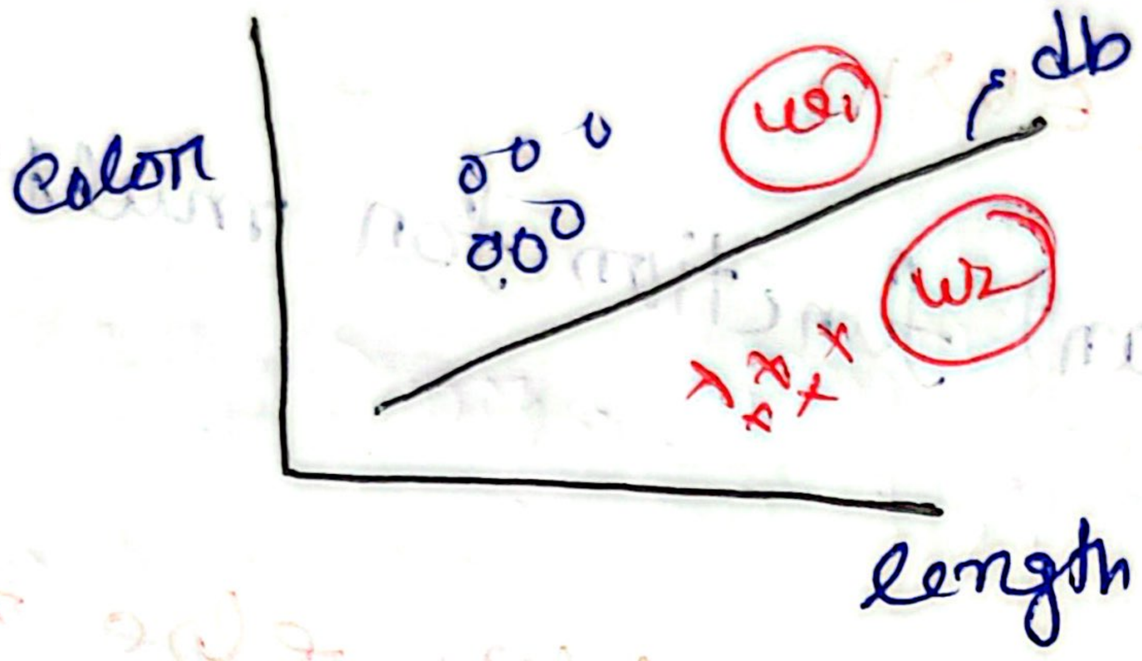
Perceptron L-03

Linear discriminant func

Model: Mathematical description of a class

Decision rule: which gives decision boundary for classification

Decision Boundary: is defined by decision rule



Line eqn

$$y = mx + c$$

$$ax + by + c \approx w_1 x_1 + w_2 x_2 + w_0 = 0$$

discriminant f^n

$$g(x) = ax + by + c = a * \text{color} + b * \text{color}$$

decision rule

$$\begin{cases} \text{if } g(x) > 0 \text{ then } x \in \omega_1 \\ g(x) < 0 \text{ then } x \in \omega_2 \end{cases}$$

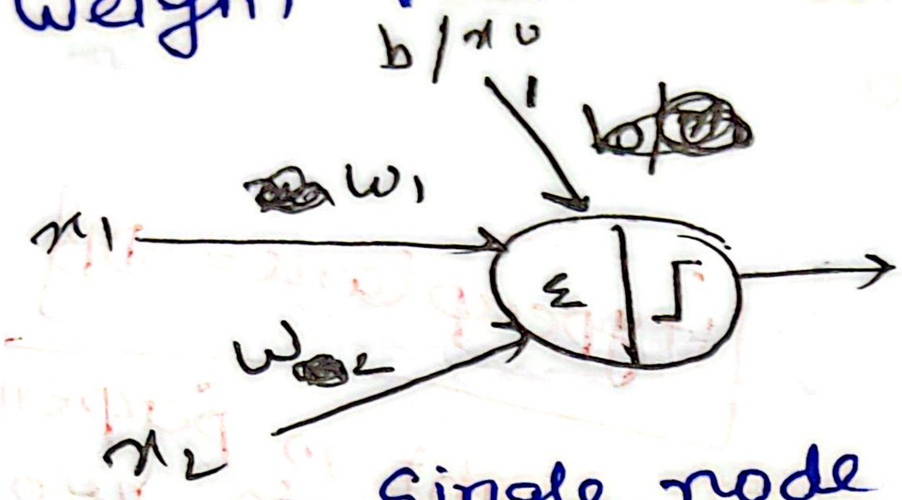
x = length
= color

Hyperplane:

Decision boundary in higher dimension. data - 3D, Hyperplane - 2D

Pattern / feature vector = $x = \{x_1, x_2, \dots x_n\}$

a weight vector = $w = \{w_1, w_2, w_3 \dots w_n\}$



$g(x)$ discriminant function

Single node of a perceptron.

$$g(x) = x_1 w_1 + x_2 w_2 + \dots x_n w_n + w_0$$

$$= \sum_{i=1}^{N} w_i x_i + w_0$$

$$\boxed{= w^T x + w_0}$$

here w = weight vector
w_0 = bias
x = feature vector

$w \cdot x + b$

$$= w_1 x_1 + w_2 x_2 + \dots w_0$$

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix} \quad [x_1 \; x_2 \dots x_n]$$

dot product = inner product

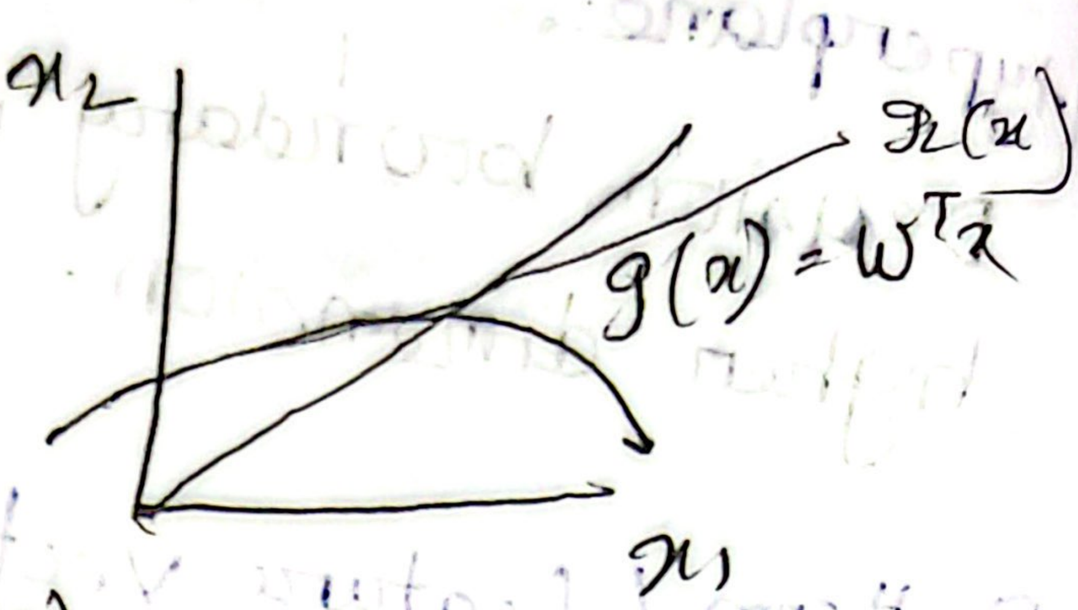
$$= \boxed{w \cdot x + b}$$

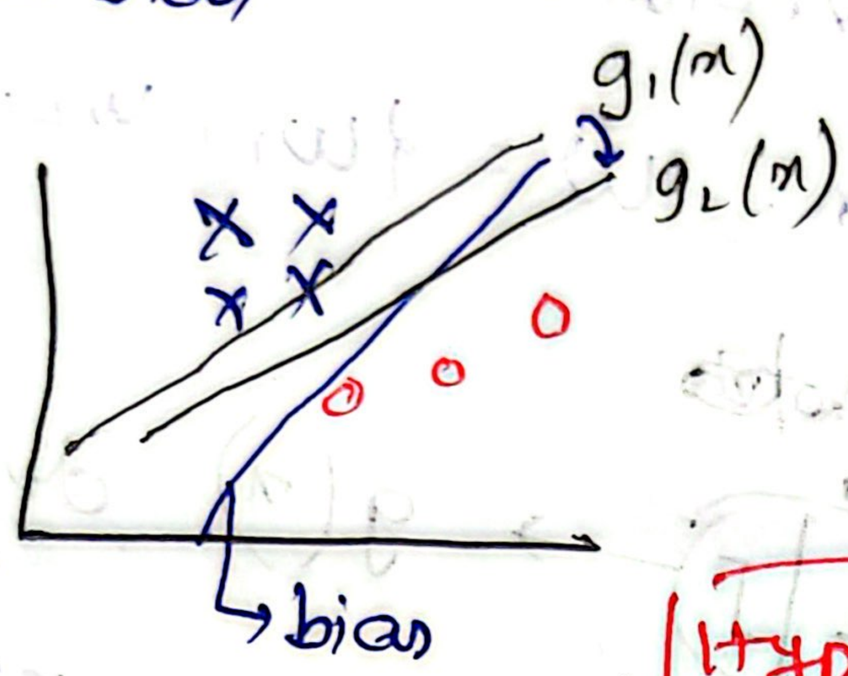
$\hookrightarrow mx + c$

DB

orientation change
→ weight

Location → bias





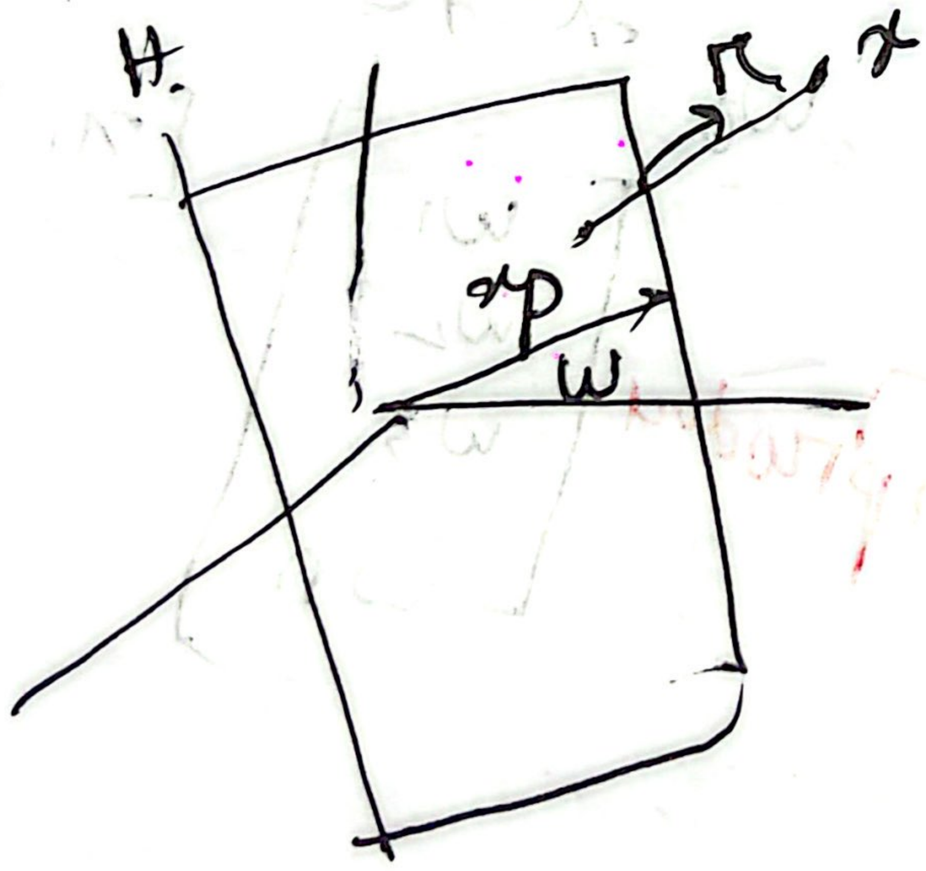
$\boxed{\text{Hyperplane} = \text{DB}}$

Exam a আসে

L-04

Proof: $\boxed{w \perp H}$ weight is perpendicular to hyperplane

The discriminant f^n $g(x)$ gives an algebric measure of the distance from x to the hyperplane



