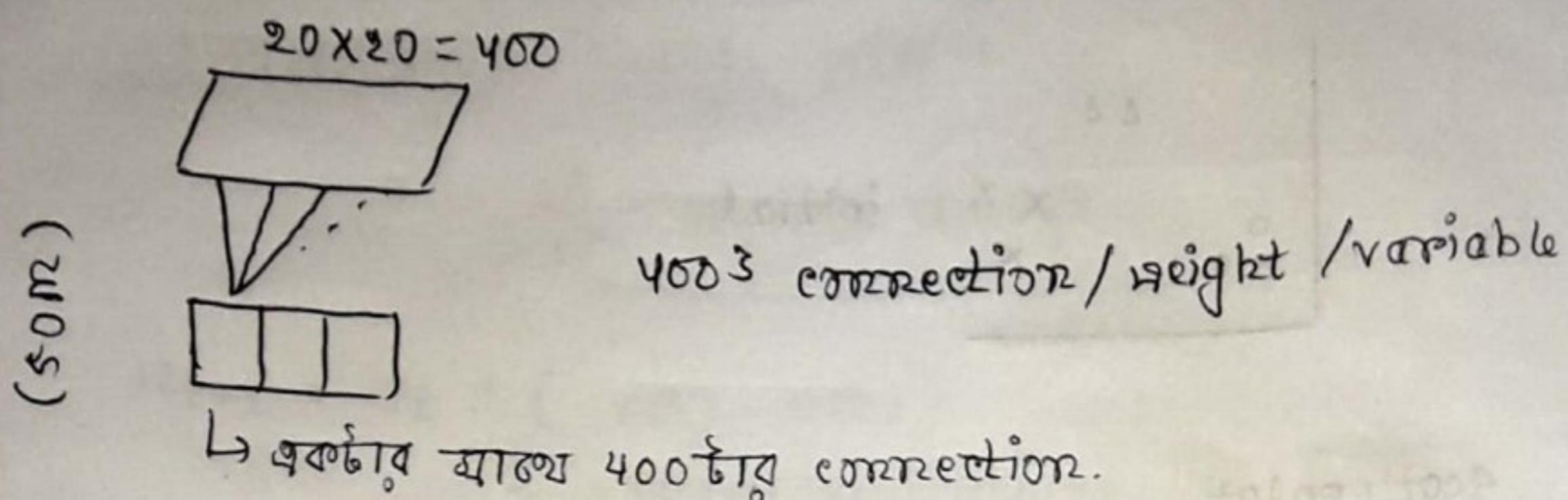
• neg के min , for आवाहन inversely proportional

तरीके आवाहन variable

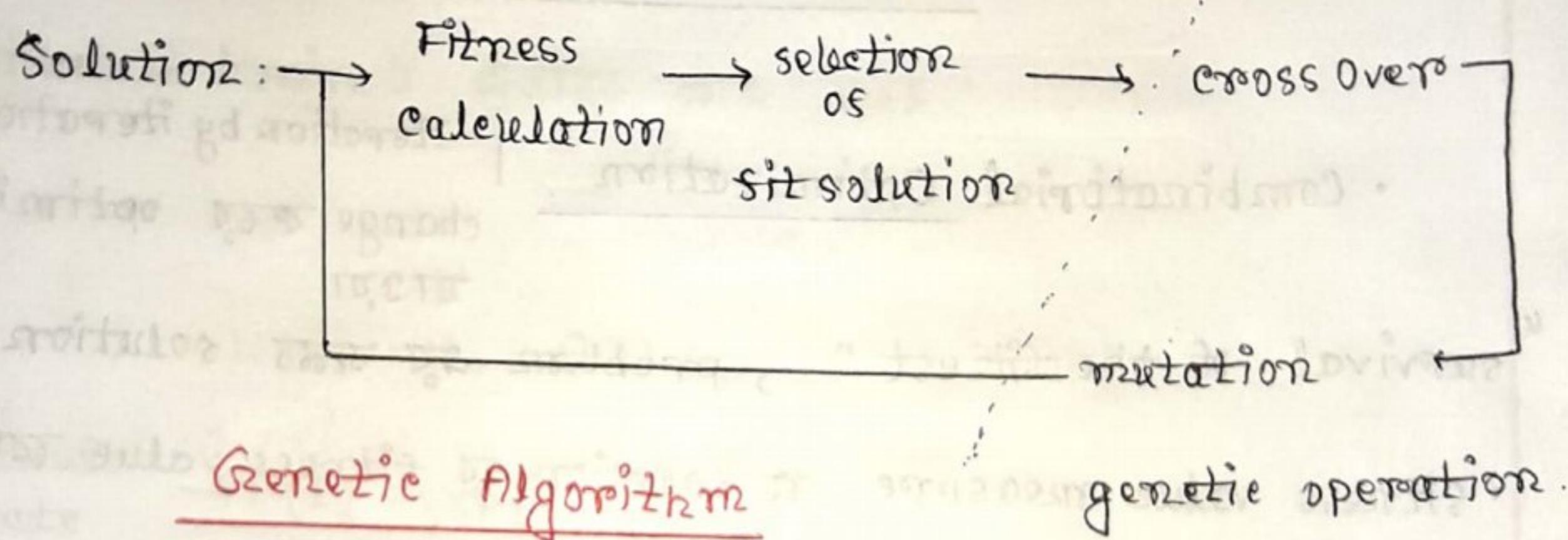
Each candidate solution is assigned a fitness value and fitter individuals are given a higher chance to mate and yield more 'fitter' individuals.

In this way we keep evolving better individuals or solution over generation.

chromosome  $\rightarrow$  solution  
 gene  $\rightarrow$  chromosome  $\therefore$  individual unit



But GA का size तक होने feasible time  $\therefore$  complete होना  
 time infinity होने के बाद भी solution नहीं होती।



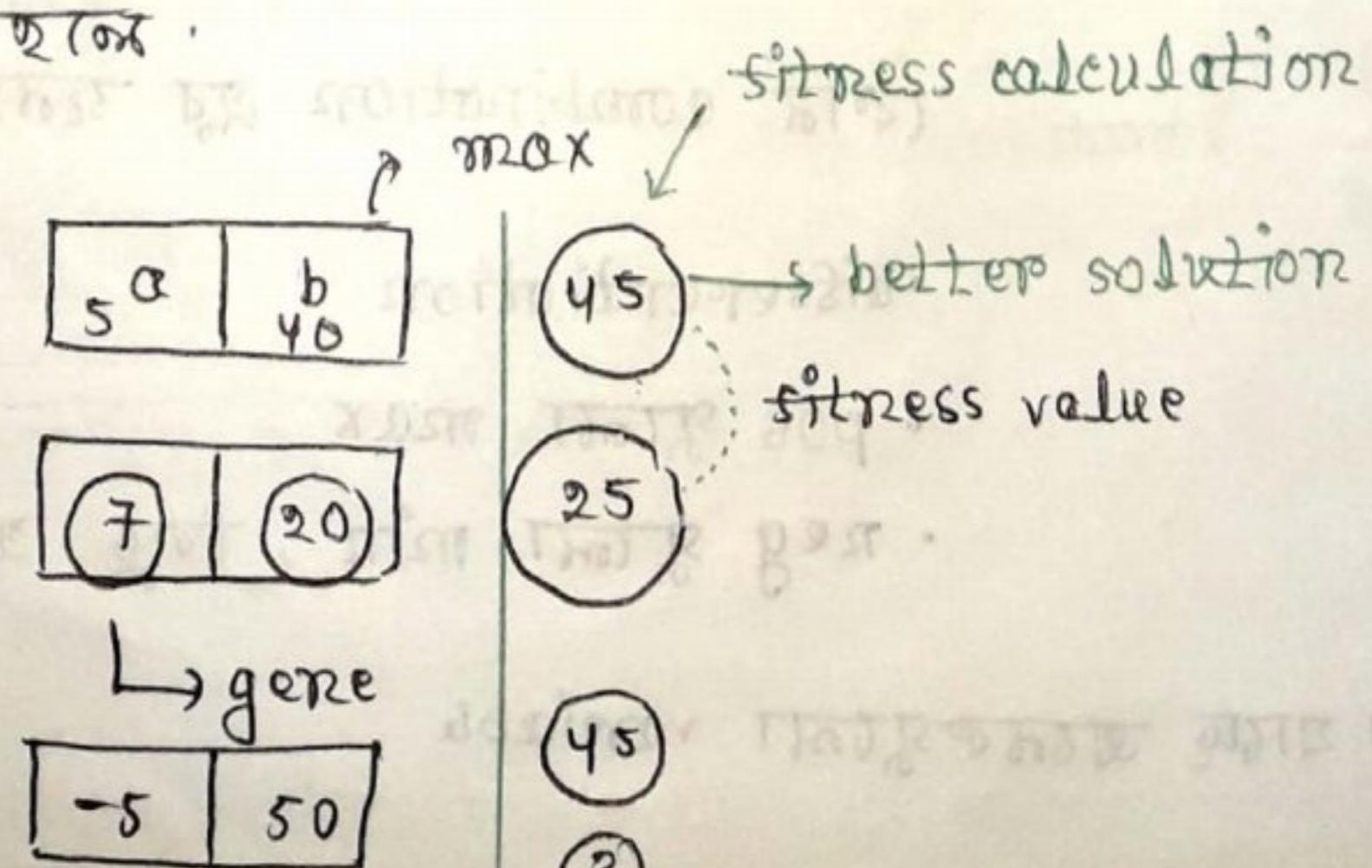
Suppose a function  $f(a,b) = a+b$

Output:  $\max f(a,b)$   $a \in [-10, 10]$

Encoding is a process of transforming from phenotype b  $[-50, 50]$

integer encoding (soln को किसी represent)

✓ integer soln.



$$\begin{array}{r}
 5 \rightarrow 4 \text{ bit} \\
 40 \rightarrow 6 \text{ bit} \\
 \hline
 10
 \end{array}$$

### Binary Encoding

↔	$a$	↔
0	1	0
1	1	

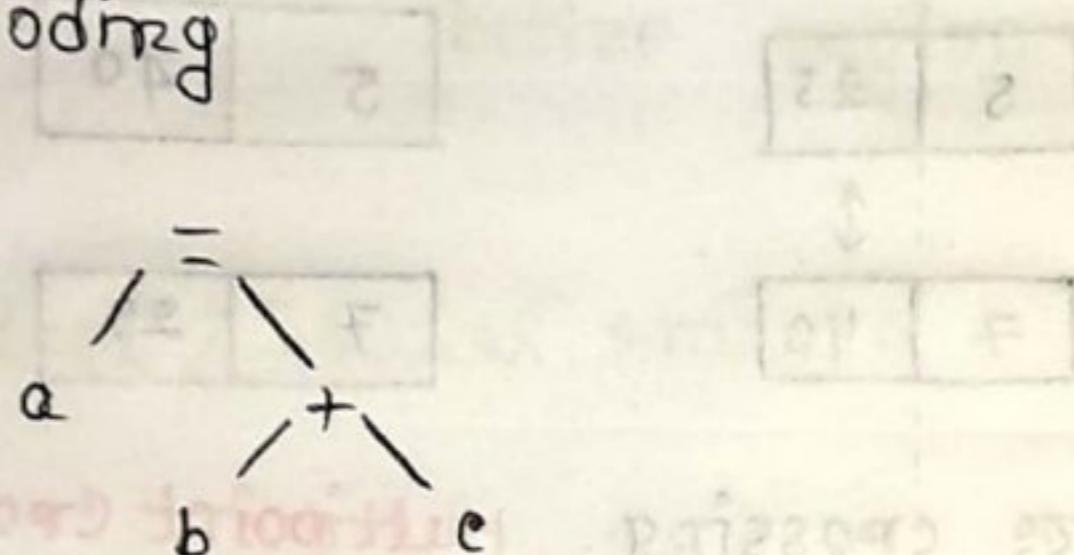
10 bit

### char Encoding

↔	28	+	4	↔
---	----	---	---	---

↔	a	+	b	↔
---	---	---	---	---

### tree encoding



encoding করে chromosome তৈরি করা হয়। তারপর genetic value থেকে  
থেকে fitness calculate করা।

Cross Over: chromosome এর cross over

better characteristics পান্তে অন্তর্ভুক্ত করা

করা হয়।

random selection

1000 solution

iteration

selection technique: roulette wheel selection use করা

rank selection

tournaments selection

এব্যাবে selection করা যায়।

এটা করে selection করে best টা নেওয়া যাবে।

randomly select

Mutation is related to the "exploration" of search space.

It may be defined as a small random break in the chromosome, to get a new solution.

Randomly select:

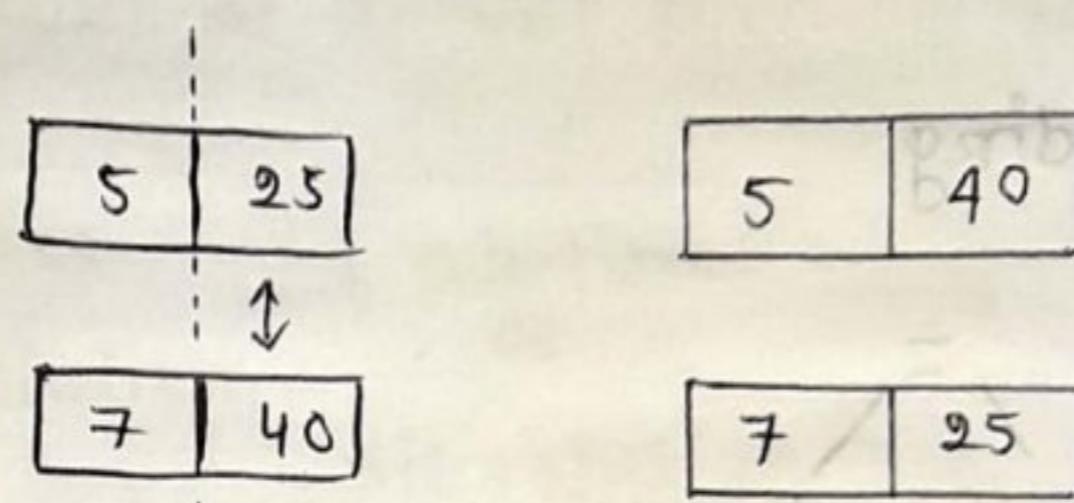
5	40
---	----

7	25
---	----

### Crossover:

single point एवं यांत्रिक crossing / swap (tails are swapped)

One point crossover

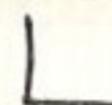


आवाह इसे selection एवं crossing Multipoint crossover

7   25	5   50
--------	--------

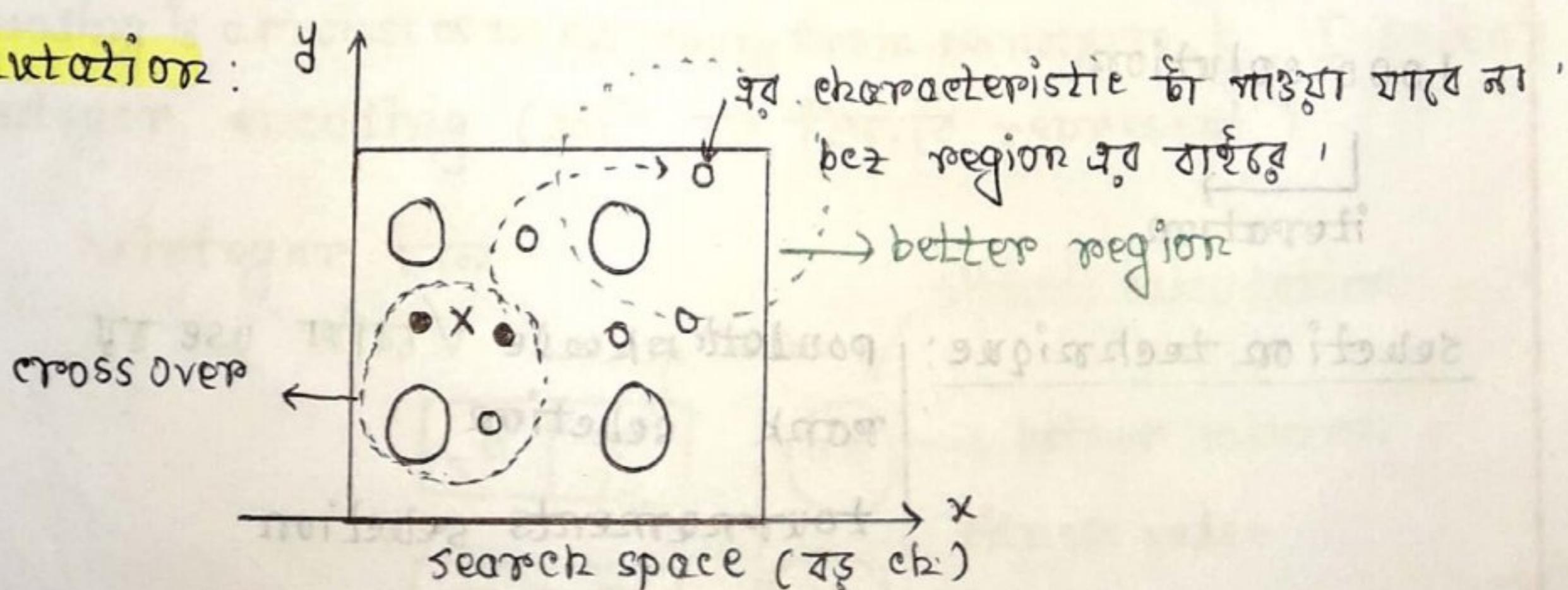
X

7   40	5   25
--------	--------



randomly [-50, 50] एवं अधिकार  
कोना इसी value पर

Mutation:



किसी गाँठवाले जेनेटिकों का शामिल होना। उसी push करा दें।

अन्य जेनेटिक जेनेटिकों का शामिल होना,

-5	-50
----	-----

solution

→ আবার fitness calculation

5	25
---	----

→ 30

30	1	2	4
----	---	---	---

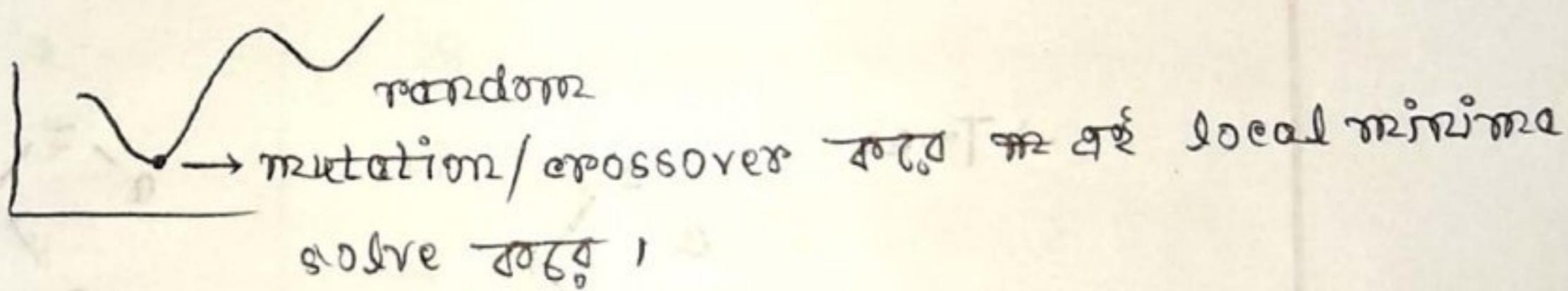
7	40
---	----

→ 47 better solution select করতে আসলে

better region এর দিকে যেতে আসব ।

Plain context / convex smooth

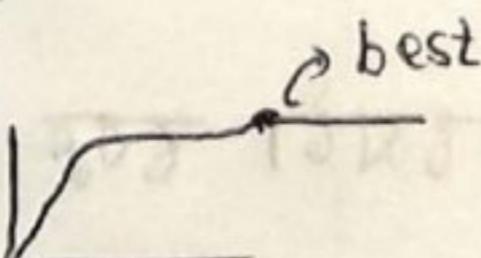
না হলে



কখন stop ?

