Single Layer Perceptron: $w_1x_1 + w_2x_2 + \dots w_nx_n = z$   $z = \begin{cases} 1 & \text{if } z \geq \theta \\ 0 & \text{if } z < \theta \end{cases}$

Perceptron rule for XOR gate : $y = \underline{x_1\bar{x_2}} + \underline{\bar{x_1}x_2} = z_1 + z_2$

$C = T - O$

$\boxed{z_1 = x_1\bar{x_2}}$. Assume, $w_1 = 1, w_2 = 1, b = 0, \alpha = 1$

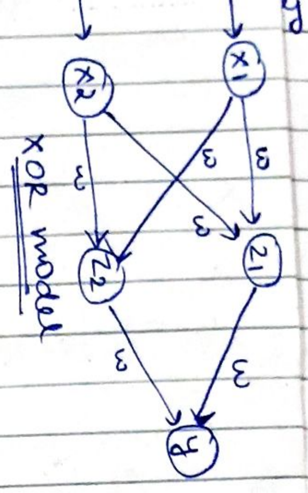
$O = b + x_1w_1 + x_2w_2 \dots$
$w_1' = w_1 + \alpha \cdot E \cdot x_1$
$w_2' = w_2 + \alpha \cdot E \cdot x_2$

7.25
75

TRUTH TABLE

| $x_1$ | $x_2$ | $t_{z_1}$ | $w_1$ | $w_2$ | O | E | $w_1'$ | $w_2'$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | -1 | 1 | -0.5 |
| 1 | 0 | 1 | 1 | -0.5 | 1 | 0 | 1 | -0.5 |
| 1 | 1 | 0 | 1 | -0.5 | 0 | 0 | 1 | -0.5 |

y.




$\boxed{z_2 = \bar{x_1}x_2}$ Assume $w_1 = 1, w_2 = 1, b = 0, \alpha = 1$

TRUTH TABLE

| $x_1$ | $x_2$ | $t_{z_2}$ | $w_1$ | $w_2$ | O | E | $w_1'$ | $w_2'$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | -1 | -0.5 | 1 |
| 1 | 1 | 0 | -0.5 | 1 | 0 | 0 | -0.5 | 1 |

$\boxed{y = z_1 + z_2} = z_1 \cdot w_i$ OR $z_2 w_j$

| $x_1$ | $x_2$ | $t_{z_1}$ | $t_{z_2}$ | y | $w_1$ | $w_2$ | O | E | $w_1'$ | $w_2'$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |

Multilayer Perceptron: Sigmoid function

$h = f(w \cdot x + b)$.
$O = f(w[h_1, h_2] + b)$.

Eg:- input   hash layer   output

$$x_1 \xrightarrow{w} h_1 \xrightarrow{w} O, \quad x_1 \xrightarrow{w} h_2, \quad x_2 \xrightarrow{w} h_1, \quad x_2 \xrightarrow{w} h_2 \xrightarrow{w} O$$

What happens if we pass i/p $x = [2, 3]$ where $w = [0, 1]$ and bias $b = 0$. using sigmoid activation function

$h_1 = h_2 = f(w \cdot x + b) = f((0 * 2) + (1 * 3)) + 0 = f(3) = 0.9526$.

$O = f(w \cdot [h_1, h_2] + b) = f((0 * h_1) + (1 * h_2)) + 0 = f(0.9526) = 0.7216$

$\boxed{\text{Perceptron Learning}}$

$O = b + x_1w_1 + x_2w_2 + \dots$
$\Delta w_1 = \alpha \cdot E \cdot x_1$, $\Delta w_2 = \alpha \cdot E \cdot x_2$, $\Delta w_3 = \alpha \cdot E \cdot x_3$

sample.

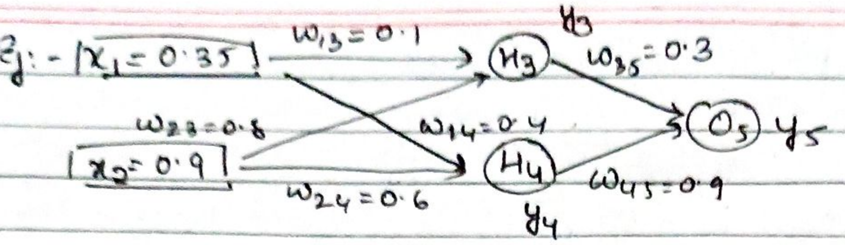
| $x_0$ | $x_1$ | $x_2$ | t | $w_0$ | $w_1$ | $w_2$ | O | E | $\Delta w_0$ | $\Delta w_1$ | $\Delta w_2$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| end of epoch 1. | | | | | | | | | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | -1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |

end of epoch

e has all os

Noise = outliers.

The previous perceptron learning function methodology works only for single layer perceptron → OR, AND, XOR.
For multilayer perceptron, we use sigmoid function with forward & backward propagation → (in neural networks)




Assume that actual o/p $y = 0.5$ & $d = 1$.

$$\boxed{y_i = f(a_i) = \frac{1}{1+e^{-a_i}} \qquad f'(a_i) = y_i(1-y_i)}$$

$a_3 = w_{13} \cdot x_1 + w_{23} \cdot x_2 = (0.1 * 0.35) + (0.8 * 0.9) = 0.755$

$y_3 = f(a_3) = \frac{1}{1+e^{-0.755}} = 0.68$

$a_4 = w_{14} * x_1 + w_{24} * x_2 = (0.4 * 0.35) + (0.6 * 0.9) = 0.68$

$y_4 = f(a_4) = \frac{1}{1+e^{-0.68}} = 0.6637$

$a_5 = w_{35} * y_3 + w_{45} * y_4 = (0.3 * 0.68) + (0.9 * 0.6637) = 0.801$

$y_5 = f(a_5) = \frac{1}{1+e^{-0.801}} = 0.69$ $\quad \therefore$ Error $= y_{\text{target}} - y_5 = ~~0.5-0.69~~ = 0.5 - 0.69 = -0.19$

