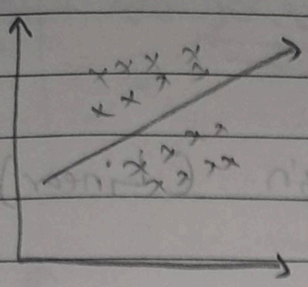
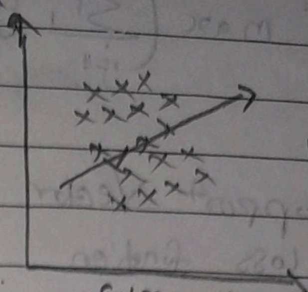
* Support Vector Machine

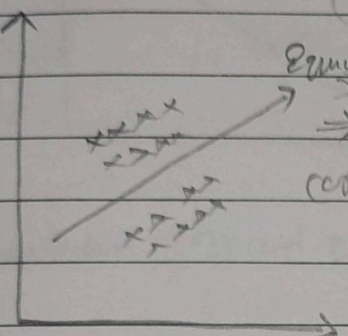


linear Seprable



Non-Linear Seprable

SVM → Kernal trick ← (we can solve this problem using)



Equation for thise line

$\rightarrow y = mx + C \longrightarrow y = m_1x_1 + m_2x_2 + m_3x_3 \ldots$

(or)

$y = w_0 x + C$

{way of representation of above Eq}

$m^T x \rightarrow [\ ]^T [\ ]$ matrix

* (norms) ⇒ (distance between) {Point and Plane}

#

{norms for distance between point and plane}

$\text{Norms} = \dfrac{mx + C}{\|m\|}$ ← magnitude

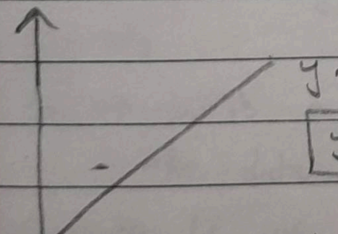
distance → $\|m\|$ ← (vector) value + direction

between point and plane

(or) ⇒ $\text{Norms} = \dfrac{W^T x + C}{\|w\|}$

$\frac{1}{0} = 0$ not

$\frac{1}{0}$ not = 0



$y = mx + C$ ($C = 0$)

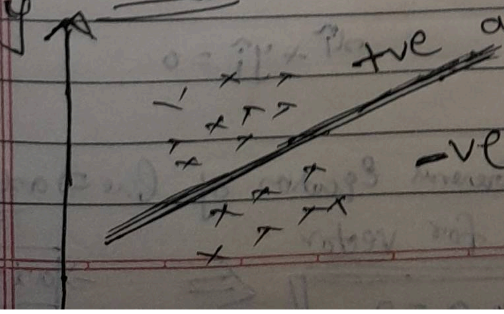
$\boxed{y = mx}$

(0,0)

$\text{norms} = \dfrac{w^T x}{\|w\|}$ ← unit Vector $\boxed{\text{Value} = 1}$

$\boxed{\text{Norms} = w^T x \Rightarrow mx}$

* Binary class



+ve above the line ⇒ +ve

−ve below the line −ve

$$\max\left(\sum_{i=1}^{n} y_i \times w^T x_i\right) \leftarrow$$ ~~(loss or cost func)~~