- iteration no. fixed রাখে দেওয়া থেকে গালৈ ।

- some point



- average solution হালে

- performance হালে আসলে

calculation: Mutation (feasible change)

→ Binary 

1	0	1	0	0
---	---	---	---	---

 (bit flip)  
↑

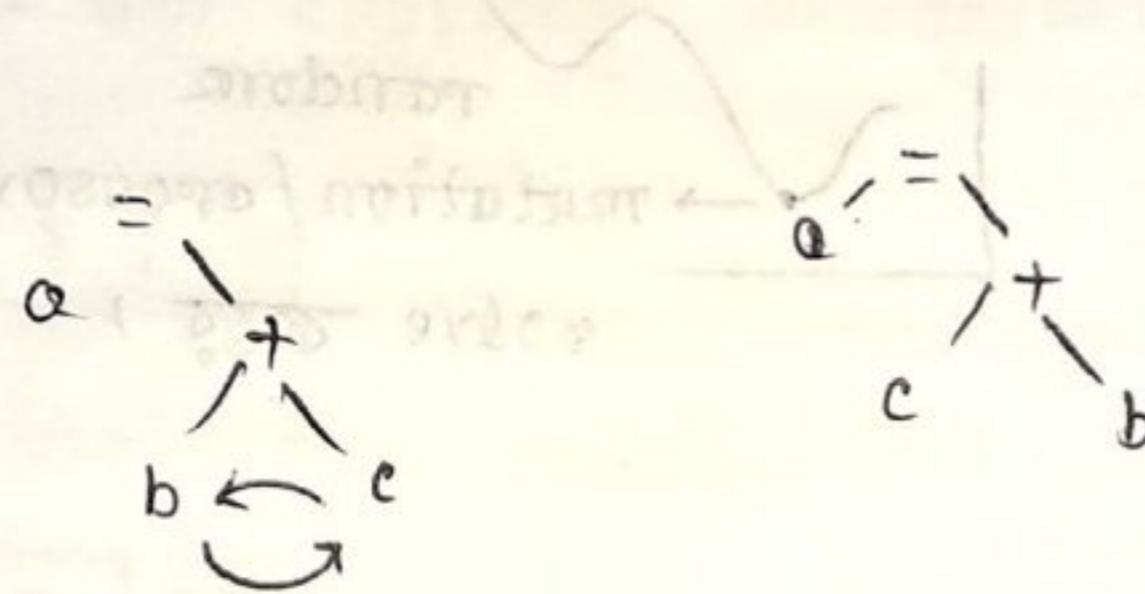
→ Integer 

172.	5	1000
------	---	------

 | replace by a value from  
| IP date amount | appropriate range.  
(अंतर्वर्षीय रेंज आवाहन)  
fitness & calculate कृदित हो।

→ zero to one probability to be fraud or  
not

→ Tree



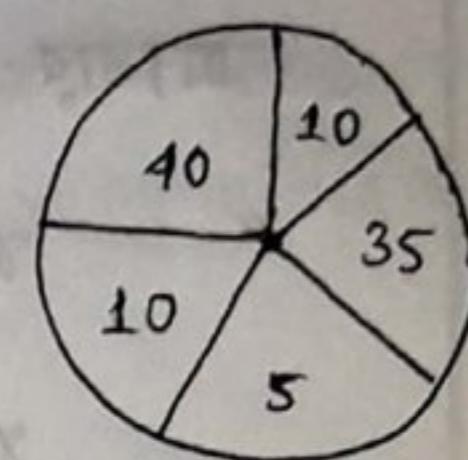
Fitness function कि हो? मेटा होम कृदितough.

convex Optimisation का गुणलिखित समाधान

Fitter chromosome or solution has a greater pie on the wheel and therefore a greater chance of landing in front of fixed point when the wheel is rotated.

### Roulette Wheel Selection

→ Generate sum(s) of all fitness



→ Pick a random number  $r$  from 0 to S

→ Select the chromosome i, for which the  $\frac{\text{sum of fitness}}{\uparrow \text{(from } i=0 \text{ to } N)}$  goes beyond  $r$ .

Chromosome	(sorted) Fitness
1	20
2	18
3	17
4	15
5	10
6	5
7	4
8	3
9	2
10	1

$$S =$$

0 থেকে S এর মধ্যে random no. select.

$$r = 50$$

fitness ত্যাগ করে দেখানে r থেকে র্তাৰ দেখাবলৈ stop

$$(20 + 18 + 17) > 50$$

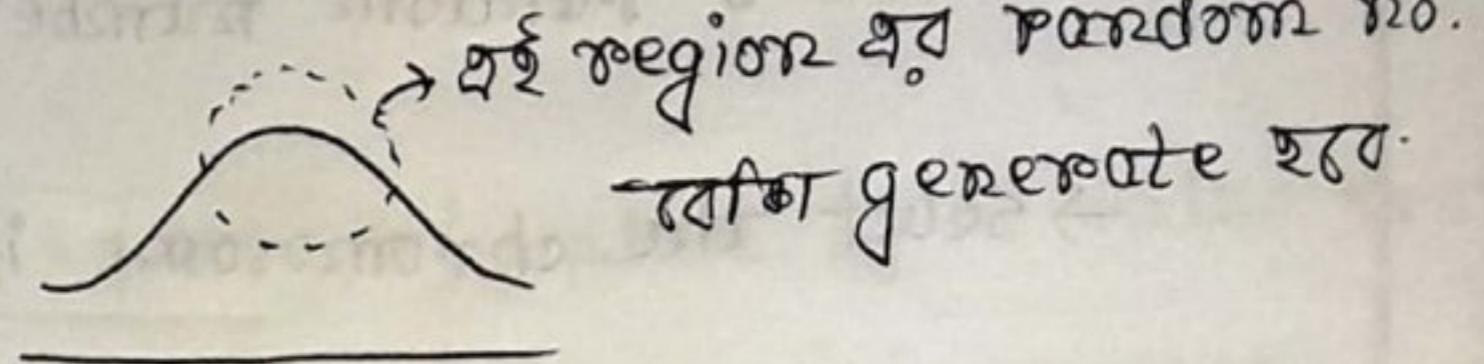
so,  $i=3$ , 3<sup>rd</sup> chromosome select হবে,

ଆଗାମ୍ ରାନ୍ଡମ୍ ରୁ କାହାରେ କମ୍ ପ୍ରସ୍ତ୍ରେ କାହାରେ

$R=30$  ହାଲେ  $i=2$  କମ୍ ହେବେ

$R=10$  ହାଲେ  $i=1$  " "

random number



ଯେ ହାଲେ କେବଳମାତ୍ର କମ୍

fitness ମୁଣ୍ଡୋ କମ୍ ହେବେ, ନାମରୁ ପ୍ରଥମ ଦିକ୍ ଥେବେଇ

select ହୁଏ କାହାରେ ଏହାକିମ୍ବଳୁର ଫିଟନ୍ସ ଅନେକ ରେଖା

fit evolution ଥିଲେ ବିନା କାହାରେ ଦିଲେ,

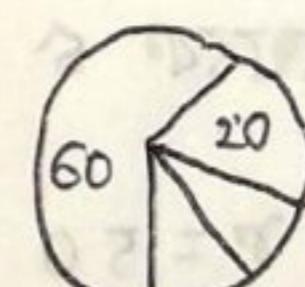
diversity ଥିଲେ ଜଣ୍ଯ ଘବଦିଲେ ବର୍କୁରୁ କାହାରେ ହେବେ,

ଫିକ୍ଟୁରୁ ଫିଟନ୍ସ ଅନେକ ରେଖା, ଫିକ୍ଟୁରୁ ଅନେକ କମ୍, bias solution

ଆଗରେ

chromosome	fitness
1	60
2	40
3	38
4	5
5	4

abnormal variation (diff କମ୍ ଆବଶ୍ୟକ)



ଏହି ପ୍ରବ୍ଲେମ ଥିଲୁ ଜଣ୍ଯ rank selection ବେ ହେବେ,

## Rank Selection:

fitness consider না করে chromosome এর rank consider করা হ'ল  
mostly used when the individuals in the population have very close fitness values

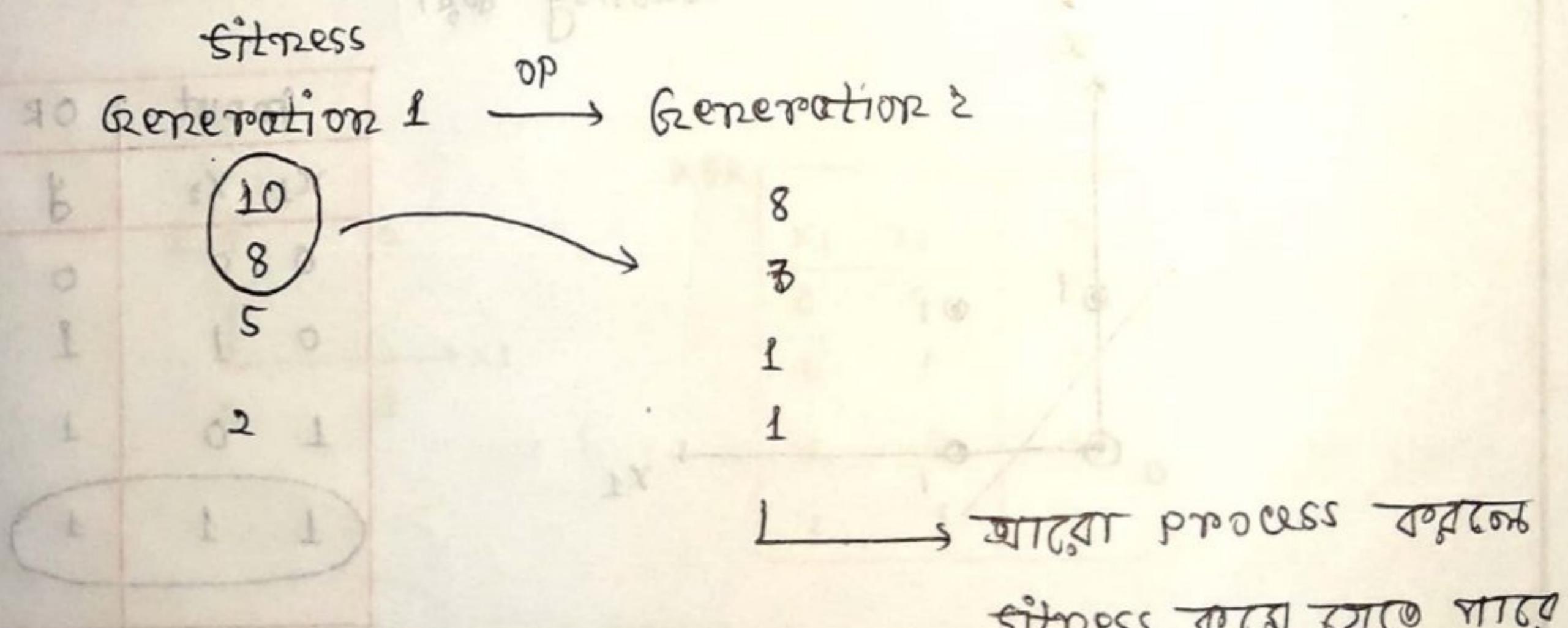
chromosome	Fitness
1	10
2	9
3	8
4	7
5	6

শীঘ্ৰ last শ্ৰেণীৰ সেকেন্দ্ৰীয়ালো select কুৱাবৰু প্ৰৱৰ্তন দেওয়া যাব'ল  
negative fitness value নিয়েও কাজ কৰতে গালৈ।

Tournament Selection: In K-way tournament, we select K solutions at random.

- Pick best 2 from random 3 and cross over  
(যাবৰু সিংগুলের দফতা)

## Elite

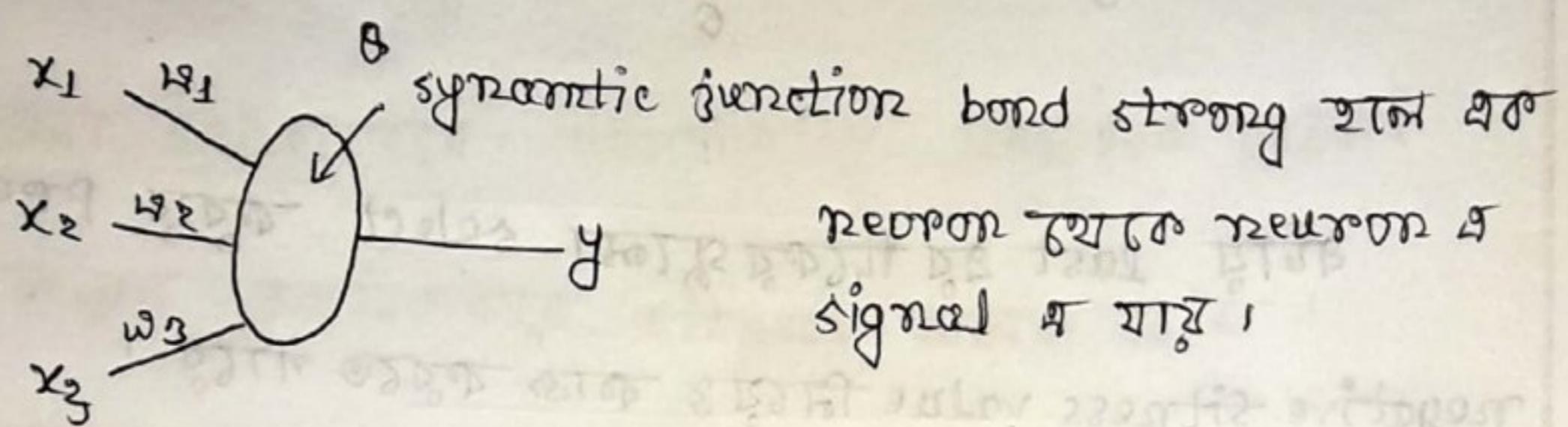


একজুড়ে আগের gen থেকে better নোট চোসে।

Techniques of  $\square$  selection

$\square$  crossover

## \* Supervised Learning (NN)

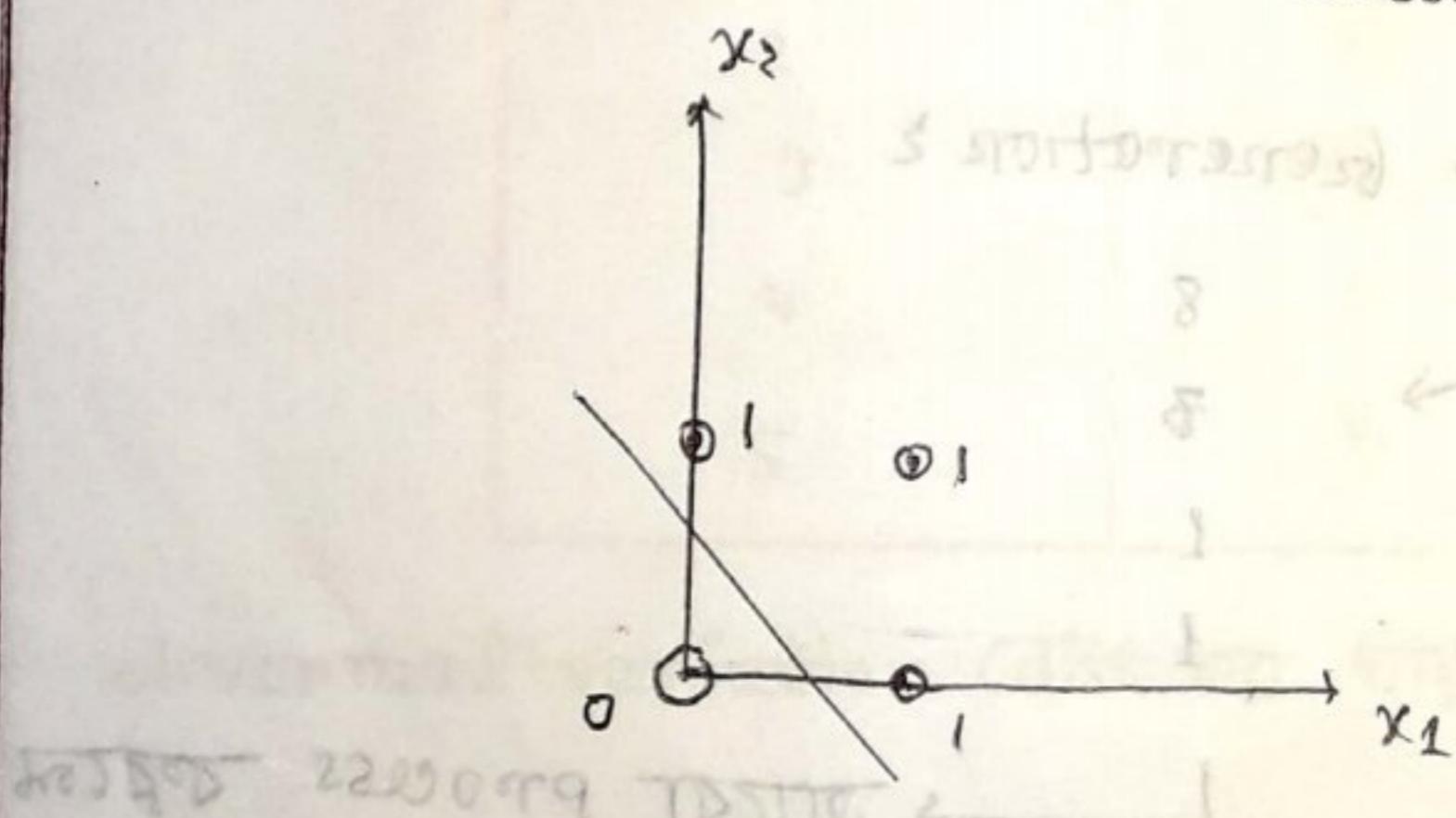


real life input

input vector  
 $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$   $y$  (label)

