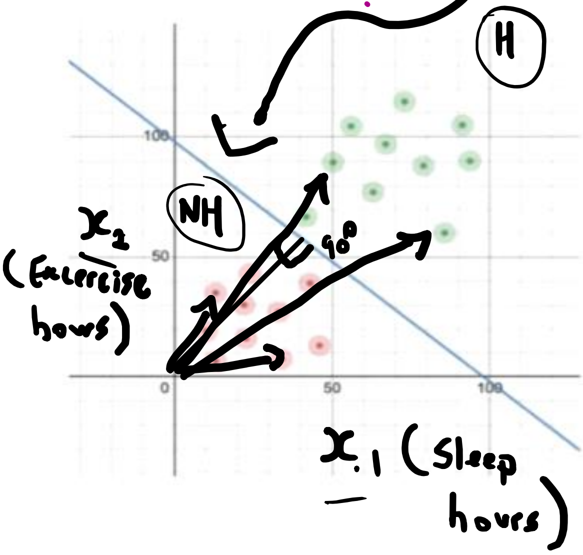
Recap

27 October 2025 18:37

$$\left[\left(x_1 + x_2 - 100 = 0\right)\right]$$



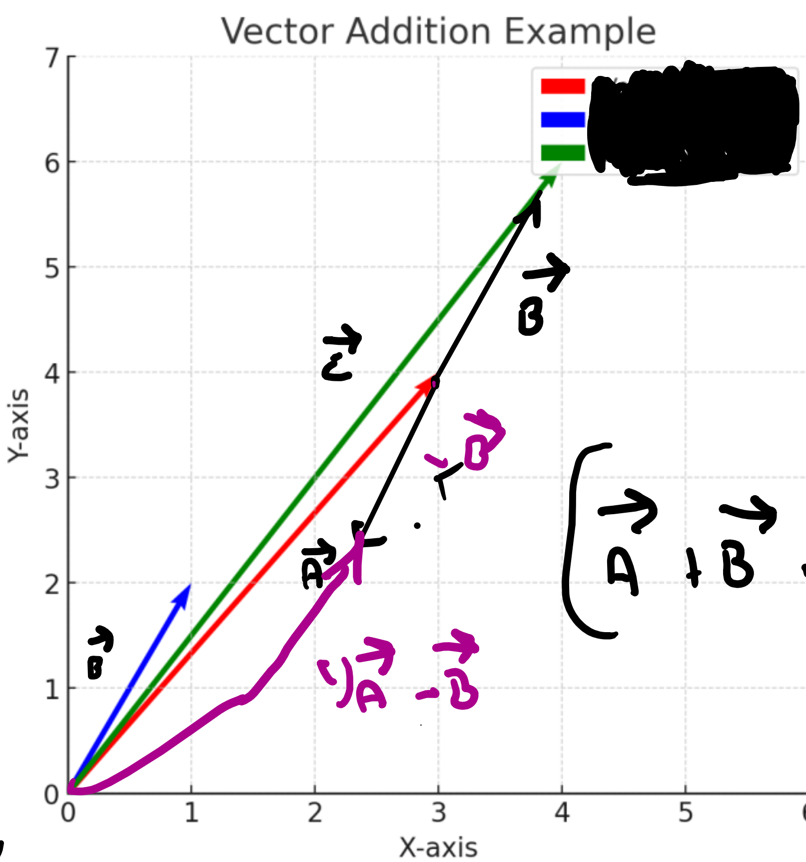
$x_1 x_2 \to$ features.

$$(w_1x_1 + w_2x_2 + w_0 = 0)$$

$$1 \qquad 1 \qquad -100$$

$$w^t x + w_0 = 0$$

$$\underbrace{[1, 1]}_{\text{weight vector}} \underbrace{\begin{bmatrix}x_1\\x_2\end{bmatrix}}_{\text{feature vectors.}} - 100 = 0$$



$$\left[\vec{A} + \vec{B} = \vec{C}\right]$$

[Shortest distance is always perpendicular !]

