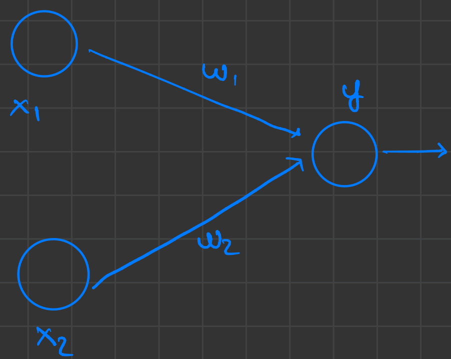
Neural Network

→ learn patterns and relationships in data

→ NN learning rules = algorithms

Simple artificial Neural Network



weights :–

excitative :– +ve

inhibitory :– -ve

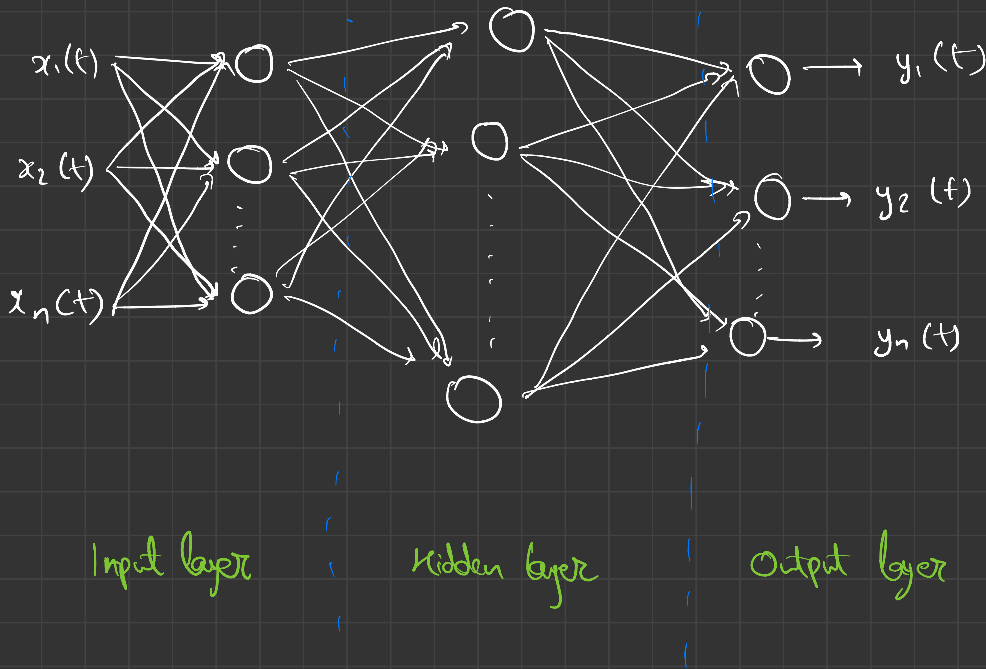
inputs

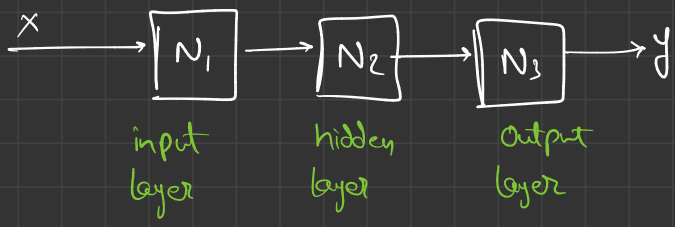
$$x_1w_1 + x_2w_2 + x_3w_3 + \cdots + x_nw_n = y_{in}$$

Basically,

$$x_1w_1 + x_2w_2 = y_{in} \rightarrow \boxed{} \rightarrow$$ firing or non firing of neuron

Multi Layered Neural Network (MNN)





Why ANN

→ massive parallelism (many operations simultaneously)
→ Adaptivity

Biological Neural Network vs ANN

Soma ⟷ Y (Σ)

Dendrite ⟷ Weights (w)