For o/p unit:

$\delta_5 = y_5(1-y_5)(y_{\text{target}} - y_5)$

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

$$\boxed{\begin{array}{l} \Delta w_{ji} = \alpha\, \delta_j\, O_i \\ \text{o/p unit } \delta_j = O_j(1-O_j)(t_j - O_j) \\ \text{hidden unit } \delta_j = O_j(1-O_j) \sum_k \delta_k w_{kj} \end{array}}$$

For hidden units:

$\delta_3 = y_3(1-y_3)(0.3 * -0.0406) = -0.00265$

$\delta_4 = y_4(1-y_4)(0.9 * -0.0406) = -0.0082$

$\therefore$ ~~$\Delta w_{45} = \alpha\, \delta_4 O_3 = \alpha\, \delta_4$~~ ~~Similarly~~ Update all weights.

| i | j | $w_{ij}$ | $\delta_j$ | $x_i$ | $\alpha$ | $w_{ij}'$ |
|---|---|---|---|---|---|---|
| 1 | 3 | 0.1 | -0.00265 | 0.35 | 1 | 0.0991 |
| 2 | 3 | 0.8 | -0.00265 | 0.9 | 1 | 0.7976 |
| 1 | 4 | 0.4 | -0.0082 | 0.35 | 1 | 0.3971 |
| 2 | 4 | 0.6 | -0.0082 | 0.9 | 1 | 0.5926 |
| 3 | 5 | 0.3 | -0.0406 | 0.68 | 1 | 0.2724 |
| 4 | 5 | 0.9 | -0.0406 | 0.6637 | 1 | 0.8731 |

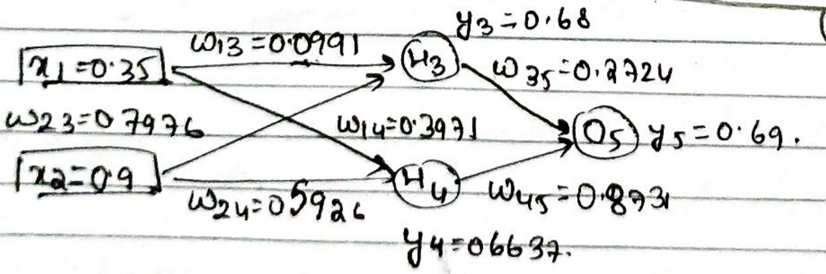
$\Delta w_{45} = \alpha\, \delta_5\, y_4$

$= 1 * -0.0406 * 0.6637$

$= -0.0269$

$w_{45}' = \Delta w_{45} + w_{45}$

$= -0.0269 + 0.9 = 0.8731$




$a_3 = (w_{13} * x_1) + (w_{23} * x_2) = (0.0991 * 0.35) + (0.7976 * 0.9) = 0.7525$

$y_3 = f(a_3) = 1/1+e^{-0.7525} = 0.6797$

$a_4 = (w_{14} * x_1) + (w_{24} * x_2) = (0.3971 * 0.35) + (0.5926 * 0.9) = 0.6723$

$y_4 = f(a_4) = 1/1+e^{-0.6723} = 0.6620$

$a_5 = (w_{35} * y_3) + (w_{45} * y_4) = (0.2724 * 0.6797) + (0.8731 * 0.6620) = 0.7631$

$y_5 = f(a_5) = 1/1+e^{-0.7631} = 0.6820$ $\quad \therefore \varepsilon = y_t - y_5 = 0.5 - 0.6820 = -0.182$

(Keep repeating until $\varepsilon = 0$)

## SVM

How to select best hyperplane? give 1: $\textcircled{5} + 2x_1 + 5x_2$ & 2: $\textcircled{5} + 20x_1 + 50x_2$ — bias: ignore

$D1 = \sqrt{2^2+5^2} = 5.39 \therefore \frac{2}{\|w\|} = \frac{2}{5.39} = 0.37$ ← max, so best hyperplane.

$D2 = \sqrt{20^2+50^2} = 53.85 \therefore \frac{2}{\|w\|} = \frac{2}{53.85} = 0.037$

## Linear SVM

given +ve points: $(4,1), (4,-1), (6,0)$ & -ve points: $(1,0), (0,1), (0,-$

Augmented vector = $s_1 = \begin{pmatrix}1\\0\\1\end{pmatrix}, s_2 = \begin{pmatrix}4\\1\\1\end{pmatrix}, s_3 = \begin{pmatrix}4\\-1\\1\end{pmatrix}$ } only use support vectors!!

add bias.

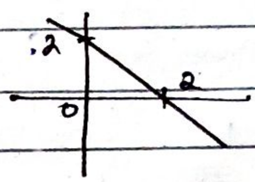
$\alpha_1 s_1 s_1 + \alpha_2 s_2 s_1 + \alpha_3 s_3 s_1 = -1 \Rightarrow \alpha_1 \begin{bmatrix}1\\0\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\end{bmatrix} + \alpha_2 \begin{bmatrix}4\\1\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\end{bmatrix} + \alpha_3 \begin{bmatrix}4\\-1\\1\end{bmatrix}\begin{bmatrix}1\\0\\1\end{bmatrix} = 2\alpha_1 + 5\alpha_2 + 5\alpha_3 = -$

$\alpha_1 s_1 s_2 + \alpha_2 s_2 s_2 + \alpha_3 s_3 s_2 = +1 \Rightarrow \alpha_1 \begin{bmatrix}1\\0\\1\end{bmatrix}\begin{bmatrix}4\\1\\1\end{bmatrix} + \alpha_2 \begin{bmatrix}4\\1\\1\end{bmatrix}\begin{bmatrix}4\\1\\1\end{bmatrix} + \alpha_3 \begin{bmatrix}4\\-1\\1\end{bmatrix}\begin{bmatrix}4\\1\\1\end{bmatrix} = 5\alpha_1 + 18\alpha_2 + 16\alpha_3 = +$