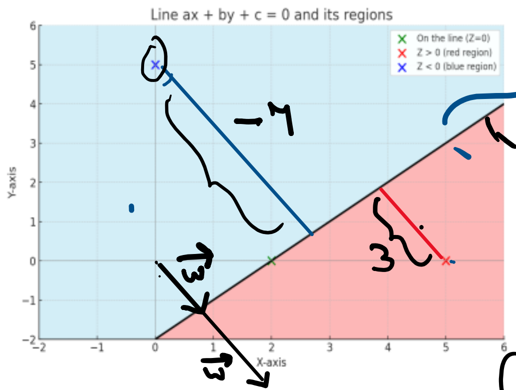
$> 0 \Rightarrow$ +ve H.F
$< 0 \Rightarrow$ neg

We know $\{w_1x_1 + w_2x_2 + w_0\}$ tells us the [location of a point (x_1, x_2)] based on

$>, <, = 0$

$= 0$ ⇓ on the line

$x_1 - x_2 - 2 = 0$



→ Hypotneus.

$x_1 - x_2 - 2 = 0 \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

$2x_1 - 2x_2 - 4 = 0 \quad \frac{2}{-2}$

$\frac{2(x_1 - x_2 - 4)}{2} = \frac{0}{2}$

$(x_1 - x_2 - 2 = 0)$

for $(5, 0)$

$x_1 - x_2 - 2 = 3$

for $(0, 5)$

$x_1 - x_2 - 2 = -7$

} ⇒ Signed distance

[In a way, does this represent distance ?]

How can I get rid of the sign ?

$\| w_1x_1 + w_2x_2 + w_0 \| \Rightarrow$ distance ? ✗

This still does not represent Shortest (perpendicular)

This still does not represent shortest (perpendicular) distance. Why?

$$[x_1 - x_2 - 2 = 0]$$

Weight vector is what?

$$W = [1, -1]$$

Let me multiply the equation by 2 (scale up weight vector)

$$2(x_1 - x_2 - 2) = 0$$

$$[(2x_1 - 2x_2 - 4 = 0)]$$

$$\left[\begin{aligned} &2x_1 - 2x_2 - 4 = 0 \\ &2(x_1 - x_2 - 2) = 0 \\ &x_1 - x_2 - 2 = 0/2 \\ &x_1 - x_2 - 2 = 0 \end{aligned}\right]$$

Value for $(\overset{x_1}{5}, \overset{x_2}{0})$

$$(10 - 0 - 4 = 6)$$

$$W = \begin{bmatrix} 2 \\ -2 \end{bmatrix}$$

Weight vector scaled up by 2

But wait, aren't $x_1 - x_2 - 2 = 0$ and $2x_1 - 2x_2 - 4 = 0$ the same boundaries?

$$w^T x + w_0$$

We still got different distances!!

So is it the true distance?

$\|W^T x + W_0\|$

Why did we get different distances? (Remove impact of Magnitude of weight vector)

$\|W^T x + W_0\|$ gives us Unnormalized distance.

We need to normalize this distance such that

[Scale of W is ignored.] How can we do this?

$\Downarrow$

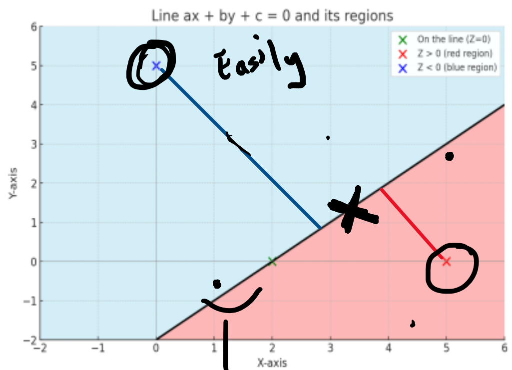
Magnitude

Divide by the Magnitude of weight vector.

$$\left(\frac{\|W^T x + W_0\|}{\|W\|}\right) = \frac{|W^T x + W_0|}{\sqrt{w_1^2 + w_2^2}}$$ ✓

[True perpendicular distance] ignoring the magnitude of weight vector

Why is this so important?