Axon ⟷ Output

Q Compare ANN with BNN

Binary Sigmoidal Function

$$f(x) = \frac{1}{1 + e^{-\sigma x}}$$

Bipolar sigmoidal function

$$b(x) \quad 2f(x) - 1 \Rightarrow 2\left(\frac{1}{1+e^{-\sigma x}}\right) - 1$$

$$b(x) = \frac{1 - e^{-\sigma x}}{1 + e^{-\sigma x}}$$

σ = steepness parameter

Bias

$$y_{net} = \sum_{i=1}^{n} x_i w_i + b$$

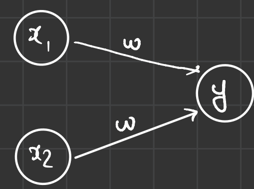
Q. WAP to generate following activation func.

i) Logistic
ii) Hyperbolic tangent
iii) Linear.

LAB

$\boxed{\text{XOR}}$

x_1	x_2	y	y_{net}
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	2



$\theta \geq 1$

$$kw \geq \theta \geq (k-1)w$$

k = no. of inputs
w = weight

$w=1$, $\quad 2 \geq 1 \geq 1$

Feed Forward Networks

Back Propagation Network

(Question in final exam)

Find gradient of error wrt weights.

$$\Delta w \propto \frac{\partial e}{\partial w}$$

Training the Algorithms

1. Initialization of weights & bias
2. Feed Forward (Generate the actual output)
3. Back propagation of errors
4. Updation of weights and biases

Q. input pattern $[0.6 \quad 0.8 \quad 0]$
target output $= 0.9$
learning rate $\alpha = 0.3$

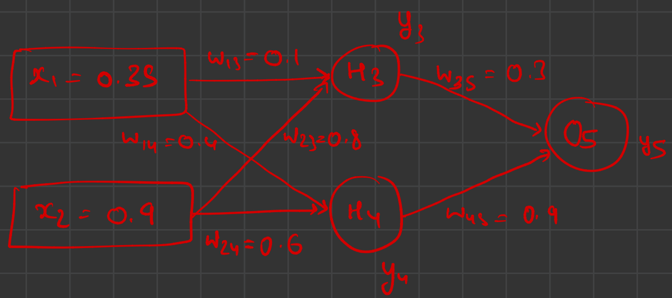
A:- Initialize weigh & bias,

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

$$W_0 = [-1]$$

$$V = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array} \begin{array}{c} \begin{array}{ccc} z_1 & z_2 & z_3 \end{array} \\ \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 2 \\ 0 & 3 & 1 \end{bmatrix} \end{array} \qquad V_0 \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}$$

Q.



A:-

$$a_j = \sum_i (w_{ij} \times x_i) \qquad y_j = f(a_j) = \frac{1}{1+e^{-a_j}}$$

$$a_1 = (w_{13} \times x_1) + (w_{23} \times x_1) = 0.755$$

$$y_3 = \frac{1}{1+e^{-0.755}} = 0.68$$

$$a_2 = (w_{14} \times x_1) + (w_{24} \times x_2) = 0.68$$

$$y_4 = f(a_2) = \frac{1}{1+e^{-0.68}} = 0.6637$$

$$a_3 = (w_{35} \times y_3) + (w_{45} \times y_4) = 0.801$$

$$y_5 = f(a_3) = \frac{1}{1+e^{-0.801}} = 0.69$$

$$\text{Error} = \underset{(.5)}{y_{target}} - \underset{(.69)}{y_5} = -0.19$$

Each weight changed by :-

$$\Delta w_{ji} = \eta \, \delta_i \, O_i$$

learning rate

change

output

$$\delta_j = O_j (1 - O_j)(t_j - O_j)$$

$$\delta_j = O_j (1 - O_j) \sum_k \delta_k w_{kj}$$

For output :-

$$\delta_5 = y(1-y)(y_{target} - y)$$

output × (1 − output) × error

$$\delta_5 = 0.69(1-0.69)(0.5-0.69) = -0.0406$$

For hidden layer :-

$$\delta_3 = y_3(1-y_3) \, w_{35} \times \delta_5$$
$$= 0.68(1-0.68)(0.3-0.0406) = -0.00265$$

$$\delta_4 = y_4(1-y_4) \, w_{45} \times \delta_5 = -0.0082$$

Change in weights

$$w_{ji} = \eta \, \delta_j \, O_i$$

$$\Delta w_{45} = \eta \delta_5 y_4 = -0.0269$$

$$w_{45}(\text{new}) = w_{45}(\text{old}) + \Delta w_{45} = 0.08731$$

$$\Delta w_{14} = \eta \delta_4 x_1 = -0.00287$$

$$w_{14}(\text{new}) = w_{14}(\text{old}) + \Delta w_{14} = 0.3971$$

Crisp logic

→ Definite
→ Yes or no
→ " Temperature is 25°C "