x কে hyperplane এর উপর project করব

Eq^n of Hyperplane

$$\boxed{H = w^T x + w_0 = 0}$$

→ $\vec{w}$ (weight) and r is parallel or r is $\perp$ (perpendicular) to H

→ $\vec{w}$ the direction of H

→ r is the distance from x point to H.

We want to determine the relationship weight vector and hyperplane → W is perpendicular to H

vector projection:

$$\boxed{x = x_p + r \cdot \frac{w}{\|w\|}}$$

→ $w \perp H$ so, $w \cdot H = 0$

Let, a, b are two points on H.

$g(a) = w^T a + w_0 = 0$

$g(b) = w^T b + w_0 = 0$

$\therefore g(a) = g(b)$

$\Rightarrow w^T a + w_0 = w^T b + w_0$

$\Rightarrow w^T a - w^T b = 0$

$\Rightarrow w^T (a-b) = 0$

$\Rightarrow \boxed{w \cdot (a-b) = 0}$

$\boxed{w \perp (a-}$

দুটি ভেক্টরের dot product 0

Scanned with CamScanner

a and b are two points on plane H.

$\therefore$ w is $\perp$ H

$\boxed{w \perp H}$

$\boxed{g(x) = \text{value}}$ → Exam এ আসবে

$$g(x) = w^T x + w_0$$

$$= w^T \left(x_p + r \frac{w}{\|w\|} \right) + w_0$$

$$= w^T x_p + r \cdot \frac{w^T w}{\|w\|} + w_0$$

$$= \left(w^T x_p + w_0 \right) + r \cdot \frac{w^T w}{\|w\|}$$

$$= g(x_p) + r \cdot \frac{w^T w}{\|w\|}$$

as x_p is on the hyper plane so

$$g(x_p) = 0$$

$$= r \cdot \frac{\|w\|^2}{\|w\|}$$

$$\boxed{w^T w = \|w\|^2}$$

$$= r \cdot \|w\|$$

$r \rightarrow$ distance from x to x_p

$$\boxed{g(x) = r \cdot \|w\|}$$

Scanned with CamScanner

$$\boxed{r = \frac{g(x)}{\|w\|}}$$

Let, $w = \begin{bmatrix} 5 \\ -2 \end{bmatrix}$ $g(x) = 5x_1 - 2x_2 + 3$

Find the value of r for $x(1,1)$

$$r = \frac{g(x)}{\|w\|} \qquad \|w\| = \sqrt{(5)^2 + (-2)^2} = \sqrt{29}$$

$$g(x) = 5x_1 - 2x_2 + 3$$
$$= 5(1) - 2(1) + 3$$

$$r = \frac{5 - 2 + 3}{\sqrt{29}} = \frac{6}{\sqrt{29}}$$

How to optimize the weight vector

Design of weight vector w (two category linear separable case)

$g(x) = w^T x + w_0$ convert in homogenious form,

$g(x) = a^t y$ $\left[\begin{array}{l} a \rightarrow \text{modified weight vector} \\ y \rightarrow \text{augmented feature vector} \end{array}\right]$

$a \rightarrow [w_1, w_2, \ldots w_1, w_0]$

$w_0 \downarrow$ bias

$y \rightarrow \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_t \end{bmatrix}$ → last a 1 থাকবে augmented feature

$$\overset{a}{[w_1 \; w_2 \; w_3 \cdots w_d \; w_0]} \quad \overset{y}{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \\ 1 \end{bmatrix}}$$

$$a \cdot y = w_1 x_1 + w_2 x_2 \cdots + w_d x_d + w_0$$

$$= \sum_{i=1}^{d} w_i x_i + w_0 = w^T x + w_0$$

So,

$$\boxed{g(x) = a^t y}$$

Decision rule

$a^t y > 0 \Rightarrow y \in \omega_1$

$a^t y < 0 \Rightarrow y \in \omega_2$

Now, instead of two critera (decision rule)

Let's use 1

- take sample class ω_1 as it is
- negating the sample of class ω_2 } Norm

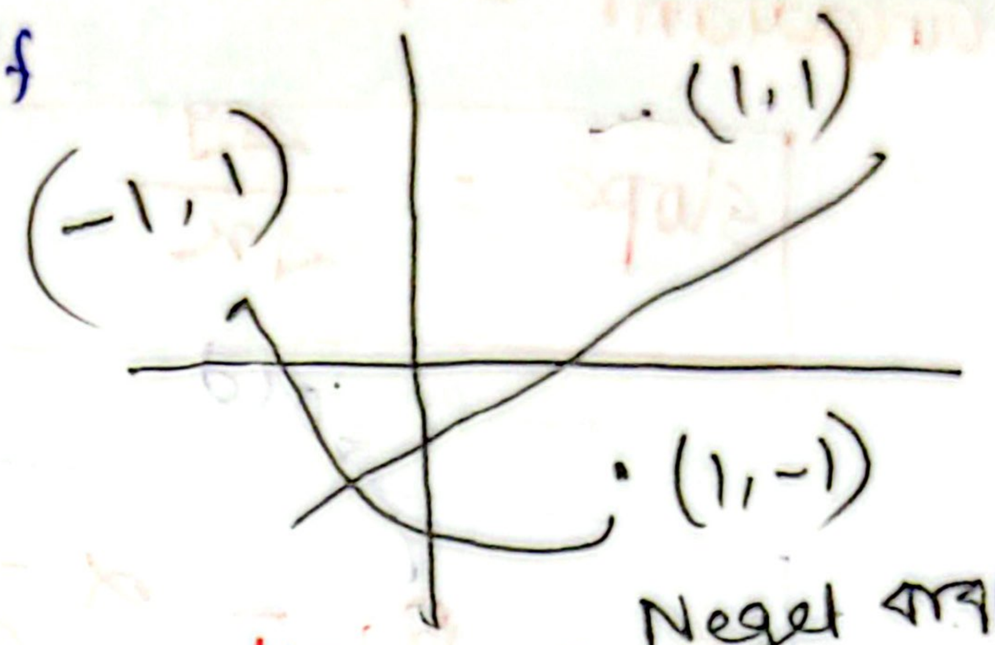
বাউন্ডারীর এক পাশে সব।

Now, $a^t y_1 > 0$

– Criterion function $J(a)$

Normalination

[$J(a)$ has to be minimized if a in the correct weight vector]



Negel করলে sign change

Target:

$J(a)$, $a \rightarrow$ weight vector.

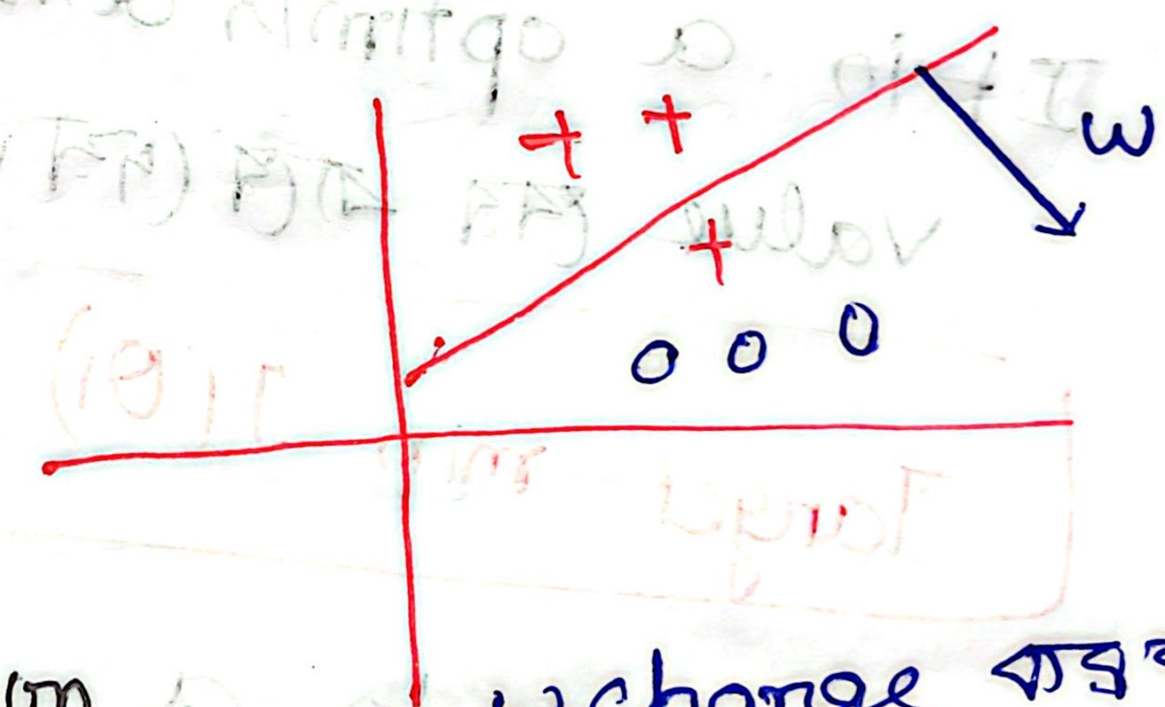
$J(a)$ কে minimize করব (optimization algorithm use করে) যে weight টা দিলে এই $J(a)$ টা minimize হবে সবচেয়ে বোঝা, তাহলেই সবগুলো data clarify হবে।

যেগুলো satisfy করে
সেগুলো correctly clarify আর
যেগুলো করবে না min clarify

For minimizing $J(a)$ we can use Gradiant Descent Algo.