→ Confusion

[In ML, the perpendicular distance represents the confidence score!]

[Logistic Regression
SVM]

[Feature vector]

[feature vector] ⌊ SVM ⌋

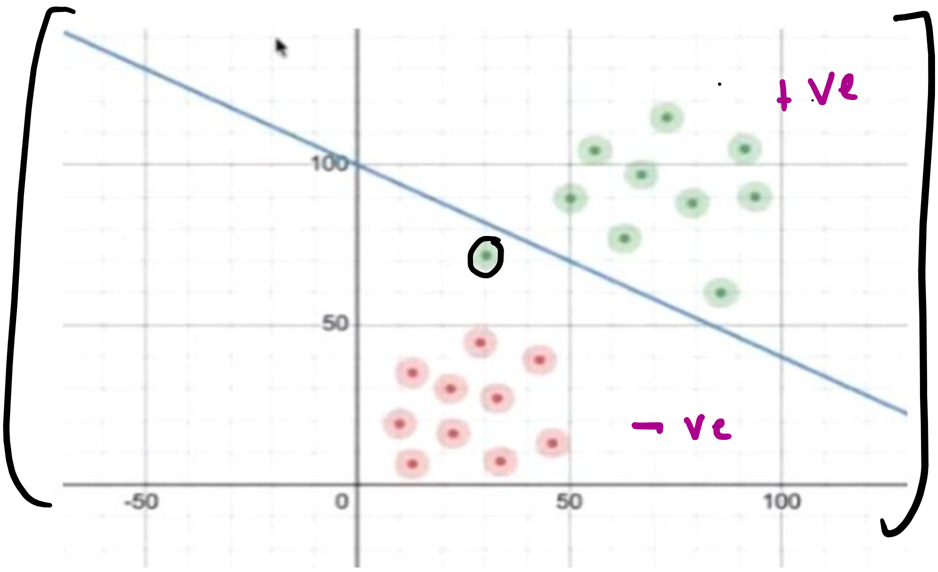
Introduction to loss function - misclassification

23 August 2025 18:35

Features

Target

Age (x_1)	Daily Exercise (mins) (x_2)	Health Status (y)	Model Prediction (Predicted y)
25	60	+1	+1
30	30	+1	+1
45	10	-1	-1
50	5	-1	-1
28	20	+1	-1
35	50	+1	+1
40	15	-1	-1



⇒ Correct the Misclassification

I train a Model on the points above, lets say I get the boundary as shown. What is wrong?

−ve

Actual label : +ve
Assigned label = -ve

Will the Model know it has Made a Mistake ?

In this case, according to our dataset (historical data)
We know label should be +1, but
Model has predicted -1.

So, how should the Model go about correcting its own Mistake?