Universe of discourse

contains all possible elements with specific characteristics

Max - min composition

$$R = \begin{array}{c|c} & \begin{matrix} 1 & 3 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 3 \\ 5 \end{matrix} & \begin{bmatrix} \boxed{0 \quad 1 \quad 0} \\ 0 \quad 0 \quad 1 \\ 0 \quad 0 \quad 0 \end{bmatrix} \end{array} \qquad S = \begin{array}{c|c} & \begin{matrix} 1 & 3 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 3 \\ 5 \end{matrix} & \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \end{array}$$

Max - min :—

$\min\{0\ 0\ 0\}$

$\max = 0$

$$R \circ S = \begin{array}{c|c} & \begin{matrix} 1 & 3 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 3 \\ 5 \end{matrix} & \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{array}$$

$\min\{0, 1, 0\}$

$\max = 1$

$$[0 \quad 0 \quad 0 \quad 0.9 \quad 1] \times \begin{bmatrix} 0 & \cdots \\ 1 & \cdots \\ 1 & \cdots \\ 0.3 & \cdots \\ 1 & \cdots \end{bmatrix}$$

min min min min min min

$\max\{0, 0, 0, 0.3, 1\}$

$\max = \{1, \ldots\}$

CRISP LOGIC

$\boxed{P \Rightarrow Q}$ can be given by $\boxed{\sim P \vee Q}$

Hint :- $P \Rightarrow Q$ is F only for T F

$$R = (A \times B) \cup (\bar{A} \times Y)$$

Q. $(\sim(P \wedge Q) \Rightarrow R) \wedge P \wedge Q$ Simplify

A. Using formula, $P \rightarrow Q : \sim P \vee Q$

$\Rightarrow (\sim\sim(P \wedge Q) \vee R) \wedge P \wedge Q$

$\Rightarrow ((P \wedge Q) \vee R) \wedge (P \wedge Q)$

Here $(A \vee B) \wedge A = A$

$\therefore \Rightarrow \underline{P \wedge Q}$

QUANTIFIER

All (universal) $\forall \rightarrow \Rightarrow$

Some (existential) $\exists \rightarrow \wedge$

Fuzzy Logic $T(\tilde{P})$

Fuzzy set always in pairs

$-$	Negation	$T(\bar{\tilde{P}})$	$1 - T(\tilde{P})$
$\wedge$	Conjuction	$\tilde{P} \wedge \tilde{Q}$	$\min(T(\tilde{P}), T(\tilde{Q}))$
$\vee$	Disjunction	$\tilde{P} \vee \tilde{Q}$	$\max(T(\tilde{P}), T(\tilde{Q}))$
$\Rightarrow$	Implication	$\tilde{P} \Rightarrow \tilde{Q}$	$\bar{\tilde{P}} \vee \tilde{Q} = \max(1 - T(\tilde{P}), T(\tilde{Q}))$

$$R = (\tilde{A} \times \tilde{B}) \cup (\bar{\tilde{A}} \times Y)$$

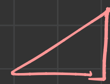
$$\Rightarrow \quad \mu_{\tilde{R}}(x, y) = \max((\min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)), 1 - \mu_{\tilde{A}}(x))$$

Matrix $\tilde{A} \times \tilde{B}$ (min values from set)

$$\tilde{R} = (\tilde{A} \times \tilde{B}) \cup (\bar{\tilde{A}} \times \tilde{C})$$

$$\mu_{\tilde{R}}(x, y) = \max(\min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)), \min(1 - \mu_{\tilde{A}}(x), \mu_{\tilde{C}}(y)))$$

In Universe of discourse, take all membership values as 1.

Centroid of triangle =  $\Rightarrow \frac{2}{3} \times l$

 $\Rightarrow \frac{1}{3} \times l$

Centroid of rectangle =  $\Rightarrow \frac{1}{2} \times l$

Note :- Take centroid from origin

↳ so add lengths from origin till centroid.