J, weight এর function
$a \rightarrow$ weight

Gradiant Descent:



1) Convex Optimination

w change করব
যার কারণে dB/
hyperplane আসবে।
$w \perp dB$ থাকবে।

2) Non Convex optimination

minimum value / maximum value Gradiant descent কাজ করে না

Gradiant descent → derivative

$$\boxed{\text{slope} = \frac{\Delta y}{\Delta x}}$$

$$\boxed{\theta_1 = \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)}$$

old

~~old~~ new point

learning rate

derivative

- take tangent of the line
- what's the slope of the line target

= what is Gradiant Descent Algo:

It is a optimization algo যে minimum value বের করে দেয়।

$$\boxed{\text{Target } \min J(\theta_1) \text{ wrt to } \theta_1}$$

in $J(a)$ wrt to a

update rule: $\boxed{a(k+1) = a(k) - \eta \Delta J(a(k)}$

here at k iteration $\boxed{a(k) \rightarrow \text{value of weight vector at } k \text{ iteration}}$

Scanned with CamScanner

Update rule:

$$a(k+1) = a(k) - \eta \nabla J(a(k))$$

Perceptron Criterion:

যেই data এর জন্য min classified হবে

তার জন্য:

→ design on misclassified data

$$J_p(a) = \sum_{\forall y \to \text{misclassified}} -a^t y$$

→ '-' যাতে positive হয়

$$\nabla_a J_p(a) = \sum_{\forall y - \text{misclassified}} (-y)$$

For update →

$$a(0) = \text{arbitary}$$

$$a(k+1) = a(k) + \sum_{\forall y - \text{misclassified}} y$$

training এর time a parameter change হয়

L-05

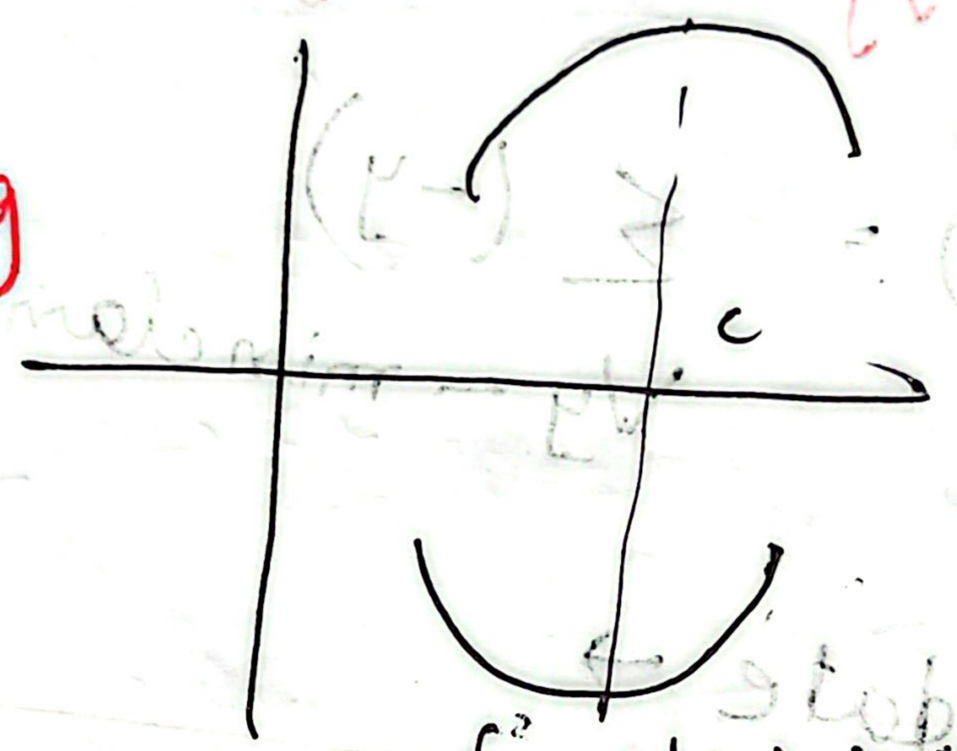
To determine our function is a maximize problem or minimize problem, we need to determine the second derivative

Optimal value (max/min) : 1st point
এর derivative 0 হলে

2nd derivate < 0 হলে maximum
> 0 হলে minimum

$f'(c) = 0$

$f''(c) < 0$



$f''(c) < 0$
maximum → decreasing

minimum → increasing

$f''(c) > 0$

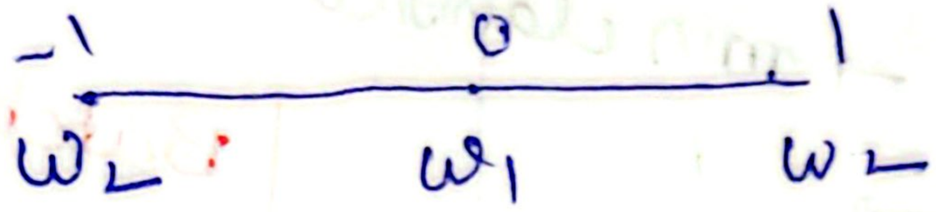
$f''(c) > 0$

φ function → Phi function

data কে lower dimension থেকে higher dimen এ নিলে আসে

$$g(x) = w^T x + w_0$$
$$= w_1 x_1 + w_2 x_2 + \cdots + w_0$$

Given:



$\omega_1 = 0$

$\omega_2 = -1, 1$

Given, $\phi(x) = [x^2 \;\; x \;\; 1]$

মান বসাবে

ϕ function (Apply)

$\Rightarrow$

	ω_1	ω_2	
x	0	$-1, +1$	
y	$\begin{bmatrix}0\\0\\1\end{bmatrix}$	$\begin{bmatrix}1\\-1\\1\end{bmatrix}$	$\begin{bmatrix}1\\1\\1\end{bmatrix}$

$\rightarrow$ 3d data

Apply Normalization. (Neget করব - sign change করব)

$\omega_1 \rightarrow$ same

$\begin{bmatrix}0\\0\\1\end{bmatrix}$	$\begin{bmatrix}-1\\-1\\-1\end{bmatrix}$	$\begin{bmatrix}-1\\-1\\-1\end{bmatrix}$	$\rightarrow$ Apply Normalization

intial weight vector

$w = [0 \;\; 0 \;\; 0]$

learning rate

$\eta = 1$

Update rule

$$a(k+1) = a(k) + \eta \sum_{y \rightarrow \text{misc}} y$$

D.R

$$\boxed{a^t y_i > 0}$$ $\rightarrow$ সঠিক না হয় মাগত

y	w^T	$w^T y > 0$
0 0 1 -1 1 -1 -1 -1 -1	0 0 0	0 → misclassified 0 → " 0 → "
0 0 1 -1 1 -1 -1 -1 -1	-2 0 -1	-1 → mc +3 → cc +3 → cc
0 0 1 -1 1 -1 -1 -1 -1	-2 0 0	0 → mc 2 → cc 2 → cc
0 0 1 -1 1 -1 -1 -1 -1	-2 0 1	1 → cc 1 → cc 1 → cc

$$\boxed{w^T = [-2 \quad 0 \quad 1]}$$

[Batch update
many at a t

update rule

$a(k+1) = a(k) + \eta \sum y \quad \forall y \to$

$\sum y = [-2 \quad 0 \quad -1]$

$a(k+1) = [0 \quad 0 \quad 0] +$
$1 * [-2 \quad 0 \quad -$

$= [-2 \quad 0 \quad -1]$

2nd,

$a(k+1) = [-2 \quad 0 \quad -1] +$
$1 * [0 \quad 0 \quad 1]$

$= [-2 \quad 0 \quad 0]$

3rd,

$a(k+1) = [-2 \quad 0 \quad -1] +$
$1 * [0 \quad 0 \quad 1]$

Single update / One at a time update rules

* Single update

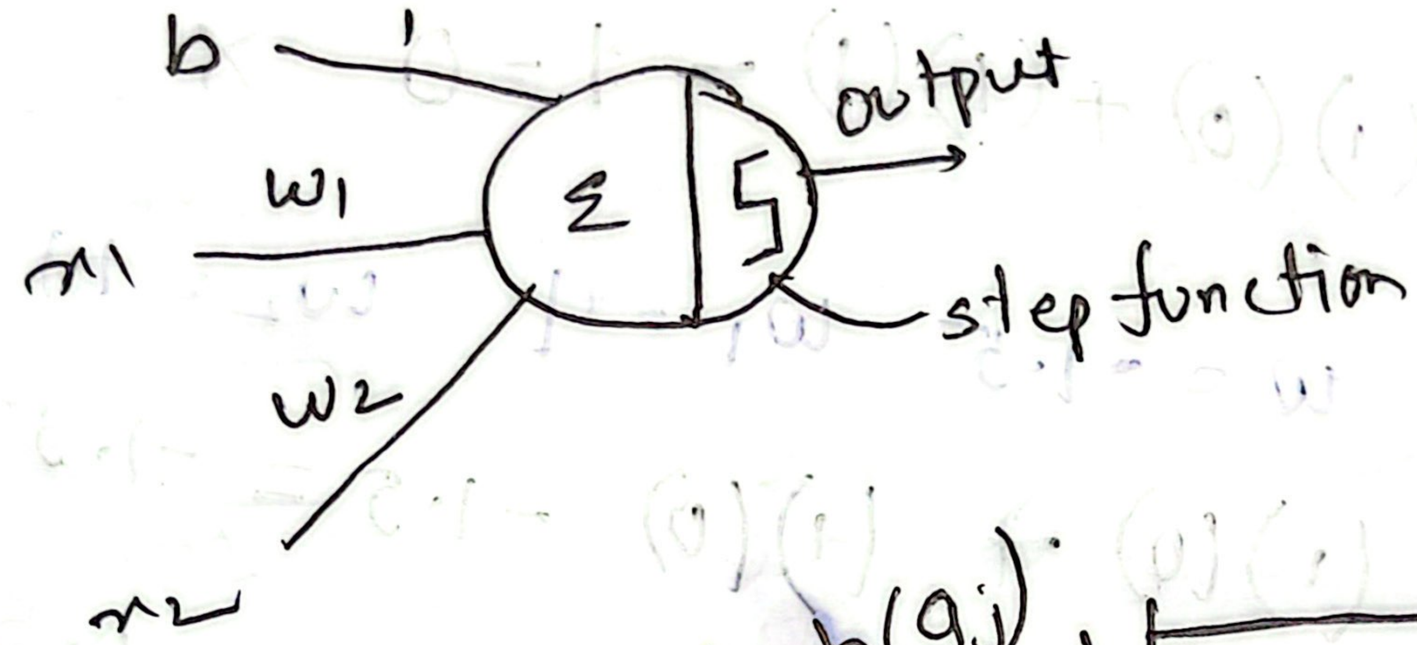
$$a(k+1) = a(k) + \eta y$$

$\forall y \rightarrow mc$

L-06 - Multilayer Perceptron

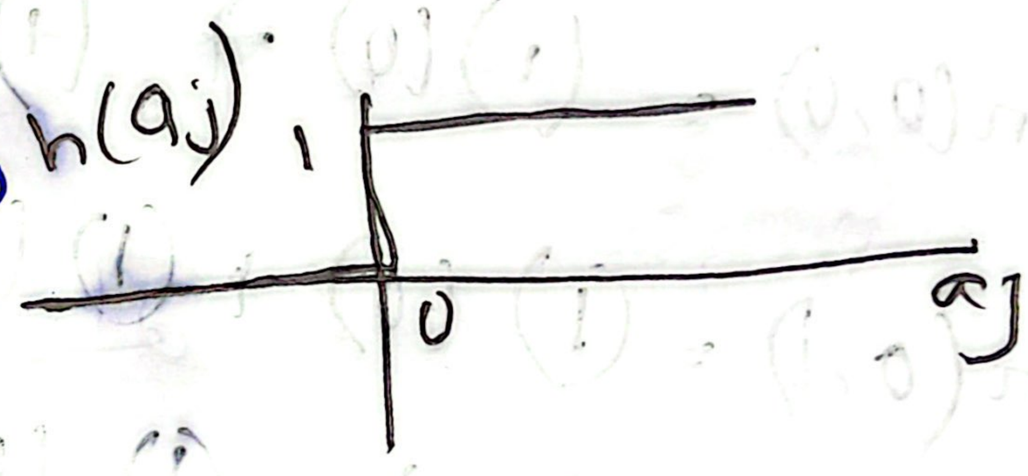
Logic Gates:

- Design perceptron to represent boolean f^n



step function এর Back properties

1) vanishing Gradient problem হয়



$h > 0 = 1$

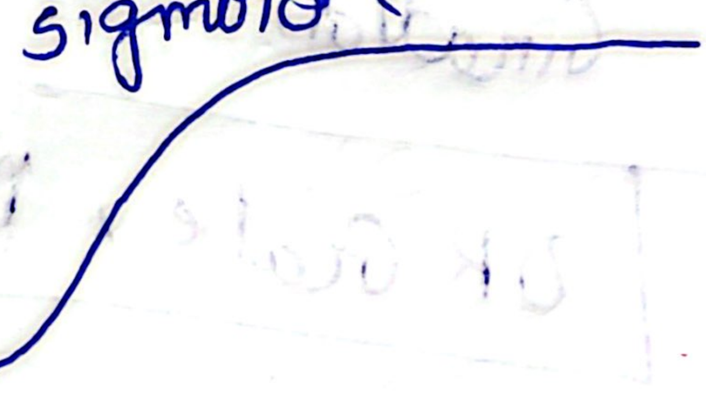
$h < 0 = 0$

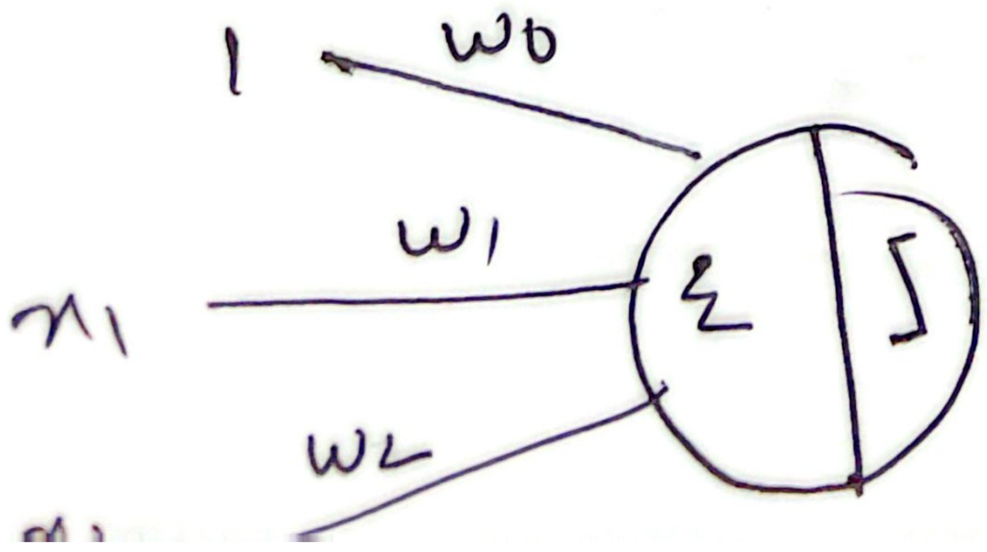
$h = 0;$

And Gate

x_1	x_2	out
0	0	0
0	1	0
1	0	0
1	1	1

sigmoid ← derivati





Scanned with CamScanner

what will be the weight to estimate And

$$\boxed{w^T x + w_0 \approx w_1 x_1 + w_2 x_2 + w_0}$$

Option 1, $w = -1$ $w_1 = +1$ $w_2 = +1$

$h > 0 = 1$

$h < 0 = 0$

$h = 0$ confl

For $(0,0) = (1)(0) + 1(0) - 1 = -1$
$= 0$

$F(0,1) = (1)(0) + (1)(1) - 1 = 0$ ×

Option 2, $w = -1.5$ $w_1 = +1$ $w_2 = +1$

For $(0,0) = (1)(0) + (1)(0) - 1.5 = -1.5 = 0$

For $(0,1) = (1)(0) + (1)(1) - 1.5 = -0.5 = 0$

For $(1,0) = (1)(1) + (1)(0) - 1.5 = -0.5 = 0$

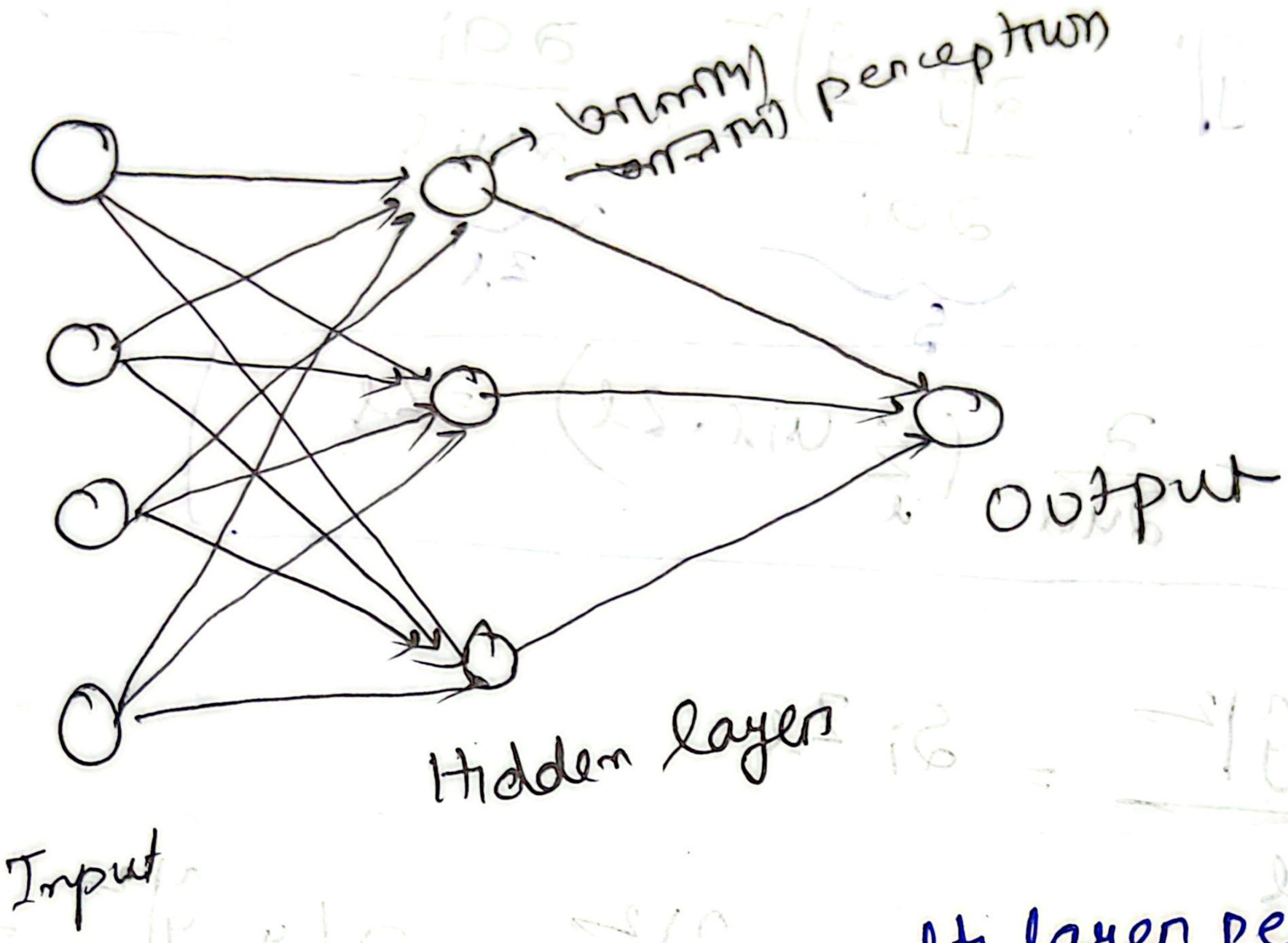
For $(1,1) = (1)(1) + (1)(1) - 1.5 = 0.5 = 1$