Linearly separable problem (Line द्वारा अलगा करना)

classify करना



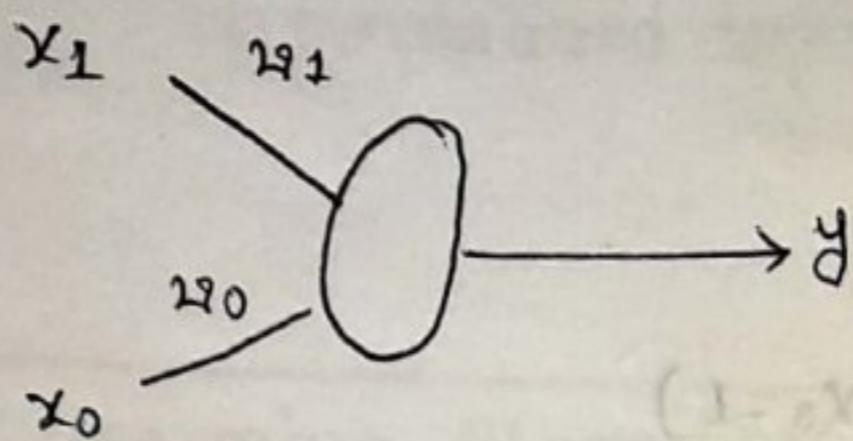
Input	OR
0 0	0
0 1	1
1 0	1
1 1	1

OR is a linearly separable problem.

$$y = mx + c$$

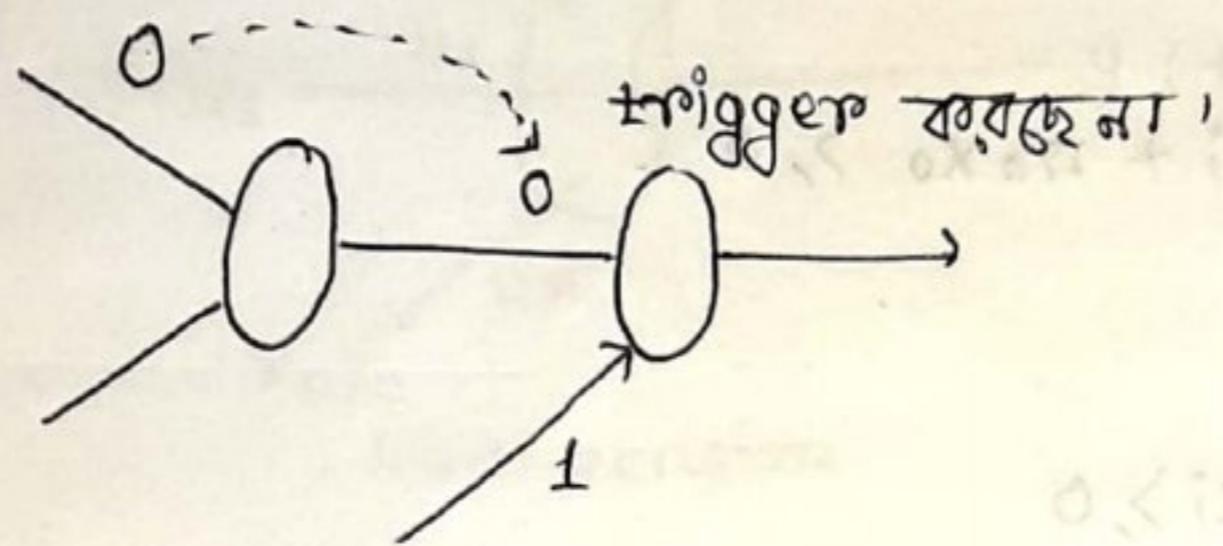
$$y = w_1 x_1 + w_0 \cdot 1$$

$$= w_1 x_1 + w_0 \cdot 1, \text{ where } x_0 = 1$$



threshold এর টাণ্ডে গোলে বাই  
বায়বে।

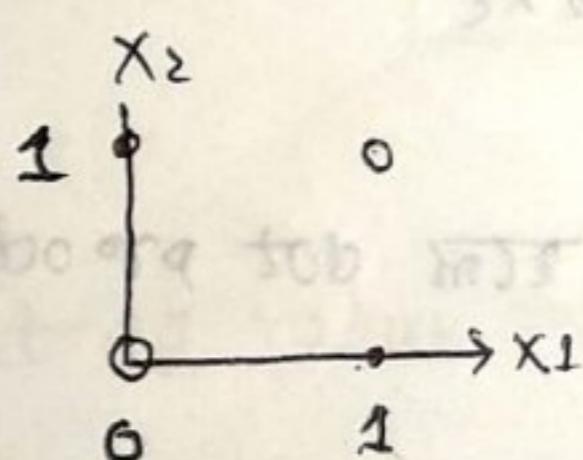
$$y = \sum w_i x_i \quad (\text{Line})$$



$$y = \sum w_i x_i > 0$$

$$f(w, x) = \sum w_i x_i$$

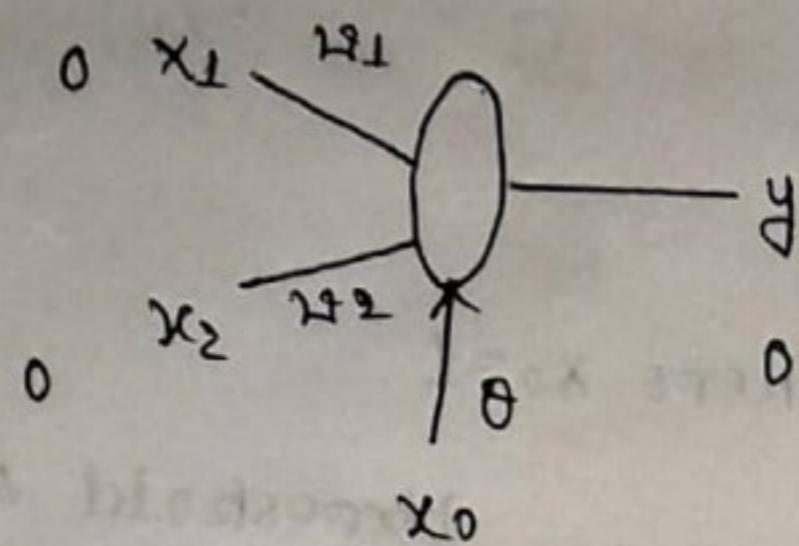
$$y = \begin{cases} 1 & f(w, x) \geq 0 \\ 0 & f(w, x) < 0 \end{cases}$$



XOR

$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	0

Linearly separable NT!



$$\sum_{i=1}^2 w_i x_i + w_0 x_0 > 0$$

$\geq w_0 x_0 \quad (x_0 = 1)$

$$\Rightarrow \sum_{i=0}^2 w_i x_i - w_0 x_0 > 0$$

$$\Rightarrow \sum_{i=1}^2 w_i x_i + w_0 x_0 > 0$$

$$\Rightarrow \sum_{i=0}^2 w_i x_i > 0$$

↳ input fixed

$$\sum \vec{w} \cdot \vec{x} = [w_0 \ w_1 \ w_2] \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

$$= w_0 x_0 + w_1 x_1 + w_2 x_2$$

ନେଟ୍ ଏବଂ ଡିପ୍ଲୋଯ୍ ରେ କୌଣସି ଦିଆଯାଇଛି

କହୁବୁ ମାତ୍ର ନା, ତାହିଁ ଟାଙ୍କେ କହୁବୁ ଶ୍ରୀ

$w^T$

The above diagram

bias

$$w_0x_0 + w_1x_1 + w_2x_2 \quad [x \text{ জানা}]$$

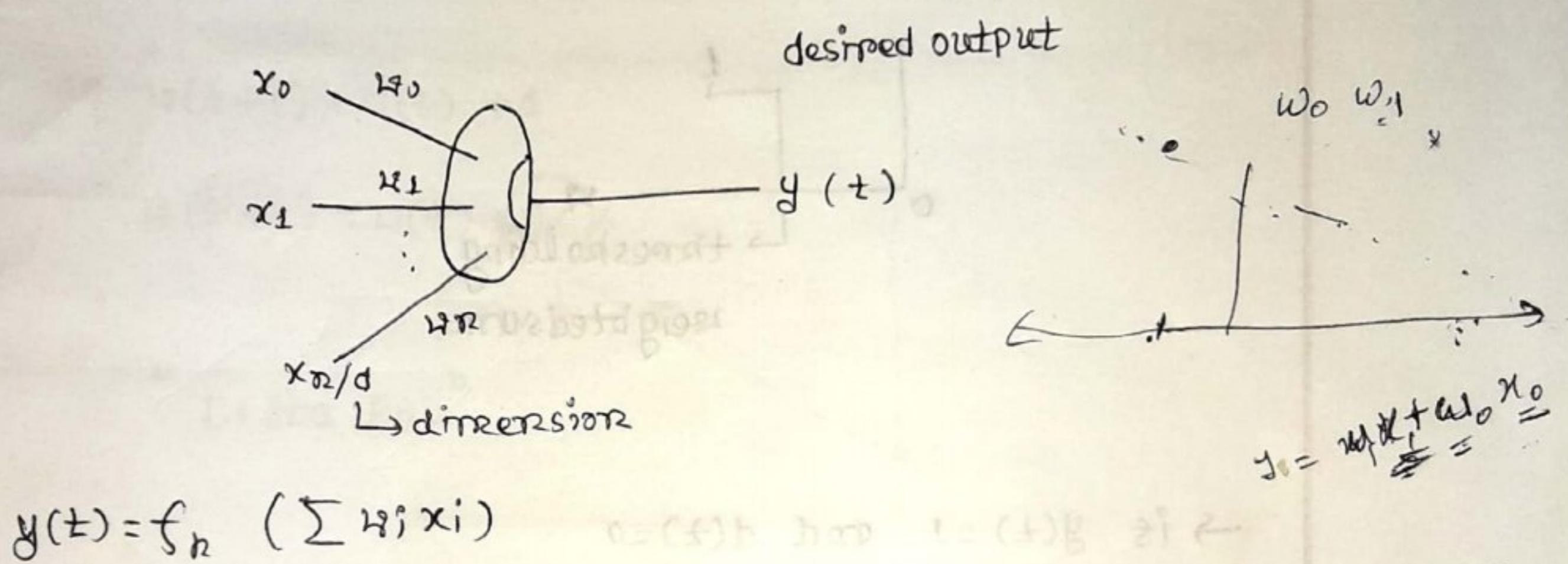
$$( -L, L )$$

fraction & consider ক্ষেত্রে হবে

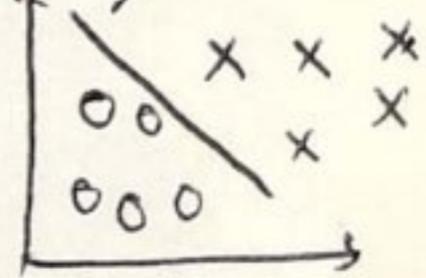
normalized ক্ষেত্রে হবে, তখন রাখতে infinite

SE

### Perception Learning algorithm:



Linearly separable problem



Weight এর value randomly initialize

→ Initialization  $w \in [0,1]$

Place  $x_0, \dots, x_n$  and

→ compute  $y(t) = f(\sum w_i x_i)$

$x_0$  এর মাঝে  $y_0(t)$  ) পাইবে কোন ?

variation থাকলে weight change

Rosenblatt

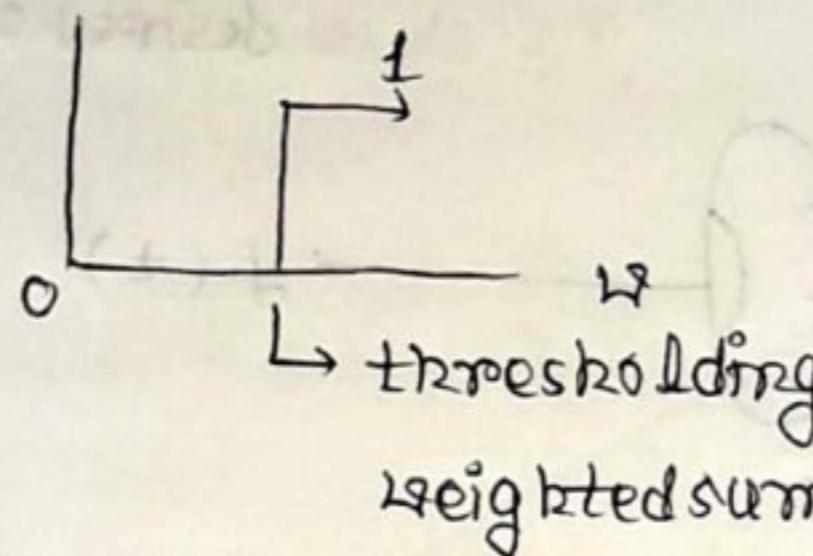
$s_h \rightarrow$  hard limiting threshold

→ update weight, if  $y(t)$  and  $d(t)$  are same.

$$w(t+1) = w(t)$$

→ if  $y(t) = 0$  and  $d(t) = 1$   
    └ target

$$w(t+1) = w(t) + x_i(t)$$



→ if  $y(t) = 1$  and  $d(t) = 0$

$$w(t+1) = w(t) - x_i(t)$$

└ iteration

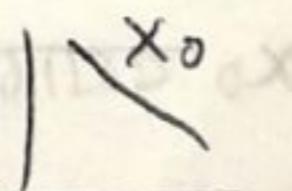
adaptive change

$$w(t+1) = w(t) - \eta x_i(t)$$

└  $\eta \in [0, 1]$ , control parameter



single unit processing



- - a. Iteration by iteration adapt - -

$$\omega(t+1) = \omega(t) + \text{change}$$

$$d(t) - y(t) = [0, 1] \quad [\text{hard limiting threshold}]$$

$$d(t) - y(t) = [0, 1]$$

$\downarrow$  fraction error       $\downarrow$  error fraction

$$\Delta = d(t) - y(t)$$

$$\text{so, } \omega(t+1) = \omega(t) + \Delta$$

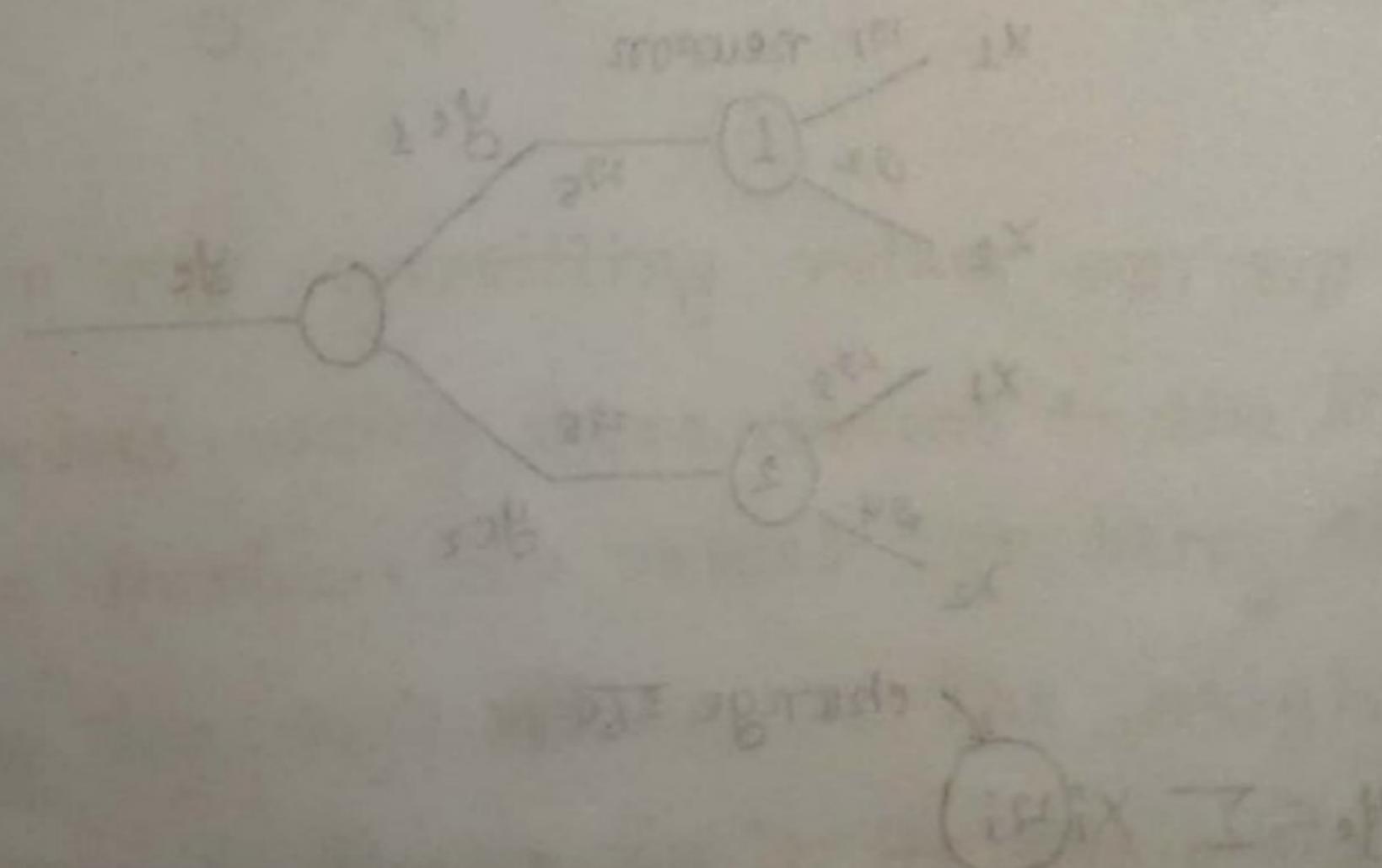
$$\omega(t+1) = \omega(t) + (\eta \Delta) x$$

$\downarrow$

"Delta Rule"

Batch Processing: അനേകം നേരുകളായി മാറ്റുക

gradient descent.



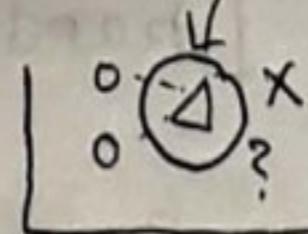
7 D

unknown pattern

\* Rogue pattern : ये sample गुणात्मक classify करते confusion

\* MLP

unknown pattern, majority voting & circle



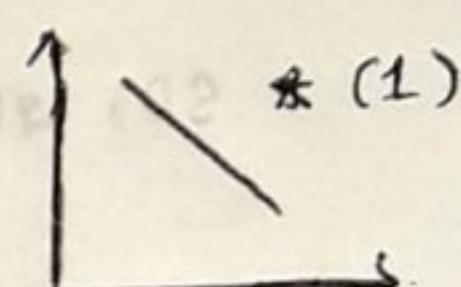
KNN

आजछे यदि एक actually 'x'.