UNIT - 4

LINEAR REGRESSION

Formulae

$$r = r_{xy} = \frac{Cov(x, y)}{S_x * S_y}$$

$$Cov(x,y) = r(S_x * S_y)$$

$$Cov(x,y) = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{n} = \frac{\Sigma x_i y_i}{n} - \bar{x}\bar{y}$$

$$S_x = \sqrt{\frac{\Sigma(x_i - \bar{x})^2}{n}} = \sqrt{\frac{\Sigma x_i^2}{n} - \bar{x}^2}$$

$$S_y = \sqrt{\frac{\Sigma(y_i - \bar{y})^2}{n}} = \sqrt{\frac{\Sigma y_i^2}{n} - \bar{y}^2}$$

$$r = \frac{n\Sigma x_i y_i - \Sigma x_i \times \Sigma y_i}{\sqrt{n\Sigma x_i^2 - (\Sigma x_i)^2}\sqrt{n\Sigma y_i^2 - (\Sigma y_i)^2}}$$

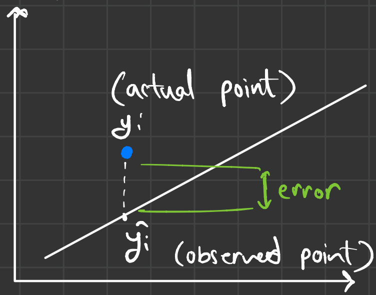
y depends on x , y on x ⇒

$$y = a + bx$$

regression coefficient of y on x

b_{yx}

For y on x



error ⇒

$$e_i = y_i - \hat{y}_i$$

$$b = \frac{Cov(x, y)}{S_x^2} = \frac{r S_x S_y}{S_x^2} = \frac{r S_y}{S_x}$$

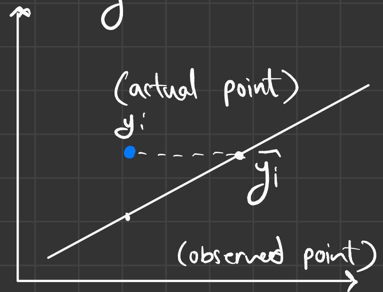
$$\bar{y} = a + b\bar{x}$$

$$a = \bar{y} - b\bar{x}$$

mean of x

mean of y

For x on y



$$e = y_i - \hat{y}_i$$

Example :-

x	2	4	5	5	8	10
y	6	7	9	10	12	12

estimate y when x is 13 & x when y is 15

y on x $\qquad$ x on y

A:-

x_i	y_i	$x_i y_i$	x_i^2	y_i^2
2	6	12	4	36
4	7	28	16	49
5	9	45	25	81
5	10	50	25	100
8	12	96	64	144
10	12	120	100	144
34	56	351	234	554

mean $\rightarrow \bar{x} = \underline{5.6667}$

mean $\rightarrow \bar{y} = \underline{9.3333}$

$S_x^2 = \frac{\sum x_i^2}{n} - \bar{x}^2$

$S_y^2 = \frac{\sum y_i^2}{n} - \bar{y}^2$

$y = a + bx$

$x = a' + b'x$

don't take same variables

M2 — back propagation, fuzzy, linear till today 2nd sum

Regression alternate way (Easier way)

Q. x	y	$x-\bar{x}$	$y-\bar{y}$	$(x-\bar{x})^2$	$(x-\bar{x})(y-\bar{y})$	(y_p) predicted output
1	3	-2	-0.6	4	1.2	2.8
2	4	-1	0.4	1	-0.4	3.2
3	2	0	-1.6	0	0	3.6
4	4	1	0.4	1	0.4	4
5	5	2	1.4	4	2.8	4.4
3	3.6	0	0	10	4	

$$y = mx + b$$

slope

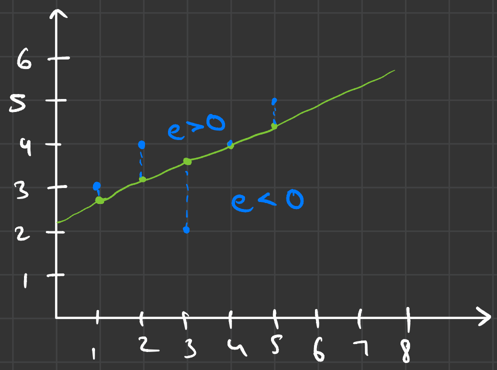
$$m = \frac{\Sigma (x-\bar{x})(y-\bar{y})}{\Sigma (x-\bar{x})^2} = \frac{4}{10} = 0.4$$