Gradient descent দিয়ে এই গুলো খুব বার বার আসে

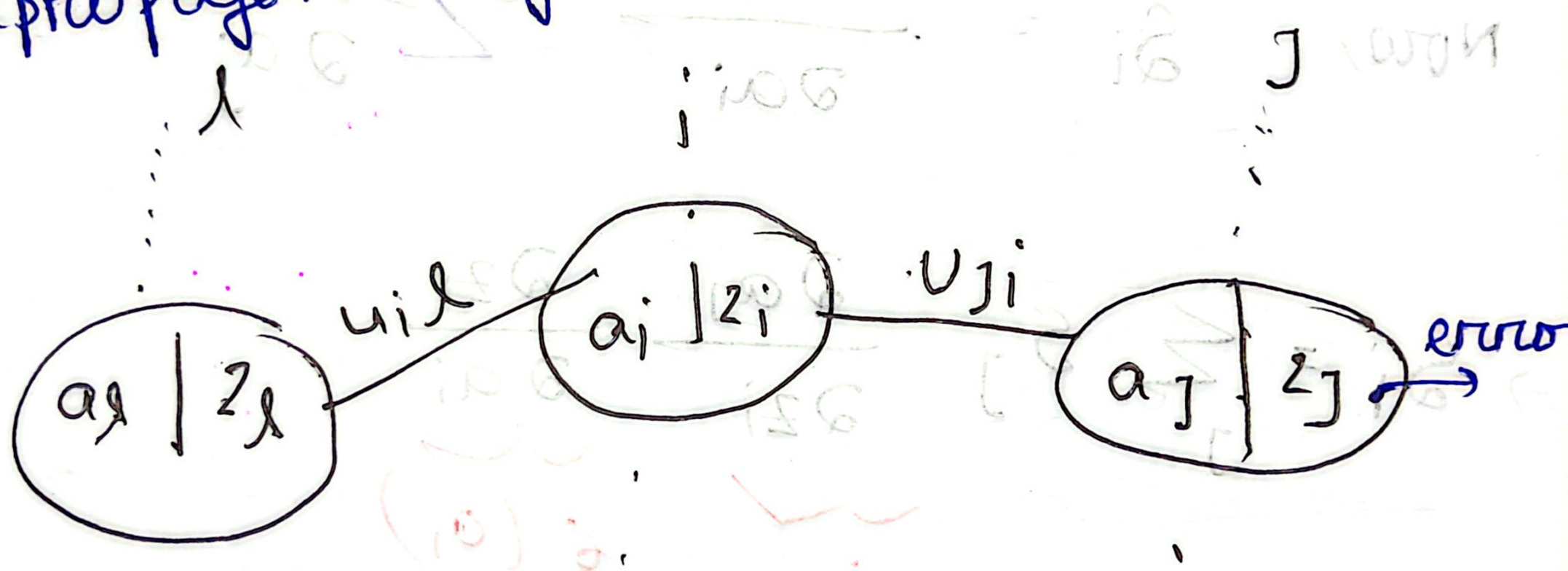
$\boxed{\text{OR Gate, NOT Gate}}$ → HW

XOR Gate একাই single layer perceptron দিয়ে করা যায় না।

Multi layer



Backpropagation Algo of multilayer perceptron



Activation function → sigmoid

$z_i = \sigma(a_i)$

$$\boxed{\sigma(a) = \frac{1}{1+e^{-a}}}$$

Loss Function: $|y - \hat{y}|$

True, predic

$$a_i = \sum_l u_{il} \cdot z_l$$

$$\frac{\partial |y-\hat{y}|^2}{\partial u_{il}} = \underbrace{\frac{\partial |y-\hat{y}|^2}{\partial a_i}}_{\delta} \cdot \underbrace{\frac{\partial a_i}{\partial u_{il}}}_{z_l}$$

$$\boxed{\frac{\partial a_i}{\partial u_{il}} = \frac{\partial}{\partial u_{il}} \left(\sum_l u_{il} \cdot z_l \right) = z_l}$$

$$\frac{\partial |y-\hat{y}|^2}{\partial u_{il}} = \delta_i z_l$$

Now, $\delta_i = \dfrac{\partial (y-\hat{y})^2}{\partial a_i} = \sum \dfrac{\partial |y-\hat{y}|^2}{\partial a_j} \cdot \dfrac{\partial a_j}{\partial a_i}$

$$\Rightarrow \delta_i = \sum_j \delta_j \underbrace{\frac{\partial a_j}{\partial z_i}}_{u_{ji}} \cdot \underbrace{\frac{\partial z_i}{\partial a_i}}_{\sigma'(a_i)}$$

$$\delta_i = \sum_j \delta_j u_{ji} \sigma'(a_i)$$

$$\boxed{\delta_i = \sigma'(a_i) \sum_j \delta_j u_{ji}}$$

Scanned with CamScanner

Layer K

$$\delta_K = \frac{\partial |y-\hat{y}|^2}{\partial a_K}$$

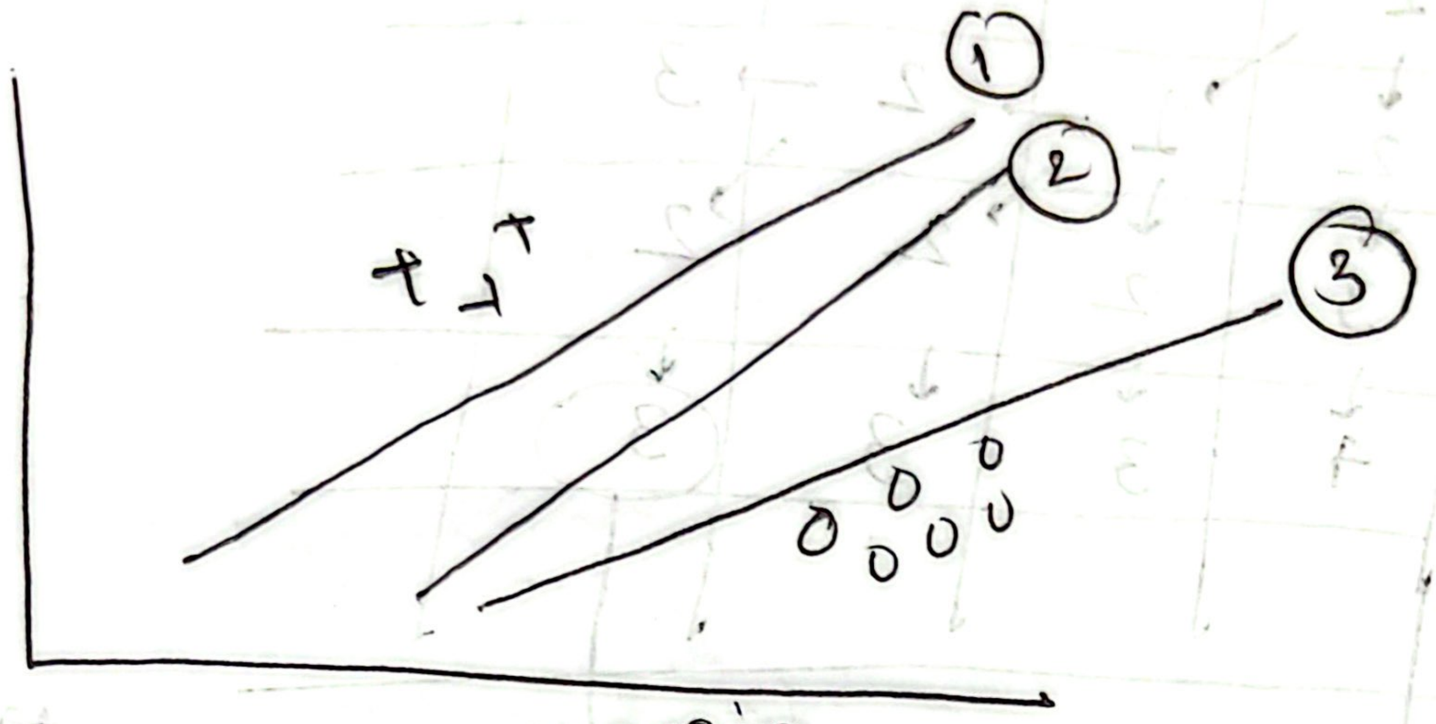
$$= \frac{\partial |y-\hat{y}|^2}{\partial \hat{y}}$$

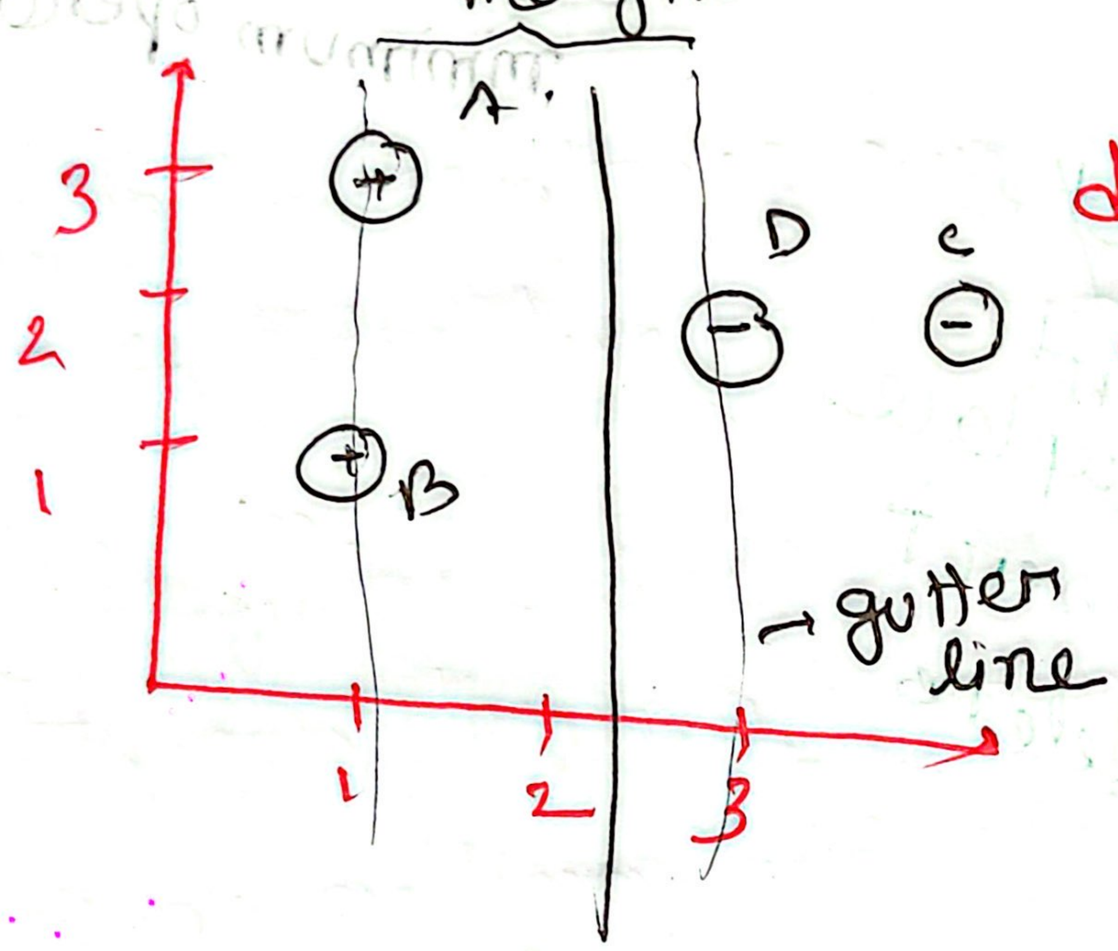
$$\delta_K = 2|y-\hat{y}| \quad \checkmark$$

Scanned with CamScanner

L-07

SVM – Maximum margin classifier.