### Multilayer Perceptron :

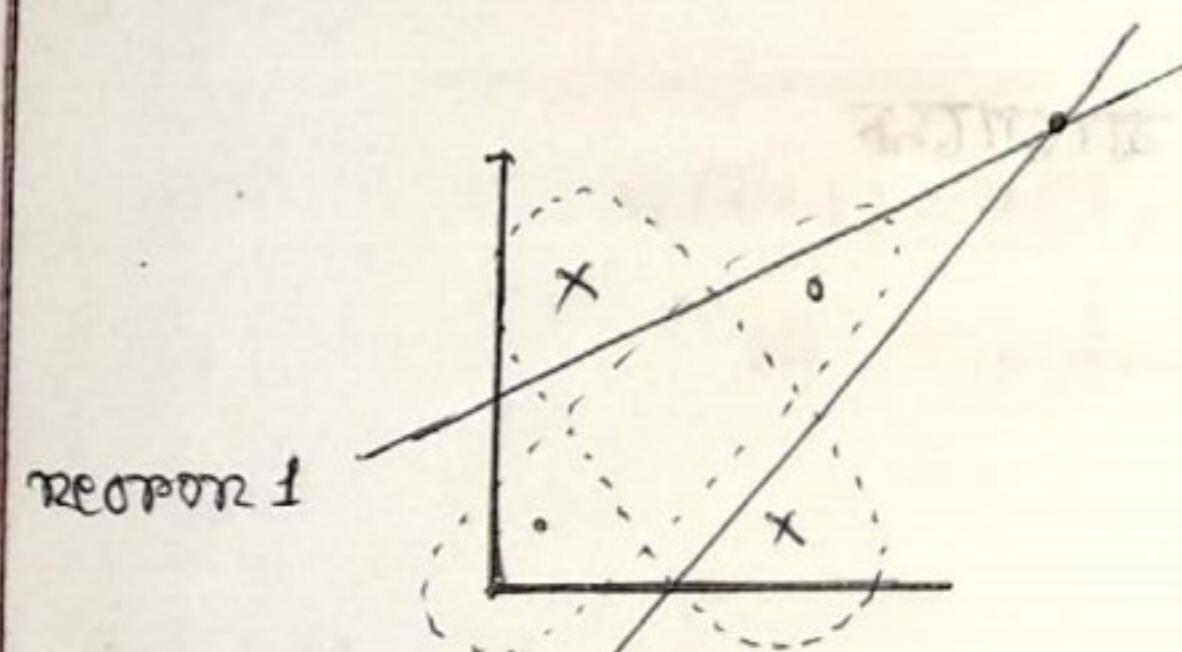
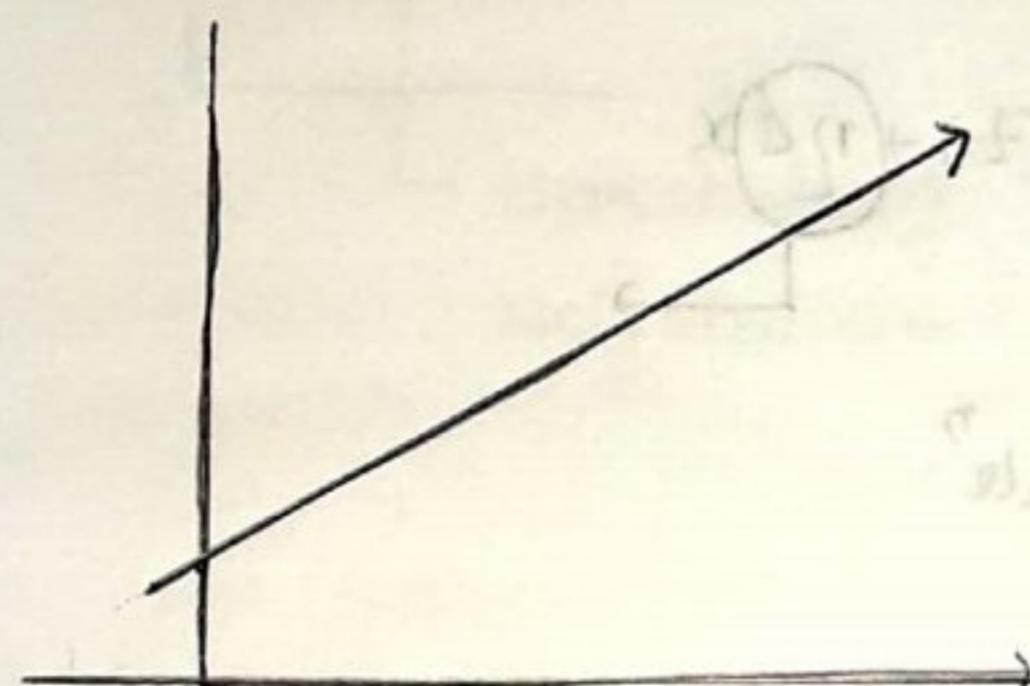
Nonlinearly separable problems का classify करते

OR



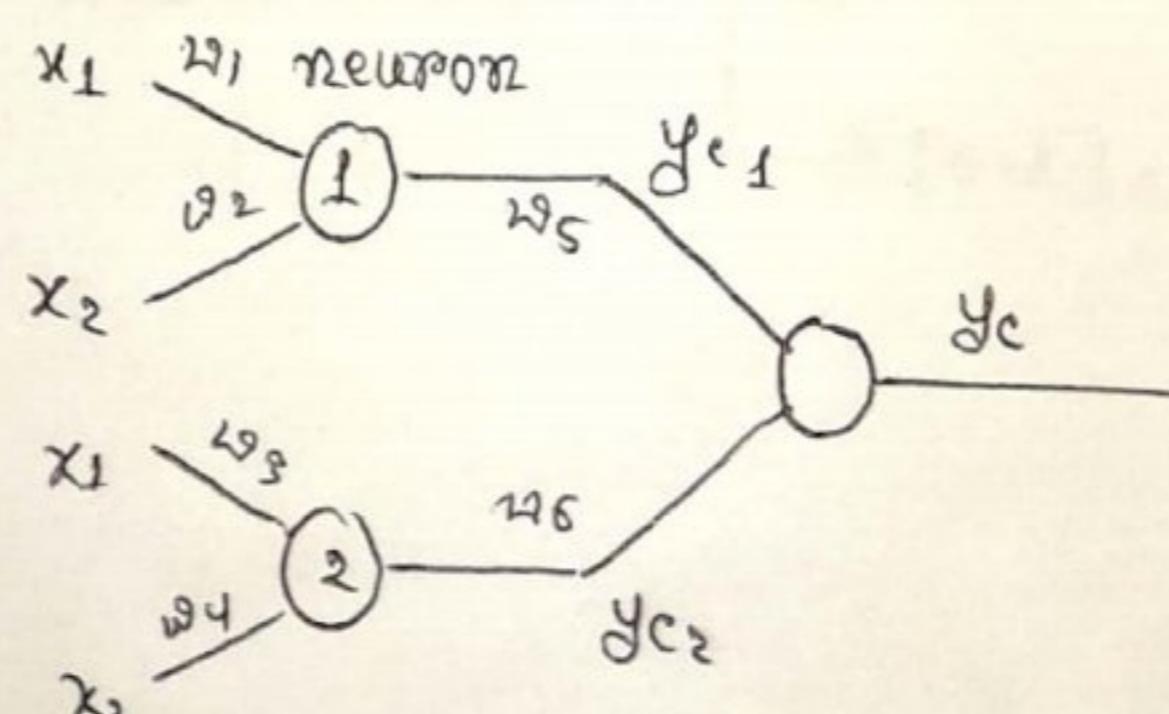
$$y_c = \sum w_j x_j$$

classification line



nonlinear के अनेकों लाई ए  
उसी कानून याएँ।

neuron 2



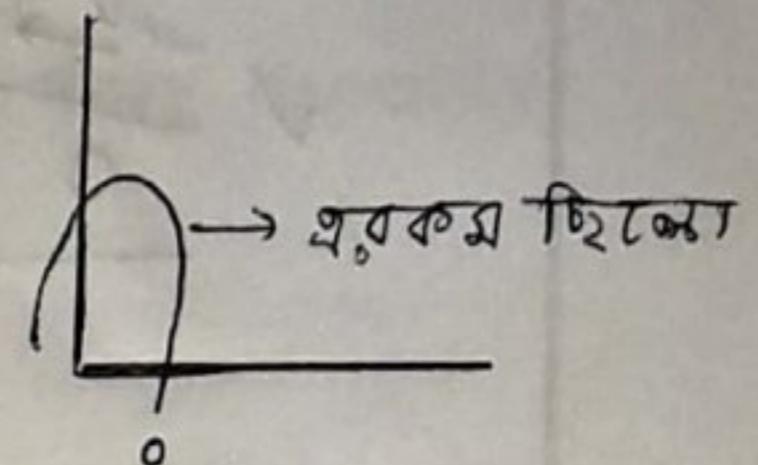
change  $w_{ij}$

$$y_c = \sum x_i w_{ij}$$

generalization power: এর নতুন pattern কে কর্তৃতা successfully

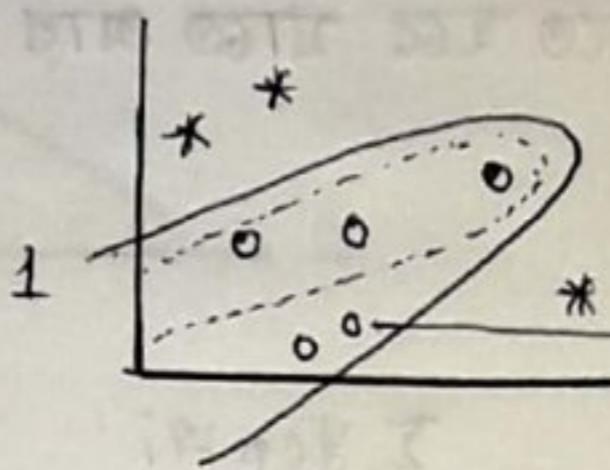
$$w_{t+1} = w_t \pm \Delta w$$

যেখানে কর্তৃত হবে



$w_i$  কে initialize  $[-1, 1]$  করে suppose

✓ after tuning  $\rightarrow$

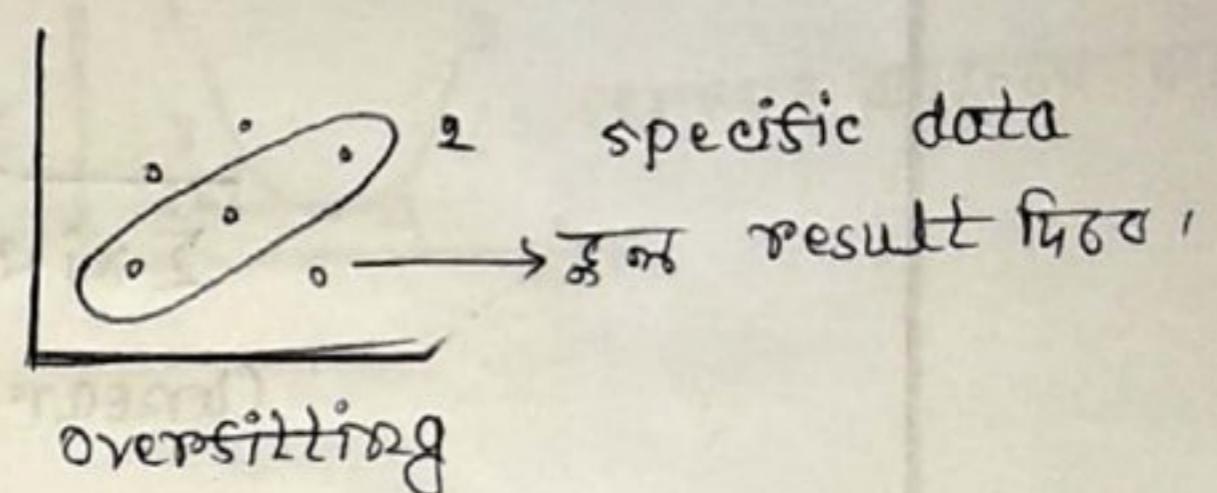


variation margin

unknown sample

generalized

অনেক বেশি tuning করার পর



specific data

result ফল

overshooting

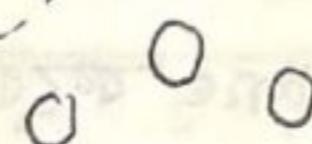
1 no figure ই unknown sample কে better classify করতে পারে।

শীঘ্র generalization power দেবি।

iii catastrophic Forgetting: মনোদ্রিকের data ছেন যায়।

Overshooting এ earlier sample থেকে ছেন যাবে।

⊗⊗⊗⊗⊗... overshooting.



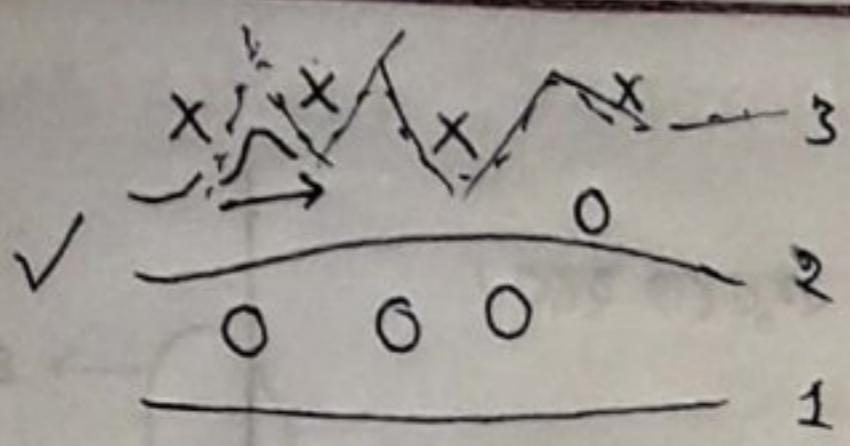
Regularization দ্বারা overshooting reduce করা যায়।

↳ involves adding extra element to the loss function,

which punishes our model for being too complex or using too high values in the weight matrix.

This may try to limit its flexibility.

K-fold  
validation



weight এর টিকার নিয়ে গুলি  
depend করে।

weight কে stop করে দিতে হবে যাতে আর প্রয়োগ না হয়।

(linear)

$$\sum x_i w_i$$

$$\sum y_i w_i$$

$$\sum x_i^2 w_i$$

(linear)

Line কে appropriate position এ নিতে

→ supervised learning

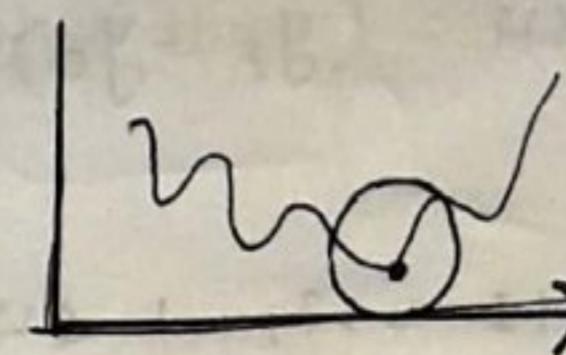
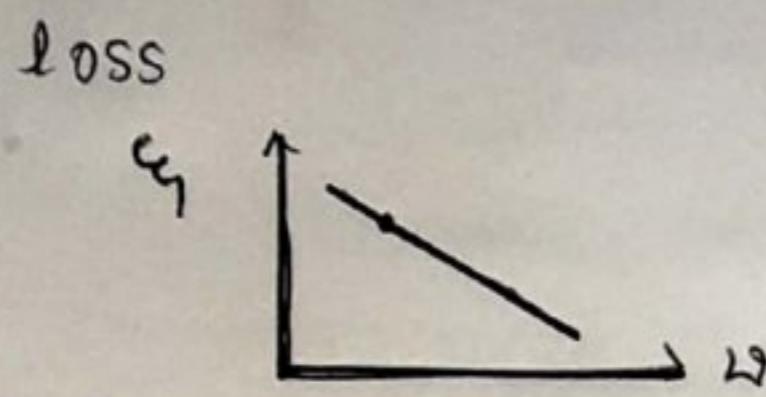
Given	Computed	
$y_t$	$y_c$	Error
1	1	0
2	1	1
1	2	1
2	2	1

③ Line করে একজন কাজানো

$$(y_t - y_c)^2$$

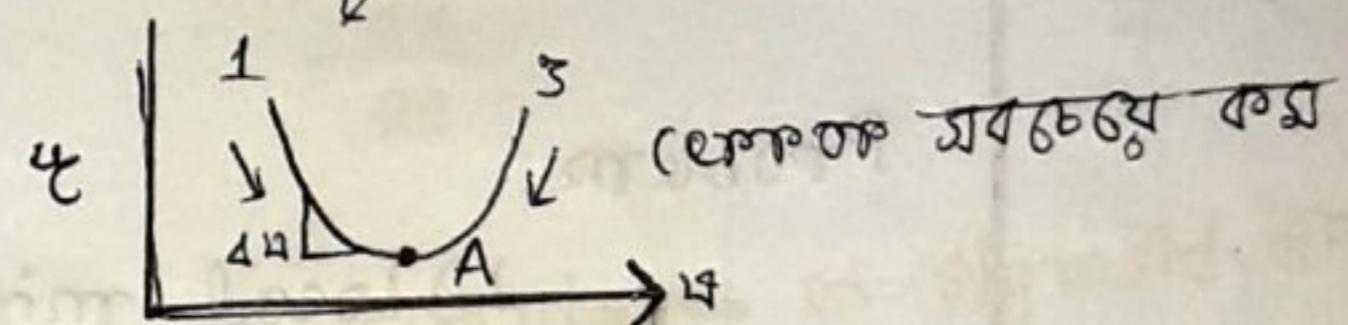
$$\xi(\hat{y}) = (\hat{y}_t - y_c)^2$$

L  $\rightarrow x_i$   $\circlearrowright \rightarrow$  weight এর টাঙ্গা depend.



$$w_{t+1} = w_t - \Delta w$$

L  $\rightarrow$  stop হিয়েবে  
চিন্তা বাধতে হবে।

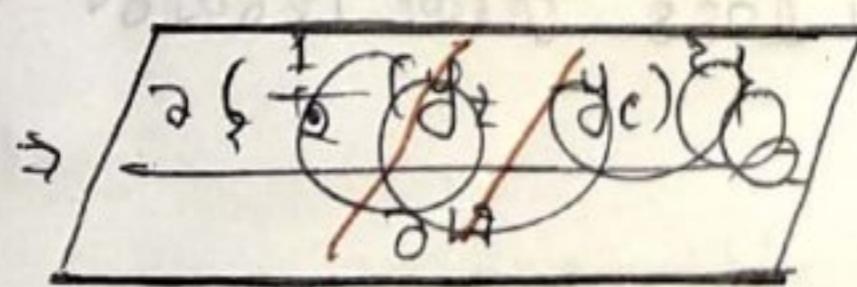


$$\Delta w = \frac{d\xi}{dw} = \frac{\partial \xi}{\partial w}$$

L  $\rightarrow$  partial (multiple function এর ব্যুৎপন্ন)

$$\xi(\hat{y}) = \frac{1}{2}(\hat{y}_t - y_c)^2$$

$$\frac{\partial \xi}{\partial w} = \frac{\partial \xi}{\partial y_c} \cdot \frac{\partial y_c}{\partial w}$$



$$\frac{\partial}{\partial w} = \frac{\partial \left( \frac{1}{2} (\hat{y}_t - y_c)^2 \right)}{\partial y_c} \cdot \frac{\partial y_c}{\partial w}$$

$$\Rightarrow \Delta w = -1 (\hat{y}_t - y_c) \cdot x_i$$

$$= (\hat{y}_t - y_c) x$$

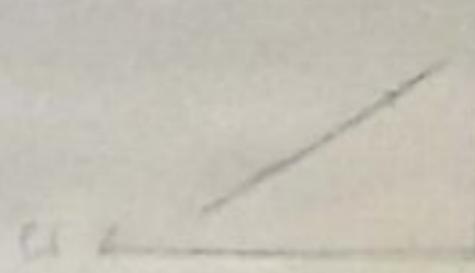
delta rule

## Parameter Optimization:

Gradient descent ଏହି ପାର୍ମିଟରର loss function କେ ମିନିମିଜେ  
 ↳ ବନ୍ଦରେ

$$\text{so } \Delta w = (y_t - y_c) x$$

$$w_{t+1} = w_t + \Delta w$$



### Problems:

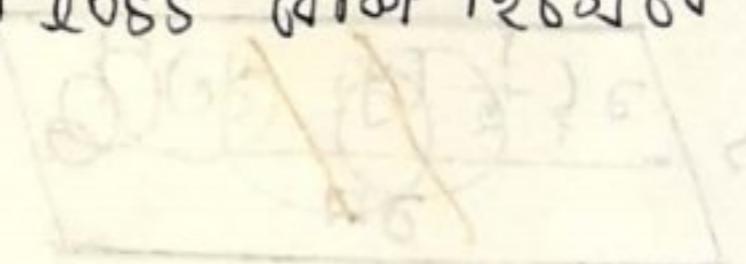
1) Local minima କେ ଟ୍ରାପ୍‌ଡ୍ ହେଲେ ଗାଠେ  
 loss ଏହି ରୁଳ୍ୟ ଦେଖିଲେ ୧୮୦୦,

2) Overtfitting କିମ୍ବା କୁଣ୍ଡ ହୁଏନି,

ଯେ ଟିକ୍ ପ୍ରଦିତ କୁଣ୍ଡରେ loss function ଏହି ରୁଳ୍ୟ କାହା ହେବେ ?