The model will know

The model will know

(x_1, x_2, y_1) ✓

$\Downarrow$

$\boxed{\text{Input}}$

$\Downarrow$

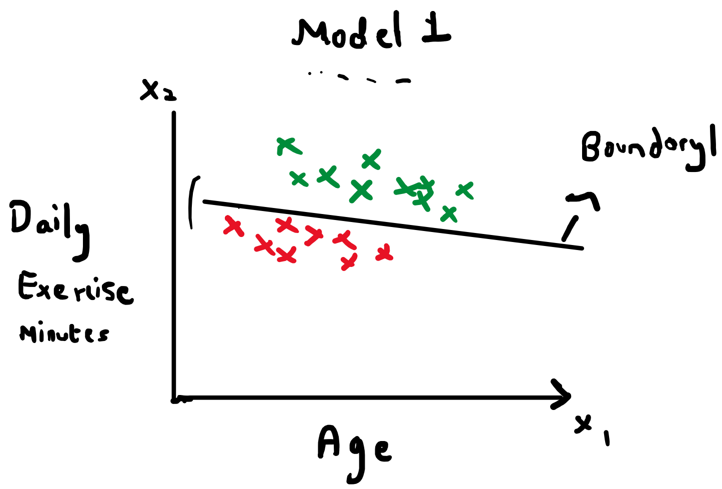
$\boxed{\text{output.}}$

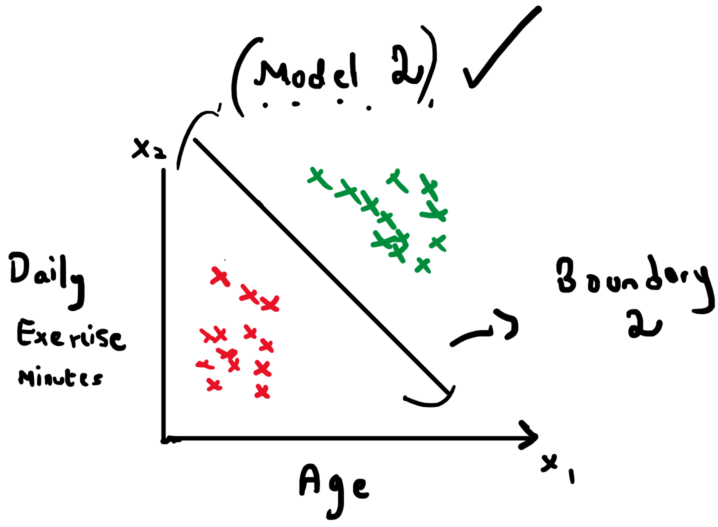
$\Downarrow$

$$[w_1x_1 + w_2x_2 + w_0] = y_{pred}$$

Introduction to loss function - role of distance

23 August 2025 18:52





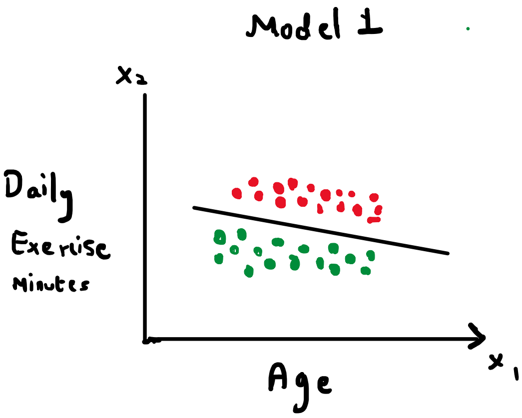
Model output => boundary

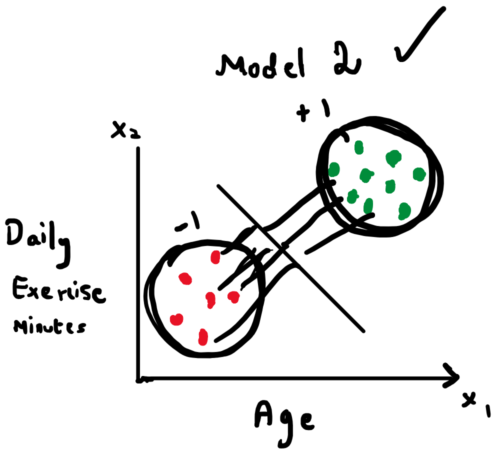
Which of the above is the better Model ?

Model 2

Why ?

Higher distance





Which of the two is a better model? Model 2

Should I Maximize or Minimize distance?

Let us distance of points from boundary are given by $d_1, d_2, d_3 \cdots$ etc. How can I mathematically define "Maximize distance"?

$$\sum \left[\frac{w^t x + w_0}{\|w\|} \right]$$

one problem!