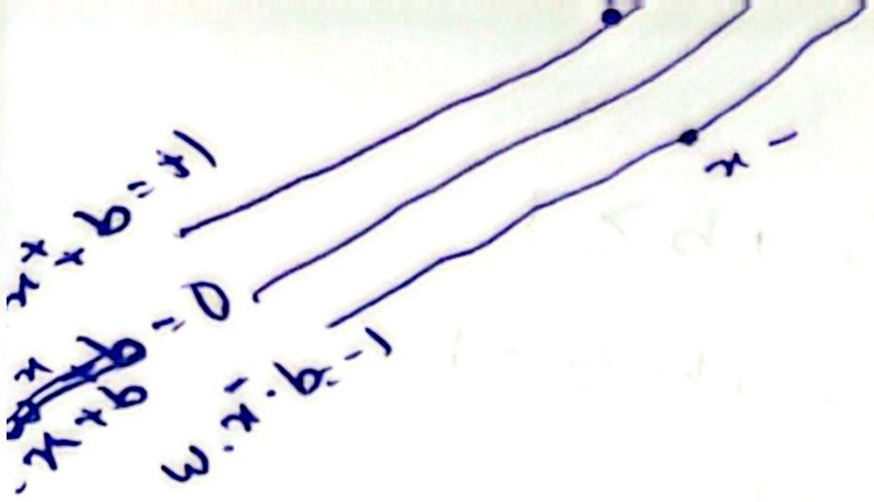
data point → vectors
nearest এ point থাকে। নাম decision boundary draw করা হয় তাদের support vector.

① Gutter: A line through closest '+' and '−' points near db ↳ gutterline

* parallel to db

② Support Vector: points that decide db.

③ Margin: distance betⁿ two gutter.



$$\boxed{w^T x + b = 0}$$

↳ db

$w \cdot x^+ + b = +1 \Rightarrow w \cdot x^+ + b - 1 = 0$ — (i)

$w \cdot x^- + b = -1 \Rightarrow w \cdot x^- + b + 1 = 0$ — (ii)

(i) − (ii) ⇒

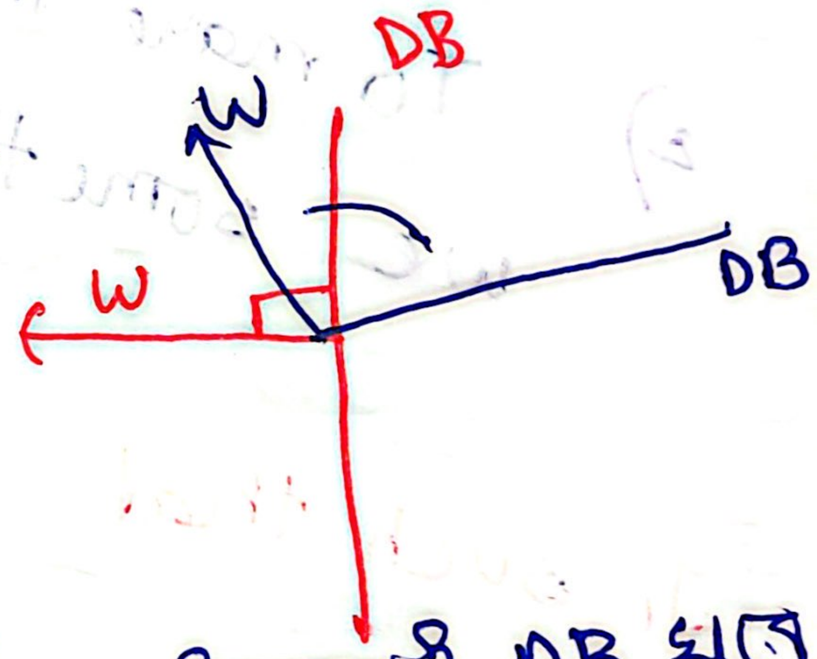
$w \cdot x^+ + b - 1 = w \cdot x^- + b + 1$

$\Rightarrow w \cdot (x^+ - x^-) = 2$

$$\boxed{M = (x^+ - x^-) \cdot \frac{w}{\|w\|}}$$ → eq^n of margin.

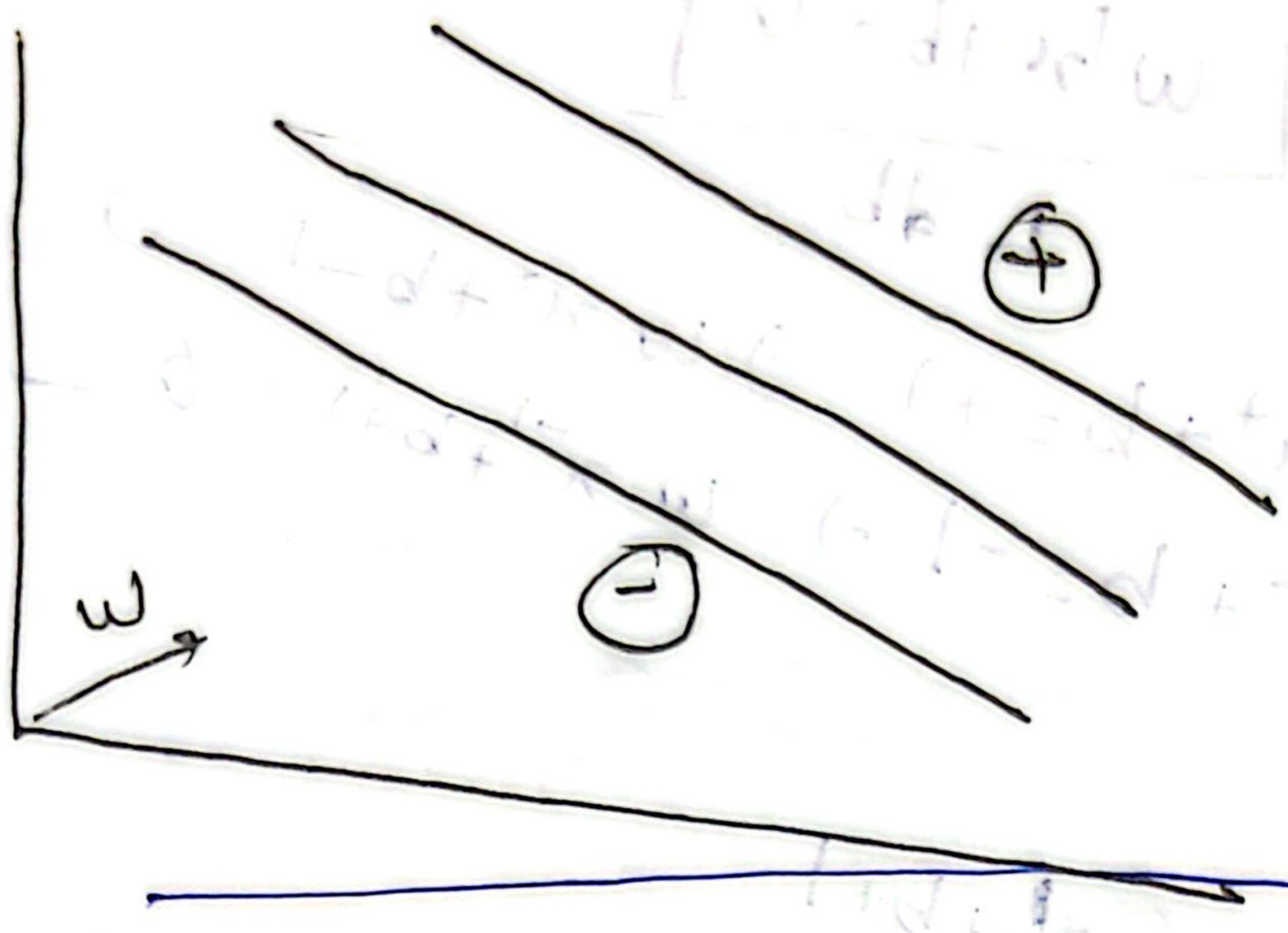
$$\boxed{M = \frac{2}{\|w\|}}$$ → maximise the margin

* Decision Boundary এর সাথে weight perpendicular হবে



এইভাবেই DB খুঁজে and always perpendicular হবে

Part of SVM



$\bar{w} \cdot \bar{x}_+ + b \geq 1$

$\bar{w} \cdot \bar{x}_- + b \leq -1$

44.09

$$\boxed{\begin{aligned} w \cdot x_+ + b &\geq 1 \rightarrow \text{positive} \\ w \cdot x_- + b &\leq -1. \end{aligned}}$$

1) carrying two equation in pain. Thats why we have to introduce new variable to make life easier.

2) To make the math convinient.

we sometimes add additional variables.

y_i such that $y_i = +$ for +ve samples

$y_i = -$ " −ve "

Decision Rule

$$\boxed{y_i(\bar{w} \cdot \bar{x}_i + b) \geq 1}$$

we want to maximize the margin

কাত

$$\max \frac{2}{\|w\|} \sim \max \frac{1}{\|w\|} \sim \min \|w\| \sim \min \frac{1}{2} \|w\|^2$$

We have to use lagrange multiplier,

*** How lagrange multiplier works** → Quer এসে।

1) $f(x,y) = 2 - x^2 - 2y^2$

2) $g(x,y) = x + y - 1 = 0$

$$L(x,y,\lambda) = f(x,y) - \lambda g(x,y)$$
$$= 2 - x^2 - 2y^2 - \lambda(x+y-1)$$

$$\nabla L(x,y,\lambda) = \nabla f(x,y) - \lambda \nabla g(x,y) = 0$$

$$\frac{\partial}{\partial x} L(x,y,\lambda) = -2x - \lambda = 0 \quad \text{— (i)}$$

$$\frac{\partial}{\partial y} L(x,y,\lambda) = 4y - \lambda = 0 \quad \text{— (ii)}$$

$$\frac{\partial}{\partial \lambda} L(x,y,\lambda) = x + y - 1 \quad \text{— (iii)}$$

$$x = \frac{2}{3}, \quad y = \frac{1}{3}$$

$$\lambda = -\frac{4}{5} \qquad f = \frac{4}{3}$$

বাকি math গুলো

L-08

minimize

$$L(w) = \frac{\|\vec{w}\|^2}{2} \rightarrow f(x)$$

subject to

$$y_i(\vec{w} \cdot \vec{x}_i + b) \geq 1)$$

Using lagrange function:

$$L_p = \frac{\|\vec{w}\|^2}{2} + \sum_{i=1}^{N} \lambda_i (y_i(wx_i + b) - 1)$$

$x \rightarrow$ input

$y \rightarrow$ sign

$\lambda \rightarrow$ multiplier variable

so, এখানে $\partial/\partial w, b$

minimize করে maximize করা হয়

$L_p \rightarrow$ primal $\rightarrow$ minimize করে maximize করা হয়

dual $\rightarrow$

New constraints are,

$$\frac{\partial L_p}{\partial \vec{w}} = w = \sum_{i=1}^{N} \lambda_i y_i x_i$$

$$\frac{\partial L_p}{\partial b} \Rightarrow \sum_{i=1}^{N} \lambda_i y_i = 0$$

still not solvable অনেক variable

From KKT transform,

① $\lambda_i \geq 0$

② $\lambda_i [y_i (\vec{w} \vec{x}_i + b) - 1] = 0$

① $\lambda_i \neq 0$, যেই value গুলো হলো support vector

* put $\vec{w} = \sum_{i=1}^{N} \lambda_i y_i \vec{x_i}$ and $\sum_{i=1}^{N} \lambda_i y_i = 0$