ସାଥେ ଏହି calculated output. ଏହି ଏହି ୦ ଥିଲେ ୧ ଏହି ଝାକେ ହୈବେ ,

ଆଶନେ ଏହି log value negative ହାହରେ , loss ଦେଖି ଶିଖିବେ ଦେଶାତେ  
 ଏକଟର ଛୁଟେ ନାହାର୍ଟିଂ ହିଟିଙ୍ କରିବେ ,

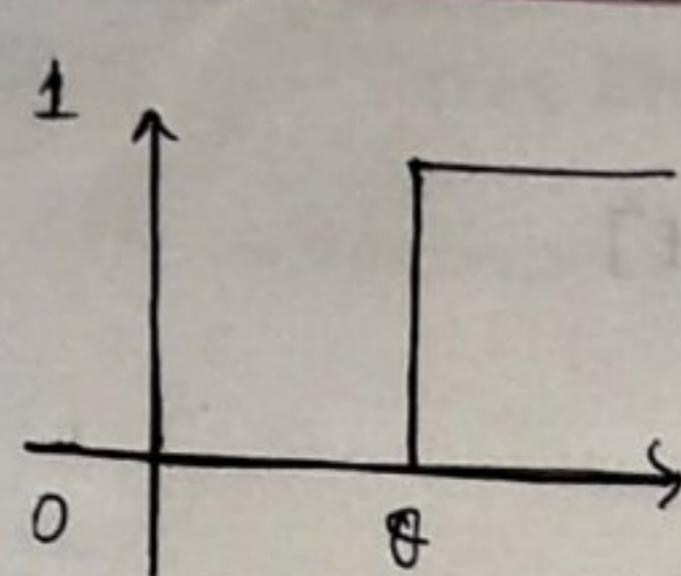


$$\frac{(x - 0)^6 - 0^6}{0^6 - 0^6} = \frac{(x - 0)^6}{0^6} = \frac{x^6}{0^6}$$

$$x^6 - (0^6 - 0^6) = x^6 - 0^6 = x^6$$

$$x(3B - 1) =$$

7E

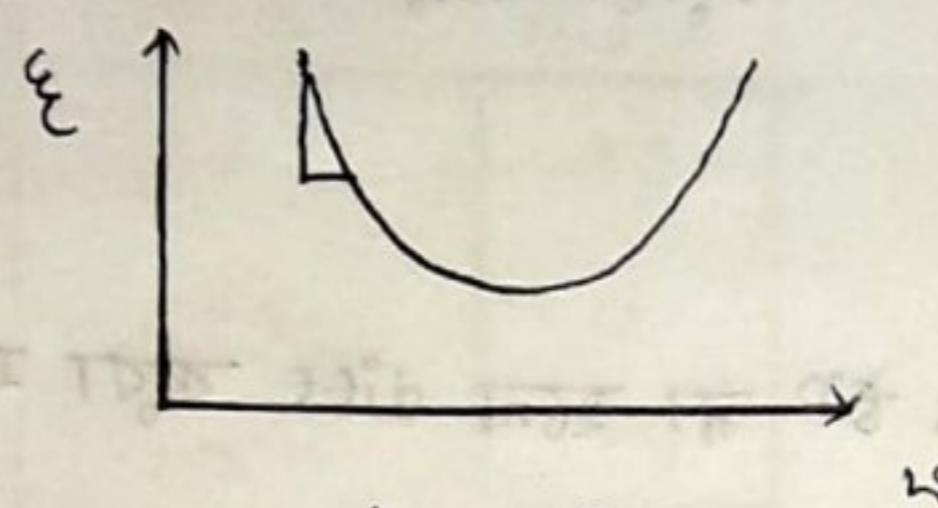


$$y_c = f(x)$$

$$\sum x_i w_i > \theta \rightarrow 1$$

$$\sum x_i w_i < \theta \rightarrow 0$$

hard limiting threshold.

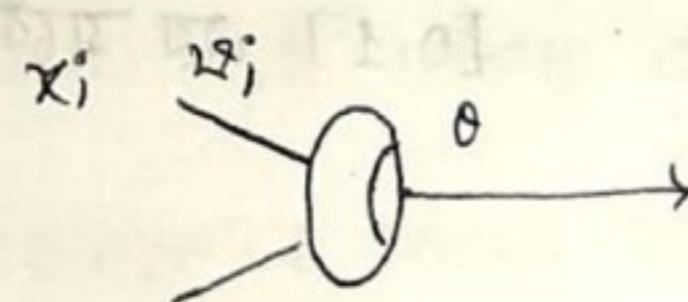


$$\frac{\partial E}{\partial \theta}$$

Advantages of sigmoid fn:

1) similar to step function,  
output is between 0 and 1.  
The curve crosses 0.5 at  $\hat{z} = 0$ ,  
which we can set up rules for  
activation function. 05

2) It doesn't have a zero on  
its curve, and have simple  
derivative.



Loss function

$\rightarrow$  uncertainty

Cross entropy loss:  $-t_i \log y_i - (1-t_i) \log (1-y_i)$

$$\frac{\partial E}{\partial \theta} = \frac{\partial E}{\partial y_i} \cdot \frac{\partial y_i}{\partial \theta} = \frac{\partial E}{\partial y_i} \cdot \frac{\partial z_i}{\partial \theta}$$

$$\therefore \frac{\partial E}{\partial y_i} = \frac{-y_i - t_i}{y_i(1-y_i)}$$

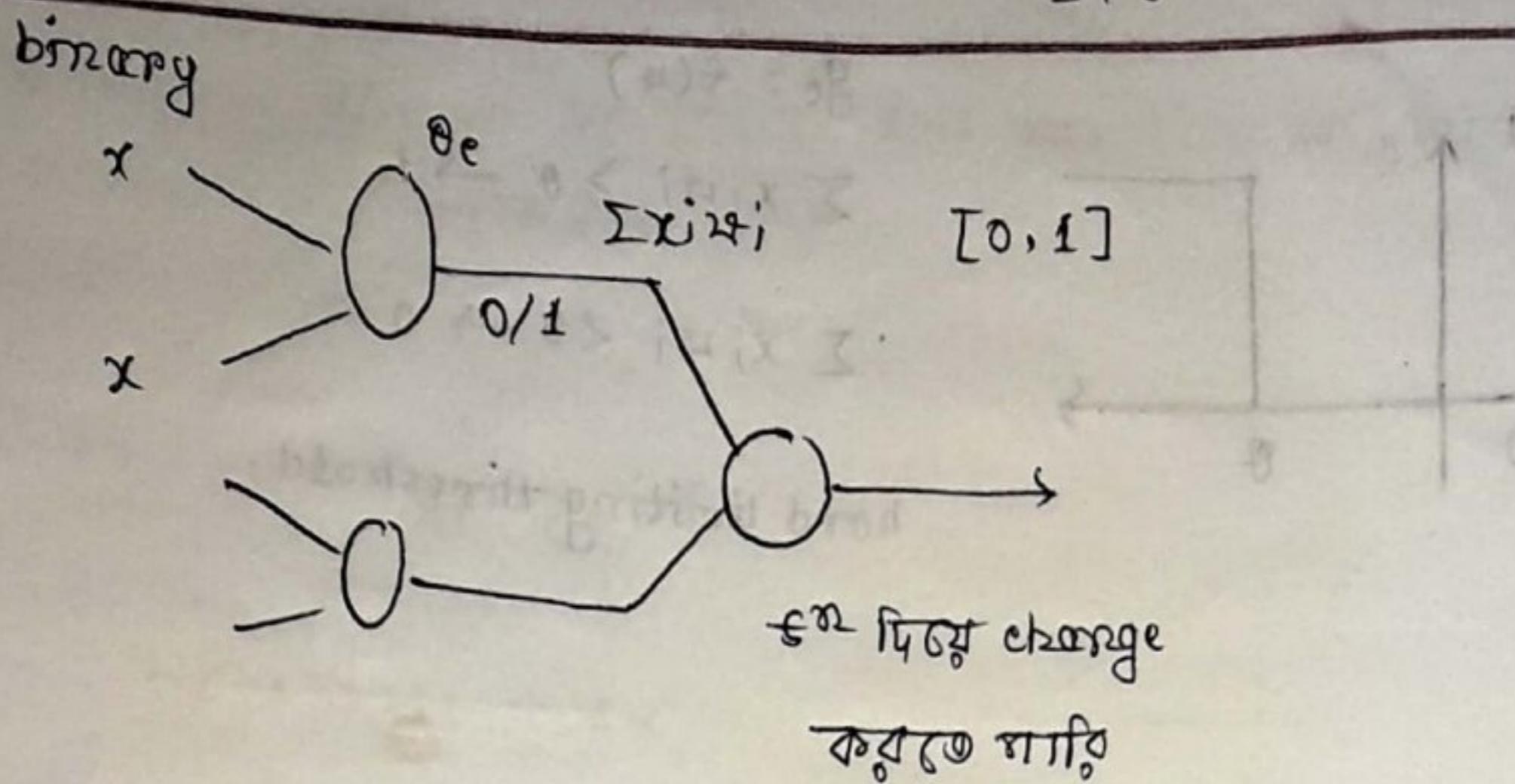
$$\therefore \frac{\partial z_i}{\partial \theta} = y_i(1-y_i)$$

$$\therefore \frac{\partial E}{\partial \theta} = \frac{\partial (x_i \cdot z_i)}{\partial \theta} = x_i$$

Step function also has no useful derivative (its derivatives is '0' everywhere or undetermined at the 0 point on x-axis)  
 We can't use step function to gradually change the weights, then it should not be choice.

$$z \rightarrow \sum x_i z_i$$

$$\text{sigmoid function } \sigma(z) = \frac{1}{1 + e^{-Kz}}$$



continuous एवं ना होने द्वारा फूटा शाय का!

$$\sum w_i x_i$$

$$\sigma(\sum x_i z_i)$$

continuous space में जिये आजबे!

[0,1] इस तरीके continuous

$$\frac{\partial E}{\partial \theta} = - \frac{(y_t - t)}{y(1-y)}$$

$$\frac{\partial y_c}{\partial z} = y(1-y)$$

$$\frac{\partial z}{\partial w} = x$$

using these three, we can get,

$$\frac{\partial E}{\partial w} = - \frac{(y_t - t)}{y(1-y)} \cdot y(1-y) \cdot x$$

$$= -(y_t - t)x$$

classification : cross Entropy (Binary)

Regression → continuous : Absolute Error, Mean square Err,

$$\frac{1}{n} |y_t - y_c|$$

$$\frac{1}{n} (y_t - y_c)^2$$

$$ME \frac{1}{n} (y_t - y_c) \text{ Mean Error}$$

$\hat{y}_t$	$y_c$	$\hat{y}_t - y_c$	mean error
1	.9	0.1	0.1
0.9	1.5	-0.6	0
0.2	0.3	-0.1	-0.1
		-0.4	0 → এর মান শোনা কঠিন আছে।

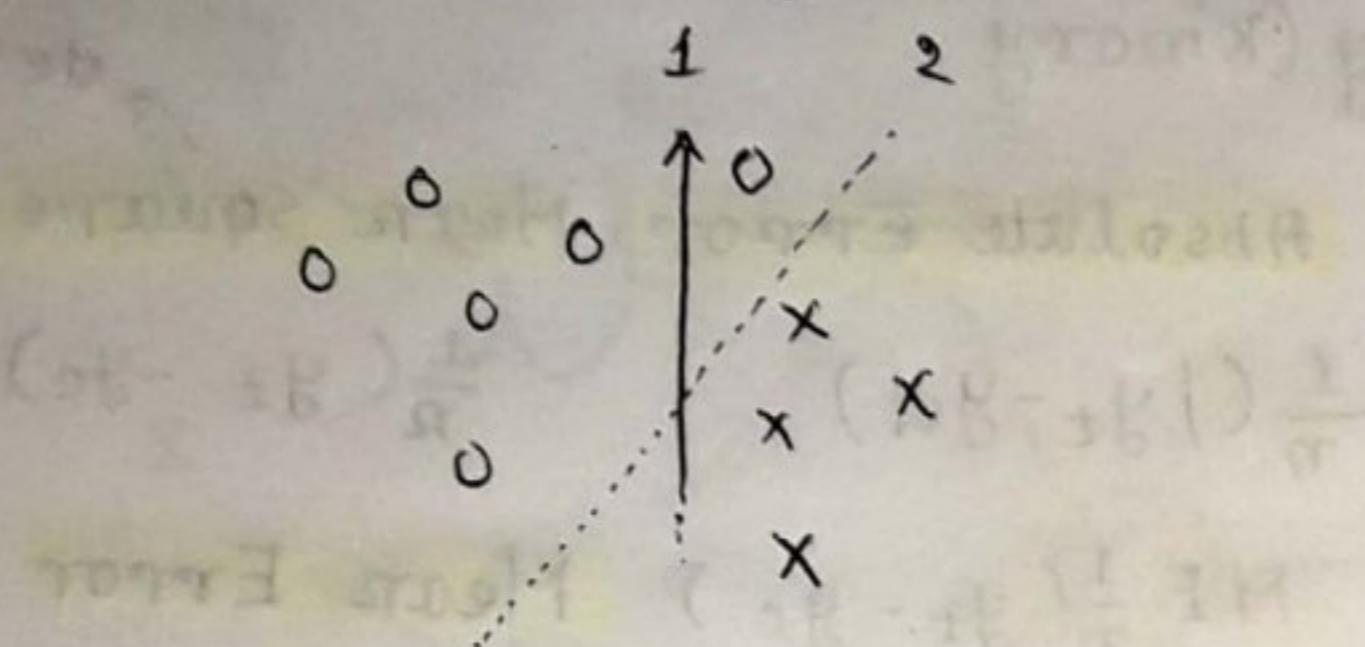
Mean error এর problem হলো একে negative value  
neutralise করে দিতে পারে, ফলে  
appropriately represent করতে পারে না।

MSE -তে এই problem নেই। কিন্তু এটা small value স্থলোকে বড় value'ত  
convert করে। (error কে focus করে)

$\hat{y}_t$	$y_c$	$(\hat{y}_t - y_c)^2$	$ y_t - y_c $
2	3	1	1
81	9	9	3
5	1	16	4
$\frac{1}{3} (26) = 8$		$\frac{1}{3} \times 8 = 2$	

Mean square Error: model is punished for making larger mistakes.  
larger mistakes results in more errors.

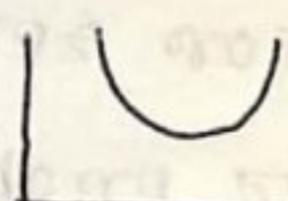
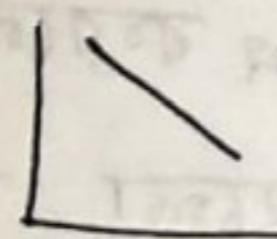
outlier निये correction ना करने तो चलते



→ Mean Absolute Error

outlier देने आवाजे MAE बेटर.