$\alpha_1 s_1 s_3 + \alpha_2 s_2 s_3 + \alpha_3 s_3 s_3 = +1 \Rightarrow \alpha_1 \begin{bmatrix}1\\0\\1\end{bmatrix}\begin{bmatrix}4\\-1\\1\end{bmatrix} + \alpha_2 \begin{bmatrix}1\\0\\1\end{bmatrix}\begin{bmatrix}4\\-1\\1\end{bmatrix} + \alpha_3 \begin{bmatrix}4\\-1\\1\end{bmatrix}\begin{bmatrix}4\\-1\\1\end{bmatrix} = 5\alpha_1 + 16\alpha_2 + 18\alpha_3 = +$

from above eq: we get $\alpha_1 = -3, \alpha_2 = +1, \alpha_3 = 0$

$\therefore w = \sum_{1}^{3} \alpha_i * s_i = -3\begin{bmatrix}1\\0\\1\end{bmatrix} + 1\begin{bmatrix}4\\1\\1\end{bmatrix} + 0\begin{bmatrix}4\\-1\\1\end{bmatrix} = \begin{bmatrix}1\\1\\-2\end{bmatrix}$



$\therefore$ hyperplane has offset $-2$ and is $(1,1)$.

## Non-Linear SVM

given +ve: $\left(\begin{bmatrix}2\\2\end{bmatrix}\begin{bmatrix}2\\-2\end{bmatrix}\begin{bmatrix}-2\\-2\end{bmatrix}\begin{bmatrix}-2\\2\end{bmatrix}\right)$ & -ve: $\left(\begin{bmatrix}1\\1\end{bmatrix}\begin{bmatrix}1\\-1\end{bmatrix}\begin{bmatrix}-1\\-1\end{bmatrix}\begin{bmatrix}-1\\1\end{bmatrix}\right)$

$$\phi\begin{pmatrix}x_1\\x_2\end{pmatrix} = \begin{cases}\begin{pmatrix}4 - x_2 + |x_1 - x_2|\\4 - x_1 + |x_1 - x_2|\end{pmatrix} & \text{if } \sqrt{x_1^2 + x_2^2} > 2\\ \begin{pmatrix}x_1\\x_2\end{pmatrix} & \text{otherwise}\end{cases}$$

new $\begin{pmatrix}x_1\\x_2\end{pmatrix}$ values: +ve: $\left(\begin{bmatrix}2\\2\end{bmatrix}\begin{bmatrix}10\\6\end{bmatrix}\begin{bmatrix}6\\6\end{bmatrix}\begin{bmatrix}6\\10\end{bmatrix}\right)$, -ve: $\left(\begin{bmatrix}1\\1\end{bmatrix}\begin{bmatrix}1\\-1\end{bmatrix}\begin{bmatrix}-1\\-1\end{bmatrix}\begin{bmatrix}-1\\1\end{bmatrix}\right)$.

Now they plot a linear SVM; proceed with linear SVM steps.

## TYPES OF KERNELS

- linear: $k(x,y) = x^T y$
- polynomial: $k(x,y) = (x^T y)^q$ ← degree, order.
- inhomogeneous polynomial: $k(x,y) = (c + x^T y)^q$, constant.
- gaussian kernel: $k(x,y) = e^{-\frac{(x-y)^2}{2\sigma^2}}$
- sigmoid kernel: $\tanh(kx \cdot y - \sigma)$

MOD 4

$\|C_1\| = \sqrt{a^2+b^2+c^2+\dots}$

Jaccard Similarity of $(C_1, C_2) = (C_1 \cap C_2) / (C_1 \cup C_2)$

Cosine Similarity of $(C_1, C_2) = \cos(C_1, C_2) = (C_1 \cdot C_2) / \|C_1\|\|C_2\|$

if vectors are correlated, angle is 0° and $\cos(x,y) = 1$

if vectors are orthogonal, angle is 90° and $\cos(x,y) = 0$.

Inter Cluster Distance:

- Single Linkage - distance of closest pair of data objects
- Complete Linkage - distance of farthest pair of data objects
- Average Linkage - avg distance of pairs of data objects.
- Centroid method - mean of 2 clusters.

Agglomerate Clustering using Single Linkage

| | P1 | P2 | P3 | P4 | P5 | P6 |
|---|---|---|---|---|---|---|
| P1 | 0 | 0.23 | 0.22 | 0.37 | 0.34 | 0.24 |
| P2 | 0.23 | 0 | 0.14 | 0.19 | 0.14 | 0.24 |
| P3 | 0.22 | 0.14 | 0 | 0.13 | 0.28 | 0.10 |
| P4 | 0.37 | 0.19 | 0.13 | 0 | 0.23 | 0.22 |
| P5 | 0.34 | 0.14 | 0.28 | 0.23 | 0 | 0.39 |
| P6 | 0.24 | 0.24 | 0.10 | 0.22 | 0.39 | 0 |

min

→

| | P1 | P2 | P3,P6 | P4 | P5 |
|---|---|---|---|---|---|
| P1 | 0 | 0.23 | 0.22 | 0.37 | 0.34 |
| P2 | 0.23 | 0 | 0.14 | 0.19 | 0.14 |
| P3,P6 | 0.22 | 0.14 | 0 | 0.13 | 0.28 |
| ~~P4~~ | ~~0.37~~ | ~~0.19~~ | ~~0.13~~ | ~~0~~ | ~~0.23~~ |
| P5 | 0.34 | 0.14 | 0.28 | 0.23 | 0 |