in $L_p = \frac{\|\vec{w}\|^2}{2} - \sum_{i=1}^{N} \lambda_i (y_i (\vec{w} \cdot x_i + b) - 1)$

$$L_p = \frac{\|\vec{w}\|^2}{2} - \sum_{i=1}^{N} \lambda_i (y_i (\vec{w} \cdot \vec{x_i} + b) - 1)$$

$$= \frac{\|\vec{w}\|^2}{2} - \sum_{i=1}^{N} \lambda_i y_i \vec{w} \vec{x_i} - \sum_{i=1}^{N} \lambda_i y_i b + \sum_{i=1}^{N} \lambda_i$$

$$= \frac{\|\vec{w}\|^2}{2} - \vec{w} \sum_{i=1}^{N} \lambda_i y_i x_i - b \sum_{i=1}^{N} \lambda_i y_i + \sum_{i=1}^{N} \lambda_i$$

$$= \frac{\|\vec{w}\|^2}{2} - \vec{w} \cdot \vec{w} - b \times 0 + \sum_{i=1}^{N} \lambda_i$$

$$= \frac{\vec{w} \cdot \vec{w}}{2} - \vec{w} \cdot \vec{w} + \sum_{i=1}^{N} \lambda_i$$

$$\left[\vec{w} = \sum_{i=1}^{N} \lambda_i y_i \vec{x} \qquad \sum_{i=1}^{N} \lambda_i y_i = 0 \right]$$

$$= \sum_{i=1}^{N} \lambda_i - \frac{\vec{w} \cdot \vec{w}}{2}$$

$$= \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i=1}^{N} \lambda_i y_i \vec{x} \cdot \sum_{j=1}^{N} \lambda_j y_j \vec{x_j}$$

$$= \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j} \lambda_i y_i \vec{x_i} \cdot \lambda_j y_j \vec{x_j}$$

$\boxed{x_i \cdot x_j}$

দুইটি data dot prod kernal (বলা যাবে

$\boxed{\text{Quadratic Problem solver}}$

$\lambda \rightarrow$ Lagrange multiplier

$$L_D = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j$$

After solving α_i

$$\frac{\partial L_P}{\partial w} = 0$$

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i$$

$$\boxed{\vec{w} \cdot \vec{x_i} + b = 1}$$ → এখান থেকে bias পাবো

Math - SVM

A_1	A_2	y	α_i
0.38	0.47	+	65.52
0.49	0.61	−	65.52
0.92	0.41	−	0
0.74	0.89	−	0
0.18	0.58	+	0
0.41	0.35	+	0
0.93	0.81	−	0
0.21	0.10	+	0

যাদের এর value 0 না, তারা support vector.

Scanned with CamScanner

$\cancel{w_1 = \sum \lambda_i y_i x_i}$

$\cancel{= 65.52 \times 1 \times 0.38 +}$

The first two tuples are support vectors in this case.

Let $w = (w_1, w_2)$ and b denote the parameter to be determined.

$$w_1 = \sum_i \lambda_i y_i x_{i1} = 65.52 \times 1 \times 0.38 + 65.52 \times -1 \times 0.49$$

$$= -6.64$$

$$w_2 = \sum \lambda_i y_i x_{i2} = 65.52 \times 1 \times 0.47 + 65.52 \times -1 \times 0.61$$

$$= -9.32$$

The parameter b can be calculated for each support vector as follows

$$b_1 = 1 - wx_1$$

$$= 1 - (-6.64) \times 0.38 - (-9.32) \times 0.47$$

$$= 7.93$$

$$b_2 = 1 - wx_2$$

$$= 1 - (-6.64) \times 0.48 - (-9.32) \times 0.61$$

$$= 7.93$$

Scanned with CamScanner

Averaging this value b_1 and b_2

we get $b = 7.93$

$w_1 = -6.64$

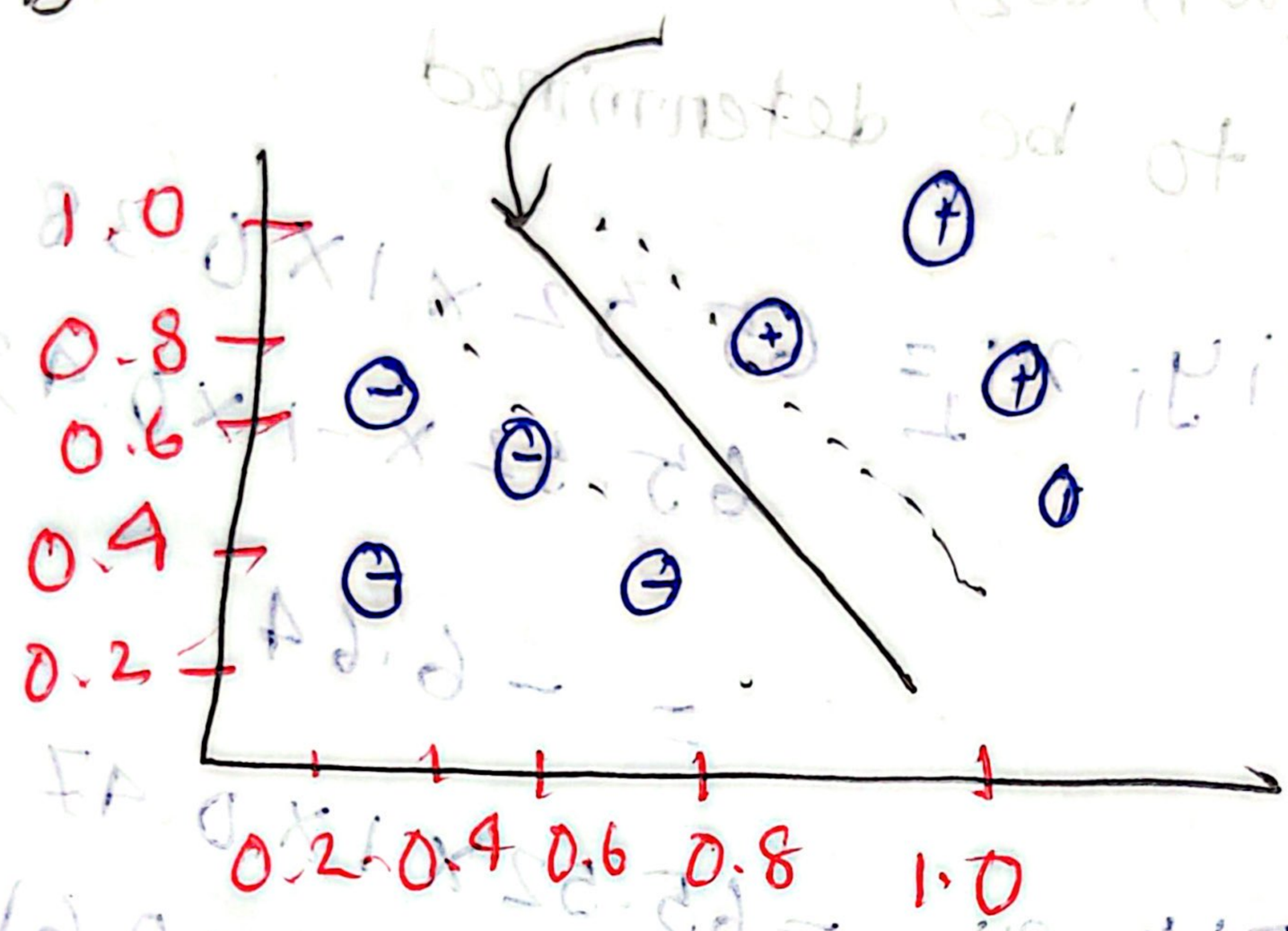
$w_2 = -9.32$

$b = +7.93$

margin

MMH

$-6.64x_1 - 9.32x_2 + 7.93 = 0$



Test data $x = (0.5, 0.5)$

$$g(x) = w.x + b$$
$$= -6.64(0.5) - 9.32(0.5) + 7.93$$
$$= -0.05$$
$$= -ve$$

Test data goes to negative class

goes to under MMH.

L-09

Soft Margin classification.

optimization criterion be

Minimize

$$\boxed{\frac{1}{2} w \cdot w + C \sum_{k=1}^{R} \varepsilon_k}$$ → soft margin classifier.

Hard margin

$\phi(w) = \frac{1}{2} w^T w$ is minimized for all $\{(x_i, y_i)\}$

$y_i (w^T x_i + b) \geq 1$

Soft margin

$\phi(w) = \frac{1}{2} w^T w + C \sum \varepsilon_i$ is minimized for all $\{(x_i, y_i)\}$

$y_i (w^T x_i + b \geq 1 - \varepsilon_i$ and $\varepsilon_i \geq 0$ for all i

The kernel trick

প্রতিটা point কে φ function এ না দিয়ে, এমন একটা function $k(x_i, x_j)$, kernel function directly এই data point কে higher dimension এ separate করে দেবে আলাদা করে φ এর মান লাগবে না। and অবশ্যই দুই feature space কে dot product/inner product

Example

2 dimensional vector $x = [x_1, x_2]$

let $K(x_i, x_j) = (1 + x_i^T x_j)$ → kernal function

Need to show that $k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$

$$K(x_i, x_j) = (1 + x_i^T x_j)^2$$

$$= 1 + x_{i1}^2 x_{j1}^2 + 2x_{i1} 2x_{j1} x_{i2} x_{j2} + x_{i2}^2 x_{j2}^2 + 2x_{i1} x_{j1} + 2x_{i2} x_{j2}$$

$$= \left[1 \;\; x_{i1}^2 \;\; \sqrt{2}\, x_{i1} x_{i2} \;\; x_{i2}^2 \;\; \sqrt{2}\, x_{i1} \;\; \sqrt{2}\, x_{i2}\right]$$

$$\left[1 \;\; x_{j1}^2 \;\; \sqrt{2}\, x_{j1} x_{j2} \;\; x_{j2}^2 \;\; \sqrt{2}\, x_{j1} \;\; \sqrt{2}\, x_{j2}\right]$$