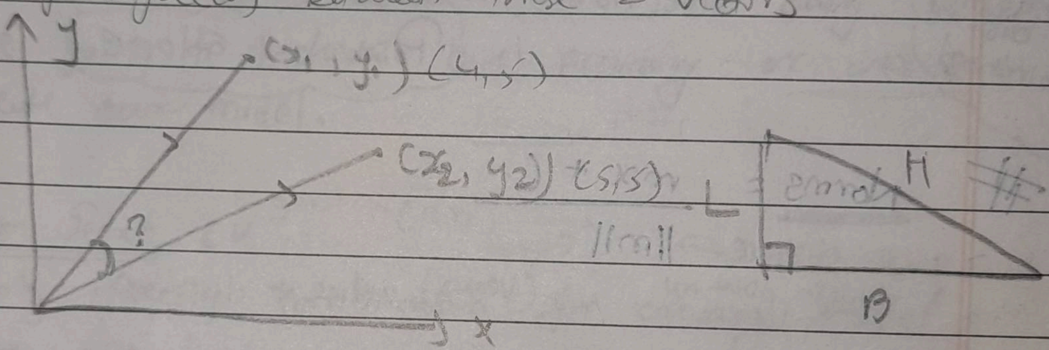
data point

* (1) Support Vector → max margin (Linear)
  (2) Loss function

* Non-Linear Seprable Data (NLSD)
  (1) Polynomial } Kernel trick
  (2) RBF

SVM notebook ⇒ MLDepiVector

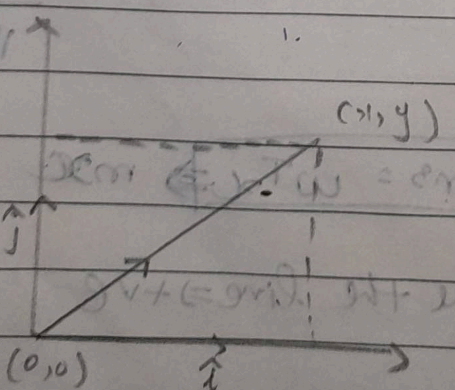
* # Calculating angle(θ) between these 2 vectors



$\tan\theta = \frac{L}{B}$

$$\boxed{\theta = \tan^{-1}\frac{L}{B}}$$

*



Vector = magnitude + direction

$x\hat{i} + y\hat{j} = 0$

If the line going through origin the c will be 0

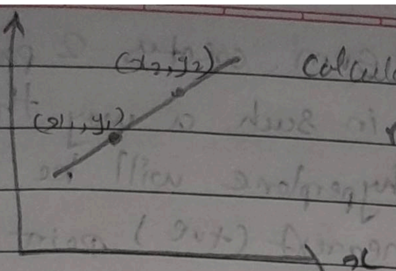
General Equation of line ⇒ $ax + by + c = 0$

Equation for vector ⇐ → $\boxed{ax + by = 0}$

$a\hat{i} + b\hat{j} = 0$

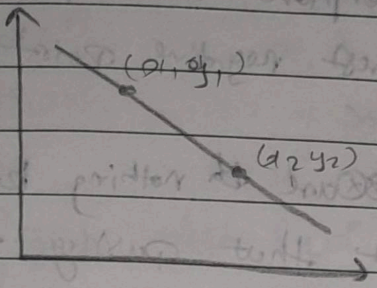
→ calculate slope (m)

$m = \frac{y_2 - y_1}{x_2 - x_1}$ (slope will be always +ve)

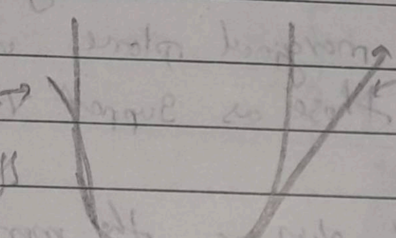


→ calculate slope (m)

$m = \frac{y_2 - y_1}{x_2 - x_1}$ (slope always −ve)



Gradient Decent curve =)



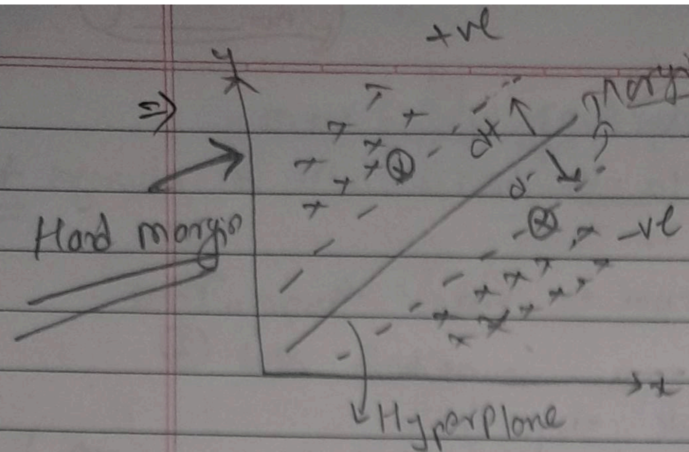
- Kernal trick
  ↳ we will convert low dimensions into higher dimensions
  this polynomial and RBF

Interview Question ⇒ What is Difference between SVC and SVM ?

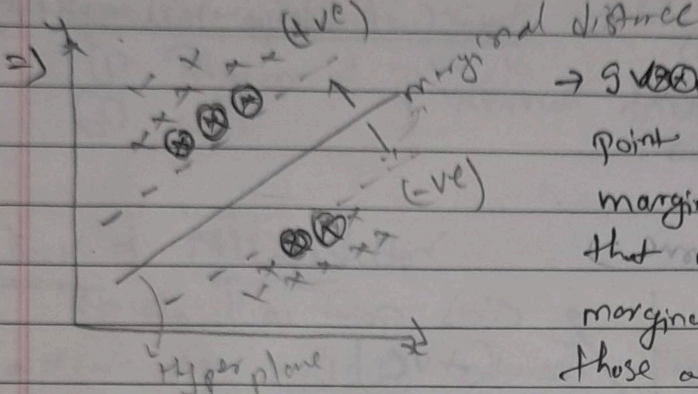
Ans ⇒

SVC → It is a linear classifier which is able to separate the data that is called support vector classifier