MSE error को magnify करते। फिर performance loss लगता है।



plot

gradient descent

custom Loss function (Tensorflow)

→ निजेके तथा define करते use करता।

$$\begin{array}{|c|c|c|c|} \hline & b_1 - b_2 & b_2 - b_3 & b_3 - b_1 \\ \hline 1 & e & e & e \\ \hline 2 & e & e & e \\ \hline 3 & e & e & e \\ \hline \end{array}$$
$$S = 8 \times \frac{1}{3}$$
  
$$\begin{array}{|c|c|c|c|} \hline & b_1 - b_2 & b_2 - b_3 & b_3 - b_1 \\ \hline 1 & e & e & e \\ \hline 2 & e & e & e \\ \hline 3 & e & e & e \\ \hline \end{array}$$
$$S = (2e) \frac{1}{3}$$

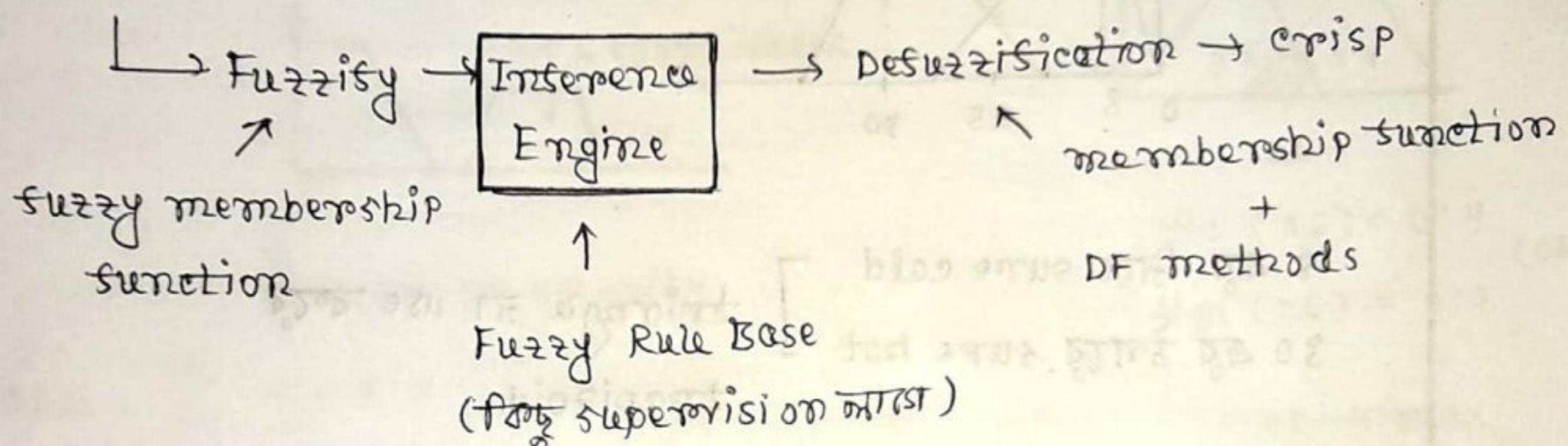
## Fuzzy Logic

[0,1] vs 0/1  
Fuzzy Crisp

↳ Embedded system, AC, motor control

Supervision করা, রাখা risky

**crisp** (discrete)



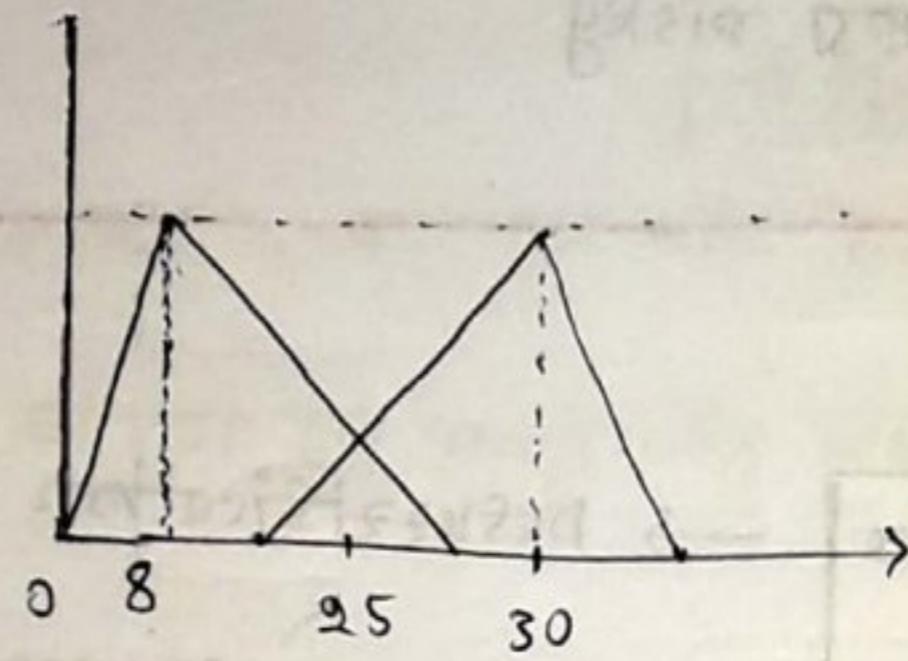
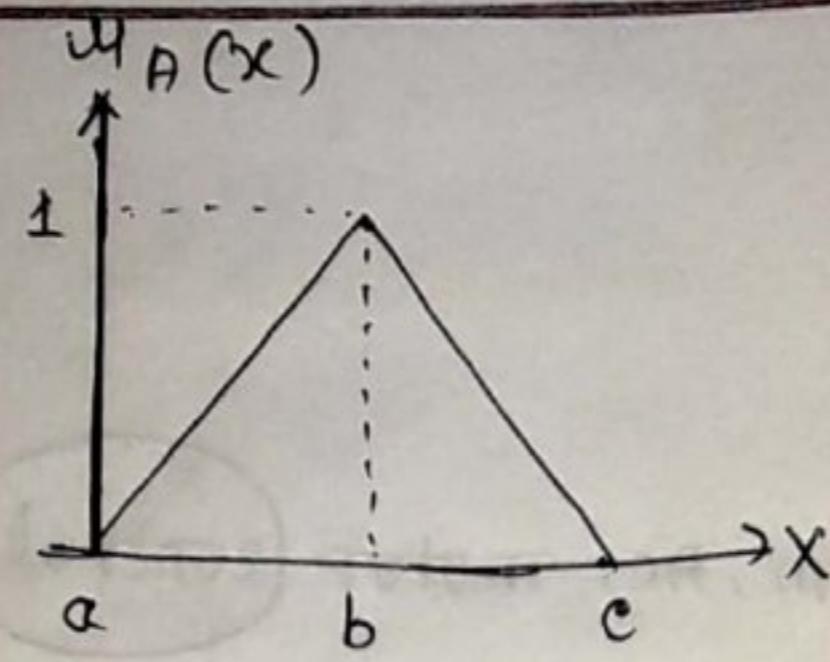
feeling	humidity	Target Temp
hot	high	→ °C

## Fuzzification:

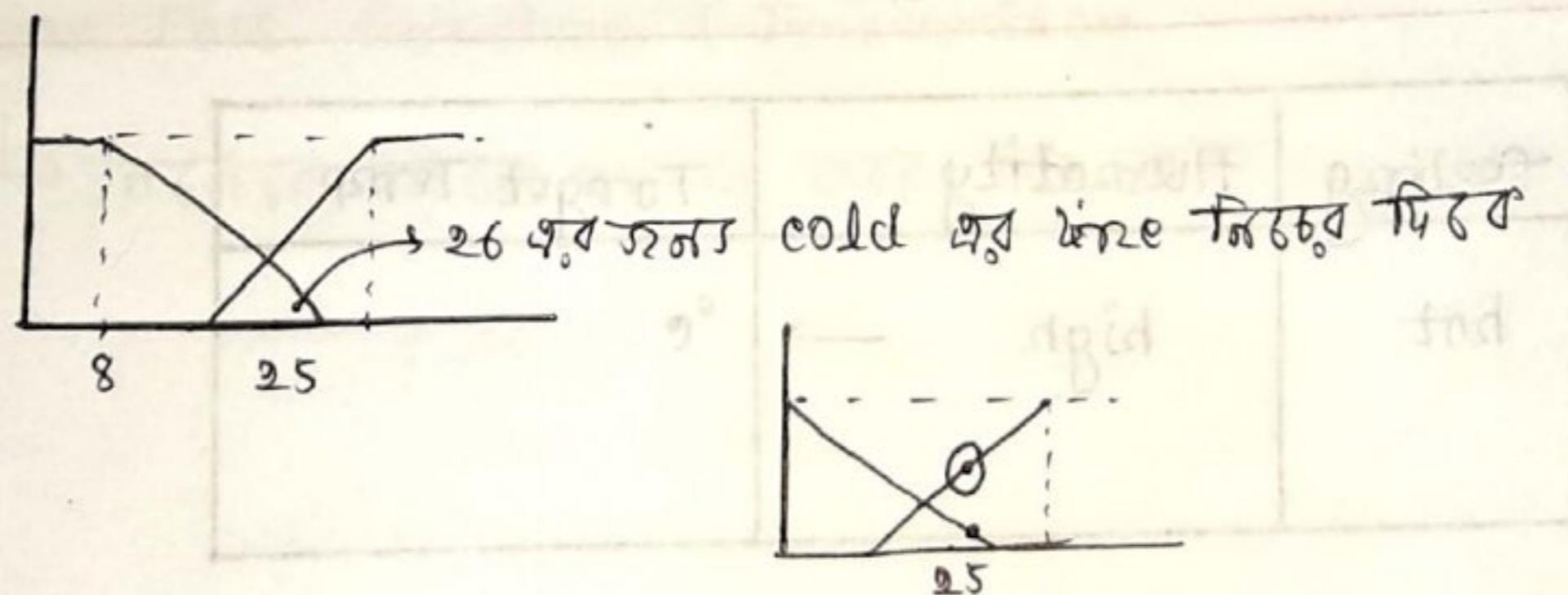
Membership function: ① Triangular

$$U_A(x) \rightarrow [0,1]$$

$$\begin{array}{l} \text{Feel} \\ \hline \text{Hot} \\ \text{cold} \end{array} \quad U_{\text{Feel}}(\text{Hot}) \rightarrow [0,1]$$



8 এর নিচে super cold  
 30 এর উপরে super hot ]  
 triangle কা use করে  
 trapezoid



functional value

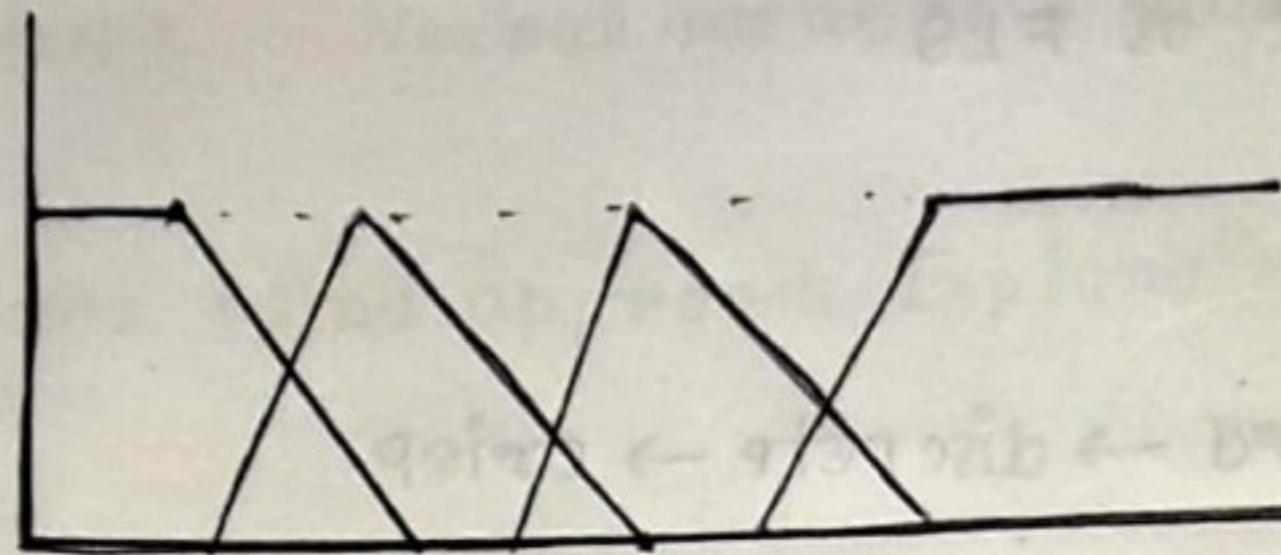
$$\frac{x-a}{a-c}$$

$$[x_0] \leftarrow (x)$$

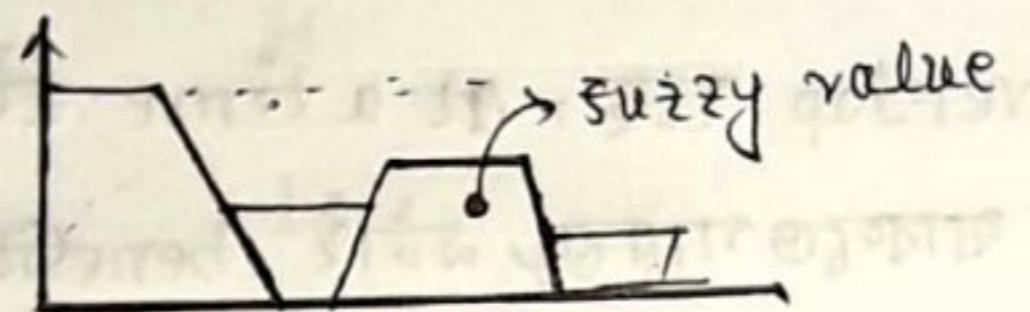
$$[x_0] \leftarrow (t_0)$$

real life experience/expert advice নিয়ে a, b, c fixed রোধ ইয়!

## Defuzzification

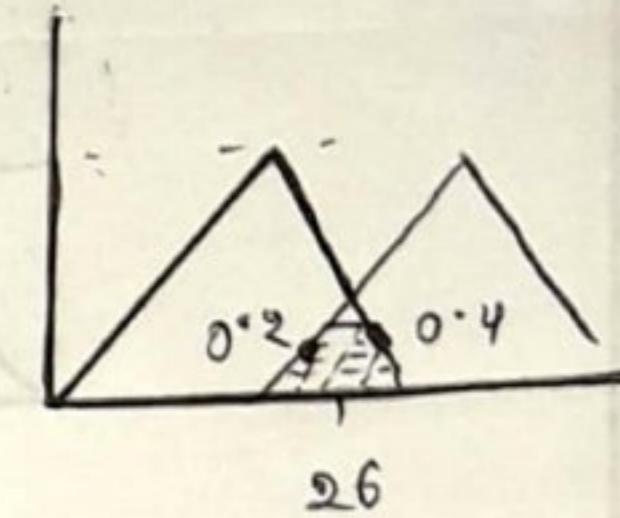


After clipping



center of gravity

$$COG_r = \frac{\sum \mu_A * x}{\sum \mu_A}$$



$$\mu_c(26) = 0.4 \quad (\text{OR})$$

$$\mu_{\text{OR}}(26) = 0.2$$

OR  $\rightarrow \max$

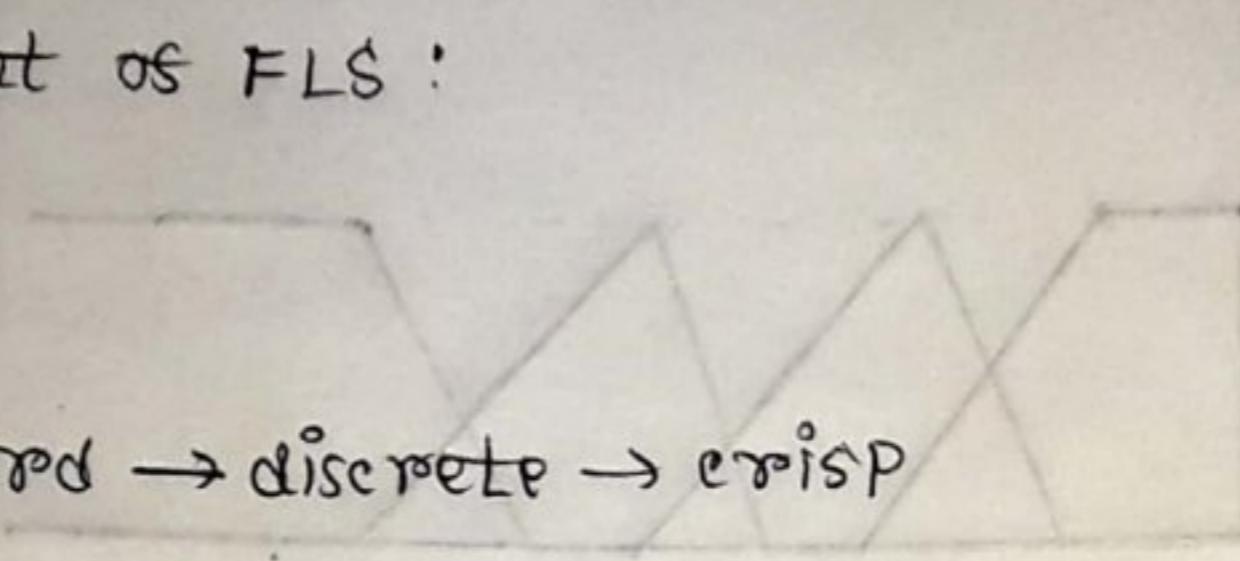
AND  $\rightarrow \min$

(लिंग अंडर)

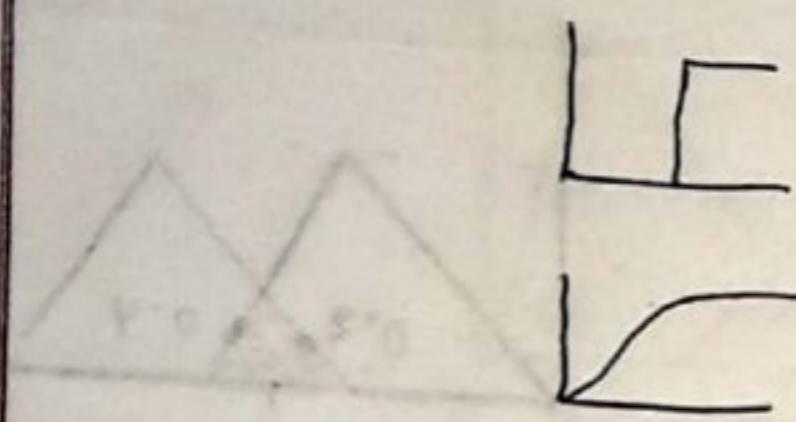
$$\mu(x) = \begin{cases} \frac{x-a}{b-a} & \\ \end{cases}$$

(slide)

Main component of FLS :



Linguistic : word  $\rightarrow$  discrete  $\rightarrow$  crisp



overlap হচ্ছে, at a time এটা membership  
এ আবণ্ণে পারছে, এটাই benefit.

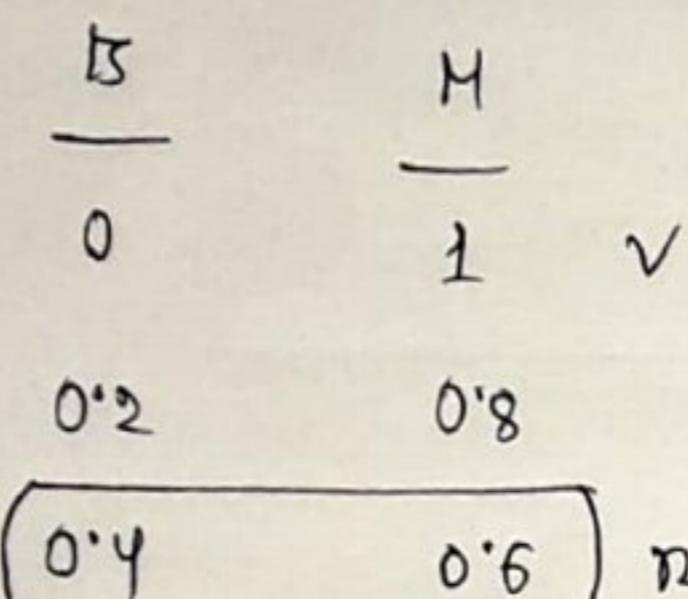
Association Rule :

Fuzzy set : একাধিক set এ আবণ্ণে পারছে।

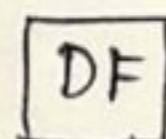
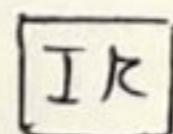
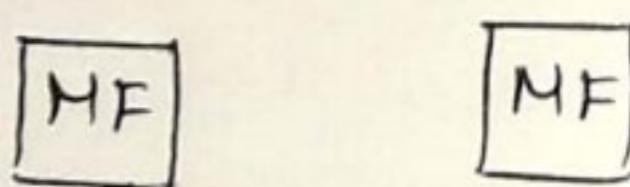
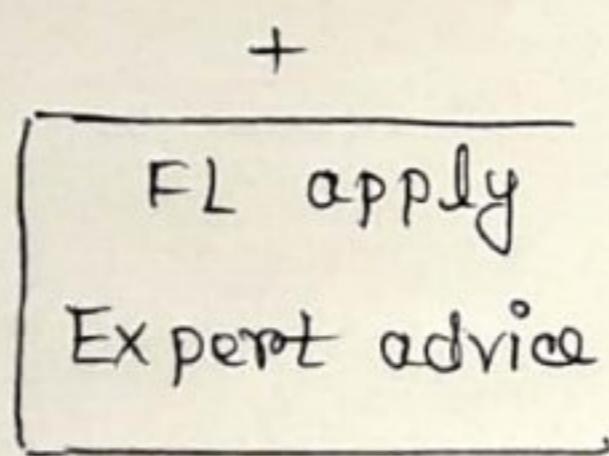
Hybrid Network : Neural network + other network

Neural Network blind approach. Explanation ability वैश्व !