min

just delete needed row & column, no merging

| | P1 | P2 | P3,P6,P4 | P5 |
|---|---|---|---|---|
| P1 | 0 | 0.23 | 0.22 | 0.34 |
| P2 | 0.23 | 0 | 0.14 | 0.14 |
| P3,P6,P4 | 0.22 | 0.14 | 0 | 0.28 |
| ~~P5~~ | ~~0.34~~ | ~~0.14~~ | ~~0.28~~ | ~~0~~ |

min

→

| | P1 | P2,P5 | P3,P6,P4 | ~~P5~~ |
|---|---|---|---|---|
| P1 | 0 | 0.23 | 0.22 | |
| P2,P5 | 0.23 | 0 | 0.14 | |
| ~~P3,P6,P4~~ | ~~0.22~~ | ~~0.14~~ | ~~0~~ | |
| ~~P5~~ | | | | |

min

→

| | P1 | P2,P5,P3,P6,P4 |
|---|---|---|
| P1 | 0 | 0.23 |
| P2,P5,P3,P6,P4 | 0.23 | 0 |

[{(P3,P6),P4},(P2,P5)], P1

Agglomerate clustering using complete Linkage

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 4 | 7 | 9 | 1 |
| 2 | 4 | 0 | 3 | 5 | 3 |
| 3 | 7 | 3 | 0 | 2 | 6 |
| 4 | 9 | 5 | 2 | 0 | 8 |
| ~~5~~ | ~~1~~ | ~~3~~ | ~~6~~ | ~~8~~ | ~~0~~ |

min

→

| | 1,5 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1,5 | 0 | 4 | 7 | 9 |
| 2 | 4 | 0 | 3 | 5 |
| 3 | 7 | 3 | 0 | 2 |
| 4 | 9 | 5 | 2 | 0 |

min

What did we do?

Leave (1,1) because = 0.

Now while merging, chose max element from each column. to form new row(1,5)

4>3, 7>6, 9>8 so 1,5 = 0 4 7 9.

write as it is.

| | 1,5 | 2 | 3,4 |
|---|---|---|---|
| 1,5 | 0 | 4 | 9 |
| 2 | 4 | 0 | 5 |
| 3,4 | 9 | 5 | 0 |

7<9, 3<5, ignore 0

so new(3,4) = 9 5 0

update both row & column.

→

| | 1,5 | 2 | 3,4 |
|---|---|---|---|
| 1,5 | 0 | 4 | 9 |
| ~~2~~ | ~~4~~ | ~~0~~ | ~~5~~ |
| 3,4 | 9 | 5 | 0 |

min

→

| | (1,5)2 | (3,4) |
|---|---|---|
| (1,5)2 | 0 | 9 |
| (3,4) | 9 | 0 |

DIANA

| | a | b | c | d | e |
|---|---|---|---|---|---|
| a | 0 | 9 | 3 | 6 | 11 |
| b | 9 | 0 | 7 | 5 | 10 |
| c | 3 | 7 | 0 | 9 | 2 |
| d | 6 | 5 | 9 | 0 | 8 |
| e | 11 | 10 | 2 | 8 | 0 |

① $C_i = \{a, b, c, d, e\}$, $C_j = \{\phi\}$.

② initial iteration:

$D_a = \frac{1}{4}(a,b + a,c + a,d + a,e) = \frac{1}{4}(9+3+6+11) = 7.25$

$D_b = \frac{1}{4}(b,a + b,c + b,d + b,e) = \frac{1}{4}(9+7+5+10) = 7.75$.

$D_c = 5.25$, $D_d = 7.00$, $D_e = 7.75$ — MAX

Since $D_b, D_e$ are max, chose $D_b$ randomly.

③ $C_i = \{a, c, d, e\}$, $C_j = \{b\}$.

④ 2nd iteration:

$D_a = \frac{1}{3}(a,c + a,d + a,e) - \frac{1}{1}(a,b) = \frac{20}{3} - 9 = -2.33$.

$D_c = \frac{1}{3}(c,a + c,d + c,e) - \frac{1}{1}(c,b) = \frac{14}{3} - 7 = -2.33$.

$D_d = \frac{1}{3}(d,a + d,c + d,e) - \frac{1}{1}(d,b) = \frac{23}{3} - 7 = 0.67$.

$D_e = \frac{1}{3}(e,a + e,c + e,d) - \frac{1}{1}(e,b) = \frac{21}{3} - 7 = 0$.

⑤ $D_d$ is max & > 0 so

$C_i = \{a, c, e\}$

$C_j = \{b, d\}$.

⑥ 3rd iteration:

$D_a = \frac{1}{2}(a,c + a,e) - \frac{1}{2}(a,b + a,d) = \frac{14}{2} - \frac{15}{2} = -0.5$

$D_c = \frac{1}{2}(c,a + c,e) - \frac{1}{2}(c,b + c,d) = \frac{5}{2} - \frac{16}{2} = -13.5$

$D_e = \frac{1}{2}(e,a + e,c) - \frac{1}{2}(e,b + e,d) = \frac{13}{2} - \frac{18}{2} = -2.5$.

All are < 0 so stop.
No change in
$C_i = \{a, c, e\}$
$C_j = \{b, d\}$.

⑦ we divide $C_i$ & $C_j$.

$C_i$: max {(a,c), (a,e); (c,e)} = max (3, 11, 2) = 11.

$C_j$: max (b,d) = max (5) = 5

As 11 > 5 we split $C_i$.

| | a | c | e |
|---|---|---|---|
| a | 0 | 3 | 11 |
| c | 3 | 0 | 2 |
| e | 11 | 2 | 0 |

⑧ Now repeat the process using only $C_i = \{a, c, e\}$ & $C_j = \{\phi\}$

DBSCAN

| | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 |
|---|---|---|---|---|---|---|---|---|
| S1 | 0 | | | | | | | |
| S2 | 4.24 | 0 | | | | | | |
| S3 | 4.47 | 5.1 | 0 | | | | | |
| S4 | 3.16 | 4 | 1.41 | 0 | | | | |
| S5 | 2 | 5.83 | 4 | 3.16 | 0 | | | |
| S6 | 1 | 3.61 | 5 | 3.61 | 3 | 0 | | |
| S7 | 6.08 | 3.61 | 3.61 | 3.61 | 6.71 | 6 | 0 | |
| S8 | 2 | 3.16 | 2.83 | 1.41 | 2.83 | 2.24 | 4.12 | 0 |

Given $\varepsilon = 3.5$ & minPts = 3.