(prediction) ✓

(Actual) ✓

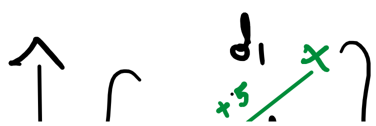
$$\text{Max} \left(\{ d_1 + d_2 + d_3 + d_4 + d_5 + d_6 \cdots\cdots d_n \} \right)$$

$\Downarrow$

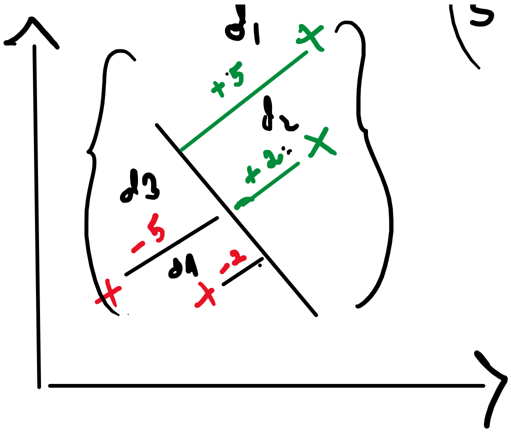
(n points) Sum of distance b/w point and boundary for all individual points in my data

$$\left\{ d = \frac{w_1 x_1 + w_2 x_2 + w_0}{\sqrt{w_1^2 + w_2^2}} \right\} \Rightarrow$$ I want sum of this for all points

But without the mod $|\cdot|$ what is the problem?



$$(5 + 2 + -5 - 2) = 0$$



$$= 0$$

What happens if I sum the above 4 distances?

$$d = \frac{\left| w_1 x_1 + w_2 x_2 + w_0 \right|}{\sqrt{w_1^2 + w_2^2}}$$

$\Rightarrow$ What exactly does this tell me?

I can add Mod or square the distances. But does

$$d = \frac{w_1 x_1 + w_2 x_2 + w_0}{\sqrt{w_1^2 + w_2^2}}$$

tell us anything about actuals (y_i)?

I want an equation for distance that:-

i) Ensure positive/negative distances don't cancel out. ✓

ii) Actuals are also part of the equation, so the model

ii) Actuals are also part of the equation, so the Model "knows" what it is doing.

+ve

$\left(\frac{w^T x + w_0}{\|w\|}\right)$ Signed Prediction distance	Actual label (y)	
-5.2	-1	+5.2
-3.7	-1	+3.7
-1.0	-1	
0.5	+1	
2.3	+1	
4.1	+1	
6.8	+1	
-7.4	-1	

Note :- This is not predicted label.

$$\left(\sum \left(\frac{(w^T x + w_0)}{\|w\|} \right) y_i \right)$$

Solution :- distance × label

distance	label	positive_distance
-5.2	-1	5.2
-3.7	-1	3.7
-1.0	-1	1.0
0.5	+1	0.5
2.3	+1	2.3
4.1	+1	4.1
6.8	+1	6.8
-7.4	-1	7.4

Now summing them up using summation function

$$\left[\left\{ \sum_{i=0}^{n} d = \left(\sum_{i=1}^{n} \left(\frac{\bar{w}^T x_i + w_0}{\|w\|}\right) y_i \right) \right\} \right]$$

distance

label / Actual y or target

$\sum$ distance × y_i [Gain function]

[Gain function]

We call this the gain function.

(3) Sum of all distances b/w boundary and points in our data

$$\left\{ G(X, \bar{w}, w_0) = \sum_{i=1}^{n} \left(\frac{\bar{w}^T \bar{x}_i + w_0}{\|\bar{w}\|} \right) y_i \right\}$$

In order to find out the best classifier we need to maximize the gain function G

(Maximize function) $\Rightarrow$ Better Model.

Should we Maximize this or Minimize this?

But in ML, we dont work on Maximizing.