### Neuro-fuzzy system



neural network यदि एरको output हो, तरिके  
high confidence decision नियम पाइ ना !



✓ Heterogeneous (असाधारण) तरिका use हो

Homo ( neural network एवं हेटोट्रोप मेंbership function  
वीजिट्रो हो )

loss function (single item)

cost (multiple, avg/ loss)  
item total

Optimization problem

perfect weight calculate करना चाहते हैं, तो optimized value तक  
जुनियर करते हैं, इसका लगानी problem of learning is cast as optimization  
problem.

Loss function

The function we want to minimize or maximize is called objective  
function or criterion. When we are minimizing it, we may also call it  
the cost function, loss function, or error function.

Maximum Likelihood

probability use (maximize)

0 0.3

Maximum Likelihood seeks to find the optimum values for the parameters  
by maximizing a likelihood function derived from training data.

## Loss function for Regression

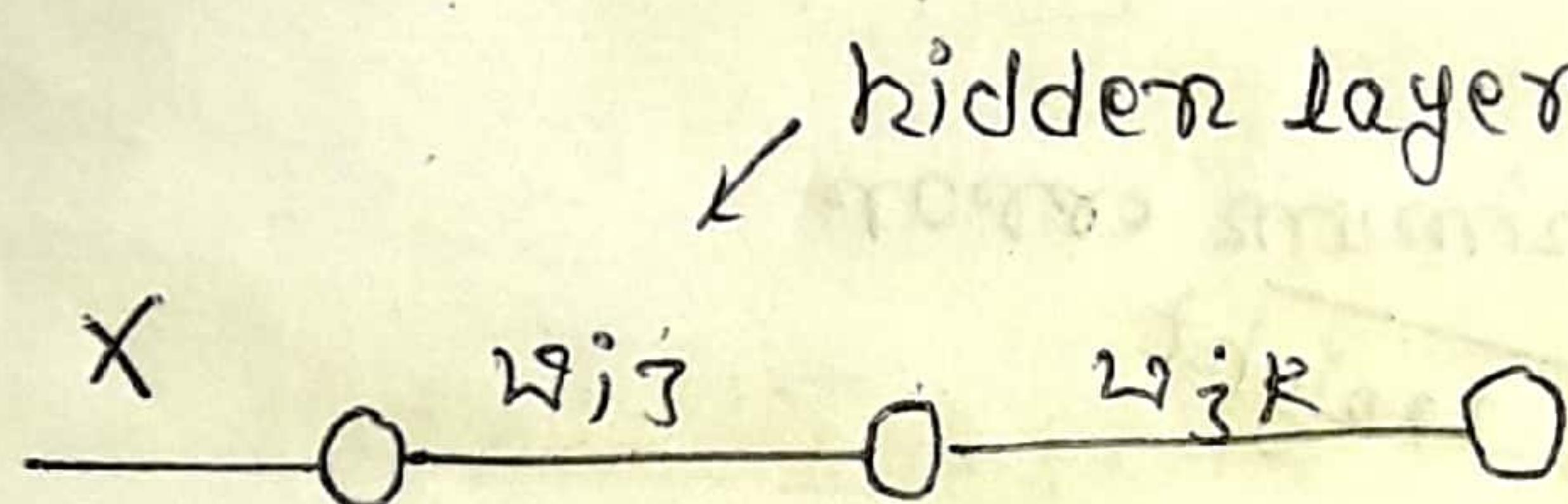
$$\text{norm}(X) = \frac{1}{\sqrt{\sum X_i^2}}$$

cross Entropy (Log loss)

Binary      categorical

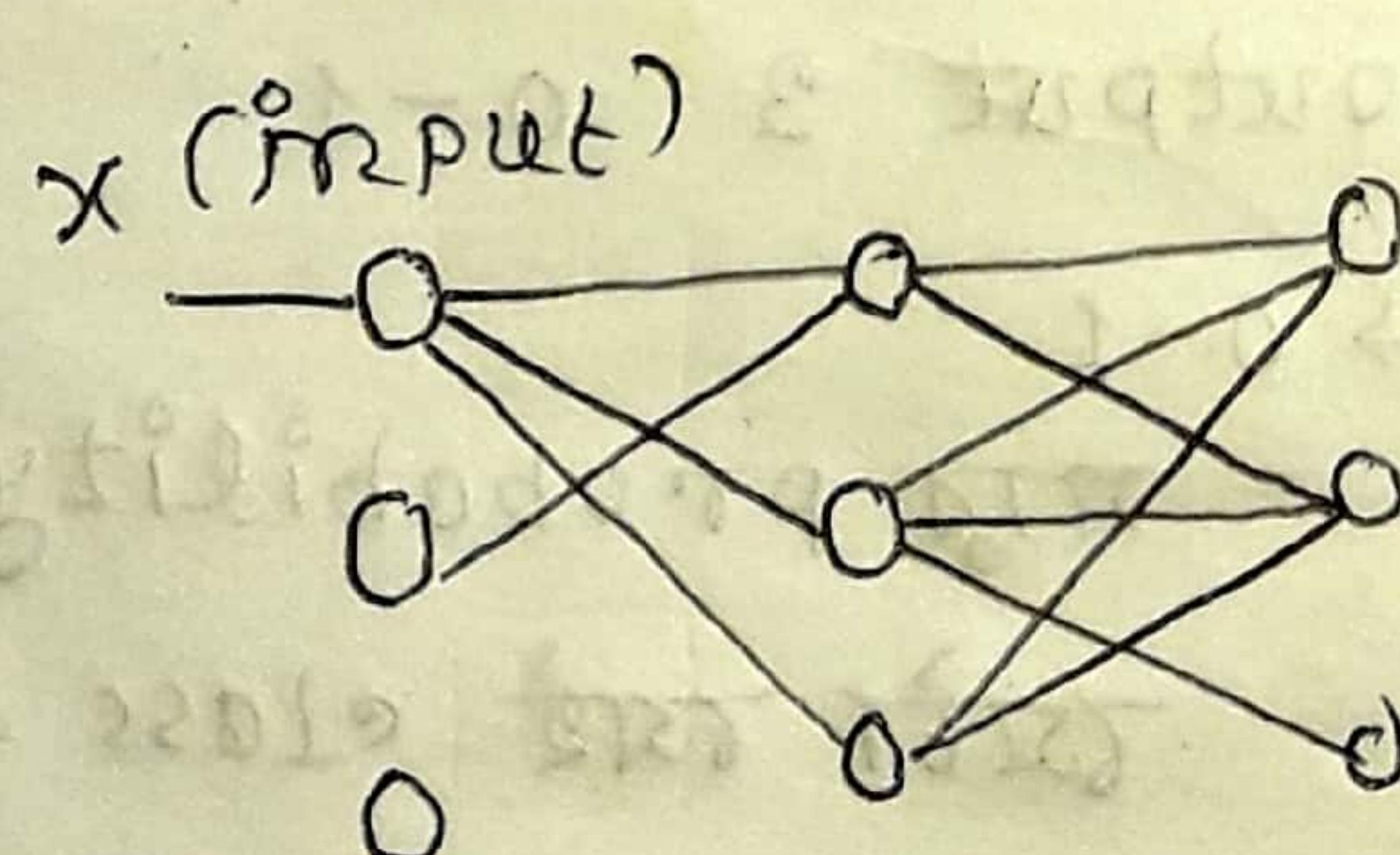
The error between two probability distributions is measured cross-entropy.

27.06.2020

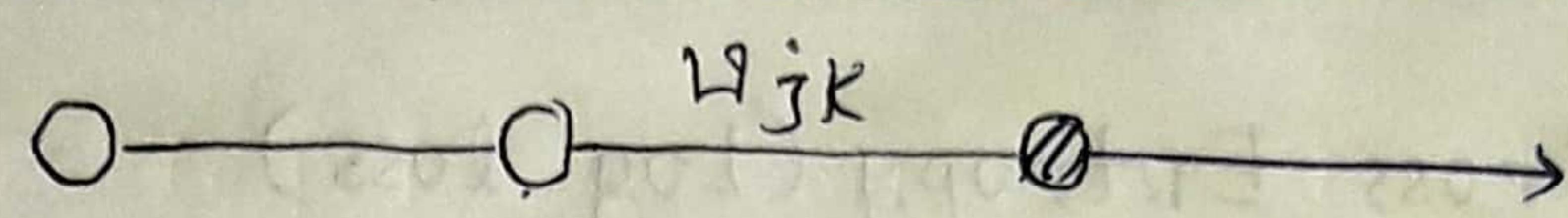


$\rightarrow \text{Net} = \sum w_{ij}x_i \rightarrow \text{arbitrary large value}$   
 multi layer       $a_j = \sigma(\text{Net}_j) = (1 + e^{-\text{Net}_j})^{-1}$

XOR  $\rightarrow$  nonlinear separate problem



fully connected



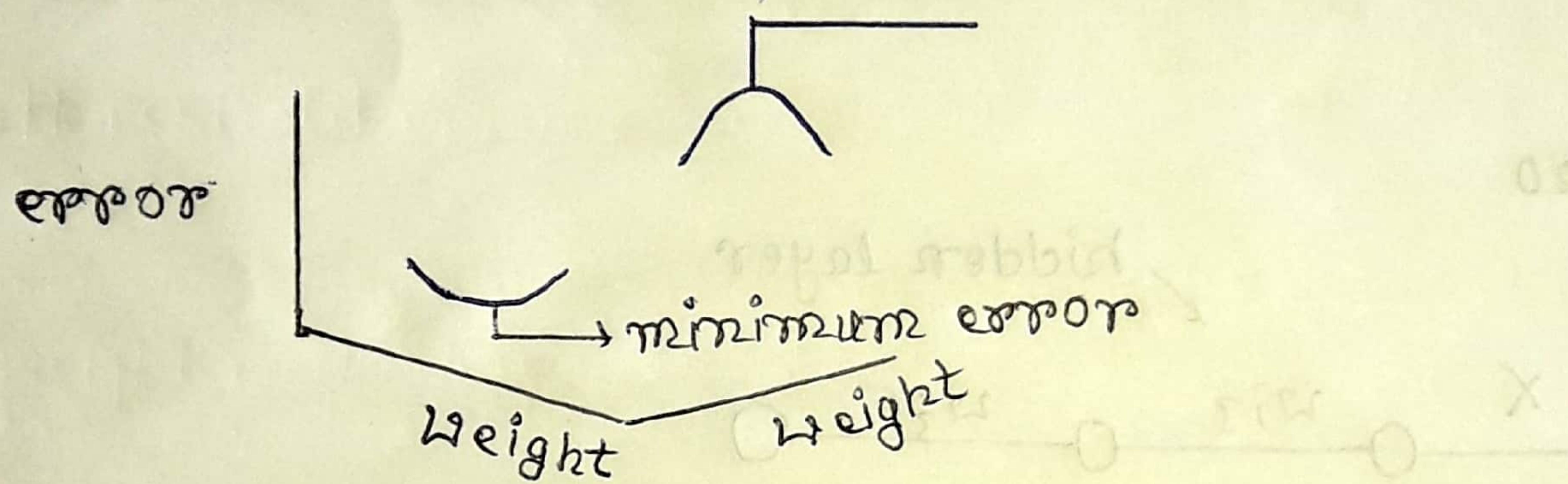
$$\text{Net} = \sum w_{jk} a_i$$

$$E = \frac{1}{2} \sum_k (t_k - a_k)^2$$

$$a_k = \sigma(\text{Net}_k) =$$

এটা class এরে output node কয়েটা কাণ্ডে?

১টা OR এটা, যদিগুলো class, তারপরে output node হবে better.

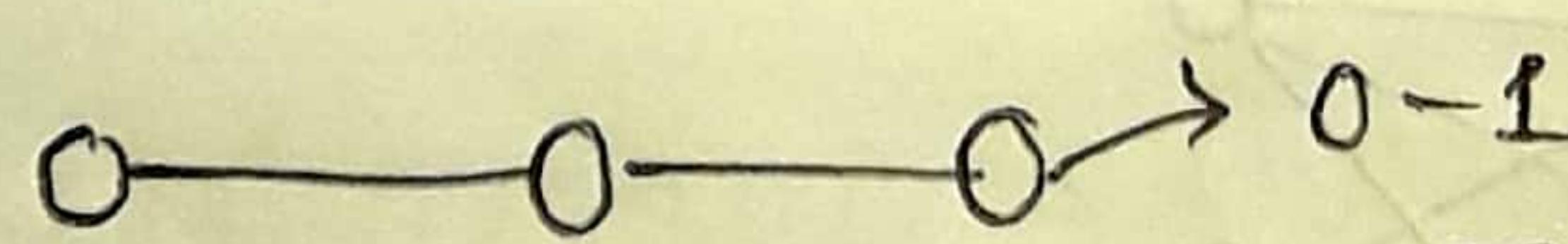


$$\text{Error fn: } E = \frac{1}{2} \sum_k (t_k - a_k)^2 \quad \text{squared error}$$

Forward:

exponential  $0 \rightarrow 1$  এর মধ্যে :

$\sigma(0-1) \rightarrow \text{output } 3(0-1)$



মান probability দেশ

মুক্তি অর্থে class এর অন্তর্ভুক্ত

Backpropagation:

$$\Delta w_k \propto -\frac{\partial E}{\partial w_k} \quad \text{Learning rate}$$

error හෝ change හෝ තුළයේ depend කාලීන weight change

$$E = \frac{1}{2} \sum_k (t_k - o_k)$$

$\hookrightarrow o_k$  හෝ තුළයේ depended  
change හැකි !

$$o_k = \sigma(\text{net}_k)$$

$\hookrightarrow$  හෝ තුළයේ depended

$$\text{net}_k = \sum_l w_{lk} a_l \quad \text{විශ්වාස හෝ layer } l \text{ න්‍යා පිළිබඳ}$$

$$\Delta w_k = -\frac{\partial E}{\partial w_k} \cdot \frac{\partial o_k}{\partial \text{net}_k} \cdot \frac{\partial \text{net}_k}{\partial w_k}$$

$\hookrightarrow$  sigmoid න්‍යා පිළිබඳ වාර්ගීකාරී

$o_k(1-o_k)$  prove ක්‍රියා යායි !

ইতুরের layer এর মালেক E change

সামোর দ্বারা পর্ত কিনা না।

ফলো  $\Delta w_i$  এর জন্য  $w_k$  থাকবে।

$$\Delta_{ji}^w \propto - \sum \frac{\partial E}{\partial w_k} \cdot \frac{\partial w_k}{\partial w_{ik}} \cdot \frac{\partial w_{ik}}{\partial w_i}$$

↳ পর্ত এর উপর depend করে  
না  $w_k$  এর উপর।

Threshold error:

weight initialize random

squared error না নিয়ে cross entropy নিলে  $e^{-2}$  টি

change হয়ে যাবে।

ভালো error function, differentiable না হলে কোনো

ভালো result আওয়া যাবে না।

05.11.20

## Hopfield Net

associative memory network and fully-connected network.

computer এর memory দে data কেস রাখলে কি যোগ্য হতে পাবে?

- $1000 \times 3 \times 256$  bit জাগতে 1টি image store করতে storage limited
- Backup রাখতেও same size এর memory জাগতে so, costly  
Disk failure

Associative Memory, association এর মাধ্যমে store রাখা!