For each column find values < 3.5

S1: S4, S5, S6, S8 ; S2: S8 ; S3: S4, S8 ;
S4: S1, S5, S3, S8 ; S5: S1, S4, S6, S8 ;
S6: S1, S5, S8 ; S7: $\phi$ , S8: S1, S2, S3, S4, S5, S6

if count of each grp < 3 ⇒ Noise.

| | | |
|---|---|---|
| S1 | Core | Boundary |
| S2 | Noise | Boundary |
| S3 | Core | Boundary |
| S4 | Core | Boundary |
| S5 | Core | Boundary |
| S6 | Core | Boundary |
| S7 | Noise | Noise |
| S8 | Core | Boundary |

* if S1 is found in other grps → boundary.
* S7 is not found in any other grp → noise

Noise = outliers.

K means| Q. given 8 clusters A1, A2, A3, A4, A5, A6, A7, A8. Assume A1, A4, A7 as the initial seeds (centres) of the cluster.

| Data point x | y | Distance to C1 x 2, y 10 | C2 x 5, y 8 | C3 x 1, y 2 | New cluster no. |
|---|---|---|---|---|---|
| 2 | 10 | 0.00 | 3.61 | 8.06 | 1 |
| 2 | 5 | 5.00 | 4.24 | 3.16 | 3 |
| 8 | 4 | 8.49 | 5.00 | 7.28 | 2 |
| 5 | 8 | 3.61 | 0.00 | 7.21 | 2 |
| 7 | 5 | 7.07 | 3.61 | 6.71 | 2 |
| 6 | 4 | 7.21 | 4.12 | 5.39 | 2 |
| 1 | 2 | 8.06 | 7.21 | 0.00 | 3 |
| 4 | 9 | 2.24 | 1.41 | 7.62 | 2 |

← Assign new cluster as cluster with MIN value distance.

- new value for C1 = (2,10) since there is only value for C1
- new value for C2 = (mean of x, mean of y) $= \left(\frac{8+5+7+6+4}{5}, \frac{4+8+5+4+9}{5}\right)$ = (6,6).
- new value for C3 = (1.5, 3.5) similarly.

2nd iteration

| Data Points x | y | Distance to C1' x y | C2' x y | C3' x y | Prev Cluster | New Clusters |
|---|---|---|---|---|---|---|
| 2 | 10 | 0.00 | 5.66 | 6.52 | 1 | 1 |
| 2 | 5 | 5.00 | 4.12 | 1.58 | 3 | 3 |
| 8 | 4 | 8.49 | 2.83 | 6.52 | 2 | 2 |
| 5 | 8 | 3.61 | 2.24 | 5.70 | 2 | 2 |
| 7 | 5 | 7.07 | 1.41 | 5.70 | 2 | 2 |
| 6 | 4 | 7.21 | 2.00 | 4.53 | 2 | 2 |
| 1 | 2 | 8.06 | 6.40 | 1.58 | 3 | 3 |
| 4 | 9 | 2.24 | 3.61 | 6.04 | 2 | ~~2~~ 1 |

value mismatch.

- new value for C1 $= \left(\frac{2+4}{2}, \frac{10+9}{2}\right) = (3, 9.5)$
- new value for C2 $= \left(\frac{8+5+7+6}{4}, \frac{4+8+5+4}{4}\right)$ = (6.5, 5.25)
- new value for C3 $= \left(\frac{2+1}{2}, \frac{5+2}{2}\right) = (1.5, 3.5)$

↙ keep repeating until prev cluster = new cluster

K modes cluster for categorical data | Chose P1, P7, P8 as leaders.

| person | hair | eye | skin | no. of Dissimilarities P1 | P7 | P8 | Clusters |
|---|---|---|---|---|---|---|---|
| P1 | blonde | amber | fair | 0 | 2 | 2 | 1 |
| P2 | brunette | gray | brown | 3 | 3 | 3 | 1 |
| P3 | red | hazelgreen | brown | 3 | 1 | 3 | 2 |
| P4 | black | amberhazel | brown | 3 | 3 | 1 | 3 |
| P5 | brunette | grayamber | fair | 1 | 2 | 2 | 1 |
| P6 | black | greengray | brown | 3 | 3 | 2 | 3 |
| P7 | red | green | fair | 2 | 0 | 2 | 2 |
| P8 | black | hazel | fair | 2 | 2 | 0 | 3 |

| | hair | eye | skin |
|---|---|---|---|
| For cluster 1, most observed properties are | brunette | amber, | fair. |
| For cluster 2, most observed properties are | red, | green, | brown/fair. |
| For cluster 3, most observed properties are | black, | hazel | brown |

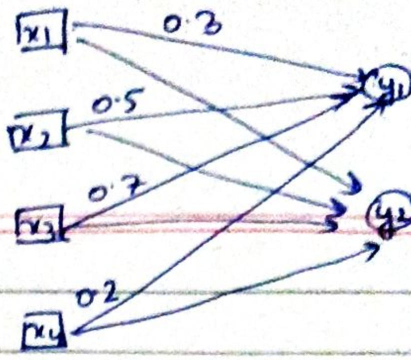
∴ new leaders are

| | hair | eye | skin |
|---|---|---|---|
| C1 | brunette | amber | fair |
| C2 | red | green | fair |
| C3 | black | hazel | brown |