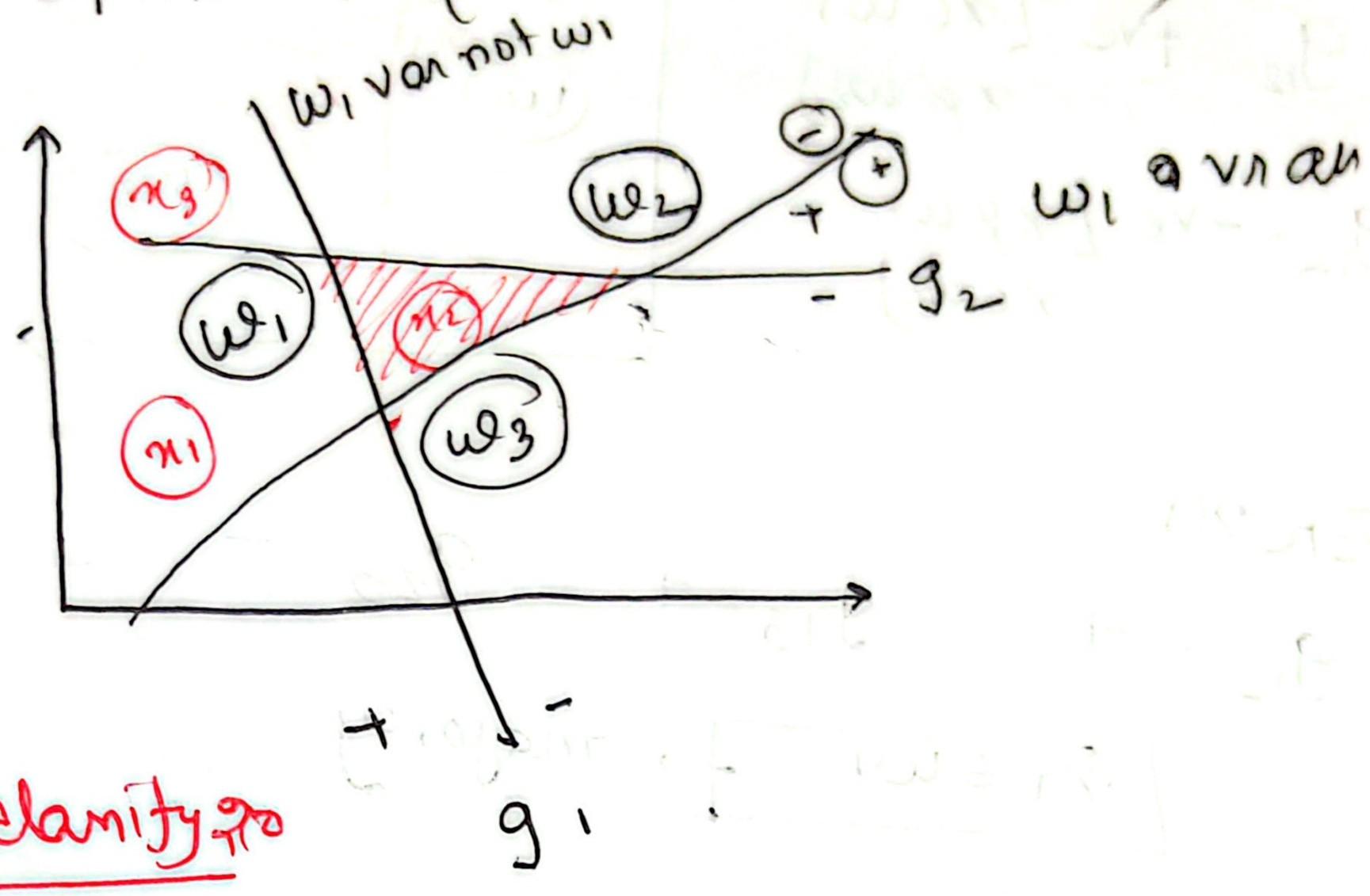
$$= \varphi(x_i)^T \varphi(x_j)$$

where $\varphi(x) = [1 \;\; x_1^2 \;\; \sqrt{2}\, x_1 x_2 \;\; x_2^2 \;\; \sqrt{2}\, x_1 \;\; \sqrt{2}\, x_2$

Multiclass problem

one vs rest

Option 1 ω_i vs $\bar{\omega}_i$ (ω_i vs not ω_i / all)



* x_1 কিভাবে classify হবে

For x_1 $g_1 = +$ $g_3 = -$

$g_2 = -$

$\therefore x_1 \in \omega_1$

For x_2

$g_1 = -$

$g_2 = -$

$g_3 = -$

$x_2 \in$ undefined area

For x_3, $g_1 = +$, $g_2 = +$ $g_3 = -$

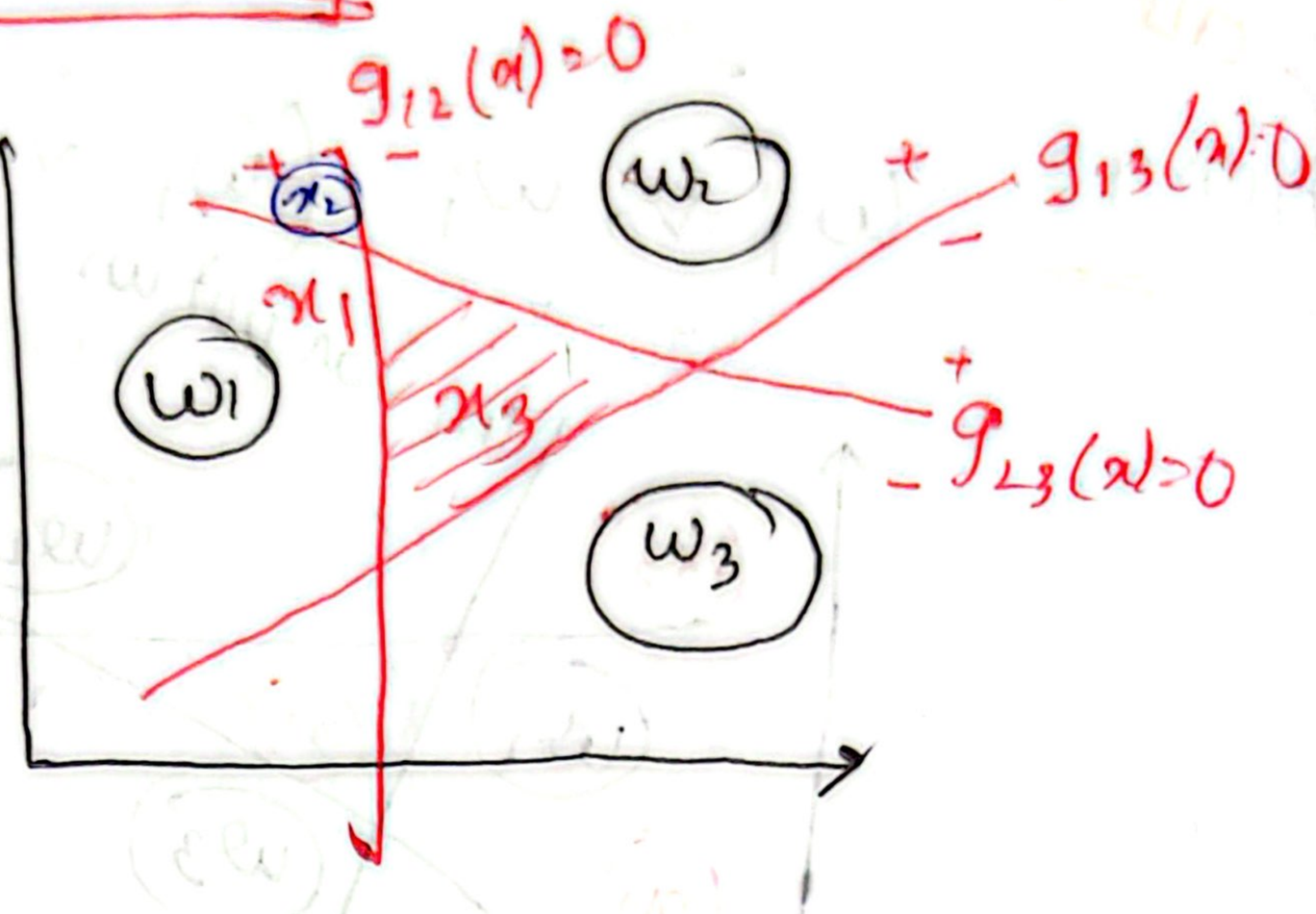
2 classes claim x_3

$x_3 \in$ confusion area

Option 2 $\boxed{\omega_1 \text{ vs } \omega_2}$

$g_{12} = +ve \; [x \in \omega_1, \; x \notin \omega_2]$

$g_{12} = -ve \; [x \notin \omega_1, \; x \in \omega_2]$



For x_1

$g_{12} = +$ $\quad g_{13} = +$ $\quad g_{23} = -$

$\boxed{x_1 \in \omega_1} \rightarrow$ majority

For x_2, $g_{12} = +$ $\quad g_{13} = +$ $\quad g_{23} = +$

$\therefore \boxed{x_2 \in \omega_1}$

আমরা confusion দূর করতে পেরেছি

For x_3, $g_{12} = -$

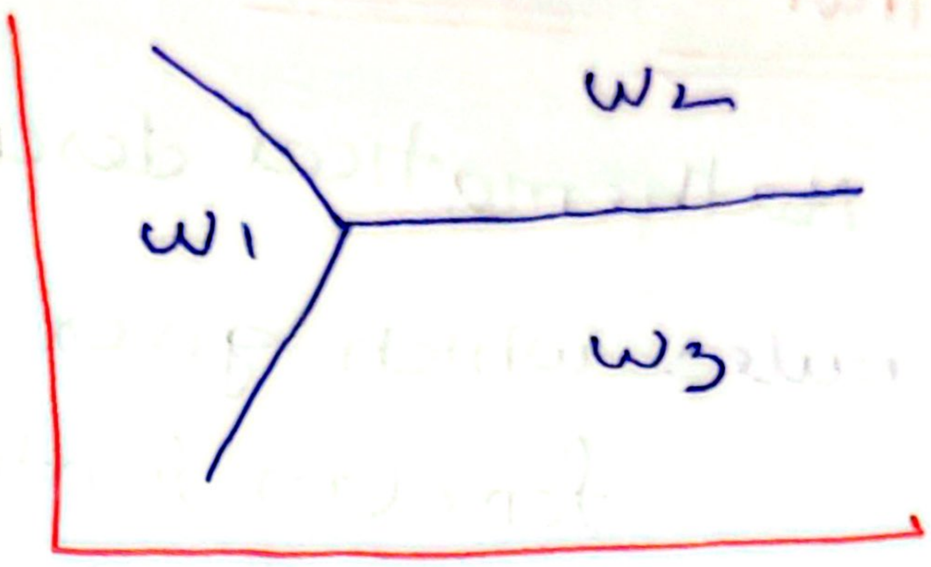
$g_{13} = +$

$g_{23} = -$

$x_3 \rightarrow$ Undefined

Option-3 ← Linear machine

* There is no undefined but has a tie point



Decision rule:

if $g_i > g_j$ for all $j \neq i$

then $x \in w_i$

General discriminant function for multiclass:

Decision rule:

If $g_1 > g_2$ then $x \in w_1$ else $x \in w_2$

$$g_1(x) = w_1^T x + {}^1w_0$$

$$g_2(x) = w_2^T x + {}^2w_0$$

$[g_1 = 0$
$g_2 = 0]$

There will be a tie if $g_1 = g_2$

$$\Rightarrow \boxed{(w_1^T - w_2^T)x + {}^1w_0 - {}^2w_0 = 0}$$

↳ General equation