We want to Minimize functions.

How can I convert Max $\sum \frac{(w^T x + w_0) y_i}{\|w\|}$ into a Minimization problem?

Max $\left[\left(\underbrace{\sum_{i=1}^{n} \left(\frac{\bar{w}^T x_i + w_0}{||w||}\right) y_i} \right) \right]$ Max Gain

Max Dis is same as Min - Dist

(Min ?)

Min (- Gain)

↓ Min $\left(\underbrace{- \sum_{i=1}^{n} \left(\frac{\bar{w}^T x_i + w_0}{||w||}\right) y_i} \right)$

(Loss Function)

[Min loss function] => Mathematically the rules to minimize functions are more well defined.

[Convex optimization]

Loss Fn

-100 ✓

-50

d d

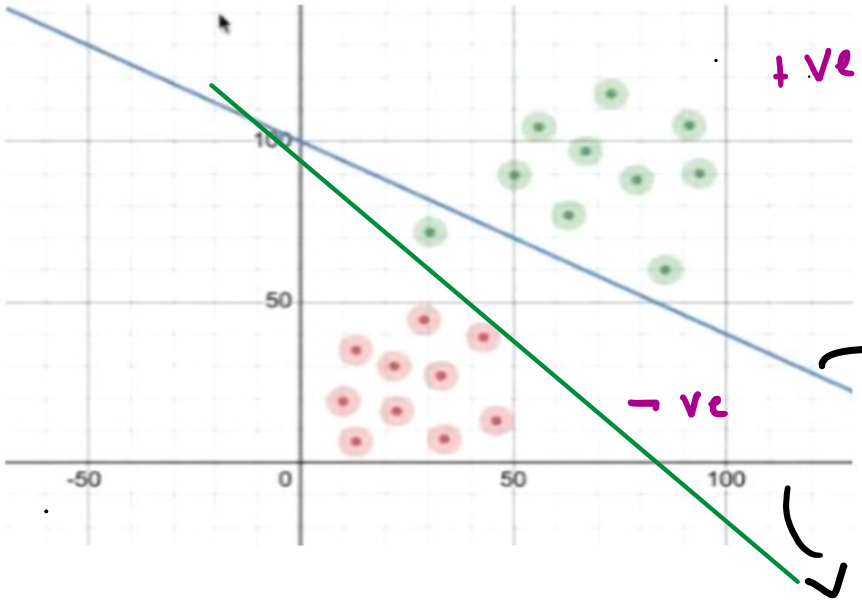
Gain Fn => 100 50

?

[

Goal of minimizing loss function

27 October 2025 22:19



Minimize loss Fn ?

→ B1

B2 ✓

↳ lesser loss

How?

Loss fn ✓

[Convex Optimization] ✓

Today ML, DL, LLMS,
GenAI
Image Gen, tex Gen,
Simple Model

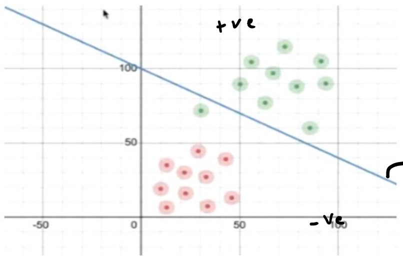
Simple Model
to
predict diabetes

[1940s 1950s]

[Perceptron Algorithm]

⇓

[Forefather of ML / DL]



→ Boundary

1

In the above scenario, in order to correct the Misclassification, should we Move the point or the boundary?