একাধিক জন মিলে রাখতে।

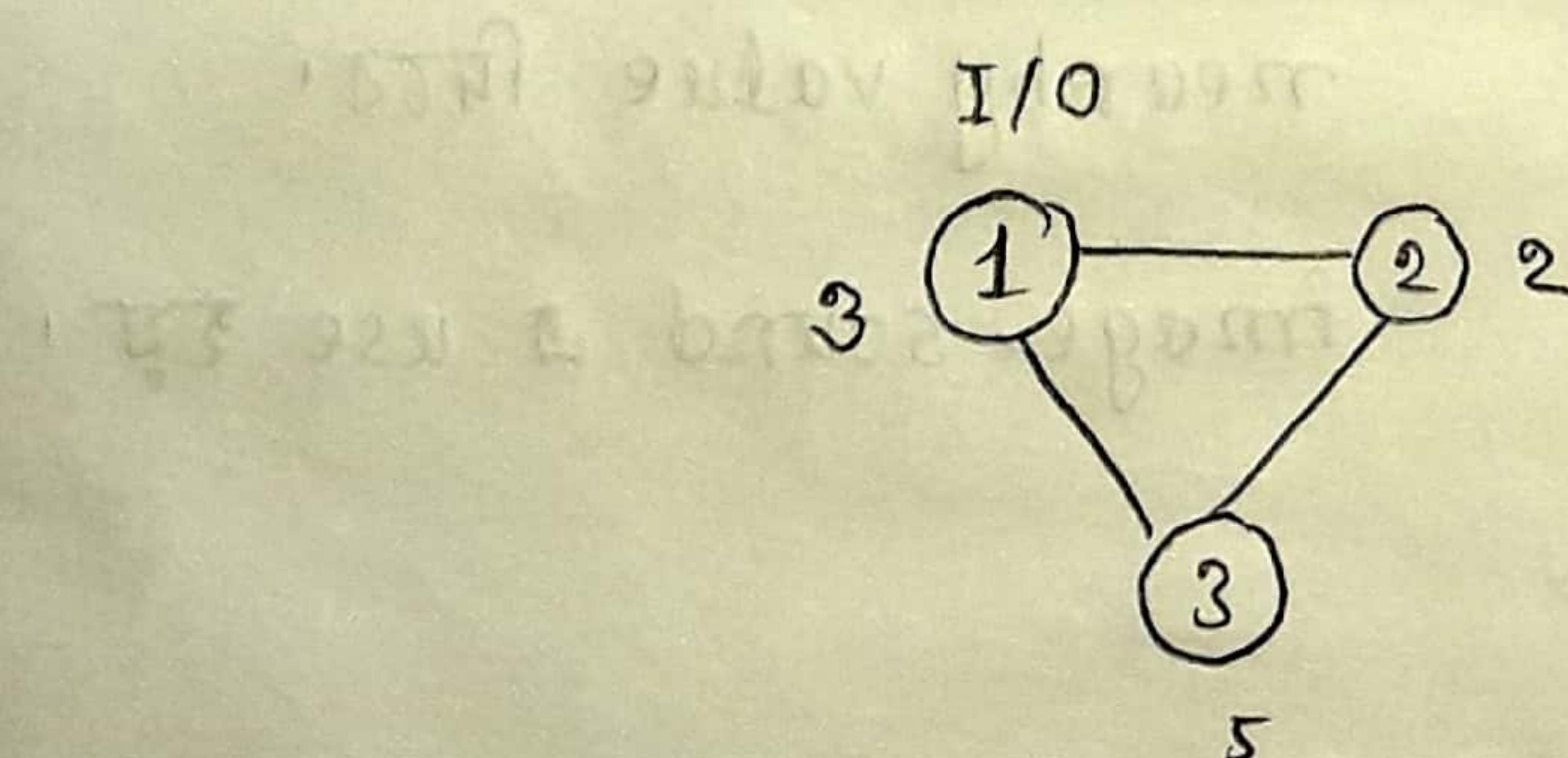
Suppose উজনকে (১) মাধ্যমে মনে রাখতে চালাব।

" একজন ই, অ্যারেকেজন ত " একক

so single number/unit হিয়েবে না হোগে আলাদাভাবে একটি

concept হিয়েবে। একেকটা neuron একেকটা part store করবে।

fully connected Network

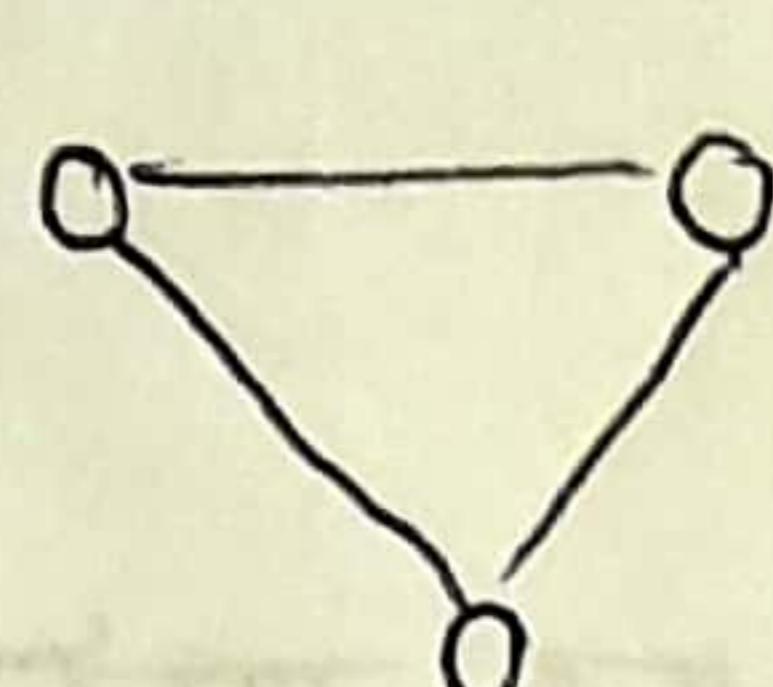


একটা node এ input/output হিয়েবে কাজ কৰবে।

Backpropagation & feature missing हले छान्ति

पाइया याएँना, फिरू होप्सिल् पोस्सिल्

द्रुग्जनार् memory छे तरके associative memory. द्रुविधा  
फै? → प्रश्नलाई data बाटिना केने memory size fixed  
fixed neuron ए अनेकवेशि विंडो store करा याएँ।



111  
123  
143  
321

→ 3 digit एवं अनेक

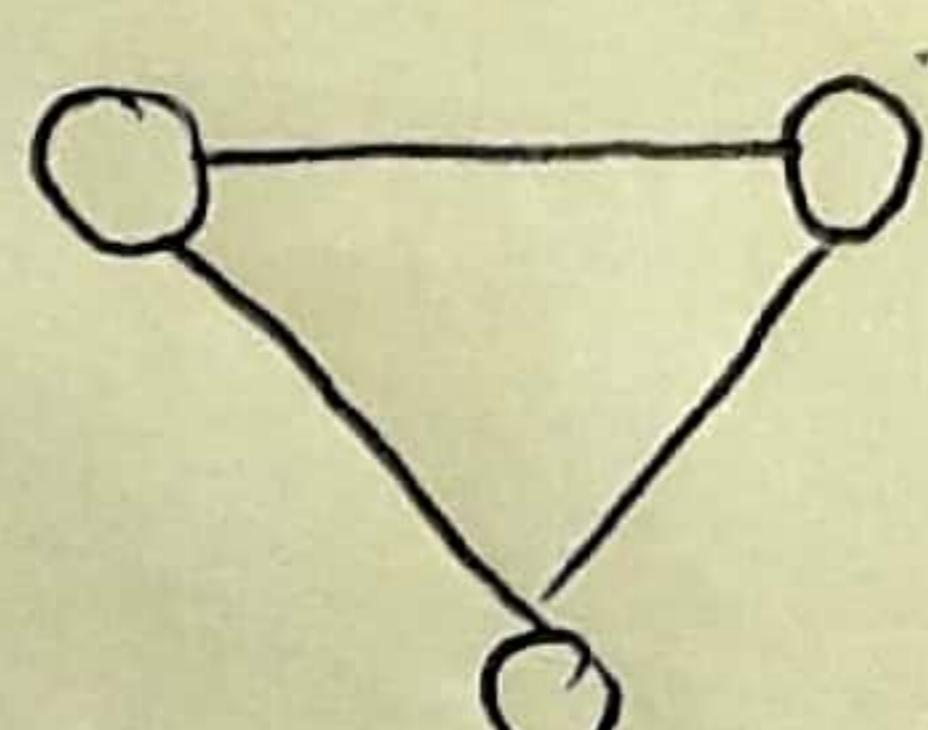
प्रश्ना बाटि 3 टी neuron  
store कर्ते बाथ्ये।

[ SVM एवं Kernel वे dimension analysis ]

NN रिक्ष → catastrophic forgetting  
forgetting

Associative एवं problem → Network data छैने यत्रे आहे।  
store

325  
125



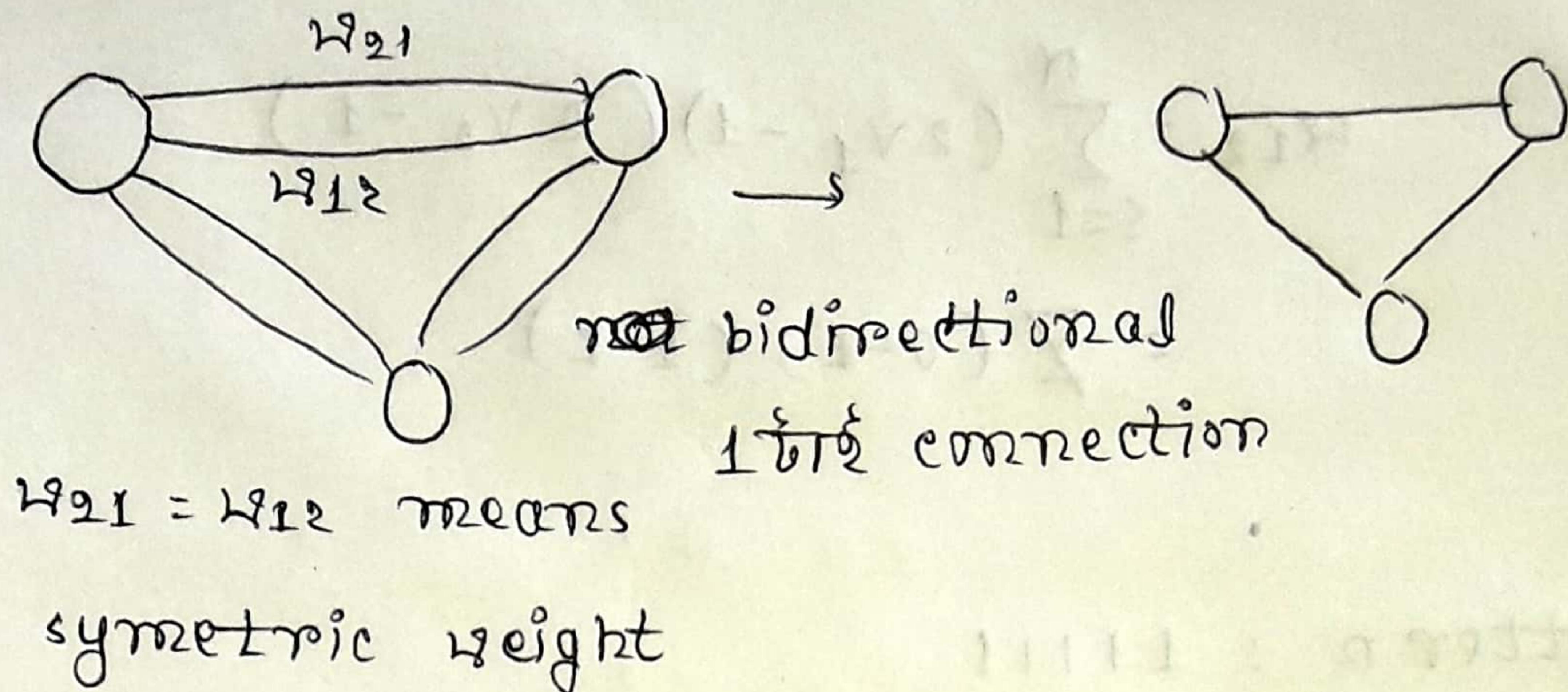
324 search बाबते दिल,  
nearby value दिवे।

image, sound ए व्हे हाय।

3 टी node associate

चुंजे पाइयार् success राते कर्म।

adjacency matrix හියාවේ graph තාක්ෂණ ප්‍රසෙන කළු



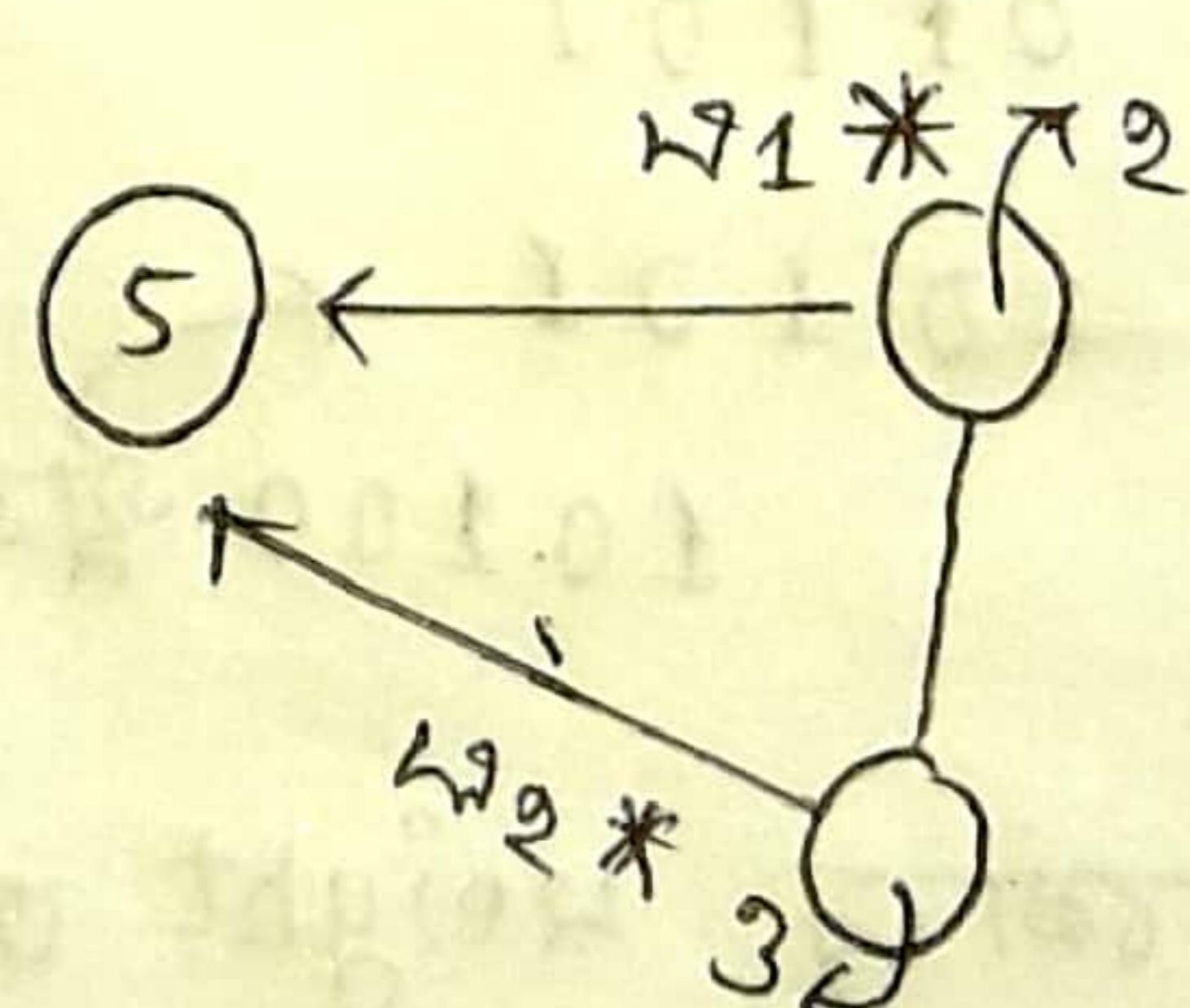
1. යැඟැදුළු මිනු

2. සිංහල

step 1: store and weight update

bipolar Input  $1, -1$

weight update / result



pattern num  
 $v_1 = 101$

$v_2 = 110$

$v_3 = 001$

$v^4 = 111$

ගැනීමේ  $v_2^2 = 1$   
දැනු  
සෑතු

$$w_{ij} = \sum 2v_i^j - 1$$

$$w_{ij} = \sum (2v_i^s - 1) (2v_j^s - 1)$$

Pattern 01101 : (5 অংক pattern number)

$$\begin{aligned} w_{12} &= \sum_{s=1}^N (2v_1^s - 1)(2v_2^s - 1) \\ &= \sum (0-1)(2-1) \end{aligned}$$

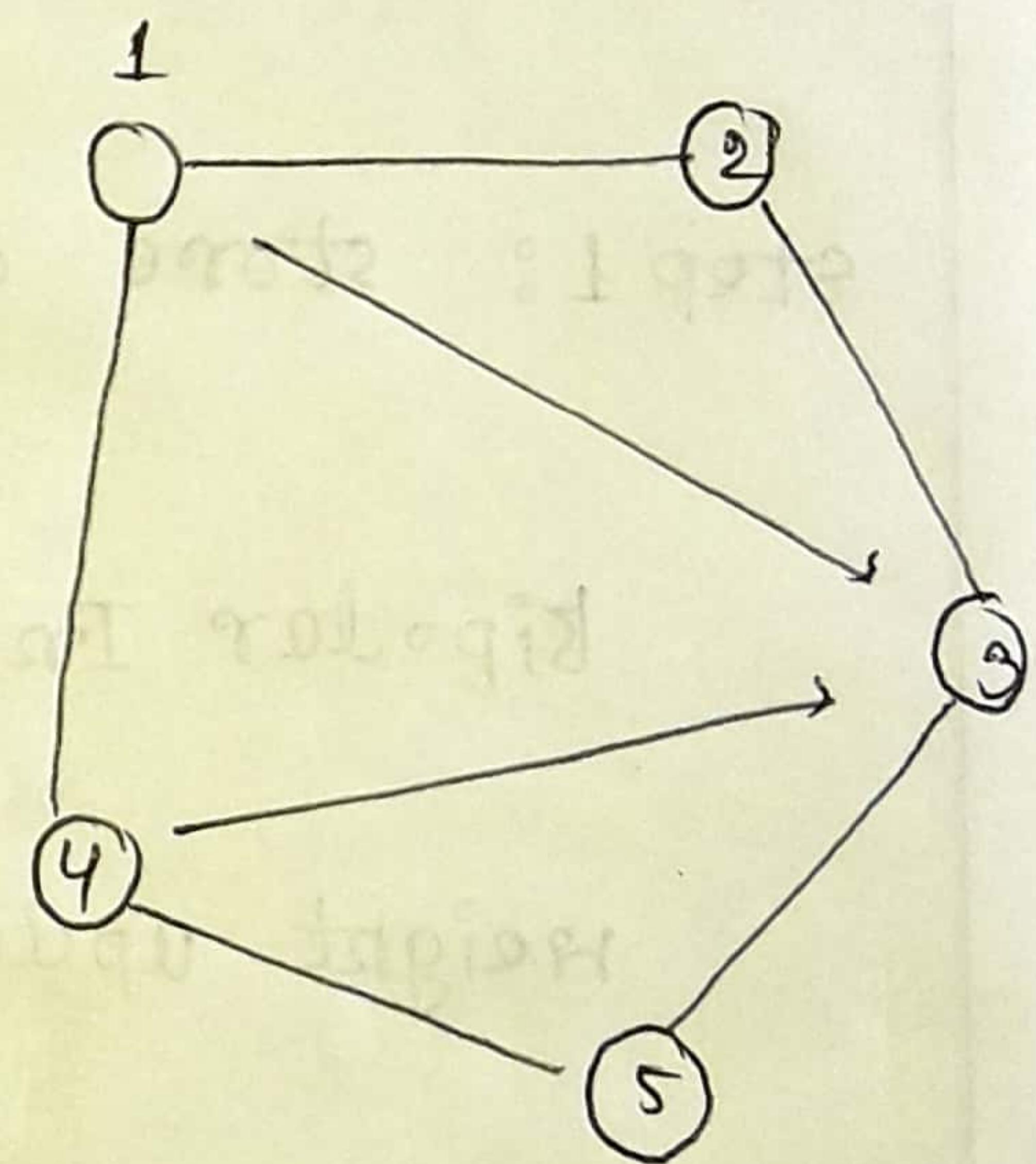
Pattern : 11111

Search :

$$v_3^m = \sum w_{ij} v_j$$

$v_j$

$$v_3^s = v_1^s w_{i3} + v_2^s w_{23}$$



Input : 01101

10101

10100 আউটপুট গেটে

step

এই step তারে weight আছে।

২য় step এ 11111 আবরণ এজন্য আবাবে input কিম্বা 11111 ফির।

$$v_3^m = \sum$$

এখন যেগোনোটা আগে calculate করা হবে,

same process iteration করতে করতে আবাবে change হবে না।

তখন process end.