SOM: given training samples $x_1$ (1010), $x_2$ (1000), $x_3$ (1111), $x_4$ (0110)

and o/p units $\begin{bmatrix} \text{unit 1} \\ \text{unit 2} \end{bmatrix} = \begin{bmatrix} 0.3 & 0.5 & 0.7 & 0.2 \\ 0.6 & 0.5 & 0.4 & 0.2 \end{bmatrix}$

$\alpha = 0.6$




Distance b/w $x_1$ & unit 1

$d^2 = (1-0.3)^2 + (0-0.5)^2 + (1-0.7)^2 + (0-0.2)^2 = 0.87$ MIN

Distance b/w $x_1$ & unit 2

$d^2 = (1-0.6)^2 + (0-0.5)^2 + (1-0.4)^2 + (0-0.2)^2 = 1.1$

update wts of winning UNIT 1 using $w' = w + \alpha(x - w)$

new UNIT 1 wt $= [0.3\ 0.5\ 0.7\ 0.2] + 0.6([1\,0\,1\,0] - [0.3\ 0.5\ 0.7\ 0.2])$

2nd iteration $= [0.72\ \ 0.2\ \ 0.88\ \ 0.08]$ (unit 2 remains same)

Distance b/w $x_2$ & unit 1

$d^2 = (1-0.72)^2 + (0-0.2)^2 + (0-0.88)^2 + (0-0.08)^2 = 0.74$ MIN

Distance b/w $x_2$ & unit 2

$d^2 = (1-0.6)^2 + (0-0.5)^2 + (0-0.4)^2 + (0-0.2)^2 = 0.4$.

New unit 1 wts $= [0.72\ 0.2\ 0.88\ 0.08] + 0.6([1000] - [0.72\ 0.2\ 0.88\ 0.08])$

$= [0.89\ \ 0.08\ \ 0.35\ \ 0.083]$ (unit 2 remains same)

continue the same with $x_3$ & $x_4$.

$x_1$: (1010) → unit 1, $x_2$ (1000) → unit 1, $x_3$ (1111) → unit 2, $x_4$ (0110) → unit 2

## Expectation Maximization

given samples

1. H T T T H H T H T H
2. H H H H T H H H H H
3. H T H H H H H T H H
4. H T H T T T H H T T
5. T H H H T H H H T H

Assume initial $\theta_A = 0.6$, $\theta_B = 0.5$.

$P\left(\frac{E}{Z_A}\right) = \binom{n}{x} \theta_A^{x} (1-\theta_A)^{n} = \binom{9}{1} 0.6^9\, 0\ldots$

$= 0.0\ldots$

$P\left(\frac{E}{Z_B}\right) = \binom{n}{x} \theta_B^{x} (1-\theta_B)^{n} = \binom{9}{1} 0.5^9\, 0.5^1$

$= 0.009$.

$P\left(\frac{Z_A}{E}\right) = \frac{0.036}{0.036+0.009} = 0.8$, $P\left(\frac{Z_B}{E}\right) = \frac{0.009}{0.036+0.009} = 0.20$

Coin A: $P\left(\frac{Z_A}{E}\right) \times n = 0.8 \times 9 = 2.2H$, $P\left(\frac{Z_A}{E}\right) \times x = 0.8 \times 1 = 0.8$

Coin B: $P\left(\frac{Z_B}{E}\right) \times n = 0.2 \times 9 = 2.8H$, $P\left(\frac{Z_B}{E}\right) \times x = 0.2 \times 1 = 0.2T$

similarly repeat for all

| Coin A | Coin B. |
|---|---|
| ① 2.2H, 2.2T | 2.8H, 2.8T |
| ② 7.2H, 0.8T | 1.8H, 0.2T |
| ③ 5.9H, 1.5T | 2.1H, 0.5T |
| ④ 1.4H, 2.1T | 2.6H, 2.9T |
| ⑤ 4.5H, 1.9T | 2.5H, 1.1T |
| $\Sigma = 21.3H, 8.6T$ | $\Sigma = 11.7H, 8.4T$ |

new $\theta_A = \frac{21.3}{21.3+8.6} = 0.71$ repeat.

new $\theta_B = \frac{11.7}{11.7+8.4} = 0.58$.

PCA

| F | S1 | S2 | S3 | S4 |
|---|---|---|---|---|
| $x_1$ | 4 | 8 | 13 | 7 |
| $x_2$ | 11 | 4 | 5 | 14 |

mean $\bar{x_1} = 8, \bar{x_2} = 8.5$

① covariance matrix $S = \begin{bmatrix} cov(x_1,x_1) & cov(x_1,x_2) \\ cov(x_2,x_1) & cov(x_2,x_2) \end{bmatrix}$

② $cov(x_1,x_1) = \frac{1}{N-1}\sum_{i=1}^{N}(x_{1i}-\bar{x_1})(x_{1i}-\bar{x_1})$

$= \frac{1}{3}\{(4-8)(4-8)+(8-8)(8-8)+(13-8)(13-8)+(7-8)(7-8)\}$

$= 14.$

③ $Cov(x_1,x_2) = \frac{1}{N-1}\sum_{i=1}^{N}(x_{1i}-\bar{x_1})(x_{2i}-\bar{x_2}) = \frac{1}{3}\{(4-8)(11-8.5)+(8-8)(4-8.5)+(13-8)(5-8.5)$

$+(7-8)(14-8.5)\} = -11$ ④ $cov(x_2,x_1) = cov(x_1,x_2) = -11$

⑤ $cov(x_2,x_2) = \frac{1}{N-1}\sum_{i=1}^{N}(x_{2i}-\bar{x_2})(x_{2i}-\bar{x_2}) = \frac{1}{3}\{(11-8.5)^2+(4-8.5)^2+(5-8.5)^2+(14-8.5)^2\} = 23$