$$\bar{y} = m\bar{x} + b$$

$$b = \bar{y} - m\bar{x}$$
$$= 3.6 - 1.2$$
$$b = 2.4$$

$$\hat{y}_i = mx_i + b$$

Predicted output



— → $x-\hat{y}$ line → predicted output

• → actual points (y)

R-square value gives the best fit line to minimize the error.

Closer to 1 is good

R- square $\quad R^2 = \dfrac{\sum (y_p - \bar{y})^2}{\sum (y - \bar{y})^2}$

predicted outputs

→ Estimation of table

$y_p - \bar{y}$	$(y-\bar{y})^2$	$(y_p-\bar{y})^2$
-0.8	0.36	0.64
-0.4	0.16	0.16
0	2.56	0
0.4	0.16	0.16
0.8	1.96	0.64
	5.2	1.6

$$R^2 = \frac{\sum (y_p - \bar{y})^2}{\sum (y - \bar{y})^2} = \frac{1.6}{5.2} = 0.3$$

Multiple Regression

$$y = B_0 + \overbrace{B_1 x_1 + B_2 x_2 + \dots\dots + B_k x_k}^{\text{independent variable / predicted variable / Regression variable}} + E$$

dependent variable / Target variable / Criterion variable

where, $B_0 \rightarrow$ y intercept

$B_1, B_2 \dots B_k \rightarrow$ regression coefficients

$\varepsilon \rightarrow$ random error

Only 2 independent variables for exam.

So,
$$\bar{y} = B_0 + B_1 \bar{x}_1 + B_2 \bar{x}_2$$

$$B_1 = \frac{(\Sigma x_2^2)(\Sigma x_1 y) - (\Sigma x_1 x_2)(\Sigma x_2 y)}{(\Sigma x_1^2)(\Sigma x_2^2) - (\Sigma x_1 x_2)^2}$$

$$B_2 = \frac{(\Sigma x_1^2)(\Sigma x_2 y) - (\Sigma x_1 x_2)(\Sigma x_1 y)}{(\Sigma x_1^2)(\Sigma x_2^2) - (\Sigma x_1 x_2)^2}$$

Q. Home price in Mumbai

Area	Bedroom	Age	Price
2600	3	20	565 000
3000	4	15	610 000
3200	3	18	595 000
3600	3	30	600 000
4000	5	8	760 000

$$\text{Price} = b + m_1 \times \text{area} + m_2 \times \text{bedrooms} + m_3 \times \text{Age} + \varepsilon$$

dependent on age, bedroom & area

0

Q. Predict the values of y given x_1 & x_2

Subject	y	x_1	x_2	x_1^2	x_2^2	$x_1 x_2$	$x_1 y$	$x_2 y$
1	-3.7	3	8	9	64	24	11.1	39.6
2	3.5	4	5	16	25	20	14	17.5
3	2.5	5	7	25	49	35	12.5	17.5
4	11.5	6	3	36	9	18	69	34.5
5	5.7	2	1	4	1	2	11.4	5.7
6	2	3	2					

problem

$\bar{y} = 19.5$
$\bar{x}_1 = 20$
$\bar{x}_2 = 24$

$$B_1 = \frac{(\Sigma x_2^2)(\Sigma x_1 y) - (\Sigma x_1 x_2)(\Sigma x_2 y)}{(\Sigma x_1^2)(\Sigma x_2^2) - (\Sigma x_1 x_2)^2} = 2.74$$

$$B_2 = \frac{(\Sigma x_1^2)(\Sigma x_2 y) - (\Sigma x_1 x_2)(\Sigma x_1 y)}{(\Sigma x_1^2)(\Sigma x_2^2) - (\Sigma x_1 x_2)^2} = -1.52$$