SVM → If we are going to transform our data then it is called SVM. There are three ways transforming the data = Polynomial, RBF

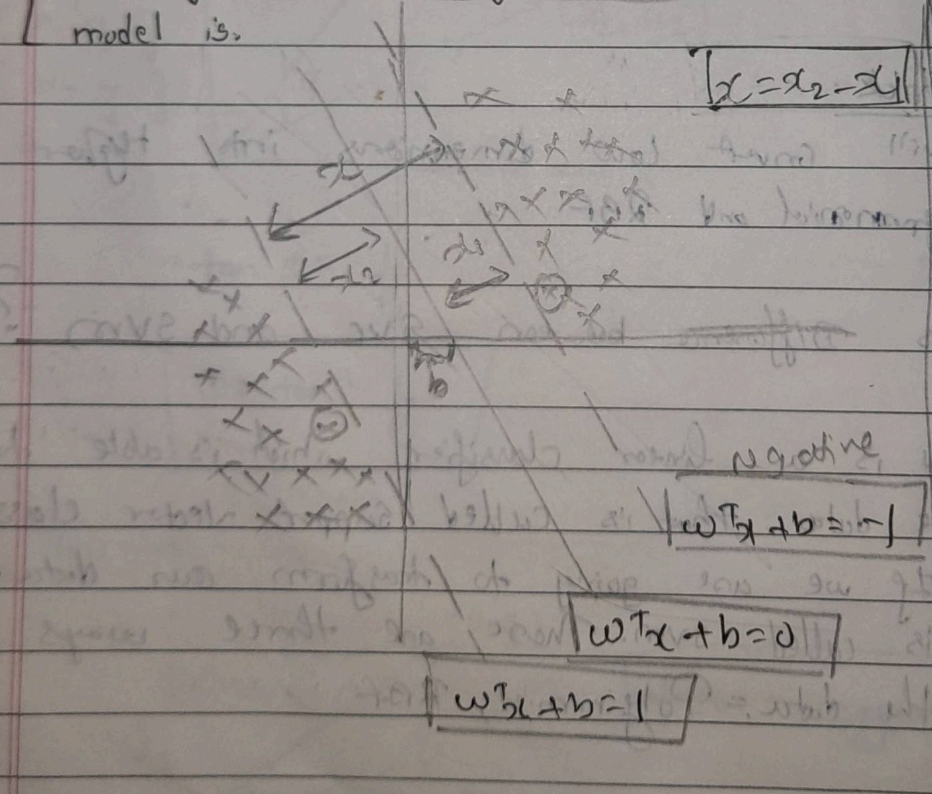


=> svm creates 2 parallel hyperplanes in such a way that one of the hyperplane will be passing to the nearest (+ve) point and the other hyperplane will be passing to the nearest negative point.

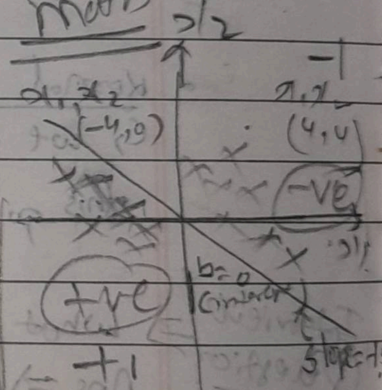


→ SV are nothing but they are the point that passing through the marginal plane, Any no of points that are passing through this particular marginal plane we will consider those as support vector

Note:- { The higher the marginal distance the more generalized our model is.



Maths



c or b = 0

$y = w^T x + 0$

$y = w^T x$

$= \begin{bmatrix} -1 \\ 0 \end{bmatrix} [-4 \ 0]$

$= 4$ // going to be always +ve

→ In order to compute the distance between two marginal distance.