Weights

Boundary $\quad W_1 x_1 + W_2 x_2 + W_0 = 0$

$(W^T x + W_0 = 0)$

We need to change the boundary! This means, We need to Modify our equation

$\{W_1 x_1 + W_2 x_2 + W_0 = 0\} \Rightarrow w^T x + w_0 = 0$

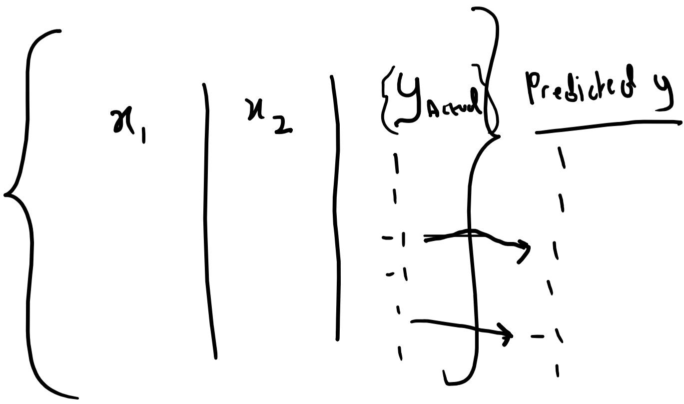
This involves modifying W_1, W_2, W_0 => Weights

(Modifying Means Updating.)

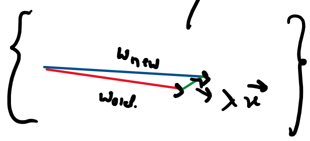
{ML lingo => ([Updating Weights]) ✡ ✡}

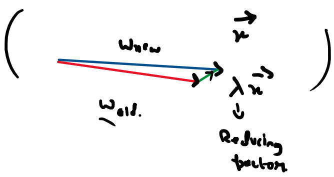
[Optimal Weights / Boundary]

x



$x \rightarrow [W_{new} = W_{old} + \lambda x]$



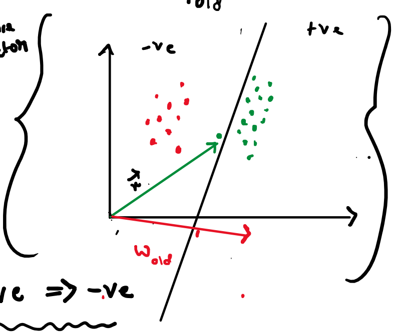


Is there a Misclassification below?

Now?

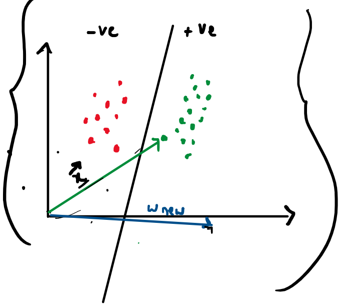
$\vec{x}$ ⇓ feature vector

M_{old}



+ve => -ve

M_{new} ✓



Blue => W_{new}

Red => W_{old}

$W_{old} + \lambda x = W_{new}$

$[W_{new} = W_{old} + \lambda x]$

where λ is a small no. (Scaling factor)

$(+ \to -)$

1 $[W_{new} = W_{old} + \lambda x_i]$ $\Rightarrow$ Boundary moves

if Actual = +1
But predicted as −1

$W_{new} = W_{old} + \lambda x_i$

$(- \to +)$

$W_{new} = W_{old} - \lambda x_i$

into 1

y_i

Similarly, if Actual = −1 $\Rightarrow$ other way round
and predicted = +1

2 $W_{new} = W_{old} - \lambda x_i$ Boundary moves clockwise

Instead of having two different update equations, can we combine both to one?

$$[W_{new} = W_{old} + y_i \lambda x_i] \Rightarrow \text{General perceptron update algorithm}$$