$$y = B_0 + B_1 x_1 + B_2 x_2$$

$$\bar{y} = B_0 + B_1 \bar{x}_1 + B_2 \bar{x}_2$$

$$B_0 = 0.25$$

$y = ?$

$x_1 = 3$
$x_2 = 2$

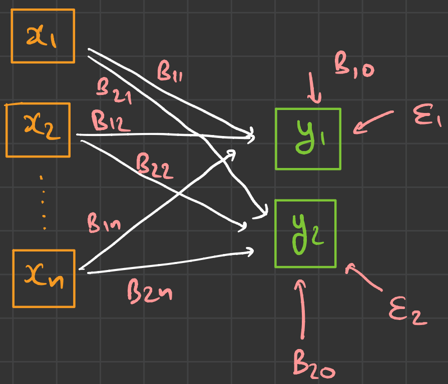
$$y = B_0 + B_1 x_1 + B_2 x$$

$$y = 5.43$$

Multivariate Linear Regression

(multiple output - multiple linear regression)

Example



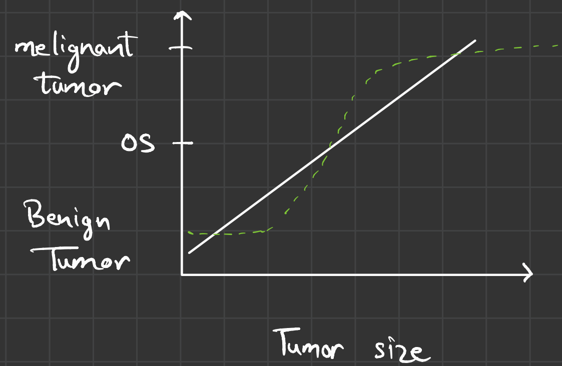
$$y_1 = B_{10} + B_{11}x_1 + B_{12}x_2 + \dots + B_{1n}x_n + \varepsilon_1$$

$$y_2 = B_{20} + B_{21}x_1 + B_{22}x_2 + \dots + B_{2n}x_n + \varepsilon_2$$

Logistic Regression

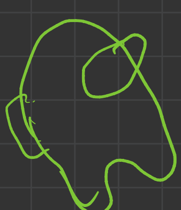
Binary classifier (Classify into two groups)

Example :-



maximum likelihood :- performance index (like R^2)

→ should cover maximum samples



Hypothesis representation

$$h_\theta(x) = \theta^T x$$

where, θ = model parameter (slope)
x = idependent variable
$h_\theta(x)$ = prediction function

$$0 < h_\theta(x) < 1$$

$$h_\theta(x) = g(\theta^T x)$$

$$g(z) = \frac{1}{1+e^{-z}}$$

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$$

$h_\theta(x)$ is the estimated probability that $y = 1$ on input x.

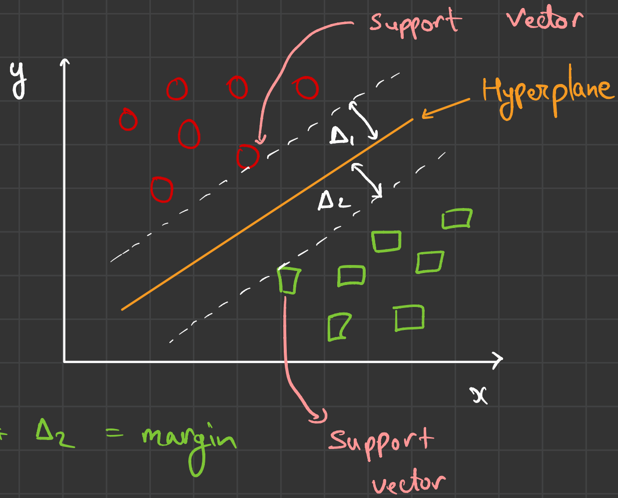
$$h_\theta(x) = P(y=1 / x : \theta)$$

$$P(y=1 / x:\theta) + P(y=0 / x:\theta) = 1$$

$$P(y=0 / x:\theta) = 1 - P(y=1 / x:\theta)$$

↳ Hypothesis representation

Support Vector Machine

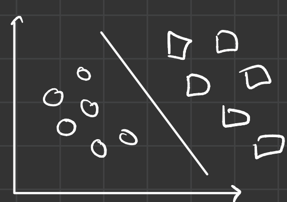


$\Delta_1 + \Delta_2$ = margin

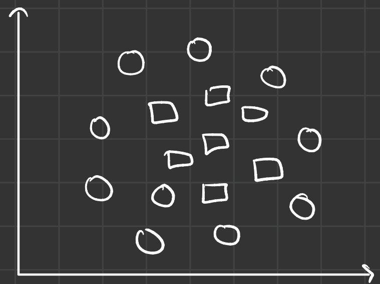
Margin should be maximum to increase accuracy & avoid misclassification