$y = w^T x + 0$

$= \begin{bmatrix} -1 \\ 0 \end{bmatrix} [4 \ 4]$

$= -4$ // (-ve)

Linear Algebra

$w^T x_1 + b = -1$

$(-)\ w^T x_2 + b = (-)\ 1$

$w^T (x_2 - x_1) = 2$

This is my optimization function and we need to maximize this

$\frac{w^T}{\|w\|} (x_2 - x_1) = \boxed{\frac{2}{\|w\|}}$ ↑↑↑

$\|w\|$ ← norm of w

update

~~Such that~~ (w^*, b^*) max $\frac{2}{\|w\|}$ → optimization function

y_i

such that

Condition $y_i \begin{cases} +1 & w^T x + b \geq 1 \\ -1 & w^T x + b \leq -1 \end{cases}$

we can also represent like this also

$y_i * w^T x_i + b_i \geq 1$

This we also called Regularization

Include 2 terminology to optimize my model better

$\Rightarrow (w^*, b^*) = \min \frac{\|w\|}{2} + C_i \sum_{i=1}^{n} \xi_i$ (zeta)

How many errors my model can consider? (misclassification data)

Value of the error (whatever distance i am getting w.r.t. this error i am just going to do summation)

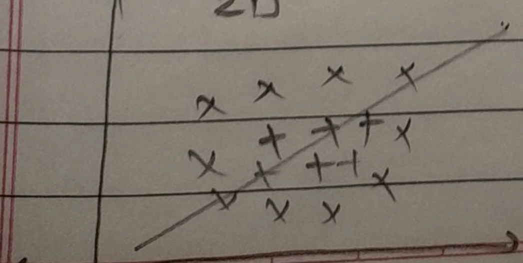
* SVM Kernels ⟺ # convert into 2 Dim into 3 Dim

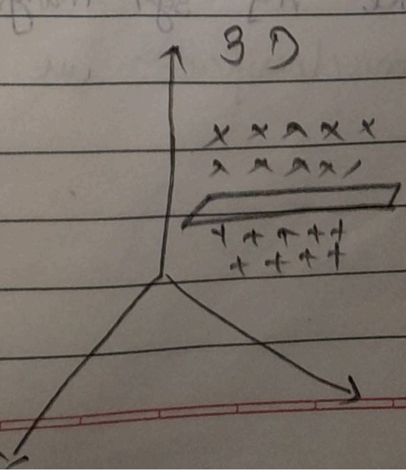
→ (1) Polynomial Kernel

(2) RBF Kernel

(3) Sigmoid Kernel

Soft margin



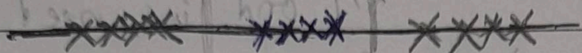


Page:

Trasformation can be done using {mathematical ~~equation~~ formula}

→ 1 dimension

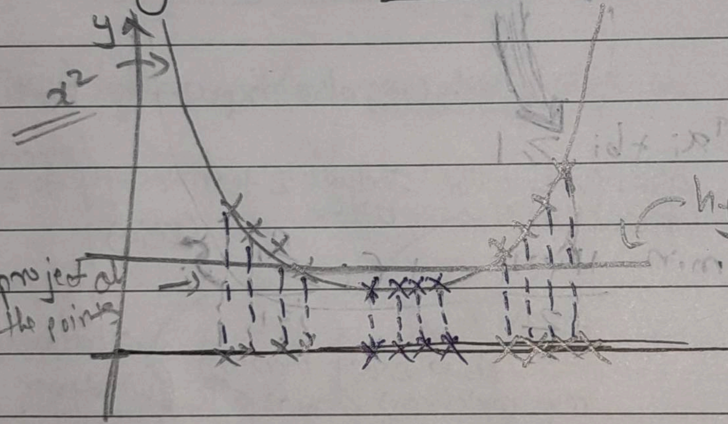
f_1



Note:- If i want to classify these two points, i can create one hyperplane over here but the "error will be v.v. high" in order to classify this particular thing

Convert :- 1d $\xrightarrow{to}$ 2d Then probably i will able to probably draw a hyperplane and try to split this particular data

Considering Equation, $y = f(x)$ this function in nothing but x^2



Types of Margin
↳ ① Soft Margin
↳ ② Hard Margin

Soft → In Soft Margin there will be some percentage of error. Suppose my soft margin has an error of C ; $C=3$ (3 misclassification we can consider)

Polynomial Kernel

Independent — x_1 x_1 — Dependent — y

this can we write as

$\therefore y = f(x_1, x_2)$

$f(x_1, x_2) = (x_1^T \cdot x_2 + 1)^d$ ← dimensions

doing Traspose operation — vector operation — identity matrix

$$= \begin{bmatrix} \vec{x_1} \\ x_2 \end{bmatrix} \cdot \begin{bmatrix} x_1 & x_2 \end{bmatrix}$$

#

$$= \begin{bmatrix} x_1^2 & x_1 x_2 \\ x_1 x_2 & x_2^2 \end{bmatrix}$$

← from here we will take unique elements

This two Dim (x_1, x_2) now are becoming 5 dimension

x_1	x_2	y	x_1^2	x_2^2	$x_1 x_2$

This are my all the features now, which i will probably using In order to convert, this into a Higher Dim and create a plane line