$\therefore S = \begin{bmatrix} 14 & -11 \\ -11 & 23 \end{bmatrix}$ ⑥ finding eigen values $det(S-\lambda I)=0 \Rightarrow det\begin{bmatrix} 14-\lambda & -11 \\ -11 & 23-\lambda \end{bmatrix} = 0.$

$\Rightarrow (14-\lambda)(23-\lambda)-(-11\times -11)=0 \Rightarrow \lambda^2-37\lambda+201=0$

$\therefore \lambda_1 = 30.3849, \lambda_2 = 6.6151.$

⑦ finding eigen vectors $(S-\lambda I)U = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 14-\lambda & -11 \\ -11 & 23-\lambda \end{bmatrix}\begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = 0 \Rightarrow \begin{bmatrix} (14-\lambda)U_1 - 11U_2 \\ -11U_1 + (23-\lambda)U_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$

from $(14-\lambda)U_1 - 11U_2 = 0 \Rightarrow \frac{U_1}{11} = \frac{U_2}{14-\lambda} = t$ $\therefore U_1 = 11t$ & $U_2 = (14-\lambda)t.$

~~from $(-11U_1+(23-\lambda)U_2 = 0 \Rightarrow U!$~~ $\therefore U = \begin{bmatrix} 11 \\ 14-\lambda \end{bmatrix}$

⑧ Length of $U_1 = \sqrt{11^2+(14-\lambda_1)^2} = \sqrt{11^2+(14-30.3849)^2} = 19.7348.$

$e_1 = \begin{bmatrix} \frac{11}{\|U_1\|} \\ \frac{14-\lambda_1}{\|U_1\|} \end{bmatrix} = \begin{bmatrix} \frac{11}{19.7348} \\ \frac{14-30.3849}{19.7348} \end{bmatrix} = \begin{bmatrix} 0.5574 \\ -0.8303 \end{bmatrix}$

Length of $U_2 = \sqrt{11^2+(14-\lambda_2)^2} = \sqrt{11^2+(14-6.6151)^2} = 13.249.$

$e_2 = \begin{bmatrix} \frac{11}{\|U_2\|} \\ \frac{14-\lambda_2}{\|U_2\|} \end{bmatrix} = \begin{bmatrix} \frac{11}{13.249} \\ \frac{14-6.6151}{13.249} \end{bmatrix} = \begin{bmatrix} 0.8303 \\ 0.5574 \end{bmatrix}$

⑨ Computation of first principal components.

$(e_1)^T \begin{bmatrix} x_{1i}-\bar{x_1} \\ x_{2i}-\bar{x_2} \end{bmatrix} = \begin{bmatrix} 0.5574 \\ -0.8303 \end{bmatrix}^T \begin{bmatrix} x_{11}-\bar{x_1} \\ x_{21}-\bar{x_2} \end{bmatrix} = \begin{bmatrix} 0.5574 & -0.8303 \end{bmatrix}\begin{bmatrix} 4-8 \\ 11-8.5 \end{bmatrix}$

$\Rightarrow = 0.5574(4-8) - 0.8303(11-8.5) = -4.30535.$

Similarly compute for all $\begin{bmatrix} x_{12}-\bar{x_1} \\ x_{22}-\bar{x_2} \end{bmatrix} = \begin{bmatrix} 8-8 \\ 4-11.5 \end{bmatrix}, \begin{bmatrix} x_{13}-\bar{x_1} \\ x_{23}-\bar{x_2} \end{bmatrix} = \begin{bmatrix} 13-8 \\ 5-8.5 \end{bmatrix},$ $\begin{bmatrix} x_{14}-\bar{x_1} \\ x_{24}-\bar{x_2} \end{bmatrix} = \begin{bmatrix} 7-8 \\ 14-8.5 \end{bmatrix}.$

we get

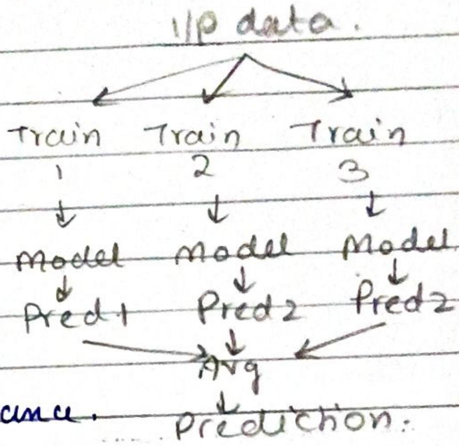
| feature | S1 | S2 | S3 | S4 |
|---|---|---|---|---|
| $x_1$ | 4 | 8 | 13 | 7 |
| $x_2$ | 11 | 4 | 5 | 14 |
| Principal Comp. | -4.3052 | 3.7361 | 5.6928 | -5.1238 |

MODS

Decision Trees

Pros: Trees can be displayed graphically - easily interpreted - human like decision making - can handle qualitative predictors without need to create dummy variables.

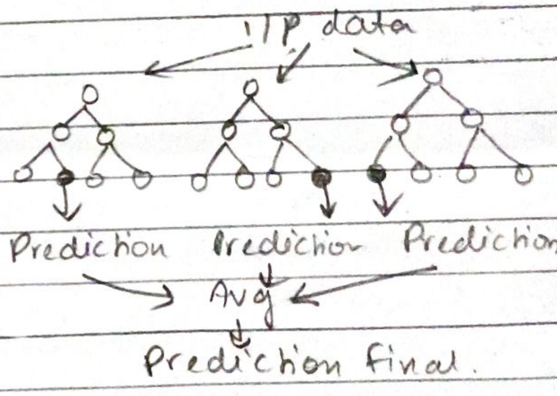
Cons: do not have good accuracy as regression / classification models - non-robust (a small change can cause a large change) - high variance ~~but by using~~




BAGGING

- reducing variance
- Regression: Construct B regression trees with B bootstrapped training sets & avg. the resulting prediction. Resulting trees grow deep & not pruned so high variance but low bias. Averaging trees → reduce variance.