Hyperplane → Linear → LSVM

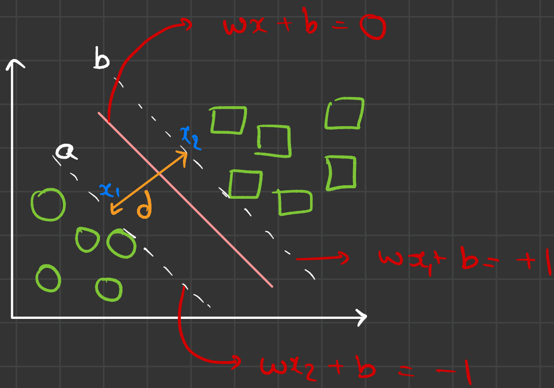
ex :-



Hyperplane → non - linear → NSVM



Ideal :-



d = margin

$$
\begin{array}{rl}
 & wx_1 + b = 1 \\
(-) & \underline{wx_2 + b = -1}
\end{array}
$$

$$\underbrace{w\,(x_1 - x_2)}_{\text{margin}} = 2$$

margin :- $(x_1 - x_2) = \dfrac{2}{|w|}$

From Support vectors,

$w_0 x_i + b_0 \geq 1$ for $d_i = 1$

$w_0 x_i + b_0 \leq -1$ for $d_i = -1$

Convert into single line by multiplying d_i,

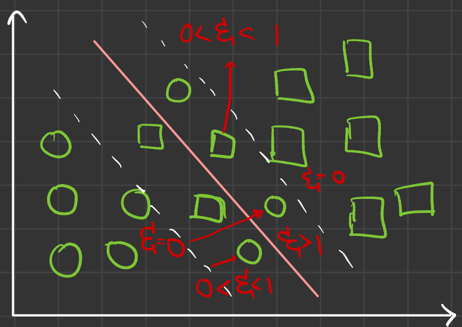
$d_i (w_0 x_i + b_0) \geq 1$ for $i = 1 \ldots\ldots N$

↳ If -ve :- missclassification

Hard margin :- $d = \frac{2}{||w||}$

Practical :-

(If we have outliers)



optimal hyperplane for non-separable data

$d_i (w x_i + b) \geq 1 - \xi_i \quad (i = 1 \ldots N)$

↳ slack variable

Soft margin

$$\phi(\xi) = C \sum_{i=1}^{N} \xi_i$$

Combine hard + soft margin

$$\phi(w, \xi_i) = \frac{\|w\|}{2} + C \sum_{i=1}^{N} \xi_i$$

low

high width
misclassification

high

less width
no-misclassification

Naive Bayes

Bayes Theorem :- $P(A/B) = \dfrac{P(B/A) \times P(A)}{P(B)}$

Naive Bayes

Q. Fruit = { Yellow , Sweet , Long }

Fruit	Yellow	Sweet	Long	Total
Orange	350	450	0	650
Banana	400	300	35	400
Others	50	100	50	150
Total	800	850	400	1200

A. $$P(\text{Yellow} / \text{Orange}) = \frac{P(\text{Orange} / \text{Yellow}) \times P(\text{Yellow})}{P(\text{Orange})}$$

$$= \frac{\frac{350}{800} \times \frac{800}{1200}}{650/1200}$$

$$P(\text{Sweet} / \text{Orange}) = \frac{P(\text{Orange} / \text{Sweet}) \times P(S)}{P(\text{Orange})}$$

$$= \frac{\frac{450}{850} \times \frac{850}{1200}}{\frac{650}{1200}} = 0.69$$

$$P(\text{Long} / \text{Orange}) = 0$$

$$\text{Fruit} = \{\text{Yellow}, \text{Sweet}, \text{Long}\}$$

$$P(\text{Fruit} / \text{Orange}) = P(\text{Yellow} / \text{Orange}) \times P(\text{Sweet} / \text{Orange}) \times P(L/O)$$