Where $y_i \Rightarrow$ Actual

if $y_i = +1 \Rightarrow W_{new} = W_{old} + \lambda x_i$

if $y_i = -1 \Rightarrow W_{new} = W_{old} - \lambda x_i$

0.004 0.003

$$[W_{new} = W_{old} + y_i \lambda x_i]$$

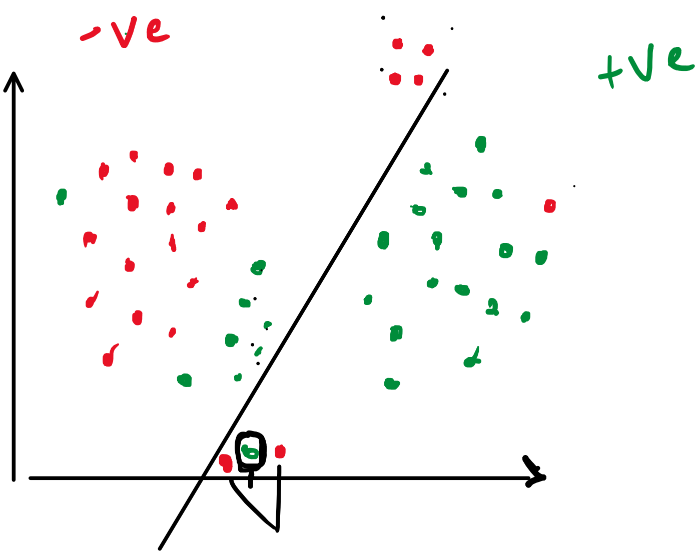
$[\lambda \Rightarrow$ Learning rate$]$ (0.01)

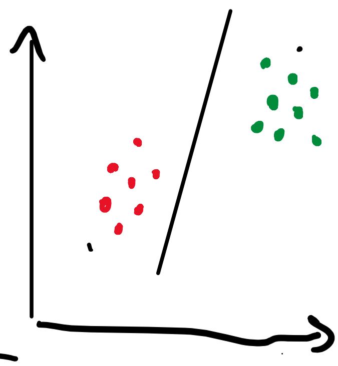
$\Downarrow$ [0.01 or 0.001]

More details on Perceptron

27 August 2025 17:53

[Simple perceptron Algorithm]

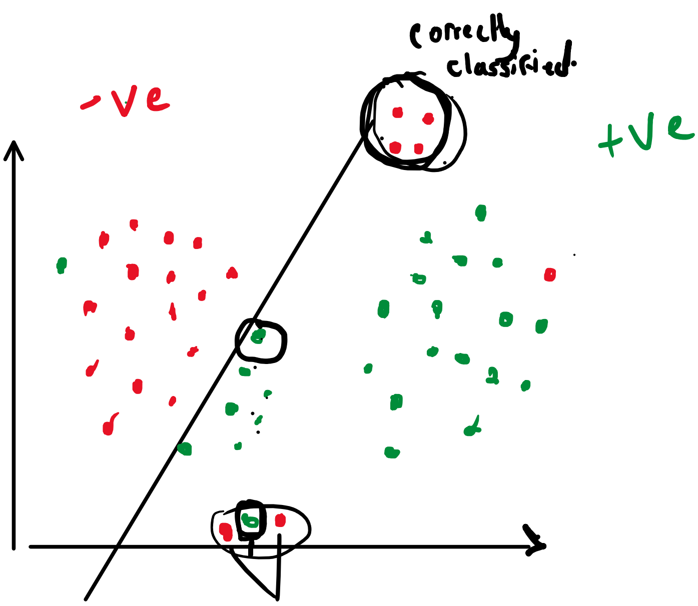




[Linearly Seperable data]

i) What boundary to start with?

[Random]



Not a perfect solution !!

[Visiting data once → 1 epoch]

Visiting data multiple times

↓

Multiple epoch

[epochs / iterations]

(10, 50, 100) ✓ ✓ ✓

[Each epoch
↓
visit every datapoint]

Interview
⇓

i) Multiple epochs doesnt guarantee good result

ii) Works perfectly only for linear data

iii) Requires lot of experimentation

iii) Requires lot of experimentation

↳ $\underbrace{0.1 \quad 0.01 \quad 0.001}_{L \cdot R\ (\lambda)}$ $\quad$ epoch

10 20 30.

[Convex Optimization]