RANDOM FOREST




- In bagging, no. of decision trees are b built on bootstrapped ~~no~~ trainings sets but in random forest, if the no. of examples in traing set is N, take a sample of n examples at random but with replacement

ADA BOOST

→ Assume if CGPA >= 9 → Y., else N.

Eg.-

| CGPA | Class | Pred Class. |
|---|---|---|
| >=9 | Y | Y |
| <9 | $\boxed{Y}$ | $\boxed{N}$ |
| >=9 | $\boxed{N}$ | $\boxed{Y}$ |
| <9 | N | N |
| >=9 | Y | Y |
| >=9 | Y | Y |

① Compute weighted error

$\epsilon = \Sigma\, H_i(d_j)\, wt(d_j)$

$H_i(d_j) = 0$ for correct prediction

$H_i(d_j) = 1$ for wrong pred.

Assuming all wts are 1/6 initially

$\epsilon = 2 * \frac{1}{6} = 0.333$ (bcuz 2 mismatch)

② Compute wt. for each weak classifier $\alpha_{CGPA} = \frac{1}{2} \log \frac{(1-\epsilon)}{\epsilon}$

$= \frac{1}{2} \log \frac{(1-0.333)}{0.333} = 0.347.$

③ Calculate normalizing factor. : $Z_{CGPA}$ = {wt (correctly classified) * (no of correctly classified) * $e^{-\alpha_{CGPA}}$} + {wt (wrongly classified) * (no. of wrongly classified) * $e^{+\alpha_{CGPA}}$} $= (\frac{1}{6} * 4 * e^{-0.347}) + (\frac{1}{6} * 2 * e^{+0.347})$

$= 0.9428.$

(4) update wts for all data instances

$$wt(dj)_{i+1} = \frac{wt(dj)_{CGPA} \text{ of correct instance} * e^{-\alpha CGPA}}{Z_{CGPA}}$$

$$= \frac{\frac{1}{6} * e^{-0.347}}{0.9428} = 0.1249.$$

$$wt(dj)_{i+1} = \frac{wt(dj)_{CGPA} \text{ of wrong instances} * e^{+\alpha CGPA}}{Z_{CGPA}}$$

$$= \frac{\frac{1}{6} * e^{0.347}}{0.942} = 0.2501$$

Original wts

| | correct | wrong | wrong | correct | cor | correct |
|---|---|---|---|---|---|---|
| ∴ Original wts ~~for correct~~ = | 1/6 | 1/6 | 1/6 | 1/6 | 1/6 | 1/6. |
| New wts = | 0.1249 | 0.2501 | 0.2501 | 0.1249 | 0.1249 | 0.1249. |

—x—

Extra notes:

Silhoutte Index: → goodness.

1 = clusters are well apart from each other.

0 = clusters are very close.

-1 = clusters are assigned the wrong way.

Silhoutte Score $= 1 - \frac{a}{b}$

where a = avg intra-cluster distance b/w each point in cluster

b = avg inter-cluster distance b/w all clusters.

Q. Dissimilarity Matrix:

| | P1 | P2 | P3 | P4 |
|---|---|---|---|---|
| P1 | 0 | 0.10 | 0.65 | 0.55 |
| P2 | 0.10 | 0 | 0.70 | 0.60 |
| P3 | 0.65 | 0.65 | 0 | 0.30 |
| P4 | 0.55 | 0.55 | 0.30 | 0 |

• Cluster label:

| Point | Cluster label |
|---|---|
| P1 | 1 |
| P2 | 1 |
| P3 | 2 |
| P4 | 2. |

P1) $a = \frac{0.1}{1} = 0.1$, $b = \frac{0.65+0.55}{2} = 0.6$. ∴ $SC = 1 - \frac{a}{b} = 1 - \frac{0.1}{0.6} = 0.8333$

P2) $SC = 1 - \frac{0.1}{(0.7+0.6)/2} = 0.846.$

P3) $SC = 1 - \frac{0.3}{(0.65+0.7)/2} = 0.556.$

P4) $SC = 1 - \frac{0.3}{(0.55+0.6)/2} = 0.478$

Cluster 1:
Avg SC $= \frac{0.833+0.846}{2}$
$= 0.84.$

Cluster 2:
Avg SC $= \frac{0.556+0.478}{2}$
$= 0.517.$

Overall:
Avg SC $= \frac{0.840+0.517}{2} = 0.68.$

## RAND INDEX → similarities.

0: indicates 2 clustering methods that don't agree on clustering

1: indicates 2 clustering methods perfectly agree on clusters.

$R = (a+b) / {}^nC_2$

Dataset = {A, B, C, D, E}.

m1: {1, 1, 1, 2, 2}.

m2: {1, 1, 2, 2, 3}.

a = ~~no.~~ unordered pairs belonging to same cluster across both clustering methods : {A, B}, a = 1

[∵ In ~~clust~~ C1: (A, B) = (1, 1), ~~(B)~~ C2: (A, B) = (1, 1)].

b = unordered pairs belonging to diff ~~clustr~~ clusters

{(A, D), (A, E), (B, D), (B, E), (C, E)}.

[∵ In C1 : (A, D) = (1, 2) and C2 : (A, D) = (1, 2)]

[∵ In C2 : (A, E) = (1, 2) and C1 : (A, E) = (1, 3)] ∴ b = 5.

$R = (a+b) / {}^nC_2 = (1+5)/10 = 6/10$

- Accuracy = TP + TN / TP + TN + FP + FN
- Precision = TP / TP + FP
- Sensitivity / Recall = TP / TP + FN
- Specificity = TN / FP + TN
- F1 score : Avg = (P+R)/2 , F1 score = 2 (P·R) / P+R

high bias = linear regres, Logistic regres.

low var ↳ same

high var = Decision Tree, KNN, SVM.