$$= \underline{\underline{0}}$$

Then, $P(\text{Fruit} / \text{Banana}) \ldots$

$$= P(Y/B) \times P(S/B) \times P(L/B)$$

$$= 1 \times 0.75 \times 0.89 = 0.65$$

$$P(\text{Fruit} / \text{Others}) = P(Y/\text{Others}) \times P(S/\text{others}) \times P(L/\text{others})$$

$$= 0.33 \times 0.66 \times 0.33$$

$$= 0.072$$

Since $P(F/B)$ is highest

$\therefore$ Ans $\Rightarrow$ Fruit being Banana

i.e., Banana would be sweetest, yellowest and longest.

Steps :-

1. Calculate prior probability

$P(\text{play tennis} = Y) = 9/14 = 0.64$

$P(\text{play tennis} = N) = 5/14 = 0.36$

2. Calculate conditional probabilities of individual attributes

Outlook	Y	N
Sunny	2/9	3/5
Overcast	4/9	0
Rain	3/9	2/5

Temp	Y	N
Hot	2/9	2/5
Mild	4/9	2/5
Cold	3/9	1/5

Humidity	Y	N
High	3/9	4/5
Normal	6/9	1/5

Wind	Y	N
Strong	3/9	3/5
Weak	6/9	2/5

3)

$$V_{NB} = \underset{v_j \in \{yes, no\}}{\arg\max} P(v_j) \prod_i \left(P(a_i / v_j)\right)$$

$$= \underset{v_j \in \{yes, no\}}{\arg\max} P(v_j) \, P(outlook = sunny / v_j) \times P(cool / v_j) \times P(high / v_j) \times P(strong / v_j)$$

$$V_{NB}(yes) = P(yes) \times P(sunny / yes) \times P(cool / yes) \times P(high / yes) \times P(strong / yes)$$

$$= 0.64 \times \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} = 0.00527$$

$V_{NB}(no) = P(N) \times P(S/N) \times P(C/N) \times$

$P(H/N) \times P(S/N)$

$= 0.36 \times \frac{3}{5} \times \frac{1}{5} \times \frac{4}{5} \times \frac{3}{5}$

$= 0.0207$

As $V_{NB}(No)$ is higher.

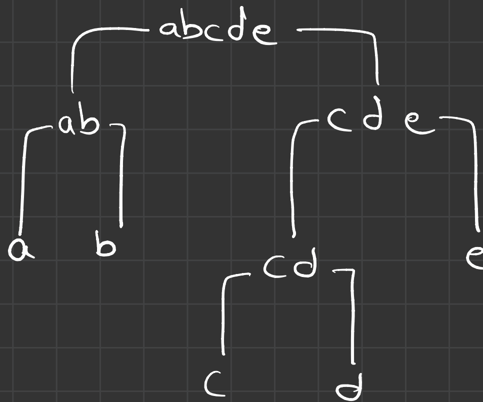
∴ We can't play tennis on that day.

To check Answer,

$$V_{NB}(yes) = \frac{V_{NB}\ yes}{V_{NB}\ yes + V_{NB}\ no}$$

$$V_{NB}(No) = \frac{V_{NB}\ no}{V_{NB}\ yes + V_{NB}\ No}$$

Hierarchial Clustering

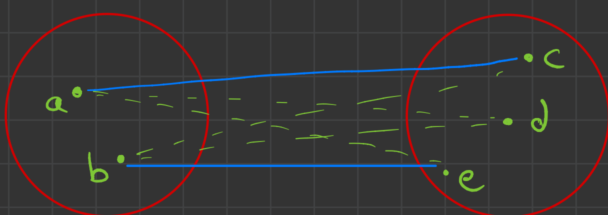


(agglomerative clustering)

bottom → up merge leaf nodes

$N =$ no. of points

no. of levels $= N - 1$



Complete :- $\max \{ d(x, y),\ x \in A,\ y \in B \}$

Single :- $\min \{ d(x, y),\ x \in A,\ y \in B \}$

Average :- $\frac{1}{|A||B|} \sum_{\substack{x \in A \\ y \in B}} d(x, y)$